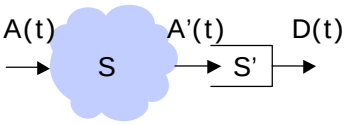
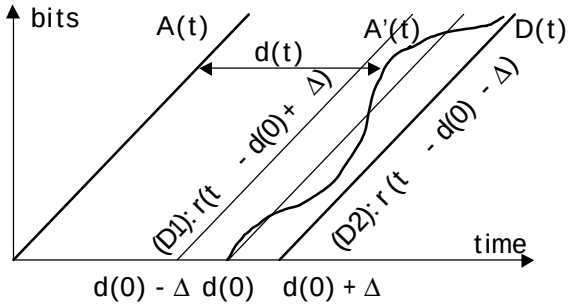
PERFORMANCE EVALUATION
OF
COMPUTER AND COMMUNICATION SYSTEMS

Jean-Yves Le Boudec
EPFL





Version $2.1.\beta$ of May 3, 2007
Available at `http://perfeval.epfl.ch`

Summary of Changes

2004 Feb Misc Bug fixes
2004 Sept Redesigned Format; solutions to inline questions are in footnotes instead of at the end.
2004 Nov 22 Beta Version 2: redesigned book entirely. Goal of version 2 is to (1) remove close dependency on external textbooks and (2) simplify what can be simplified. Done for chapters 1, 2 and 3 (version 1) that now become chapters 1 and 2. Added confidence intervals for quantiles. Added appendix with tables for quantiles.
2005 March 4 Bug fixes
2006 March 31 Miscellaneous bug fixes
2005 April 2 Modified chapter on tests to make it more self-contained
2006 April 21 Modified Factorial Analysis (bug fixes, removed the non orthogonal case).
2007 February 26 Miscellaneous bug fixes; Drafted model fitting section.
2007 April 19 Redesigned model fitting chapter, as replacement of “Factors”
2007 April 25 Chapter “Tests”: Extended tests of common variance to more than 2 samples.

I am very grateful to Anthony Davison, of EPFL for allowing me to access a beta version of his book “Statistical Models”. I thank Richard Weber, of Cambridge University and Chadi Barakat, of INRIA, who made their lecture notes freely available on the web and allowed me to use them as constituent material in this course. I am grateful to François Baccelli and Pierre Brémaud who helped me get some understanding of their fields. Many thanks go to to Mourad Kara for discussions and inputs, Božidar Radunović, Ruben Merz, Slaviša Sarafijanović, Milan Vojnović, Manuel Flury, Assane Gueye and Olivier Gallay for the final preparation of the manuscript, associated exercises and some figures.

Contents

Index

PREFACE

Performance Evaluation is often the critical part in evaluating the results of a research project. Many of us are familiar with simulations, but it is often difficult to address questions like: Should I eliminate the beginning of the simulation in order to wait until the system stabilizes ? I simulate a random way point model but the average speed in my simulation is not as expected. What happened ? The reviewers of my study complained that I did not provide confidence intervals. What is that ? How do I get them ?

This book is the set of lecture notes for a course given at EPFL. With this book and some accompanying practicals, you will be able to answer these and other questions, more generally, to evaluate the performance of computer and communication systems and master the theoretical foundations of performance evaluation and of the corresponding software packages.

In the past, many textbooks on performance evaluation have given the impression that this is a complex field, with lots of baroque queuing theory excursions, which can be exercised only by performance evaluation experts. It does not have to be so. In contrast, performance evaluation can and should be performed by any computer engineering specialist who designs a system. When a plumber installs pipes in our house, one expects her to properly size their diameters; the same holds for computer engineers.

This book is not for the performance evaluation specialist. It is for **every computer engineer or scientist** who is active in the development or operation of software or hardware systems. The **required background** is an elementary course in probability and one in calculus.

The first objective of the book is to make performance evaluation usable by all computer engineers and scientists. The foundations of performance evaluation are in statistics and queuing theory, therefore, *some* mathematics is involved and the text cannot be overly simplified. However, it turns out that much of the complications are not in the general theories, but in the exact solution of specific models. For example, some textbooks on statistics (but none of the ones I cite in the reference list) develop various solution techniques for specific models, the vast majority of which are encapsulated in commercially or freely available software packages like Matlab, S-PLUS, Excel, Scilab or R.

To avoid this pitfall, I focused first on the **what** before the **how**. Indeed, the most difficult question in a performance analysis is often “what to do”; once you know what to do, it is less difficult to find a way with your usual software tools or by shopping the web. For example, what do we do when we fit a model to data using least square fitting (the answer is in Chapter 4) ? I also looked for solution methods that are universal, i.e., that apply in all situations, simple or complex. For

example, computing confidence or prediction intervals can be made simple and systematic is we use the median and not the mean; if we have to use the mean, the use of likelihood ratio statistic is quite universal and requires little intellectual sophistication about the model. I give a coverage of queuing theory that focuses on universal laws and patterns rather than the solution of specific queuing networks. During a performance analysis, one is often confronted with the dilemma: should I use an approximate model for which exact solutions exist, or should I use approximate solutions for a more exact model ? I took the second option as much as possible. A benefit is to use methods that apply practically always, instead of dwelling on the meanders of explicit, exact closed forms that apply only with unrealistic, restrictive assumptions.

Part I is a self-contained first course that addresses the first objective. It contains all the material needed by an engineer who wishes to evaluate the performance of a computer or communication system. Chapter 1 gives a methodology and serves as introduction to the rest of the book. Chapter 2 describes how to quantify the accuracy of results. In Chapter 4 we present a general method for fitting an explanatory model to data. Chapter 3 discusses simulation and its application to performance evaluation. Chapter 5 describes performance patterns, i.e., facts that repeatedly appear in various situations, and whose knowledge considerably helps the performance evaluation. Chapter 6 discusses patterns specific to queuing.

A second objective is to introduce the computer engineer to more **specialized topics**, that are not more complex, but whose applicability is restricted to more specific areas. This is covered by **Part II**. Chapter 7 describes the techniques of tests. Chapter 8 discusses the background needed for load generation. Chapter 9 describes the techniques used for forecasting the load intensity. Chapter 10 describes the concepts of long range dependence, a feature found in most traffic traces. Last, Chapter 11 describes Palm calculus, which relates the different viewpoints resulting from measurements done by different operators. This is generally considered too complicated for applied textbooks, but, here too, I found that it is possible to convey the main ideas and results in a simple, accessible way.

A typical course for computer engineers would consist of Part I and, depending on the focus of the students, a few selected topics from Part II. Sections marked with a $\star$ can be omitted or skimmed, depending on the reader's inclination. This applies to both Parts I and II. Text in small font size can be skipped at first reading.

Performance evaluation is primarily an art, and involves using sophisticated tools such as mathematical packages, measurement tools and simulation tools. See the web site of the EPFL lecture on Performance Evaluation for some examples of **practicals** designed around this book.

The text is intended for **self-study**. It contains many **inline questions**; I invite the alert readers to try and answer the questions as they read.

QUESTION 0.0.1. *Where is the answer to an inline question ?* [1]

Every chapter contains a **review section** that summarizes the main points and also contains further inline questions. The **exercise section** can be used as assignments in a lecture. The solutions are available on request; if time permits, a solution manual will eventually be available. The **Index** collects all terms and expressions that are highlighted in the text like *this* and also serves as a notation list. An appendix gives background material on probability and calculus.

[1] In a footnote on the same page

PART I

A FIRST COURSE IN PERFORMANCE EVALUATION

CHAPTER 1

METHODOLOGY

Contents

1.1 WHAT IS PERFORMANCE EVALUATION ?

In the context of this book, performance evaluation is about quantifying the service delivered by a computer or communication system. For example, we might be interested in knowing the response time experienced by a customer performing a reservation over the Internet; or we might be interested in comparing two compilers for a multiprocessor machine.

The *performance metric* is a measurable quantity that precisely captures what we want to measure – it can take many forms. There is no general definition of a performance metric: it is system dependent, and its definition requires understanding the system and its users well. We will often mention examples where the metric is throughput (number of tasks completed per time unit) or response time (time elapsed between a start and an end events). For each performance metric, we may be interested in average, 95-percentile, worst-case, etc. We discuss this point in detail in Chapter 2.

A particular feature of computer or communication systems is that their performance depends dramatically on the *workload* (or simply *load*) they are subjected to. The load characterizes the quantity and the nature of requests submitted to the system. Consider for example the problem of quantifying the performance of a web server. We could characterize the load by a simple concept such as the number of requests per second. This is called the intensity of the workload. In general, the performance deteriorates when the intensity increases, but often the deterioration is sudden; this is due to the non-linearity of queuing systems – an example of *performance pattern* that is discussed in Section 1.3 and Chapter 6. The performance of a system depends not only on the intensity of of the workload, but also its nature; for example, on a web server, all requests are not equivalent: some web server softwares might perform well with *get* requests for frequently used objects, and less well with requests that require database access, for some others it might be different. This is addressed by using standardized mixes of web server requests. They are generated by a *benchmark*, defined as a load generation process that intends to mimic a typical user behaviour. In Chapter 8 we study how such a benchmark can be constructed.

EXAMPLE 1.1: QUIZ. Consider the following cases and answer the next question.

1. Design web server code that is efficient and fast.
2. Compare TCP-SACK versus TCP-new Reno for hand-held mobile devices.
3. Compare Windows 2000 Professional versus Linux.
4. Design a rate control for an internet audio application.
5. Compare various wireless MAC protocols.
6. Say how many servers a video on demand company needs to install.
7. Compare various compilers.
8. How many control processor blades should this Cisco router have ?
9. Compare various consensus algorithms.
10. Design bug-free code.
11. Design a server farm that will not crash when the load is high.
12. Design call center software that generates guaranteed revenue.
13. Size a hospital's information system.
14. What capacity is needed on an international data link ?
15. How many new servers, if any, should I install next quarter for my business application ?

QUESTION 1.1.1. *Say which examples require a detailed identification of the intensity of the workload.* [1]

If you score more than 12 correct answers, then proceed with this course. Else, go back to the beginning of the lecture.

EXAMPLE 1.2: Consider the following performance evaluation results:

(A1) PC configuration 1 is 25% faster than PC configuration 2 when running Excel

[1] Examples 6, 8,13, 14, 15.

(A2) For your video on demand application, the number of required servers is 35, and the number of disk units is 68.
(A3) Using the new version of `sendfile()` increases the server throughput by 51%

QUESTION 1.1.2. *What is the difference between Examples (A1) to (A3) ?* [2]

The *goal* of a performance evaluation study is usually either a *comparison* of design alternatives i.e. quantify the improvement brought by a design option or *system dimensioning*, i.e. determining the size of all system components for a given planned utilization. Comparison of designs requires a well-defined load model; however, the exact value of its intensity does not have to be identified. In contrast, system dimensioning requires a detailed estimation of the load intensity. Like any prediction exercise, this is very hazardous. For any performance evaluation, it is important to know whether the results depend on a workload prediction or not. Forecasting techniques are the object of Chapter 9.

The benefit of a performance evaluation study has to be weighted against its cost and the cost of the system. In practice, detailed performance evaluations are done by product development units (system design). During system operation, it is not economical (except for huge systems such as public communication networks) to do so. Instead, manufacturers provide *engineering rules*, which capture the relation between load intensity and performance. Example (A2) above is probably best replaced by an engineering rule such as:

EXAMPLE 1.3: ENGINEERING RULE.

(E2) For your video on demand application, the number of required servers is given by $N_1 = \lceil \frac{R}{59.3} + \frac{B}{3.6} \rceil$ and the number of disk units by $N_2 = \lceil \frac{R}{19.0} + \frac{B}{2.4} \rceil$, where R [resp. B] is the number of residential [resp. business] customers.

In this lecture, we study the techniques of performance evaluation that apply to all these cases. However, how to implement a high performance system (for example: how to efficiently code a real time application in Linux) or how to design bug-free systems are *outside* the scope.

QUESTION 1.1.3. *Among the examples in Example 1.1 on page 4, say which ones fall within the scope of this lecture ?* [3]

1.2 HOW TO EVALUATE PERFORMANCE

The first step is to clearly define the **goal** of the performance evaluation, as discussed in the previous section. Once the goal is identified, it remains to define a **metric** and a **load** model. All of this requires knowing the system and its use.

[2] (A1), (A3) are about a comparison; (A2)is about dimensioning
[3] All except 1, 4, 10

EXAMPLE 1.4: WINDOWS VERSUS LINUX. Assume you want to compare Windows versus Linux. Chen and its co-authors did it in [Chen95-SOSP].

QUESTION 1.2.1. *What metric and load model would you use ?* [4]

The performance evaluation can then proceed with a solution method, which usually falls in one of the three cases below. Which method to use depends on the nature of the problem and the skills or taste of the evaluation team.

- **Measurement** of the real system. Like in physics, it is hard to measure without disturbing the system. Some special hardware devices (e.g.: optical splitters in network links) sometimes can prevent any disturbance. If, in contrast, measurements are taken by the system itself, the impact has to be analyzed carefully. Measurements are not always possible (eg. if the system does not exist yet). It sometimes requires a complex instrumentation.
- Discrete Event **Simulation**: a simplified model of the system and its load are implemented in software. Time is simulated and typically flows orders of magnitude more slowly than real time. The performance of interest is measured as on a real system, but measurement side-effects are usually not present. It is often easier than a measurement study, but not always. It is the most widespread method and is the object of Chapter 3.
- **Analytical**: A mathematical model of the system is analyzed numerically. This is viewed by some as a special form of simulation. It is often much quicker than simulation, but sometimes wild assumptions need to be made in order for the numerical procedures to be applicable. Analytical methods are often used to gain insight during a development phase, or also to learn fundamental facts about a system, which we call "patterns". The chapters in Part II make abundant use of analytical methods. We also show in Chapter 6 how some performance analyses can be solved approximately in a very simple way, using bottleneck analysis; see Section 6.5 for a example.

Further, one needs to establish a list of *factors*: these are elements in the system or the load that affect the performance. Ignoring some hidden factors may invalidate the result of the performance evaluation.

QUESTION 1.2.2. *Consider again comparing Windows versus Linux. Can you imagine what factors might play an important role in the analysis ? What external factors have to be taken care of during the evaluation ?* [5]

Knowing all factors is a tedious, but necessary task. This implies that you have to know your system well, or be assisted by people who know it well.

[4] Chen et al used the metric: number of cycles, instructions, data read/write operations. The load was generated by various benchmarks: "syscall" generates elementary operations (system calls); "memory read" generates references to an array; an application benchmark runs a popular application (here: ghostview).

[5] From [Chen95-SOSP]: External factors are: background activity; multiple users; network activity. These were reduced to a minimum by shutting the network down and allowing one single user. The different ways of handling idle periods in Windows NT and NetBSD also need to be accounted for, because they affect the interpretation of measurements. Cycle counts in idle periods of NetBSD have to be removed.

1.3 PERFORMANCE PATTERNS

Last, performance evaluation is simpler if the evaluator is aware of known performance patterns. Look at the top figure on the cover page: "We doubled the throughput, you'll be twice as fast". Does it make sense ? Is it a reasonable promise ? The behaviour of queuing systems follow some well known patterns. If we know them, we are likely to come more quickly to a conclusion. We discuss this example in Section 6.5. The prominent pattern in queuing is *bottlenecks*. In most systems of interest, the overall performance is dictated by the behaviour of the weakest components, called the bottlenecks.

EXAMPLE 1.5: BOTTLENECKS. You are asked to evaluate the performance of an information system. An application server can be compiled with two options, A and B. An experiments was done: ten test users (remote or local) measured the time to complete a complex transaction on four days. On day 1, option A is used; on day 2, option B is. The results are in Table 1.3. The expert concluded that the performance for remote users is independent of the choice of an information system, but A has higher performance for local users. Six months later, the same experiment is done, but now the results are different, i.e., A is always better.

QUESTION 1.3.1. *Can you think of an interpretation ?* [6]

	remote	local
A	123	43
	189	38
	99	49
	167	37
	177	44
B	107	62
	179	69
	199	56
	103	47
	178	71

	remote	local
A	141	75
	175	71
	192	62
	187	73
	125	58
B	201	90
	178	83
	193	102
	182	78
	186	92

Table 1.1: Data for Example 1.5 on page 7: measured performance of an information systems with two compiler options A and B. Test users measured the time to complete a complex transaction. Left: results of first tests. Right: results six months later.

The important thing about bottleneck is that they depend on all parameters of the system and the load: a component may be a bottleneck in some conditions, not in others. Knowing bottlenecks may considerably **simplify the performance evaluation**, as illustrated by the following example. More details can be found in Section 6.3.3.

[6] We cannot know from this simple series of facts. In fact, further measurements showed that all remote users access the information system via modem lines and an internet provider, which is the bottleneck in the first case. In the second case, the bottleneck is the server itself.

EXAMPLE 1.6: CPU MODEL. A detailed screening of a transaction system shows that one transaction costs in average: 1'238'400 CPU instructions; 102.3 disk accesses; 4 packets sent on the network. The processor can handle 10^9 instructions per second; the disk can support 10^4 accesses per second; the network can support 10^4 packets per second. We would like to know how many transactions per second the system can support.

QUESTION 1.3.2. *Can you give a rough estimate ? If you want more accuracy, what would you study in detail ?* [7]

Patterns are discussed in Chapter 5 and Chapter 6. The next example illustrates some of them.

EXAMPLE 1.7: PATTERNS. Consider the following scenarios.

1. The web server used for online booking at the "Fête des Vignerons" was so popular that it collapsed under the load, and was unavailable for several hours.
2. Buffers were added to an operating system task, but the overall performance was degraded (instead of improved, as expected).
3. When too many users are using the international link, the response time is poor
4. When too many users are present on the wireless LAN, no one gets useful work done
5. A traffic volume increase of 20% caused traffic jams
6. A new road was opened in the city center but there was no improvement
7. New parking facilities were created but there was no improvement

and the following patterns

(a) non-linearity of response time with respect to load
(b) congestion collapse (useful work decreases as load increases)
(c) performance is determined by bottleneck

QUESTION 1.3.3. *For each of the examples above, say which of the three patterns is present.* [8]

[7] The utilization per transaction is: CPU:0.12% – disk:1.02% –network:0.04%. The disk is the bottleneck; an upper bound on the capacity is 99 tps. To obtain more details, a first step is to model queuing at disk access, to see at which number of tps delays start becoming large. A global queuing model of CPU, disk access and network is probably not necessary.

[8] 1b; 2: maybe b, maybe other (see Chapter 5); 3a; 4b; 5b; 6: maybe b, maybe other (see Chapter 5); 7c

1.4 OTHER GUIDELINES FOR A SUCCESSFUL PERFORMANCE EVALUATION

1.4.1 THE SCIENTIFIC METHOD

The scientific method applies to any technical work, not only to performance evaluation. However, in the author's experience, lack of scientific method is one prominent cause for failed performance studies. In short, the scientific method simply requires that you do not believe in a conclusion unless it is thoroughly tested.

EXAMPLE 1.8: JOE'S KIOSK. Joe's e-kiosk sells online videos to customers equipped with wireless PDAs. Before deployment, performance evaluation tests are performed, as shown on Figure 1.1(a).

QUESTION 1.4.1. *What do you conclude about the throughput ?* [9]

Joe concludes that the bottleneck is the wireless LAN and decides to buy and install 2 more base stations. After installation, the results are on Figure 1.1(b).

QUESTION 1.4.2. *How do you interpret this ?* [10]

Joe scratches his head and decides to go more carefully about conclusions. Measurements are taken on the wireless LAN; the number of collisions is less than 0.1%, and the utilization is below 5%. This confirms that the wireless LAN is *not* a bottleneck. Joe makes the hypothesis that the bottleneck may be on the server side. After doubling the amount of real memory allocated to the server process, the results are as shown on Figure 1.1(c).

QUESTION 1.4.3. *What do you think ?* [11]

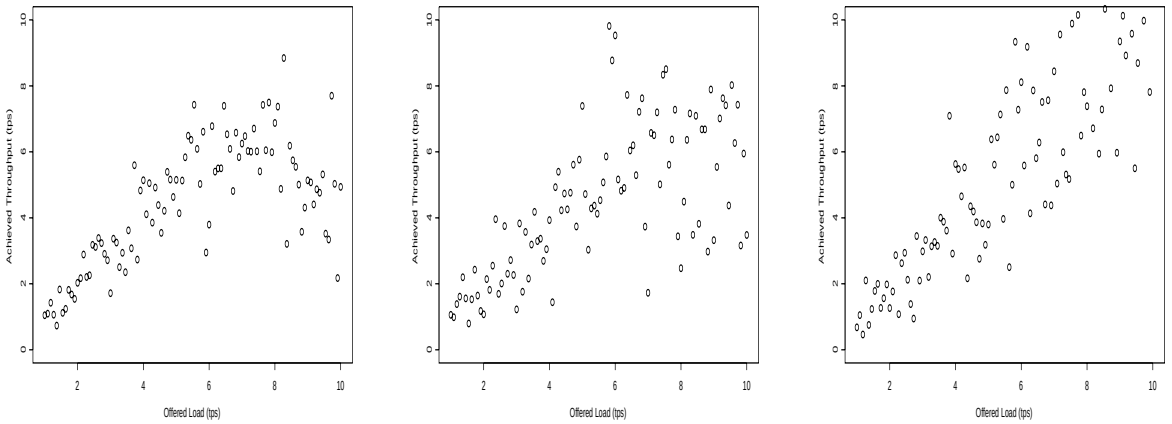
First, a common pitfall is to draw conclusions from an experiment that was not explicitly designed to validate these conclusion. The risk is that hidden factors might interfere, as illustrated by the previous example. Indeed, Joe concluded from the first experiment that the LAN performance would be improved by added a base station; this may have been *suggested* by the result of Figure 1.1(a), but this is not sufficient. It is necessary to perform other experiments, designed to validate this potential conclusion, before making a final statement.

EXAMPLE 1.9: IS ATM UBR BETTER THAN ATM ABR ?. In [Manthorpe00], the authors evaluate whether the ATM-UBR protocol is better than ATM-ABR (both are alternative methods used to manage switches used in communication networks). They use a typical scientific method, by posing each potential conclusion as a hypothesis and designing experiments to try and invalidate them:

[9] It reaches a maximum at around 8 tps.

[10] There is no improvement. The conclusion that the wireless LAN was a bottleneck was wrong.

[11] The bottleneck is now removed, which confirms that the real memory was the limiting factor.



(a) Question 1.4.1 (b) Question 1.4.2 (c) Question 1.4.3

Figure 1.1: Performance results for Joe's server. X-axis: offered load; Y-axis: achieved throughput, both in transactions per second.

> ABSTRACT. We compare the performance of ABR and UBR for providing high-speed network interconnection services for TCP traffic. We test the hypothesis that UBR with adequate buffering in the ATM switches results in better overall goodput for TCP traffic than explicit rate ABR for LAN interconnection. This is shown to be true in a wide selection of scenarios. Four phenomena that may lead to bad ABR performance are identified and we test whether each of these has a significant impact on TCP goodput. This reveals that the extra delay incurred in the ABR end-systems and the overhead of RM cells account for the difference in performance. We test whether it is better to use ABR to push congestion to the end-systems in a parking-lot scenario or whether we can allow congestion to occur in the network. Finally, we test whether the presence of a "multiplexing loop" causes performance degradation for ABR and UBR. We find our original hypothesis to be true in all cases. We observe, however, that ABR is able to improve performance when the buffering inside the ABR part of the network is small compared to that available at the ABR end-systems. We also see that ABR allows the network to control fairness between end-systems.

Second, give the **accuracy** of your quantitative results. Consider the measured data in Table 1.3. There is a lot of variability in them; saying that the average response time is better with B than A is not sufficient; it is necessary to give uncertainty margins, or confidence intervals. This is the objects of the techniques discussed in Chapter 2.

Last, make the results of your performance evaluation easily **reproducible**. This implies that all assumptions are made explicit and documented.

1.4.2 DIJKSTRA'S PRINCIPLE

Like the scientific method, it is a common sense principle that applies to any technical activity. It is known under several equivalent forms, all of which can be summarized by: **Remove what can be removed**.

- (*Occam*:) if two models explain some observations equally well, the simplest one is preferable
- (*Dijkstra*:) It is when you cannot remove a single piece that your design is complete.
- (Common Sense:) Use the adequate level of sophistication.

For example, using a detailed simulation to answer Question 1.3.2 would violate this principle.

1.5 REVIEW

1.5.1 CHECK-LIST

PERFORMANCE EVALUATION CHECKLIST

PE1 **Define your goal.** For example: dimension the system, find the overload behaviour; evaluate alternatives. Do you need a performance evaluation study ? Aren't the results obvious ? Are they too dependent on the input factors, which are arbitrary ?

PE2 **Identify the factors.** What are all the factors ? are there external factors which need to be controlled ?

PE3 **Define your metrics.** For example: response time, server occupancy, number of transactions per hour, Joule per Megabyte.

PE4 **Define offered load.** How is it expressed: transactions per second, number of users, number of visits per hour ? Is it measured on a real system ? artificial load generated by a simulator, by a synthetic load generator ? load model in a theoretical model ?

PE5 **Know your bottlenecks.** The performance often depends only on a small number of factors, often those whose utilization (= load/capacity) is high. Make sure what you are evaluating is one of them.

PE6 **Know your system well.** Know the system you are evaluating and list all factors. Use evaluation tools that you know well.

GENERAL PURPOSE CHECKLIST

S1 **Scientific Method**
do {Define hypothesis; design experiments; validate } **until** validation is OK

S2 Quantify the **accuracy** of your results.

S3 Make your findings **reproducible**; define your assumptions.

D1 **Remove** what can be removed.

1.5.2 REVIEW QUESTIONS

QUESTION 1.5.1. *Consider examples 11 and 12 in Example 1.1 on page 4. Which performance pattern do they correspond to ?* [12]

QUESTION 1.5.2. *Consider slides 298 and 299 in Nitin Vaidya's tutorial at Mobicom 2000 [Vaidya00-Mobicom1]. The author studies the performance of TCP on a mobile ad-hoc network, as a function of speed (of mobile). What can you conclude from these two slides ?* [13]

QUESTION 1.5.3. *Consider slides 300–305 in Nitin Vaidya's tutorial at Mobicom 2000 [Vaidya00-Mobicom2]. What can you conclude from these six slides ?* [14]

QUESTION 1.5.4. *What further measurements could be done to confirm the conclusion drawn in Question 1.3.1.* [15]

1.6 EXERCISES

EXERCISE 1.1. *Read [Singh02-Sigmetrics] and answer the following questions.*

1. *is the goal of the evaluation well defined ? What is it ?*
2. *are the factors identified ? What are they ?*
3. *what performance indices are chosen ?*
4. *how is the workload generated ?*
5. *are there implicit assumptions that should have been formulated ?*
6. *are the experiments or results reproducible ?*
7. *what conclusions can be drawn from the study ?*
8. *is the approach scientific ? do you believe the conclusions ? why ?*
9. *what techniques are used for the evaluation ?*
10. *is the level of sophistication adequate ?*
11. *was a performance analysis justified (aren't the results obvious or too dependent on input factors, which are arbitrary) ?*
12. *is there any part that can be removed ?*
13. *are the graphics OK ?*
14. *what aspects of the evaluation do you like or dislike?*

EXERCISE 1.2. *Same question with [Tan02-Sigmetrics]*

[12] Absence of congestion collapse.

[13] That mobility decreases throughput.

[14] That the previous conclusion was premature.

[15] Pose as assumption that the performance is a function of proportion of remote users and total load. Make measurements where these two factors take different values and analyze the dependency (for example, using a linear regression, see Part **??**).

CHAPTER 2

CONFIDENCE INTERVALS

In most measurements or simulations, we obtain data with some variability. The goal of this chapter is to review the techniques used to summarize such data into a small set of useful numbers, and to quantify the accuracy of the summarized data. Unfortunately, there are several competing summarization results, some of which are in widespread use due to historical more than scientific reasons. We first review these results, then we discuss their use in our setting. We use standard definitions of probability theory recalled in appendix.

Contents

2.1 SUMMARIZING PERFORMANCE DATA

WHAT IS SUMMARIZATION ? Assume you have obtained a large set of results for the value of a performance metric. This can be fully described by the distribution of the data, and illustrated by a histogram. The histogram displays on the y-axis the ratio of data that fall in the bin on the x axis. Summarizing means compressing it into one or a few numbers that represent both its average and variability. In practice of communication and information systems, this is done by either one of the following two:

Median and Quantile. A median is a value that falls in the middle of the distribution, i.e. 50% of the data is below and 50% above. A p%-quantile leaves p% of the observation below and $100 - p$% above. The median gives some information about the average, while extreme quantiles give information about the dispersion. A commonly use plot is the *Box Plot*. It shows the median, the 25% and 75% quantiles (called "quartiles") and the "outliers", defined as data points that are a fixed fraction away from the quartiles. It also shows variability by the following heuristic. It plots a line that extends to the most extreme value up to 1.5 times the inter-quartile distance (distance 3 on Figure 2.1).

The *sample median* of a data set is defined as follows. Assume there are n data points $x_1, ..., x_n$. Sort the points in increasing order and obtain $x_{(1)} \leq ... \leq x_{(n)}$. If n is odd, the median is $x_{(\frac{n+1}{2})}$, else $\frac{1}{2}(x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)})$. More generally, the *sample q- quantile* is defined as $\frac{x_{(k')}+x_{(k'')}}{2}$ with $k' = \lfloor qn + (1-q) \rfloor$ and $k' = \lceil qn + (1-q) \rceil$. $\lfloor x \rfloor$ is the largest integer $\leq x$ and $\lceil x \rceil$ is the smallest integer $\geq x$

Mean and Standard Deviation. The *mean* m of a data set $x_1, ..., x_n$ is $m = \frac{1}{n}\sum_{i=1}^{n} x_i$. It gives some information about the average. The *standard deviation s of a data set* is defined by $s^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - m)^2$ or $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - m)^2$ (either conventions are used – see Section 2.2 for an explanation). It gives information about the variability. The use of standard deviation is rooted in the belief that data roughly follows a normal distribution, with some mean μ and some variance σ^2. The normal distribution is characterized by histogram with Bell shape (see appendix). It is very frequently encountered because of the central limit theorem that says that an average of many things tends to be normal (but see some exceptions in Chapter 8). If such a hypothesis is true, and if we had $m \approx \mu$ and $\sigma \approx s$, then with 95% probability, the data sample would lie in the interval $m \pm 1.96s$ (see the normal distribution table in appendix). This justifies the use of *mean-variance* plots like in Figure 2.1 that use as a measure of variability the interval $m \pm 1.96s$ (distance 3 on Figure 2.1). This is also called a *prediction interval* since it predicts a likely range for a future sample (Section 2.5).

EXAMPLE 2.1: COMPARISON OF TWO OPTIONS. An operating system vendor claims that the new version of the database management code significantly improves the performance. We measured the execution times of a series of commonly used programs with both options. The data are displayed in Figure 2.1. The raw displays and histograms show that both options have the same range, but it seems (graphically) that the new system more often provides a smaller execution time. The box plots are more suggestive; they show that the average and the range are about half for the new system.

In Section 2.7 we discuss the differences between these two modes of summarization.

Comparing Data Sets is easily done with their *empirical cumulative distribution functions* (ECDFs). The ECDF of a data set $x_1, ..., x_n$ is the function f defined by

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} 1_{\{x_i \leq x\}} \tag{2.1}$$

so that $f(x)$ is the proportion of data samples that do not exceed x. On Figure 2.2 we see that the new data set clearly outperforms the old one.

2.2 CONFIDENCE INTERVALS FOR MEDIAN AND OTHER QUANTILES

2.2.1 WHAT IS A CONFIDENCE INTERVAL ?

For any number that we display, we should give some statement about its accuracy: this is a scientific principle (Chapter 1). *Confidence intervals* quantify the uncertainty about a summarized

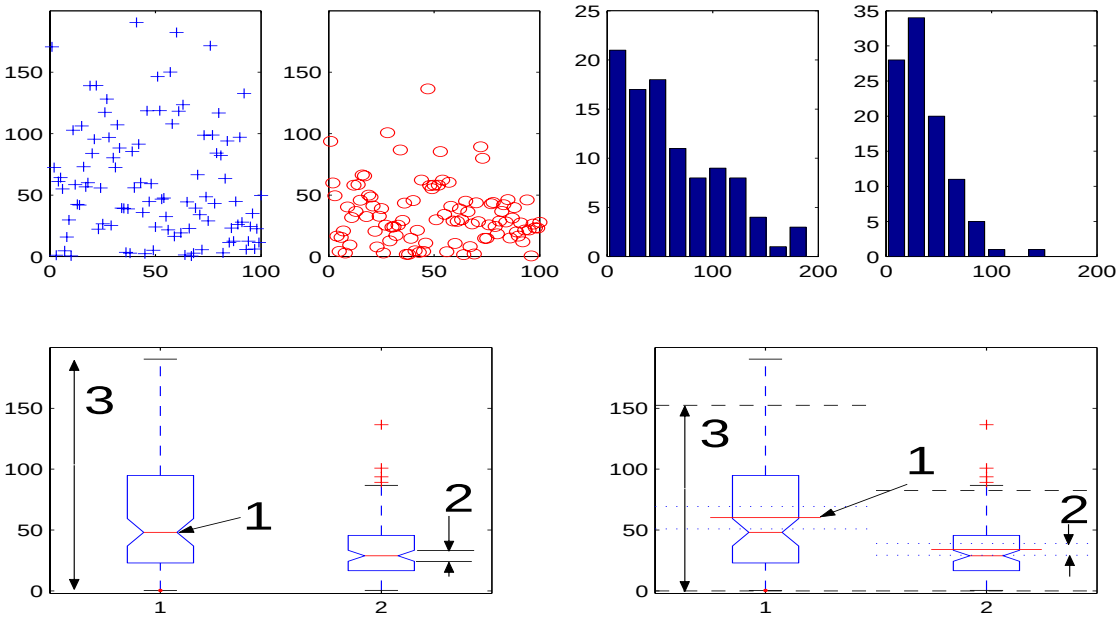


Figure 2.1: Data for Example 2.1 on page 15. Top : measured execution time, in ms, for 100 transactions with the old (left) and new (right) code, followed by histograms. Bottom left: Box Plot, showing median (1), confidence interval for the median (2) and variability (3) for both old and new code. Bottom right: Box Plots overlaid with: mean (1), confidence interval for the mean (2) prediction interval for a sample (3), using formulas for the normal case.

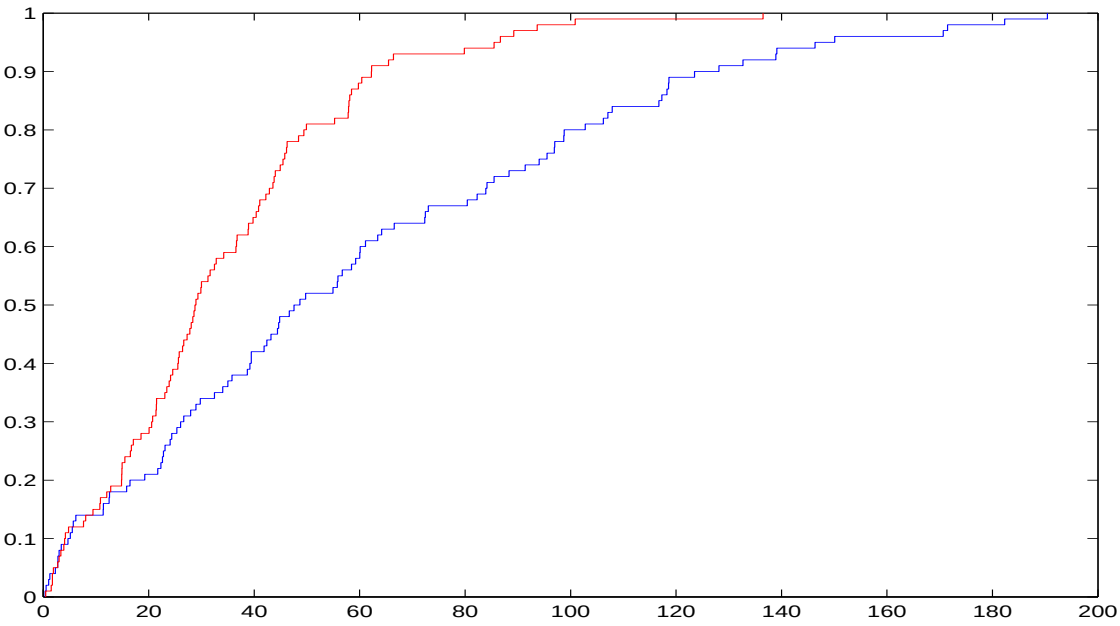


Figure 2.2: Data of Example 2.1 on page 15. Empirical distribution functions for the old code (right curve) and the new one (left curve). The new outperforms the old, the improvement is significant at the tail of the distribution.

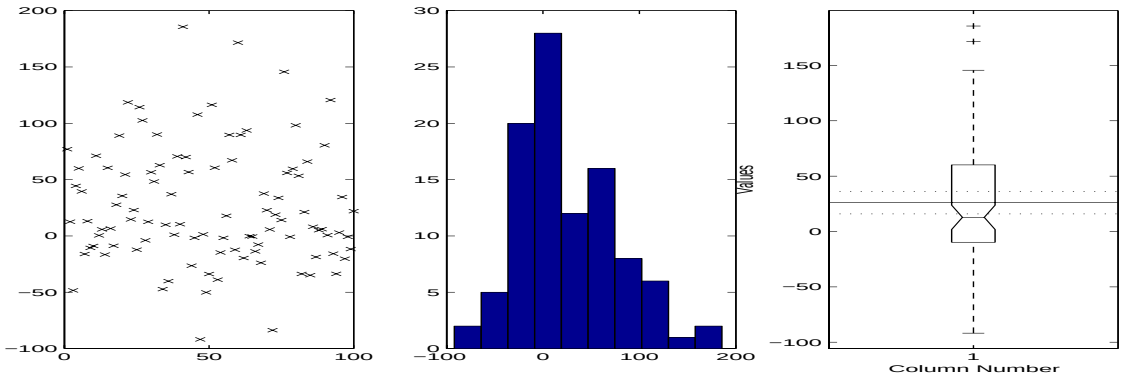


Figure 2.3: Data for Example 2.2 on page 17: reduction in run time (in ms). Right: Box plot with mean and confidence interval for mean.

data due to the randomness of the measurements.

EXAMPLE 2.2: COMPARISON OF TWO OPTIONS, CONTINUED. We wish to quantify the improvement due to the new system. To this end, we measure the reduction in run time for the same sequence of tasks as on Figure 2.1 (both data sets on Figure 2.1 come from the same transaction sequences – statisticians say that this is a *paired experiment*). The differences are displayed in Figure 2.3, with Box-Cox and mean/standard deviation diagrams. For example, the mean of the reduction in run time is 26.1 ± 10.2.The uncertainty margin is called the confidence interval for the mean. It is obtained by the method explained in this section. Here, the mean reduction is non negligible, but the uncertainty about it is large.

Figure 2.1 and Figure 2.3 show confidence intervals for the mean (horizontal lines) and for the median (notches in Box plot). Note that the confidence interval is **not the same as a measure of variability**, though it is related, as we discuss in Section 2.7: on Figure 2.1 the confidence interval for the mean is considerably smaller than the variability interval given by $m \pm 1.96s$. There is a confidence interval for each of the summarized data given earlier: median, quantile, mean and standard deviation.

2.2.2 ASSUMPTION ON INDEPENDENCE

We assume that the collected data comes from a set of *independent* and identically distributed (*iid*) samples. We discuss this assumption in this section.

WHAT DOES IID MEAN ?

Iid is a property of a stochastic model, not of the data. When we say, by an abuse of language, that the collected data set is iid, we mean that we can do as if the collected data $x_1, ..., x_n$ is a sample (i.e. a simulation output) for a sequence of random variables $X_1, ..., X_n$, where $X_1, ..., X_n$ are independent and all have the same (maybe unknown) distribution with cdf $F()$.

To generate such as sample, we draw a random number from the distribution $F()$, using a random number generator (see Section 3.6). Independence means that the random numbers generated at every step i are discarded and not re-used in the future steps $i+1, ...$ Another way to think of independence is with conditional probabilities: for any set of real numbers A

$$\mathbb{P}(X_i \in A \mid X_1 = x_1, ..., X_{i-1} = x_{i-1}) = \mathbb{P}(X_i \in A) \tag{2.2}$$

i.e. **if we know the distribution** $F(x)$, observing $X_1, ..., X_{i-1}$ does not give more information about what could happen to X_i.

Note the importance of the "if" statement in the last sentence: if we remove it, the sentence is no longer true. To understand why, consider a sample $x_1, ..., x_n$ for which we assume to know that it is generated from a sequence of iid random variables $X_1, ..., X_n$ with normal distribution but with unknown parameter (μ, σ^2). If we observe for example that the average of $x_1, ..., x_{n-1}$ is 100 and all values are between 0 and 200, then we can think that it is very likely that x_n is also in the interval $[0, 200]$ and that it is unlikely that x_n exceeds 1000. Though the sequence is iid, we did gain information about the next element of the sequence having observed the past. There is no contradiction: if we know that the parameters of the random generator are $\mu = 100$ and $\sigma^2 = 10$ then observing $x_1, ..., x_{n-1}$ gives us no information about x_n.

QUESTION 2.2.1. *Give an example of identically distributed but dependent random variables.* [1]

HOW DO I KNOW IN PRACTICE IF THE IID ASSUMPTION IS VALID ?

If your performance data comes from a *designed experiment*, i.e. a set of simulation or tests that is entirely under your control, then it is up to you to design things in such a way that the collected data are iid. This is done as follows.

Every experiment has a number of factors, i.e., parameters that are likely to influence the outcome. Most of the factors are not really interesting, but you have to account for them in order to avoid hidden factor errors (see Section 4.6 for details). The experiment generates iid data if the values of the factors are chosen in an iid way, i.e., according to a random procedure that is the same for every measured point, and is memoriless. Consider Example 2.1 on page 15, where the run time for a number of transactions was measured. One factor is the choice of the transaction. The data is made iid if, for every measurement, we choose one transactions **randomly with replacement** in a list of transactions.

A special case of designed experiment is simulation. Here, the method is to generate **replications** without resetting the random number generator, as explained in Section 3.3.

Else (i.e. your data does not come from a designed experiment but from measurements on a running system) there is little chance that the complete sequence of measured data is iid. A simple fix is to **randomize the measurements**, in such a way that from one measurement point to the other there is little dependence. For example, assume you are measuring the response time of an operational web server by data mining the log file. The response time to consecutive requests is highly correlated at the time scale of the minute (due to protocols like TCP); one common solution is to choose requests at random, for example by selecting one request in average every two minutes. If you are in doubt, you can verify the iid-ness by the methods discussed in Section 2.4.4.

[1] Here is a simple one: assume $X_1, X_3, X_5, ...$ are iid with cdf $F()$ and let $X_2 = X_1$, $X_4 = X_3$ etc. The distribution of X_i is $F()$ but the distribution of X_2 conditional to $X_1 = x_1$ is a dirac at x_1, thus depends on x_1. The random choices taken for X_1 influence (here deterministically) the value of X_2.

DO WE NEED THE IID ASSUMPTION ?

The iid assumption is not mandatory, it is just a convenient one, which makes the computation of confidence intervals easy (using the methods described in the rest of this chapter). It is possible to obtain confidence intervals even when the data does not appear to be iid, but this is an order of magnitude more complicated. In Section 2.9, we study such an example.

2.2.3 CONFIDENCE INTERVAL FOR MEDIAN AND OTHER QUANTILES

We explain now how these confidence intervals are computed, which also serves as an illustration of the general method for computing confidence intervals. The confidence interval for the median is shown by notches on Box plots (Figure 2.1, (3) on Box plot). We start with the median and then extend it to other quantiles.

Recall that we interpret the data $x_1, ..., x_n$ as a sample for a sequence of iid random variables $X_1, ..., X_n$, with common cdf $F()$. The distribution $F()$ is non-random but is unknown. It has a well defined median m, defined by $\mathbb{P}(X_i \leq m) = 0.5$. We can never know m exactly, but we **estimate** it by $\hat{m}(x_1, ..., x_n)$, equal to the sample median defined in Section 2.11.1 (in Section 2.8 we discuss the choice of an estimator in more detail). Note that the value of the estimated median depends on the data, so it is random: for different measurements, we obtain different estimated medians. The goal of a confidence interval is to bound this uncertainty. It is defined relative to a *confidence level* γ; typically $\gamma = 0.95$ or 0.99:

DEFINITION 2.2.1. *A* **confidence interval** *at level γ for the fixed but unknown parameter m is an interval $(u(X_1, ..., X_n), v(X_1, .., X_n))$ such that*

$$\mathbb{P}\left(u(X_1, ..., X_n) < m < v(X_1, ..., X_n)\right) \geq \gamma \tag{2.3}$$

In other words, the interval is constructed from the data, such that with at least 95% probability (for $\gamma = 0.95$) the true value of m falls in it. Note that **it is the confidence interval that is random, not the unknown parameter** m.

A confidence interval for the median or a quantile is obtained thanks to the following theorem.

THEOREM 2.2.1. *Let $X_1, ..., X_n$ be n iid random variables whose common distribution has a density. Let $X_{(1)} \leq X_{(2)} \leq ... \leq X_{(n)}$ be the order statistic, i.e. the set of values of X_i sorted in increasing order. For $0 < p < 1$ let m_p be a p-quantile of the common distribution of the X_is. A confidence interval for m_p is $[X_{(j)}, X_{(k)}]$ where j and k satisfy $B_{n,p}(k-1) - B_{n,p}(j-1) \geq \gamma$ ($B_{n,p}$ is the cdf of the binomial distribution). See the tables in Section 14.2 for practical values. For large n, the binomial cdf can be approximated by a normal distribution, as shown in the tables.*

Note. The assumption that the distribution has a density is for simplicity of exposition. In practice it holds when the values of X_i are real numbers. Else, typically, X_i takes integer values; the *distribution quantiles* are defined as follows. The cdf $F()$ is defined for integer arguments only; we can extend it to real arguments by linear interpolation: $F(x) = (x - n)F(n + 1) + (n + 1 - x)F(n)$ where n is the integer part of x. This extension is continuous and a p- quantile is defined as a value of x such that $F(x) = p$. In such cases the results are essentially the same.

Proof. The distribution of the X_is has a density, so the cdf is continuous (it has no jump) and the true (unknown) quantile m_p satisfies $\mathbb{P}(X_i < m_p) = p$. Let $Z_i = 1$ if $X_i < m_p$, 0 otherwise and $N = \sum_{i=1}^{n} Z_i$, i.e. N is the number of times that X_i is below m_p. We have the event equalities

$$\begin{aligned} \{X_{(j)} < m_p\} &= \{N \geq j\} \\ \{X_{(k)} \geq m_p\} &= \{N \leq k-1\} \end{aligned}$$

thus

$$\mathbb{P}\left(X_{(j)} < m_p \leq X_{(k)}\right) = \mathbb{P}(j \leq N \leq k-1) = \mathbb{P}(N \leq k-1) - \mathbb{P}(N \leq j-1)$$

Now Z_i are iid Bernoulli(p) random variables thus N is Binomial(n, p). Further, X_i has a density and thus $(X_{(j)}, X_{(k)})$ as well (Chapter 12) and $\mathbb{P}\left(X_{(j)} < m_p \leq X_{(k)}\right) = \mathbb{P}\left(X_{(j)} < m_p < X_{(k)}\right)$. For large n, we approximate the binomial cdf by N_{μ,σ^2} with $\mu = np$ and $\sigma^2 = np(1-p)$.

The values in Section 14.2 are chosen such that j and k are as symmetric as possible around $\frac{n+1}{2}$.

□

For $n = 10$, the theorem and the table in Section 14.2 say that a 95%-confidence interval for the median is $\left[X_{(2)}, X_{(9)}\right]$. The table also says that in fact this confidence interval is at the level 0.979. Due to the discrete nature of the solution, it is not possible here to obtain exactly a confidence level of 95%. Also recall that the estimated median is $\frac{X_{(5)}+X_{(6)}}{2}$.

For $n = 31$ the table gives the interval $\left[X_{(10)}, X_{(22)}\right]$. Note that this is not the only interval that can be obtained from the theorem. Indeed, we have:

j	k	$\mathbb{P}\left(X_{(j)} < m_{0.5} < X_{(k)}\right)$
9	21	0.959
10	22	0.971
11	23	0.959

Thus we have **several** possible confidence intervals. The table simply picked one for which the indices are closest to being symmetrical around the estimated median, i.e. the indices j and k are equally spaced around $\frac{n+1}{2}$, which is used for estimating the median. In some cases, like $n = 32$, we do not find such an interval exactly; we have for instance:

j	k	$\mathbb{P}\left(X_{(j)} < m_{0.5} < X_{(k)}\right)$
10	22	0.965
11	23	0.965

Here, the table arbitrarily picked the former.

Note that for small values of n, no confidence interval is possible at levels 0.95% or 0.99%. This is because the probability that the true quantile is outside any of the observed data is still large. For larger values of n, the confidence interval becomes much smaller.

EXAMPLE 2.3: Figure 2.1 shows the confidence intervals for the medians computed with this method.

2.3 CONFIDENCE INTERVAL FOR THE MEAN AND STANDARD DEVIATION

Estimating the mean with confidence interval is more complicated than for the median. Like before we assume that the collected data is iid. Here the normal distribution plays a special role, due to the central limit theorem which says that an average of many things that are not heavy tailed tends to be normally distributed (see Chapter 12 for the central limit theorem and Chapter 8 for the definition of heavy tail). Specifically, there are two special cases of interest:

- **Normal, IID**: the common distribution is normal. Simple formulae are available (Section 2.3.1 but we need to verify normality (Section 2.4).
- **Large Sample, IID**: if the data is not normal but the sample size is large ($n \geq 30$ or more, depending on how much the distribution deviates from a normal one) then a normal asymptotic with simple formulas can be used (Section 2.3.2). Verification can be done as explained in Section 2.4.
- **General IID**: else the bootstrap estimate can be used. However, it tends to understimate the confidence intervals.

2.3.1 NORMAL IID CASE

We assume the common cdf of all X_is is normal N_{μ,σ^2}, where the parameters μ and σ^2 are fixed but unknown. The problem becomes now to estimate the mean μ and the standard deviation σ^2. The solution is provided by the following theorem.

THEOREM 2.3.1. *Let $X_1, ..., X_n$ be a sequence of iid random variables with common distribution N_{μ,σ^2}. Define*

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.4}$$

$$\hat{\sigma}_n^2 = \frac{1}{n-1}\sum_{i=1}^{n} (X_i - \hat{\mu}_n)^2 \tag{2.5}$$

Then

- *The distribution of $\sqrt{n}\frac{\hat{\mu}_n - \mu}{\hat{\sigma}_n}$ is Student's t_{n-1}; a confidence interval for the mean at level $1-\alpha$ is*

$$\hat{\mu}_n \pm \eta\frac{\hat{\sigma}_n}{\sqrt{n}} \tag{2.6}$$

  where η is the $\left(1-\frac{\alpha}{2}\right)$ quantile of the student distribution t_{n-1}.
- *The distribution of $(n-1)\frac{\hat{\sigma^2}_n}{\sigma^2}$ is χ^2_{n-1}. A confidence interval at level $1-\alpha$ for the standard deviation is*

$$[\hat{\sigma}_n\sqrt{\frac{\zeta}{n-1}}, \hat{\sigma}_n\sqrt{\frac{\xi}{n-1}}] \tag{2.7}$$

  where ζ and ξ are quantiles of χ^2_{n-1}: $\chi^2_{n-1}(\zeta) = \frac{\alpha}{2}$ and $\chi^2_{n-1}(\xi) = 1-\frac{\alpha}{2}$.

The distributions: χ^2 and Student's t are defined in Chapter 12. For instance, with $n = 100$ and confidence level 0.95, we find from the tables in Section 14.1: $\eta = 1.98$, $\zeta = 73.4$, and $\xi = 128.4$. This gives the confidence intervals for mean and standard deviation (we drop the index n): $[\hat{\mu} - 0.198\hat{\sigma}, \hat{\mu} + 0.198\hat{\sigma}]$ and $[0.86\hat{\sigma}, 1.14\hat{\sigma}]$. Note that the amplitudes of the confidence intervals decrease roughly like $\frac{1}{\sqrt{n}}$.

Proof. 1. The random variable $\hat{\mu}_n$ is normal $N_{\mu, \frac{\sigma^2}{n}}$. The random variable $\hat{\sigma}_n^2$ has expectation σ^2 and a distribution equal to $\sigma^2 \chi^2_{n-1}$. This follows from Section 12.5.

2. The second bullet follows immediately.

3. Further, the general theory in Section 12.5 shows that $\hat{\sigma}_n^2$ is independent of $\hat{\mu}_n$. This, together with the definition of the student t, shows the first bullet.

□

Comment 1. $\hat{\mu}_n$ and $\hat{\sigma}_n$ are **estimators** of the mean and standard deviation. The choice of $\hat{\mu}_n$ (which is the *sample mean*) appears to be fairly natural for estimating the distribution mean. In contrast, a natural estimator for the variance would be the mean square error $s_n^2 = \frac{1}{n}\sum_{i=1}^n (X_i - \hat{\mu}_n)^2$, which differs from the definition in Equation (2.5) by the factor $\frac{1}{n}$ instead of $\frac{1}{n-1}$; this is required for the statements in the theorem to hold exactly (see in the proof). We discuss a general theory of estimators in Section 2.8. In practice, it is not required to have an extreme accuracy for the estimator of σ^2 (since it is a second order parameter); thus using $\frac{1}{n-1}$ or $\frac{1}{n}$ makes little difference. s_n is often called the *sample standard deviation*.

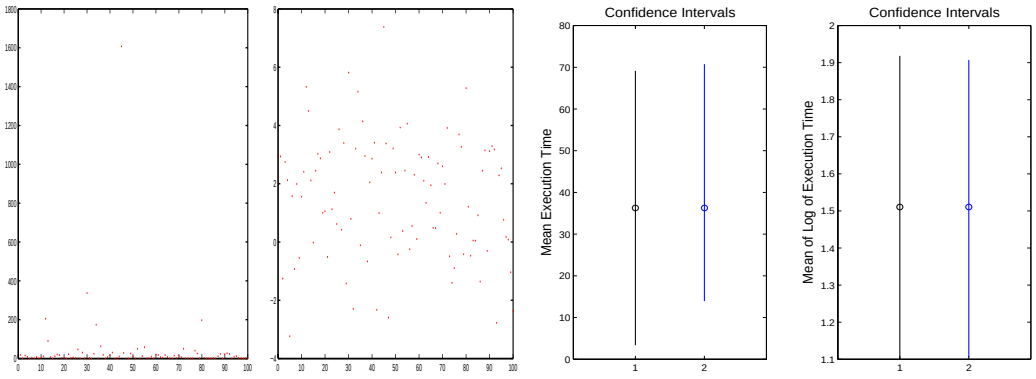
Comment 2. The confidence intervals in the theorem are not the only possible ones. Any interval of the form $[\hat{\mu}_n - \eta_1 \frac{\hat{\sigma}_n}{\sqrt{n}}, \hat{\mu}_n + \eta_2 \frac{\hat{\sigma}_n}{\sqrt{n}}]$ where $t_{n-1}(-\eta_1) = \alpha_1$, $t_{n-1}(\eta_2) = 1 - \alpha_2$ and $\alpha_1 + \alpha_2 = \alpha$ is also a confidence interval; for example, with $n = 100$, $\eta_1 = 2.37$ and $\eta_2 = 1.77$ correspond to $\alpha_1 = 0.01$ and $\alpha_2 = 0.04$. In practice, as in the theorem, we take $\alpha_1 = \alpha_2 = \frac{\alpha}{2}$.

Comment 3. The random variables $\sqrt{n}\frac{\hat{\mu}_n - \mu}{\hat{\sigma}_n}$ and $(n-1)\frac{\hat{\sigma}_n^2}{\sigma^2}$ are constructed from the data, but their distribution is free of the parameters μ and σ. They are called *pivots*. The trick to find confidence intervals is to obtain a pivot.

EXAMPLE 2.4: FILE TRANSFER TIMES. Figure 2.4 shows the file transfer times obtained in 100 independent simulation runs, displayed in natural and log scales. The last panel shows 95%-confidence intervals for the mean of the data and the mean of the log of the data, computed with Theorem 2.3.1 (assuming the data is normal) and with the bootstrap method (explained in Section 2.3.3) for verification. The normal assumption is valid in log scale, but not in natural scale.

QUESTION 2.3.1. *Does the confidence interval for the mean depend on the estimator of the variance ? Conversely ?* [2]

[2] Yes; No



(a) (Data) (b) (Log of data) (c) (Confidence Intervals)

Figure 2.4: File transfer times for 100 independent simulation runs, with confidence intervals computed with (1) Theorem 2.3.1 (assuming the data is normal) and with (2) the bootstrap method Section 2.3.3)

2.3.2 GENERAL CASE, n LARGE

When the data sample is large, we can use the following asymptotic result for the mean; there is no simple result for the variance.

THEOREM 2.3.2. *Let $X_1, ..., X_n$ be n iid random variables with a common distribution that has a mean μ and a variance σ^2. Define $\hat{\mu}_n$ and s_n^2 by*

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.8}$$

$$s_n^2 = \frac{1}{n}\sum_{i=1}^{n} (X_i - \hat{\mu}_n)^2 \tag{2.9}$$

The distribution of $\sqrt{(}n)\frac{\hat{\mu}_n - \mu}{s_n^2}$ tends to the normal distribution $N_{0,1}$ when $n \to +\infty$. An approximate confidence interval for the mean at level $1-\alpha$ is

$$\hat{\mu}_n \pm \eta \frac{s_n}{\sqrt{n}} \tag{2.10}$$

where η is the $\left(1 - \frac{\alpha}{2}\right)$ quantile of the normal distribution $N_{0,1}$.

Proof. By the central limit theorem, $\sqrt{(}n)\frac{\hat{\mu}_n - \mu}{\sigma^2}$ converges in distribution to $N_{0,1}$. Now

$$s_n^2 = \frac{1}{n}\sum_{i=1}^{n} X_i^2 - \hat{\mu}_n^2$$

and by the strong law of large numbers, $\frac{1}{n}\sum_{i=1}^{n} X_i^2$ converges almost surely to $\mathbb{E}(X_1^2) = \sigma^2 - \mu^2$ and $\hat{\mu}_n^2$ to μ^2; thus s_n^2 converges almost surely to σ^2. The rest follows from Theorem 12.2.1.

□

For instance, with $n = 100$ and confidence level 0.95, we find from the table in Section 14.1: $\eta = 1.96$. This gives the confidence intervals for the mean (we drop the index n): $[\hat{\mu} - 0.196s, \hat{\mu} +$

$0.196s]$. If it happens that X_i has a normal distribution, we can compare this approximate result to the exact one, given after Theorem 2.3.1: 1.98 is replaced by 1.96 and s by $\hat{\sigma}$; the difference is of the order of 2%, which is negligible since the amplitude of the confidence interval is a second order quantity. For n as low as 30, the difference is still negligible (ca 7%). In general, there is "continuity" effect: if the distribution of X_i is not far from normal, the approximation in the theorem is good for small values of n

Comment 1. The same theorem holds if we replace s_n by $\hat{\sigma}_n^2$, see the discussion after Theorem 2.3.1.

Comment 2. There is no simple result for a confidence interval for the standard deviation. Such an interval would require an estimate of the fourth moment, which is usually not done.

EXAMPLE 2.5: Figure 2.5 shows confidence intervals for the Example 2.1 on page 15 computed with the asymptotic result in Theorem 2.3.2.

2.3.3 THE BOOTSTRAP METHOD

is a simple, yet efficient method, that can be applied when the data is not normal, all transformations to make it normal also fail (Section 2.6.1), and we are not sure whether the sample size is large enough to justify using the asymptotic results in Section 2.3.2.

The bootstrap method is general and can be used for any estimator. Consider a sample $\vec{x} = (x_1, ..., x_n)$ obtained from n iid realizations of one random variable. We want to find a confidence interval for some statistic $t(\vec{x})$. For the mean we have $t(\vec{x}) = \frac{1}{n}\sum_{i=1}^{n} x_i$. The *bootstrap method* uses the sample $\vec{x} = (x_1, ..., x_n)$ as an approximation of the true, unknown distribution. It works as follows.

Fix some number R (defined later) and create R *bootstrap replicates* $\vec{X}^r$, $r = 1, ..., R$. Each bootstrap replicate $\vec{X}^r = (X_1^r, ..., X_n^r)$ is a random vector of size n, like the original data. All X_i^r are independent copies of the same random variable, obtained by drawing from the set $\{x_1, ..., x_n\}$ (with replacement). Thus, in the case where all x_k are distinct, for any fixed r, i, k, we have $\mathbb{P}(X_i^r = x_k) = \frac{1}{n}$.

Now for each r, compute $T^r = t(\vec{x^r})$. It is the value of the statistic obtained at the rth "replayed" experiment. The *percentile bootstrap estimate* at level $1 - \alpha$ is an approximate confidence interval for the statistic t (for example the mean), defined as

$$\left(T_{\left((R+1)\frac{\alpha}{2}\right)}, T_{\left((R+1)(1-\frac{\alpha}{2})\right)}\right) \tag{2.11}$$

where $(T_{(r)})_{r=1,...,R}$ is the order statistic of $(T^r)_{r=1,...,R}$.

The value of R needs to be chosen such that there are sufficiently many points outside the interval. A good value is $R = \frac{50}{\alpha} - 1$. For example, with $\alpha = 0.05$, take $R = 999$ and the confidence interval is $\left(T_{(25)}, T_{(975)}\right)$.

In essence, we have used the sample data to obtain an empirical estimate of the distribution of the statistic t.

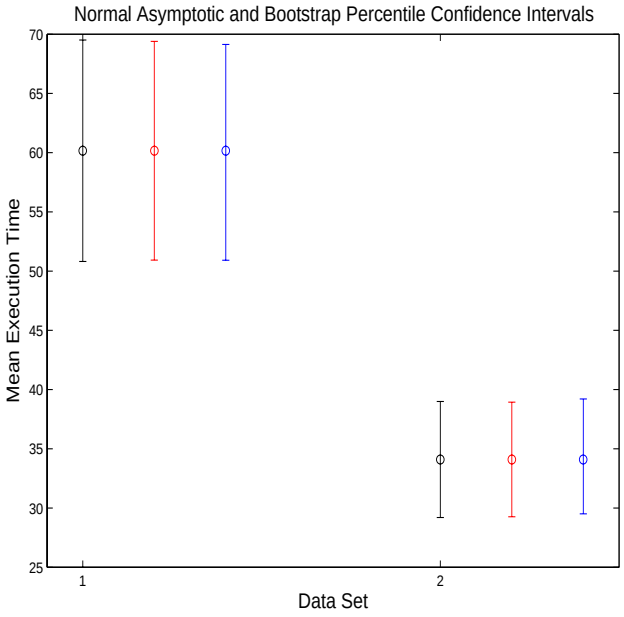


Figure 2.5: Confidence intervals for both compiler options of Example 2.1 on page 15 computed with three different methods: assuming data would be normal (Theorem 2.3.1) (left); assuming n is large enough for the asymptotic result in Theorem 2.3.2 to hold (center) and with the bootstrap method (right).

In general, the percentile estimate is an approximation that tends to be slightly too small (see Figure 2.5 for an example). For a theoretical justification of the bootstrap method, and other applications, see [DavisonHinkley97-book].

EXAMPLE 2.6: COMPARISON OF TWO OPTIONS. (Example 2.1 on page 15 continued). The data sets are not normal, as shown in Section 2.4 but we may think that n is large and apply Theorem 2.3.2. The confidence intervals of mean execution times for old and new compiler options obtained by the three methods are shown on Figure 2.5. Though the normal assumption is not valid, the result obtained with it correct as it does not differ significantly from the asymptotic result.

2.4 VERIFYING ASSUMPTIONS

The methods in the previous section make some assumptions that need to be verified, as we explain now. In this chapter we stay with simple methods, based on visual inspection of qq-plots (defined in the next section). More formal, automated methods use tests, as described in Section 9.7.1 on Page 217.

2.4.1 QQPLOTS

A *probability plot*, also called *qq-plot*, compares two samples X_i, Y_i, $i = 1, ..., n$ in order to determine whether they come from the same distribution. Call $X_{(i)}$ the *order statistic*, obtained by sorting X_i in increasing order. Thus $X_{(1)} \leq X_{(2)} \leq ....$ The qq-plot displays the points $(X_{(i)}, Y_{(i)})$. If the points are approximately along a straight line, then the distributions of X_i and Y_i can be assumed to be the same, modulo a change of scale and location.

Most often, we use qqplots to check the distribution of Y_i against a probability distribution F. To do so, we plot $(x_i, Y_{(i)}$, where x_i is an estimation of the expected value of $\mathbb{E}(Y_{(i)})$, assuming the marginal of Y_i is F. The exact value of $\mathbb{E}(Y_{(i)})$ is hard to obtain. Assume that F is strictly increasing; a simple approximation is

$$x_i := F^{-1}\left(\frac{i}{n+1}\right)$$

This is justified as follows. Let $U_i = F(Y_i)$. The distribution of U_i is uniform on $[0, 1]$. Further, $U_{(i)} = F(X_{(i)})$ for all i. It can be shown [Davisson-02] that

$$\mathbb{E}\left(U_{(i)}\right) = \frac{i}{n+1},$$

which has a simple interpretation if we think that the order statistic of U_i has to be placed evenly on $[0, 1]$. Also, as n is large, $U_{(i)}$ converges to its expectation. Thus we can approximate as follows

$$\mathbb{E}(Y_{(i)}) = \mathbb{E}(F^{-1}(U_{(i)})) \approx F^{-1}(\mathbb{E}(U_{(i)})) = x_i$$

which is done in the qqplots shown by statistical packages.

2.4.2 VERIFYING THE NORMAL ASSUMPTION

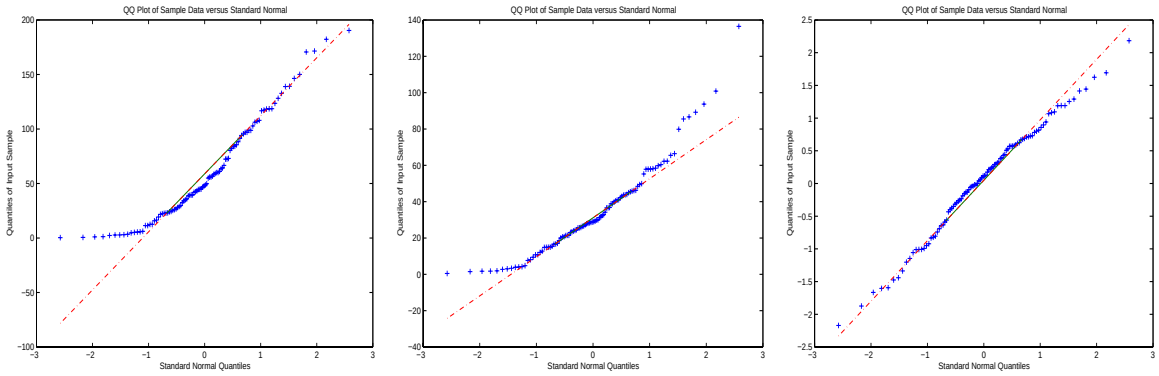
is best done by visual inspection of a normal qq-plot. More formal methods based on tests are described in Section 9.7.1 on Page 217, but they do not necessarily provide a better diagnostic than visual inspection (but they can be used in an automated way). See Figure 2.6 for an example.

2.4.3 VERIFYING THE ASYMPTOTIC REGIME

When n is large, we can use the asymptotic result in Theorem 2.8.1: the distribution of the sample mean is asymptotically normal, even if $x_1, ..., x_n$ is not. The problem is to know whether n is large or not. Ideally, we would like to test whether the distribution of t is normal, but we cannot do it since we have only one value.

The bootstrap method can be used to solve this problem. The method consists in examining the R bootstrap replicates T^r as in Section 2.3.3; if they appear to be normal, it is an indication that the distribution of t is normal. For example, Figure 8.4 shows that the asymptotic regime is indeed reached for the data sets in Example 2.1 on page 15.

EXAMPLE 2.7: FILE TRANSFER TIMES. (Continuation of Example 2.4 on page 22). Figure 2.8 shows the qq-plots of the bootstrap replicates used for estimating the mean



(a) (old code) (b) (new code) (c) (normal sample)

Figure 2.6: Normal qqplots of data in Figure 2.1 and of an artificially generated sample from the normal distribution with the same number of points. For both data sets the small values are smaller (lighter left tail).They do not appear to come from a normal distribution.

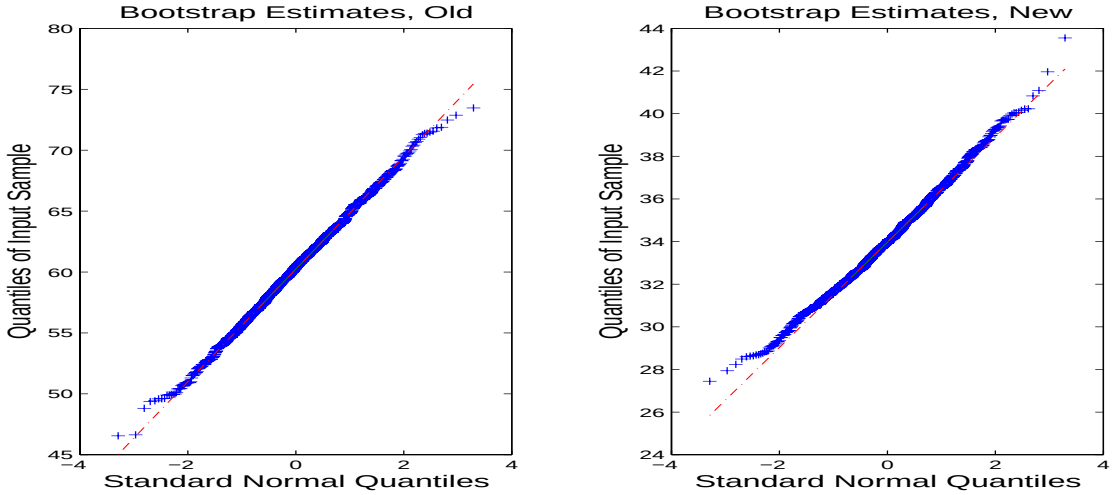


Figure 2.7: Bootstrap replicates of the estimators of the mean for both compiler options of (Example 2.1 on page 15 (qqplots). They appear to be normally distributed, thus the normal asymptotic regime is reached and the use of Theorem 2.3.2 to compute confidence intervals for the mean is valid.

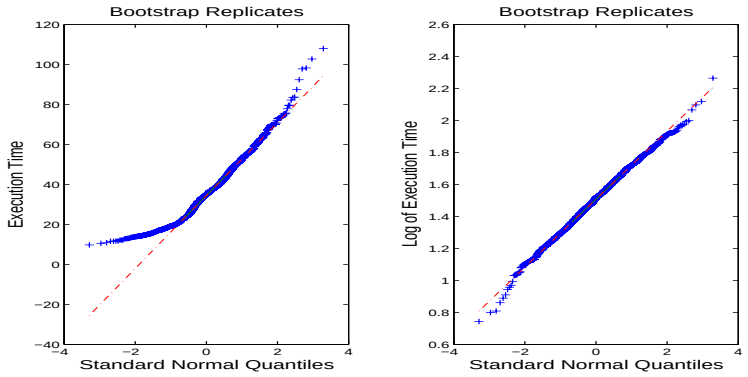


Figure 2.8: QQplots of bootstrap replicates of the estimator of the mean for the file transfer data in Figure 2.4. The bootstrap replicates of the data are not normally distributed, but those of the log of the data are.

of the data and of the log of the data. For the original data, the bootstrap replicates do not appear to be normal, thus the asymptotic result in Theorem 2.3.2 does not apply. It is the opposite for the log of the data.

QUESTION 2.4.1. *Compare this finding to the confidence intervals found in Figure 2.4.* [3]

2.4.4 VERIFYING THE IID ASSUMPTION

In many cases, the IID assumption can be verified by screening the method by which the data is produced, as discussed in Section 2.2.2. If there is some doubt, the following methods can be used:

1. (Visual Inspection of ACF Plot): If the data appears to be stationary (no trend, no seasonal component), then we can plot the sample autocorrelation coefficients, which are an estimate of the true autocorrelation coefficients ρ_k (defined in Equation (2.25). If the data is iid, then $\rho_k = 0$ for $k \geq 1$, and the sample autocorrelation coefficients fall within the values $\pm 1.96/\sqrt{n}$ (where n is the sample size) with 95% probability. An autocorrelation plot displays these bounds as well. A visual inspection can determine if this assumption is valid. For example, on Figure 2.17 we see that there some autocorrelation in the first six diagrams but not in the last two.
2. (Visual Inspection of *Lag-Plot*): We can also plot the value of the data at time t versus at time $t + h$, for different values of h (lag plots). If the data is iid, the lag plots do not show any trend. On Figure 2.15 we see that there is a negative trend at lag 1.
3. (Turning Point Test): A test provides an automated answer, but is sometimes less sure than a visual inspection. A test usually has a null hypothesis and returns a so called "p-value" (see Chapter 7 for an explanation). If the p-value is smaller than $\alpha = 1 - \gamma$, then the test rejects

[3] The figure shows the confidence intervals with the normal assumption and the bootstrap percentile estimates. With $n = 100$, the normal assumption (Theorem 2.3.1) and the asymptotic regime (Theorem 2.3.2) give practically the same result. Thus we expect the confidence intervals obtained with either the normal assumption or the asymptotic regime to be wrong for the data, and correct for the log of the data, consistent with Figure 2.4.

the null hypothesis at the confidence level γ. The turning point test, defined in Section 9.7.1 on Page 217, computes the number of times that the data goes from increasing to decreasing. This value should be close to $2/3$ if the data is iid. See Section 2.9.3 for an example.

2.5 PREDICTION INTERVAL

The confidence intervals studied before quantify the accuracy of a mean or median; this is useful for diagnostic purposes, for example we can assert from the confidence intervals on Figure 2.3 that the new option does reduce the run time, because the confidence intervals for the mean (or the median) are in the positive numbers.

Sometimes we are interested in a different viewpoint and would like to characterize the **variability** of the data: for example we would like to summarize what can be expected for an arbitrary future (non observed) transaction. Clearly, this run time is random. A *prediction interval* at level γ is an interval that we can compute by observing a realization of $X_1, ..., X_n$ and such that, with probability γ, a future transaction will have a run time in this interval. Intuitively, if the common cdf of all X_is would be known, then a prediction interval would simply be an inter-quantile interval, for example $[m_{\alpha/2}, m_{1-\alpha/2}]$, with $\alpha = 1 - \gamma$. For example, if the distribution is normal with known parameters, a prediction interval at level 0.95 would be $\mu \pm 1.96\sigma$. However, there is some additional uncertainty, due to the fact that we do not know the distribution, or its parameters a priori, and we need to estimate it. The prediction interval capture both uncertainties. Formally, the definition is as follows.

DEFINITION 2.5.1. *Let $X_1, ..., X_n, X_{n+1}$ be a sequence of random variables. A prediction interval at level γ is an interval of the form $[u(X_1, ..., X_n), v(X_1, ..., X_n)]$ such that*

$$\mathbb{P}\left(u(X_1, ..., X_n) \leq X_{n+1} \leq v(X_1, ..., X_n)\right) \geq \gamma \tag{2.12}$$

Note that the definition does not assume that X_i is iid, however we focus in this chapter on the iid case (but see Section 2.9 for a discussion of the more general case). The trick is now to find functions u and v that are pivots, i.e. their distribution is known even if the common distribution of the X_is is not (or is not entirely known).

There is one general result, which applies in practice to sample sizes that are not too small ($n \geq 39$), which we give next.

2.5.1 PREDICTION FOR AN IID SAMPLE BASED ON ORDER STATISTIC

THEOREM 2.5.1 (General Case). *Let $X_1, ..., X_n, X_{n+1}$ be an iid sequence and assume that the common distribution has a density. Let $X^n_{(1)}, ..., X^n_{(n)}$ be the order statistic of $X_1, ..., X_n$. For $1 \leq j \leq k \leq n$:*

$$\mathbb{P}\left(X^n_{(j)} \leq X_{n+1} \leq X^n_{(k)}\right) = \frac{k-j}{n+1} \tag{2.13}$$

thus for $\alpha \geq \frac{2}{n+1}$, $[X^n_{(\lfloor (n+1)\frac{\alpha}{2} \rfloor)}, X^n_{(\lceil (n+1)\left(1-\frac{\alpha}{2}\right) \rceil)}]$ is a prediction interval at level at least $\gamma = 1 - \alpha$.

For example, with $n = 999$, a prediction interval at level 0.95 ($\alpha = 0.05$) is $[X_{(25)}, X_{(975)}]$. This theorem is similar to the bootstrap result in Section 2.3.3, but is exact and much simpler.

Proof. transform X_i into $U_i = F(X_i)$ which is iid uniform. For uniform RVs, use the fact that $\mathbb{E}(U_{(j)}) = \frac{j}{n+1}$ (Chapter 12). Then

$$\begin{aligned}
&\mathbb{P}\left(U^n_{(j)} \le U_{n+1} \le U^n_{(k)} | U^n_{(1)} = u_{(1)}, ..., U^n_{(n)} = u_{(n)}\right) \\
&= \mathbb{P}\left(u_{(j)} \le U_{n+1} \le u_{(k)}\right) \\
&= u_{(k)} - u_{(j)}
\end{aligned}$$

The former is since U_{n+1} is independent of $(U_1, ..., U_n)$ and the latter since U_{n+1} has a uniform distribution on $[0, 1]$. Thus

$$\mathbb{P}\left(U^n_{(j)} \le U_{n+1} \le U^n_{(k)}\right) = \mathbb{E}\left(U^n_{(k)} - U^n_{(j)}\right) = \frac{k-j}{n+1}$$

□

QUESTION 2.5.1. *We have obtained n simulation results and use the prediction interval $[m, M]$ where m is the smallest result and M the largest. For which values of n is this a prediction interval at level at least 95% ?* [4]

For very small n, this result gives poor prediction intervals with values of γ that maybe far from 100%. For example, with $n = 10$, the best prediction we can do is $[x_{\min}, x_{\max}]$, at level $\gamma = 81\%$. If we can assume that the data is normal, we have a stronger result, shown next.

2.5.2 PREDICTION FOR AN IID SAMPLE, NORMAL CASE

THEOREM 2.5.2 (Normal iid Case). *Let $X_1, ..., X_n, X_{n+1}$ be an iid sequence with common distribution N_{μ,σ^2}. Let $\hat{\mu}_n$ and $\hat{\sigma}^2_n$ be as in Theorem 2.3.1. The distribution of $\sqrt{\frac{n}{n+1}}\frac{X_{n+1}-\hat{\mu}_n}{\hat{\sigma}_n}$ is Student's t_{n-1}; a prediction interval at level $1-\alpha$ is*

$$\hat{\mu}_n \pm \eta\sqrt{1 + \frac{1}{n}}\hat{\sigma}_n \tag{2.14}$$

where η is the $\left(1 - \frac{\alpha}{2}\right)$ quantile of the student distribution t_{n-1}.
For large n, an approximate prediction interval is

$$\hat{\mu}_n \pm \eta'\hat{\sigma}_n \tag{2.15}$$

where η' is the $\left(1 - \frac{\alpha}{2}\right)$ quantile of the normal distribution $N_{0,1}$.

For example, for $n = 100$ and $\alpha = 0.05$ we obtain the prediction interval (we drop the index n): $[\hat{\mu} - 1.99\hat{\sigma}, \hat{\mu} + 1.99\hat{\sigma}]$. Compare to the confidence interval for the mean given by Theorem 2.3.1 where the width of the interval is $\approx 10 = \sqrt{n}$ times smaller. For a large n, the prediction interval is approximately equal to $\hat{\mu}_n \pm \eta\hat{\sigma}_n$, which is the interval we would have if we ignore the uncertainty due to the fact that the parameters μ and σ are estimated from the data. For n as small as 26, the difference between the two is 7% and can be neglected in most cases.

[4] The interval is $[X_{(1)}, X_{(n)}]$ thus the level is $\frac{n-1}{n+1}$. It is ≥ 0.95 for $n \ge 39$. We need at least 39 samples to provide a 95% prediction interval.

Proof. First note that X_{n+1} is independent of $\hat{\mu}_n, \hat{\sigma}_n$. Thus $X_{n+1} - \hat{\mu}_n$ is normal with mean 0 and variance

$$\text{var}(X_{n+1}) + \text{var}(\hat{\mu}_n) = \sigma^2 + \frac{1}{n}\sigma^2$$

Further, $\hat{\sigma}_n/\sigma^2$ has a χ^2_{n-1} distribution and is independent of $X_{n+1} - \hat{\mu}_n$. By definition of Student's t, the theorem follows.

□

The normal case is also convenient in that it requires the knowledge of only two statistics, the mean $\hat{\mu}_n$ and the mean of squares (from which $\hat{\sigma}_n$ is derived).

Comment There is no "large n" result, like there is in Theorem 2.3.2: a prediction interval depends on the original distribution of the X_is, unlike confidence intervals for the mean that depend only on first and second moments due to the central limit theorem.

2.6 RESCALING

2.6.1 BOX-COX TRANSFORMATION

If we want to use Theorem 2.5.2, we need to make sure that the normal assumption holds (using for example a normal qqplot). If it does not, an alternative is to rescale the data, using a tranformation. In our context, a commonly used method is the *Box-Cox transformation* which often gives good results. It has one shape parameter s and is given by

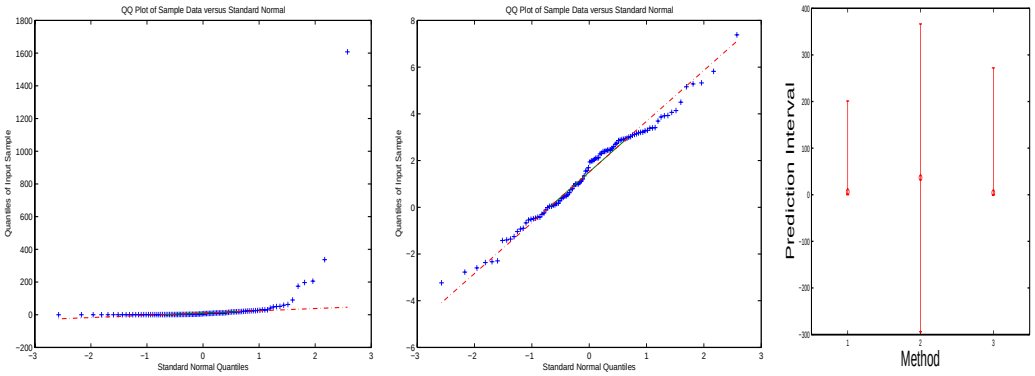
$$b_s(x) = \begin{cases} \frac{x^s - 1}{s} & , \ s \neq 0 \\ \ln x & , \ s = 0 \end{cases} \tag{2.16}$$

Commonly used parameters are $s = 0$ (log tranformation), $s = -1$ (inverse), $s = 0.5$ and $s = 2$.

EXAMPLE 2.8: FILE TRANSFER TIMES. (Continuation of Example 2.4 on page 22). Figure 2.9 shows the qq-plots of the file transfer times and their logs. It shows that the data is not normal but the log of the data is. The last panel shows 95%-prediction intervals. The left interval is obtained with the method of quantiles (Theorem 2.5.1); the middle one by (wrongly) assuming that the distribution is normal and applying Theorem 2.5.1 – it differs largely. The right interval is obtained with a log transformation. First, a prediction interval $[u(Y_1, ..., Y_n), v(Y_1, ...Y_n)]$ is computed for the transformed data $Y_i = \ln(X_i)$; the prediction interval is mapped back to the original scale to obtain the prediction interval $[\exp(u(\ln(X_1, ..., \ln(X_n))), \exp(v(\ln(X_1, ..., \ln(X_n)))]$. We leave it to the alert reader to verify that this reverse mapping is indeed valid. The left and right intervals are in good agreement, but the middle one is obviously wrong.

The prediction intervals also show the central values (with small circles). For the first one, it is the median. For the second one, the mean. For the last one, $\exp\left(\frac{\sum_{i=1}^n Y_i}{n}\right)$, i.e. the back transformed of the mean of the transformed data.

QUESTION 2.6.1. *The prediction intervals are not all symmetric around the central values. Explain why.* [5]



(a) (QQ-plot) (b) (QQ-plot of log of data) (c) (Prediction Intervals)

Figure 2.9: File transfer times for 100 independent simulation runs, with prediction intervals computed with the three methods discussed in Example 2.10 on page 34: (1) based on order statistics (2) based on mean and standard deviation (3) based on mean and standard deviation after re-scaling.

This example shows that it is is important to verify the normality assumption before applying formulae based on mean and standard deviation. If a Box Cox transformation is used, the optimal value of the exponent s can be done by visual inspection of qq-plots, or using the formal method described in Section 2.8.

2.6.2 HARMONIC, GEOMETRIC AND OTHER MEANS

The previous section illustrated that it may be more meaningful to rescale the data, for example with a Box-Cox transformation. Assume we transform a data set $x_1, ..., x_n$ by an invertible (thus strictly monotonic) mapping $b()$ into $y_1, ...y_n$, i.e. $y_i = b(x_i)$ and $x_i = b^{-1}(y_i)$ for $i = 1, ..., n$. We called *transformed sample mean* the quantity $b^{-1}(\frac{1}{n}\sum_{i=1}^{n} y_i)$, i.e. the back-transform of the mean of the transformed data. Similarly, the *transformed distribution mean* of the distribution of a random variable X is $b^{-1}(\mathbb{E}(b(X))$. When $b()$ is a Box-Cox transformation with index $s = -1, 0$ or 2 we obtain the classical following definitions, valid for a positive data set $x_i, i = 1..., n$ or a random variable X:

	Transformation	*Transformed Sample Mean*	*Transformed Distribution Mean*
Harmonic	$b(x) = 1/x$	$\frac{1}{\frac{1}{n}\sum_{i=1}^{n}\frac{1}{x_i}}$	$\frac{1}{\mathbb{E}(\frac{1}{X})}$
Geometric	$b(x) = \ln(x)$	$(\prod_{i=1}^{n} x_i)^{\frac{1}{n}}$	$e^{\mathbb{E}(\ln X)}$
Quadratic	$b(x) = x^2$	$\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}$	$\sqrt{\mathbb{E}(X^2)}$

[5]First interval: the distribution of the data is obviously not symmetric, so the median has no reason to be in the middle of the extreme quantiles. Second interval: by nature, it is strictly symmetric. Third interval: it is the exponential of a symmetric interval; exponential is not an affine transformation, so we should not expect the transformed interval to be symmetric.

THEOREM 2.6.1. *A confidence interval for a transformed mean is obtained by the inverse transformation of a confidence interval for the mean of the transformed data.*

For example, a confidence interval for the geometric mean is the exponential of a confidence interval for the mean of the logarithms of the data.

Proof. Let m' be the distribution mean of $b(X)$. By definition of a confidence interval, we have $\mathbb{P}(u(Y_1, ..., Y_n) < m' < v(Y_1, ..., Y_n)) \geq \gamma$ where the confidence interval is $[u, v]$. If $b()$ is increasing (like the Box-Cox transformation with $s \geq 0$) then so is $b^{-1}()$ and this is equivalent to $\mathbb{P}\left(b^{-1}(u(Y_1, ..., Y_n)) < b^{-1}(m') < b^{-1}(v(Y_1, ..., Y_n))\right) \geq \gamma$. Now $b^{-1}(m')$ is the transformed mean, which shows the statement in this case. If $b()$ is decreasing (like the Box-Cox transformation with $s < 0$) then the result is similar with inversion of u and v.

□

EXAMPLE 2.9: The right panel on Figure 2.4 shows confidence intervals for the geometric mean of the file transfer data.

We have seen in Example 2.10 on page 34 that a prediction interval for the original data can be obtained by reverse-transforming a prediction interval for the transformed data. In contrast, the results above show that this is not true for confidence intervals for the means. By reverse-transforming a confidence interval for the mean of the transformed data, we obtain a confidence interval for another type of mean (harmonic, etc.).

2.7 WHICH SUMMARIZATION TO USE ?

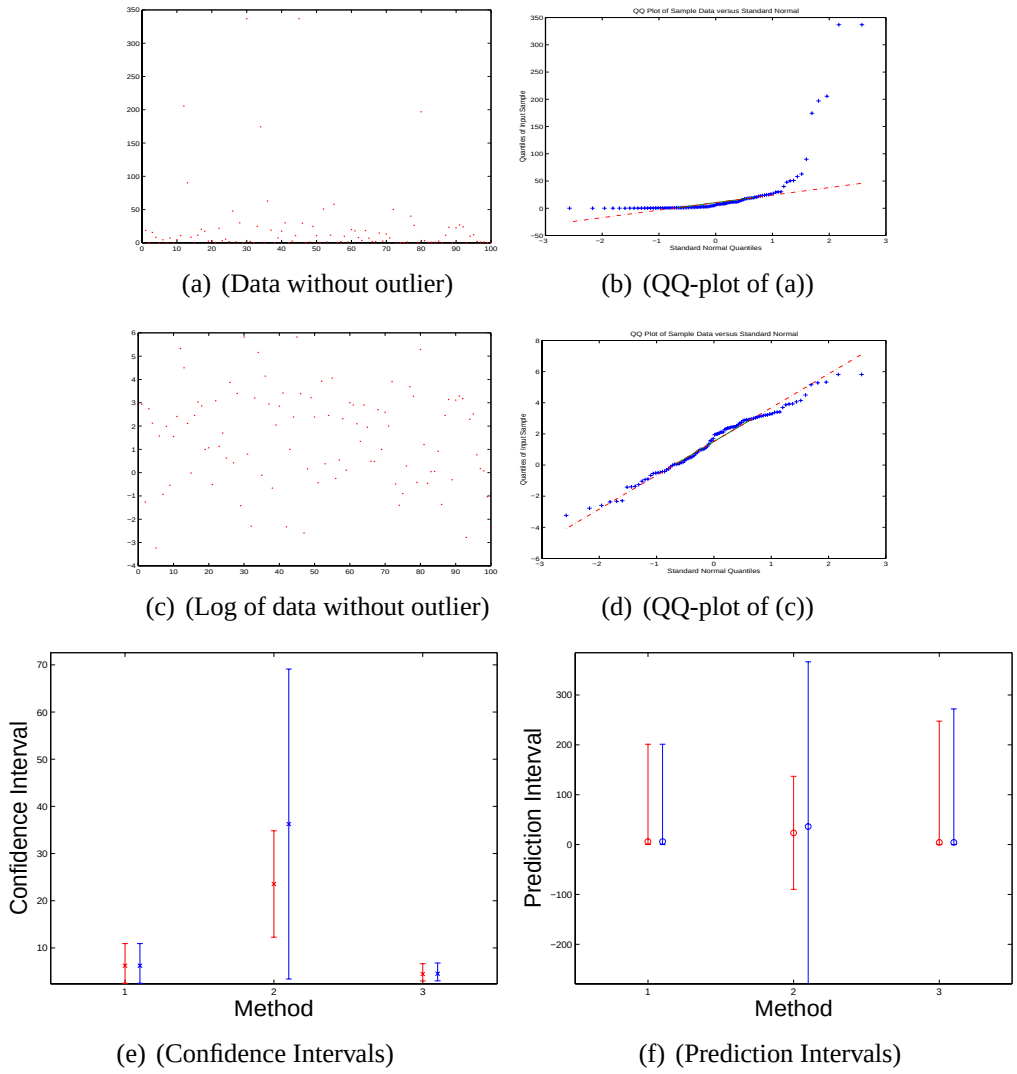
In the previous sections we have seen various summarization methods. In this section we discuss the use of these different methods.

The methods differ in their objectives: **confidence interval** for central value versus **prediction intervals**. The former quantify the accuracy of the estimated central value, the latter reflects how variable the data is. Both aspects are related (the more variable the data is, the less accurate the estimated central value is) but they are not the same.

The methods differ in the techniques used, and overlap to a large extend. They fall in two categories: methods based on the order statistic (Theorems 2.2.1 and 2.5.1) or based on mean and standard deviation (Theorems 2.3.1, 2.3.2, 2.5.2). The two methods differ in their **robustness versus compactness**.

2.7.1 ROBUSTNESS: OUTLIERS

Methods based on the order statistic are more robust to outliers. An *outlier* is a value that significantly differs from the average. The median and the prediction interval based on order statistic are not affected by a few outliers, contrary to the mean and the prediction interval based on mean and standard deviation, as illustrated by the following example.



(a) (Data without outlier) (b) (QQ-plot of (a))

(c) (Log of data without outlier) (d) (QQ-plot of (c))

(e) (Confidence Intervals) (f) (Prediction Intervals)

Figure 2.10: File transfer times for 100 independent simulation runs with outlier removed. Confidence intervals are without (left) and with (right) outlier, and with method (1) median (2) mean and (3) geometric mean. Prediction intervals are without (left) and with (right) outlier, computed with the three alternative methods discussed in Example 2.10 on page 34: (1) order statistics (2) based on mean and standard deviation (3) based on mean and standard deviation after re-scaling.

EXAMPLE 2.10: FILE TRANSFER WITH ONE OUTLIER. In fact in the data of Example 2.10 on page 34 there is one very large value, 5 times larger than the next largest value. One might be tempted to remove it, on the basis that such a large value might be due to measurement error. A qqplot of the data without this "outlier" is shown on Figure 2.10, compare to the corresponding qq-plot with the outlier in Figure 2.9 (b). The prediction intervals based on order statistics are not affected, but the one based on mean and standard deviation is completely different.

The outlier is less of an outlier on the re-scaled data (with the log transformation). The qqplot of the rescaled data is not affected very much, neither is the prediction interval based on mean and standard deviation of the rescaled data. Similarly, the confidence intervals for median and geometric mean are not affected, whereas that for the mean is.

In this example, we should not remove the outlier. In Section 8 we will see that such large values are normal and common in some cases. However, care should be taken to screen the data collection procedure for **true outliers**, namely values that are wrong because of measurement errors or problems.

The example illustrates the following facts:

- Outliers may affect the prediction and confidence intervals based on mean and standard deviation.
- This may go away if the data is properly rescaled. An outlier in some scale may not be an outlier in some other scale.
- In contrast, confidence intervals for the median and prediction intervals based on order statistics are more robust to outliers. They are not affected by re-scaling.

2.7.2 COMPACTNESS

Assume we wish to obtain both a central value with confidence interval and a prediction interval for a given data set. If we use methods based on order statistics, we will obtain a confidence interval for the median, and, say, a prediction interval at level 95%. Variability and accuracy are given by different sample quantiles, and cannot be deduced from one another. Furthermore, if we later are interested in 99% prediction intervals rather than 95%, we need to recompute new estimates of the quantiles.

In contrast, if we use methods based on mean and standard deviation, we obtain both confidence intervals and prediction intervals at any level with just 2 parameters (the sample mean and the sample standard deviation). In particular, the sample standard deviation gives indication on both accuracy of the estimator and variability of the data. However, as we saw earlier, these estimators are meaningful only in a scale where the data is roughly normal.

Also, mean and standard deviation are less complex to compute than estimators based on order statistics, which require sorting the data. In particular, mean and standard deviation can be computed incrementally online, by keeping only 2 counters (sum of values and sum of squares). This reason is less valid today than some years ago, since there are sorting algorithms with complexity $n \ln(n)$ but it may still be valid in some cases.

2.8 ⋆ PARAMETRIC ESTIMATION THEORY

The confidence intervals seen in the previous section are special cases of parametric estimation theory, which we shortly describe in this section. It can be skipped at first reading. The results of this section are used to compute confidence intervals in some cases where the simple methods described earlier do not apply.

2.8.1 THE PARAMETRIC ESTIMATION FRAMEWORK.

Consider a data set x_i, $i = 1..., n$, that we view as the realization of a stochastic system (in other words, the output of a simulator). The framework of parametric estimation theory consists

in assuming that θ is fixed, but unknown. We usually assume that the model has a density of probability, and that the density of probability that the output is $x_1, ..., x_n$ depends on the parameter θ; we denote it with $f(x_1, ..., x_n|\theta)$. It is also called the *likelihood* of the observed data. An *estimator* of θ is any function $T()$ of the observed data. A good estimator is one such that, in average, $T(X_1, ..., X_n)$ is "close" to the true value θ.

EXAMPLE 2.11: IID NORMAL DATA. Assume we can believe that our data is iid and normal with mean μ and variance σ^2.

QUESTION 2.8.1. *What is the likelihood ?* [6]

Here $\theta = (\mu, \sigma)$ and an estimator of θ is $\hat{\theta} = (\hat{\mu}_n, \hat{\sigma}_n)$ given by Theorem 2.3.1. Another, slightly different estimator is $\hat{\theta}_1 = (\hat{\mu}_n, s_n)$ given by Theorem 2.3.2.

An estimator provides a random result: for every realization of the data set, a different estimation is produced. The "goodness" of an estimator is captured by the following definitions. Here X is the random data set, $T(X)$ is the estimator and $\mathbb{E}_\theta$ means the expectation when the unknown but fixed parameter value is θ.

- *Unbiased estimator*: $\mathbb{E}_\theta(T(X)) = \theta$. For example, the estimator $\hat{\sigma}_n^2$ of variance of a normal iid sample given by Theorem 2.3.1 is unbiased.
- *Consistent family of estimators*: $\mathbb{P}_\theta(|T(X) - \theta|) > \epsilon) \to 0$ when the sample size n goes to ∞. For example, the estimator $(\hat{\mu}_n, \hat{\sigma}_n^2)$ of Theorem 2.3.1 is consistent. This follows from the weak law of large numbers.

2.8.2 MAXIMUM LIKELIHOOD ESTIMATOR (MLE)

A commonly used method for deriving estimators is that of *Maximum Likelihood*. The maximum likelihood estimator is the value of θ that maximizes the likelihood $f(x_1, ..., x_n|\theta)$. This definition makes sense if the maximum exists and is unique, which is often true in practice. A formal set of conditions is the regularity condition in Definition 2.8.1.

EXAMPLE 2.12: MLE FOR IID NORMAL DATA. Consider a sample $(x_1, ..., x_n)$ obtained from a normal iid random vector $(X_1, ..., X_n)$. The likelihood is

$$\frac{1}{\left(\sqrt{2\pi}\sigma\right)^n} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{\sigma^2}\right) \tag{2.17}$$

We want to maximize (2.17), where $x_1, ..., x_n$ are given and $\mu, v = \sigma^2$ are the variables. For a given σ, the maximum is reached when $\mu = \hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} x_i$. Let μ have this

[6] $\frac{1}{\left(\sqrt{2\pi}\sigma\right)^n} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{\sigma^2}\right)$

value and find the value of σ that maximizes the resulting expression, or to simplify, the log of it. We thus have to maximize

$$-n\ln(\sigma) - \frac{1}{2\sigma^2}S_{x,x} + ct \tag{2.18}$$

where ct is a constant with respect to σ and $S_{x,x} := \sum_{i=1}^{n}(x - \hat{\mu}_n)^2$. This is a simple maximization problem in one variable σ, which can be solved by computing the derivative. We find that there is a maximum for $\sigma^2 = \frac{S_{x,x}}{n}$. The maximum likelihood estimator of (μ, σ^2) is thus precisely the estimator in Theorem 2.3.2.

We say that an estimation method *invariant by re-parametrization* if the following holds. Assume the method produces some estimator $T(X)$ for θ. Assume we re-parametrize the problem by considering that the parameter is $\phi(\theta)$, where ϕ is some invertible mapping. For example, a normal iid sample can be parametrized by $\theta = (\mu, \sigma^2)$ or by $\phi(\theta) = (\mu, \sigma)$.

QUESTION 2.8.2. *What is the mapping ϕ in this case ?* [7]

The method is said invariant by re-parametrization if the estimator of $\phi(\theta)$ is $\phi(T(X))$. This means that the method always gives the same estimator, no matter how we decide to parametrize the model.

The maximum likelihood method *is* invariant by re-parametrization. This is because the property of being a amximum is invariant by re-parametrization. It is an important property in our context, since the model is usually not given a priori, but has to be invented by the performance analyst.

A method that provides an unbiased estimator cannot be invariant by re-parametrization, in general. For example, $(\hat{\mu}_n, \hat{\sigma}_n^2)$ of Theorem 2.3.1 is an unbiased estimator of (μ, σ^2), but $(\hat{\mu}_n, \hat{\sigma}_n)$ is a **biased** estimator of (μ, σ) (because usually $\mathbb{E}(S)^2 \neq \mathbb{E}(S^2)$ except if S is non-random). Thus, the property of being unbiased is incompatible with invariance by re-parametrization, and may thus be seen as an inadequate requirement for an estimator.

In Section 2.8.4, we give a result that shows that MLE for an iid sample with finite variance is asymptotically unbiased, i.e. the bias tends to 0 as the sample size increases. Further, it is consistent. Before that, we need to talk about efficiency.

2.8.3 EFFICIENCY AND FISHER INFORMATION

The *efficiency* of an estîmator $T(X)$ of the parameter θ is defined as the expected square error $\mathbb{E}_\theta(\|T(X) - \theta\|^2)$ (here we assume that θ takes values in some space Θ where the norm is defined). The efficiency that can be reached by an estimator is captured by the concept of Fisher information, that we define now.

Assume first to simplify that $\theta \in \mathbb{R}$. The *observed information* is defined by

$$J(\theta) = -\frac{\partial^2 l(\theta)}{\partial \theta^2}$$

where $l(\theta)$ is the *log-likelihood*, defined by

$$l(\theta) = \ln \text{lik}(\theta) = \ln f(x_1, ..., x_n|\theta)$$

[7] $\phi(x, y) = (x, \sqrt{y})$ defined for $x \in \mathbb{R}$ and $y \geq 0$.

The *Fisher information*, or *expected information* is defined by

$$I(\theta) = \mathbb{E}_\theta(J(\theta)) = \mathbb{E}_\theta\left(-\frac{\partial^2 l(\theta)}{\partial\theta^2}\right)$$

For an iid model $X_1, ..., X_n$, $l(\theta) = \sum_i \ln f_1(x_i|\theta)$ and thus $I(\theta) = nI_1(\theta)$, where $I_1(\theta)$ is the Fisher information for a one point sample X_1.

In general, the parameter θ is multi-dimensional, i.e., varies in an open subset Θ of $\mathbb{R}^k$. Then J and I are symmetric matrices defined by

$$[J(\theta)]_{i,j} = -\frac{\partial^2 l(\theta)}{\partial\theta_i\partial\theta_j}$$

and

$$[I(\theta)]_{i,j} = -\mathbb{E}_\theta\left(\frac{\partial^2 l(\theta)}{\partial\theta_i\partial\theta_j}\right)$$

The Cramer-Rao theorem says that the efficiency of any **unbiased** estimator is lower bounded by $\frac{1}{I(\theta)}$. Further, under the conditions in Definition 2.8.1, the MLE for an iid sample is asymptotically maximally efficient, i.e. $\mathbb{E}\left(\|T(X) - \theta\|\right)/I(\theta)$ tends to 1 as the sample size goes to infinity.

The Cramer-Rao lower bound justifies the name of "information". The variance of the MLE is of the order of the Fisher information: the higher the information, the more the sample tells us about the unknown parameter θ. The Fisher information is not the same as entropy, used in information theory. There are some (complicated) relations – see [CoverThomas91-book] chapter 16.

In the next section we give a more accurate result, that can be used to give approximate confidence intervals for large sample sizes.

2.8.4 ASYMPTOTIC CONFIDENCE INTERVALS

Here we need to assume some regularity conditions. Assume the sample comes from an iid sequence and further, that the following regularity conditions are met.

DEFINITION 2.8.1. *Regularity Conditions for Maximum Likelihood Asymptotics, [Davison02-book]*

1. *The set Θ of values of θ is compact (closed and bounded) and the true value θ_0 is not on the boundary.*
2. *(identifiability) for different values of θ, the densities $f(x|\theta)$ are different.*
3. *(regularity of derivatives) There exist a neighborhood B of θ_0 and a constant K such that for $\theta \in B$ and for all i, j, k, n : $\frac{1}{n}\mathbb{E}_\theta(|\partial^3 l_X(\theta)/\partial\theta_i\partial\theta_j\partial\theta_k|) \leq K$*
4. *For $\theta \in B$ the Fisher information has full rank*
5. *For $\theta \in B$ the interchanges of integration and derivation in $\int \frac{\partial f(x|\theta)}{\partial\theta_i}dx = \frac{\partial}{\partial\theta_i}\int f(x|\theta)dx$ and $\int \frac{\partial^2 f(x|\theta)}{\partial\theta_i\partial\theta_j}dx = \frac{\partial}{\partial\theta_i}\int \frac{\partial f(x|\theta)}{\partial\theta_j}dx$ are valid*

The following theorem is proven in [Davison02-book].

THEOREM 2.8.1. *Under the conditions in Definition 2.8.1, the MLE exists, converges almost surely to the true value. Further* $I(\theta)^{\frac{1}{2}}(\hat{\theta}-\theta)$ *converges in distribution towards a standard normal distribution, as* n *goes to infinity. It follows that, asymptotically:*

1. *the distribution of* $\hat{\theta}-\theta$ *can be approximated by* $N\left(0, I(\hat{\theta})^{-1}\right)$ *or* $N\left(0, J(\hat{\theta})^{-1}\right)$
2. *the distribution of* $2\left(l(\hat{\theta})-l(\theta)\right)$ *can be approximated by* χ^2_k *(where* k *is the dimension of* Θ*).*

The quantity $2\left(l(\hat{\theta})-l(\theta)\right)$ is called the *likelihood ratio statistic*.

Note. In the examples seen in this part of the course, the regulariy conditions are always satisfied, as long as : the true value θ lies within the interior of its domain, the derivatives of $l(\theta)$ are smooth (for example, if the density $f(x|\theta)$ has derivatives at all orders) and the matrices $J(\theta)$ and $I(\theta)$ have full rank.

If the regularity conditions hold, then we have an equivalent definition of Fisher information:

$$[I(\theta)]_{i,j} := -\mathbb{E}_\theta\left(\frac{\partial^2 l(\theta)}{\partial\theta_i\partial\theta_j}\right) = \mathbb{E}_\theta\left(\frac{\partial l(\theta)}{\partial\theta_i}\frac{\partial l(\theta)}{\partial\theta_j}\right)$$

this follows from differentiating with respect to θ the identity $\int f(\vec{x}\theta)d\vec{x}=1$.

Item 2 is more approximate than item 1, but does not require to compute the second derivative of the likelihood.

Theorem 2.8.1 also holds for non-iid cases, as long as the Fisher information goes to infinity with the sample size.

QUESTION 2.8.3. *Theorem 2.8.1 provides two asymptotic pivots. What are they ?* [8]

EXAMPLE 2.13: FISHER INFORMATION OF NORMAL IID MODEL. Assume $(X_i)_{i=1\dots n}$ is iid normal with mean μ and variance σ^2. The observed information matrix is computed from the likelihood function; we obtain:

$$J = \begin{pmatrix} \frac{n}{\sigma^2} & \frac{2n}{\sigma^3}(\hat{\mu}_n-\mu) \\ \frac{2n}{\sigma^3}(\hat{\mu}_n-\mu) & \frac{-n}{\sigma^2}+\frac{3}{\sigma^4}\left(S_{xx}+n(\hat{\mu}_n-\mu)^2\right) \end{pmatrix}$$

and the expected information matrix (Fisher's information) is

$$I = \begin{pmatrix} \frac{n}{\sigma^2} & 0 \\ 0 & \frac{2n}{\sigma^2} \end{pmatrix}$$

The following corollary is used in practice. It follows immediately from the theorem.

COROLLARY 2.8.1 (Asymptotic Confidence Intervals). *When* n *is large, approximate confidence intervals can be obtained as follows:*

1. *For the* i*th coordinate of* θ*, the interval is:* $\hat{\theta}_i \pm \eta\sqrt{\left[I(\hat{\theta})^{-1}\right]_{i,i}}$ *or* $\hat{\theta} \pm \eta\sqrt{\left[J(\hat{\theta})^{-1}\right]_{i,i}}$*, where* $N_{0,1}(\eta) = \frac{1+\gamma}{2}$ *(for example, with* $\gamma = 0.95$*,* $\eta = 1.96$*).*

[8] $I(\theta)^{\frac{1}{2}}(\hat{\theta}-\theta)$ and $2\left(l(\hat{\theta})-l(\theta)\right)$.

2. *If θ is in $\mathbb{R}$: the interval can be defined implicitly as $\{\theta \ : \ l(\hat{\theta}) - \frac{\xi}{2} \leq l(\theta) \leq l(\hat{\theta})\}$, where $\chi_1^2(\xi) = \gamma$. For example, with $\gamma = 0.95$, $\xi = 3.84$.*

EXAMPLE 2.14: ESTIMATE A PROBABILITY (THE OPINION POLL). We have developed a simulation scheme for wireless networks and simulated it. We run n independent, identically distributed simulations; each simulation run produces a binary output (success or failure of synchronization). We want to estimate the probability of success. The model is an iid sequence X_i, $i = 1, ..., n$, with $\mathbb{P}(X_i = 1) = p$ and $\mathbb{P}(X_i = 0) = 1 - p$. The parameter is p, and we want to estimate it. This is the same as estimating a confidence interval for the output of a binary opinion poll. We will compare the use of the second item of Corollary 2.8.1 to a direct method and to Theorem 2.3.2.

1. Likelihood Ratio Statistic (Corollary 2.8.1). The likelihood of the sample $x_1, ..., x_n$ is $p^k(1-p)^{n-k}$ where $k = \sum_{i=1}^n x_i$, and the log-likelihood is

$$l(p) = k \ln(p) + (n - k) \ln(1 - p)$$

It is maximum for $\hat{p} = \frac{k}{n}$ (in other words, the MLE of p is the frequency of success).

If n is large, we can use the second item of Corollary 2.8.1 and plot $l(p)$. A 95% confidence interval is the set of p defined by $l(p) \geq l(\hat{p}) - 1.92$. Figure 2.11 shows examples for various values of n. The resulting confidence intervals are shown on Table 2.1.

2. Direct Evaluation. We can compare to a direct evaluation. Let $T = \sum_{i=1}^n X_i$. The distribution of T is binomial. For $n \geq 30$ it is well approximated around its mean by the normal distribution with mean np and variance $np(1-p)$. Thus, a good approximation for the distribution of

$$\frac{1}{\sqrt{np(1-p)}} (T - np)$$

is the standard normal distribution $N_{0,1}$. Thus, with probability γ, we have approximately

$$\left| \frac{T - np}{\sqrt{np(1-p)}} \right| \leq \eta \tag{2.19}$$

with $N_{0,1}(\eta) = \frac{1+\gamma}{2}$. We are given a sample with $T = k$. A γ- confidence set is the set of values of p that satisfies Equation (2.19) where we take $T = k$.

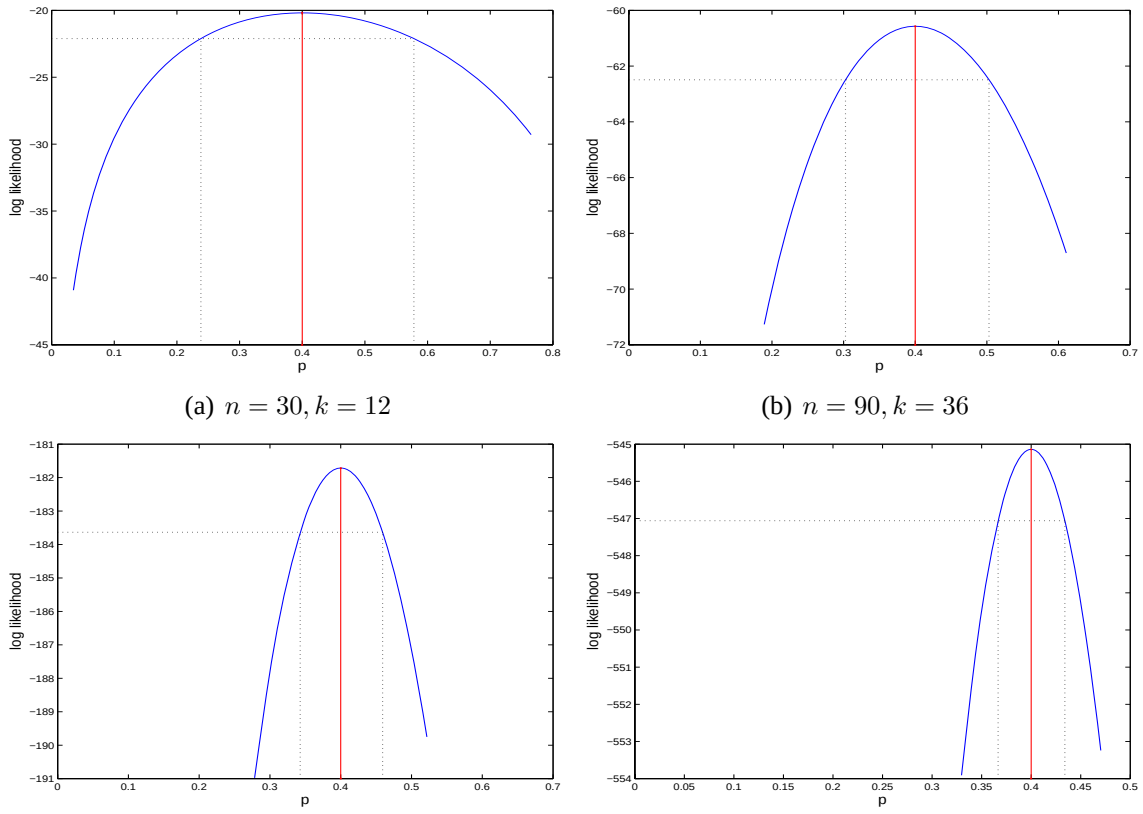
The function $p \to \frac{k-np}{\sqrt{np(1-p)}}$ is plotted on Figure 2.12. We see that is is a decreasing function of p. Thus the set defined implicitly by Equation (2.14) is an interval, and can simply be obtained numerically. The results are in Table 2.1.

3. Normal Approximation (Theorem 2.3.2). The estimator of the mean is $\hat{\mu}_n = \frac{T}{n}$ and the estimator of the variance is

$$S_n = \frac{1}{n} \sum_{i=1}^n X_i^2 - \hat{\mu}_n^2 = \hat{\mu}_n(1 - \hat{\mu}_n)$$

since $X_i^2 = X_i$. Thus an approximate confidence interval is given by

$$\left| \frac{T - np}{\sqrt{n\hat{\mu}_n(1 - \hat{\mu}_n)}} \right| \leq \eta$$



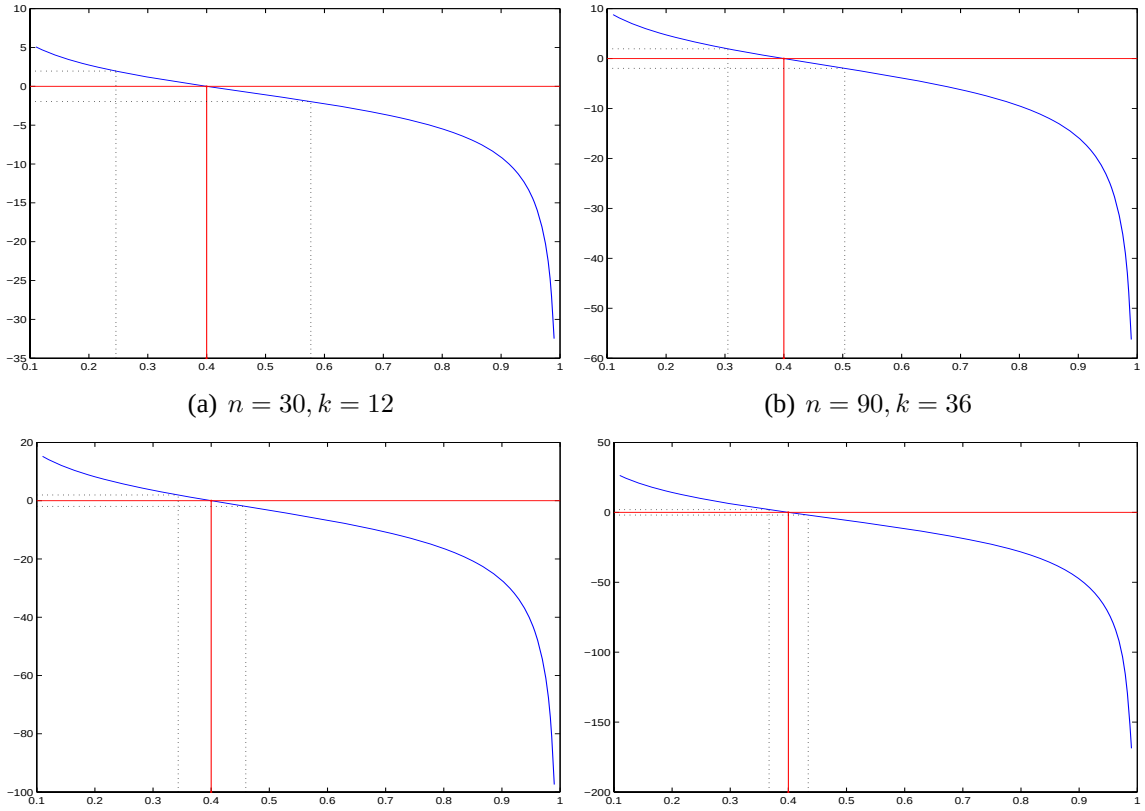
(a) $n = 30, k = 12$ (b) $n = 90, k = 36$

(c) $n = 270, k = 108$ (d) $n = 810, k = 324$

Figure 2.11: Log-likelihood as a function of the unknown parameter p (probability of success) in an experiment with n trials that produced k successes (Example 2.14 on page 40), and the resulting 95% confidence intervals for p on the x-axis. The MLE is $\hat{p} = 0.4$ for all cases.

n	Likelihood Ratio Statistic	Direct	Normal Approximation
30	$0.238 - 0.578$	$0.246 - 0.577$	$0.225 - 0.575$
90	$0.303 - 0.503$	$0.305 - 0.503$	$0.299 - 0.501$
270	$0.343 - 0.459$	$0.343 - 0.459$	$0.342 - 0.458$
810	$0.367 - 0.434$	$0.367 - 0.434$	$0.366 - 0.434$

Table 2.1: Comparison of 95% confidence intervals for p for Example 2.14 on page 40, for various values of n and $k = 0.4n$.



(a) $n = 30, k = 12$ (b) $n = 90, k = 36$

(c) $n = 270, k = 108$ (d) $n = 810, k = 324$

Figure 2.12: Computation of confidence interval by the Direct method in Example 2.14 on page 40.

The results are in Table 2.1. Compare with the direct method: the normal approximation replaces the fixed, but unknown value p by its estimator $\hat{\mu}_n$, thus we expect it to give good values for large n.

Conclusion. We see that all three methods coincide within 1% or less, for sample sizes as small as $n = 30$. The first method (likelihood ratio statistic) is simpler and more systematic, since all it requires is more general, as it applies to cases with more than one parameter, as we see in Section 2.8.5.

EXAMPLE 2.15: LAZY NORMAL IID. Assume our data comes from an iid normal model X_i, $i = 1, ...n$. We compare the exact confidence interval for the mean (from Theorem 2.3.1) to the approximate ones given by the corollary.

The MLE of (μ, σ) is $(\hat{\mu}_n, s_n)$. The exact confidence interval is

$$\hat{\mu}_n \pm \eta' \frac{\hat{\sigma}_n}{\sqrt{n}}$$

with $\hat{\sigma}_n^2 = S_{x,x}/(n-1)$ and $t_{n-1}(\eta') = \frac{1+\gamma}{2}$.

Now we compute the approximate confidence interval obtained from the Fisher information. We have

$$I(\mu, \sigma)^{-1} = \begin{pmatrix} \frac{\sigma^2}{n} & 0 \\ 0 & \frac{\sigma^2}{2n} \end{pmatrix}$$

thus the distribution of $(\mu - \hat{\mu}_n, \sigma - s_n)$ is approximately normal with 0 mean and covariance matrix $\begin{pmatrix} \frac{\sigma^2}{n} & 0 \\ 0 & \frac{\sigma^2}{2n} \end{pmatrix}$. It follows that $\mu - \hat{\mu}_n$ is approximately $N(0, \frac{s_n^2}{n})$, and an approximate confidence interval is

$$\hat{\mu}_n \pm \eta \frac{s_n}{\sqrt{n}}$$

with $s_n = s_{x,x}/n$ and $N_{0,1}(\eta) = \frac{1+\gamma}{2}$.

Thus the use of Fisher information gives the same asymptotic interval for the mean as Theorem 2.3.2. This is quite general: the use of Fisher information is the generalization of the large sample asymptotic of Theorem 2.3.2.

We can also compare the approximate confidence interval for σ. The exact interval is given by Theorem 2.3.1: with probability γ we have

$$\frac{\xi_2}{n-1} \leq \frac{\hat{\sigma}_2^n}{\sigma^2} \leq \frac{\xi_1}{n-1}$$

with $\chi^2_{n-1}(\xi_2) = \frac{1-\gamma}{2}$ and $\chi^2_{n-1}(\xi_1) = \frac{1+\gamma}{2}$. Thus an exact confidence interval for σ is

$$\hat{\sigma}_n \left[\sqrt{\frac{n-1}{\xi_1}}, \sqrt{\frac{n-1}{\xi_2}} \right] \tag{2.20}$$

With Fisher information, we have that $\sigma - s_n$ is approximately $N_{0, \frac{\sigma^2}{2n}}$ Thus with probability γ

$$|\sigma - s_n| \leq \eta \frac{\sigma}{\sqrt{2n}}$$

n	30	60	120
Exact	$0.7964 - 1.3443$	$0.8476 - 1.2197$	$0.8875 - 1.1454$
Fisher	$0.7847 - 1.3162$	$0.8411 - 1.2077$	$0.8840 - 1.1401$

Table 2.2: Confidence Interval for σ for an iid, normal sample of n data points by exact method and asymptotic result with Fisher information (Corollary 2.8.1). The values are the confidence bounds for the ratio $\frac{\sigma}{\hat{\sigma}_n}$ where σ is the true value and $\hat{\sigma}_n$ the estimated standard deviation as in Theorem 2.3.1.

with $N_{0,1}(\eta) = \frac{1+\gamma}{2}$.

Divide by σ and obtain, after some algebra, that with probability γ:

$$\frac{1}{1+\frac{\eta}{\sqrt{2n}}} \leq \frac{\sigma}{s_n} \leq \frac{1}{1-\frac{\eta}{\sqrt{2n}}}$$

Taking into account that $s_n = \sqrt{\frac{n-1}{n}}\hat{\sigma}_n$, we obtain the approximate confidence interval for σ

$$\hat{\sigma}_n \left[\sqrt{\frac{n-1}{n}} \frac{1}{1+\frac{\eta}{\sqrt{2n}}}, \sqrt{\frac{n-1}{n}} \frac{1}{1-\frac{\eta}{\sqrt{2n}}}, \right] \tag{2.21}$$

For $n = 30, 60, 120$ and $\gamma = 0.95$, the confidence intervals are as shown in Table 2.2; the difference is negligible already for $n = 30$.

QUESTION 2.8.4. *Which of the following are random variables:* $\hat{\theta}$, θ, $l(\theta)$, $l(\hat{\theta})$, $J(\theta)$, $I(\theta)$, $J(\hat{\theta})$, $I(\hat{\theta})$ *?* [9]

2.8.5 CONFIDENCE INTERVAL IN PRESENCE OF NUISANCE PARAMETERS

In many cases, the parameter has the form $\theta = (\mu, \nu)$, and we are interested only in μ (for example, for a normal model: the mean) while the remaining element ν, that still need to be estimated, is considered a nuisance (for example: the variance). In such cases, we can use the following theorem to find confidence intervals.

[9] In the classical, non Bayesian framework: $\hat{\theta}$, $l(\theta)$, $l(\hat{\theta})$, $J(\theta)$, $J(\hat{\theta})$, $I(\hat{\theta})$ are RVs. θ and $I(\theta)$ are non-random but unknown.

THEOREM 2.8.2 ([Davison02-book]). *Under the conditions in Definition 2.8.1, assume that $\Theta = M \times N$, where M, N are open subsets of $\mathbb{R}^p, \mathbb{R}^q$. Thus the parameter is $\theta = (\mu, \nu)$ with $\mu \in M$ and $\nu \in N$ (p is the "dimension", or number of degrees of freedom, of μ).*
For any μ, let $\hat{\nu}_\mu$ be the solution to

$$l(\mu, \hat{\nu}_\mu) = \max_\nu l(\mu, \nu)$$

and define the profile log likelihood pl by

$$pl(\mu) := \max_\nu l(\mu, \nu) = l(\mu, \hat{\nu}_\mu)$$

Let $(\hat{\mu}, \hat{\nu})$ be the MLE. If (μ, ν) is the true value of the parameter, the distribution of $2\left(pl(\hat{\mu}) - pl(\mu)\right)$ tends to χ^2_p.
An approximate confidence region for μ at level γ is

$$\{\mu \in M : pl(\mu) \geq pl(\hat{\mu}) - \frac{1}{2}\xi\}$$

where $\chi^2_p(\xi) = \gamma$.

The theorem essentially says that we can find an approximate confidence interval for the parameter of interest μ by computing the profile log-likelihood for all values of μ around the estimated value. The estimated value is the one that maximizes the profile log-likelihood. The profile log likelihood is obtained by fixing the parameter of interest μ to some arbitrary value and compute the MLE for the other parameters. A confidence interval is obtained implicitly as the set of values of μ for which the profile log likelihood is close to the maximum. In practice, all of this is done numerically.

EXAMPLE 2.16: LAZY NORMAL IID REVISITED. Consider the log of the data in Figure 2.4, which appears to be normal. The model is $Y_i \sim iidN_{\mu,\sigma^2}$ where Y_i is the log of the data. Assume we would like to compute a confidence interval for μ but are too lazy to apply the exact student statistic in Theorem 2.3.1.

For any μ, we estimate the nuisance parameter σ, by maximizing the log-likelihood:

$$l(\mu, \sigma) = -\frac{1}{2}\left(n \ln \sigma^2 + \frac{1}{\sigma^2}\sum_i (Y_i - \mu)^2\right)$$

It comes

$$\hat{\sigma}^2_\mu = \frac{1}{n}\sum_i (Y_i - \mu)^2 = \frac{1}{n}S_{YY} + (\bar{Y} - \mu)^2$$

and thus

$$pl(\mu) := l(\mu, \hat{\sigma}_\mu) = -\frac{n}{2}(\ln \hat{\sigma}^2_\mu + 1)$$

On Figure 2.13 we plot $pl(\mu)$. We find $\hat{\mu} = 1.510$ as the point that maximizes $pl(\mu)$. A 95%-confidence interval is obtained as the set $\{pl(\mu) \geq pl(\hat{\mu}) - \frac{1}{2}3.84\}$. We obtain the interval $[1.106, 1.915]$. Compare to the exact confidence interval obtained with Theorem 2.3.1, which is equal to $[1.103, 1.918]$: the difference is negligible.

QUESTION 2.8.5. *Find an analytical expression of the confidence interval obtained with the profile log likelihood for this example and compare with the exact interval.* [10]

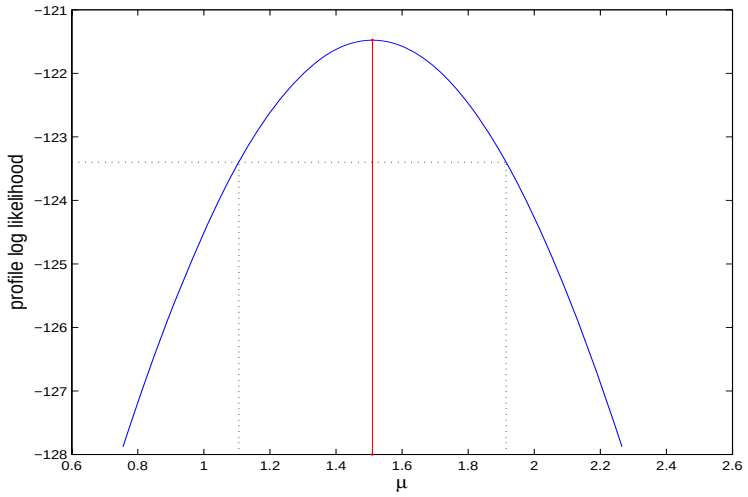


Figure 2.13: Profile log-likelihood for parameter μ of the log of the data in Figure 2.4. The confidence interval for μ is obtained by application of Theorem 2.8.2.

EXAMPLE 2.17: RE-SCALING. Consider the data in Figure 2.4, which does not appear to be normal in natural scale, and for which we would like to do a Box-Cox transformation. We would like a confidence interval for the exponent of the transformation.

The transformed data is $Y_i = b_s(X_i)$, and the model now assumes that Y_i is iid $\sim N_{\mu,\sigma^2}$. We take the unknown parameter to be $\theta = (\mu, \sigma, s)$. The distribution of X_i, under θ is:

$$f_{X_i}(x|\theta) = b_s'(x) f_{Y_i}(b_s(x)|\mu,\sigma) = x^{s-1} h(b_s(x)|\mu,\sigma^2)$$

where $h(x|\mu,\sigma^2)$ is the density of the normal distribution with mean μ and variance σ^2.

[10] The profile log likelihood method gives a confidence interval defined by

$$\frac{(\hat{\mu}-\mu)^2}{\frac{S_{YY}}{n}} \leq e^{\frac{\eta}{n}} - 1 \approx \frac{\eta}{n}$$

Let $t := \frac{\hat{\mu}-\mu}{\sqrt{\frac{S_{YY}}{n(n-1)}}}$ be the student statistic. The asymptotic confidence interval can be rewritten as

$$t^2 \leq (n-1)(e^{\frac{\eta}{n}} - 1) \approx \frac{\eta(n-1)}{n}$$

An exact confidence interval is

$$t^2 \leq \xi^2$$

where $\xi = t_{n-1}(1-\alpha/2)$. For large n, $\xi^2 \approx \eta$ and $\frac{n-1}{n} \approx 1$ so the two intervals are equivalent.

The log-likelihood is

$$l(\mu, \sigma, s) = C - n \ln \sigma + \sum_i \left((s-1) \ln x_i - \frac{(b_s(x_i) - \mu)^2}{2\sigma^2} \right)$$

where C is some constant (independent of the parameter). For a fixed s it is maximized by the MLE for a Gaussian sample

$$\hat{\mu}_s = \frac{1}{n} \sum_i b_s(x_i)$$

$$\hat{\sigma}_s^2 = \frac{1}{n} \sum_i (b_s(x_i) - \hat{\mu})^2$$

We can use a numerical estimation to find the value of s that maximizes $l(\hat{\mu}_s, \hat{\sigma}_s, s)$; see Figure 2.14 for a plot. The estimated value is $\hat{s} = 0.0041$, which gives $\hat{\mu} = 1.5236$ and $\hat{\sigma} = 2.0563$.

We now give a confidence interval for s, using the asymptotic result in Theorem 2.8.2. A 95% confidence interval is readily obtained from Figure 2.14, which gives the interval $[-0.0782, 0.0841]$.

QUESTION 2.8.6. *Does the confidence interval justify the log transformation ?* [11]

Alternatively, by Theorem 2.8.1, we can approximate the distribution of $\hat{\theta} - \theta$ by a centered normal distribution with covariance matrix $J(\hat{\theta})^{-1}$. After some algebra, we compute the Fisher information matrix. We compute the second derivative of the log-likelihood, and estimate the Fisher information by the observed information (i.e. the value of the second derivative at $\theta = \hat{\theta}$). We find:

$$J = \begin{pmatrix} 23.7 & 0 & -77.1 \\ 0 & 47.3 & -146.9 \\ 77.1 & -146.9 & 1291.1 \end{pmatrix}$$

and

$$J^{-1} = \begin{pmatrix} 0.0605 & 0.0173 & 0.0056 \\ 0.0173 & 0.0377 & 0.0053 \\ 0.0056 & 0.0053 & 0.0017 \end{pmatrix}$$

The last term of the matrix is an estimate of the variance of $\hat{s} - s$. The 0.95 confidence interval obtained from a normal approximation is $\hat{s} \pm 1.96\sqrt{0.0017} = [-0.0770, 0.0852]$.

2.9 NON INDEPENDENT SAMPLES

Often there is an a priori reason to believe that a data set was generated in an iid way: this is the case for independent simulation runs, or for a controlled experiment where all factors have been randomized. However, this is not always the case, for example for measurements collected during system operation. If there is suspicion that the data might not be iid, then the confidence intervals used in this chapter cannot be used. There is no simple rule for what to do in such a context. We first explain what the problem is by quantifying the bias, then we study two examples.

[11] Yes, since 0 is in the interval.

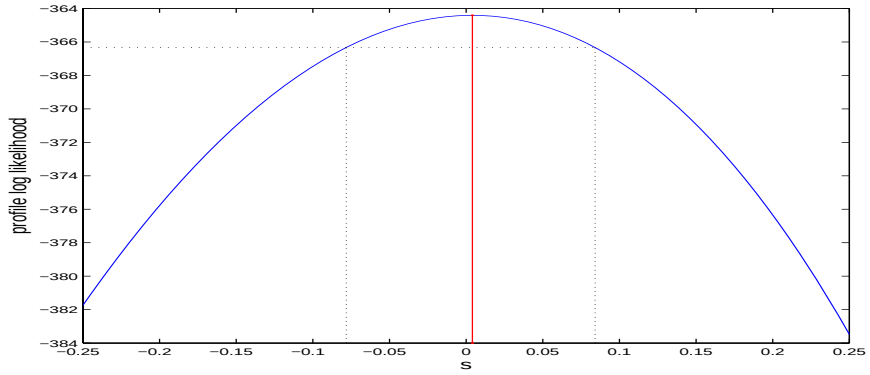


Figure 2.14: Profile log-likelihood for Example 2.17 on page 46, as a function of the Box-Cox exponent s. The maximum likelihood estimator of s is the value that maximizes the profile log likelihood: a confidence interval for s is the set of s for which the profile log likelihood is below the horizontal dashed line.

2.9.1 NON-IID BIAS

Assume we would like to estimate the mean μ of a sample $X_1, ..., X_n$, whose variance σ^2 is known. If the data would be iid, we would use the statistic $T = \sqrt{n}\frac{\bar{\mu}_n - \mu}{\sigma}$, the distribution of which is asymptotically centered normal with variance = 1 (with $\bar{\mu}_n = 1/n \sum_{t=1}^{n} X_t$). A 95%-confidence interval for the mean μ would be

$$I_{iid} = \bar{\mu}_n \pm 1.96\sigma/\sqrt{n} \tag{2.22}$$

Now assume the sample is not iid; the variance v of T is given by (Section 12.5.1)

$$v = \frac{1}{n\sigma^2} \sum_{(i,j)\in\{1,...,n\}^2} \Omega_{i,j} \tag{2.23}$$

where Ω is the covariance matrix of the sample, defined by

$$\Omega_{i,j} = \mathbb{E}\left(X_i X_j\right) - \mathbb{E}\left(X_i\right)\mathbb{E}\left(X_j\right) \tag{2.24}$$

Note that $\Omega_{i,i} = \sigma^2$. Further, in the iid case, $\Omega_{i,j} = 0$ for $i \neq j$, thus $v = 1$ as expected. Otherwise, v is not equal to 1.

Assume further that $\Omega_{i,j}$ depends only on the difference $|i - j|$ (for example because the process X_i is "second order stationary", see Chapter 9). It is usual to define the correlation ρ_k by:

$$\rho_k = \frac{\Omega_{i,i+k}}{\sigma^2} \tag{2.25}$$

The correlation ρ_k is number between -1 and 1. When it is positive, X_i and X_{i+k} tend to be similar, when it is negative, X_{i+k} tends to be small when X_i is large; it is 0 for all $k > 0$ when the process is iid.

By the change of variable $(i, j) \to (i, k)$, with $k = |i - j|$ in Equation (2.23), we obtain

$$v = 1 + 2\sum_{k=1}^{n-1}(1 - \frac{k}{n})\rho_k \tag{2.26}$$

Thus $v > 1$ if the data is positively correlated ($\rho_k > 0$), and vice-versa.

The central limit theorem still holds, provided that $\sum_{k\in\mathbb{N}} |\rho_k| < +\infty$, in which case the distribution of T is asymptotically $N(0, v)$ for large sample size.

A correct confidence interval is thus

$$I = \bar{\mu}_n \pm 1.96 v\sigma/\sqrt{n} \tag{2.27}$$

Compare this equation to Equation (2.22): the term v is the *non iid bias* for the confidence interval of the mean. If the process is positively correlated, the correct confidence interval is larger than obtained with the incorrect iid assumption, and vice-versa.

2.9.2 ⋆ EXAMPLE. JOE'S BALANCE DATA.

Joe's shop sells online access to visitors who download electronic content. At the end of day $t-1$, Joe's employee counts the amount of cash c_{t-1} present in the cash register and puts it into the safe. In the morning of day t, the cash amount c_{t-1} is returned to the cash register. The total amount of service sold (according to bookkeeping data) during day t is s_t. During the day, some amount of money r_t is sent to the bank. At the end of day t, we should have $c_t = c_{t-1} + s_t - r_t$. However, there are always small errors in counting the coins, in bookkeeping and in returning change. Joe computes the balance $Y_t = c_t - c_{t-1} - s_t + r_t$ and would like to know whether there is a systematic source of errors (i.e. Joe's employee is losing money, maybe because he is not honest, or because some customers are not paying for what they take).

The data for Y_t is shown on Figure 2.15. The sample mean is -13.95, which is negative. However, we need a confidence interval for μ before risking any conclusion.

CONFIDENCE INTERVAL FOR BALANCE ASSUMING IID MODEL. If we would assume that the errors Y_t are iid, then a confidence interval would be given by Theorem 2.3.2. In fact, the qqplot indicates that the data looks normal so we can use the student statistic in Theorem 2.3.1: the sample standard deviation is $S = 141.6$, so the 95%-confidence interval is $-13.95 \pm \eta S/\sqrt{n} \approx [-43, 15]$, where n is the sample size and $\eta = 1.986$. Thus, with the iid model, we cannot conclude that there is a fraud.

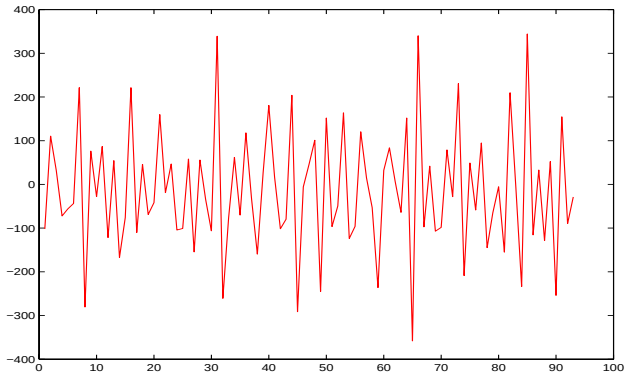
However, we need to verify the iid assumption before giving an interpretation. The data appears to be stationary (no trend or seasonal behaviour) thus we can use the ACF diagram. Figure 2.15 shows that there is a strong correlation at lag 1. This is confirmed by the lag plot. Thus, we can conclude that the iid assumption does not hold for this data set.

CONFIDENCE INTERVAL WITH MOVING AVERAGE MODEL. To go further, we need a valid model. Assume that the coin counting and bookkeeping processes have random, independent errors:

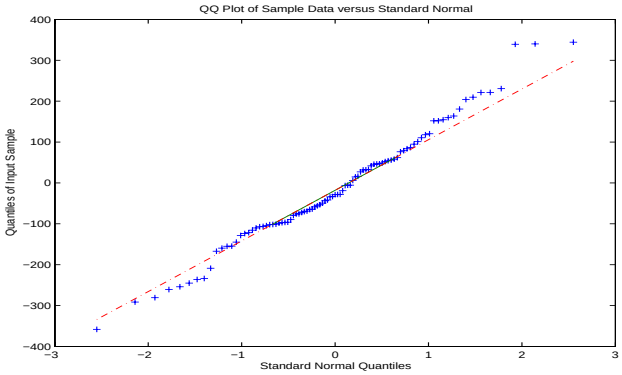
$$C_t = c_t + \epsilon_t \tag{2.28}$$

$$S_t - r_t = s_t - r_t + \epsilon'_t \tag{2.29}$$

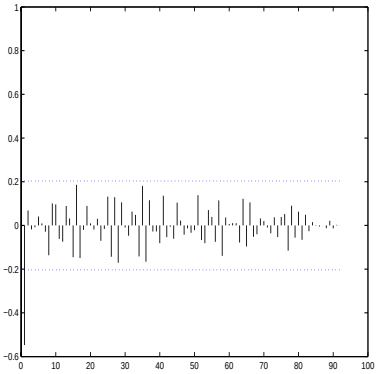
where upper case if for reported (observed) values and lower case for the true (non observed) values. Also assume that there is an external flow of money $\mu + \epsilon_t$" every day (a negative μ is a



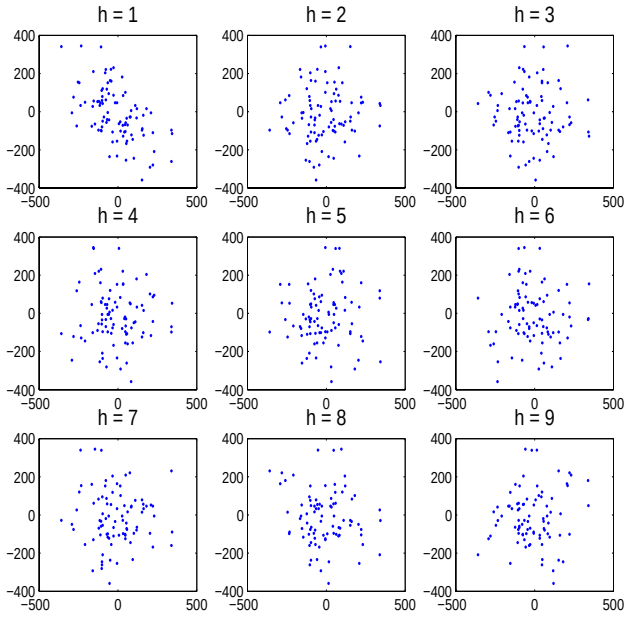
(a) Data



(b) Data



(c) Autocorrelation



(d) Lag Plots

Figure 2.15: Daily balance at Joe's wireless access shop over 93 days. The lag plots show $x(t)$ versus $x(t+h)$ where $x(t)$ is the time series in (a). The data appears to have some correlation at lag 1 and is thus clearly not iid.

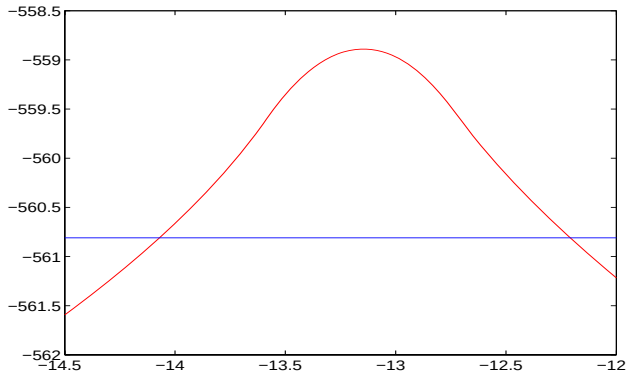


Figure 2.16: Profile Log Likelihood for the Moving Average model of Joe's balance data. The horizontal line is at a value $\eta/2 = 1.92$ below the maximum, with $\chi^2_1(\eta) = 0.95$; it gives an approximate confidence interval for the mean of the data on the x axis.

loss of money). Assume that all ϵs are iid and independent of each other. Then we have

$$Y_t := C_t - C_{t-1} = c_t - c_{t-1} - s_t + r_t + \epsilon_t - \epsilon_{t-1} - \epsilon'_t$$

and

$$c_t = c_{t-1} - s_t + r_t + \mu + \epsilon_t"$$

It follows that

$$Y_t = \mu + \epsilon_t" + \epsilon_t - \epsilon_{t-1} - \epsilon'_t$$

The auto-covariance of Y_t at lag $h \geq 2$ is 0 because all ϵs are independent of each other. Thus the model is compatible with the lag and auto-correlation plots in Figure **??**.

In Chapter 9, we study such processes. It is easy to see that $Y_t - \mu$ is stationary, gaussian and with 0 mean. Such processes that, in addition, have the property that the autocorrelation is 0 except at lags 0 and 1 are said to be moving average processes of order 1 (in short, MA(1)). Thus, $Y_t - \mu$ is an MA(1) process.

The general method of maximum likelihood estimation in Section 2.8 applies, as we see now. We are interested in obtaining a confidence interval for μ. We use the MLE asymptotic in Theorem 2.8.2 on Page 45.

Note. In Theorem 2.8.2, we saw that it applies to an iid model, which is not the case here; however, we can easily map our model to an iid one, as follows. The model can be written as $Y_t = \epsilon_t + \alpha\epsilon_{t-1}$ where ϵ_t is iid N_{0,σ^2}, with the convention that $Y_1 = \epsilon_1$. The random vector $\vec{Y}_n = (Y_1, ..., Y_n)^T$ is derived from the random vector $\vec{E}_n = (\epsilon_1, ..., \epsilon_n)^T$ by $\vec{Y}_n = HE + \vec{\mu}$ where $\vec{\mu} = (\mu, ..., \mu)^T$) and

$$H_n = \begin{pmatrix} 1 & 0 & 0 & ... & 0 \\ \alpha & 1 & 0 & ... & 0 \\ 0 & \alpha & 1 & ... & 0 \\ & & ... & & \\ 0 & 0 & ... & \alpha & 1 \end{pmatrix}$$

Let $\theta = (\mu, \sigma, \alpha)$ be the parameter of the model. Note that H_n is invertible and we also have $\vec{E}_n = H_n^{-1}(\vec{Y}_n - \vec{\mu})$. Thus we could imagine that we observe ϵ_t instead of Y_t. The log-likelihood of this derived model is the log of the density $f_{\vec{E}_n}(\epsilon|\theta)$. By the formula of change of variable, we have

$$f_{\vec{E}_n}(\epsilon|\theta) = |\det(H_n)|\, f_{\vec{Y}_n}(y|\theta)$$

Now $\det(H_n) = 1$ thus the log-likelihood of the derived model is the same as for the original model. Thus we can apply Theorem 2.8.2.

Here, it is plausible that the sample size is large enough. For any fixed μ, we compute the profile log-likelihood. It is obtained by fitting an MA(1) process to $W_t := Y_t - \mu$. Good statistical packages give not only the MLE fit, but also the log-likelihood of the fitted model, which is exactly the profile log-likelihood $pl(\mu)$. The MLE $\hat{\mu}$ is the value of μ that maximizes $pl(\mu)$, and $-2(pl(\hat{\mu}) - pl(\mu))$ is approximately χ^2_1. Figure 2.16 shows a plot of $pl(\mu)$. It follows that $\hat{\mu} = -13.2$ and an approximate 95%-confidence interval is $[-14.1, -12.2]$. Contrary to the iid model, this suggests that there *is* a loss of money, in average 13€ per day.

2.9.3 SUB-SAMPLING

If the data appears not ii, a solution may be to sub-sample, i.e. randomly select a very small fraction of the measured data, and verify that the iid assumption can be made for the selected data. The hope is that correlation disappears between data samples that are far apart. We verify that the sub sampled data is iid by the methods discussed in Section 2.4.4.

Sub-sampling means keeping only a fraction p of the data. A simple way would be to keep every pn data sample, where n is the total number of points, but this is not recommended as such a strict periodic sampling may introduce unwanted anomalies (called aliasing). A better method is to decide independently for each data point, with probability p, whether it is sub-sampled or not.

EXAMPLE 2.18: CPU DATA. Execution times for $n = 7632$ consecutive requests are measured and displayed on the upper left panel of Figure 2.17. The data appears stationary and roughly normal so the auto-correlation function can be used to test independence. The plot on the lower left panel of the figure shows a strong correlation. The sub-sampled data is obtained as follows. For every index $i = 1...n$, decide with probability $p = 1/2$ whether the point is kept. This gives the second plot on the figure. Then repeat the process. This gives sub-sampled data with $p = 1/2$ to $1/2^7 = 1/128$. The figure shows that the data looses correlation when the sampling probability is $p = 1/64$. The turning point test for the subsampled data with $p = 1/64$ has a p-value of 0.52648, thus at confidence level 0.95 we accept the null hypothesis, namely, the data is iid. The sub-sampled data has 114 points, and the confidence interval obtained from this for the mean of the sub-sampled data is $[65.5, 71.7]$, using the normal asymptotic formula of Theorem 2.3.1. Compare with the confidence interval that would be obtained if we would (wrongly) assume the data to be iid : $[69.2, 69.9]$. The iid assumption grossly underestimates the confidence interval because the data is positively correlated.

EXAMPLE 2.19: ETHERNET BYTE COUNT. The number of bytes transferred over an Ethernet local area network is shown on Figure 2.18 Figure 10.3 on Page 251. There are 360000 data points before sub-sampling. The data has correlation at all time scales, and sub-sampling cannot remove it: after sampling only one out of 1000

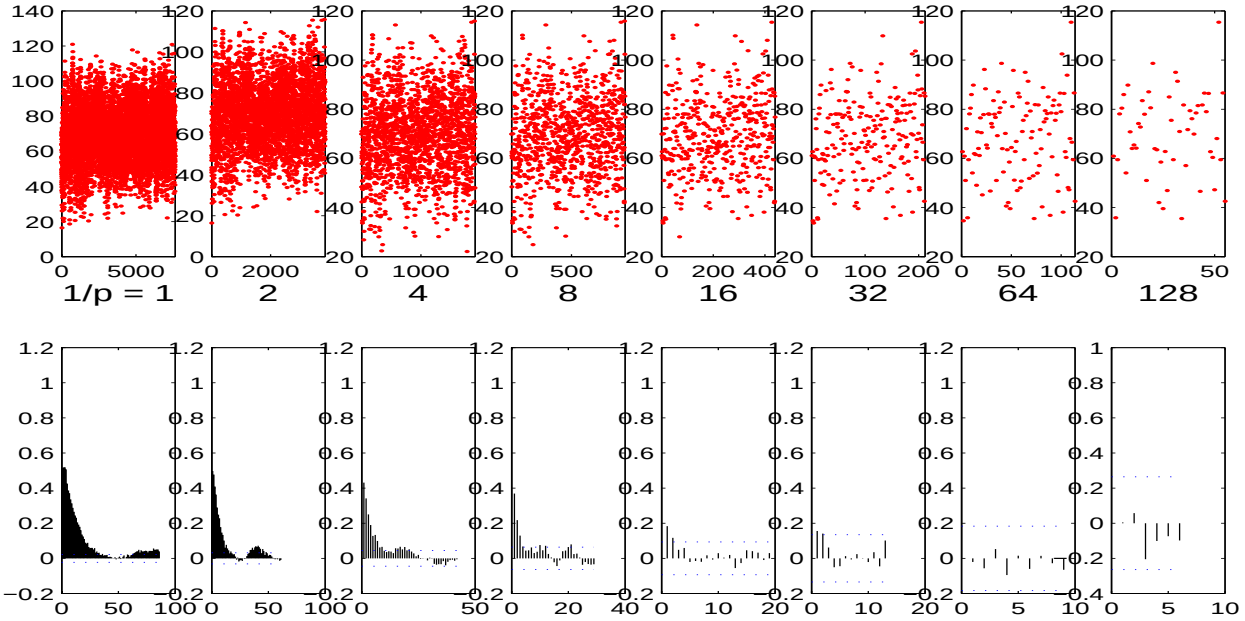


Figure 2.17: Execution times for $n = 7632$ requests (top left) and autocorrelation function (bottom left). and for the data sub-sampled with probability $p = 1/2$ to $1/2^7 = 1/128$. The data is correlated, but the sub-sampled data appears to be non-correlated for $p \geq 1/64$.

data points in average, there is still correlation. This is an example of long range dependent data. Estimating the mean of such a data set requires fitting it to a long range dependent model such as fractional arima (Chapter 10).

In summary, sub-sampling works well if the data has short range dependence.

2.10 OTHER ASPECTS OF CONFIDENCE/PREDICTION INTERVALS

2.10.1 INTERSECTION OF CONFIDENCE/PREDICTION INTERVALS

In some cases we have several confidence or prediction intervals for the same quantity of interest. For example, we can have a prediction interval I based on mean and standard deviation or I' based on order statistics. A natural deduction is to consider that the intersection $I \cap I'$ is a better confidence interval. This is almost true:

THEOREM 2.10.1. *If the random intervals I, I' are some confidence intervals at level $\gamma = 1 - \alpha$, $\gamma' = 1 - \alpha'$ then the intersection $I \cap I'$ is a confidence interval at level at least $1 - \alpha - \alpha'$. The same holds for prediction intervals.*

Proof. We do the proof for a confidence interval for some quantity θ, the proof is the same for a

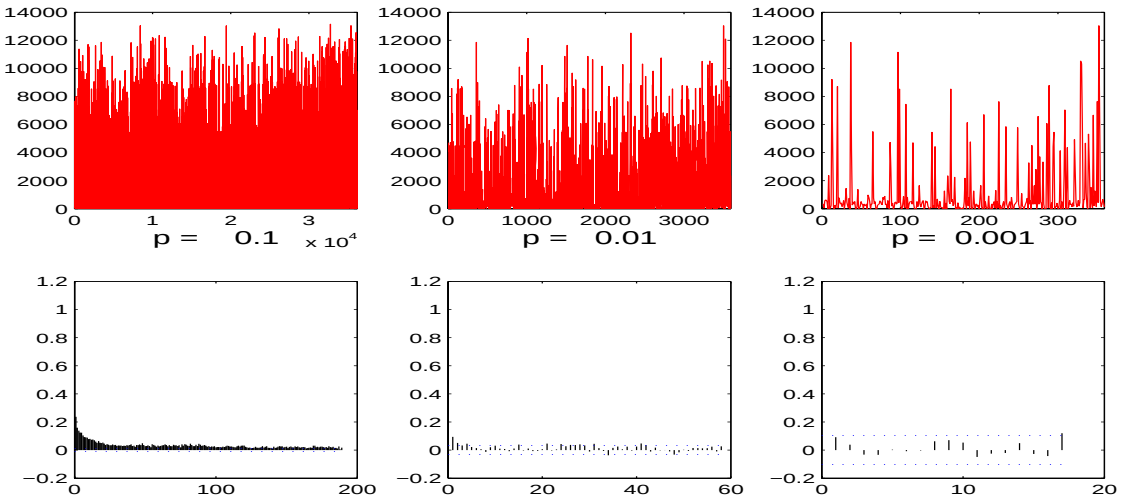


Figure 2.18: Ethernet Byte Counts of Figure 10.3 on Page 251, sub-sampled with probabilities 10^{-1} to 10^{-3} (top: data plots; bottom: auto-correlation). Sub-sampling does not remove the dependence.

prediction interval. By definition $\mathbb{P}(\theta \notin I) \leq \alpha$ and $\mathbb{P}(\theta \notin I') \leq \alpha'$. Thus

$$\mathbb{P}(\theta \notin I \cap I') = \mathbb{P}\left((\theta \notin I) \text{ or } (\theta \notin I')\right) \leq \mathbb{P}\left(\theta \notin I\right) + \mathbb{P}\left(\theta \notin I'\right)) \leq \alpha + \alpha'$$

□

EXAMPLE 2.20: FILE TRANSFER TIMES. (Continuation of Example 2.10 on page 34). We can compute two prediction intervals at level 0.975, using the order statistic method and the mean and standard deviation after rescaling (the prediction obtained without rescaling is not valid since the data is not normal). We obtain $[0.0394, 336.9]$ and $[0.0464, 392.7]$. We can conclude that a prediction interval at level 0.95 is $[0.0464, 336.9]$, which is better than the two.

Compare this interval to the prediction intervals at level 95% for each of the two methods; they are $[0.0624, 205.6]$ and $[0.0828, 219.9]$. Both are better.

Thus, for example if we combine two confidence intervals at level 97.5% we obtain a confidence interval at level 95%. As the example shows, this may be less good than an original confidence interval at level 95%.

QUESTION 2.10.1. *We estimate the mean of an iid data set by two different methods and obtain 2 confidence intervals at level* 95%*:* $I_1 = [2.01, 3.87]$, $I_2 = [2.45, 2.47]$. *Since the second interval is smaller, we discard the first and keep only the second. Is this a correct* 95% *confidence interval ?* [12]

[12] No, by doing so we keep the interval $I = I_1 \cap I_2$, which is a 90% confidence interval, not a 95% confidence interval.

2.10.2 THE MEANING OF CONFIDENCE

When we say that an interval I is a confidence interval at level 0.95 for some parameter θ, we mean the following. If we could repeat the experiment many times, in about 95% of the cases, the interval I would indeed contain the true value θ.

QUESTION 2.10.2. *Assume* 1000 *students independently perform a simulation of an M/M/1 queue with load factor* $\rho = 0.9$ *and find a* 95% *confidence interval for the result. The true result, unknown to these (unsophisticated) students is* 9*. The students are unsophisticated but conscientious, and all did correct simulations. How many of the 1000 students do you expect to find a wrong confidence interval, namely one that does* not *contain the true value ?* [13]

2.11 REVIEW

2.11.1 SUMMARY

1. A **confidence** interval is used to quantify the **accuracy** of a parameter estimated from the data.
2. For computing the central value of a data set, you can use either mean or median. Unless you have special reasons (see below) for not doing so, the median is a preferred choice as it is more robust. You should compute not only the median but also a confidence interval for it, using Table 14.1 on Page 338.
3. A **prediction** interval reflects the **variability** of the data. For small data sets ($n < 38$) it is not meaningful. For larger data sets, it can be obtained by Theorem 2.5.1.
4. A confidence interval for the mean characterizes both the **variability** of the data and the **accuracy** of the measured average. In contrast, a confidence interval for the median does not reflect well the variability of the data, therefore if we use the median we need both a confidence interval for the median and some measure of variability (the quantiles, as on a Box Plot). Mean and standard deviation give an accurate idea of the **variability** of the data, but only if the data is roughly normal. If it is not, it should be re-scaled using for example a Box-Cox transformation. Normality can be verified with a qq-plot.
5. The standard deviation gives an accurate idea of the **accuracy** of the mean if the data is normal, but also if the data set is large. The latter can be verified with a bootstrap method.
6. The geometric [resp. harmonic] mean is meaningful if the data is roughly normal in log [resp. $1/x$] scale. A confidence interval for the geometric [resp. harmonic] mean is obtained as the exponential [resp. inverse] of the mean in log [resp. $1/x$] scale.
7. All estimators in this chapter are valid only if the data points are independent (non correlated). This assumption must be verified, either by designing the experiments in a randomized way, (as is the case with independent simulation runs), or by formal correlation analysis as seen in the examples of Section 2.9. If the data set is correlated but very large, sub-sampling a small number of samples may be a solution.

Assume that we have obtained the outputs $x_1, ..., x_n$ from n independent replications. We want a confidence interval for the median and for the mean.

[13] Approximately 50 students should find a wrong interval.

CONFIDENCE INTERVAL FOR THE MEDIAN A confidence interval for the median is $[x_{(j)}, x_{(k)}]$, where $x_{(j)}$ is the jth value in ascending order. The values of j and k are taken from Table 14.1 on Page 338.

COMPUTING CONFIDENCE INTERVAL FOR THE MEAN

1. Test whether $x_1, ..., x_n$ roughly fits a normal distribution (visual test on qqplot).
2. If yes, apply the student t-statistic to obtain a confidence interval for the mean. The confidence interval is
$$\bar{x} \pm \eta \frac{s}{\sqrt{n}} \tag{2.30}$$
with $s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$ and $t_{n-1}(\eta) = \frac{1+\alpha}{2}$. Here, t_{n-1} is the student cdf with $n-1$ degrees of freedom and α is the confidence level (a typical value is $\alpha = 0.95$).
We are frequently in this case because each output x_i is often itself an average of many entities, and tends to be normally distributed.
3. Else (i.e. the sample $(x_1, ...x_n)$ does not appear to be normal), by the law of large numbers, $\bar{x}$ might still be normal, if n is large. The confidence interval is
$$\bar{x} \pm \eta \frac{s}{\sqrt{n}}$$
with $N_{0,1}(\eta) = \frac{1+\alpha}{2}$. If $n \geq 24$, the value of η is within 5% of that obtained by Equation (2.30).
Test whether n is large enough by the bootstrap method (Section 2.3.3). Do a qq-plot of the R bootstrap estimates T^r; if they appear to be normal, n is large enough.
4. Else (i.e. the sample $(x_1, ...x_n)$ does not appear to be normal and n is not large enough), use the bootstrap percentile estimate (Section 2.3.3).

2.11.2 REVIEW QUESTIONS

QUESTION 2.11.1. *Compare (1) the confidence interval for the median of a sample of n data values, at level 95% and (2) a prediction interval at level at least 95%, for $n = 9, 39, 99$.* [14]

QUESTION 2.11.2. *Call $L = \min\{X_1, X_2\}$ and $U = \max\{X_1, X_2\}$. We do an experiment and find $L = 7.4$, $U = 8.0$. Say which of the following statements is correct: (1) the probability of the event $\{L \leq \theta \leq U\}$ is 0.5 (2) the probability of the event $\{7.4 \leq \theta \leq 8.0\}$ is 0.5* [15]

QUESTION 2.11.3. *How do we expect a 90% confidence interval to compare to a 95% one ? Check this on the tables in Section 14.2.* [16]

[14] From the tables in Chapter 14 and Theorem 2.5.1 we obtain: (confidence interval for median, prediction interval): $n = 9$: $[x_{(2)}, x_{(9)}]$, impossible; $n = 39$: $[x_{(13)}, x_{(27)}]$, $[x_{(1)}, x_{(39)}]$; $n = 99$: $[x_{(39)}, x_{(61)}]$, $[x_{(2)}, x_{(97)}]$. The confidence interval is always smaller than the prediction interval.

[15] In the classical (non-Bayesian) framework, (1) is correct and (2) is wrong. There is nothing random in the event $\{7.4 \leq \theta \leq 8.0\}$, since θ is a fixed (though unknown) parameter. The probability of this event is either 0 or 1, here it happens to be 1. Be careful with the ambiguity of a statement such as "the probability that θ lies between L and U is 0.5". In case of doubt, come back to a probability space. The probability of an event can be interpreted as the ideal proportion of simulations that would produce the event.

[16] It should be smaller. If we take more risk we can accept a smaller interval. We can check that the values of j [resp. k] in the tables confidence intervals at level $\gamma = 0.95$ are larger [resp. smaller] than at confidence level $\gamma = 0.99$.

QUESTION 2.11.4. *A data set has 70 points. Give the formulae for confidence intervals at level 0.95 for the median and the mean* [17]

QUESTION 2.11.5. *A data set has 70 points. Give formulae for a prediction intervals at level 95%* [18]

QUESTION 2.11.6. *A data set $x_1, ...x_n$ is such that $y_i = \ln x_i$ looks normal. We obtain a confidence interval $[\ell, u]$ for the mean of y_i. Can we obtain a confidence interval for the mean of x_i by a transformation of $[\ell, u]$?* [19]

QUESTION 2.11.7. *Assume a set of measurements is corrupted by an error term that is normal, but positively correlated. If we would compute a confidence interval for the mean using the IID hypothesis, would the confidence interval be too small or too large ?* [20]

2.12 EXERCISES

EXERCISE 2.1. *$X_1, ..., X_n$ are drawn from a distribution $N(\mu, \sigma^2)$ with unknown parameters. We want to estimate μ with confidence level equal to 0.95. Let $\bar{X} = \frac{1}{n}\sum_i X_i$ and $S = \frac{1}{n-1}\sum_i \left(X_i - \bar{X}\right)^2$.*

1. *Using the t-statistic, what is the confidence interval, as a function of $\bar{X}$ and S ?*
2. *Assume we do the following approximation. We estimate σ by S, and do as though it was the true value. What confidence interval does this give ? Do a numerical comparison for $n = 5, 10, 20, 40, 80, 160$.*

EXERCISE 2.2. *Consider the example in Section **??** on Page **??**, item 3. For large n, what is the value of the confidence interval obtained by application of Theorem 2.8.1 item 2 ?*

EXERCISE 2.3. *Find confidence intervals for the M/D/1 simulation of H1.*

EXERCISE 2.4. *Getting Started with the S language: language basics, plots, arrays and functions.*

1. *Read the tutorial on S by Diego Kuonen, Sections 3, 4, 6 and 8. Additional documentation for those who would like to go further is*
   (a) *S-PLUS user guide Chapter 9 (Command Line Window)*
   (b) *S-PLUS programmer's guide Chapter 5,"Writing Functions", Section "Organizing Computations"*

[17] Median: from the table in Section 14.2 $[x_{(27)}, x_{(44)}]$. Mean: from Theorem 2.3.2: $\hat{\mu} \pm 0.2343S$ where $\hat{\mu}$ is the sample mean and S the sample standard deviation. The latter is assuming the normal approximation holds, and should be verified by either a qqplot or the bootstrap.

[18] From Theorem 2.5.1: $[\min_i x_i, \max_i x_i]$.

[19] No, we know that $[e^\ell, eu]$ is a confidence interval for the geometric mean, not the mean of x_i. In fact x_i comes from a log-normal distribution, whose mean is $e^{\mu + \frac{\sigma^2}{2}}$ where μ is the mean of the distribution of y_i, and σ^2 its variance.

[20] By an analog reasoning as in Section 2.9: too small. We underestimate the error. This phenomenon is known in physics under the term *personal equation*: if the errors are linked to the experimenter, they are positively correlated.

(c) *S-PLUS programmer 's guide, Chapter 8 "Traditional Graphics", Section "Frequently Used Plot Options"*

2. *Analyze the following program. Write and run it.*

```
# generates one qqplot of a normal sample
# with n elements
oneNormalSample<- function(n,...){
  qqnorm(rnorm(n),main=paste("Random Normal ",n," Samples"),...);
  abline(1)
  }

nbRows <- 3
par(mfrow=c(nbRows,nbRows))
indexArray <- c(0:(nbRows^2-1))
lapply(10*2^indexArray,oneNormalSample)
```

3. *Replace the last two lines by a for loop and run your new program.*

EXERCISE 2.5. *Write a program in Matlab that generates a sample of n iid standard normal variables, and display the corresponding histogram. Repeat the operation 9 times, for $n = 10, 20, 40, 80...$ and display the results on 3×3 panel.*

EXERCISE 2.6. *Plots and Distributions*

1. *Plot the densities of the following distributions: Normal(m, s), Student(n), Exponential(m).*
2. *Write a program which generates a sample of $n = 500$ RVs having a distribution in one of the above. Do it for all the distributions given above. Display the corresponding standard normal QQ-plots.*
3. *How do you interpret an S-shape in a normal QQ-plot ? A U-shape ?*

EXERCISE 2.7. *Exploratory Data Analysis.*

1. *Import the data of Table 1.3 by copying the 2 files indicated in a complementary document. There is one file for the first period (days 1 and 2) and one for the second period (days 181 and 182)*
2. *Do a visual display of the data: 4 plots on one page for each of the two data sets, showing the 5 values per plot.*
3. *(Factor Analysis and Box-Plot) Fix one factor (A/B, remote/local, period), and one value of it. Plot a box plot while varying values of other factors (box plot should have 4 values on x axis). Change the value of the fixed factor and redo the box plot. Repeat the same for other 2 factors.*
4. *Fix one value of factors A/B and period. Calculate means for remote and local. Do the same calculation for other values of factors A/B and first/second measurement. Do box plot with remote/local means on y axis and values of factors A/B and first/second measurement on x axis. Do you find that distance is an important factor for the system performance? Do the same plot by taking factor A/B on y axis. Do you find it important for the system performance?*

5. *Do you confirm the conclusions that were drawn in Chapter 2 ?*

EXERCISE 2.8. *Compute a confidence interval.*

1. *Import the values of achieved throughput that were used to build Figure 1.1(b) by copying the file indicated in a complementary document (this constitutes the set of y-value of Figure 1.1(b), in the order that they were measured, whereas Figure 1.1(b), not necessarily in the order shown on Figure 1.1(b)).*
2. *Plot the data.*
3. *Assume the data is the realization of a sequence of iid normal random variables. Find a confidence interval for the mean. Verify the validity of your model with visual tests of residues versus data and qqplot.*

USEFUL S-PLUS COMMANDS

- `dnorm, dt, dgamma, dexp ...`: densities of normal, student, gamma, exponential distributions
- `rnorm, rt ...`: random samples of these distributions
- `pnorm, pt, ...`: cumulative distribution function
- `qnorm, qt, ...`: quantile function (inverse of cumulative distribution function)
- `qqplot`: QQplot
- `plot.design, plot.factor, interaction.plot` exploratory data analysis
- `boxplot`: Box-Plot (showing mean, quantiles and extreme values)

USEFUL MATLAB COMMANDS

- `normpdf, gammapdf, chi2pdf ...`: densities of normal, gamma distributions
- `randn, chi2rnd, gammarnd`: random samples of these distributions
- `normcdf, gammacdf, ...`: cumulative distribution function
- `norminv, chi2inv ...`: quantile function (inverse of cumulative distribution function)
- `qqplot`: QQ-plot
- `boxplot`: Box-Plot (showing mean, quantiles and extreme values).
- `normfit`: confidence interval using t-statistic

CHAPTER 3

SIMULATION

Contents

3.1 WHAT IS A SIMULATION ?

A simulation is an experiment in the computer (biologists say “in silico”) where the real environment is replaced by the execution of a program.

EXAMPLE 3.1: MOBILE SENSORS. You want to build an algorithm A for a system of n wireless sensors, carried by mobile users, which send information to a central database. A simulation of the algorithm consists in implementing the essential features of the program in computer, with one instance of A per simulated sensor. The main difference between a simulation and a real implementation is that the real, physical world (here: the radio channel, the measurements done by sensors) is replaced by events in the execution of a program.

3.1.1 SIMULATED TIME AND REAL TIME

In a simulation the flow of time is controlled by the computer. A first task of your simulation program is to simulate parallelism: several parallel actions can take place in the real system; in your program, you serialize them. Serializing is done by maintaining a *simulated time*, which is the time at which an event in the real system is supposed to take place. Every action is then decomposed into instantaneous events (for example, the beginning of a transmission), and we assume that it is impossible that two instantaneous events take place exactly at the same time.

Assume for example that every sensor in Example 3.1 on page 62 should send a message whenever there is a sudden change in its reading, and at most every 10 minutes. It may happen in your simulation program that two or more sensors decide to send a message simultaneously, say within a window of 10 μs; your program may take much more than 10 μs of *real time* to execute these events. In contrast, if no event happens in the system during 5 minutes, your simulation program may jump to the next event and take just of few ms to execute 5 mn of simulated time. The real time depends on the performance of your computer (processor speed, amount of memory) and of your simulation program.

3.1.2 SIMULATION TYPES

There are many different types of simulations. We use the following classification.

DETERMINISTIC / STOCHASTIC. A deterministic simulation has no random components. It is used when we want to verify a system where the environment is entirely known, maybe to verify the feasibility of a schedule, or to test the feasibility of an implementation.

In most cases however, this is not sufficient. The environment of the system is better modelled with a random component, which makes the output of the simulation also random.

TERMINATING / NON-TERMINATING. A terminating simulation ends when specific conditions occurs. For example, if we would like to evaluate the execution time of one sequence of operations in a well defined environment, we can run the sequence in the simulator and count the simulated time. A terminating simulation is typically used when

- we are interested in the lifetime of some system
- or when the inputs are time dependent

EXAMPLE 3.2: JOE'S COMPUTER SHOP. We are interested in evaluating the time it takes to serve n customers who request a file together at time 0. We run a simulation program that terminates at time T_1 when all users have their request satisfied. This is a terminating simulation; its output is the time T_1.

ASYMPTOTICALLY STATIONARY / NON-STATIONARY. This applies to a non-terminating, stochastic simulation only. Stationarity is a property of the stochastic model being simulated. For an in-depth discussion of stationarity, see Chapter 11.

A stationary simulation is such that you gain no information about its age by analyzing it. For example, if you run a stationary simulation and take a snapshot of the state of the system at times 10 and 10'000 seconds, there is no way to tell which of the two snapshots is at time 10 or 10'000 seconds.

In practice, a non terminating simulation is rarely exactly stationary, but can be *asymptotically stationary*. This means that after some simulated time, the simulation becomes stationary.

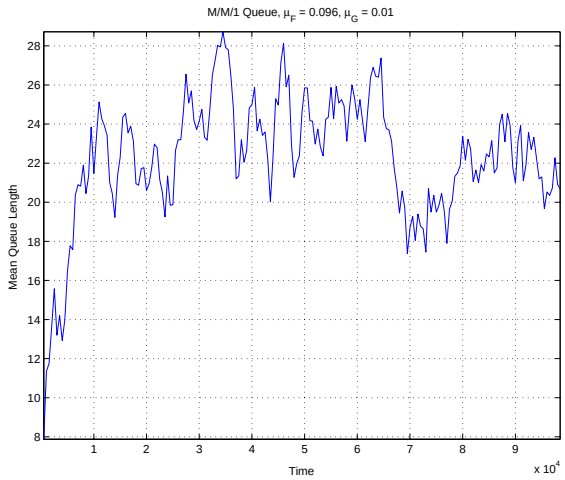
More precisely, a simulation program with time independent inputs can always be thought of as the simulation of a Markov chain. A Markov chain is a generic stochastic process such that, in order to simulate the future after time t, the only information we need is the state of the system at time t. This is usually what happens in a simulation program. The theory of Markov chains (see Chapter 11) says that the simulation will either converge to some stationary behaviour, or will diverge. If we want to measure the performance of the system under study, it is most likely that we are interested in its stationary behaviour.

EXAMPLE 3.3: INFORMATION SERVER. An information server is modelled as a queue. The simulation program starts with an empty queue. Assume the arrival rate of requests is smaller than the server can handle. Due to the fluctuations in the arrival process, we expect some requests to be held in the queue, from time to time. After some simulated time, the queue starts to oscillate between busy periods and idle periods. At the beginning of the simulation, the behaviour is not typical of the stationary regime, but after a short time it becomes so (Figure 3.1 (a)).

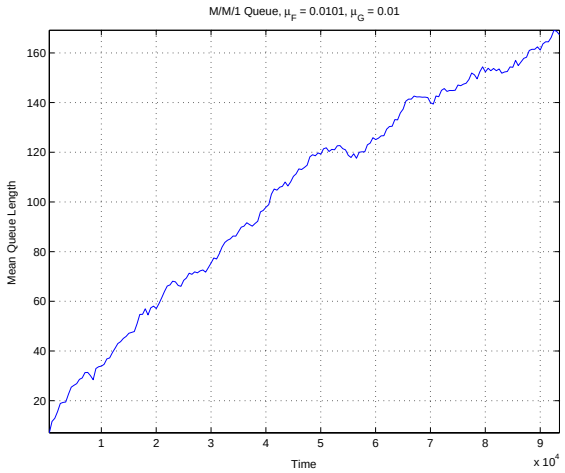
If in contrast the model is unstable, the simulation output may show a non converging behaviour (Figure 3.1 (b)).

In practice, there are two main reasons for non asymptotic stationarity.

1. **unstable** models: In a queuing system where the input rate is larger than the service capacity, the buffer occupancy grows unbounded. The longer the simulation is run, the larger the mean queue length is. Instead of growing unbounded, an unstable system may sometimes "freeze", like in the unstable random waypoint (Chapter 11).
2. models with **seasonal or growth** components, or more generally, time dependent inputs; for example: internet traffic grows month after month and is more intense at some times of the day. Simulations that incorporate such aspects are terminating simulations, for which the simulation duration is pre-defined.



(a) Utilization = 0.96



(b) Utilization = 1.01

Figure 3.1: Simulation of the information server in Example 3.3 on page 63, with exponential service and interarrival times. The graphs show the number of requests in the queue as a function of time, for two values of the utilization factor.

In most cases, when you perform a non-terminating simulation, you should make sure that your simulation is asymptotically stationary. Otherwise, the output of your simulation depends on the length of the simulation. It is not always easy, though, to know in advance whether a given simulation model is asymptotically stationary. Chapter 11 gives some examples.

QUESTION 3.1.1. *Among the following sequences* X_n

1. X_n, $n \geq 1$ *is iid*
2. X_n $n \geq 1$ *is drawn as follows.* X_1 *is sampled from a given distribution* $F()$*. To obtain* X_n, $n \geq 2$ *we first flip a coin (and obtain* 0 *with probability* $1-p$, 1 *with probability* p*). If the coin returns* 0 *we let* $X_n = X_{n-1}$*; else we let* $X_n =$ *a new sample from the distribution* $F()$*.*
3. $X_n = \sum_{i=1}^{n} Z_i$, $n \geq 1$, *where* Z_n, $n \geq 1$ *is an iid sequence*

say which ones are stationary. [1]

3.2 SIMULATION TECHNIQUES

There are many ways to implement a simulation program. We mention the two mostly used techniques in our context.

[1] 1. yes 2. yes: (X_1, X_2) has the same joint distribution as, for example (X_{10}, X_{11}). In general $(X_n, X_{n+1}, ..., X_{n+k})$ has the same distribution for all n. This is an example of non-iid, but stationary sequence. 3. No, in general. For example, if the common distribution $F()$ has a finite variance σ^2, the variance of X_n is $n\sigma^2$, and grows with n, which is contradictory with stationarity.

3.2.1 DISCRETE EVENT SIMULATION

Many computer and communication systems are often simulated using *discrete event simulation* for example with the ns2 simulator [ns2 home page]. It works as follows. The core of the method is to use a global time `currentTime` and an *event scheduler*. Events are objects that represent different transitions; all event have an associated firing time. The event scheduler is a list of events, sorted by increasing firing times. The simulation program picks the first event in the event scheduler, advances `currentTime` to the firing time of this event, and executes the event. The execution of an event may schedule new events with firing times $\geq$`currentTime`, and may change or delete events that were previously listed in the event scheduler. The global simulation time `currentTime` cannot be modified by an event. Thus, the simulation time jumps from one event firing time to the next – hence the name of discrete event simulation. In addition to simulating the logic of the system being modelled, events have to update the counters used for statistics.

EXAMPLE 3.4: DISCRETE EVENT SIMULATION OF A SIMPLE SERVER. A server receives requests and serves them one by one in order of arrival. The times between request arrivals and the service times are independent of each other. The distribution of the time between arrivals has cdf $F()$ and the service time has cdf $G()$. The model is in fact a *GI/GI/1* queue, which stands for general independent inter-arrival and service times. An outline of the program is given below. The program computes the mean response time and the mean queue length.

CLASSES AND OBJECTS We describe this example using an object oriented terminology, close to that of the Java programming language. All you need to know about object oriented programming to understand this example is as follows. An object is a variable and a class is a type. For example `arrival23` is the name of the variable that contains all information about the 23rd arrival, it is of the class `Arrival`. Classes can be nested, for example the class `Arrival` is a sub-class of `Event`. A method is a function whose definition depends on the class of the object. For example, the method `execute` is defined for all objects of the class `Event`, and is inherited by all subclasses such as `Arrival`. When the method `execute` is applied to the object `arrival23`, the actions that implement the simulation of an arrival are executed (for example, the counter of the number of requests in the system is incremented).

Global Variables and Classes

- `currentTime` is the global simulated time; it can be modified only by the main program.
- `eventScheduler` is the list of events, in order of increasing time.
- An event is an object of the class `Event`. It has an attribute `firingTime` which is the time at which it is to be executed. An event can be executed (i.e. the `Event` class has a method called `execute`), as described later.

  There are three `Event` subclasses: an event of the class `Arrival` represents the actions that occur when a request arrives; `Service` is when a request enters service; `Departure` is when a request leaves the system. The event classes are described in detail later.

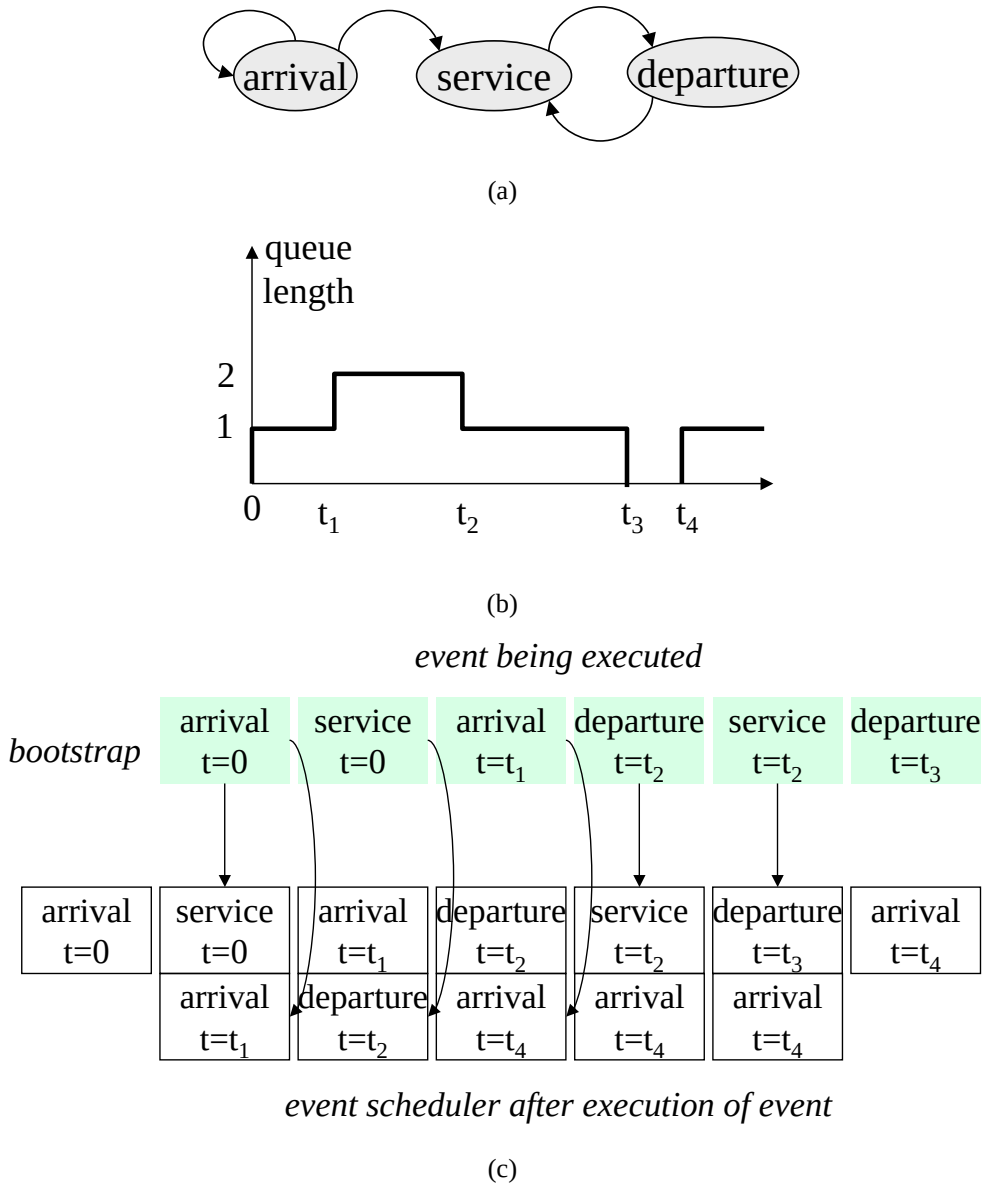


Figure 3.2: (a) Events and their dependencies for Example 3.4 on page 65. An arrow indicates that an event may schedule another one. (b) A possible realization of the simulation and (c) the corresponding sequence of event execution. The arrows indicate that the execution of the event resulted in one or several new events being inserted into the scheduler.

- The object `buffer` is the FIFO queue of Requests. The queue length (in number of requests) is `buffer.length`. The number of requests served so far is contained in the global variable `nbRequests`. The class `Request` is used to describe the requests arriving at the server. At a given point in time, there is one object of the class `Request` for every request present in the system being modelled. An object of the class `Request` has an arrival time attribute.
- Statistics Counters: `queueLengthCtr` is $\int_0^t q(s)ds$ where $q(s)$ is the value of `buffer.length` at time s and t is the current time. At the end of the simulation, the mean queue length is `queueLengthCtr`$/T$ where T is the simulation finish

time.

The counter `responseTimeCtr` holds $\sum_{m=1}^{m} R_m$ where R_m is the response time for the mth request and n is the value of `nbRequests` at the current time. At the end of the simulation, the mean response time is `responseTimeCtr`$/N$ where N is the value of `nbRequests`.

Event Classes. For each of the three event classes, we describe now the actions taken when an event of this class is "executed".

- `Arrival`: *Execute Event's Actions*. Create a new object of class `Request`, with arrival time equal to `currentTime`. Queue it at the tail of `buffer`.

  Schedule Follow-Up Events. If `buffer` was empty before the insertion, create a new event of class `Service`, with the same `firingTime` as this event, and insert it into `eventScheduler`.

  Draw a random number Δ from the distribution $F()$. Create a new event of class `Arrival`, with `firingTime` equal to this event `firingTime`$+\Delta$, and insert it into `eventScheduler`.

- `Service`: *Schedule Follow-Up Events*. Draw a random number Δ from the distribution $G()$. Create a new event of class `Departure`, with `firingTime` equal to this event's `firingTime`$+\Delta$, and insert it into `eventScheduler`.

- `Departure`: *Update Event Based Counters*. Let `c` be the request at the head of `buffer`. Increment `responseTimeCtr` by $d - a$, where d is this event's `firingTime` and a is the arrival time of the request `c`. Increment `nbRequests` by 1.

  Execute Event's Actions. Remove the request `c` from `buffer` and delete it.

  Schedule Follow-Up Events. If `buffer` is not empty after the removal, create a new event of class `Service`, with `firingTime` equal to this event's `firingTime`, and insert it into `eventScheduler`.

Main Program

- *Bootstrapping*. Create a new event of class `Arrival` with `firingTime` equal to 0 and insert it into `eventScheduler`.
- *Execute Events*. While the simulation stopping condition is not fulfilled, do the following.
  - *Increment Time Based Counters.* Let `e` be the first event in `eventScheduler`. Increment `queueLengthCtr` by $q(t_{\text{new}} - t_{\text{old}})$ where $q =$`buffer.length`, t_{new}=`e.firingTime` and $t_{\text{old}} =$`currentTime`.
  - Execute `e`.
  - Set `currentTime` to `e.firingTime`
  - Delete `e`
- *Termination*. Compute the final statistics:
  `meanQueueLength=queueLengthCtr/currentTime`
  `meanResponseTime=responseTimeCtr/nbRequests`

Figure 3.2 illustrates the program.

QUESTION 3.2.1. *Is the mean queue length an event-based or a time-based statistic ? The mean response time ?* [2]

QUESTION 3.2.2. *Can consecutive events have the same firing time ?* [3]

QUESTION 3.2.3. *What are the generic actions that are executed when an event is executed ?* [4]

QUESTION 3.2.4. *Is the model in Example 3.4 on page 65 stationary ?* [5]

3.2.2 STOCHASTIC RECURRENCE

This is another simulation method that applies to some classes of models. It is usually much more efficient than discrete event simulation, but applies only to relatively simple models.

We assume here that the system to be simulated can be put in the form of a stochastic recurrence, i.e. a recurrence of the form:

$$\begin{cases} X_0 = x_0 \\ X_{n+1} = f(X_n, Z_n) \end{cases} \tag{3.1}$$

where X_n is the state of the system at the nth transition (For any realization, X_n is in some possibly complicated state space $\mathcal{X}$), x_0 is a fixed, given state in $\mathcal{X}$, Z_n is some stochastic process that can be simulated (for example a sequence of iid random variables, or a Markov chain), and f is a deterministic mapping.

The simulated time T_n at which the nth transition occurs is assumed to be included in the state variable X_n.

EXAMPLE 3.5: RANDOM WAYPOINT.

The *random waypoint* is a model for a mobile point, and can be used to simulate the mobility pattern in Example 3.1 on page 62. It is defined as follows. The state variable is $X_n = (M_n, T_n)$ where M_n is the position of the mobile at the nth transition (the nth "waypoint") and T_n is the time at which this destination is reached. The point M_n is chosen at random, uniformly in a given convex area $\mathcal{A}$. The speed at which the mobile travels to the next waypoint is also chosen at random uniformly in $[v_{\min}, v_{\max}]$.

The random waypoint model can be cast as a stochastic recurrence by letting $Z_n = (M_{n+1}, V_{n+1})$, where M_{n+1}, V_{n+1} are independent i.i.d. sequences, such that M_{n+1} is uniformly distributed in $\mathcal{A}$ and V_{n+1} in $[v_{\min}, v_{\max}]$. We have then the stochastic recurrence

$$X_{n+1} := (M_{n+1}, T_{n+1}) = (M_{n+1}, T_n + \frac{\|M_{n+1} - M_n\|}{V_n})$$

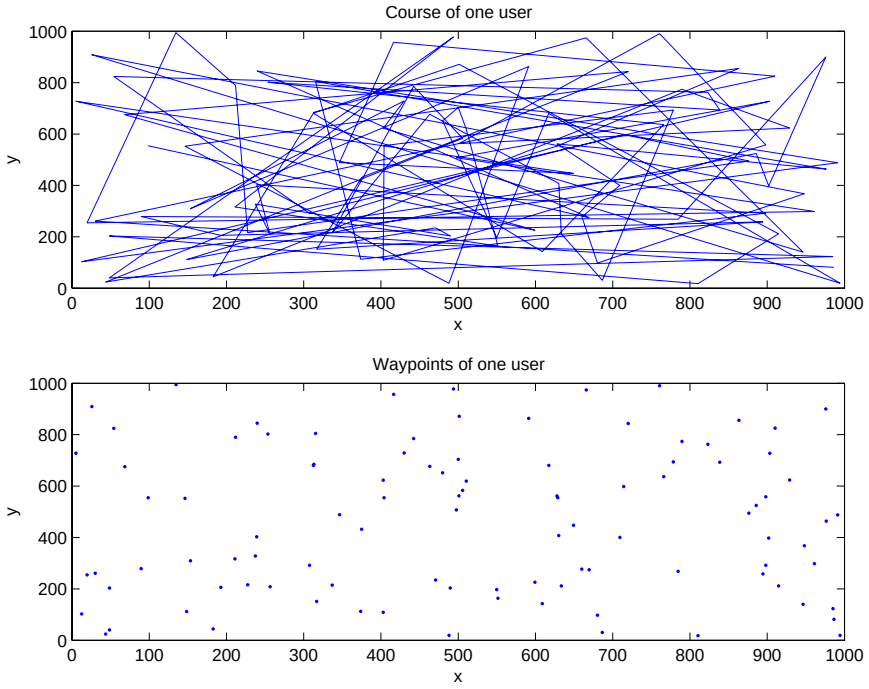
See Figure 3.3 for an illustration.

[2] Mean queue length: time based. Mean response time: event based.

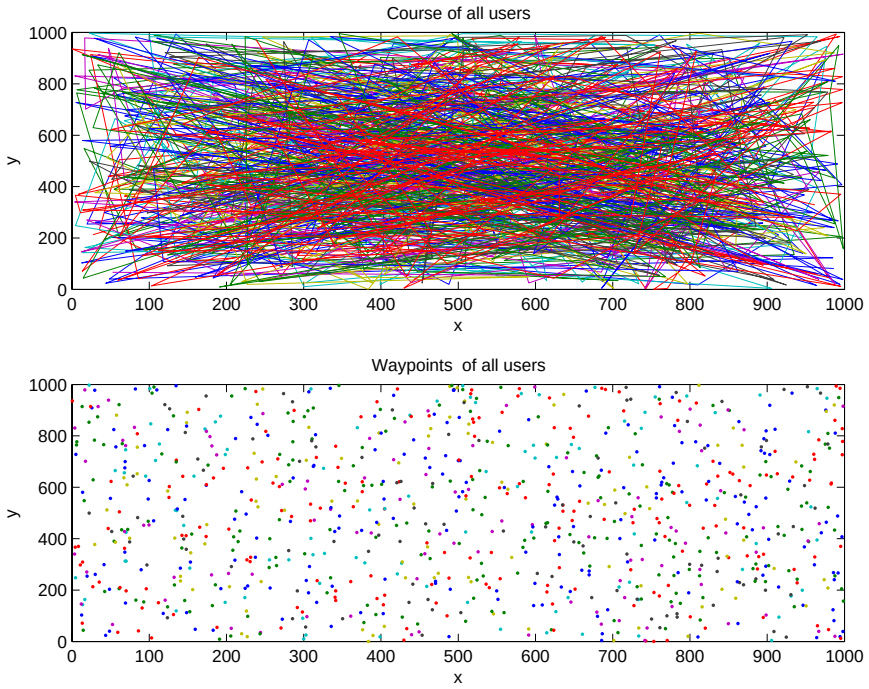
[3] Yes. In Example 3.4 on page 65, a `Departure` event when the queue is not empty is followed by a `Service` event with the same firing time.

[4] 1. Update Event Based Counters 2. Execute Event's Actions 3. Schedule Follow-Up Events.

[5] It depends on the parameters. Let a [resp. b] be the mean of $F()$ [resp. $G()$]. The utilization factor of the queue is $\rho = \frac{b}{a}$. If $\rho < 1$ the system is stable and thus asymptotically stationary, else not (see Chapter 6).



(a) 1 mobile



(b) 10 mobiles

Figure 3.3: Simulation of the random waypoint model.

Once a system is cast as a stochastic recurrence, it can be simply simulated as a direct implementation of Equation (3.1), for example in Matlab.

QUESTION 3.2.5. *Is the random waypoint model asymptotically stationary ?* [6]

STOCHASTIC RECURRENCE VERSUS DISCRETE EVENT SIMULATION It is always possible to express a stochastic simulation as a stochastic recurrence, but both representations may have very different memory and CPU requirements. Which representation is best depends on the problem at hand.

EXAMPLE 3.6: SIMPLE SERVER AS A STOCHASTIC RECURRENCE. (Continuation of Example 3.4 on page 65). Consider implementing the simple server in Example 3.4 on page 65 as a stochastic recurrence. To simplify, assume we are interested only in the mean queue length and not the mean response time. This can be implemented as a stochastic recurrence as follows.

Let X_n represent the state of the simulator just *after* an arrival or a departure, as follows:

$$X_n = (t_n, b_n, q_n, a_n, d_n)$$

with t_n = the simulated time at which this transition occurs, b_n =`buffer.length`, q_n = `queueLengthCtr` (both just after the transition), a_n = the time interval from this transition to the next arrival and d_n= the time interval from this transition to the next departure.

Let Z_n be a couple of two random numbers, drawn independently of anything else, with distribution uniform in $(0, 1)$.

The initial state is

$$t_0 = 0,\ b_0 = 0,\ q_0 = 0,\ a_0 = F^{-1}(u),\ d_0 = \infty$$

where u is a sample of the uniform distribution on $(0, 1)$. The reason for the formula $a_0 = F^{-1}(u)$ is explained in Section 3.6: a_0 is a sample of the distribution with cdf $F()$.

The recurrence is defined by $f((t, b, q, a, d), (z_1, z_2)) = (t', b', q', a', d')$ with

if $a < d$ // this transition is an arrival
  $\Delta = a$
  $t' = t + a$
  $b' = b + 1$
  $q' = q + b\Delta$
  $a' = F^{-1}(z_1)$
  if $b == 0$ then $d' = G^{-1}(z_2)$ else $d' = d - \Delta$
else // this transition is a departure
  $\Delta = d$
  $t' = t + d$
  $b' = b - 1$

[6] For $v_{\min} > 0$ it is asymptotically stationary. For $v_{\min} = 0$ it is not: the model "freezes" (the number of waypoints per time unit tends to 0). See Chapter 11 for a justification).

$$q' = q + b\Delta$$
$$a' = a - \Delta$$
$$\text{if } b' > 0 \text{ then } d' = G^{-1}(z_1) \text{ else } d' = \infty$$

3.3 COMPUTING THE ACCURACY OF STOCHASTIC SIMULATIONS

A simulation program is expected to output some quantities of interest. For example, for a simulation of the algorithm A it may be the average number of lost messages. The output of a stochastic simulation is random: two different simulation runs produce different outputs. Therefore, it is not sufficient to give one simulation result; in addition, we need to give the accuracy of our results.

3.3.1 INDEPENDENT REPLICATIONS

A simple and very efficient method to obtain confidence intervals is to use *replication*. Perform n independent replications of the simulation, each producing an output $x_1, ..., x_n$. Be careful to have truly random seeds for the random number generators, for example by accessing computer time (Section 3.5).

3.3.2 COMPUTING CONFIDENCE INTERVALS

You have to choose whether you want a confidence interval for the median or for the mean. The former is straightforward to compute, thus should be preferred in general.

Methods for computing confidence intervals for median and mean are summarized in Section 2.11.1.

EXAMPLE: APPLICATION TO EXAMPLE 3.2 ON PAGE 63. Figure 3.4 shows the time to transfer all files as a function of the number of customers. The simulation outputs do not appear to be normal, therefore we test whether n is large, by looking at the qqplot of the the bootstrap replicates. We find that it looks normal, so we can use the student statistic. By curiosity, we also compute the bootstrap percentile estimate and find that both confidence intervals are very close, the bootstrap percentile estimate being slightly smaller.

There are other methods of obtaining confidence intervals, but they involve specific assumptions on the model; see [LawKelton-2000].

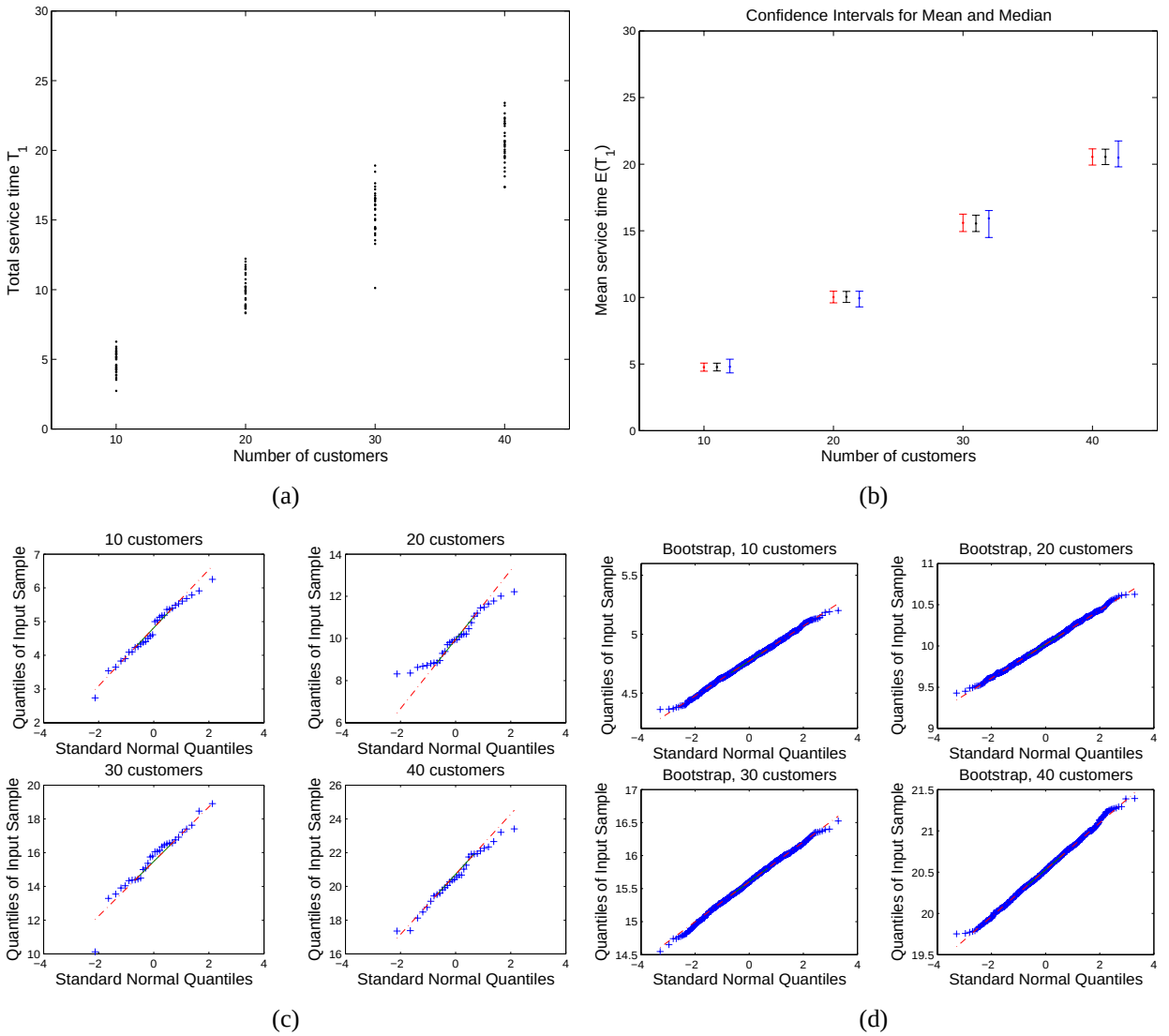


Figure 3.4: Time to serve n files in Joe's computer shop (Example 3.2 on page 63): (a) results of 30 independent replications, versus number of customers (b) 95%confidence intervals for the mean obtained with the normal approximation (left), with the bootstrap percentile estimate (middle); 95% confidence interval for the median (right). (c) qqplot of simulation outputs, showing deviation from normality (d) qq-plots of the bootstrap replicates, showing normality.

3.3.3 NON-TERMINATING SIMULATIONS

Non-terminating simulations should be asymptotically stationary (Section 3.1.2). When you simulate such a model, you should be careful to do *transient removal*. This involves determining:

- when to start measuring the output (this is the time at which we consider that the simulation has converged to its stationary regime
- when to stop the simulation

Unfortunately, there is no simple, bullet proof method to determine these two numbers. In theory, convergence to the stationary regime is governed by the value of the second eigenvalue modulus of the transition matrix of the markov chain that represents your simulation. In all but very special

cases, it is impossible to estimate this value. A practical method for removing transients is to look at the data produced by the simulation, and visually determine a time after which the simulation output does not seem to exhibit a clear trend behaviour. For example, in Figure **??** (a), the measurements could safely start at time $t = 1$. This is the same stationarity test as with time series (Chapter 9).

Determining when to stop a simulation is more tricky. The simulation should be large enough for transients to be removable. After that, you need to estimate whether running the simulation for a long time reduces the variance of the quantities that you are measuring. In practice, this is hard to predict a priori. A rule of thumb is to run the simulation long enough so that the output variable looks gaussian across several replications, but not longer than necessary.

3.4 MONTE CARLO SIMULATION

Monte Carlo simulation is a method for computing probabilities, expectations, or, in general, integrals when direct evaluations is impossible or too complex. It simply consists in estimating the expectation as the mean of a number of independent replications.

Formally, assume we are given a model for generating a data sequence $\vec{X}$. The sequence may be iid or not. Assume we want to compute $\beta = \mathbb{E}\left(\varphi(\vec{X})\right)$. Note that this covers the case where we want to compute a probability: if $\varphi(\vec{x}) = 1_{\{\vec{x}\in A\}}$ for some set A, then $\beta = \mathbb{P}(\vec{X} \in A)$.

Monte-Carlo simulation consists in generating R iid replicates $\vec{X}^r$, $r = 1, ..., R$. The Monte-Carlo estimate of β is

$$\hat{\beta} = \frac{1}{R}\sum_{r=1}^{R} \varphi(\vec{X}^r) \tag{3.2}$$

A confidence interval for β can then be computed using the methods in Chapter 2 for a confidence interval for the mean. By adjusting R, the number of replications, we can control the accuracy of the method, i.e. the width of the confidence interval.

EXAMPLE 3.7: p-VALUE OF A TEST. Let $X_1, ..., X_n$ be a sequence of iid random variables that take values in the discrete set $\{1, 2, ..., I\}$. Let $q_i = \mathbb{P}(X_k = i)$. Let $N_i = \sum_{k=1}^{n} 1_{\{X_k=i\}}$ (number of observation that are equal to i). Assume we want to compute

$$p = \mathbb{P}\left(\sum_{i=1}^{k} N_i \ln \frac{N_i}{nq_i} > a\right) \tag{3.3}$$

where $a > 0$ is given. This computation arises in the theory of goodness of fit tests, when we want to test whether X_i does indeed come from the model defined above. For large values of the sample size n we can approximate β by a χ^2 distribution (see Section 7.5), but for small values there is no analytic result.

We use Monte-Carlo simulation to compute p. We generate R iid replicates $X_1^r, ..., X_n^r$ of the sequence ($r = 1, ..., R$). This can be done by using the inversion method described in this chapter. For each replicate r, let

$$N_i^r = \sum_{k=1}^{n} 1_{\{X_k^r=i\}} \tag{3.4}$$

R	$\hat{p}$	margin
30	0.2667	0.1582
60	0.2500	0.1096
120	0.2333	0.0757
240	0.1917	0.0498
480	0.1979	0.0356
960	0.2010	0.0254
1920	0.1865	0.0174
3840	0.1893	0.0124
7680	0.1931	0.0088

Table 3.1: Computation of p in Example 3.7 on page 74 by Monte Carlo simulation. The parameters of the model are $I = 4$, $q_1 = 9/16, q_2 = q_3 = 3/16$, $q_4 = 1/16$, $n = 100$ and $a = 2.4$. The table shows the estimate $\hat{p}$ of p with its 95% confidence margin versus the number of Monte-Carlo replicates R. With 7680 replicates the relative accuracy (margin/$\hat{p}$) is below 5%.

The Monte Carlo estimate of p is

$$\hat{p} = \frac{1}{R}\sum_{r=1}^{R} 1_{\{\sum_{i=1}^{k} N_i \ln \frac{N_i}{nq_i} > a\}} \tag{3.5}$$

We compute a confidence interval by using a normal approximation, as explained in Example 2.14 on page 40. The sample variance is estimated by

$$\hat{\sigma} = \sqrt{\frac{\hat{p}(1-\hat{p})}{R}} \tag{3.6}$$

and a confidence interval at level 0.95 is $\hat{p} \pm 1.96\hat{\sigma}$. Assume we want a **relative** accuracy at least equal to some fixed value ϵ (for example $\epsilon = 0.05$). This is achieved if

$$\frac{1.96\hat{\sigma}}{\hat{p}} \leq \epsilon \tag{3.7}$$

which is equivalent to

$$R \geq \frac{3.92}{\epsilon^2}\left(\frac{1}{\hat{p}} - 1\right) \tag{3.8}$$

We can test for every value of R whether Equation (3.8) is verified and stop the simulation when this happens. Table 3.1 shows some results; we see that p is equal to 0.19 with an accuracy of 5%; the number of Monte Carlo replicates is proportional to the relative accuracy to the power -2.

3.5 RANDOM NUMBER GENERATORS

The simulation of any random process uses a basic function (such as `rand` in Matlab) that is assumed to return independent uniform random variables. Arbitrary distributions can be derived from there, as explained in Section 3.6.

In fact, `rand` is a *pseudo-random number generator*. It produces a sequence of numbers that appear to be random, but is in fact perfectly deterministic, and depends only on one initialization value of its internal stated, called the *seed*. There are several methods to implement pseudo random number generators; they are all based on chaotic sequences, i.e. iterative processes where a small difference in the seed produces very different outputs.

Simple random number generators are based on *linear congruences* of the type $x_n = ax_{n-1} \bmod m$. Here the internal state after n calls to `rand` is the last output x_n; the seed is x_0. Like for any iterative algroithm, the sequence is periodic, but for appropriate choices of a and m, the period may be very large.

EXAMPLE 3.8: LINEAR CONGRUENCE. A widespread generator (for example the default in ns2) has $a = 16'807$ and $m = 2^{31} - 1$. The sequence is $x_n = \frac{sa^n \bmod m}{m}$ where s is the seed. m is a prime number, and the smallest exponent h such that $a^h = 1 \bmod m$ is $m - 1$. It follows that for any value of the seed s, the period of x_n is exactly $m - 1$.Figure 3.5 shows that the sequence x_n indeed looks random.

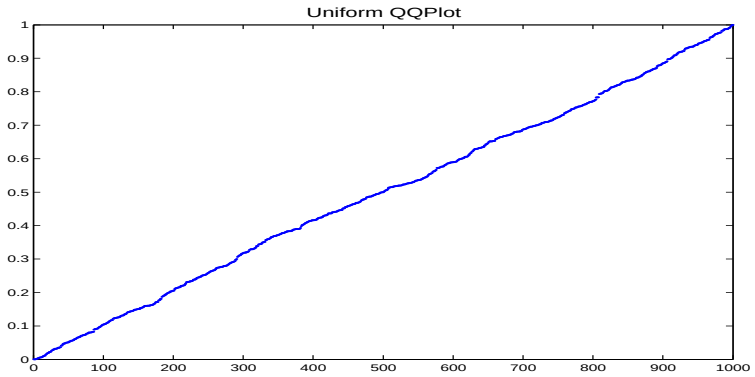
The period of a random number generator should be much smaller than the number of times it is called in a simulation. The generator in Example 3.8 on page 75 has a period of ca. 2×10^9, which may be too small for very large simulations. There are other generators with much longer periods, for example the "Mersenne Twister" [Matsumoto-98] with a period of $2^{19937} - 1$. They use other chaotic sequences and combinations of them.

Perfect pseudo-random number generators do not exist; only truly random generators can be perfect. Such generators exist: for example, quantum mechanics generator is based on the fact that the state of a photon is believed to be truly random. For a general discussion of generators in the framework of simulation, see [Hechenleitner-02]. Figure 3.6 illustrates a potential problem when the random number generator does not have a long enough period.

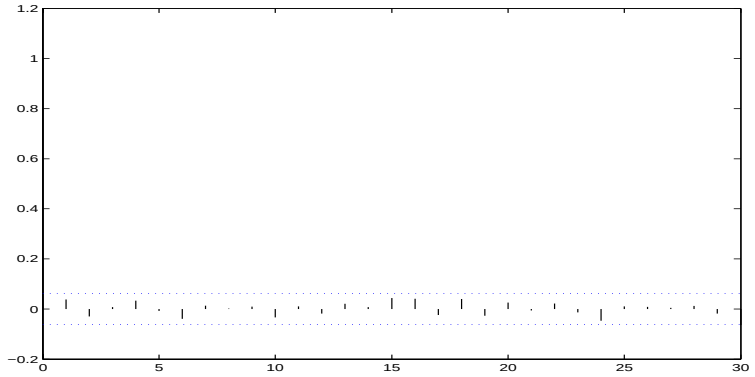
USING A RANDOM NUMBER GENERATOR IN PARALLEL STREAMS For some (obsolete) generators as in Example 3.8 on page 75, choosing small seed values in parallel streams may introduce a strong correlation (whereas we would like the streams to be independent).

EXAMPLE 3.9: PARALLEL STREAMS WITH INCORRECT SEEDS. Assume we need to generate two parallel streams of random numbers. This is very frequent in discrete event simulations; we may want to have one stream for the arrival process, and a second one for the service process. Assume we use the linear congruential generator of Example 3.8 on page 75, and generate two streams x_n and x'_n with seeds $s = 1$ and $s' = 2$. Figure 3.7 shows the results: we see that the two streams are strongly correlated. In contrast, taking $s' =$ the last value x_N of the first stream does not have this problem.

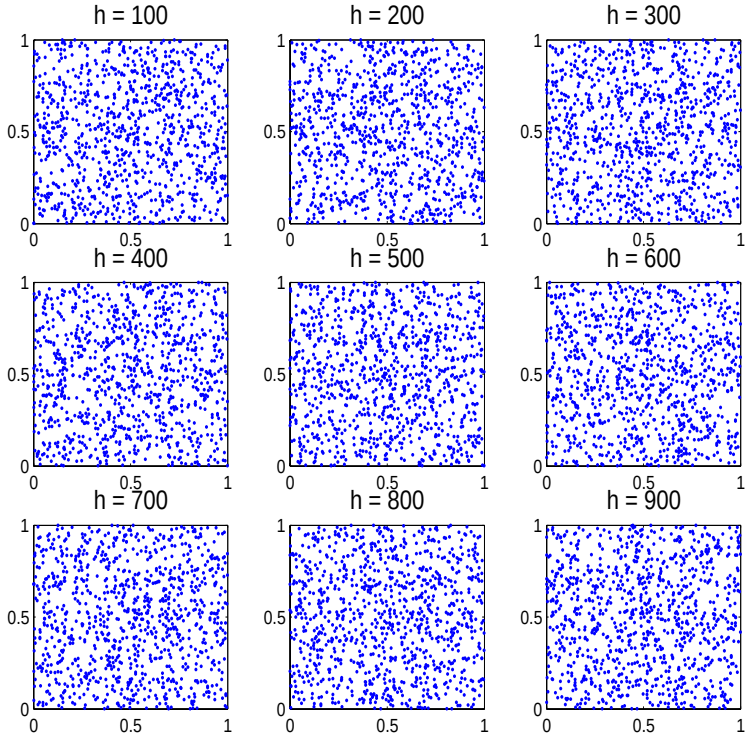
More modern generators as mentioned above do not have this problem either.



(a)

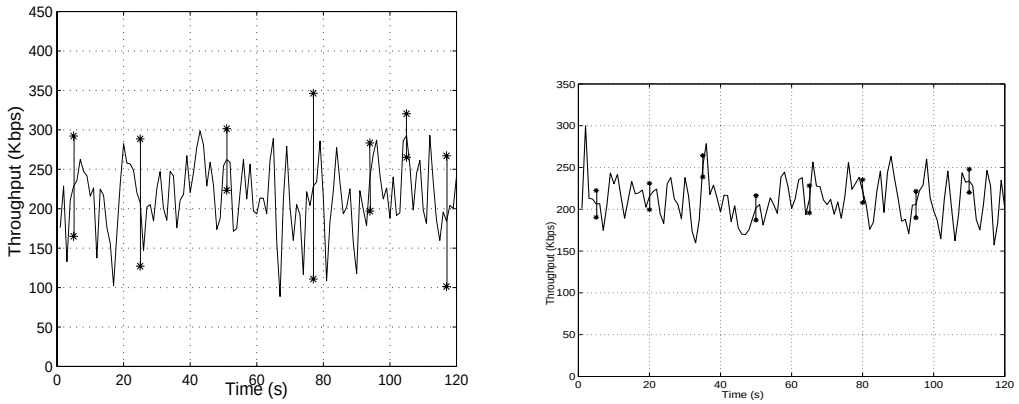


(b)



(c)

Figure 3.5: 1000 successive numbers for the generator in Example 3.8 on page 75. (a) QQplot against the uniform distribution in $(0, 1)$, showing a perfect match. (b) autocorrelation function, showing no significant correlation at any lag (c) lag plots at various lags, showing independence.



(a) Linear Congruence with $a = 16'807$ and $m = 2^{31} - 1$ (b) L'Ecuyer"'s generator[LecuyerSimConf-01]

Figure 3.6: Simulation outputs for the throughput of TCP connections over a wireless ad-hoc network. The wireless LAN protocol uses random numbers for its operation. This simulation consumes a very large number of calls to `rand`. The simulation results obtained with both generators are different: Lecuyer's generator produces consistently smaller confidence intervals.

SEEDING THE RANDOM NUMBER GENERATOR A safe way to make sure that replications are reasonably independent is to use the internal state of the generator at the end of the 1st replication as seed for the second replication and so one. This way, if the generator has a long enough sequence, the different replications have non overlapping sequences.

In practice, though, we often want independent replications to be run in parallel, so this mode of operation is not possible. A common practice is to take as seed a truly random number, for example derived from the computer clock.

3.6 HOW TO SAMPLE FROM A DISTRIBUTION

In this section we discuss methods to produce a sample X for a random variable that has a known distribution. We assume that we have a random number generator, that provides us with independent samples of the uniform distribution on $(0, 1)$. We focus on two methods of general applicability: inversion and rejection sampling.

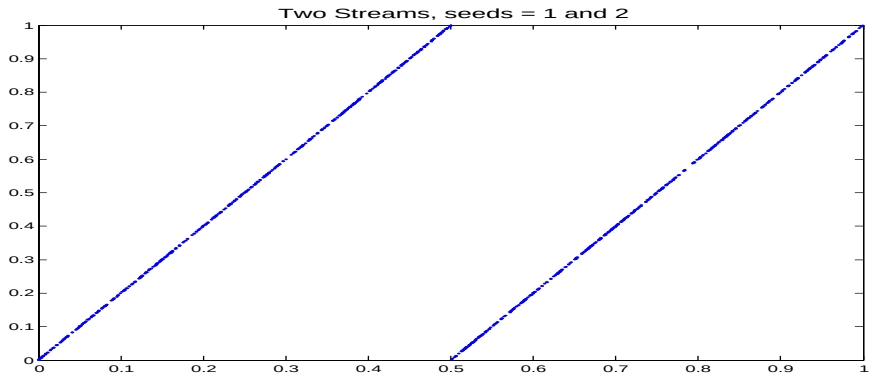
3.6.1 BY INVERSION OF CDF

This applies to real or integer valued random variable, when the cdf is easy to invert.

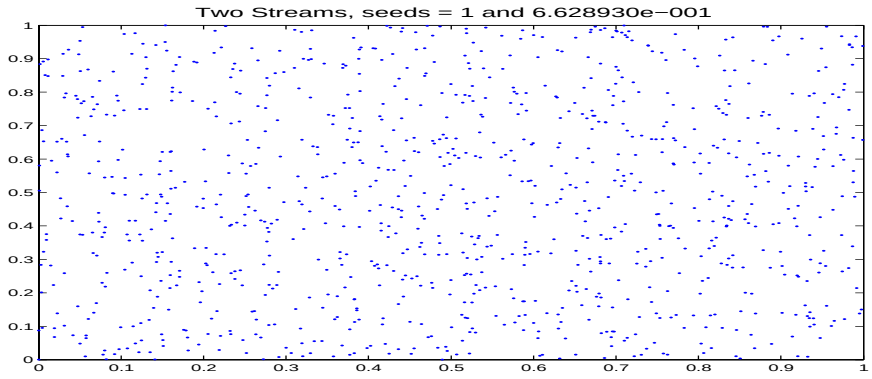
THEOREM 3.6.1. *Let F be the cdf of a random variable X with values in $\mathbb{R}$. Define the pseudo-inverse, F^{-1} of F by*

$$F^{-1}(p) = \sup\{x : F(x) \leq p\}$$

Let U be a sample of a random variable with uniform distribution on $(0, 1)$; $F^{-1}(U)$ is a sample of X.



(a)



(b)

Figure 3.7: x_n versus x'_n for two streams generated with the linear congruential in Example 3.8 on page 75. (a) seed values are 1 and 2 (b) seed values are (1, last value of first stream).

Proof. Take some arbitrary $c \in \mathbb{R}$. Let E be the event $E = \{F^{-1}(U) \leq c\}$. We want to show that $\mathbb{P}(E) = F(c)$.

We have the following equivalences (the following statements have the same truth value):

$$\begin{aligned} & \{F^{-1}(U) \leq c\} \\ \Leftrightarrow \quad & \{\sup\{x : F(x) \leq U\} \leq c\} \\ \Leftrightarrow \quad & \{\forall x \in \mathbb{R}\ F(x) \leq U \Rightarrow x \leq c\} \\ \Leftrightarrow \quad & \{\forall x \in \mathbb{R}\ x > c \Rightarrow F(x) > U\} \\ \Leftrightarrow \quad & \{\forall x > c\ F(x) > U\} \end{aligned}$$

The first equivalence is by definition of the pseudo-inverse. The second is by the definition of a $\sup$. The third is by the boolean equivalence of $(A \Rightarrow B)$ and (not$B \Rightarrow$not$A$). The fourth is simple re-writing. Thus we have shown that $E = \{U < \inf_{x>c} F(x)\}$. Now, by definition of an $\inf$:

$$\left\{U < \inf_{x>c} F(x)\right\} \subseteq \{\forall x > c\ F(x) > U\} \subseteq \left\{U \leq \inf_{x>c} F(x)\right\}$$

and, because $F()$ is right-continuous, we have $\inf_{x>c} F(x) = F(c)$. Thus

$$\{U < F(c)\} \subseteq \{\forall x > c\ F(x) > U\} \subseteq \{U \leq F(c)\}$$

As U is uniformly distributed on $(0, 1)$, $\mathbb{P}(U < F(c)) = \mathbb{P}(U \leq F(c)) = F(c)$ thus $F(c) \leq \mathbb{P}(E) \leq F(c)$.

□

Application to real random variable. In the case where X has a positive density over some interval I, then F is continuous and strictly increasing on I, and the pseudo-inverse is the inverse of F, as in the next example. It is obtained by solving the equation $F(x) = p$, where x is the unknown in I.

EXAMPLE 3.10: EXPONENTIAL RANDOM VARIABLE. The cdf of the exponential distribution with parameter λ is $F(x) = 1 - e^{-\lambda x}$. The pseudo-inverse is obtained by solving the equation

$$1 - e^{-\lambda x} = p$$

where x is the unknown. The solution is $x = -\frac{\ln(1-p)}{\lambda}$. Thus a sample X of the exponential distribution is obtained by letting $X = -\frac{\ln(1-U)}{\lambda}$, or, since U and $1 - U$ have the same distribution:

$$X = -\frac{\ln(U)}{\lambda} \tag{3.9}$$

where U is the output of the random number generator.

Application to integer random variable. Assume N is a random variable with values in $\mathbb{N}$. Let $p_k = \mathbb{P}(N = k)$, then for $n \in \mathbb{N}$:

$$F(n) = \sum_{k=0}^{n} p_k$$

and for $x \in \mathbb{R}$:

$$\begin{cases} \text{if } x < 0 \text{ then } F(x) = 0 \\ \text{else } F(x) = \mathbb{P}(N \leq x) = \mathbb{P}(N \leq \lfloor x \rfloor) = F(\lfloor x \rfloor) \end{cases}$$

We now compute $F^{-1}(p)$, for $0 < p < 1$. Let n be the smallest integer such that $p < F(n)$. The set $\{x : F(x) \leq p\}$ is equal to $(-\infty, n)$ (Figure 3.8); the supremum of this set is n, thus $F^{-1}(p) = n$. In other words, the pseudo inverse is given by

$$F^{-1}(p) = n \Leftrightarrow F(n-1) \leq p < F(n) \tag{3.10}$$

Thus, an integer valued random variable N can be sampled by: $N =$ the index n such that $F(n-1) \leq U < F(n)$, where U is the output of the random generator.

EXAMPLE 3.11: GEOMETRIC RANDOM VARIABLE. Here X takes integer values $0, 1, 2, \ldots$. The geometric distribution with parameter θ satisfies $\mathbb{P}(X = k) = \theta(1-\theta)^k$, thus for $n \in \mathbb{N}$:

$$F(n) = \sum_{k=0}^{n} \theta(1-\theta)^k = 1 - (1-\theta)^{n+1}$$

by application of Equation (3.10):

$$F^{-1}(p) = n \Leftrightarrow n \leq \frac{\ln(1-p)}{\ln(1-\theta)} < n+1$$

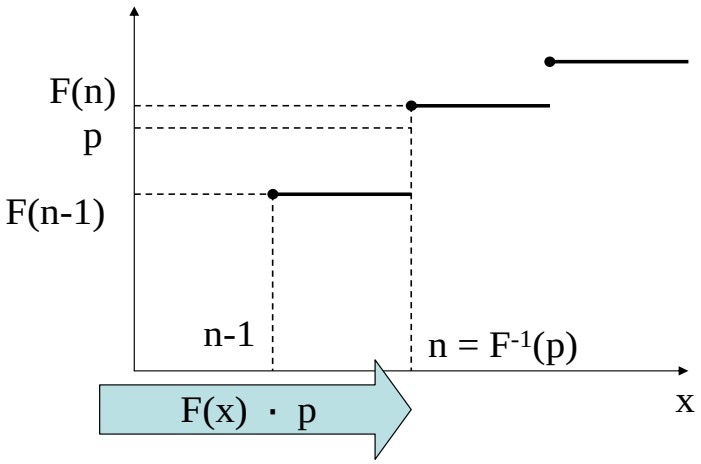


Figure 3.8: Pseudo-Inverse of cdf $F()$ of an integer-valued random variable

hence

$$F^{-1}(p) = \left\lfloor \frac{\ln(1-p)}{\ln(1-\theta)} \right\rfloor$$

and, since U and $1-U$ have the same distribution, a sample X of the geometric distribution is

$$X = \left\lfloor \frac{\ln(U)}{\ln(1-\theta)} \right\rfloor \tag{3.11}$$

QUESTION 3.6.1. *Consider the function defined by* `COIN(p)= if rand()<p 0 else 1`. *What does it compute ?* [7]

QUESTION 3.6.2. *Consider the sampling method: Draw* `COIN(p)` *until it returns 0. The value of the sample N is the number of iterations. Is this a good method for generating a sample from a geometric distribution ?* [8]

QUESTION 3.6.3. *Compare Equation (3.9) and Equation (3.11).* [9]

3.6.2 REJECTION SAMPLING

This is a method of large applicability. It can be used to generate samples of random variables when the inversion method does not work easily. It applies to random vectors of any dimension.

[7] It generates a sample of the Bernoulli random variable that takes the value 0 with p and the value 1 with probability $1-p$.

[8] The distribution of N is geometric with $\theta = 1-p$. so this method does produce a sample from a geometric distribution. However it draws in average $\frac{1}{\theta}$ random numbers from the generator, and the random number generator is usually considered an expensive computation compared to a floating point operation. If θ is small, the procedure in Example 3.11 on page 80 is much more efficient.

[9] They are similar, in fact we have $N = \lfloor X \rfloor$ if we let $\lambda = \ln(1-\theta)$. This follows from the fact that if $X \sim \exp(\lambda)$, then $\lfloor X \rfloor$ is geometric with parameter $\theta = 1 - e^{-\lambda}$

The method is based on the following result, which is of independent interest. It allows to sample from a distribution given in conditional form.

THEOREM 3.6.2 (Rejection Sampling for a Conditional Distribution). *Let X be a random variable in some space S such that the distribution of X is the conditional distribution of $\tilde{X}$ given that $\tilde{Y} \in \mathcal{A}$, where $(\tilde{X}, \tilde{Y})$ is a random variable in $S \times S'$ and $\mathcal{A}$ is a measurable subset of S.*
A sample of X is obtained by the following algorithm:

do
draw a sample of $(\tilde{X}, \tilde{Y})$
until $\tilde{Y} \in \mathcal{A}$
return$(\tilde{X})$

The expected number of iterations of the algorithm is $\frac{1}{\mathbb{P}(\tilde{Y} \in \mathcal{A})}$.

Proof. Let N be the (random) number of iterations of the algorithm, and let $(\tilde{X}_k, \tilde{Y}_k)$ be the sample drawn at the kth iteration. (These samples are independent, but in general, $\tilde{X}_k$ and $\tilde{Y}_k$ are *not* independent). Let $\theta = \mathbb{P}(\tilde{Y} \in \mathcal{A})$. We assume $\theta > 0$ otherwise the conditional distribution of $\tilde{X}$ is not defined. The output of the algorithm is $X = \tilde{X}_N$.

For some arbitrary measurable $\mathcal{B}$ in S, we compute $\mathbb{P}(\tilde{X}_N \in \mathcal{B})$:

$$\begin{aligned}
\mathbb{P}\left(\tilde{X}_N \in \mathcal{B}\right) &= \sum_{k\geq 1} \mathbb{P}\left(\tilde{X}_k \in \mathcal{B} \text{ and } N = k\right) \\
&= \sum_{k\geq 1} \mathbb{P}\left(\tilde{X}_k \in \mathcal{B} \text{ and } \tilde{Y}_1 \notin \mathcal{A}, ..., \tilde{Y}_{k-1} \notin \mathcal{A}, \tilde{Y}_k \in \mathcal{A}\right) \\
&= \sum_{k\geq 1} \mathbb{P}\left(\tilde{X}_k \in \mathcal{B} \text{ and } \tilde{Y}_k \in \mathcal{A}\right) \mathbb{P}\left(\tilde{Y}_1 \notin \mathcal{A}\right) ... \mathbb{P}\left(\tilde{Y}_{k-1} \notin \mathcal{A}\right) \\
&= \sum_{k\geq 1} \mathbb{P}\left(\tilde{X}_k \in \mathcal{B} | \tilde{Y}_k \in \mathcal{A}\right) \theta(1-\theta)^{k-1} \\
&= \sum_{k\geq 1} \mathbb{P}\left(\tilde{X}_1 \in \mathcal{B} | \tilde{Y}_1 \in \mathcal{A}\right) \theta(1-\theta)^{k-1} \\
&= \mathbb{P}\left(\tilde{X}_1 \in \mathcal{B} | \tilde{Y}_1 \in \mathcal{A}\right) \sum_{k\geq 1} \theta(1-\theta)^{k-1} \\
&= \mathbb{P}\left(\tilde{X}_1 \in \mathcal{B} | \tilde{Y}_1 \in \mathcal{A}\right)
\end{aligned}$$

The second equality is by definition of N. The third is by the independence of $(\tilde{X}_k, \tilde{Y}_k)$ and $(\tilde{X}_{k'}, \tilde{Y}_{k'})$ for $k \neq k'$. The last equality is because $\theta > 0$. This shows that the distribution of X is as required.

$N - 1$ is geometric with parameter θ thus the expectation of N is $1/\theta$.

□

EXAMPLE 3.12: DENSITY RESTRICTED TO ARBITRARY SUBSET. Consider a random variable in some space ($\mathbb{R}, \mathbb{R}^n, \mathbb{Z}$...) that has a density $f_Y(y)$. Let $\mathcal{A}$ be a set such that $\mathbb{P}(Y \in \mathcal{A}) > 0$. We are interested in the distribution of a random variable X whose density is that of Y, restricted to $\mathcal{A}$:

$$f_X(y) = K f_Y(y) 1_{\{y \in \mathcal{A}\}} \tag{3.12}$$

where $K^{-1} = \mathbb{P}(Y \in \mathcal{A}) > 0$ is a normalizing constant. This distribution is the conditional distribution of Y, given that $Y \in \mathcal{A}$.

QUESTION 3.6.4. *Show this.* [10]

Thus a sampling method for the distribution with density in Equation (3.12) is to draw samples of the distribution with density f_Y until a sample is found that belongs to $\mathcal{A}$. The expected number of iterations is $1/\mathbb{P}(Y \in \mathcal{A})$.

For example, consider the sampling of a random point X uniformly distributed on some bounded area $\mathcal{A} \subset \mathbb{R}^2$. We can consider this density as the restriction of the uniform density on some rectangle $\mathcal{R} = [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$ that contains the area $\mathcal{A}$. Thus a sampling method is to draw points uniformly in $\mathcal{R}$, until we find one in $\mathcal{A}$. The expected numbers of iterations is the ratio of the area of $\mathcal{R}$ to that of $\mathcal{A}$; thus one should be careful to pick a rectangle that is close to $\mathcal{A}$. Figure 3.9 shows a sample of the uniform distribution over a non-convex area.

QUESTION 3.6.5. *How can one generate a sample of the uniform distribution over $\mathcal{R}$?* [11]

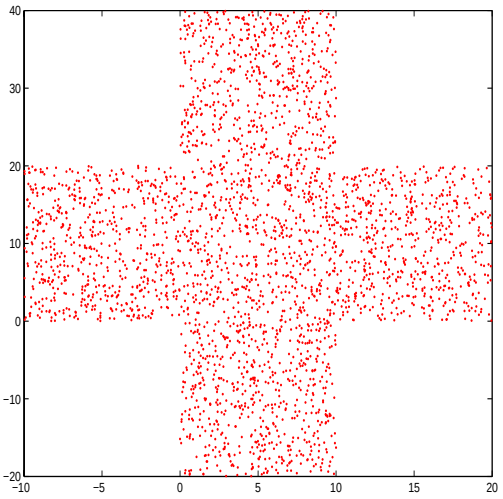


Figure 3.9: 1000 independent samples of the uniform distribution over $\mathcal{A} =$ the interior of the cross. Samples are obtained by generating uniform samples in the bounding rectangle and rejecting those samples that do not fall in $\mathcal{A}$.

EXAMPLE 3.13: HALF-NORMAL DENSITY. [6] The half-normal distribution is the distribution of the absolute value of normal random variable. It has density $\frac{2}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} 1_{\{y>0\}}$. It can easily be seen that it is also the conditional distribution of a standard normal random variable given that it is positive. We could derive a sampling method from

[10] For any (measurable) subset $\mathcal{B}$ of the space, $\mathbb{P}(X \in \mathcal{B}) = K \int_{\mathcal{B}} f_Y(y) 1_{\{y \in \mathcal{A}\}} dy = K\mathbb{P}(Y \in \mathcal{A} \text{ and } Y \in \mathcal{B}) = \mathbb{P}(Y \in \mathcal{B} | Y \in \mathcal{A})$.

[11] The coordinates are independent and uniform: generate two independent samples $U, V \sim$Unif$(0,1)$; the sample is $((1-U)x_{\min} + Ux_{\max}, (1-V)y_{\min} + Vy_{\max}$.

this, using Example 3.12 on page 81, if we knew how to sample from a normal distribution. In fact, the method presented in this example is used to generate a sample from the normal distribution, so we do not follow this track. Instead, we use the following observation.

Let Y, Z be two independent, exponential random variables, with parameter $\lambda = 1$. The conditional distribution of Y given that $Z > \frac{1}{2}(1-Y)^2$ is half-normal.

To see why, compute, for an arbitrary function ϕ:

$$\begin{aligned}
&\mathbb{E}(\phi(Y)1_{\{Z>\frac{1}{2}(1-Y)^2\}}) \\
&= \int_{y>0} \phi(y) \left(\int_{\frac{1}{2}(1-y)^2}^{+\infty} e^{-z} dz \right) e^{-y} dy \\
&= \int_{y>0} \phi(y) e^{-\frac{1}{2}(1-y)^2} e^{-y} dy \\
&= K \int_{y>0} \phi(y) e^{-\frac{1}{2}y^2} dy
\end{aligned}$$

where K is some constant. Thus

$$\mathbb{E}(\phi(Y)|Z > \frac{1}{2}(1-Y)^2) = \frac{K}{\mathbb{P}(Z > \frac{1}{2}(1-Y)^2)} \int_{y>0} \phi(y) e^{-\frac{1}{2}y^2} dy = K' \int_{y>0} \phi(y) e^{-\frac{1}{2}y^2} dy$$

where K' is some other constant. This shows that the conditional distribution of Y is half-normal.

Since sampling from an exponential distribution can easily be done by inversion of the cdf, we can now apply the previous theorem with $\tilde{X} = Y$ and obtain a sampling method for the half-normal distribution: draw independent samples Y, Z of the exponential distribution with $\lambda = 1$ until the condition $Z > \frac{1}{2}(1-Y)^2$ is true. The sample is the value of Y.

Now we come to a very general result, for all distributions that have a density.

THEOREM 3.6.3 (Rejection Sampling for Distribution with Density). *Consider two random variables X, Y with values in the same space, that both have densities. Assume that:*

- *we know a method to draw a sample of X*
- *the density of Y is known up to a normalization constant K: $f_Y(y) = Kf_Y^n(y)$, where f_Y^n is a known function*
- *there exist some $c > 0$ such that*

$$\frac{f_Y^n(x)}{f_X(x)} \leq c$$

A sample of Y is obtained by the following algorithm:

do
draw independent samples of X and U, where $U \sim$Unif$(0, c)$
until $U \leq \frac{f_Y^n(X)}{f_X(X)}$
return(X)

The expected number of iterations of the algorithm is $\frac{c}{K}$.

Proof. Apply Theorem 3.6.2 with $\tilde{X} = X$ and $\tilde{Y} = (X, U)$. All we need to show is that the conditional density of X given that $U \leq \frac{f_Y^n(X)}{f_X(X)}$ is f_Y.

To this end, pick some arbitrary function ϕ. We have

$$\begin{aligned}
&\mathbb{E}\left(\phi(X)|U \leq \frac{f_Y^n(X)}{f_X(X)}\right) \\
&= K_1\mathbb{E}\left(\phi(X)1_{\{U\leq\frac{f_Y^n(X)}{f_X(X)}\}}\right) \\
&= K_1\int \mathbb{E}\left(\phi(x)1_{\{U\leq\frac{f_Y^n(x)}{f_X(x)}\}}|X = x\right) f_X(x)dx \\
&= K_1\int \phi(x)\frac{f_Y^n(x)}{f_X(x)}f_X(x)dx \\
&= \frac{K_1}{K}\int \phi(x)f_Y(x)dx = \frac{K_1}{K}\mathbb{E}(\phi(Y))
\end{aligned}$$

where K_1 is some constant. This is true for all ϕ thus, necessarily, $K_1/K = 1$ (take $\phi = 1$).

□

A frequent use of Theorem 3.6.3 is as follows.

EXAMPLE 3.14: ARBITRARY DISTRIBUTION WITH DENSITY. Assume that: Y takes values in the bounded interval $[a, b]$, has a density $f_Y = Kf_Y^n(y)$ that can easily be computed but for the multiplicative constant K, and that we know an upper bound M on f_Y^n. We take X uniformly distributed over $[a, b]$ and obtain the sampling method:

do
draw $X \sim$Unif(a, b) and $U \sim$Unif$(0, M)$
until $U \leq f_Y^n(X)$
return(X)

Note that we do *not* need to know the multiplicative constant K. For example, consider the distribution with density

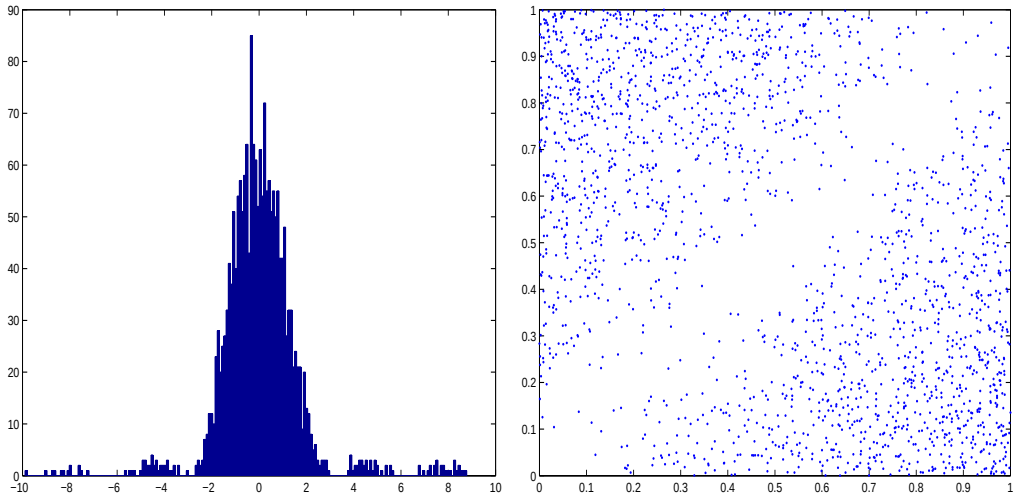
$$f_Y(y) = K\frac{\sin^2(y)}{y^2}1_{\{-a\leq y\leq a\}} \tag{3.13}$$

K is hard to compute, but a bound M on f_Y^n is easy to find ($M = 1$).

EXAMPLE 3.15: A STOCHASTIC GEOMETRY EXAMPLE. We want to sample the random vector (X_1, X_2) that takes values in the rectangle $[0, 1] \times [0, 1]$ and whose distribution has a density proportional to $|X_1 - X_2|$. We take $f_X =$ the uniform density over $[0, 1] \times [0, 1]$ and $f_Y^n(x_1, x_2) = |x_1 - x_2|$. An upper bound on the ratio $\frac{f_Y^n(x_1,x_2)}{f_X(x_1,x_2)}$ is 1. The sampling algorithm is thus:

do
draw X_1, X_2 and $U \sim$Unif$(0, 1)$
until $U \leq |X_1 - X_2|$
return(X_1, X_2)

Figure 3.10 shows an example. Note that there is no need to know the normalizing constant to apply the sampling algorithm.



(a) (b)

Figure 3.10: (a) Empirical histogram (bin size = 10) of 2000 samples of the distribution with density $f_X(x)$ proportional to $\frac{\sin^2(x)}{x^2}1_{\{-a\leq y\leq a\}}$ with $a = 10$. (b) 2000 independent samples of the distribution on the rectangle with density $f_{X_1,X_2}(x_1, x_2)$ proportional to $|x_1 - x_2|$.

3.6.3 AD-HOC METHODS

The methods of inversion and rejection sampling may be improved in some special cases. We mention in detail the case of the normal distribution, which is important to optimize because of its frequent use.

Sampling a Normal Random Variable. The method of inversion cannot be directly used, as the cdf is hard to compute. An alternative is based on the method of **change of variables**.

PROPOSITION 3.6.1. *Let (X, Y) be independent, standard normal random variables. Let*

$$\begin{cases} R = \sqrt{X^2 + Y^2} \\ \Theta = \arg(X + jY) \end{cases}$$

R and Θ are independent, R has a Rayleigh distribution (i.e is positive with density $re^{\frac{-r^2}{2}}$) and Θ is uniformly distributed on $[0, 2\pi]$.

Proof. Apply the formula for a change of variables in Section 12.1.2. We have

$$\begin{cases} X = R\cos(\Theta) \\ Y = R\sin(\Theta) \end{cases}$$

The jacobian of this transformation is R, thus

$$f_{R,\Theta}(r, \theta) = \frac{R}{2\pi} e^{-\frac{R^2}{2}}$$

□

The cdf of the Rayleigh distribution can easily be inverted: $F(r) = \mathbb{P}(R \leq r) = 1 - e^{-r^2/2}$ and $F^{-1}(p) = \sqrt{-2\ln(1-p)}$. A sampling method for a couple of two independent standard normal variables is thus (*Box-Müller method*):

draw $U \sim$Unif$(0, 1)$
$R = \sqrt{-2\ln(U)}$
draw $\Theta \sim$Unif$(0, 2\pi)$
$X = R\cos(\Theta)$, $Y = R\sin(\Theta)$
return(X, Y)

QUESTION 3.6.6. *In Example 3.13 on page 82 we obtained a method to sample from the half-normal density. How can this be used to sample a normal random variable ?* [12]

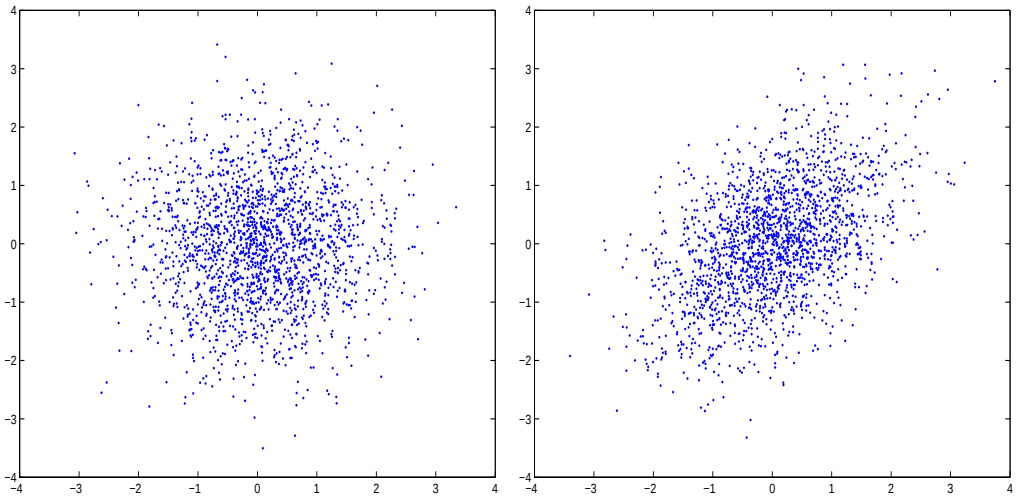
Correlated Normal Random Vectors. We want to sample $(X_1, ..., X_n)$ as a normal random vector with zero mean and covariance matrix Ω (see Section **??**). If the covariance matrix is diagonal (i.e. $\Omega i, j = 0$ for $i \neq j$) then the X_is are independent and we can sample them one by one (or better, two by two). We are interested here in the case where there is some correlation.

[12] Let Y be a sample from the standard half-normal distribution. Let Z be an independent coin tossing variable with $Z = \pm 1$ with equal probabilities. Let $X = ZY$. Z has the same distribution as $-Z$ therefore X also has the same distribution as $-X$. $X > 0$ means that $Z = 1$ therefore the conditional distribution of X given that $X > 0$ is that of Y, i.e. is the conditional distribution of a standard normal variable given that it is > 0. By symmetry, the same holds for the conditional distribution given that $X < 0$. Thus X has a standard normal distribution.

The method we show here is again based on a change of variable. There exists always a change of basis in $\mathbb{R}^n$ such that, in the new basis, the random vector has a diagonal covariance matrix. In fact, there are many such bases (one of them is orthonormal and can be obtained by diagonalisation of Ω, but is much more expensive than the method we discuss next). An inexpensive and stable algorithm to obtain one such basis is called Choleski's factorization method. It amounts to finding a matrix L such that $\Omega = LL^T$. Let Y be a standard normal vector (i.e. an iid sequence of n standard normal random variables). Let $X = LY$. The covariance matrix of X is

$$\mathbb{E}(XX^T) = \mathbb{E}(LY(LY)^T)) = \mathbb{E}(L(YY^T)L^T) = L\mathbb{E}(YY^T)L^T = LL^T = \Omega$$

Thus a sample of X can be obtained by sampling Y first and computing LY. Figure 3.6.3 shows an example.



(a) X_1, X_2 independent (b) X_1, X_2 dependent

Figure 3.11: 1000 independent samples of the normal vector X_1, X_2 with 0 mean and covariance $\Omega_{1,1} = \sigma_1^2 = 1$, $\Omega_{2,2} = \sigma_2^2 = 1$ and $\Omega_{1,2} = \Omega_{2,1} = 0$ (left), $\Omega_{1,2} = \Omega_{2,1} = 1/2$ (right). The right sample is obtained by the transformation $X = LY$ with $Y iid \sim N_{0,1}$ and $L = (1, 0; 1/2, \sqrt{3}/2)$.

Other Methods There are many ways to optimize the generation of samples. Good references are [6] and [**?**]

3.7 REVIEW

QUESTION 3.7.1. *What are real time and simulated time ?* [13]

QUESTION 3.7.2. *Why do we need to run independent replications of a simulation ? How are they obtained ?* [14]

[13] The time taken by the computer to run the simulation program; the time as experienced by the system being simulated.

[14] To obtain confidence intervals. By running multiple instances of the simulation program; if done sequentially, the seed of the random generator can be carried over from one run to the next. If replications are done in parallel on several machines, the seeds should be chosen independently by looking up a table of random numbers.

QUESTION 3.7.3. *Why do we need to verify normality when computing confidence intervals ?* [15]

QUESTION 3.7.4. *Why do we need the bootstrap method to test whether the sample size is large enough, when computing confidence intervals ?* [16]

3.8 EXERCISES

Floyd: simulating the Internet

[15] Because the computation of the confidence interval assumes that either (1) the data is approximately normal or (2) the mean of the data converges in distribution to a normal random variable.

[16] Because we have only one value of the statistic t, so we cannot perform a normality test on it.

CHAPTER 4

MODEL FITTING

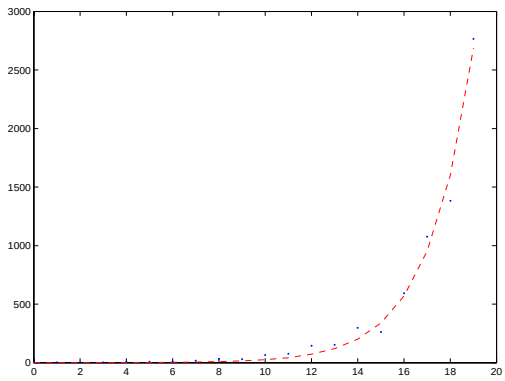
In this chapter we study how to derive a model from data, for example, fitting a curve to a series of measurements. Using a motivating example, we illustrate that fitting a model can be misleading, and that the issue can be circumvented if we interpret the model fitting problem as a statistical estimation problem. The widely used least square fitting method corresponds to the homoscedastic assumption, i.e., when the noise can be assumed to be normal iid. Verification of assumptions can be done by examination of residuals. Linear regression is a special case, also called "ANOVA", which occurs when the dependency of the model parameters is linear; there are closed form solutions for computing the model (and confidence intervals). We see that linear regression is much more general than the term "linear" suggests. In some very specific cases, the ANOVA model can be used to simplify factorial analysis, i.e. a quantitative assessment of the importance of factors. Last, we will point out to modelling patterns: the hidden factor and Simpson's paradox. The proofs of the theorems in this chapter are all based on a few geometrical properties of gaussian independent (but not identical) vectors, which are explained in appendix in Chapter 12.

Contents

4.1 WHAT IS MODEL FITTING ?

We start with a simple example.

EXAMPLE 4.1: VIRUS SPREAD DATA. The number of hosts infected by a virus is plotted versus time in hours.



The plot suggests an exponential growth, therefore we are inclined to fit these data to a model of the form

$$Y(t) = ae^{\alpha t} \tag{4.1}$$

where $Y(t)$ is the number of infected hosts at time t. We are particulary interested in the parameter α, which can be interpreted as the growth rate; the *doubling time* (time for the number of infected hosts to double) is $\frac{\ln 2}{\alpha}$. On the plot, the dashed line is the curve fitted by the method of least squares explained later. We find $\alpha = 0.5173$ per hour and the doubling time is 1.34 hour. We can use the model to predict that, 6 hours after the end of the measurement period, the number of infected hosts would be ca. 100'000.

In general, *model fitting* can be defined as the problem of finding an *explanatory model* for the data, i.e. a mathematical relation of the form

$$y_i = f_i(\vec{\beta}) \tag{4.2}$$

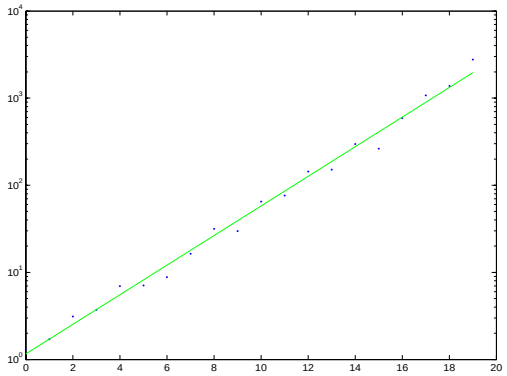
that "explains the data well", in some sense. Here y_i is the collection of measured data, i is the index of a measurement, f_i is an array of functions, and $\vec{\beta}$ is the parameter that we would like to obtain. In the previous example, the parameter is $\vec{\beta} = (a, \alpha)$ and $f_i(\vec{\beta}) = f_i(a, \alpha) = ae^{\alpha t_i}$ where t_i is the time of the ith measurement, assumed here to be known.

What does it mean to "explain the data well" ? It is generally not possible to require that Equation (4.2) holds *exactly* for all data points. Therefore, a common answer is to require that the model minimizes some metric of the discrepancy between the explanatory model and the data. A very common metric is the mean square distance $\sum_i \left(y_i - f_i(\vec{\beta})\right)^2$. The value of the growth rate α in the previous example was obtained in this way, namely, we computed a and α that minimize $\sum_i (y_i - ae^{\alpha t_i})^2$.

QUESTION 4.1.1. *How would you compute a and α ?* [1]

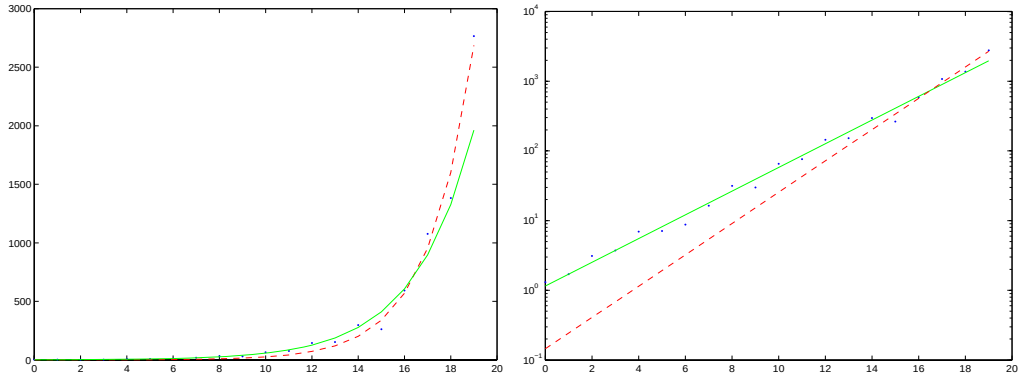
But this raises another question. What metric should one use ? What is so magical about least squares ? Why not use other measures of discrepancy, for example $\sum_i |y_i - f_i(\vec{\beta})|$ or $\sum_i \left(\ln(y_i) - \ln(f_i(\vec{\beta}))\right)^2$? The following example shows the importance of the issue.

EXAMPLE 4.2: VIRUS SPREAD DATA, CONTINUED. AMBIGUITY IN THE OPTIMIZATION CRITERION. We also plotted the number of infected hosts in log scale:



and computed the least square fit of Equation (4.2) in log scale (plain line). Namely, we computed a and α that minimize $\sum_i (\ln(y_i) - \ln(a) - \alpha t_i)^2$. We found for α the value 0.39 per hour, which gives a doubling time of 1.77 hour and a prediction at time $+6$ hours equal to ca. $39'000$ infected hosts (instead of previously $100'000$).

The two different models are compared below (in linear and log scales).



Both figures show that what visually appears to be a good fit in one scale is not so in the other. Which one should we use ?

An answer to the issue comes from statistics. The idea is to add to the explanatory model a description of the "noise" (informally defined as the deviation between the explanatory model and

[1] This is a non constrained optimization problem in two variables; we used a generic solver (`fminsearch` in matlab)

the data), and obtain a ***statistical model***. We can also think of the statistical model as a description of a simulator that was used to produce the data we have. Its parameters are well defined, but not known to us.

The statistical model usually has a few more parameters than the explanatory model. The parameters of the statistical model are estimated using the classical approach of maximum likelihood. If we believe in the statistical model, this answers the previous issue by saying that the criterion to be optimized is the likelihood. The belief in the model can be checked by examining residuals.

EXAMPLE: VIRUS SPREAD DATA, CONTINUED. A STATISTICAL MODEL.One **statistical model** for the virus spread data is

$$Y_i = ae^{\alpha t_i} + \epsilon_i \text{ with } \epsilon_i \text{ iid } \sim N_{0,\sigma^2} \tag{4.3}$$

in other words, we assume that the measured data y_i is equal to the ideal value given by the explanatory model, plus a noise term ϵ_i. Further, we assume that all noises are independent, gaussian, and with same variance. The parameter is $\theta = (a, \alpha, \sigma)$.

In Equation (4.3), we write Y_i instead of y_i to express that Y_i is a random variable. We think of our data y_i as being *one* sample produced by a simulator that implements Equation (4.3).

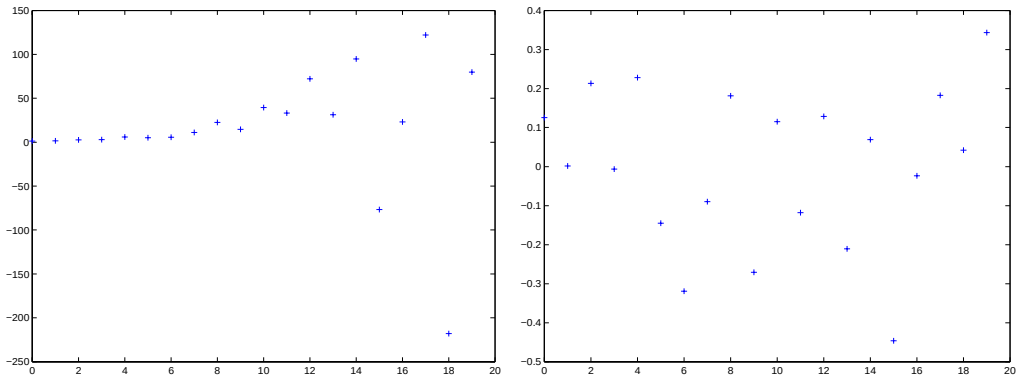
We will see in Section 4.2 that the maximum likelihood estimator for this model is the one that minimizes the mean square distance. Thus, with this model, we obtain for α the value in Example 4.1 on page 90.

A second statistical model could be:

$$\ln(Y_i) = \ln\left(ae^{\alpha t_i}\right) + \epsilon_i \text{ with } \epsilon_i \text{ iid } \sim N_{0,\sigma^2} \tag{4.4}$$

Now, we would be assuming that the noise terms in log-scale have the same variance, in other words, the noise is proportional to the measured value. Here too, the maximum likelihood estimator is obtained by minimizing the least square distance, thus we obtain for α the value in Example 4.2 on page 91.

We can validate either model by plotting the residuals:



We see clearly that the residual for the former model do not appear to be normally distributed, and the converse is true for the former model, which is the one we should adopt. Therefore, an acceptable fitting is obtained by minimizing least squares in log-scale.

QUESTION 4.1.2. *How would you compute the residuals ?* [2]

We summarize what we have learnt so far as follows.

FITTING A MODEL TO DATA

1. Define a statistical model that contains **both** the deterministic part (the one we are interested in) and a model of the noise.
2. Estimate the parameters of the statistical model using maximum likelihood. If the number of data points is small, use a brute force approach (e.g use `fminsearch`). If the number of data points is large, you may need to look in the literature for efficient, possibly heuristic, optimization methods.
3. Validate the model fit by screening the residuals, either visually, or using tests (Chapter 7).

4.2 LEAST SQUARES CORRESPOND TO GAUSSIAN, SAME VARIANCE

A very frequent case is when the statistical model has the form

$$Y_i = f_i(\vec{\beta}) + \epsilon_i \text{ for } i = 1, \ldots, I \text{ with } \epsilon_i \text{ iid } \sim N_{0,\sigma^2} \tag{4.5}$$

as in the examples before (Models in Equations (4.3) and (4.4)). Namely, the discrepancy between the explanatory model and the data is assumed to be gaussian with **same variance**. In some literature, the "same variance" assumption is called *homoscedasticity*.

THEOREM 4.2.1 (Least Squares). *For the model in Equation (4.5), the maximum likelihood estimator of the parameter $(\vec{\beta}, \sigma)$ is given by:*

1. $\hat{\beta} = \arg\min_{\vec{\beta}} \sum_i \left(y_i - f_i(\vec{\beta})\right)^2$
2. $\hat{\sigma}^2 = \frac{1}{I} \sum_i \left(y_i - f_i(\hat{\beta})\right)^2$

Proof. The log likelihood of the data is

$$l_{\vec{y}} = -\frac{I}{2} \ln(2\pi) - I \ln(\sigma) - \frac{1}{2\sigma^2} \sum_{i=1}^{I} \left(y_i - f_i(\vec{\beta})\right)^2 \tag{4.6}$$

For any fixed σ, it is maximum when $\sum_{i=1}^{I} \left(y_i - f_i(\vec{\beta})\right)^2$ is minimum, which shows item 1. Take the derivative with respect to σ and find that for any fixed $\vec{\beta}$, it is maximum for $\sigma = \frac{1}{I} \sum_i \left(y_i - f_i(\vec{\beta})\right)^2$, which shows item 2.

[2] The residuals are estimates of the noise terms ϵ_i. Let $\hat{a}$ and $\hat{\alpha}$ be the values estimated by maximum likelihood, for either model. The residuals are $r_i = y_i - \hat{a}e^{\hat{\alpha}t_i}$ for the former model, $r_i = \ln y_i - \ln\left(\hat{a}e^{\hat{\alpha}t_i}\right)$ for the latter.

□

The theorem explains what we do when we fit the explanatory model $y_i = f_i(\vec{\beta})$ to our data using least squares: we implicitly assume that the error terms in our data are independent, gaussian, and of same amplitude. We have seen in the examples above that care must be taken to validate this assumption.

The set of points in $\mathbb{R}^I$ that have coordinates of the form $f_i(\vec{\beta})$ constitue a "manifold" (for $p = 2$, it is a surface). Item 1 says that $\vec{\beta}$ is the parameter of the point $\hat{y}$ on this manifold that is the nearest to the data point $\vec{y}$, in euclidian distance. The point $\hat{y}$ is called the *predicted response*; it is an estimate of the value that $\vec{y}$ would take if there would be no noise. It is equal to the orthogonal projection of the data $\vec{y}$ onto the manifold.

QUESTION 4.2.1. *How would you compute confidence intervals for $\vec{\beta}$?* [3]

4.3 LINEAR REGRESSION

A special case of the previous section is when the explanatory model depends linearly on its parameter $\vec{\beta}$. This is called the *linear regression* model. The main fact here is that everything can be computed easily, in matrix forms.

Assume thus that the *statistical model* of our experiment has the form:

DEFINITION 4.3.1 (Linear Regression Model).

$$Y_i = (X\vec{\beta})_i + \epsilon_i \text{ for } i = 1, \ldots, I \text{ with } \epsilon_i \text{ iid } \sim N_{0,\sigma^2} \tag{4.7}$$

where the unknown parameter $\vec{\beta}$ is in $\mathbb{R}^p$ and X is a $I \times p$ matrix. The matrix X supposed to be known exactly in advance. We also assume that

H *X has rank p*

Assumption **H** means that different values of $\vec{\beta}$ give different values of the explanatory model $X\vec{\beta}$, i.e. the explanatory model is identifiable.

The elements of the known matrix X are sometimes called *explanatory variables*, and then the y_is are called the *response variables*.

EXAMPLE 4.3: JOE'S SHOP AGAIN, FIGURE 1.1(B). We assume that there is a threshold ξ beyond which the throughput collapses (we take $\xi = 70$). The statistical model is

$$Y_i = (a + bx_i)1_{x_i \leq \xi} + (c + d\xi)1_{\{x_i > \xi\}} + \epsilon_i \tag{4.8}$$

where we impose

$$a + b\xi = c + d\xi \tag{4.9}$$

[3] One method is to use the asymptotic confidence interval of Theorem 2.8.1. A second method is the bootstrap: draw R bootstrap replicates of $\vec{Y}$ and obtain R estimates of $\vec{\beta}$. Use the order statistics of the bootstrap estimates to obtain confidence intervals.

In other words, we assume the throughput response curve to be piecewise linear. Equation (4.9) expresses that the curve is continuous. Recall that x_i is the offered load and Y_i is the actual throughput.

Here we take $\vec{\beta} = (a, b, d)$ (we can derive $c = a + (b - d)\xi$ from Equation (4.9)). The dependency of Y_i on $\vec{\beta}$ is indeed linear. Note that we assume that ξ is known (see in exercise how to handle the case where ξ is to be identified).

Assume that we sort the x_is in increasing order and let i^* be the largest index i such that $x_i \leq \xi$. Re-write Equation (4.8) as

$$\begin{aligned} Y_i &= a + bx_i + \epsilon_i \text{ for } i = 1 \ldots i^* \\ Y_i &= a + b\xi + d(x_i - \xi) + \epsilon_i \text{ for } i = i^* + 1 \ldots I \end{aligned}$$

thus the matrix X is given by:

$$\begin{pmatrix} 1 & x_1 & 0 \\ 1 & x_2 & 0 \\ \ldots & \ldots & \ldots \\ 1 & x_{i^*} & 0 \\ 1 & \xi & x_{i^*+1} - \xi \\ \ldots & \ldots & \ldots \\ 1 & \xi & x_I - \xi \end{pmatrix}$$

It is simple to see that a *sufficient* condition for **H** is that there are at least two distinct values of $x_i \leq \xi$ and at least one value $> \xi$.

QUESTION 4.3.1. *Show this.* [4]

A model as in this example is sometimes called *Intervention Analysis*.

With the linear regression model, the manifold mentioned in the discussion after Theorem 4.2.1 is a linear manifold (for $p = 2$, a plane). It is equal to the linear sub-space spanned by the columns of matrix X. The nearest point is given by an orthogonal projection, which can be computed exactly. The details are given in the following theorem.

[4] We need to show, if the condition is true, that the matrix X has rank $p = 3$. This is equivalent to saying that the equation

$$X \begin{pmatrix} a \\ b \\ d \end{pmatrix} = 0$$

has only the solution $a = b = d = 0$. Consider first a and b. If there are two distinct values of x_i, $i \leq i^*$, say x_1 and x_2 then $a + bx_1 = a + bx_2 = 0$ thus $a = b = 0$. Since there is a value $x_i > \xi$, it follows that $i^* + 1 \leq I$ and $d(x_I - \xi) = 0$ thus $d = 0$.

THEOREM 4.3.1 (Linear Regression). *Consider the model in Definition 4.3.1; let $\vec{y}$ be the $I \times 1$ column vector of the data.*

1. *The $p \times p$ matrix $(X^T X)$ is invertible*
2. *(Estimation) The maximum likelihood estimator of $\vec{\beta}$ is $\hat{\beta} = K\vec{y}$ with $K = (X^T X)^{-1} X^T$*
3. *(Standardized Residuals) Define the ith residual as $e_i = \left(\vec{y} - X\hat{\beta}\right)_i$. The residuals are zero-mean gaussian but are correlated, with covariance matrix $\sigma^2(Id_I - H)$, where $H = X(X^T X)^{-1} X^T$.*
   Let $s^2 = \frac{1}{I-p} \|e\|^2 = \frac{1}{I-p} \sum_i e_i^2$ (rescaled sum of squared residuals). s^2 is an unbiased estimator of σ^2.
   The standardized residuals defined by $r_i := \frac{e_i}{s\sqrt{1-H_{i,i}}}$ have unit variance and $r_i \sim t_{N-p-1}$. This can be used to test the model by checking that r_i are approximately normal with unit variance.
4. *(Confidence Intervals) Let $\gamma = \sum_{j=1}^{p} u_j \beta_j$ be a (non-random) linear combination of the parameter $\vec{\beta}$; $\hat{\gamma} = \sum_j u_j \hat{\beta}_j$ is our estimator of γ. Let $g = \sum_k \left(\sum_j u_j K_{j,k}\right)^2$ (variance bias). Then $\frac{\hat{\gamma}-\gamma}{\sqrt{g}s} \sim t_{N-p}$. This can be used to obtain a confidence interval for γ.*

Proof. tbd

□

Comments. Item 3 states that the residuals are (slightly) biased, and it is better to use standardized residuals.

The matrix H is the projection onto the subspace spanned by the columns of X.

The predicted response is $\hat{y} = X\hat{\beta}$. It is equal to the orthogonal projection of $\vec{y}$. and is given by

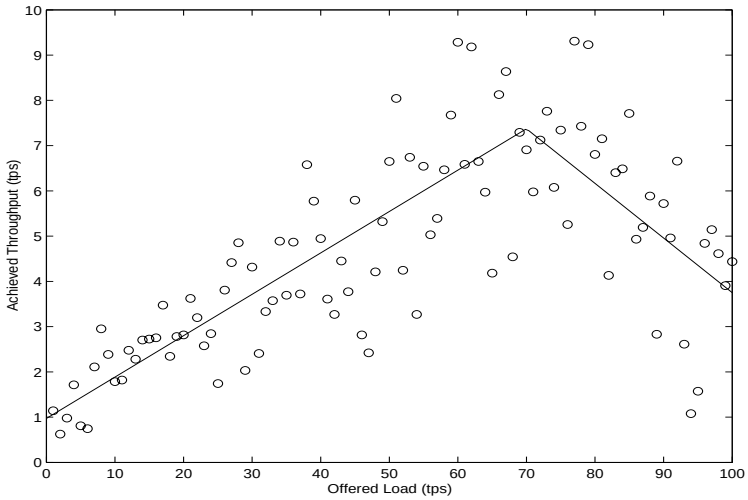
$$\hat{y} = H\vec{y} \tag{4.10}$$

The scaled sum of squared residuals s^2 is also equal to $\frac{1}{I}\left(\|\vec{y}\|^2 - \|\hat{y}\|^2\right)$. Its distribution is $\frac{1}{I}\chi^2_{I-p}$. This can be used to compute a confidence interval for σ.

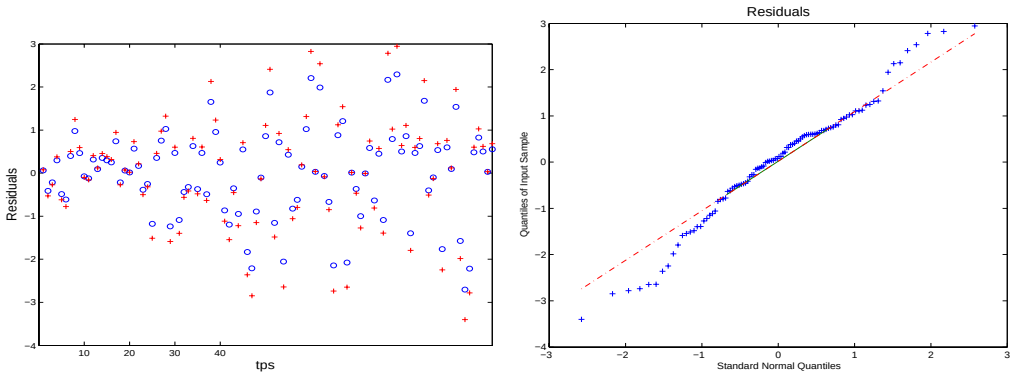
The proof of the theorem shows a slightly stronger result than item 4: the joint distribution of $\hat{\beta}$ is gaussian with mean $\vec{\beta}$ and covariance matrix $\sigma^2 KK^T$, and $\hat{\beta}$ is independent of e.

EXAMPLE: JOE'S SHOP AGAIN. CONTINUATION OF EXAMPLE 4.3. We can thus apply matrix computations given in Theorem 4.3.1; item 2 gives an estimate of (a, b, d) and thus of c. Item 4 gives confidence intervals. The values and the fitted linear regression model are shown in the table and figure below.

a	0.978 ± 0.609
b	0.0915 ± 0.0137
c	15.8 ± 2.99
d	-0.121 ± 0.037



We also computed the residuals e_i (crosses) and standardized residuals r_i (circles). There is little difference between both types of residuals. They appear reasonably normal, but one might criticize the model in that the variance appears smaller for smaller values of x. The normal qqplot of the residuals also shows approximate normality (the qqplot of standardized residuals is similar and is not shown).



QUESTION 4.3.2. *Can we conclude that there is congestion collapse ?* [5]

WHERE IS LINEARITY ? In the previous example, we see that that y_i is a linear function of $\vec{\beta}$, but **not of** x_i. This is quite general, and you should avoid a widespread confusion: linear regression is not restricted to models where the data y_i is linear with the explanatory variables x_i.

BEYOND THE LINEAR CASE

EXAMPLE: JOE'S SHOP - ESTIMATION OF ξ. In Example 4.3 on page 94 we assumed that the value ξ after which there is congestion collapse is known in advance. Now we relax this assumption. Our model is now the same as Equation (4.8), except that ξ is also now a parameter to be estimated.

[5] Yes, since the confidence interval for d is entirely positive [resp. negative].

To do this, we apply maximum likelihood estimation. We have to maximize the log-likelihood $l_{\vec{y}}(a, b, d, \xi, \sigma)$, where $\vec{y}$, the data, is fixed. For a fixed ξ, we know the value of (a, b, d, σ) that achieves the maximum, as we have a linear regression model. We plot the value of this maximum versus ξ (Figure 4.1) and numerically find the maximum. It is for $\xi = 77$.

To find a confidence interval, we use the asymptotic result in Theorem 2.8.2. It says that a 95% confidence interval is obtained by solving $l(\hat{\xi}) - l(\xi) \leq 1.9207$, which gives $\xi \in [73, 80]$.

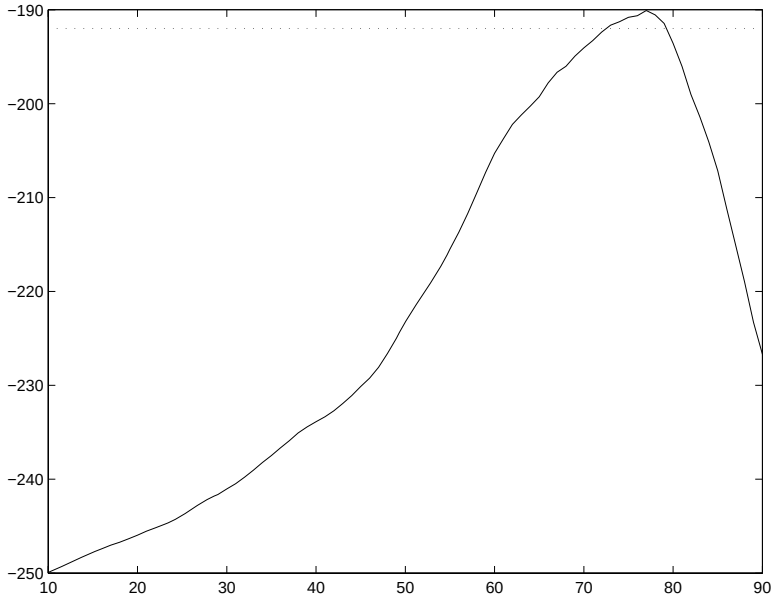


Figure 4.1: Log likelihood for Joes' shop as a function of ξ.

4.4 N-WAY ANOVA

Called *N-Way ANOVA*, it is a special case of linear regression, which is often used to capture the effect of n qualitative factors. *ANOVA* stands for "Analysis of Variance", because all statistical tests and estimations can be expressed from the sample variance (sums of squares). It is also a special case of the ANOVA model introduced in Section 7.4.1.

We describe the model for $n = 2$ (it is called in the statistics literature "2-way ANOVA with replicates"). For general values of n, the concepts are similar, but the notation becomes heavy.

DEFINITION 4.4.1 (*2-Way ANOVA*). *The statistical model is*

$$Y[i, j, r] = a + b[i] + c[j] + d[i, j] + \epsilon[i, j, r] \tag{4.11}$$

with $i = 1, ...I$, $j = 1, ..., J$, $r = 1, ..., R$ *and* $\epsilon[i, j, r]$ *are iid* $\sim N_{0,\sigma^2}$.

The variables i, j *are called factors (they take values in a discrete set). A possible value of a factor is called a level (here the levels are* $1...I$ *for the first factor).* $Y[i, j, r]$ *represents the value of the*

rth replicate of an experiment with 2 factors, when factor 1 has the value i and factor 2 has value j.

The coefficients $a, b[i], c[j], d[i,j]$ are called effects. The coefficients $d[i,j]$ are called interactions.

The additive model is a variant model where we force $d[i,j] = 0$.

To avoid underspecification, we impose the constraints:

$$\begin{cases} \sum_i b[i] = 0 \\ \sum_j c[j] = 0 \\ \text{for all } j \sum_i d[i,j] = 0 \text{ and } \text{for all } i \sum_j d[i,j] = 0 \end{cases} \tag{4.12}$$

The N-Way ANOVA model is a special case of linear regression, and everything we saw in Section 4.3 applies. The parameter $\vec{\beta}$ is the array $(a, b[], c[], d[,])$, subject to the constraints in Equation (4.12). Its dimension is $p = IJ$. The manifold spanned by the columns of the matrix X is the set of arrays $z[i,j,r]$ that depend only on i and j. Its dimension is also $p = IJ$, which shows that condition **H** in Definition 4.3.1 is satisfied. For the additive model, the dimension of the parameter is $p = I + J - 1$.

N-Way ANOVA is also a special case of the ANOVA model used for tests in Section 7.4.1: the random variables $Y[i,j,r]$ are gaussian with mean

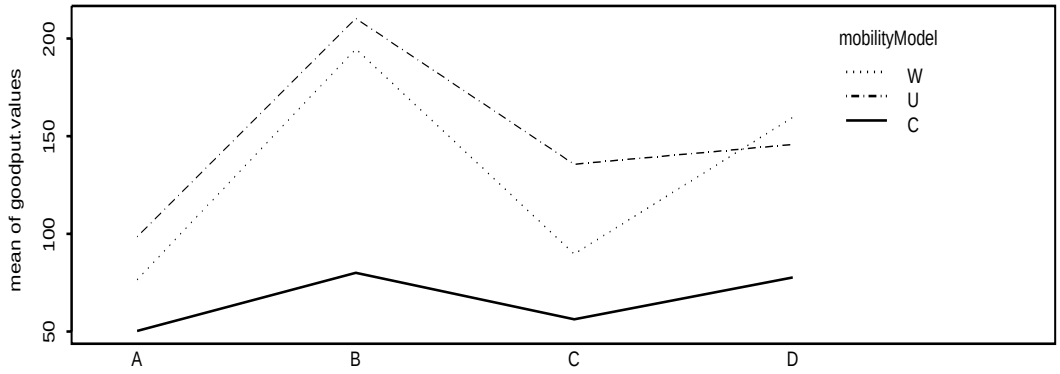
$$\mu[i,j] = a + b[i] + c[j] + d[i,j] \tag{4.13}$$

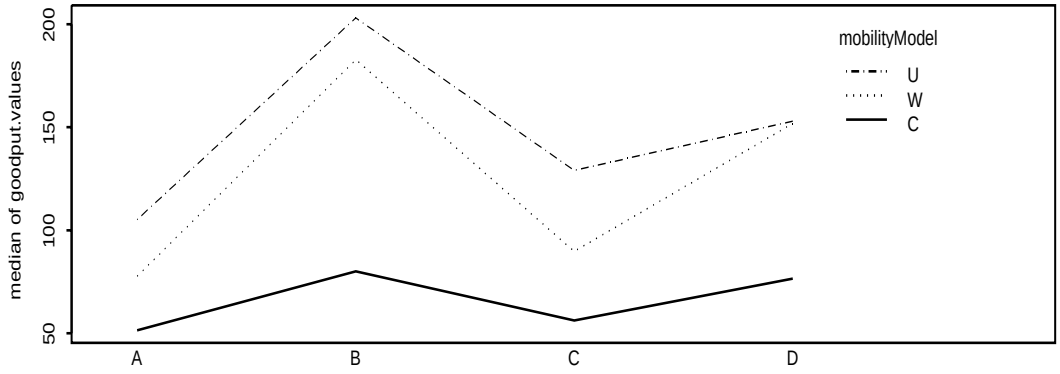
and common variance σ^2. Note that the constraints in Equation (4.12) do not put any restrictions on $\mu[i,j]$: any function $\mu[i,j]$ can be put in the form of Equation (4.13) with the constraints in Equation (4.12) being satisfied.

QUESTION 4.4.1. *Prove this.* [6]

EXAMPLE 4.4: MOBILE ROUTING. Consider the results of simulations that aim to compare 4 different routing protocols (A to D) proposed for mobile ad-hoc networks. Three mobility models (U, W and C) are used, and every experiment is repeated 4 times. The performance metric is the throughput achieved by the network. The figures show the values of the mean and median of 6 replicates.

[6] We are given the array $\mu[i,j]$, we want to find a, $b[]$, $c[]$ and $d[]$ such that Equation (4.13) and Equation (4.12) hold. Take $a = \frac{1}{IJ}\sum_{i,j}\mu[i,j]$, $b[i] = \frac{1}{J}\sum_j \mu[i,j] - a$, $c[j] = \frac{1}{I}\sum_i \mu[i,j] - a$, and $d[i,j] = \mu[i,j] - b[i] - c[j] - a$. One can verify that all required equations hold.





The model is 2-way ANOVA with 6 replicates.

If we believe that mobility does not affect the performance of a routing protocol, then we should have all interaction terms equal to 0. We will see in this section that this can be exactly tested.

The results in Theorem 4.3.1 have a simpler form, given by the following result.

THEOREM 4.4.1 (Estimation of ANOVA Model). *The maximum likelihood estimate of the parameters of the model in Definition 4.4.1 is given by*

$$\begin{aligned} \hat{a} &= \bar{y} \\ \hat{b}[i] &= \bar{y}[i,.,.] - \bar{y} \\ \hat{c}[j] &= \bar{y}[.,j,.] - \bar{y} \\ \hat{d}[i,j] &= \bar{y}[i,j,.] - \bar{y}[i,.,.] - \bar{y}[.,j,.] + \bar{y} \end{aligned}$$

where $\bar{y} = \frac{1}{IJR}\sum_{i,j,r} y[i,j,r]$ *and a notation such as* $\bar{y}[i,.,.]$ *means the average of the subarray of* y *obtained when* i *is fixed. So for example* $\bar{y}[i,.,.] = \frac{1}{JR}\sum_{j,r} y[i,j,r]$ *and* $\bar{y}[i,j,.] = \frac{1}{R}\sum_{r} y[i,j,r]$.

The variance biases are

$$\begin{aligned} g_a &= \frac{1}{IJR} \\ g_{b[i]} &= \frac{I-1}{IJR} \\ g_{c[j]} &= \frac{J-1}{IJR} \\ g_{d[i,j]} &= \frac{(I-1)(J-1)}{IJR} \end{aligned}$$

For the **additive model**, *the estimates* $\hat{a}, \hat{b}[i], \hat{c}[j]$ *are given by the same formulae, and so are the variances biases.*

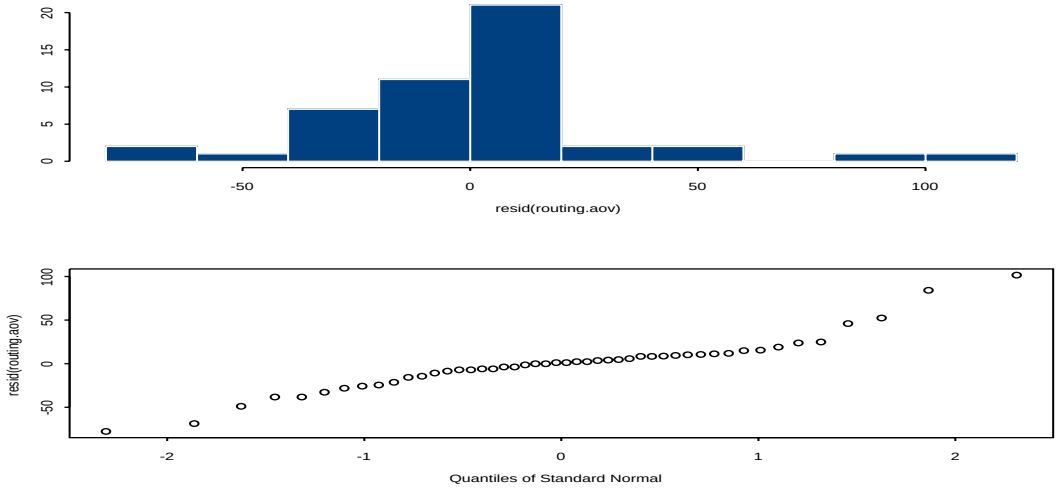
Proof.1. show that $a, b[], c[], d[,]$ satisfy the conditions in Equation (4.12).

2. Let $\hat{y}[i,j] = \hat{a} + \hat{b}[i] + \hat{c}[i] + \hat{d}[i,j]$. We want to show that $\hat{y}$ is the orthogonal projection on the subspace spanned by the columns of X. First, $\hat{y}$ belongs to the subspace by construction. So all we need now is to show that $\vec{y} - \hat{y}$ is orthogonal to the subspace. Check this by computing the inner product of a system of generating elements of the subspace.

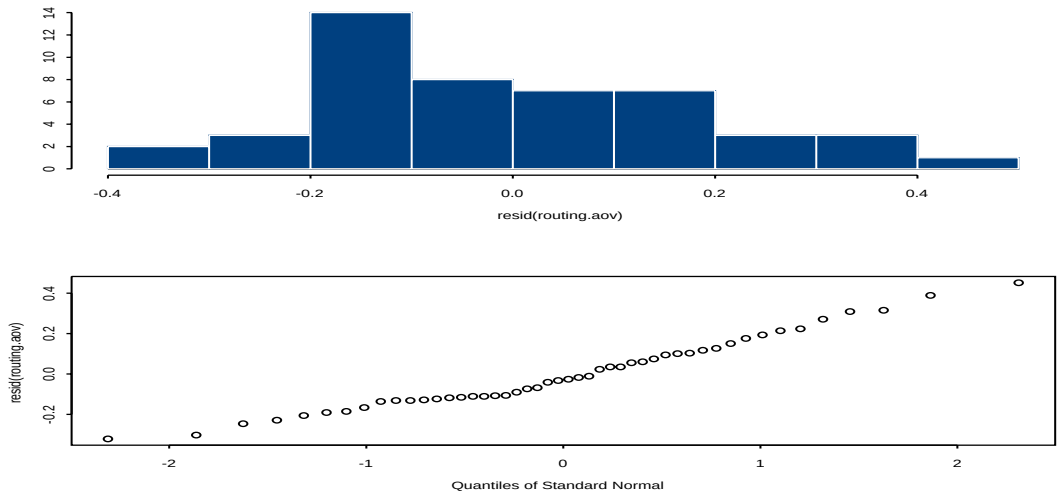
3. tbd

□

EXAMPLE 4.5: MOBILE ROUTING, CONTINUED. We applied the formulae and computed the residuals. The histogram and qqplot indicate a large deviation from normality.



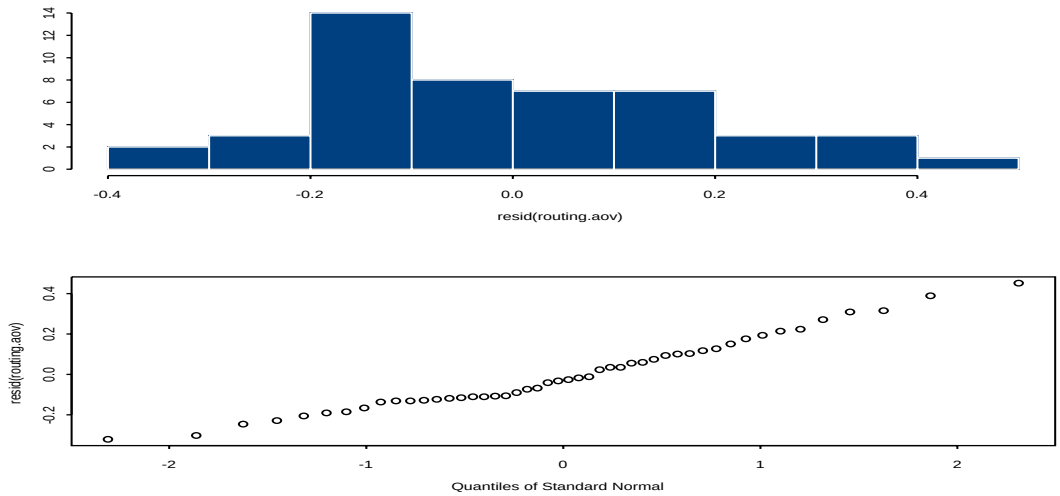
We try a Box-Cox transformation and find that changing Y to $1/Y$ does give more satisfactory residuals.



We also tried the additive model to the transformed data, namely

$$1/Y[i,j,k] = a + b[i] + c[j] + \epsilon[i,j,k]$$

We re-apply the formulae to this model and find residuals as shown below.



For the additive model, the estimated effects are:

```
Tables of effects (effect, 0.95 confidence interval)
 routing
       A        B        C        D
 0.37544 -0.31829  0.13596 -0.19311
 0.10321  0.10321  0.10321  0.10321
```

They show that A and C perform significantly better (if we believe in the additive model, which we will discuss next)

QUESTION 4.4.2. *Can you compare a confidence interval for* $b[1] - b[3]$ *?* [7]

4.5 FACTORIAL ANALYSIS

4.5.1 INTRODUCTION

The goal of *factorial analysis* is to understand the impact of each factor on some performance metric. In general, it is performed by an exhaustive application of the scientific method, as illustrated in Section 1.4.1. This may be time consuming as the number of possible combinations of factors may suffer from combinatorial explosion.

In some special cases where the "ANOVA" linear regression model holds, it is possible to have powerful results in relatively few computations. This is the main result of this section. Before studying the ANOVA Factorial Analysis model, we first see in the next section the nature of the difficulty.

4.5.2 ITERATIVE APPLICATION OF TWO FACTOR ANALYSIS

A simple way to do factorial analysis, when there are few factors, is to test the inclusion of factors one by one. This is called *Two Factor Analysis*. We explain it on one example:

EXAMPLE 4.6: We would like to interpret the data in Figure 4.2 with the model

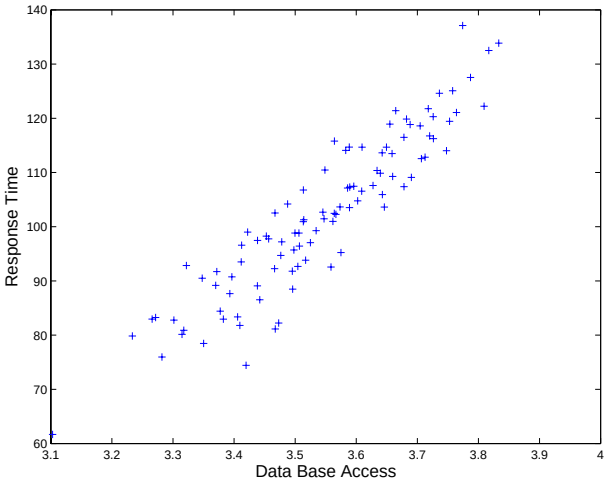
$$Z_i = a + bx_i + cy_i + \epsilon_i \tag{4.14}$$

where Z_i is the measured response time for a transaction i submitted to an information system, x_i is the number of database accesses required by this transaction, and y_i is the number of disk accesses.

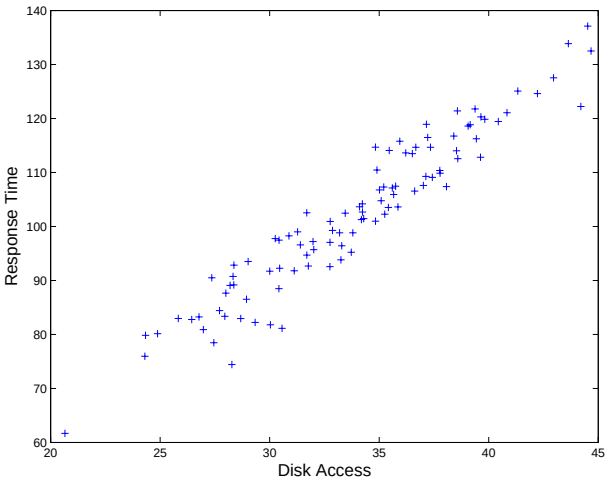
We first ask whether the combination of database and disk accesses is required to explain the data. Since the model in Equation (4.14) fits in the ANOVA framework, we can apply Theorem 7.4.1. More precisely, we test H_0: $b = c = 0$ versus H_1: $(b, c) \neq (0, 0)$. The result shows that we should reject H_0, i.e. the parameter (b, c) is significant.

```
            Df Sum of Sq  Mean Sq  F Value      Pr(F)
diskAc+dbAc 2  19685.91  9842.95  5.4e+002        0
 Residuals 97   1780.22    18.35
```

[7] We need to compute the variance bias g for $b[1] - b[3]$. We find $g = \frac{2}{JK}$. Thus the variance of the estimator of $b[1] - b[3]$ is $\frac{2s^2}{JK} = 0.0842$. The confidence interval for $b[1] - b[3]$ is thus 0.23948 ± 0.1701. It does not contain 0 thus A is better than B.



(a) Response times Z_i versus database access counts x_i



(b) Response times Z_i versus disk access counts y_i

Figure 4.2: Data for Example 4.6 on page 103

So, at this point, we conclude that the full model in Equation (4.14) is required, namely the response time is influenced by the disk and memory access counts.

We continue the analysis and ask whether, given that we accept database access in the model ($b \neq 0$), the second factor disk access is also required. We test H_0: $c = 0$ versus H_1: $c \neq 0$. The result below shows that c is *not* significant at size 0.05 (given that b is accepted in the model).

```
           Df Sum of Sq  Mean Sq  F Value      Pr(F)
   dbAc     1     55.02    55.02    2.998     0.087
Residuals 97   1780.22    18.35
```

We repeat the analysis, but now adding c before b, i.e. we test H_0: $b = 0$ versus H_1: $b \neq 0$. The result shows that the addition of b is significant !

```
           Df Sum of Sq  Mean Sq  F Value        Pr(F)
   diskAc  1   1413.42  1413.42  77.0141 5.87308e-014
Residuals 97   1780.22    18.35
```

We see that now, both database access and disk access are significant. To understand why this happened in this example, take a look at Figure 4.3. We see that data base access and disk access are strongly correlated, so adding data base access to disk access does not explain the data better (but the converse is not true).

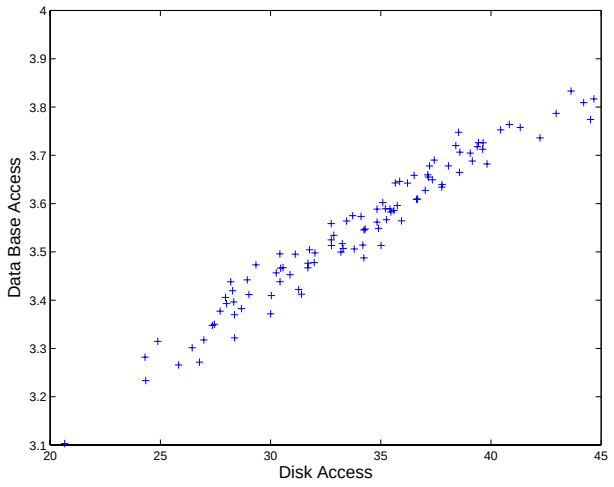


Figure 4.3: Database access counts x_i versus disk access counts y_i.

This example illustrates that when testing factors one by one, the answer may depends on the order with which the factors are considered. This is annoying, but is in the nature of the explanatory model, and is not an artifact of the statistical method.

We see next a case where this annoying phenomenon cannot occur.

4.5.3 FACTORIAL ANALYSIS WITH ORTHOGONAL FACTORS

This is a practical case where we can analyze factors for themselves, independent of the order where we add them. In practice, we use it when the factors take a small number of discrete values, with an ANOVA model. The method is based on the following theoretical framework.

DEFINITION 4.5.1 (Orthogonal Factors). *Assume a Linear Regression Model as in Definition 4.3.1 and:*

1. *The parameter can be decomposed in a unique way as $\vec{\beta} = \vec{\beta_1} + \vec{\beta_2} + ... + \vec{\beta}_{m_0}$, where the component $\vec{\beta}_m \in \mathcal{B}_m$ represents factor m. $\mathcal{B}_m$ is a linear subspace of the set of parameters.*
2. *The decomposition is with orthogonal factors, i.e.$X(\mathcal{B}_m) \perp X(\mathcal{B}_{m'})$ for all $m \neq m'$.*

EXAMPLE: RESPONSE TIME AGAIN.What is the decomposition of factors being tested in Example 4.6 on page 103 ?

The model is $Z_i = a + bx_i + cy_i + \epsilon_i$.

One possible decomposition, in two factors, is

$$(a, b, c) = (a, 0, 0) + (0, b, c)$$

The first factor is a, the second is (b, c). The first test in Example 4.6 on page 103 says that the presence of (b, c) is significant, which is equivalent to accepting that $(b, c) \neq (0, 0)$.

Are the factors orthogonal ? $\mathcal{B}_1$ is the set of $(a, 0, 0), a \in \mathbb{R}$, and $X(\mathcal{B}_1)$ is the set of vectors of length $I = 100$ of the form

$$\begin{pmatrix} a \\ a \\ \dots \\ a \end{pmatrix} = a \begin{pmatrix} 1 \\ 1 \\ \dots \\ 1 \end{pmatrix} = a\vec{e}_1$$

Similarly, $\mathcal{B}_2$ is the set of $(0, b, c), b, c \in \mathbb{R}$, and $X(\mathcal{B}_2)$ is the set of vectors of the form $b\vec{e}_2 + b\vec{e}_2$ with

$$\vec{e}_2 = \begin{pmatrix} x_1 \\ \dots \\ x_i \\ \dots \\ x_I \end{pmatrix} \text{ and } \vec{e}_3 = \begin{pmatrix} y_1 \\ \dots \\ y_i \\ \dots \\ y_I \end{pmatrix}$$

The factors are orthogonal if and only if $< \vec{e}_1, \vec{e}_2 >=< \vec{e}_1, \vec{e}_3 >= 0$. This is not the case as $< \vec{e}_1, \vec{e}_2 >= \sum_i x_i \neq 0$.

An alternative decomposition, in three factors, is

$$(a, b, c) = (a, 0, 0) + (0, b, 0) + (0, 0, c)$$

There are three spaces $X(\mathcal{B}_j)$, $j = 1, 2, 3$, each generated by $\vec{e}_j$. They are not mutually orthogonal either, so this model does not satisfy Definition 4.5.1.

EXAMPLE: N-WAY ANOVA IS WITH ORTHOGONAL FACTORS. As in Definition 4.4.1, we give the details for $N = 2$.

The parameter $\vec{\beta}$ is the array $(a, b[], c[], d[,])$ and $\mathcal{B}$ is the set of $\vec{\beta}$ that satisfy the constraints in Equation (4.12). The set $X(\mathcal{B})$ is the set of arrays $z[i, j, r]$ that depend only on i and j. Its dimension is $p = IJ$.

Consider the decomposition of $\vec{\beta}$ into 4 factors: a, $b[]$, $c[]$ and $d[,]$. The first factor is not a real factor, it represents the constant term, the second $b[]$ represents the first true factor, the third factor $c[]$ represents the second true factor; the fourth factor $d[,]$ is the interaction between the two true factors.

With the constraints of the ANOVA model, the model is with orthogonal factors. To see why, we determine the spaces $X(\mathcal{B}_m)$, $m = 1...4$.

- $X(\mathcal{B}_1)$ is obtained by letting $b[] = c[] = d[,] = 0$ in $\vec{\beta}$. Thus it is the set of arrays $z^1[i, j, k]$ that are constants: $z^1[i, j, k] = a$. The dimension is $k_1 = 1$.
- $X(\mathcal{B}_2)$ is the set of arrays z^2 of the form $z^2[i, j, k] = b[i]$ for some values of $b[]$ such that $\sum_i b[i] = 0$. The dimension is $k_2 = I - 1$.
  It comes $< z^1, z^2 >= Ja \sum_i b[i] = 0$ thus $X(\mathcal{B}_1) \perp X(\mathcal{B}_2)$.
- Similarly, $X(\mathcal{B}_3)$ is the set of arrays z^3 of the form $z^3[i, j, k] = c[j]$ for some values of $c[]$ such that $\sum_j c[j] = 0$. The dimension is $k_3 = J - 1$ and $X(\mathcal{B}_1) \perp X(\mathcal{B}_3)$.
  Further, $< z^2, z^3 >= \sum_{i,j} b[i]c[j] = (\sum_i b[i])(\sum_j c[j]) = 0$ thus $X(\mathcal{B}_2) \perp X(\mathcal{B}_3)$.
- $X(\mathcal{B}_4)$ is the set of arrays z^4 of the form $z^4[i, j, k] = d[i, j]$ for some values of $d[]$ such that $\sum_i d[i, j] = \sum_j d[i, j] = 0$. The dimension is $k_4 = (I - 1)(J - 1)$. It follows similarly that $X(\mathcal{B}_4)$ is orthogonal to $X(\mathcal{B}_m)$, $m = 1, 2, 3$.

THEOREM 4.5.1. *Consider a linear regression model with orthogonal factors. Let $\hat{y}_m$ be the predicted response if we consider only factor m (i.e. if we let $\vec{\beta}_{m'} = 0$ for all $m' \neq m$). Let*

$$SS(m) = \|\hat{y}_m\|^2 \tag{4.15}$$

$$SSR = \left\|\vec{Y} - (\hat{y}_1 + ... + \hat{y}_{m_0})\right\|^2 \tag{4.16}$$

- *The predicted response (under the model without restrictions) is $\hat{y}_1 + ... + \hat{y}_{m_0}$.*
- *The likelihood ratio statistic for the test H_0: $\vec{\beta}_m = \vec{0}$ versus H_1: $\beta_m \neq \vec{0}$ is*

$$f := \frac{SS(m)/k_m}{SSR/(N - (k_1 + ... + k_{m_0}))} \tag{4.17}$$

It has a Fisher distribution with degrees of freedom as in the fraction. Its p-value is $1 - F_{k_m, N-(k_1+...+k_{m_0})}(f)$

The test with size α accepts the inclusion of the mth factor if the p-value is less *than α, i.e. if the F-statistic in the theorem is large.*

SSR is the sum of squared residuals; $SS(m)$ is classically interpreted as the variation in the data explained by factor m. To see why, let model m be defined as the model obtained if we accept only factors 1 to m (i.e if we let $\vec{\beta}_{m'} = 0$ for $m' > m$). Under model m the predicted response is $\hat{y}_1 + ... + \hat{y}_m$ and the residual sum of squares is $\left\| \vec{Y} - (\hat{y}_{m+1} + ... + \hat{y}_{m_0}) \right\|^2$. Thus the reduction in residual sum of squares obtained when we go from model $m-1$ to model m (i.e. we add factor m) is $SS(m)$.

Note that, with our assumptions, this reduction is independent of the order of the factors, and the annoying phenomenon reported in Example 4.6 on page 103 cannot occur here.

Proof. tbd in details. Follows from Theorem 12.5.1 and Theorem 12.5.2.

□

EXAMPLE: MOBILE ROUTING, CONTINUED. Since the factors are orthogonal, we can apply Theorem 4.5.1 to the original model. The F-tests indicate that the interactions are non-significant.

```
                     Df Sum of Sq  Mean Sq  F Value     Pr(F)
             routing  3  52580.60 17526.87 13.80558 0.0000038
        mobilityModel  2  58962.28 29481.14 23.22174 0.0000003
routing:mobilityModel  6  14277.35  2379.56  1.87433 0.1122506
            Residuals 36  45703.77  1269.55
```

However, we found in Example 4.5 on page 101 that the residuals indicate a large deviation from normality and that changing Y to $1/Y$ does give satisfactory residuals. The F-tests for $1/Y$ indicate that the interactions are non significant:

```
                     Df Sum of Sq  Mean Sq  F Value     Pr(F)
             routing  3  3.576561 1.192187 28.34307 0.0000000
        mobilityModel  2  6.110433 3.055216 72.63475 0.0000000
routing:mobilityModel  6  0.275198 0.045866  1.09042 0.3867329
            Residuals 36  1.514259 0.042063
```

This shows that an additive model is adequate, namely

$$1/Y[i,j,k] = a + b[i] + c[j] + \epsilon[i,j,k]$$

We re-apply Theorem 4.5.1 to this model and find the results below. This shows that both routing and mobility model play a role in the final result.

```
              Df Sum of Sq  Mean Sq  F Value        Pr(F)
      routing  3  3.576561 1.192187 27.98160 4.19193e-010
mobilityModel  2  6.110433 3.055216 71.70843 2.86000e-014
    Residuals 42  1.789456 0.042606
```

SPECIAL CASE: FACTORS WITH ONLY 2 LEVELS. When the indices i, j in Equation (4.11) take only two values (there are only two levels), then all factor subspaces $\mathcal{B}_i$ have dimension 1. The convention in this case is to label the two levels -1 and 1. The 2-way ANOVA model in Equation (4.11) can then be re-written as

$$Y[i,j,k] = a + bi + cj + dij + \epsilon[i,j,k]$$

where a, b, c, d are scalars and $i = \pm 1, j = \pm 1, k = 1...K$. This model is sometimes called $2^k r$ *factorial analysis* (here with $k = 2$ and $r = K$). See exercise 4.7 for an example.

4.6 APPLICATION TO MODELING: HIDDEN FACTORS

EXAMPLE 4.7: TCP THROUGHPUT. The data on Figure 4.4, left, suggests that throughput increases with mobility. The right plot shows the same data, but reveals the window size. The conclusion is inverted: throughput decreases with mobility. The hidden factor influences the final result: all experiments with low speed are for small window sizes.

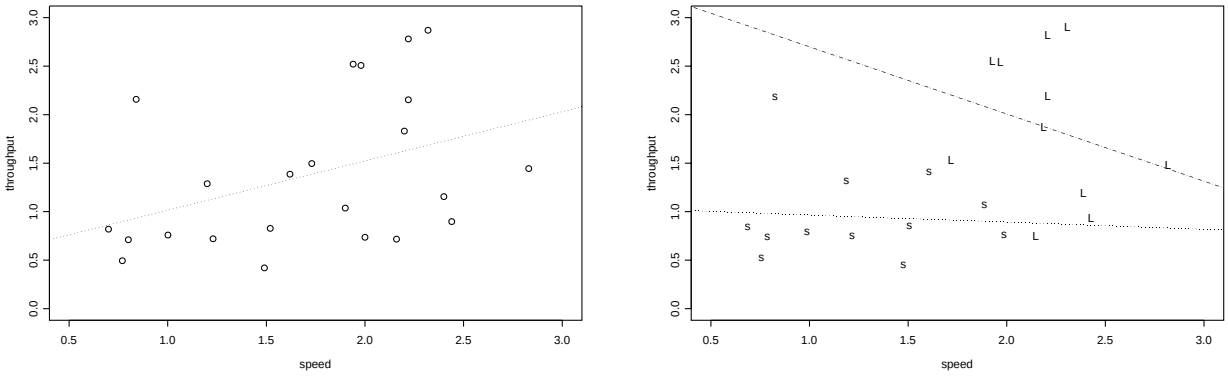


Figure 4.4: Left: plot of throughput versus speed for a mobile node. Right: same plot, but showing window size; s = small window, L = large window.

Avoiding hidden factors may be done by proper randomization of the experiments. On the example above, a proper design would have distributed window sizes randomly with respect to the speed. If randomization is not possible, then all factors have to be incorporated in the model.

QUESTION 4.6.1. *Give a linear regression model for Figure 4.4.* [8]

In conclusion: before stating that some factor has a given impact on the overall performance, make sure that there is no hidden factor that plays a role.

[8] Let Y_i be the throughput of the ith data point, s_i the speed, and $w_i = 1$ when the window size is small, $w_i = 2$ otherwise. A model is $Y_i = a_{w_i} + b_{w_i} s_i + \epsilon_i$. The unknown parameter is $\vec{\beta} = (a_1, a_2, b_1, b_2)$ with 4 degrees of freedom. The lines $y = a_i x + b_i$ are shown on Figure 4.4, right.

4.7 EXERCISES

EXERCISE 4.1. *Compute confidence intervals for Example 1 in Section 4.3.*

EXERCISE 4.2. *Compute the confidence interval for Example 3 by using Theorem* **??** *item 5 instead of the method above.*

EXERCISE 4.3. *Homework*

1. *Import the data of Table 1.3 by copying the file indicated in a complementary document. There is one single file for the whole dataset.*
2. *Do a linear regression of the response time as a function of the three factors and their interaction: user type, compiler option and experiment period. Give confidence intervals for the effects. Verify the residuals. Does it look convincing ? What effects are significant ? Can you conclude which compiler option is better*
3. *Do the same with* $1000/$*(response time). What is now the conclusion ?*
4. *Do the same analysis with user type ="R" and userType="L" separately. Can you conclude ?*

USEFUL MATLAB COMMANDS

- `anova1, anova2, anovan` perform linear regression for the N-way ANOVA model, i.e. for $N = 2$:
$$Y[i, j, k] = a + b[i] + c[j] + d[i, j] + \epsilon[i, j, k] \tag{4.18}$$
- `regress` solves the general model as in Theorem 4.3.1.

USEFUL S COMMANDS Read the S-PLUS guide to statistics, Chapter "Designed Experiments and Analysis of Variance", Section "The Two-Way Layout with Replicates"

Useful commands:

- `fac.design, data.frame`: create data structures
- `plot.design, plot.factor, interaction.plot`: graphical exploration
- `x <- aov, coefficients(x), model.tables(x,se=T)`: perform analysis of variance and display results with estimate of *standard deviation* of effect
- `lm, glm, gam`: normal, non normal linear regression (best least square estimator)

CHAPTER 5

PERFORMANCE PATTERNS

5.1 CONGESTION COLLAPSE

Consider a network where sources may send at a rate limited only by the source capabilities. Such a network may suffer of congestion collapse, which we explain now on an example.

We assume that the only resource to allocate is link bit rates. We also assume that if the offered traffic on some link l exceeds the capacity c_l of the link, then all sources see their traffic reduced in proportion of their offered traffic. This assumption is approximately true if queuing is first in first out in the network nodes, neglecting possible effects due to traffic burstiness.

Consider first the network illustrated on Figure 5.1. Sources 1 and 2 send traffic to destination nodes D1 and D2 respectively, and are limited only by their access rates. There are five links labeled 1 through 5 with capacities shown on the figure. Assume sources are limited only by their first link, without feedback from the network. Call λ_i the sending rate of source i, and $\lambda' i$ the outgoing rate.

For example, with the values given on the figures we find $\lambda_1 = 100$kb/s and $\lambda_2 = 1000$kb/s, but only $\lambda'_1 = \lambda'_2 = 10$kb/s, and the total throughput is 20kb/s ! Source 1 can send only at 10 kb/s because it is competing with source 2 on link 3, which sends at a high rate on that link; however, source 2 is limited to 10 kb/s because of link 5. If source 2 would be aware of the global situation, and if it would cooperate, then it would send at 10 kb/s only already on link 2, which would allow source 1 to send at 100 kb/s, without any penalty for source 2. The total throughput of the network would then become $\theta = 110kb/s$.

The first example has shown some inefficiency. In complex network scenarios, this may lead to a form of instability known as congestion collapse. To illustrate this, we use the network illustrated on Figure 5.2. The topology is a ring; it is commonly used in many networks, because it is a simple way to provide some redundancy. There are I nodes and links, numbered $0, 1, ..., I-1$. Source i enters node i, uses links $[(i+1) \bmod I]$ and $[(i+2) \bmod I]$, and leaves the network at node $(i+2) \bmod I$. Assume that source i sends as much as λ_i, without feedback from the network. Call λ'_i the rate achieved by source i on link $[(i+1) \bmod I]$ and λ''_i the rate achieved on link $[(i+2) \bmod I]$. This corresponds to every source choosing the shortest path to the destination. In

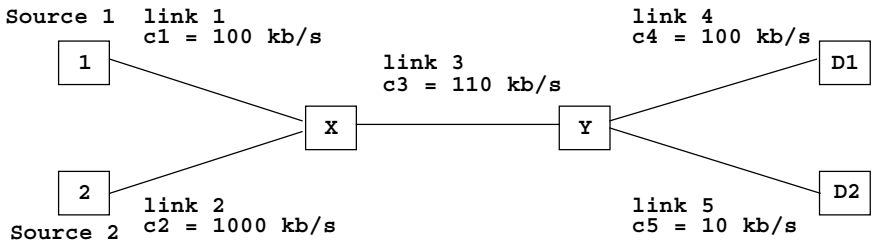


Figure 5.1: A simple network exhibiting some inefficiency if sources are not limited by some feedback from the network

the rest of this example, we omit "$\mathrm{mod}\, I$" when the context is clear. We have then:

$$\begin{cases} \lambda'_i = \min\left(\lambda_i, \frac{c_i}{\lambda_i + \lambda'_{i-1}}\lambda_i\right) \\ \lambda''_i = \min\left(\lambda'_i, \frac{c_{i+1}}{\lambda'_i + \lambda_{i+1}}\lambda'_i\right) \end{cases} \tag{5.1}$$

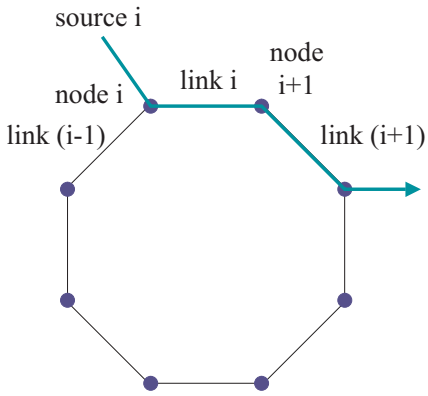


Figure 5.2: A network exhibiting congestion collapse if sources are not limited by some feedback from the network

Applying Equation 5.1 enables us to compute the total throughput θ. In order to obtain a closed form solution, we further study the symmetric case, namely, we assume that $c_i = c$ and $\lambda_i = \lambda$ for all i. Then we have obviously $\lambda'_i = \lambda$ and $\lambda''_i = \lambda''$ for some values of λ' and λ' which we compute now.

If $\lambda \leq \frac{c}{2}$ then there is no loss and $\lambda'' = \lambda' = \lambda$ and the throughput is $\theta = I\lambda$. Else, we have, from Equation (5.1)

$$\lambda' = \frac{c\lambda}{\lambda + \lambda'}$$

We can solve for λ' (a polynomial equation of degree 2) and obtain

$$\lambda' = \frac{\lambda}{2}\left(-1 + \sqrt{1 + 4\frac{c}{\lambda}}\right)$$

We have also from Equation (5.1)

$$\lambda'' = \frac{c\lambda'}{\lambda + \lambda'}$$

Combining the last two equations gives

$$\lambda'' = c - \frac{\lambda}{2}\left(\sqrt{1+4\frac{c}{\lambda}} - 1\right)$$

Using the limited development, valid for $u \to 0$

$$\sqrt{1+u} = 1 + \frac{1}{2}u - \frac{1}{8}u^2 + o(u^2)$$

we have

$$\lambda'' = \frac{c^2}{\lambda} + o(\frac{1}{\lambda})$$

Thus, the limit of the achieved throughput, when the offered load goes to $+\infty$, is 0. This is what we call *congestion collapse*.

Figure 5.3 plots the throughput per source λ'' as a function of the offered load per source λ. It confirms that after some point, the throughput decreases with the offered load, going to 0 as the offered load goes to $+\infty$.

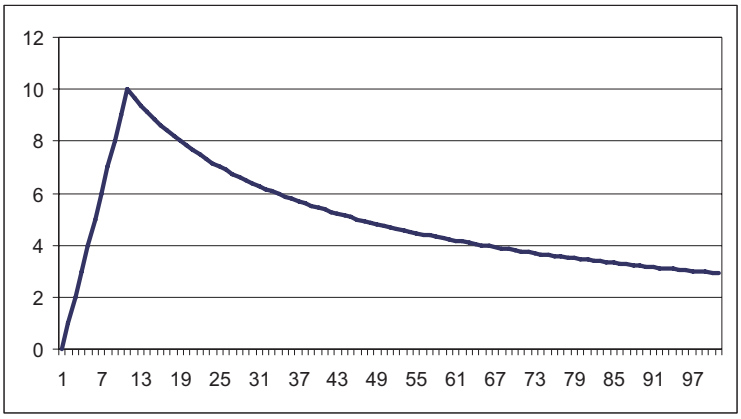


Figure 5.3: Throughput per source as a function of the offered load per source, in Mb/s, for the network of Figure 5.2. Numbers are in Mb/s. The link rate is $c = 20$Mb/s for all links.

The previous discussion has illustrated the following fact:

FACT 5.1.1 (Efficiency Criterion). *In a packet network, sources should limit their sending rate by taking into consideration the state of the network. Ignoring this may put the network into congestion collapse. One objective of congestion control is to avoid such inefficiencies.*

Congestion collapse occurs when some resources are consumed by traffic that will be later discarded. This phenomenon did happen in the Internet in the middle of the eighties. At that time, there was no end-to-end congestion control in TCP/IP. As we will see in the next section, a secondary objective is fairness.

QUESTION 5.1.1. *Can you imagine a congestion collapse scenario due to customer impatience ?* [1]

[1] to be done –see Hébuternes notes.

5.1.1 PUT MORE, GET LESS

QUESTION 5.1.2. *Can you imagine a system where adding capacity to a bottleneck makes things worse for every user ?* [2]

5.2 MULTI-USER PERFORMANCE

In many complex systems, there is not a single user and a global performance objective is not obviously defined. Maximizing the overall sum of individual performance metrics is not always intuitive.

5.2.1 EFFICIENCY VERSUS FAIRNESS

Assume that we want to maximize the network throughput, based on the considerations of the previous section. Consider the network example in Figure 5.4, where source i sends at a rate x_i, $i = 0, 1 \ldots, I$, and all links have a capacity equal to c. We assume that we implement some form of congestion control and that there are negligible losses. Thus, the flow on link i is $n_0 x_0 + n_i x_i$. For a given value of n_0 and x_0, maximizing the throughput requires that $n_i x_i = c - n_0 x_0$ for $i = 1, \ldots, I$. The total throughput, measured at the network output, is thus $Ic - (I-1)n_0 x_0$; it is maximum for $x_0 = 0$!

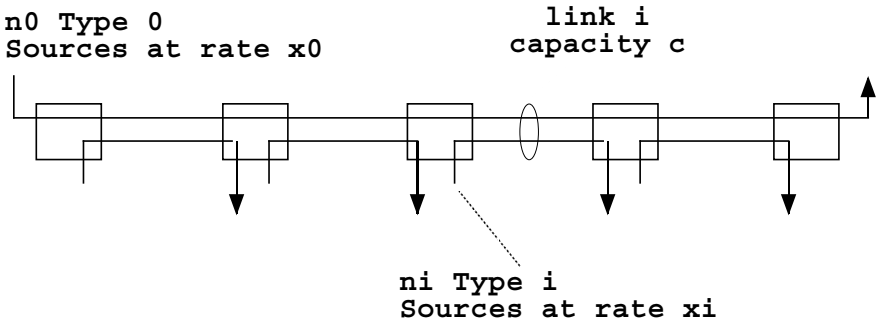


Figure 5.4: A simple network used to illustrate fairness (the"parking lot" scenario)

The example shows that maximizing network throughput as a primary objective may lead to gross unfairness; in the worst case, some sources may get a zero throughput, which is probably considered unfair by these sources.

5.2.2 MAX-MIN FAIRNESS

In a simple vision, fairness simply means allocating the same share to all. In the simple case of Figure 5.4 with $n_i = 1$ for all i, this would mean allocating $x_i = \frac{c}{2}$ to all sources $i = 0, \ldots, I$. However, in the case of a network, such a simple view does not generally make sense.

[2] Here is one example. Consider Figure 5.2 and assume now that the access link rate rate for every source is limited to 6 Mb/s. For every user, the bottleneck is the access link, and the throughput per user is 6 Mb/s. Assume now that we multiply the access link rate by 10. Figure 5.3 shows that the throughput decreases to 4 Mb/s.

Consider again the example of Figure 5.4, now with general values for n_i. If we follow the previous line of reasoning, we would allocate the fraction $\frac{c}{n_0+n_i}$ to each of the $n_0 + n_i$ sources using link i. This yields $x_i = \frac{c}{n_0+n_i}$ for $i \geq 1$; for $i = 0$, the reasoning of the previous section indicates that we should allocate $x_0 = \min_{1\leq i\leq I} \frac{c}{n_0+n_i}$. For example, with $I = 2$, $n_0 = n_1 = 1$ and $n_2 = 9$, we would allocate $x_0 = 0.1c$, $x_1 = 0.5c$ and $x_2 = 0.1c$. This allocation however would not fully utilize link 1; we could decide to increase the share of sources of type 1 since this can be done without decreasing the shares of other sources. Thus, a final allocation could be $x_0 = 0.1c$, $x_1 = 0.9c$ and $x_2 = 0.1c$. We have illustrated that allocating resources in an equal proportion is not a good solution since some sources can get more that others without decreasing others' shares. Formally, this leads to our first definition of fairness called max-min fairness.

Consider an allocation problem; define the vector $\vec{x}$ whose ith coordinate is the allocation for user i. Let $\mathcal{X}$ be the set of all feasible allocations.

DEFINITION 5.2.1 (Max-min Fairness). *[1]A feasible allocation of rates $\vec{x}$ is "max-min fair" if and only if an increase of any rate within the domain of feasible allocations must be at the cost of a decrease of some already smaller rate. Formally, for any other feasible allocation $\vec{y}$, if $y_s > x_s$ then there must exist some s' such that $x_{s'} \leq x_s$ and $y_{s'} < x_{s'}$.*

Depending on the problem, a max-min fair allocation may or may not exist. However, if it exists, it is unique (see later for a proof). We develop the theory in a special case where existence is always guaranteed. For a general set of results, see [Radunovic02-Allerton].

NETWORK MODEL We use the following simplified network model in the rest of this section. We consider a set of sources $s = 1, \ldots, S$ and links $1, \ldots, L$. Let $A_{l,s}$ be the fraction of traffic of source s which flows on link l, and let c_l be the capacity of link l. We define a network as the couple $(\vec{x}, A)$.

A *feasible allocation* of rates $x_s \geq 0$ is defined by: $\sum_{s=1}^{S} A_{l,s}x_s \leq c_l$ for all l.

Our network model supports both multicast and load sharing. For a given source s, the set of links l such that $A_{l,s} > 0$ is the path followed by the data flow with source s. In the simplest case (no load sharing), $A_{l,s} \in \{0, 1\}$; if a flow from source s is equally split between two links l_1 and l_2, then $A_{l_1,s} = A_{l_2,s} = 0.5$. In principle, $A_{l,s} \leq 1$, but this is not mandatory (in some encapsulation scenarios, a flow may be duplicated on the same link).

It can be seen (and this is left as an exercise) that the allocation in the previous example is max-min fair. The name "max-min" comes from the idea that it is forbidden to decrease the share of sources that have small values, thus, in some sense, we give priority to flows with small values.

In general, we might ask ourselves whether there exists a max-min fair allocation to our network model, and how to obtain it. This will result from the key concept of "bottleneck link".

DEFINITION 5.2.2 (Bottleneck Link). *With our network model above, we say that link l is a bottleneck for source s if and only if*

1. *link l is saturated: $c_l = \sum_i A_{l,i}x_i$*
2. *source s on link l has the maximum rate among all sources using link l: $x_s \geq x_{s'}$ for all s' such that $A_{l,s'} > 0$.*

Intuitively, a bottleneck link for source s is a link which is limiting, for a given allocation. In the previous numerical, example, link 2 is a bottleneck for sources of type 0 and 2, and link 1 is a bottleneck for the source of type 1.

THEOREM 5.2.1. *A feasible allocation of rates $\vec{x}$ is max-min fair if and only if every source has a bottleneck link.*

PROOF: **Part 1.** Assume that every source has a bottleneck link. Consider a source s for which we can increase the rate x_s while keeping the allocation feasible. Let l be a bottleneck link for s. Since l is saturated, it is necessary to decrease $x_{s'}$ for some s' such that $A_{l,s'} > 0$. We assumed that we can increase the rate of s: thus there must exist some $s' \neq s$ that shares the bottleneck link l. But for all such s', we have $x_s \geq x_{s'}$, thus we are forced to decrease $x_{s'}$ for some s' such that $x_s \geq x_{s'}$: this shows that the allocation is max-min fair.

Part 2. Conversely, assume that the allocation is max-min fair. For any source s, we need to find a bottleneck link. We proceed by contradiction. Assume there exists a source s with no bottleneck link. Call L_1 the set of saturated links used by source s, namely, $L_1 = \{l \text{ such that } c_l = \sum_i A_{l,i}x_i \text{ and } A_{l,s} > 0\}$. Similarly, call L_2 the set of non-saturated links used by source s. Thus a link is either in L_1 or L_2, or is not used by s. Assume first that L_1 is non-empty.

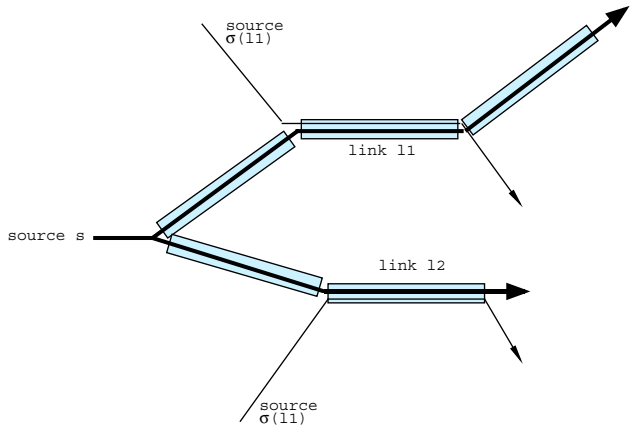


Figure 5.5: A network example showing one multicast source

By our assumption, for all $l \in L_1$, there exists some s' such that $A_{l,s'} > 0$ and $x_{s'} > x_s$. Thus we can build a mapping σ from L_1 into the set of sources $\{1, \ldots, S\}$ such that $A_{l,\sigma(l)} > 0$ and $x_{\sigma(l)} > x_s$ (see Figure 5.5 for an illustration). Now we will show that we can increase the rate x_s in a way that contradicts the max-min fairness assumption. We want to increase x_s by some value δ, at the expense of decreasing $x_{s'}$ by some other values $\delta_{s'}$, for all s' that are equal to some $\sigma(l')$. We want the modified allocation to be feasible; to that end, it is sufficient to have:

$$A_{l,s}\delta \leq A_{l,\sigma(l)}\delta_{\sigma(l)} \quad \text{for all } l \in L_1 \tag{5.2}$$

$$A_{l,s}\delta \leq c_l - \sum_i A_{l,i}x_i \quad \text{for all } l \in L_2 \tag{5.3}$$

$$\delta_{\sigma(l)} \leq x_{\sigma(l)} \quad \text{for all } l \in L_1 \tag{5.4}$$

Equation (5.2) expresses that the increase of flow due to source s on a saturated link l is at least compensated by the decrease of flow due to source $\sigma(l)$. Equation (5.3) expresses that the increase of flow due to source s on a non-saturated link l does not exceed the available capacity. Finally, equation (5.4) states that rates must be non-negative.

This leads to the following choice.

$$\delta = \min_{l \in L_1}\{\frac{x_{\sigma(l)} A_{l,\sigma(l)}}{A_{l,s}}\} \wedge \min_{l \in L_2}\{\frac{c_l - \sum_i A_{l,i} x_i}{A_{l,s}}\} \tag{5.5}$$

which ensures that Equation (5.3) is satisfied and that $\delta > 0$.

In order to satisfy Equations (5.2) and (5.4) we need to compute the values of $\delta_{\sigma(l)}$ for all l in L_1. Here we need to be careful with the fact that the same source s' may be equal to $\sigma(l)$ for more than one l. We define $\delta(s')$ by

$$\delta(s') \quad = 0 \text{ if there is no } l \text{ such that } s' = \sigma(l) \tag{5.6}$$

$$\delta(s') \quad = \max_{\{l \text{ such that } \sigma(l)=s'\}}\{\frac{\delta A_{l,s}}{A_{l,\sigma(l)}}\} \text{ otherwise} \tag{5.7}$$

This definition ensures that Equation (5.2) is satisfied. We now examine Equation (5.4). Consider some s' for which there exists an l with $\sigma(l) = s$, and call l_0 the value which achieves the maximum in (5.7), namely:

$$\delta(s') = \frac{\delta A_{l_0,s}}{A_{l_0,s'}} \tag{5.8}$$

From the definition of δ in (5.5), we have

$$\delta \leq \frac{x_{\sigma(l_0)} A_{l_0,\sigma(l_0)}}{A_{l_0,s}} = \frac{x_{s'} A_{l_0,s'}}{A_{l_0,s}}$$

Combined with (5.8), this shows that Equation (5.4) holds. In summary, we have shown that we can increase x_s at the expense of decreasing the rates for only those sources s' such that $s' = \sigma(l)$ for some l. Such sources have a rate higher than x_s, which shows that the allocation $\vec{x}$ is not max-min fair and contradicts our hypothesis.

It remains to examine the case where L_1 is empty. The reasoning is the same, we can increase x_s without decreasing any other source, and we also have a contradiction. □

THE ALGORITHM OF PROGRESSIVE FILLING The previous theorem is particularly useful in deriving a practical method for obtaining a max-min fair allocation, called "progressive filling". The idea is as follows. You start with all rates equal to 0 and grow all rates together at the same pace, until one or several link capacity limits are hit. The rates for the sources that use these links are not increased any more, and you continue increasing the rates for other sources. All the sources that are stopped have a bottleneck link. This is because they use a saturated link, and all other sources using the saturated link are stopped at the same time, or were stopped before, thus have a smaller or equal rate. The algorithm continues until it is not possible to increase. The algorithm terminates because L and S are finite. Lastly, when the algorithm terminates, all sources have been stopped at some time and thus have a bottleneck link. By application of Theorem 5.2.1, the allocation is max-min fair.

EXAMPLE Let us apply the progressive filling algorithm to the parking lot scenario. Initially, we let $x_i = 0$ for all $i = 0, \ldots, I$; then we let $x_i = t$ until we hit a limit. The constraints are

$$n_0 x_0 + n_i x_i \leq c \text{ for all } i = 1, \ldots, I$$

Thus the first constraint is hit at $t_1 = \min\{\frac{c}{n_0+n_i}\}$ and it concerns sources of type 0 and type i_0 for all values of index i_0 which minimize the expression above. Thus

$$x_0 = \min\{\frac{c}{n_0 + n_i}\}$$

In order to compute the rates for sources of other types, we continue to increase their rates. Now all constraints become independent and we finally have

$$x_i = \frac{c - n_0 x_0}{n_i}$$

If all n_i's are equal, the we see that all sources obtain the same rate. In some sense, max-min fairness ignores the fact that sources of type 0 use more network resources than those of type i, $i \geq 1$. In that case, the total throughput for the parking lot network is $\frac{(I+1)c}{2}$, which is almost half of the maximum admissible throughput of Ic.

THEOREM 5.2.2. *For the network defined above, with fixed routing parameters $A_{l,s}$, there exists a unique max-min fair allocation. It can be obtained by the algorithm of progressive filling.*

PROOF: We have already proven the existence. Assume now that $\vec{x}$ and $\vec{y}$ are two max-min fair allocations for the same problem, with $\vec{x} \neq \vec{y}$. Without loss of generality, we can assume that there exists some i such that $x_i < y_i$. Consider the smallest value of x_i that satisfies $x_i < y_i$, and call i_0 the corresponding index. Thus, $x_{i_0} < y_{i_0}$ and

$$\text{if } x_i < y_i \text{ then } x_{i_0} \leq x_i \tag{5.9}$$

Now since $\vec{x}$ is max-min fair, from Definition 5.2.1, there exists some j with

$$y_j < x_j \leq x_{i_0} \tag{5.10}$$

Now $\vec{y}$ is also max-min fair, thus by the same token there exists some k such that

$$x_k < y_k \leq y_j \tag{5.11}$$

Combining (5.10) and (5.11), we obtain

$$x_k < y_k \leq y_j < x_j \leq x_{i_0}$$

which contradicts (5.9). □

The notion of max-min fairness can be easily generalized by using weights in the definition [1, 3].

5.2.3 PROPORTIONAL FAIRNESS

The previous definition of fairness puts emphasis on maintaining high values for the smallest rates. As shown in the previous example, this may be at the expense of some network inefficiency. An alternative definition of fairness has been proposed in the context of game theory [4].

DEFINITION 5.2.3 (Proportional Fairness). *An allocation of rates $\vec{x}$ is "proportionally fair" if and only if, for any other feasible allocation $\vec{y}$, we have:*

$$\sum_{s=1}^{S} \frac{y_s - x_s}{x_s} \leq 0$$

In other words, any change in the allocation must have a negative average change. Let us consider for example the parking lot scenario with $n_s = 1$ for all s. Is the max-min fair allocation proportionally fair ?

To get the answer, remember that, for the max-min fair allocation, $x_s = c/2$ for all s. Consider a new allocation resulting from a decrease of x_0 equal to δ:

$$\begin{aligned} y_0 &= \tfrac{c}{2} - \delta \\ y_s &= \tfrac{c}{2} + \delta \; s = 1, \ldots, I \end{aligned}$$

For $\delta < \frac{c}{2}$, the new allocation $\vec{y}$ is feasible. The average rate of change is

$$\left(\sum_{s=1}^{I} \frac{2\delta}{c} \right) - \frac{2\delta}{c} = \frac{2(I-1)\delta}{c}$$

which is positive for $I \geq 2$. Thus the max-min fair allocation for this example is not proportionally fair for $I \geq 2$. In this example, we see that a decrease in rate for sources of type 0 is less important than the corresponding increase which is made possible for the other sources, because the increase is multiplied by the number of sources. Informally, we say that proportional fairness takes into consideration the usage of network resources.

Now we derive a practical result which can be used to compute a proportionally fair allocation. To that end, we interpret the average rate of change as $\nabla J_{\vec{x}} \cdot (\vec{y} - \vec{x})$, with

$$J(\vec{x}) = \sum_s \ln(x_s)$$

Thus, intuitively, a proportionally fair allocation should maximize J.

THEOREM 5.2.3. *There exists one unique proportionally fair allocation. It is obtained by maximizing $J(\vec{x}) = \sum_s \ln(x_s)$ over the set of feasible allocations.*

PROOF: We first prove that the maximization problem has a unique solution. Function J is concave, as a sum of concave functions. The feasible set is convex, as intersection of convex sets, thus any local maximum of J is an absolute maximum. Now J is strictly concave, which means that

$$\text{if } 0 < \alpha < 1 \text{ then } J(\alpha\vec{x} + (1-\alpha)\vec{y}) < \alpha J(\vec{x}) + (1-\alpha)J(\vec{y})$$

This can be proven by studying the second derivative of the restriction of J to any linear segment. Now a strictly concave function has at most one maximum on a convex set (Chapter **??**).

Now J is continuous if we allow $\log(0) = -\infty$ and the set of feasible allocations is compact (because it is a closed, bounded subset of $\mathbb{R}^S$). Thus J has at least one maximum over the set of feasible allocations.

Combining all the arguments together proves that J has exactly one maximum over the set of feasible allocations, and that any local maximum is also exactly the global maximum.

Then for any $\vec{\delta}$ such that $\vec{x} + \vec{\delta}$ is feasible,

$$J(\vec{x} + \vec{\delta}) - J(\vec{x}) = \nabla J_{\vec{x}} \cdot \vec{\delta} + \frac{1}{2}^t \vec{\delta} \nabla^2 J_{\vec{x}} \vec{\delta} + o(||\vec{\delta}||^2)$$

Now by the strict concavity, $\nabla^2 J_{\vec{x}}$ is definite negative thus

$$\frac{1}{2}^t \vec{\delta} \nabla^2 J_{\vec{x}} \vec{\delta} + o(||\vec{\delta}||^2) \ < 0$$

for $||\vec{\delta}||$ small enough.

Now assume that $\vec{x}$ is a proportionally fair allocation. This means that

$$\nabla(J)_{\vec{x}} \cdot \vec{\delta} \leq 0$$

and thus J has a local maximum at $\vec{x}$, thus also a global maximum. This also shows the uniqueness of a proportionally fair allocation.

Conversely, assume that J has a global maximum at $\vec{x}$, and let $\vec{y}$ be some feasible allocation. Call D the average rate of change:

$$D = \nabla(J)_{\vec{x}} \cdot (\vec{y} - \vec{x})$$

Since the feasible set is convex, the segment $[\vec{x}, \vec{y}]$ is entirely feasible, and

$$D = \lim_{t \to 0^+} \frac{J(\vec{x} + t(\vec{y} - \vec{x})) - J(\vec{x})}{t}$$

and thus $D \leq 0$. □

EXAMPLE Let us apply Theorem 5.2.3 to the parking lot scenario. For any choice of x_0, we should set x_i such that

$$n_0 x_0 + n_i x_i = c, i = 1, \ldots, I$$

otherwise we could increase x_i without affecting other values, and thus increase function J. The value of x_0 is found by maximizing $f(x_0)$, defined by

$$f(x_0) = n_0 \ln(x_0) + \sum_{i=1}^{I} n_i (\ln(c - n_0 x_0) - \ln(n_i))$$

over the set $0 \leq x_0 \leq \frac{c}{n_0}$. The derivative of f is

$$f'(x_0) = \frac{n_0}{x_0} - \frac{n_0}{c - n_0 x_0} \sum_{i=1}^{I} n_i$$

After some algebra, we find that the maximum is for

$$x_0 = \frac{c}{\sum_{i=0}^{I} n_i}$$

and

$$x_i = \frac{c - n_0 x_0}{n_i}$$

For example, if $n_i = 1$ for all $i = 0, \ldots, I$, we obtain:

$$\begin{aligned} x_0 &= \tfrac{c}{I+1} \\ x_i &= \tfrac{cI}{I+1} \end{aligned}$$

Compare with max-min fairness, where, in that case, the allocation is $\frac{c}{2}$ for all rates. We see that sources of type 0 get a smaller rate, since they use more network resources.

The concept of proportional fairness can easily extended to *rate* proportional fairness, where the allocation maximizes a weighted sum of logarithms [2].

UTILITY APPROACH Proportional fairness is an example of a more general fairness concept, called the "utility" approach, which is defined as follows. Every source s has a utility function u_s where $u_s(x_s)$ indicates the value to source s of having rate x_s. Every link l (or network resource in general) has a cost function g_l, where $g_l(f)$ indicates the cost to the network of supporting an amount of flow f on link l. Then, a "utility fair" allocation of rates is an allocation which maximizes $H(\vec{x})$, defined by

$$H(\vec{x}) = \sum_{s=1}^{S} u_s(x_s) - \sum_{l=1}^{L} g_l(f_l)$$

with $f_l = \sum_{s=1}^{S} A_{l,s} x_s$, over the set of feasible allocations.

Proportional fairness corresponds to $u_s = \ln$ for all s, and $g_l(f) = 0$ for $f < c_l$, $g_l(f) = +\infty$ for $f \geq c_l$. Rate proportional fairness corresponds to $u_s(x_s) = w_s \ln(x_s)$ and the same choice of g_l.

Computing utility fairness requires solving constrained optimization problems; a reference is [5].

MAX-MIN AS LIMITING CASE OF UTILITY FAIRNESS It can be shown that max-min fairness is a limiting case of utility fairness – see [Radunovic02-Allerton].

5.2.4 PUT MORE, GET LESS FOR SOME

If a multi-user performance criterion is used, then it can happen that adding some capacity decreases the performance experienced by some.

QUESTION 5.2.1. *Give an example where this happens.* [3]

[3] Consider the parking lot scenario above, with $I = 2$ nodes, $n_0 = n_1 = 1, n_2 = 9$, $c_1 = c_2 = 1$, and assume the system distributes rates in a max-min fair way. The max-min fair rate is $x_0 = x_2 = 0.1, x_1 = 0.9$. Now increase the capacity of link 2 to $c_2 = 10$. The max-min fair allocation is now $x_0 = x_1 = 0.5, x_2 = 1.044$. The rate of source 1 has decreased.

5.3 BRAESS PARADOX

We have seen earlier that adding capacity may decrease the performance seen by *some* users. In some cases, adding capacity may decrease the performance seen by *all* users. The *Braess paradox* in one such example, found in networking. It is another example of "Put more, get less".

Here is Braess' original example, slightly modified. Consider a network where users pick the routes with minimum delay. See Figure 5.6. Assume first that the delay on link 5 is infinite (the link is not open). The delay on link j is a function $D_j(\rho_j)$, where ρ_j is the load. Take $D_1(\rho) = D_4(\rho) = 2 + 10\rho$, $D_2(\rho) = D_3(\rho) = 48 + \rho$ and let the total load be $b_0 = 6$. Every user has the choice of a number of routes. Assume there are infinitely many small users. As a result, the traffic for a given source destination pairs uses only routes that minimize the delay. The resulting rate distribution is said to satisfy the *Wardrop Equilibrium* condition.

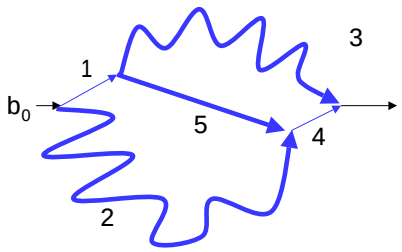


Figure 5.6: Network where the Braess paradox occurs. There are 5 links, labeled 1 to 5. Links 2 and 3 have a long fixed delay but high throughput. Links 1 and 4 have a small fixed delay but low throughput. Link 5 has medium fixed delay and high throughput. Assume all users pick a shortest delay path. Delays get worse for *all* users in the equilibrium reached after link 5 is opened.

For example, if we take $\rho_1 = 1$, $\rho_2 = 5$, the delay on route $1 - 3$ is 61, and on route $2 - 4$ it is 105; this is not a Wardrop equilibrium. But if we take $\rho_1 = \rho_2 = 3$, we have a Wardrop equilibrium and the mean delay for all is 83.

More generally, consider a network model as follows [Kelly91-nr]. J is the set of links and we define ρ_j D_j as above. S is the set of source destination pairs and R is the set of possible routes (not necessarily disjoint), where routes joining different source destination pairs considered to be distinct. Let $H_{s,r} = 1_{\{s \text{ uses } r\}}$ and $A_{j,r} = 1_{\{j \text{ is on } r\}}$. Let b_s be the total traffic demand of s, ν_r the distribution of route r. The Wardrop equilibrium is defined by the following conditions:

$$\begin{cases} H\nu = b \\ A\nu = \rho \\ \nu_r > 0 \Rightarrow \sum_{j \in r} D_j(\rho_j) = \min_{r' : H_{s,r'} = 1} \sum_{j \in r'} D_j(\rho_j) \end{cases}$$

QUESTION 5.3.1. *Write the Wardrop equilibrium conditions for the example above.* [4]

THEOREM 5.3.1 (Kelly91-nr). *Consider the network model above, and assume that $D_j(\rho)$ is continuous and increasing. There is one unique Wardrop equilibrium.*

The proof can be found in [Kelly91-rt]. It consists in associating the Wardrop equilibrium conditions to a convex optimization problem, and applying the strong duality principle.

Now let us come back to the example in Figure 5.6. Open link 5 and let its delay function be $f_5(\rho) = 6 + \rho$. The old allocation is not a Wardrop equilibrium; the new equilibrium is for $\rho_1 = 4$, $\rho_2 = 2$, $\rho_5 = 2$ and the mean delay is 92 for all. Adding a new link has made things worse for all !

This is because the equilibrium obtained by the individual decisions is not a social optimum. A general discussion of such concepts is the topic of game theory. We can also relate this example to our general discussion of bottlenecks: adding link 5 does not improve the capacity of the network, which is limited by links 1 and 3. However, this illustrates that adding capacity at the wrong place may make things worse.

A Wardrop equilibrium is, in some sense equivalent to what is called a Nash equilibrium in game theory. See [Altman01-survey] for a more accurate statement and a first introduction to game theory. See Exercise 5.5 for an example of Braess paradox with elastic traffic. See [Altman01-ITC17] for sufficient conditions for avoiding the Braess paradox.

5.4 NON MONOTONE EFFECTS IN QUEUING

5.4.1 PRIORITY QUEUES

Bramson queues. To be done from Deleval's simulation.

5.4.2 FIFO SYSTEMS

Matthew Andrew's paper on instability.

5.4.3 BELADY'S ANOMALY

Storing page references in FIFO mode leads to a situation similar to Braess' s paradox. See Table 5.1.

5.5 EXERCISES

EXERCISE 5.1. *Consider the intranet on Figure 5.7. There are three Ethernet segments at 10 Mb/s,each corresponding to a* `net:subnet` *prefix noted n_1, n_2 and n_3. Every Ethernet segment is connected to two routers as indicated on the figure. There is no external connection to this intranet. Each Ethernet segment has a number of hosts directly attached to it. The Ethernet segments are shared media, there is no Ethernet switching equipment.*

[4] to be done

Reference String	3	2	1	0	3	2	4	3	2	1	0	4	2	3	**2**	1	0	4
Cache	3	2	1	0	3	2	4	3	2	1	0	4	2	3	**3**	1	0	4
		3	2	1	0	3	2	4	3	2	1	0	4	2	**2**	3	1	0

Reference String	3	2	1	0	3	2	4	**3**	**2**	1	0	**4**	2	3	**2**	1	0	4
Cache	3	2	1	0	3	2	4	**4**	**4**	1	0	**0**	2	3	**3**	1	0	4
		3	2	1	0	3	2	**2**	**2**	4	1	**1**	0	2	**2**	3	1	0
			3	2	1	0	3	**3**	**3**	2	4	**4**	1	0	**0**	2	3	1

Reference String	3	2	1	0	**3**	**2**	4	3	2	1	0	4	**2**	3	2	1	0	4
Cache	3	2	1	0	**0**	**0**	4	3	2	1	0	4	**4**	3	2	1	0	4
		3	2	1	**1**	**1**	0	4	3	2	1	0	**0**	4	3	2	1	0
			3	2	**2**	**2**	1	0	4	3	2	1	**1**	0	4	3	2	1
				3	**3**	**3**	2	1	0	4	3	2	**2**	1	0	4	3	2

Table 5.1: Belady's anomaly. A Cache with First In, First Out replacement policy. Top: first line: list of references to objects labeled 1 to 4. References not in bold face represent cache misses. Following lines: content of the cache. The cache can hold 2 entries. Middle, Bottom: same, but cache can hold 3 [resp. 4] entries. The number of cache misses is worst (15) with the large cache than with the middle one (14).

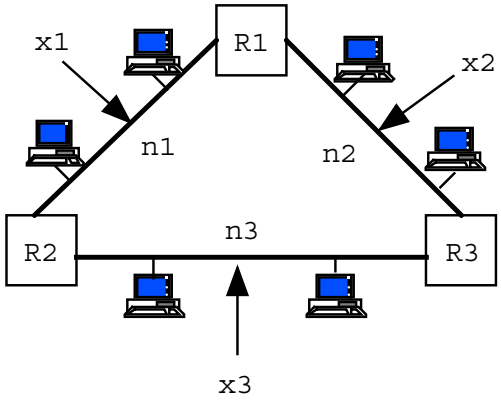


Figure 5.7: The network for Exercise 4.5

We assume that the IP routing tables in R1, R2 and R3 are setup in such a way that traffic from subnet n_i to a subnet n_j, with $i \neq j$ goes through exactly one router.

We call x_i the total traffic generated by all hosts directly attached to segment i. We neglect the effect of collisions on one Ethernet and thus assume that the maximum amount of traffic possible on every Ethernet segment is 10 Mb/s. We further assume that the destination of traffic originating from subnet i is uniformly distributed among the three subnets. Thus, for example, the amount of traffic originating from subnet 1 which has a destination in subnet 2 is $\frac{x_1}{3}$.

1. *What is the maximum value of the total traffic $x_1 + x_2 + x_3$ which is possible with these assumptions ?*
2. *We assume that there are u_i flows per segment, each with rate λ_i, $i = 1, 2, 3$. Thus $x_i = u_i\lambda_i$. If we apply max-min fairness per flow, what is the value of λ_i for the two following cases: (i) $u_i = u$ for $i = 1, 2, 3$, and (ii) $u_1 = 4$, $u_2 = 3$ and $u_3 = 2$? What is then the maximum throughput ?*
3. *Same question if we apply proportional fairness.*

EXERCISE 5.2. *Is max-min fairness equivalent to maximizing $F(\vec{x}) = \min_s(x_s)$? (Examine separately each of the tow sides of the equivalence).*

EXERCISE 5.3. *Consider n sources. The rate x_i of source i is constrained by $x_i \leq r_i$, for some fixed numbers r_i, $1 \leq n$. In addition, we require that $\sum_{i=1}^{n} x_i \leq C$ for some fixed C. With these constraints, are the max-min fair and proportionally fair rate allocations the same ?*

EXERCISE 5.4. *Consider the example in Figure 5.1.*

1. *Give an explicit formula for the throughput as a function of the parameter λ.*
2. *In the general case for this sample example, what is the maximum throughput available ? Does it correspond to an equal allocation of resources ?*

EXERCISE 5.5. *Read [Kelly01-mmi] Sections 1–4, then answer the following questions.*

1. *What is a Pareto efficient rate allocation ?*
2. *What is a Wardrop stable point (also called equilibrium) in this paper ?*
3. *For the single path routing and TCP flows, is the Wardrop equilibrium Pareto efficient ?*
4. *For the multiple path routing and TCP flows, is the Wardrop equilibrium Pareto efficient ?*

CHAPTER 6

QUEUING THEORY FOR THOSE WHO CANNOT WAIT

Queuing phenomena are very frequent in computer and communication systems, and explain a large number of performance patterns. We focus here on fundamental queuing aspects, leaving out the analytical solution of particular queuing systems; the interested reader should consult [Thiran02-LN], [Nain98-Umass] or [Kleinrock76-book] for a classical treatment of queuing systems.

Contents

6.1 DESCRIPTION OF A QUEUING SYSTEM WITH CUMULATIVE FUNCTIONS

6.1.1 CUMULATIVE FUNCTIONS

Consider a system which is viewed as a black box. It may be a network node, an information system... We use the following definitions and assumptions.

- $A(t)$ *input function* is the amount of work that arrives into the system in the time interval $[0, t]$
- $D(t)$ *output function* is the amount of work done in the time interval $[0, t]$
- $Q(t) := A(t) - D(t)$ is the backlog (unfinished work) at time t.
- Assume that there is some time $t_0 \leq 0$ at which $A(t_0) = D(t_0) = 0$. We interpret t_0 as an instant at which the system is empty.
- Let $Q(t) := A(t) - D(t)$; we interpret $Q(t)$ as the backlog (unfinished work) at time t.
- There is no loss of work.

Also define

$$d(t) = \min \{u \geq 0 : \ A(t) \leq (D(t + u)\}$$

The FIFO assumption means that $d(t)$ is the response time for a hypothetical atom of work that would arrive at time t. See Figure 6.1.

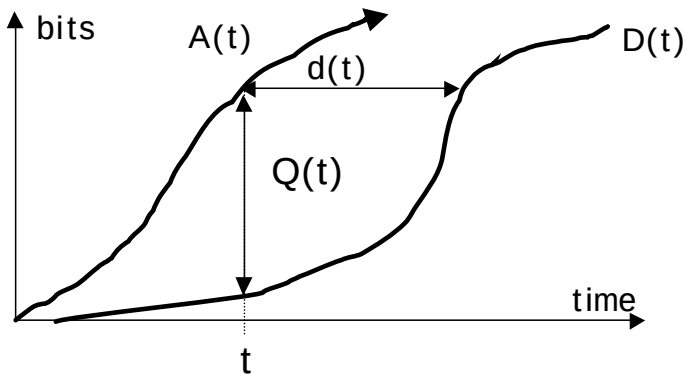


Figure 6.1: Use of cumulative functions to describe a queuing system.

EXAMPLE 6.1: PLAYOUT BUFFER. Consider a packet switched network that carries bits of information from a source with a constant bit rate r (Figure 6.2) as is the case for example, with circuit emulation. We have a first system $\mathcal{S}$, the network, with input function $A(t) = rt$. The network imposes some variable delay, because of queuing points, therefore the output $A'()$ does not have a constant rate r. What can be done to re-create a constant bit stream ? A standard mechanism is to smooth the delay

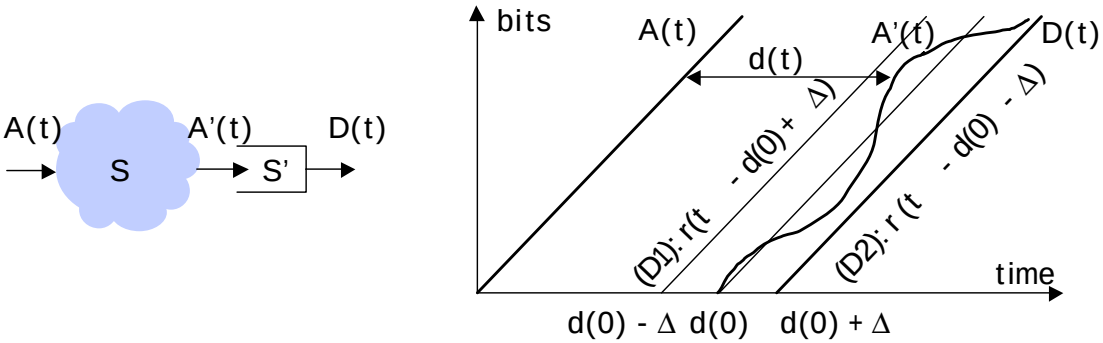


Figure 6.2: A Simple Playout Buffer Example

variation in a playout buffer. It operates as follows. When the first bit of data arrives, at time $d(0)$, it is stored in the buffer until some initial delay has elapsed. Then the buffer is served at a constant rate r whenever it is not empty. This gives us a second system $\mathcal{S}'$, with input $A'()$ and output $D()$. What initial delay should we take ? We give an intuitive, graphical solution. For a formal development, see see [LeBoudecThiran02-book], Section 1.1.1.

The second part of Figure 6.2 shows that if the variable part of the network delay (called *delay jitter*) is bounded by some number Δ, then the output $A'(t)$ is bounded by the two lines (D1) and (D2). Let us the output $D(t)$ of the playout buffer to the function represented by (D2), namely $D(t) = rt - d(0) - \Delta$. This means that we read data from the playout buffer at a constant rate r, starting at time $d(0) + \Delta$. The fact that $A'(t)$ lies above (D2) means that there is never underflow. Thus the playout buffer should delay the first bit of data by an amount equal to a bound on delay jitter.

QUESTION 6.1.1. *What is the required playout buffer size ?* [1]

6.1.2 SINGLE SERVER QUEUE

Consider a lossless, FIFO, system, with the same assumptions as in Section 6.1.1, and assume further that it is a single server queue. Formally, this means the following.

- Let $\beta(s)$ is the service capacity during an interval of duration s where there is some work to do. For example:

[1] A bound on buffer size is the vertical distance between $A(t)$ and $A'(t)$; from Figure 6.2, we see that it is equal to $2r\Delta$.

  - constant rate server: $\beta(t) = ct$ where c is some constant
  - server with latency: $\beta(t) = c(t - t_0)^+$: this server make take some time $\leq t_0$ to wake up when some new jobs arrive

- Define $s(t)$ as the largest time $\leq t$ where the system is empty, i.e. we have either
  - $Q(t) = 0$ and then $s(t) = t$
  - $Q(s(t)) = 0$ and $Q(u) > 0$ for $u = s(t+1), ..., t$

  $s(t) + 1$ is beginning of the busy period at t. By definition, the single server queue is characterized by

$$Q(t) = A(t) - A(s(t)) - \beta(t - s(t))$$

THEOREM 6.1.1 (*Reich*). *For the single server, infinite buffer queue defined above:*

$$Q(t) = \max_{s \leq t} (A(t(-A(s) - \beta(t - s))$$

Proof. tbd

□

6.1.3 APPLICATION TO SCALING OF INTERNET DELAY

We are interested in knowing whether queuing delays are going to disappear when the Internet grows to broadband. The following analysis is due to Norros [Norros94-QS] and Kelly [Kelly99-smi].

Assume traffic on an internet link grows according to three scale parameters: volume (v), speedup (s) and number of users (u). This is captured by the relation:

$$A(t) = v \sum_{i=1}^{u} A_i(st) \tag{6.1}$$

We are interested in the delay; assuming the link is a constant rate server with rate c, this is the backlog divided by c. We also assume that the capacity of the link is scaled with the increase in volume: $c = c_0 vsu$. The question is now: how does the delay depend on v, s, u ?

The maximum delay, $D(v, s, u)$ is derived from Reich's formula:

$$D(v, s, u) = \max_{t \geq 0} \left(\frac{A(t)}{c} - t \right)$$

The dependence on v and s is simple to analyse. It comes

$$D(v, s, 1) = \max_{t \geq 0} \left(\frac{vA_1(st)}{c} - t \right) = \max_{t \geq 0} \left(\frac{A_1(t)}{c_0 s} - \frac{t}{s} \right) = \frac{1}{s} D(1, 1, 1)$$

and similarly for $u \neq 1$ we have $D(v, s, u) = \frac{1}{s} D(1, 1, u)$. Thus the delay is independent of volume scaling, and is inversely proportional to the speedup factor s. The dependence on u requires more

assumptions. To go further, we assume a stochastic model, such that the queue length process $Q(t)$ is stationary ergodic. We can use Reich's formula:

$$Q(0) = \max_{t \geq 0} (A(-t) - ct)$$

where $A(-t)$ is now the amount of work that has arrived in the interval $[-t, 0]$. We assume that Equation (6.1) continues to hold. Further, we model $A_i(-t)$ by a *fractional brownian traffic* [Norros94-QS]. This is a simplified model which captures long range dependence. This means that

$$A_i(-t) = \lambda t + \sqrt{\lambda a} B_H^i(t)$$

where B_H^i is fractional brownian motion, λ the traffic intensity, and a a variance parameter. Fractional brownian motion is a gaussian process, with mean λt and variance $\lambda a t^{2H}$. Remember that $B_H(t)$ is self-similar in the sense that the process $B_H(kt)$ has the same distribution as $k^H B_H(t)$.

Assume that the A_is are independent. It follows from the properties of fractional brownian motion that $A(-t)$ is also fractional brownian traffic. Its mean is $u\lambda$ and its variance is $u\lambda a t^{2H}$, thus it has intensity $u\lambda$ and same variance parameter a.

By Reich's formula

$$D(1,1,u) = \max_{t \geq 0} \left(\frac{A(t)}{c_o u} - t \right) = \max_{t \geq 0} \left[\left(\frac{\lambda}{c_0} - 1 \right) t + \sqrt{\lambda a} B_H(t) \frac{1}{c_0 \sqrt{u}} \right]$$

Do the change of variable $t = k\tau$. It comes

$$D(1,1,u) \sim \max_{\tau \geq 0} \left[\left(\frac{\lambda}{c_0} - 1 \right) k\tau + \sqrt{\lambda a} k^H B_H(\tau) \frac{1}{c_0 \sqrt{u}} \right]$$

where $\sim$ means same distribution. Take k such that $k = \frac{k^H}{\sqrt{u}}$, i.e. $k = u^{-\frac{1}{2(1-H)}}$. Then we have

$$D(1,1,u) \sim u^{-\frac{1}{2(1-H)}} D(1,1,1)$$

In summary, the delay scales according to

$$D(v,s,u) = \frac{1}{s u^b} D(1,1,1)$$

with $b = \frac{1}{2-2H}$. In practice, we expect the Hurst parameter usually lies in the range $[0.67, 0.83]$ thus $1.5 \leq b \leq 3$. In summary, delay decreases with speedup more rapidly with the number of users.

6.2 CLASSICAL RESULTS FOR A SINGLE QUEUE

The single queue has received much attention, and there are analytical results available for a large class of systems with random arrivals and service.

6.2.1 OTHER REPRESENTATION OF A SINGLE SERVER QUEUE

There are other representations than cumulative functions, which are more adapted if we are interested not only in the workload but also in other state information. For example, consider a computer that receives tasks to process, with task n arriving at time a_n, having a processing requirement s_n, and departing at time d_n. The representation with cumulative functions can be used, by defining $A(t) = \sum_{n:\, a_n \leq t} s_n$ and $D(t) = \sum_{n:\, d_n \leq t} s_n$, but it does not directly give information about the number of tasks in the systems.

A general definition of a single server queue is by means of the sequences a_n, d_n, s_n. The system is a FIFO single server queue if it satisfies

$$d_n = \max(a_n, d_{n-1}) + s_n \tag{6.2}$$

Classical queuing theory for the FIFO single server queue is interested in Equation (6.2) where the a_n, w_n is stochastic. a_n and d_n are interpreted as arrival and departure times of "customers". In the rest of this section we replace a_n, d_n with A_n, D_n to emphasize that they are random.

6.2.2 KENDALL'S NOTATION

The classical notation for a queue, in its simplest form, is of the type $A/S/s/K$ where:

- A (character string) describes the type of arrival process: G stands for the most general arrival process, $A =$GI means that the arrival process is a point process with iid interarrival times, M is for a Poisson arrival process.
- S (character string) describes the type of service process: G for the most general service process, $S =$GI means that the service times are iid and independent of the arrival process, $S =$M is the special case of GI with exponential service times, $S =$D with constant service times.
- s and K are integers representing the number of servers and the capacity (maximum number of customers allowed in the system, queued + in service). When $K = \infty$, it may be omitted.
- The marked point process A_n, S_n is stationary.
- The service discipline is by default FIFO, otherwise it is mentioned explicitly.

6.2.3 SUMMARY OF SOME CLASSICAL RESULTS FOR THE SINGLE SERVER QUEUE

We focus now on the case $s = 1$. Quantities of interest are

- the arrival rate λ the intensity of the arrival process a_n (mean number of customer arrivals per second, also equal to the inverse of the mean interarrival time (Chapter 11)
- $\rho = \lambda \bar{S}$ (server utilization) where $\bar{S}$ is the mean service time (Palm expectation of S_n).
- the residence time $R_n = D_n - A_n$ and waiting time $W_n = R_n - S_n$ for customer n
- the number of customers in the system $N(t)$, the number of customers waiting $N_w(t)$, given by

$$N(t) = \sum_{n \in \mathbb{Z}} 1_{\{A_n \leq t\}} 1_{\{D_n > t\}}$$

$$N_w(t) = (N(t) - 1)^+$$

STABILITY An important issue in the analysis of the single server queue is stability. In mathematical terms, it means whether $N(t)$ is stationary. When the system is unstable, a typical behaviour is that the backlog grows to infinity.

THEOREM 6.2.1 (Loynes). *The single server queue is unstable for $\rho > 1$ and stable for $\rho < 1$.*

The first part says that a necessary condition for stability is $\rho \leq 1$. We give a heuristic explanation for the necessary condition is as follows. If the system is stable, all customers eventually enter service, thus the mean number of beginnings of service per second is λ. From Little's law applied to the server (see Section 6.3), we have $\rho =$ the probability that the server is busy, which is ≤ 1. The proof of the second statement is more complex – see [Baccelli88-book] for details. For $\rho = 1$ there may or may not be stability, depending on the specific queue.

Be careful that this intuitive stability result holds only for a single queue. For networks of interconnected queues, there is no such general result.

For the finite capacity queue, stability is usually for any value of ρ.

QUESTION 6.2.1. *Consider a queuing system of the form G/G/1 where the service time w_n of customer n is equal to the inter-arrival time $a_{n+1} - a_n$. What are the values of $\rho, \bar{N}$?* [2]

QUESTION 6.2.2. *Give an example of stable single server queue with $\rho = 1$.* [3]

Classical quantitative results for simple, but useful, queues are given below. The notation is explained at the end of this chapter.

M/GI/1 QUEUE Stability is for $\rho < 1$

$$\begin{cases} \bar{N} = \frac{\rho^2\kappa}{1-\rho} + \rho \text{ with } \kappa = \frac{1}{2}\left(1 + \frac{\sigma_{\bar{S}}^2}{\bar{S}^2}\right) \\ \bar{N}_w = \frac{\rho^2\kappa}{1-\rho} \\ \bar{R} = \frac{\bar{S}(1-\rho(1-\kappa))}{1-\rho} \\ \bar{W} = \frac{\rho\bar{S}\kappa}{1-\rho} \end{cases}$$

Stability is for $\rho < 1$ for all the examples below.

M/M/1 QUEUE Stability is for $\rho < 1$.

$$\begin{cases} \bar{N} = \frac{\rho}{1-\rho} \\ \bar{N}_w = \frac{\rho^2}{1-\rho} \\ \bar{R} = \frac{\bar{S}}{1-\rho} \\ \bar{W} = \frac{\rho\bar{S}}{1-\rho} \\ \sigma_N = \frac{\sqrt{\rho}}{1-\rho} \\ \sigma_R = \frac{\bar{S}}{1-\rho} \\ \mathbb{P}(N = k) = (1-\rho)\rho^k \\ \mathbb{P}^0(R \leq x) = 1 - e^{-(1-\rho)\frac{x}{\bar{S}}} \end{cases}$$

[2] $\lambda = \frac{1}{\bar{S}}$ thus $\rho = 1$. There is always exactly one customer in the queue. Thus $\bar{N} = 1$.

[3] The example in Question 6.2.1.

M/M/1/K QUEUE Stability is for any ρ.

$$\begin{cases} \mathbb{P}(N=k) = \eta(1-\rho)\rho^k 1_{\{0\le k\le K\}} \\ \eta^{=}\frac{1}{1-\rho^{K+1}} \\ \mathbb{P}^0(\text{ arriving customer is discarded }) = \mathbb{P}(N=K) \end{cases}$$

M/D/1 QUEUE Stability is for $\rho < 1$.

$$\begin{cases} \bar{N} = \frac{\rho^2}{2(1-\rho)} + \rho \\ \bar{N}_w = \frac{\rho^2}{2(1-\rho)} \\ \bar{R} = \frac{\bar{S}(2-\rho)}{2(1-\rho)} \\ \bar{W} = \frac{\rho\bar{S}}{2(1-\rho)} \\ \sigma_N = \frac{1}{1-\rho}\sqrt{\rho - 1.5\rho^2 + \frac{5}{6}\rho^3 - \frac{1}{12}\rho^4} \\ \sigma_R = \frac{\bar{S}}{1-\rho}\sqrt{\frac{1}{3}\rho - \frac{1}{12}\rho^2} \end{cases}$$

QUESTION 6.2.3. *Which of the quantities $\bar{N}, \bar{N}_w, \bar{R}, \bar{W}$ are Palm expectations ?* [4]

6.2.4 CLASSICAL RESULTS FOR MULTIPLE SERVER QUEUES

The multiple server queue is defined by the fact that at most s customers can be served in parallel. The utilization ρ is now defined by $\rho = \frac{\lambda\bar{S}}{s}$.

THEOREM 6.2.2 (Loynes). *The multiple server queue is unstable for $\rho > 1$ and stable for $\rho < 1$.*

M/M/S QUEUE Stability is for $\rho < 1$. Let

$$u = \frac{\sum_{i=0}^{s-1}\frac{(s\rho)^i}{i!}}{\sum_{i=0}^{s}\frac{(s\rho)^i}{i!}} \text{ and } p = \frac{1-u}{1-\rho u}$$

$$\begin{cases} \bar{N} = \frac{p\rho}{1-\rho} + s\rho \\ \bar{N}_w = \frac{p\rho}{1-\rho} \\ \bar{R} = \frac{p\bar{S}}{s(1-\rho)} + \bar{S} \\ \bar{W} = \frac{p\bar{S}}{s(1-\rho)} \\ \sigma_R = \frac{\bar{S}}{s(1-\rho)}\sqrt{p(2-p) + s^2(1-\rho)^2} \\ \sigma_W = \frac{1}{1-\rho}\sqrt{p\rho(1+\rho-p\rho)} \\ \mathbb{P}(N=k) = \begin{cases} \eta\frac{(s\rho)^k}{k!} \text{ if } 0\le k\le s \\ \eta\frac{s^s\rho^k}{s!} \text{ if } k > s \end{cases} \\ \eta^{-1} = \sum_{i=0}^{s-1}\frac{(s\rho)^i}{i!} + \frac{(s\rho)^s}{s!(1-\rho)} \\ \mathbb{P}^0(W\le x) = 1 - pe^{-s(1-\rho)\frac{x}{\bar{S}}} \\ \mathbb{P}(\text{all servers busy}) = \mathbb{P}(N\ge s) = p \text{ (Erlang-C formula)} \end{cases}$$

[4] $\bar{R}, \bar{W}$

M/M/S/S QUEUE (*Erlang Loss Formula*) Stability is for any ρ.

$$\begin{cases} \mathbb{P}(N=k) = \eta 1_{\{0\le k\le s\}} \frac{(s\rho)^k}{k!} \\ \mathbb{P}^0(\text{ arriving customer is discarded }) = \mathbb{P}(N=s) \text{ } \textit{Erlang-B} \text{ formula} \\ \eta^{-1} \text{ such that } \sum_{i=0}^{s} \mathbb{P}(N=i) = 1 \end{cases}$$

6.2.5 PROCESSOR SHARING

to be done, with application to TCP.

6.2.6 OTHER RESULTS

There is a huge literature on queuing systems, most of which is concerned with finding analytical expressions for specific systems. It is worth mentioning that numerical solutions of the representation of the queue is sometimes possible, thus avoiding the need for an analytical expression.

DIRECT SOLUTION Consider for example the GI/GI/1 queue, for which no explicit solution exists. The following equation can be used to obtain a numerical solution.

$$Q_n = (Q_{n-1} + S_{n-1} - A_n + A_{n-1})^+$$

where $Q_n = N(A_n^-)$ is the number of customers in the system just before the nth arrival. The knowledge of Q_n can be used to derive or approximate many other quantities. Let $U_n = Q_{n-1} + S_{n-1} - A_n + A_{n-1}$. We have $\mathbb{P}^0(Q_n = k) = \mathbb{P}^0(U_n = k)$ if $k > 0$ and $\mathbb{P}^0(Q_n = 0) = \mathbb{P}(U^n \le 0)$. Assume $\rho > 1$ and the system is stationary. Let $q_k := \mathbb{P}^0(Q_n = k), u_k := \mathbb{P}^0(U_n = k) s_k := \mathbb{P}^0(S_n = k), a_k = \mathbb{P}^0(A_n - A_{n-1} = -k)$. $Q_{n-1}, (A_n - A_{n-1})$ and S_{n-1} are mutually independent (because we consider a GI/GI/1 queue) thus u is the convolution $u = q * s * a$. Thus the array q satisfies the fixed point equation

$$\begin{cases} q_k = (q * s * a)_k \text{ if } k > 0 \\ q_0 = \sum_{i\le 0} (q * s * a)_i \end{cases}$$

This equation can be solved numerically by iteration. The convolution can be computed using the fast Fourier transform. See [Grossglauser96-Sigcomm] for an example where this method is used. There is also a large literature on advanced, analytical methods for solving the fixed point equation.

MARKOVIANISATION Consider again the GI/GI/1 queue. The distributions of the inter-arrival and service times can be approximated by PH-type distributions (Section 11.8). There exist numerical, efficient solutions for the stationary probability of the PH/PH/1 queue [LeBoudec88-Questa].

6.2.7 NON-LINEARITY OF RESPONSE TIME

The response time, queue occupancy, and for finite capacity queues, the loss probability, grow dramatically as ρ comes close to 1; see Figure 6.3 for an example. This strong non-linearity is important in practice.

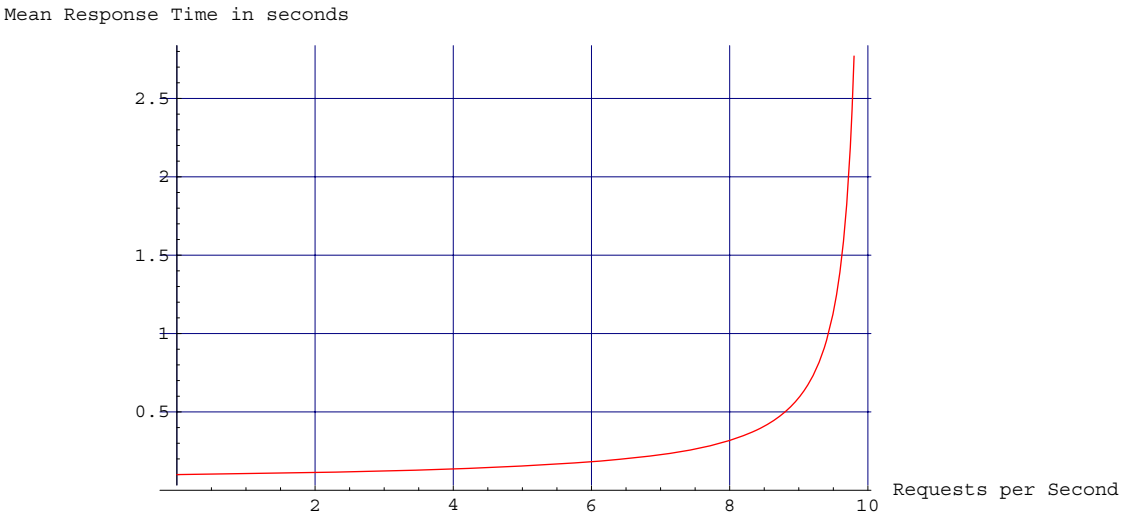


Figure 6.3: Average response time versus requests per second for a database server modeled as M/GI/1 queue

EXAMPLE 6.2: A database system services requests that can be modeled as a Poisson process. The time needed to process a request is 0.1 second and its standard deviation is estimated to 0.03. How does the average response time depend on the number of requests per second that can be served ? The solution is found by the M/GI/1 queue model and is plotted in Figure 6.3.

QUESTION 6.2.4. *What is the maximum load that can be served if an average response time of* 0.5 *second is considered acceptable ?* [5]

QUESTION 6.2.5. *What happens if this load is exceeded by 10% ? by 20% ?* [6]

Figure 6.4 shows how the response time of the M/GI/1 queue depends on the coefficient of variation $\frac{\sigma_S}{S}$.

QUESTION 6.2.6. *How do the* $M/D/1$ *and* $M/M/1$ *queue compare to Figure 6.4 ?* [7]

EXAMPLE 6.3: We would like to compare a two-processor, shared memory machine versus a collection of two independent processors, with static load sharing

[5] 8.8 requests per second.

[6] By 10%: the response time becomes 1.75 (thus is multiplied by a factor of 3.5. By 20%: the system becomes unstable $\rho > 1$; in practice it will lose requests, or enter congestion collapse.

[7] The bottom curve ($cv = 0$) is for M/D/1, the middle curve for M/M/1 ($cv = 1$).

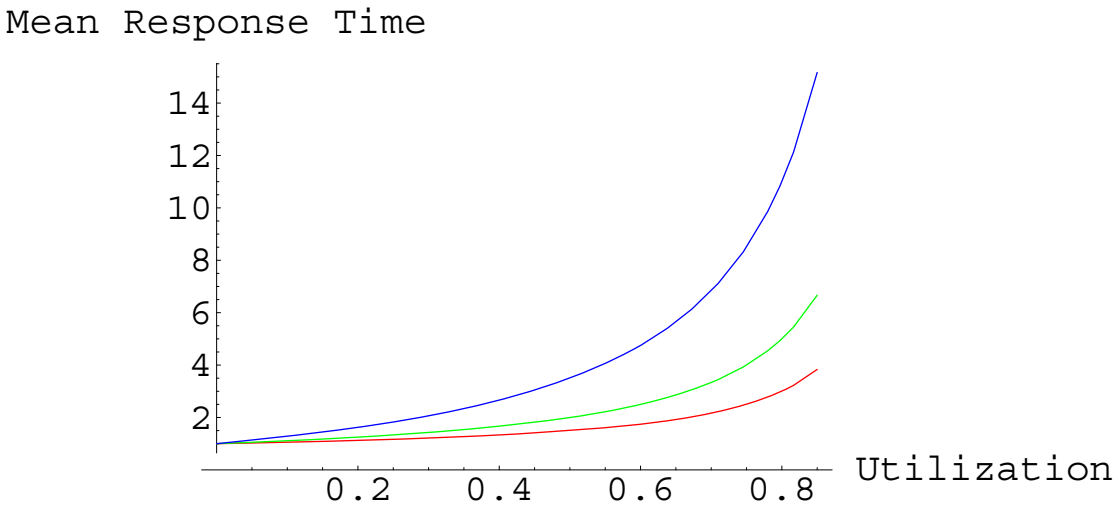


Figure 6.4: Mean response time for M/GI/1 queue, relative to service time, for different values of coefficient of variation $\frac{\sigma_S}{\bar{S}}$: from top to bottom: 2, 1 and 0.

(Figure 6.5). Assume processing times and job inter-arrival times can be modeled as independent iid exponential sequences. Thus the first [resp. second] case is modeled as one M/M/2 queue [resp. a collection of two parallel M/M/1 queues]. Assume load is balanced evenly between the two processors. Both systems have the same utilization ρ. The mean response for the first system is obtained from Section 6.2.4; we obtain $\frac{\bar{S}}{1-\rho^2}$. For the second system it is simply $\frac{\bar{S}}{1-\rho}$ (Figure 6.5).

We see that for very small loads, the systems are similar, as expected. In contrast, for large loads, the response time for the first system is much better, with a ratio equal to $1+\rho$. For example, for $\rho = 0.5$, the second system has a response time 1.5 times larger. However, the capacity is the same for both systems.

6.3 OPERATIONAL LAWS FOR QUEUING SYSTEMS

For systems that are stationary, there is a number of relations that directly derive from Chapter 11. Among them is the celebrated Little law. In this section we give the most common ones. There are many others; they can be derived from Chapter 11, in particular using the ergodic interpretation method explained in Section 11.3.5.

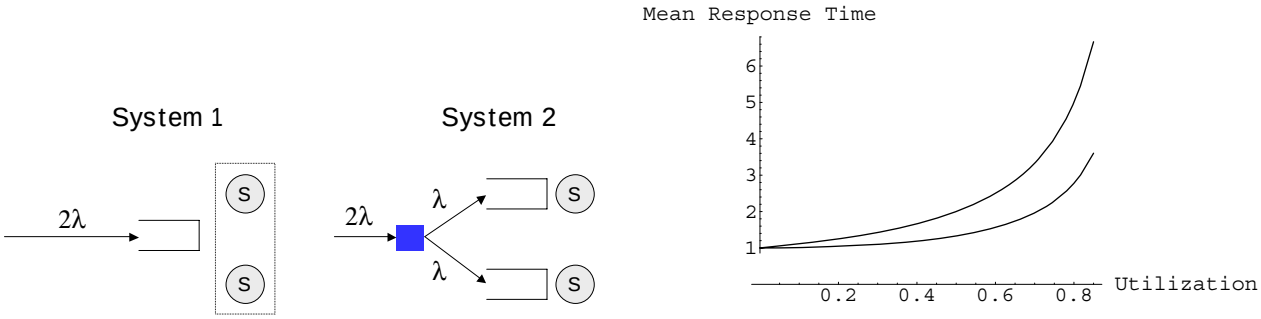


Figure 6.5: Mean response time for systems 1 (bottom) and 2 (top), relative to the service time.

6.3.1 LITTLE'S LAW AND APPLICATIONS

In practice, by the ergodic interpretation, the laws apply to large samples if we can assume that the system is stationary and ergodic. For queuing systems, this usually means that the utilization is less than 1.

THEOREM 6.3.1 (Operational Law). *Consider a stationary system that is visited by a flow of customers. For a formal definition, see Theorem 11.4.2.*

- ***[Throughput]*** *The throughput, defined as the expected number of arrivals per second, is also equal to the inverse of the expected time between arrivals.*
- ***[Little]***

  $$\lambda \bar{R} = \bar{N}$$

  where λ is the expected number of customers arriving per second, $\bar{R}$ is the expected response time seen by an arbitrary customer and $\bar{N}$ is the expected number of customers observed in the system an arbitrary time
- ***[Utilization Law]*** *If the system is a single server queue:*

  $$\mathbb{P}(\textit{server busy}) = \rho := \lambda \bar{S}$$

  If it is an s-server queue,

  $$\mathbb{E}(\textit{number of busy servers}) = s\rho$$

  with $\rho := \frac{\lambda \bar{S}}{s}$.

Proof. The first item is Proposition 11.3.2; the second item is Theorem 11.4.2. The third item is obtained by applying Little's law to the set of servers.

□

QUESTION 6.3.1. *Single server queue: with the notation in Section 6.2.3, show that $\bar{N}_w = \bar{N} - \rho$* [8]

[8] Follows from items 2 and 3 in Theorem 6.3.1.

THE INTERACTIVE USER MODEL The interactive user model is illustrated inFigure 6.6. n users send jobs to a service center. The *think time* is defined as the time between jobs sent by one user. Call $\bar{R}$ the expected response time for an arbitrary job at the service center, $\bar{Z}$ the expected think time and λ the throughout of the system.

THEOREM 6.3.2 (Interactive User).

$$\lambda(\bar{Z} + \bar{R}) = n$$

Proof. Apply Little's law to the entire system.

□

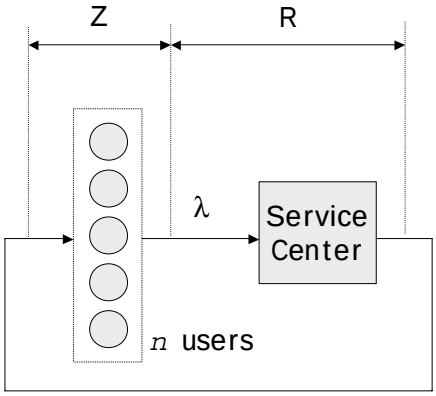


Figure 6.6: The Interactive User Model

EXAMPLE 6.4: SERVICE DESK. A car rental company in a large airport has 10 service attendants. Every attendant prepares transactions on its PC and, once completed, send them to the database server. The software monitor finds the following averages: one transaction every 5 seconds, response time = 2 s.

QUESTION 6.3.2. *What is the average think time ?* [9]

6.3.2 NETWORKS AND FORCED FLOWS

We often find systems that can be modeled as a directed graph, called a network. We consider models of the form illustrated on Figure 6.7. If the total number of customers is constant, the network is called "closed", otherwise "open".

THEOREM 6.3.3 (Network Laws). *Consider a stationary network model*

[9] 48 s

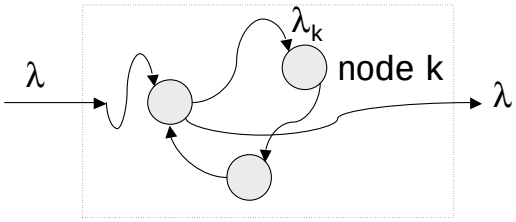


Figure 6.7: Network Model

- ***[Forced Flows]*** *$\lambda_k = \lambda V_k$, where λ_k is the expected number of customers arriving per second at node k and V_k is the expected number of visits to node k by an arbitrary customer during its stay in the network.*
- ***[Total Response Time]*** *Let $\bar{R}$ [resp. $\bar{R}_k$] be the expected total response time $\bar{R}$ seen by an arbitrary customer [resp. by an arbitrary visit to node k].*

$$\bar{R} = \sum_k \bar{R}_k V_k$$

Proof. (Forced Flows). We apply Campbell's formula. Let $F(s,t)$ be the random function which returns 1 if $t \geq s$ and the last customer who arrived before or at $-t$ is in node k at time s, else returns 0. By definition of intensity:

$$\lambda_k = \mathbb{E}\left(\sum_{n\in\mathbb{Z}} F(-A_n, 0)\right)$$

where A_n is the point process of customer arrivals. Campbell's formula applied to $F(-t,0)$ gives:

$$\mathbb{E}(\sum_{n\in\mathbb{Z}} F(-A_n,0) = \lambda \sum_{t\in\mathbb{N}} \mathbb{E}^{-t}(F(t,0)) = \lambda \sum_{t\in\mathbb{N}} \mathbb{E}^0(F(0,t))$$

where the last part is by stationarity. Thus

$$\lambda_k = \lambda \mathbb{E}^0\left(\sum_{t\in\mathbb{N}} F(0,t)\right) = \lambda V_k$$

(Total Response Time) Let $\bar{N}$ [resp. $\bar{N}_k$] be the expected number of customers in the service system [resp. in node k]. We have $\bar{N} = \sum_k \bar{N}_k$. Apply Little' and the Forced Flows laws.

□

EXAMPLE 6.5: Transactions on a database server access the CPU, disk A and disk B (Figure 6.8). The statistics are: $V_{\mathsf{CPU}} = 102, V_{\mathsf{A}} = 30, V_{\mathsf{B}} = 68$ and $\bar{R}_{\mathsf{CPU}} = 0.192\,s,\ \bar{R}_{\mathsf{A}} = 0.101\,s,\ \bar{R}_{\mathsf{B}} = 0.016\,s$

QUESTION 6.3.3. *What is the average response time for a transaction ?* [10]

[10] $23.7\,s$

6.3.3 BOTTLENECK ANALYSIS

Common sense (PE step G5) tells us to analyze bottlenecks first. Beyond this, simple performance bounds in stationary regime can be found by using the following two principles:

1. waiting time is ≥ 0
2. a server utilization is bounded by 1

We illustrate the method on one generic example.

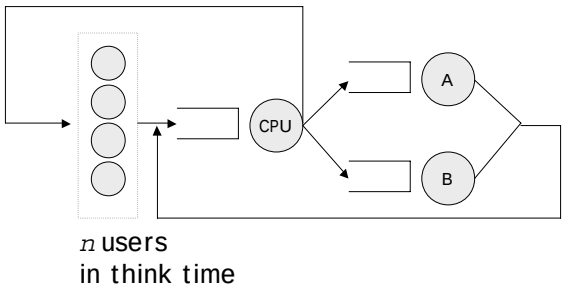


Figure 6.8: Network example used to illustrate bottleneck analysis. n attendants serve customers. Each transaction uses CPU, disk A or disk B. Av. numbers of visits per transaction: $V_{\mathsf{CPU}} = 102, V_{\mathsf{A}} = 30, V_{\mathsf{B}} = 17$; av. service time per transaction: $\bar{S}_{\mathsf{CPU}} = 0.004\,s,\ \bar{S}_{\mathsf{A}} = 0.011\,s,\ \bar{S}_{\mathsf{B}} = 0.013\,s$; think time $Z = 1\,s$.

Consider a queuing network, an example of which is given in Figure 6.8. It is a combination of Figure 6.6 and Figure 6.7. Transactions are issued by a pool of n customers which are either idle (in think time) or using the network. In addition, assume that every network node is a single server queue, and let $\bar{S}_k$ be the average service time per visit at node k. Thus $\bar{R}_k - \bar{S}_k$ is the average waiting time per visit at node k. The throughput λ is given by the interactive user model:

$$\lambda = \frac{n}{Z + \sum_k V_k \bar{R}_k} \tag{6.3}$$

and by forced flows, the utilization of the server at node k is $\rho_k = \lambda V_k \bar{S}_k$. Applying the two principles above gives the constraints on λ:

$$\begin{cases} \lambda \leq \frac{n}{\bar{Z} + \sum_k V_k \bar{S}_k} \\ \lambda \leq \frac{1}{\max_k V_k \bar{S}_k} \end{cases} \tag{6.4}$$

Similarly, using Equation (6.3) and Equation (6.4), we find the following constraints on the response time $\bar{R} = \sum_k V_k \bar{R}_k$:

$$\begin{cases} \bar{R} \geq \sum_k V_k \bar{S}_k \\ \bar{R} \geq n \left(\max_k V_k \bar{S}_k \right) - \bar{Z} \end{cases} \tag{6.5}$$

Figure 6.9 illustrates the bounds.

QUESTION 6.3.4. *Draw the response time bounds for this example.* [11]

QUESTION 6.3.5. *Which of the bounds is accurate for low load ? For high load ?* [12]

[11] tbd

[12] For low load, the former bound in Equation (6.4) is accurate because queuing times are small. For high loads, we do not know. If the system suffers from congestion collapse, the bounds may be very optimistic. In contrast, for an ideal system, the throughput is driven by its bottleneck and the latter bound may be accurate.

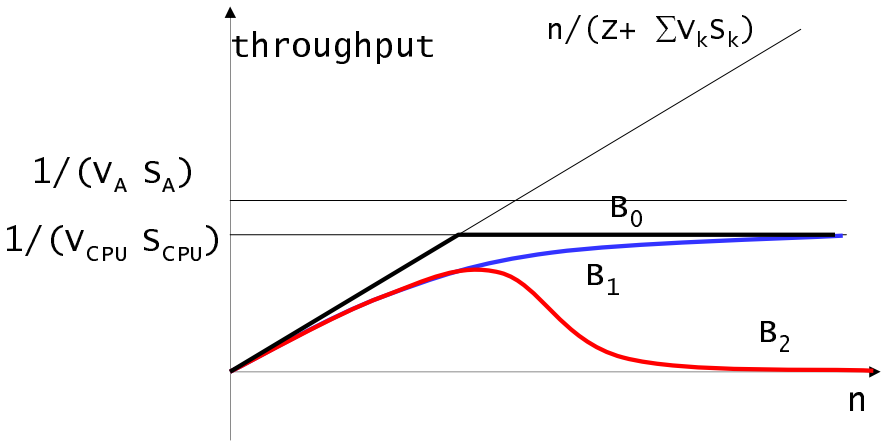


Figure 6.9: Throughput bound (B0) obtained by bottleneck analysis for the system in Figure 6.8, as a function of the number of users n. B1, B2: typical throughput values for a system without [resp. with] congestion collapse.

BOTTLENECK A node k that maximizes $V_k \bar{S}_k$ is called, in this model, a *bottleneck*. To see why a bottleneck determines the performance, consider improving the system by decreasing the value of $V_k \bar{S}_k$ (by reducing the number of times the resource is used, or by replacing the resource by a faster one). If k is not a bottleneck, this does not affect asymptote on Figure 6.9, and only marginally increases the slope of the bound at the origin, unlike if k is a bottleneck. 11

QUESTION 6.3.6. *What is the bottleneck on the example of Figure 6.8 ?* [13]

QUESTION 6.3.7. *What happens to the example of Figure 6.8 if the CPU processing time is reduced from* 0.004 *to* 0.003 *? to* 0.002 *?* [14]

6.4 PRIORITIES

Kleinrock's Conservation law (derived from Campbell). μc rule. Daigle's queuing models. Priority queuing and Daigle. Instability results.

6.5 CASE STUDY

Consider the question asked by the picture on the cover: "double the throughput, divide the response time by 2 ". Does this statement hold ?

First we apply the principles in Chapter 1.

- Goal: evaluate impact of doubling the capacity of a skilift on the response time.
- Factors: c = capacity of skilift in people per second.

[13] The CPU.

[14] The disk A becomes the bottleneck. Decreasing the CPU processing time to 0.002 does not improve the bound significantly.

- Metrics: response time. A more detailed reflection leads to considering the waiting time, as this is the one that affects customer's perception. We are interested in average and peak values.
- Load: we consider two load models : (1) heavy burst of arrival (after a train or a bus arrives to the skilift) (2) peak hour stationary regime

6.5.1 QUEUING MODEL

We can model the skilift as the queuing system illustrated in Figure 6.10. The first queue models the gate; it is a single server queue. Its service time is the time between two passages through the gate, when there is no idle period and is equal to $1/c$. The second queue represents the transportation time. It is an infinite server queue, with no waiting time. Since our performance metric is the waiting time, we ignore the second queue in the rest of the analysis.

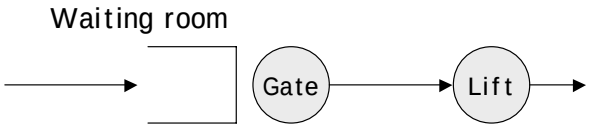


Figure 6.10: Queuing Model of Skilift

6.5.2 TRANSIENT ANALYSIS

Assume the arrival of skiers is one single burst (all arrive at the same time). Also assume that all skiers use the same time to go through the gate, which is roughly true in this scenario. The model in Section 6.1.1 applies, with $A(t) =$ the number of skiers arriving in $[0, t]$ and $D(t) =$ the number of skiers that entered the skilift in $[0, t]$. Thus the delay $d(t)$ is the waiting time, excluding the time spent on the skilift. We also have $\beta(t) = ct$, with $c =$ the capacity of the skilift, in skiers per second. We have $A(t) = B$ for $t \geq 0$. Figure 6.11 shows that doubling the capacity does divide the worst case waiting time by two.

QUESTION 6.5.1. *Is the average waiting time also divided by 2 ?* [15]

QUESTION 6.5.2. *Assume the arrival of skiers is bursty, but not as sudden. For example, we take* $A(t) = kct$ *for* $0 \leq t \leq t_0$ *and* $A(t) = A(t_0)$ *for* $t \geq t_0$*, with* $k \geq 1$*. What is now the conclusion ?* [16]

6.5.3 STATIONARY ANALYSIS

Assume now we are observing the system in the middle of the peak hour. We can model the gate as a G/D/1 queue. It is difficult to give a more accurate statement about the arrival process without

[15] Yes, the waiting time seen by an average customer arriving as number y ($0 \leq y \leq B$) is linear in y, thus is equal to the worst case response time divided by a 2.

[16] The response time is reduced by a factor higher than 2 (draw a picture).

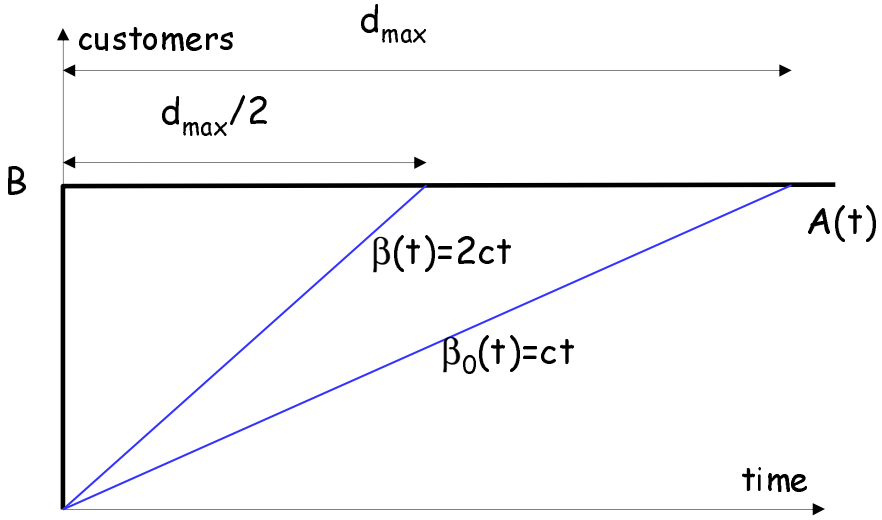


Figure 6.11: Transient Analysis: A burst of skiers arrives at time 0. Impact of doubling the capacity of the skilift.

performing actual measurements. If a Poisson model is acceptable (many independent arrivals of skiers from various slopes) then the M/G/1 results apply and the average response time is given in Figure 6.3. The queuing time is the value on the curve minus the offset at 0, and the utilization ρ (x-value of Figure 6.3) is $\frac{\lambda}{c}$.

Doubling the capacity means that the utilization factor is halved, assuming this has no effect on the arrival rate. The effect on the response time depends on where we stood on the curve. If the system was close to saturation, the effect is a large reduction of the average waiting time. The effect on the peak waiting time (here: 0.95-quantile) requires more sophisticated formulae (see [Cost224-book]) but is similar.

It is probably unrealistic to assume that a reduction in waiting time has no effect on the arrival rate. A better, though simplified, model is illustrated in Figure 6.12. It is a variant of the interactive user model in Figure 6.6. Here we assume that the mean number $\bar{N}$ of skiers in the system is independent of c. We apply bottleneck analysis. Let T be the throughput of the skilift and $\bar{Z}$ the time spent on the lift or on the slope. We have

$$\begin{cases} T \leq \frac{\bar{N}}{\frac{1}{c}+\bar{Z}} \\ T \leq c \end{cases}$$

and thus

$$\bar{W} \geq \max\left(\frac{\bar{N}-1}{c} - \bar{Z}, 0\right)$$

Figure 6.12 shows the bound as a function of $\frac{1}{c}$ for sake of comparison with Figure 6.3. A few points obtained by simulation are also plotted. This strongly suggests that the function f that maps $\frac{1}{c}$ to the average response time is convex; the graph of a convex function is below its chords, thus

$$f(\frac{1}{2c}) < \frac{1}{2}f(\frac{1}{c})$$

and reducing the capacity **does reduce the waiting time by at least a factor 2**.

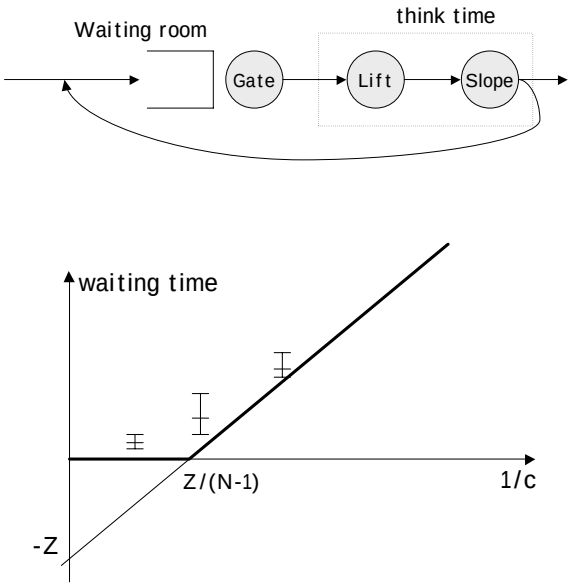


Figure 6.12: First: A Model that accounts for dependency of arrival rate and waiting time. Second: Waiting time for this model in Figure 6.12 as a function of $\frac{1}{c}$, where c is skilift capacity. Thick line: bound predicted by bottleneck analysis. A few simulation results are shown with 95% confidence interval.

We also see that a key value is $c^* = \frac{\bar{N}-1}{\bar{Z}}$. If c is much larger than c^*, the waiting time is small, so doubling the capacity has little effect anyhow. For c much smaller than c^*, the waiting time increases at an almost constant rate. Thus we should target c of the order of c^*. For a highly congested system ($2c$ much smaller than c^*) the offset at 0 becomes negligible and the response time is almost linear in $1/c$, thus doubling the capacity does reduce the waiting time by 2, roughly speaking – but the system is still congested after doubling the capacity.

QUESTION 6.5.3. *For which values of c should the bound be accurate ?* [17]

[17] For small c and for large c.

6.6 SUMMARY OF NOTATION

Notation	Definition
A/S/s/K	Kendall notation: arrival process/service process/ number of servers/ capacity of queue including customers in service
λ	arrival rate
s	number of servers
$\bar{S}, \sigma_S$	mean and standard deviation of service time
$\rho = \frac{\lambda \bar{S}}{s}$	server utilization
$N, \bar{N}, \sigma_N$	number of customers in system, its mean and standard deviation
$N_w, \bar{N}_w, \sigma_{N_w}$	number of customers waiting, its mean and standard deviation
$R, \bar{R}, \sigma_R$	time spent in system (residence time), its mean and standard deviation
$W, \bar{W}, \sigma_W$	waiting time, its mean and standard deviation
V_k	mean number of visits per customer to node k
$\bar{Z}$	av. think time in interactive user model

6.7 EXERCISES

EXERCISE 6.1. *Consider the Surge model with one UE. Assume the average inactive off period id Z, the average active off period is Z', the average number of URLs requested per active period is V, and the average response time for a URL request is R: What is the throughput of λ of one UE ?*

EXERCISE 6.2. *Consider again Question 9.8.20. How do you interpret the fact that the response time varies linearly with the number of processes active in the system ?*

EXERCISE 6.3. *Read [Tan02-Sigmetrics] and answer the following questions.*

1. *is the goal of the evaluation well defined ? What is it ?*
2. *are the factors identified ? What are they ?*
3. *what performance indices are chosen ?*
4. *how is the workload generated ?*
5. *are there implicit assumptions that should have been formulated ?*
6. *are the experiments or results reproducible ?*
7. *what conclusions can be drawn from the study ?*
8. *is the approach scientific ? do you believe the conclusions ? why ?*
9. *what techniques are used for the evaluation ?*
10. *is the level of sophistication adequate ?*
11. *was a performance analysis justified (aren't the results obvious or too dependent on input factors, which are arbitrary) ?*
12. *is there any part that can be removed ?*
13. *are the graphics OK ?*
14. *what aspects of the evaluation do you like or dislike ?*

PART II

SELECTED TOPICS IN PERFORMANCE EVALUATION

CHAPTER 7

TESTS

"No test can prove me right, a single test can prove me wrong".[1]

Contents

[1] Free adaptation of a sentence attributed to Albert Einstein

Contents

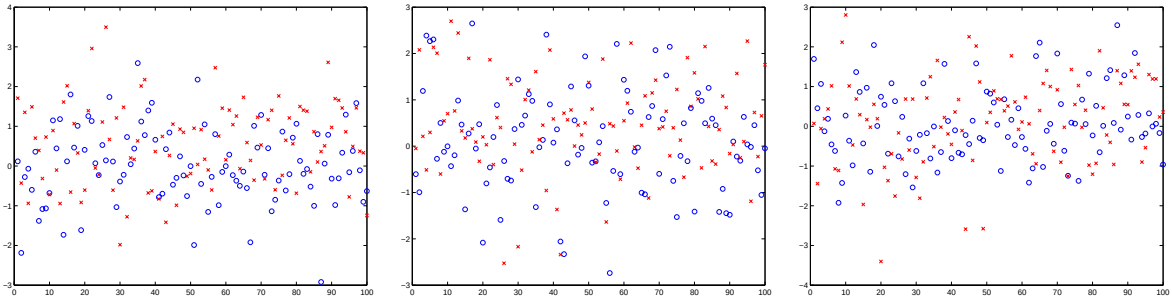
7.1 INTRODUCTION

We use tests to decide whether some assertions on some distributions are true or not. We have seen in Chapter 2 that visual tests may be used for such a purpose. Tests are an objective way to reach the same goal.

EXAMPLE 7.1: NON PAIRED DATA. A simulation study compares the execution time, on a log scale, with two compiler options. See Figure 7.1 for some data. We would like to test the hypothesis that compiler option 0 is better than 1. For one parameter set, the two series of data come from different experiments.

We can compute a confidence interval for each of the compiler options. The data looks normal, so we apply the student statistic and find the confidence intervals shown on the figure.

For parameter set 1, the confidence intervals are disjoint, so it is clear that option 0 performs better. For parameter sets 2 and 3, the intervals are overlapping, so we cannot conclude at this point.



(a) Parameter set 1 (b) Parameter set 2 (c) Parameter set 3

Parameter Set	Compiler Option 0	Compiler Option 1
1	$[-0.1669; 0.2148]$	$[0.3360; 0.7400]$
2	$[-0.0945; 0.3475]$	$[0.2575; 0.6647]$
3	$[-0.1150; 0.2472]$	$[-0.0925; 0.3477]$

Figure 7.1: Data for Example 7.1 on page 151. Top: Logarithm of execution time, on a log scale, with two compiler options (o=option 0, x=option 1) for three different parameter sets. Bottom: confidence interval for the means.

We see from this example that confidence intervals may be used in some cases for hypothesis testing, but not always. We study in this chapter how tests can be used to disambiguate such cases.

QUESTION 7.1.1. *(Example 7.1 on page 151) For one parameter set, the two data series come from different experiments. Assume, in contrast, they would come from matching pairs, i.e. the nth data point for compiler options 0 and 1 come from the same transaction. How could you decide whether compiler option 1 is better ?* [2]

[2] Compute the differences and a confidence interval for the median or the mean of the difference, and see if the confidence interval in entirely positive.

7.2 THE NEYMAN-PEARSON FRAMEWORK

7.2.1 DEFINITIONS

We are given a data sample x_i, $i = 1, ..., n$. We assume that sample is the output generated by some unknown model. We consider two possible hypotheses about the model, H_0 and H_1, and we would like to infer from the data which of the two hypotheses is true. In the Neyman-Pearson framework, the two hypotheses play different roles: H_0, the *null hypothesis*, is the conservative one. We do not want to reject it unless we are fairly sure. H_1 is the *alternative* hypothesis.

For example, with Example 7.1 on page 151, the model could be: all data points for compiler option 0 [resp. 1] are generated as iid random variables with some distribution F_0 [resp. F_1]. Then H_0 is: "$F_0 = F_1$" and H_1 is "F_0 and F_1 differ by a shift in location". This is the model used by the Wilcoxon Rank Sum test (see Example 7.9 on page 177 for more details).

Another, commonly used model, for the same example could be: all data points for compiler option 0 [resp. 1] are generated as iid random variables with some normal distribution N_{μ_0,σ^2} [resp. N_{μ_1,σ^2}]. Then H_0 is: "$\mu_0 = \mu_1$" and H_1 is "$\mu_0 \neq \mu_1$". This is the model used by the so-called "Analysis of variance" (see Example 7.4.1 on page 161 for more details).

The *critical region*, also called *rejection region* C of a test is a set of values of the tuple $(x_1, ..., x_n)$ such that if $(x_1, ..., x_n) \in C$ we reject H_0, and otherwise we accept H_0. The critical region entirely defines the test.

The output of a test is thus a binary decision: "accept H_0", or "reject H_0". The output depends on the data, which is random, and may be wrong with some (hopefully small) probability. We distinguish two types of errors

- A *type 1* error occurs if we reject H_0 when H_0 is true
- Conversely, a *type 2* error occurs if accept H_0 when H_1 is true.

The art of test development consists in minimizing both error types. However, it is usually difficult to minimize two objectives at a time. The probability of a type 1 error is called the *size* of the test. A Neyman-Pearson test is designed such that the size has a fixed, small value (in our setting, typically 5%); a good test is one that, in addition, minimizes the probability of a type 2 error.

EXAMPLE 7.2: COMPARISON OF TWO OPTIONS, REDUCTION IN RUN TIME. The reduction in run time due to a new compiler option is given in Figure 2.3 on Page 17. Assume that we know that the data comes from some iid $X_i \sim N_{\mu,\sigma^2}$. Assume we know that $\sigma = 50$. This is not realistic and we will remove such assumptions in practice, but this is convenient to make the point here.

Assume also that we want to test H_0: $\mu = \mu_0$ against H_1: $\mu = \mu_1$ with $\mu_0 = 0$ and $\mu_1 = 40$. We build a test by taking a rejection region of the form

$$C = \left\{ (x_1, ..., x_n) \text{ such that } \frac{x_1 + ... + x_n}{n} > k \right\} \tag{7.1}$$

In other words, we reject H_0 if the sample mean is too large (since the alternative hypothesis H_1 assumes $\mu = \mu_1 > \mu_0$). We want a test of size $\alpha = 0.05$. This allows

us to compute k, as follows (where $\bar{X} = \frac{1}{n}\sum_i X_i$). We want k such that

$$\begin{aligned} \alpha &= \mathbb{P}_{H_0}\left((X_1, ..., X_n) \in C\right) \\ &= \mathbb{P}_{H_0}\left(\bar{X} > k\right) = \mathbb{P}_{H_0}\left(\frac{\sqrt{n}}{\sigma}\bar{X} > \frac{\sqrt{n}}{\sigma}k\right) \end{aligned}$$

Now, under H_0, $\frac{\sqrt{n}}{\sigma}\bar{X}$ has a standard normal distribution. Thus we want k such that $\frac{\sqrt{n}}{\sigma}k = \eta$, with $N_{0,1}(\eta) = 1 - \alpha$ ($\eta = 1.645$ for $\alpha = 0.05$). Thus we reject H_0 when the sample mean is larger than $k = \frac{1.645 \times \sigma}{\sqrt{n}}$ ($= 8.23$ for $n = 100$). We have $\bar{x} = 26.1$, so we reject H_0.

The probability of an error of type 2 is

$$\begin{aligned} \beta &= \mathbb{P}_{H_1}\left(\bar{X} \leq k\right) \\ &= \mathbb{P}_{H_1}\left(\sqrt{n}\frac{\bar{X} - \mu_1}{\sigma}) \leq \sqrt{n}\frac{k - \mu_1}{\sigma}\right) \\ &= N_{0,1}\left(\sqrt{n}\frac{\bar{X} - \mu_1}{\sigma}\right) \end{aligned}$$

For $n = 100$, $\beta \approx 10^{-10}$.

Now reverse the hypotheses, so that we have H_0: $\mu = 0$ against H_1: $\mu = \mu_1$. We take a rejection region of the form

$$C = \left\{(x_1, ..., x_n) \text{ such that } \frac{x_1 + ... + x_n}{n} < k'\right\} \tag{7.2}$$

and we compute k in a similar way. We find

$$k' = -\frac{\sigma}{\sqrt{n}}\eta + \mu_1 = 31.77$$

Since the sample mean is in the rejection region, we also reject H_0 in this case ! This shows how the preferential treatment given to H_0 by the Neyman-Pearson framework.

7.2.2 p-VALUE OF A TEST.

For many tests, the rejection region has the form $\{T(x) > m_0\}$, where x is the observation, $T()$ some mapping, and $m_0$is a parameter that depends on the size of the test. (In Example 7.2 on page 152 we have $T(x) = \bar{x}$ for the former case, $T(x) = -\bar{x}$ for the latter.)

The p-value of a test is defined as the probability, under H_0, that T is larger than the observed value.

DEFINITION 7.2.1. *The p-value of an observation x is* $\mathbb{P}_{H_0}(T(X) > T(x))$.

In this formula, X is a random variable that represents a hypothetical replication of the experiment, whereas x is the data that we have observed.

More formally, call ϕ the mapping

$$\begin{aligned} [0, +\infty) &\rightarrow [0, +\infty) \\ m &\mapsto \sup_{\theta \in H_0} \mathbb{P}_\theta \left\{T(X) > m\right\} \end{aligned}$$

Here θ is a model, and $\theta \in H_0$ means that the model satisfies the hypothesis H_0. Note that ϕ is wide-sense decreasing. The p-value of an observation x is

$$p^*(x) := \phi(T(x))$$

PROPOSITION 7.2.1. *Assume that ϕ is strictly decreasing. The test is equivalent to: reject H_0 iff $p^*(x) < \alpha$, where α is the size of the test.*

Proof. The rejection region is

$$C := \{x : \ T(x) > m_0\} = \{x : \ \phi(T(X)) < \alpha\} = \{x : \ p^*(x) < \alpha\}$$

□

The assumption that ϕ is strictly decreasing is usually true in practice. In other words, **the test rejects H_0 when the p-value is smaller than the test size α**.

The interest of the p-value is the explicit dependence on α. It gives more information than just a binary answer.

QUESTION 7.2.1. *What is the relation between α, ϕ and m_0 ?* [3]

EXAMPLE: CONTINUATION OF EXAMPLE 7.2 ON PAGE 152. For the first test ($\mu_0 = 0$ versus $\mu_1 = 40$), the rejection region is $\{\bar{x} > k\}$ and $T(x) = \bar{x}$. Thus

$$\begin{aligned} p^* &= \mathbb{P}_{H_0}\left(\bar{X} > \bar{x}\right) \\ &= \mathbb{P}_{H_0}\left(\sqrt{n}\frac{\bar{X} - \mu_0}{\sigma} > \sqrt{n}\frac{\bar{x} - \mu_0}{\sigma}\right) \\ &= 1 - N_{0,1}\left(\sqrt{n}\frac{\bar{x} - \mu_0}{\sigma}\right) \end{aligned}$$

We find $p^* = 1.0489e-010$ which is small, therefore we reject H_0.

In the second test, the rejection region has the form $\{\bar{x} < k\}$. The p-value is now

$$\begin{aligned} p^* &= \mathbb{P}_{H_0}\left(-\bar{X} > -\bar{x}\right) = \mathbb{P}_{H_0}\left(\bar{X} < \bar{x}\right) \\ &= N_{0,1}\left(\sqrt{n}(\bar{x} - \mu_1)/\sigma\right) \end{aligned}$$

We find $p^* = 0.0027$ which is less small but still smaller than 0.05, therefore we also reject H_0.

7.3 LIKELIHOOD RATIO TESTS

In this section we introduce a generic framework, used in most of this chapter, for constructing tests. We give the application to simple tests for paired data and for goodness of fit.

[3] $\alpha = \phi(m_0)$

7.3.1 DEFINITION OF LIKELIHOOD RATIO TEST

ASSUMPTIONS AND NOTATION We assume some probability space parameterized by some $\theta \in \Theta$. Consider $\Theta_0 \subset \Theta$ (nested models.). We have $H_0 :=$ "$\theta \in \Theta_0$" whereas $H_1 :=$ "$\theta \in \Theta \setminus \Theta_0$". For a given statistic (random variable) X and value x of X, define :

- $l_x(\theta) := \ln f_X(x|\theta)$ where $f_X(.|\theta)$ is the probability density of the model, when the parameter is θ.
- $l_x(H_0) = \sup_{\theta \in \Theta_0} l_x(\theta)$
- $l_x(H_1) = \sup_{\theta \in \Theta} l_x(\theta)$

For example, assume some data comes from an iid sequence of normal RVs $\sim N(\mu, \sigma)$. We want to test $\mu = 0$ versus $\mu \neq 0$. Here $\Theta = \{(\mu, \sigma > 0)\}$ and $\Theta_0 = \{(0, \sigma > 0)\}$.

If H_0 is true, then, approximately, the likelihood is maximum for $\theta \in \Theta_0$ and thus $l_x(H_0) = l_x(H_1)$. In the opposite case, the maximum likelihood is probably reached at some $\theta \notin \Theta_0$ and thus $l_x(H_1) > l_x(H_0)$. This gives an idea for a generic family of tests:

DEFINITION 7.3.1. *The likelihood ratio test is defined by the rejection region*

$$C = \{l_x(H_1) - l_x(H_0) > k\}$$

where k is chosen based on the required size of the test.

The test statistic $l_x(H_1) - l_x(H_0)$ is called *likelihood ratio* for the two hypotheses H_0 and H_1.

Thus we reject $\theta \in \Theta_0$ when the likelihood ratio statistic is large. The Neyman-Pearson lemma ([Weber-C11] Section 6.3) tells us that, in the simple case where Θ_0 and Θ_1 contain only one value each, the likelihood ratio test minimizes the probability of type 2 error. Most tests used in this lecture are actually likelihood ratio tests. As we will see later, for large sample size, there are simple, generic results for such tests.

There is a link with the theory of maximum likelihood estimation. Under the conditions in Definition 2.8.1, define

- $\hat{\theta}_0$: the MLE of θ when we restrict θ to be in Θ_0
- $\hat{\theta}$: the unrestricted MLE of θ

Then $l_x(H_0) = l_x(\hat{\theta}_0)$ and $l_x(H_1) = l_x(\hat{\theta})$.

QUESTION 7.3.1. *Why can we be sure that $l_x(\hat{\theta}) - l_x(\hat{\theta}_0) \geq 0$?* [4]

EXAMPLE: CONTINUATION OF EXAMPLE 7.2 ON PAGE 152. We want to test H_0: $\mu = \mu_0$ against H_1: $\mu = \mu_1$. Thus $\Theta_0 = \{\mu_0\}$ and $\Theta = \{\mu_0, \mu_1\}$. The log-likelihood of an observation is

$$l_x(\mu) = \frac{-n}{2} \ln\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2} \sum_i (x_i - \mu)^2$$

[4] As long as the MLEs exist: by definition, $l_x\left(\hat{\theta}\right) \geq l_x(\theta)$ for any θ.

and the likelihood ratio statistic is

$$l_x(H_1) - l_x(H_0) = \max\{l_x(\mu_1), l_x(\mu_0)\} - l_x(\mu_0) = [l_x(\mu_1) - l_x(\mu_0)]^+$$

where $[r]^+$ denotes the maximum of r and 0. The likelihood ratio test is of the form $[l_x(\mu_1) - l_x(\mu_0)]^+ > k$, which, for $k > 0$ is equivalent to $l_x(\mu_1) - l_x(\mu_0) > k$. After some algebra, it comes

$$l_x(H_1) - l_x(H_0) = \frac{n}{2\sigma^2}\left(2\bar{x}(\mu_1 - \mu_0) + (\mu_0^2 - \mu_1)^2\right)$$

which is an increasing function of $\bar{x}$, thus for $\mu_1 > \mu_0$, as in the first case ($\mu_1 = 40$ and $\mu_0 = 0$), the rejection region for the likelihood ratio test has the form $\bar{x} > k$. In contrast, if $\mu_1 < \mu_0$, it has the form $\bar{x} < k$. Thus the tests derived heuristically in Example 7.2 on page 152 are in fact likelihood ratio tests.

7.3.2 STUDENT TEST FOR SINGLE SAMPLE (OR PAIRED DATA)

This test applies to a single sample of data, assumed to be normal with unknown mean and variance. It can also be applied to two paired samples, after computing the differences.

The model is: $X_1, ..., X_n \sim$ iid N_{μ,σ^2} where μ and σ are not known. The hypotheses are:

H_0: $\mu = \mu_0$ against H_1: $\mu \neq \mu_0$

where μ_0 is a fixed value.

We compute the likelihood ratio statistic. We have, after some algebra:

$$\begin{aligned}
l_x(H_1) - l_x(H_0) &= \max_{\mu,\sigma^2} \ln f_X(x|\mu,\sigma^2) - \max_{\sigma^2} \ln f_X(x|\mu_0,\sigma^2) \\
&= \frac{n}{2}\left(-\ln\left(\sum_i (x_i - \bar{x})^2\right) + \ln\left(\sum_i (x_i - \mu_0)^2\right)\right) \\
&= \frac{n}{2}\left(-\ln\left(\sum_i (x_i - \bar{x})^2\right) + \ln(\sum_i (x_i - \bar{x})^2 + n(\bar{x} - \mu_0)^2)\right) \\
&= \frac{n}{2}\ln\left(1 + \frac{n(\bar{x} - \mu_0)^2}{\sum_i (x_i - \bar{x})^2}\right)
\end{aligned}$$

Let $T(x)$ be the student statistic (Theorem 2.3.1):

$$T(x) = \sqrt{n}\frac{\bar{x} - \mu_0}{\hat{\sigma}} \tag{7.3}$$

with $\hat{\sigma}^2 = \frac{1}{n-1}\sum_i (x_i - \bar{x})^2$. We can write the likelihood ratio statistic as

$$l_x(H_1) - l_x(H_0) = \frac{n}{2}\ln\left(1 + \frac{T(x)^2}{n-1}\right) \tag{7.4}$$

which is an increasing function of $|T(x)|$. The rejection region thus has the form

$$C = \{|T(x)| > \eta\}$$

We compute η from the condition that the size of the test is α. Under H_0, $T(X)$ has a student distribution t_{n-1} (Theorem 2.3.1). Thus

$$\eta = t_{n-1}^{-1}\left(1 - \frac{\alpha}{2}\right) \tag{7.5}$$

For example, for $\alpha = 0.05$ and $n = 100$, $\eta = 1.98$.

The p-value is

$$p^* = 2(1 - t_{n-1}(T(x))) \tag{7.6}$$

EXAMPLE 7.3: PAIRED DATA. This is a variant of Example 7.2 on page 152. Consider again the reduction in run time due to a new compiler option, as given in Figure 2.3 on Page 17. We want to test whether the reduction is significant. We assume the data is iid normal and use the student test:

H_0: $\mu = 0$ against H_1: $\mu \neq 0$

The test statistic is $T(x) = 5.08$, larger than 1.98, so we reject H_0. Alternatively, we can compute the p-value and obtain $p^* = 1.80e - 006$, which is small, so we reject H_0.

We can compare this test to the use of a confidence interval. A confidence interval for μ is (Theorem 2.3.1)

$$\bar{x} \pm \eta\frac{\hat{\sigma}}{\sqrt{n}} \tag{7.7}$$

We could decide to reject H_0 iff μ_0 is not in the confidence interval, i.e.

$$|\bar{x} - \mu_0| > \eta\frac{\hat{\sigma}}{\sqrt{n}} \tag{7.8}$$

which is exactly the same as the condition $T(x) > \eta$, which is the rejection condition of the student test. Thus there is equivalence between testing for the mean equal to μ_0 and asking whether μ_0 is in a confidence interval for the mean.

This result is quite general: consider a generic model parametrized with some $\theta \in \Theta \subset \mathbb{R}$. There is equivalence between tests of the form

$\theta = \theta_0$ against H_1: $\theta \neq \theta_0$

with computing confidence intervals for θ [Weber-C11]. For such cases, we do not need a general theory of tests, since we can simply use confidence intervals as discussed in Chapter 2. However, there are many tests that cannot be put in this form.

7.3.3 THE SIMPLE GOODNESS OF FIT TEST

Assume we are given n data points $x_1, ..., x_n$, assumed to be generated from an iid sequence, and we want to verify whether their common distribution is a given distribution $F()$. A traditional method is to compare the empirical histogram to the theoretical one. Applying this idea gives the following likelihood ratio test. We call it the *simple goodness of fit test* as the null hypothesis is for a given, fixed distribution $F()$ (as opposed to a family of distributions, which would give a *composite* goodness of fit test).

To compute the empirical histogram, we partition the set of values of X into *bins* B_i. Let $N_i = \sum_{k=1}^{n} 1_{\{B_i\}}(X_k)$ (number of observation that fall in bin B_i) and $q_i = \mathbb{P}\{X_1 \in B_i\}$. If the data comes from the distribution $F()$ the distribution of N is multinomial $M_{n,\vec{q}}$, i.e.

$$\mathbb{P}\{N_1 = n_1, ..., N_k = n_k\} = \left(\frac{n!}{n_1!...n_k!} \right) p_1^{n_1}...p_k^{n_k} \tag{7.9}$$

The test is

H_0: N_i comes from the multinomial distribution $M_{n,\vec{q}}$

against

H_1: N_i comes from a multinomial distribution $M_{n,\vec{p}}$ for some arbitrary $\vec{p}$.

We now compute the likelihood ratio statistic. The parameter is $\theta = \vec{p}$. Under H_0, there is only one possible value so $\hat{\theta}_0 = \vec{q}$. From Equation (7.9), the likelihood is

$$l_{\vec{p}}(\vec{x}) = C + \sum_{i=1}^{k} n_i \ln(p_i) \tag{7.10}$$

where $n_i = \sum_{k=1}^{n} 1_{\{B_i\}}(x_k)$ and $C = \ln(n!) - \sum_{i=1}^{k} \ln(n_i!)$. C is a constant and can be ignored in the rest. To find $\hat{\theta}$, we have to maximize Equation (7.10) subject to the constraint $\sum_{i=1}^{k} p_i = 1$. The function to maximize is concave in p_i, so we can find the maximum by the lagrangian technique. The lagrangian is

$$L(\vec{p}, \lambda) = \sum_{i=1}^{k} n_i \ln(p_i) + \lambda(1 - \sum_{i=1}^{k} p_i) \tag{7.11}$$

The equations $\frac{\partial L}{\partial p_i} = 0$ give $n_i = \lambda p_i$. Consider first the case $n_i \neq 0$ for all i. We find λ by the constraint $\sum_{i=1}^{k} p_i = 1$, which gives $\lambda = n$ and thus $\hat{p}_i = \frac{n_i}{n}$. Finally, the likelihood ratio statistic is

$$l_{H_1}(\vec{x}) - l_{H_0}(\vec{x}) = \sum_{i=1}^{k} n_i \ln \frac{n_i}{nq_i} \tag{7.12}$$

In the case where $n_i = 0$ for some i, the formula is the same if we adopt the convention that, in Equation (7.38), the term $n_i \ln \frac{n_i}{nq_i}$ is replaced by 0 whenever $n_i = 0$.

We now compute the p-value. It is equal to

$$\mathbb{P}\left(\sum_{i=1}^{k} N_i \ln \frac{N_i}{nq_i} > \sum_{i=1}^{k} n_i \ln \frac{n_i}{nq_i} \right) \tag{7.13}$$

where $\vec{N}$ has the multinomial distribution $M_{n,\vec{q}}$.

For large n, we will see in Section 7.5 a simple approximation for the p-value. If n is not large, there is no known closed form, but we can use Monte Carlo simulation as discussed in Section 3.4.

EXAMPLE 7.4: MENDEL [WEBER-C11]. Mendel crossed peas and classified the results in 4 classes of peas $i = 1, 2, 3, 4$. If his genetic theory is true, the probability that a pea belongs to class i is $q_1 = 9/16, q_2 = q_3 = 3/16, q_4 = 1/16$. In one experiment, Mendel obtained $n = 556$ peas, with $N_1 = 315, N_2 = 108, N_3 = 102$ and $N_4 = 31$. The test is

H_0 : "$\vec{q} = \vec{p}$" against H_1 : "$\vec{p}$ is arbitrary"

The test statistic is

$$\sum_{i=1}^{k} n_i \ln \frac{n_i}{nq_i} = 0.3092 \tag{7.14}$$

We find the p-value by Monte-Carlo simulation (Example 3.7 on page 74) and find $p = 0.9191 \pm 0.0458$. The p-value is (very) large thus we accept H_0.

7.4 ANOVA

In this section we cover a family of exact tests when we can assume that the data is normal. It applies primarily to cases with multiple, unpaired samples.

7.4.1 ANALYSIS OF VARIANCE (ANOVA) AND F-TEST

Analysis of variance (ANOVA) is used when we can assume that the data is a family of independent normal variables, with an arbitrary family of means, but with common variance. The goal is to test some property of the mean. The name ANOVA is explained by Theorem 7.4.1.

ANOVA is found under many variants, and the basis is often obscured by complex computations. All variants of ANOVA are based on a single result, which we give next; they differ only in how a projection is computed.

ASSUMPTIONS AND NOTATION FOR ANOVA

- The data is a collection of N *independent, normal* random variables X_r, where the index r is in some finite set R (with $N =$ number of elements in R).
- $X_r \sim N(\mu_r, \sigma^2)$, i.e. all variables have the *same variance* (this is pompously called "homoscedasticity"). The common variance is fixed but unknown.
- Call $\vec{\mu} := (\mu_r)_{r \in R}$. We assume that $\vec{\mu} \in M$, where M is a linear subspace of $\mathbb{R}^R$. Let $k = \dim M$. The parameter is $\theta = (\vec{\mu}, \sigma)$ and the parameter space is $\Theta = M \times (0, +\infty)$
- We want to test the nested model $\vec{\mu} \in M_0$, where M_0 is a linear sub-space of M. Let $k_0 = \dim M_0$. We have $\Theta_0 = M_0 \times (0, +\infty)$.

- Π_M [resp. Π_{M_0}] is the orthogonal projector on M [resp. M_0]
- $F_{m,n}()$ is the Fisher distribution with degrees of freedom m, n

---

EXAMPLE: NON PAIRED DATA.(Continuation of Example 7.1 on page 151) Consider the data for one parameter set. The model is

$$X_i = \mu_1 + \epsilon_{1,i} \;\; Y_j = \mu_2 + \epsilon_{2,j} \tag{7.15}$$

with $\epsilon_{i,j} \sim$ iid N_{0,σ^2}.

We can model the collection of variables as $X_1, ..., X_m, Y_1, ..., Y_n$ thus $R = \{1, ..., m+n\}$ and $N = n + m$. We have then

- $M = \{(\mu_1, ...\mu_1, \mu_2, ...\mu_2), \mu_1 \in \mathbb{R}, \mu_2 \in \mathbb{R}\}$ and $k = 2$
- $M_0 = \{(\mu, ...\mu, \mu, ...\mu), \mu \in \mathbb{R}\}$ and $k_0 = 1$
- $\Pi_M(x_1, ..., x_m, y_1, ..., y_n) = (\bar{x}, ..., \bar{x}, \bar{y}, ..., \bar{y})$, where $\bar{x} = (\sum_{i=1}^m x_i)/m$ and $\bar{y} = (\sum_{j=1}^n y_j)/n$.
- $\Pi_{M_0}(x_1, ..., x_m, y_1, ..., y_n) = (\bar{z}, ..., \bar{z}, \bar{z}, ..., \bar{z})$, where $\bar{z} = (\sum_{i=1}^m x_i + \sum_{j=1}^n y_j)/(m+n)$.

This model belongs to the family of "one way ANOVA" models, and can be solved using statistical packages.

EXAMPLE 7.5: FRUITFLIES. [Weber-C11] The longevity of different varieties of fruitflies was measured, on groups of 25 flies. The results are:

Group	*Mean life (days)*	*standard deviation*
1	63.56	16.4522
2	64.80	15.6525
3	63.36	14.5398

(here the *standard deviation* is $\sqrt{\frac{1}{n-1}\sum_{i=1}^n (x_i - \bar{x})^2}$). The model is

$$X_{i,j} = \mu_i + \epsilon_{i,j} \;\; 1 \leq n_i \; i = 1, ..., k \tag{7.16}$$

with $\epsilon_{i,j} \sim$ iid N_{0,σ^2}. It is also called the generic one-way ANOVA model (one way because there is one "factor", index i. Here i represents the variety of fruitflies, and j the index of a sample within a variety.

The collection is $X_r = X_{i,j}$ so $R = \{(i,j), i = 1, ..., k = 3 \text{ and } j = 1, ..., n_i\}$ and $N = \sum_i n_i$. We have

- $M = \{(\mu_{i,j}) \text{ such that } \mu_{i,j} = \mu_i$ is independent of $j\}$; the dimension of M is $k = 3$.
- $M_0 = \{(\mu_{i,j}) \text{ such that } \mu_{i,j} = \mu$ is independent of $i, j\}$ and $k_0 = 1$.
- $\Pi_M(\vec{x})$ is the vector whose (i,j)th coordinate is independent of j and is equal to $\bar{x}_{i.} := (\sum_{j=1}^{n_i} x_{i,j})/n_i$.

- $\Pi_{M_0}(\vec{x})$ is the vector whose coordinates are all identical and equal to the overall mean $\bar{x}_{..} := (\sum_{i,j} x_{i,j})/N$.

THEOREM 7.4.1 (ANOVA). *1. The Maximum Likelihood Estimators for both restricted and general models are given by*

- $\hat{\mu}_0 = \Pi_{M_0}(\vec{x})$, $\hat{\sigma}_0^2 = \frac{1}{N}\|\vec{x} - \hat{\mu}_0\|^2$
- $\hat{\mu} = \Pi_M(\vec{x})$, $\hat{\sigma}^2 = \frac{1}{N}\|\vec{x} - \hat{\mu}\|^2$

where $\vec{x}$ is the value of the random variable $\vec{X}$.

2. *The likelihood ratio for the test of H_0: "$\vec{\mu} \in M_0, \sigma > 0$" against H_1: "$\vec{\mu} \in M \setminus M_0, \sigma > 0$" is*

$$-\frac{N}{2}\ln\frac{SS1}{SS0} = \frac{N}{2}\ln\left(1 + \frac{SS2}{SS1}\right)$$

where

$$SS0 := \|\vec{x} - \hat{\mu}_0\|^2 = N\hat{\sigma}_0^2 = SS1 + SS2$$

$$SS1 := \|\vec{x} - \hat{\mu}\|^2 = N\hat{\sigma}^2$$

$$SS2 := \|\hat{\mu} - \hat{\mu}_0\|^2 = N \times (\hat{\sigma}^2 - \hat{\sigma}_0^2)$$

3. *Define the test statistic f by*

$$f := \frac{SS2/(k - k_0)}{SS1/(N - k)}$$

The distribution of f (when we replace $\vec{x}$ by $\vec{X}$) under H_0 is $F_{k-k_0,N-k}$.
F is often called the F-value of the test.

4. *The likelihood ratio test of size α rejects $\vec{\mu} \in M_0$ when f is large, i.e., when $f > \eta$, where $F_{k-k_0,N-k}(\eta) = 1 - \alpha$. The p-value is $p^* = 1 - F_{k-k_0,N-k}(f)$.*

Proof. Apply Theorem 12.5.3

□

EXAMPLE: APPLICATION TO EXAMPLE 7.1 ON PAGE 151. We assume homoscedasticity. We can test this hypothesis by applying the test in Section 7.4.4.

The theorem gives the following computations:

- $\hat{\mu} = (\bar{X}, ..., \bar{X}, \bar{Y}, ..., \bar{Y})$ and $\hat{\sigma} = \frac{1}{n+m}(\sum_i (X_i - \bar{X})^2 + \sum_j (Y_j - \bar{Y})^2)$
- $\hat{\mu}_0 = (\bar{Z}, ..., \bar{Z}, \bar{Z}, ..., \bar{Z})$ with $\bar{Z} = (m\bar{X} + n\bar{Y})(m + n)$ and $\hat{\sigma}_0 = \frac{1}{n+m}(\sum_i (X_i - \bar{Z})^2 + \sum_j (Y_j - \bar{Z})^2)$
- $SS1 = \sum_i (X_i - \bar{X})^2 + \sum_j (Y_j - \bar{Y})^2) = S_{XX} + S_{YY}$
- $SS2 = m(\bar{Z} - \bar{X})^2 + n(\bar{Z} - \bar{Y})^2 = (\bar{X} - \bar{Y})^2/(1/m + 1/n)$
- the f value is $SS2/SS1/(m + n - 2)$.

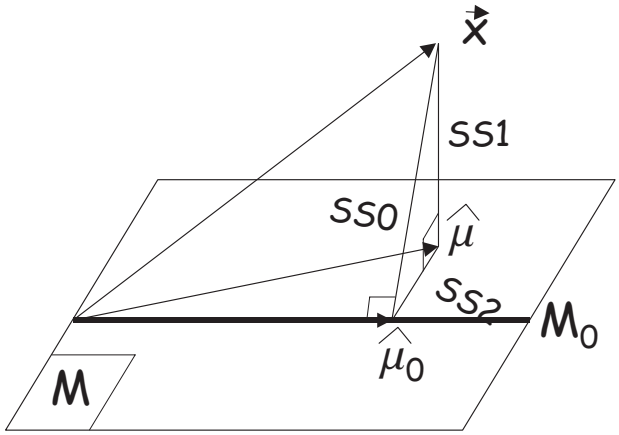


Figure 7.2: Illustration of quantities in Theorem 7.4.1

Parameter Set 1	SS	df	MS	F	Prob>F
Columns	13.2120	1	13.2120	13.4705	0.0003116
Errors	194.2003	198	0.9808		
total	207.4123	199			
Parameter Set 2	SS	df	MS	F	Prob>F
Columns	5.5975	1	5.5975	4.8813	0.0283
Errors	227.0525	198	1.1467		
total	232.6500	199			
Parameter Set 3	SS	df	MS	F	Prob>F
Columns	0.1892	1	0.1892	0.1835	0.6689
Errors	204.2256	198	1.0314		
total	204.4148	199			

Table 7.1: ANOVA Tests for Example 7.1 on page 151 (Non Paired Data)

The ANOVA tables for parameter sets 1 to 3 are given in Table 7.1. The F-test rejects the hypothesis of same mean for parameter sets 1 and 2, and accepts it for parameter set 3. The software used to produce this example uses the following terminology:

- SS2: “Columns” (explained variation, variation between columns, or between groups)
- SS1: “Error” (residual variation, unexplained variation)
- SS0: “Total” (total variation)

QUESTION 7.4.1. *Compare to the confidence intervals given in the introduction.* [5]

QUESTION 7.4.2. *What are SS0, SS1 and SS2 for parameter set 1 ?* [6]

[5] For parameter set 1, the conclusion is the same as with confidence interval. For parameter sets 2 and 3, confidence intervals did not allow one to conclude. ANOVA disambiguates these two cases.

[6] The column “SS” gives, from top to bottom: SS2, SS1 and SS0.

INTERPRETATION. Item 2 in the theorem justifies the name "ANOVA": the likelihood ratio statistic depends only on estimators of variance. Note that this is very specific of homoscedasticity.

The equality

$$SS0 = SS1 + SS2$$

can be interpreted as a decomposition of sum of squares, as follows. Consider Θ_0 as the base model, with k_0 dimensions for the mean; we ask ourselves whether it is worth considering the more complex model Θ, which has $k > k_0$ dimensions for the mean. From its definition, we can interpret those some of squares as follows.

- $SS2$ is the sum of squares explained by the model Θ, or explained variation.
- $SS1$ is the residual sum of squares
- $SS0$ is the total sum of squares

The likelihood ratio test accepts Θ when $SS2/SS1$ is large, i.e., when the percentage of sum of squares $SS2/SS1$ (also called percentage of variation) explained by the model Θ is high.

The dimensions are interpreted as degrees of freedom:

- $SS2$ (explained variation) is in the orthogonal of M_0 in M, with dimension $k - k_0$: the number of degrees of freedom for $SS2$ is $k - k_0$
- $SS1$ (residual variation) in in the orthogonal of M in $\mathbb{R}^R$. The number of degrees of freedom for $SS1$ is $N - k$

EXAMPLE: FRUITFLIES.The numerical solution of Example 7.5 on page 160is shown in the table below.

Source	SS	df	MS	F	Prob>F
Columns	30.427	2	15.213	0.0628	0.9392
Errors	17449.92	72	242.36		
total	17480.35	74			

Thus we accept H_0, namely, longevity is not impacted by the variety.

QUESTION 7.4.3. *Write down the expressions of MLEs, $SS1$, $SS2$ and the F-value.* [7]

[7]

- $\hat{\mu}$ is the vector whose (i,j)th coordinate is independent of j and is equal to $\bar{X}_{i.} := \sum_{j=1}^{n_i} X_{i,j}/n_i$.
- $SS1 = \sum_{i,j}(X_{i,j} - \bar{X}_{i.})^2$
- $\hat{\sigma}^2 = \frac{1}{N}SS1$
- $\hat{\mu}_0$ is the vector coordinates are all identical and equal to the overall mean $\bar{X}_{..} := (\sum_{i,j} X_{i,j})/N$
- $SS2 = \sum_i n_i(\bar{X}_{i.} - \bar{X}_{..})^2$
- $SS0 = SS1 + SS2$
- $\hat{\sigma}_0^2 = \frac{1}{N}SS0$
- $F = SS2/SS1 * (N-k)/(k-1)$

7.4.2 STUDENT TEST AS SPECIAL CASE OF ANOVA

In the specila case where $k - k_0 = 1$ (as in Example 7.1 on page 151) the F-statistic is the square of a student statistic, and a student test could be used instead. This is sometimes used by some statistics packages.

7.4.3 TESTING FOR SPECIFIC VALUES

By an additive change of variable, we can extend the ANOVA framework to the case where $M_0 \subset M$ are affine (instead of linear) varieties of $\mathbb{R}^R$. This includes testing for a specific value.

For example, assume we have the model

$$X_{i,j} = \mu_i + \epsilon_{i,j} \tag{7.17}$$

with $\epsilon_{i,j} \sim$ iid N_{0,σ^2}. We want to test

H_0: "$\mu_i = \mu_0$ for all i" against H_1: "μ_i unconstrained"

We change model by letting $X'_{i,j} = X_{i,j} - \mu_0$ and we are back to the ANOVA framework.

7.4.4 TESTING FOR A COMMON VARIANCE

We often need to verify that the common variance assumption holds. This can be done as follows.

$I > 2$ DATA SETS

We are given a data set $x_{i,j}$, $i = 1, ..., I$, $j = 1, ..., n_i$. We assume that it is a realization of the model

$$X_{i,j} \sim iidN(\mu_i, \sigma_i^2) \tag{7.18}$$

We assume that the normal assumption holds and we want to test

H_0 $\sigma_i = \sigma > 0$ for all i
H_1 $\sigma_i > 0$

We make a likelihood ratio test. We compute the likelihood ratio statistic. We need first to compute the maximum likelihood under H_1. The log-likelihood of the model is

$$l_x(\vec{\mu}, \vec{\sigma}) = -\frac{1}{2}\left[\ln(2\pi) + \sum_{i=1}^{I}\left(2n_i \ln(\sigma_i) + \sum_{j=1}^{n_i}\frac{(x_{i,j} - \mu_i)^2}{\sigma_i^2}\right)\right] \tag{7.19}$$

To find the maximum under H1, observe that the terms in the summation do not have cross dependencies, thus we can maximize each of the I terms separately. The maximum of the ith term is for

$$\mu_i = \hat{\mu}_i := \frac{1}{n_i}\sum_{j=1}^{I} x_{i,j} \tag{7.20}$$

$$\sigma_i^2 = s_i^2 := \frac{1}{n_i}\sum_{j=1}^{I}(x_{i,j} - \hat{\mu}_i)^2 \tag{7.21}$$

and thus

$$l_x(H_1) = -\frac{1}{2}\left[\ln(2\pi) + \sum_{i=1}^{I} n_i \left(2\ln(s_i) + 1\right)\right] = -\frac{1}{2}\left[\ln(2\pi) + n + 2\sum_{i=1}^{I} n_i \ln(s_i)\right] \quad (7.22)$$

where $n = \sum_{i=1}^{I} n_i$.

Under H_0 the likelihood is as in Equation (7.19) but with σ_i replaced by the common value σ. To find the maximum,we use the ANOVA theorem. The maximum is for $\mu_i = \hat{\mu}_i$ as in Equation (7.20) and

$$\sigma^2 = s^2 := \frac{1}{n}\sum_{i=1}^{I}\sum_{j=1}^{n_i}(x_{i,j} - \hat{\mu}_i)^2 = \sum_{i=1}^{I}\frac{n_i}{n}s_i^2 \quad (7.23)$$

and thus

$$l_x(H_0) = -\frac{1}{2}\left[\ln(2\pi) + \sum_{i=1}^{I} n_i \frac{s_i^2}{s^2} + 2n\ln(s)\right] = -\frac{1}{2}\left[\ln(2\pi) + n + 2n\ln(s)\right] \quad (7.24)$$

The test statistic is the likelihood ratio statistic $lrs = l_x(H_1) - l_x(H_0)$:

$$lrs = n\ln(s) - \sum_{i=1}^{I} n_i \ln(s_i) \quad (7.25)$$

The test has the form: reject H_0 when $lrs > K$ for some constant K.

The p-value can be obtained using either Monte-Carlo simulation or large sample asymptotics. The former method proceeds as follows. The problem is now to compute $\mathbb{P}(T > l)$ where T is a random variable distributed like

$$n\ln(s) - \sum_{i=1}^{I} n_i \ln(s_i) \quad (7.26)$$

and assuming H_0 holds. We generate R replicated samples of T. To generate these samples, observe that all we need is to generate the random variables s_i. They are independent, and distributed like $\sigma^2\chi^2_{n_i-1}$. Note that T is independent of the specific value of the unknown but fixed parameter σ, thus we can let $\sigma = 1$ in the Monte Carlo simulation.

Alternatively, one can use the large sample asymptotic. The distribution of $2 \times lrs$ is approximately χ^2_{I-1}; this gives the approximate p-value:

$$p^* \approx 1 - \chi^2_{I-1}(2n\ln(s) - 2\sum_{i=1}^{I} n_i \ln(s_i))$$

$I = 2$ DATA SETS: F-TEST

For two samples, the rejection region of the test of common variance can be computed explicitly, by showing that, in this case, the test is an F-test.

We can rewrite the likelihood ratio statistic as

$$lrs = \frac{1}{2}\left[n\ln(n_1 f + n_2) - n_1 \ln(f)\right] + C \quad (7.27)$$

where C is a constant term (assuming n_1 and n_2 are fixed) and

$$f = \frac{s_1^2}{s_2^2} \tag{7.28}$$

The derivative of lrs with respect to f is

$$\frac{\partial lrs}{\partial f} = \frac{n_1 n_2 (f-1)}{2f(n_1 f + n_2)} \tag{7.29}$$

thus lrs decreases with f for $f < 1$ and increases for $f > 1$. Thus the rejection region, defined as $\{lrs > K\}$, is also of the form $\{K_1 < f < K_2\}$.

Now define

$$F = \frac{\hat{\sigma}_1^2}{\hat{\sigma}_2^2} \tag{7.30}$$

with

$$\hat{\sigma}_i^2 := \frac{1}{n_i - 1} \sum_{j=1}^{I} (x_{i,j} - \hat{\mu}_i)^2 \tag{7.31}$$

We take $\hat{\sigma}_i^2$ instead of s_i^2 in order to obtain an F-test, as we see next. Note that

$$F = fC' \tag{7.32}$$

where C' is a constant, so the set $\{K_1 < f < K_2\}$ is equal to the set $\{C'K_1 < F < C'K_2\}$ with $\eta = C'K_1$ and $\xi = C'K_2$.

Under H_0, the distribution of F is Fisher with parameters $(m-1, n-1)$, so we have a Fisher test. The bounds η and $F > \xi$ are classically computed by the conditions

$$\begin{cases} F_{m-1,n-1}(\eta) = \alpha/2 \\ F_{m-1,n-1}(\xi) = 1 - \alpha/2 \end{cases}$$

EXAMPLE: FRUITFLIES AGAIN. We want to teste whether the data for groups 1 and 3 in Example 7.5 on page 160 have the same variance. We have $F_{24,24}(\xi) = 1 - \alpha/2$. Thus $\eta = 0.44$ and $\xi = 2.27$. The F statistic is 1.2804 so we accept H_0.

7.5 ASYMPTOTIC RESULTS

In many cases it is hard to find the exact distribution of a test statistic. An interesting feature of likelihood ratio tests is that we have a simple asymptotic result.

7.5.1 LIKELIHOOD RATIO STATISTIC

The following theorem derives immediately from Theorem 2.8.2.

THEOREM 7.5.1. *Consider a likelihood ratio test (Section 7.3) with $\Theta = M \times N$, where M, N are open subsets of $\mathbb{R}^p, \mathbb{R}^q$ and denote $\theta = (\mu, \nu)$. Also let*

- $\Theta_0 = \{\mu = 0\} = \{\theta = (0, \nu), \nu \in N\}$
- $\Theta = \{\theta = (\mu, \nu), \mu \in M\, \nu \in N, \nu \neq 0\}$

We test the hypothesis $H_0 := \{\mu = 0\} = \{\theta \in \Theta_0\}$ against $H_1 := \{\mu \neq 0\} = \{\theta \in \Theta \setminus \Theta_0\}$. Assume that the conditions in Definition 2.8.1 hold. Then, approximately

$$2\left(l_x(\hat{\theta}) - l_x(\hat{\theta}_0)\right) \sim \chi^2_p$$

(p is the number of degrees of freedom that H_1 adds to H_0). It follows that the p-value of the likelihood ratio test can be approximated for large sample sizes by

$$p^* \approx 1 - \chi^2_p\left(2(l_x(\hat{\theta}) - l_x(\hat{\theta}_0))\right) \tag{7.33}$$

EXAMPLE: APPLICATION TO EXAMPLE 7.1 ON PAGE 151. Using Theorem 7.4.1 and Theorem 7.5.1 we find that

$$2lrs := N \ln\left(1 + \frac{SS2}{SS1}\right) \sim \chi^2_1$$

The corresponding p-values are:

```
Parameter Set 1 pchi2 = 0.0002854
Parameter Set 1 pchi2 = 0.02731
Parameter Set 1 pchi2 = 0.6669
```

They are all very close to the exact values (given by ANOVA).

7.5.2 APPLICATION TO NON PAIRED DATA, DIFFERENT VARIANCES

We show in this section how the asymptotic result may be useful when the hypothesis of same variance does not hold. Assume we are given two unpaired series of data, and we want to test whether they have the same mean.

The model is

$$X_i = \mu_1 + \epsilon_{1,i} \quad Y_j = \mu_2 + \epsilon_{2,j} \tag{7.34}$$

with $\epsilon_{i,j} \sim$ iid N_{0,σ^2}. We apply Section 7.3 and compute first the MLEs. For the unrestricted MLE $\hat{\theta} = (\hat{\mu}_1, \hat{\mu}_2, \hat{\sigma}_1, \hat{\sigma}_2)$ we find

- $\hat{\mu}_1 = \bar{x}$, $\hat{\mu}_2 = \bar{y}$,
- $\hat{\sigma}_1^2 = \frac{1}{m} S_{xx}$, $\hat{\sigma}_2^2 = \frac{1}{n} S_{yy}$,
- $l_{x,y}(\hat{\theta}) = cst - \frac{m}{2} \ln S_{xx} - \frac{n}{2} \ln S_{yy}$

(with $S_{xx} = \sum_i (x_i - \bar{x})^2$). The restricted MLE $\hat{\theta}_0 = (\hat{\mu}, \hat{\mu}, \hat{\sigma}'_1, \hat{\sigma}'_2)$ cannot be obtained explicitly. We have

$$l_{x,y}(\mu, \mu, \sigma'_1, \sigma'_2) = cst - \frac{m}{2} \ln \sigma'^2_1 - \frac{S_{xx} + m(\bar{x} - \mu)^2}{2\sigma'^2_1} - \frac{n}{2} \ln \sigma'^2_2 - \frac{S_{yy} + n(\bar{y} - \mu)^2}{2\sigma'^2_2} \quad (7.35)$$

By differentiating with respect to μ we find that

$$\hat{\mu} = \frac{S_1 \bar{x} + S_2 \bar{y}}{S_1 + S_2} \quad (7.36)$$

with $S_1 = m/\sigma'^2_1$ and $S_2 = n/\sigma'^2_2$. By substituting Equation (7.36) in Equation (7.35), we obtain the log-likelihood as a function of two variables σ'_1, σ'_2. We maximize it numerically to obtain $\hat{\sigma}'_1, \hat{\sigma}'_2$.

The likelihood ratio statistic is

$$T = \frac{1}{2} \left(l_{x,y}(\hat{\theta}) - l_{x,y}(\hat{\theta}_0) \right) \quad (7.37)$$

it can be computed once we know all values of MLEs. We know that, under the assumption that the means are equal, and asymptotically, $T \sim \chi^2_p$. Now $p = 1$ since there are 4 free parameters for θ and 3 for θ_0. Thus, the test for equality of means has rejection region of the form $C = \{T > \eta\}$ where $\chi^2_1(\eta) = 1 - \alpha$, i.e. $\eta = \xi^2$ where ξ is the $1 - \alpha/2$ quantile of the standard normal distribution.

QUESTION 7.5.1. *What is the p-value of the test ?* [8]

EXAMPLE 7.6: FRUITFLIES AGAIN. ([Weber-C11] Example 12.2.) Consider some other data series about the longevity of fruitflies:

Group	*Mean life (days)*	*standard deviation*
1	63.56	16.4522
4	56.76	14.9284

We would like to test equality of mean for groups 1 and 4. We first test for same variance and find $F = 1.21$. The rejection region is $\eta < F < \xi$ with $F_{m-1,n-1}(\eta) = \frac{\alpha}{2}$, $F_{m-1,n-1}(\xi) = 1 - \frac{\alpha}{2}$. At size $\alpha = 0.05$, we get $\eta = 0.44, \xi = 2.27$ so we accept the hypothesis of equal variance. Testing for equality of mean is thus done with ANOVA; we find $f = 2.3423$ and p-value $p = F_{m-1,n-1}(f) = 0.132$. Alternatively, we use a t-test and get $t = 1.53 < t^{-1}_{48}(0.975) = 2.01$ and we accept equality of means at size $\alpha = 0.05$.

---

[8] $$p^* = 1 - \chi^2_1(T) = 2(1 - N(\sqrt{T}))$$
where N is the standard normal distribution function. We reject equality of means when p^* is smaller than α.

Assume now that we have the same data, except for the number of samples which is now equal to 500. Let us repeat the analysis. The test for same variance gives $F = 1.21$ but now $\eta = 1.158$ (for $\alpha = 0.10$, or $\eta = 1.192$ for $\alpha = 0.05$). So at size 0.05 already we reject the hypothesis of same variance. We cannot apply ANOVA.

Zum Glück, n is large, so we apply the MLE asymptotics instead. We find

- Unrestricted model: $\hat{\mu}_1 = 63.56, \hat{\mu}_2 = 56.76,\ \hat{\sigma}_1 = 16.43574,\ \hat{\sigma}_2 = 14.91346$
- Restricted model: $\hat{\mu} = 59.8145,\ \hat{\sigma}'_1 = 16.8571,\ \hat{\sigma}'_2 = 15.2230$
- $T = 45.9 > 3.84$ thus we reject the hypothesis of equalities of means (at a size $\alpha = 0.05$).

7.5.3 PEARSON CHI-SQUARED STATISTIC AND GOODNESS OF FIT

We can apply the large sample asymptotic to goodness of fit tests as defined in Section 7.3.3. This gives a simpler way to compute the p-value, and allows to extend the test to the *composite goodness of fit* test, defined as follows.

COMPOSITE GOODNESS OF FIT Similar to Section 7.3.3, assume we are given n data points $x_1, ..., x_n$, generated from an iid sequence, and we want to verify whether their common distribution comes from a given family of distributions $F(|\theta)$ where the parameter θ is in some set Θ_0. We say that the test is composite because the null hypothesis has several possible values of θ. We compare the empirical histograms: we partition the set of values of X into *bins* B_i. Let $N_i = \sum_{k=1}^n 1_{\{B_i\}}(X_k)$ (number of observation that fall in bin B_i) and $q_i = \mathbb{P}_\theta\{X_1 \in B_i\}$. If the data comes from a distribution $F(|\theta)$ the distribution of N_i is multinomial $M_{n,\vec{q}(\theta)}$. The likelihood ratio statistic test is

H_0: N_i comes from a multinomial distribution $M_{n,\vec{q}(\theta)}$, with $\theta \in \Theta_0$

against

H_1: N_i comes from a multinomial distribution $M_{n,\vec{p}}$ for some arbitrary $\vec{p}$.

We now compute the likelihood ratio statistic. It is similar to the derivation in Section 7.3.3. Let $\hat{\theta}$ be the MLE of θ under H_0. $\hat{\theta}_0 = \vec{q}$. We find that the likelihood ratio statistic is

$$lrs = l_{H_1}(\vec{x}) - l_{H_0}(\vec{x}) = \sum_{i=1}^k n_i \ln \frac{n_i}{nq_i(\hat{\theta})} \tag{7.38}$$

The p-value is

$$\sup_{\theta \in \Theta_0} \mathbb{P}\left(\sum_{i=1}^k N_i \ln \frac{N_i}{nq_i} > \sum_{i=1}^k n_i \ln \frac{n_i}{nq_i(\hat{\theta})} \right) \tag{7.39}$$

where $\vec{N}$ has the multinomial distribution $M_{n,q(\vec{\theta})}$. It can be computed by Monte Carlo simulation as in the case of a simple test, but this may be difficult because of the supremum.

An alternative for large n is to use the asymptotic result in Theorem 7.5.1. It says that, for large n, under H_0, the distribution of $2lrs$ is approximately chi^2_m, with $m =$ the number of degrees of

freedom that H_1 adds to H_0. Here H_0 has k_0 degrees of freedom (where k_0 is the dimension of Θ_0) and H_1 has $I-1$ degrees of freedom (where I is the number of bins). Thus the p-value of the test is approximately

$$1 - \chi^2_{I-k_0-1}(2lrs) \tag{7.40}$$

where $\chi^2_{I-k_0-1}$ is the cdf of the chi-squared distribution with $I-k_0-1$ degrees of freedom.

EXAMPLE: IMPACT OF ESTIMATION OF (μ,σ). We want to test whether the data set on the right of Figure 7.3 has a normal distribution. We use a histogram with 10 bins. We need first to estimate $\hat{\theta} = (\hat{\mu}, \hat{\sigma})$.

1. Assume we do this by fitting a line to the qqplot. We obtain $\hat{\mu} = -0.2652, \hat{\sigma} = 0.8709$. The values of $nq_i(\hat{\theta})$ and n_i are:

```
7.9297    7.0000
11.4034    9.0000
18.0564   17.0000
21.4172   21.0000
19.0305   14.0000
12.6672   17.0000
 6.3156    6.0000
 2.3583    4.0000
 0.6594    3.0000
 0.1624    2.0000
```

The likelihood ratio statistic as in Equation (7.38) is $lrs = 7.6352$. The p-value is obtained using a χ^2_7 distribution ($m = 10-2-1$): $p1 = 0.0327$, thus we would reject normality at size 0.05.

2. It is not correct to simply fit (μ,σ) on the qqplot. The theory says that we should find (μ,σ) that maximizes the log likelihood of the model. This is equivalent to minimizing the likelihood ratio statistic $l_{H_1}(x) - l_{\mu,\sigma}(x)$ (note that the value of $l_{H_1}(x)$ is easy to compute). We do this with a numerical optimization procedure and find now $\hat{\mu} = -0.0725, \hat{\sigma} = 1.0269$. The corresponding values of $nq_i(\hat{\theta})$ and n_i are now:

```
 8.3309    7.0000
 9.5028    9.0000
14.4317   17.0000
17.7801   21.0000
17.7709   14.0000
14.4093   17.0000
 9.4783    6.0000
 5.0577    4.0000
 2.1892    3.0000
 1.0491    2.0000
```

Note how the true value of $\hat{\mu}, \hat{\sigma}$ provides a better fit to the tail of the histogram. The The likelihood ratio statistic is now $lrs = 2.5973$, which also shows a much better fit. The p-value, obtained using a χ^2_7 distribution is now $p1 = 0.6362$, thus we accept that the data is normal.

3. Assume we would ignore that (μ,σ) is estimated from the data, but would do as if the test were a simple goodness of fit test, with H_0 : "The distribution is $N_{-0.0725,1.0269}$"

instead of H_0 : "The distribution is normal". We would compute the p value using a χ^2_9 distribution ($m = 10 - 1$) and would obtain: $p2 = 0.8170$, a value larger than the true p-value. This is quite general: if we estimate some parameter and pretend it is a priori known, then we overestimate the p-value.

PEARSON CHI-SQUARED STATISTIC. In the case where n is large, $2\times$ the likelihood ratio statistic can be replaced by the *Pearson chi-squared statistic*, which has the same asymptotic distribution. It is defined by

$$pcs = \sum_{i=1}^{I} \frac{(n_i - nq_i(\hat{\theta}))^2}{nq_i(\hat{\theta})} \tag{7.41}$$

Indeed, when n is large we expect, under H_0 that $n_i - nq_i(\hat{\theta})$ is relatively small, i.e.

$$\epsilon_i = \frac{n_i}{nq_i(\hat{\theta})} - 1 \tag{7.42}$$

is small. An approximation of $2lrs$ is found from the second order development around $\epsilon = 0$:

$$\ln(1+\epsilon) = \epsilon - \frac{1}{2}\epsilon^2 + o(\epsilon^2) \tag{7.43}$$

and thus

$$\begin{aligned}
lrs &= \sum_i n_i \frac{n_i}{nq_i(\hat{\theta})} n \sum_i (1+\epsilon_i) q_i(\hat{\theta}) \ln(1+\epsilon_i) \\
&= n \sum_i \left(\epsilon_i - \frac{1}{2}\epsilon_i^2 + o(\epsilon_i^2)(1+\epsilon_i) q_i(\hat{\theta}) \right) \\
&= n \sum_i q_i(\hat{\theta}) \epsilon_i \left(1 - \frac{1}{2}\epsilon_i + o(\epsilon_i)(1+\epsilon_i) \right) \\
&= n \sum_i q_i(\hat{\theta}) \epsilon_i \left(1 + \frac{1}{2}\epsilon_i + o(\epsilon_i) \right) \\
&= n \sum_i q_i(\hat{\theta}) \epsilon_i + n \sum_i q_i(\hat{\theta}) \frac{1}{2}\epsilon_i^2 + n \sum_i o(\epsilon_i^2)
\end{aligned}$$

Note that $\sum_i q_i(\hat{\theta})\epsilon_i = 0$ thus

$$lrs \approx \frac{1}{2} pcs \tag{7.44}$$

The Pearson Chi-squared statistic was historically developed before the theory of likelihood ratio tests, which explains why it is commonly used.

In summary, for large n, the composite goodness of fit test is solved by computing either $2lrs$ or pcs. The p-value is $1 - \chi^2_{n-k_0-1}(2lrs)$ or $1 - \chi^2_{I-k_0-1}(pcs)$. If either is small, we reject H_0, i.e. we reject that the distribution of X_i comes from the family of distributions $F(|\theta)$.

SIMPLE GOODNESS OF FIT TEST. This is a special case of the composite test. In this case $m = n - 1$ and thus the p-value of the test (given in Equation (7.13) can be approximated for large n by $1 - \chi^2_{I-1}(2lrs)$ or $\chi^2_{I-1}(pcs)$. Also, the likelihood ratio statistic $\sum_{i=1}^{k} n_i \ln \frac{n_i}{nq_i}$ can be replaced by the Pearson-Chi-Squared statistic, equal to

$$\sum_{i=1}^{I} \frac{(n_i - nq_i)^2}{nq_i} \tag{7.45}$$

EXAMPLE: MENDEL'S PEAS, CONTINUATION OF EXAMPLE 7.4 ON PAGE 159. The likelihood ratio statistic is $lrs = 0.3092$ and we found by Monte Carlo a p-value $p^* = 0.9191 \pm 0.0458$. By the asymptotic result, we can approximate the p-value by $\chi^2_3(2lrs) = 0.8922$.

The Pearson Chi-squared statistic is $pcs = 0.6043$, very close to $2lrs = 0.618$. The corresponding p value is 0.8954.

7.5.4 TEST OF INDEPENDENCE

The same ideas as in Section 7.5.3 can be applied to a *test of independence*. We are given a sequence (x_k, y_k), which we interprete as a sample of the sequence (X_k, Y_k), $k = 1, ..., n$. The sequence is iid ((X_k, Y_k) is independent of $(X_k, Y_{k'})$ and has the same distribution). We are interested in knowing whether X_k is independent of Y_k.

To this end, we compute an empirical histogram of (X, Y), as follows. We partition the set of values of X [resp. Y] into I [resp. J] bins B_i [resp. C_j]. Let $N_{i,j} = \sum_{k=1}^{n} 1_{\{B_i\}}(X_k) 1_{\{C_j\}}(Y_k)$ (number of observation that fall in bin (B_i, C_j)) and $p_{i,j} = \mathbb{P}\{X_1 \in B_i \text{ and } Y_1 \in C_j\}$. The distribution of N is multinomial. The test of independence is

H_0: "$p_{i,j} = q_i r_j$ for some q and r such that $\sum_i q_i = \sum_j r_j = 1$"

against

H_1: "$p_{i,j}$ is arbitrary"

The MLE under H_0 is $\hat{p}^0_{i,j} = \frac{n_{i.}}{n} \frac{n_{.j}}{n}$ where $n_{i,j} = \sum_{k=1}^{n} 1_{\{B_i\}}(x_k) 1_{\{C_j\}}(y_k)$ and

$$\begin{cases} n_{i.} = \sum_j n_{i,j} \\ n_{.j} = \sum_i n_{i,j} \end{cases} \tag{7.46}$$

The MLE under H_1 is $\hat{p}^1_{i,j} = \frac{n_{i,j}}{n}$. The likelihood ratio statistic is thus

$$lrs = \sum_{i,j} n_{i,j} \ln \frac{n n_{i,j}}{n_{i.} n_{.j}} \tag{7.47}$$

To compute the p-value, we use, for large n, a χ^2_m distribution. The numbers of degrees of freedom under H_1 is $IJ - 1$, under H_0 it is $(I - 1) + (J - 1)$, thus $m = (IJ - 1) - (I - 1) - (J - 1) = (I - 1)(J - 1)$. The p-value is thus

$$p^* = \left(1 - \chi^2_{(I-1)(J-1)}\right)(2lrs) \tag{7.48}$$

As in Section 7.5.3, $2lrs$ can be replaced, for large n, by the Pearson Chi-squared statistic:

$$pcs = \sum_{i,j} \frac{\left(n_{i,j} - \frac{n_{i.}n_{.j}}{n}\right)^2}{\frac{n_{i.}n_{.j}}{n}} \tag{7.49}$$

EXAMPLE 7.7: BRASSICA OLERACEA GEMMIFERA. A survey was conducted at the campus cafeteria, where customers were asked whether they like Brussels sprouts. The answers are:

$i\backslash j$	Male	Female	*Total*
Likes	454	251	705
Dislikes	295	123	418
No Answer / Neutral	267	148	415
Total	1016	522	1538

We would like to test whether affinity to Brussels sprouts is independent of customer's gender.

Here we have $I = 3$ and $J = 2$, so we use a χ^2 distribution with $m = 2$ degrees of freedom. The likelihood ratio statistic and the p-value are

$$lrs = 2.6489, \quad p = 0.0707 \tag{7.50}$$

so we accept H_0, i.e. affinity to Brussels sprouts is independent of gender.

Note that the Pearson Chi-squared statistic is

$$pcs = 5.2178 \tag{7.51}$$

which is very close to $2lrs$.

7.6 OTHER TESTS

7.6.1 GOODNESS OF FIT TESTS BASED ON AD-HOC PIVOTS

In addition to the Pearson χ^2 test, the following two tests are often used. They apply to a continuous distribution, thus do not require quantizing the observations. Assume X_i, $i = 1, .., n$ are iid samples. We want to test H_0: the distribution of X_i is F against non H_0.

Define the empirical distribution $\hat{F}$ by

$$\hat{F}(x) := \frac{1}{n} \sum_{i=1}^{n} 1_{\{X_i \leq x\}} \tag{7.52}$$

Kolmogorv-Smirnov The pivot is

$$T = \sup_x |\hat{F}(x) - F(x)|$$

That the distribution of this random variable is independent of F is not entirely obvious, but can be derived easily in the case where F is continuous and strictly increasing, as follows. The idea is to change the scale on the x-axis by $u = F(x)$. Formally, define

$$U_i = F(X_i)$$

so that $U_i \sim U(0,1)$. Also

$$\hat{F}(x) = \frac{1}{n}\sum_i 1_{\{X_i \leq x\}} = \frac{1}{n}\sum_i 1_{\{U_i \leq F(x)\}} = \hat{G}(F(x))$$

where $\hat{G}$ is the empirical distribution of the sample U_i, $i = 1, ..., n$. By the change of variable $u = F(x)$, it comes

$$T = \sup_{u \in [0,1]} |\hat{G}(u) - u|$$

which shows that the distribution of T is independent of F. Its distribution is tabulated in statistical software packages. For a large n, its tail can be approximated by $\tau \approx \sqrt{-(\ln \alpha)/2}$ where $\mathbb{P}(T > \tau) = \alpha$.

Anderson-Darling Here the pivot is

$$A = n \int_{\mathbb{R}} \frac{\left(\hat{F}(x) - F(x)\right)^2}{F(x)(1 - F(x))} dF(x)$$

The test is similar to K-S but is less sensitive to outliers.

QUESTION 7.6.1. *Show that A is indeed a pivot.* [9]

EXAMPLE 7.8: FILE TRANSFER DATA. We would like to test whether the data in Figure 7.3 and its log are normal. We cannot directly apply Kolmogorov Smirnov since we do not know exactly in advance the parameters of the normal distribution to be tested against. An approximate method is to estimate the slope and intercept of the straight line in the qqplot. We obtain

```
Original Data
slope     =    0.8155
intercept =    1.0421

Transformed Data
slope     =    0.8709
intercept =   -0.2652
```

[9] Use the fact that $\hat{F}(x) = \hat{G}(F(x))$ and do the change of variable $u = F(x)$ in the integral.

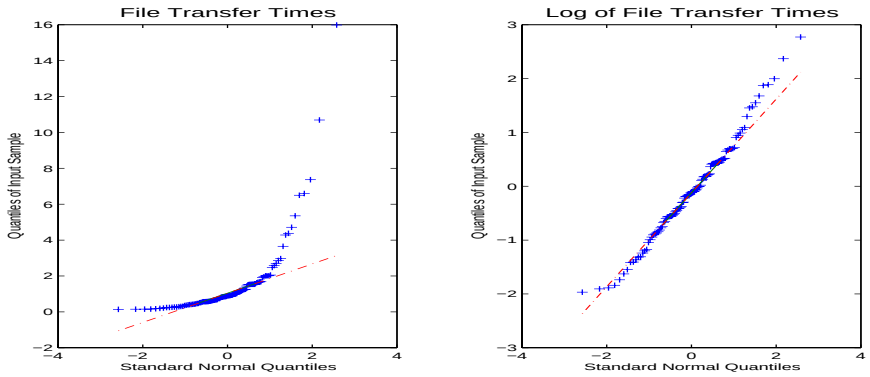


Figure 7.3: Normal qqplots of file transfer data and its logarithm.

For example, this means that for the original data we take for H_0: "the distribution is $N(\mu = 1.0421, \sigma^2 = 0.8155^2)$". We can now use the Kolmogorov-Smirnov test and obtain

```
Original Data
h =     1           p =     0.0000

Transformed Data
h =     0           p =     0.2964
```

Thus the test rejects the normality assumption for the original data and accepts it for the transformed data.

This way of doing is approximate in that we used estimated parameters for H_0. This introduces some bias, similar to using the normal statistic instead of student when we have a normal sample. The bias should be small when the data sample is large, which is the case here.

A fix to this problem is to use a variant of KS, for example the Lilliefors, or to use different normality tests such as Jarque Bera (see Example 8.1 on page 185) or Shapiro-Wilk. The Lilliefors test is a heuristic that corrects the p-value of the KS to account for the uncertainty due to estimation. In this specific example, with the Lilliefors test we obtain the same results as previously.

7.6.2 ROBUST TESTS

We give two examples of test that make no assumption on the distribution of the sample (but assume it is iid). They are *non parametric* in the sense that they do not assume a parameterized family of densities.

MEDIAN TEST The model is $X_i \sim$ iid with some distribution $F()$ with a density. We want to test

H_0: "the median of F is 0" against H_1: "unspecified"

A simple test is based on confidence interval, as mentioned in Section 7.3.2. Let $I(x)$ be a confidence interval for the median (Theorem 2.2.1). We reject H_0 if

$$0 \notin I(x) \tag{7.53}$$

This test is robust in the sense that it makes no assumption other than independence.

WILCOXON SIGNED RANK TEST. It is used to test a 0 median, for example when comparing paired experiments. Assume the data comes from an iid model $X_1, ...X_n$, with some unspecified, but symmetric, distribution. The null hypothesis is that the median is 0. The *Wilcoxon Signed Rank Statistic* is

$$W = \sum_{j=1}^{n} \text{rank}(|X_j|)\text{sign}(X_j)$$

where $\text{rank}(|X_j|)$ is the rank in increasing order (the smallest value has rank 1) and $\text{sign}(X_j)$ is -1 for negative data, $+1$ for positive, and 0 for null data. If the median is positive, then many values with high rank will be positive and W will tend to be positive and large. We reject the null hypothesis when $|W|$ is large.

It can be shown that the distribution of W under H_0 is always the same. It is tabulated and contained in software packages. For non small data samples, it can easily be approximated by a normal distribution. We now compute its mean and variance.

Under H_0 is

$$\mathbb{E}_{H_0}(W) = \sum_{j=1}^{n} \mathbb{E}_{H_0}(\text{rank}(|X_j|)\mathbb{E}_{H_0}(\text{sign}(X_j))$$

since under H_0 $\text{rank}(|X_j|)$ is independent of $\text{sign}(X_j)$. Thus $E_{H_0}(W) = 0$. The variance is

$$\mathbb{E}_{H_0}(W^2) = \sum_{j=1}^{n} \mathbb{E}_{H_0}(\text{rank}(|X_j|)^2\text{sign}(X_j)^2) = \sum_{j=1}^{n} \mathbb{E}_{H_0}(\text{rank}(|X_j|)^2)$$

since $\text{sign}(X_j)^2 = 1$. Now $\sum_j \text{rank}(|X_j|)^2 = \sum_j j^2$ is non random thus

$$\text{var}_{H_0}(W) = \sum_{j=1}^{n} \mathbb{E}_{H_0}(\text{rank}(|X_j|)^2) = \mathbb{E}_{H_0}(\sum_j \text{rank}(|X_j|)^2) = \sum_{j=1}^{n} j^2 = \frac{n(n+1)(2n+1)}{6}$$

EXAMPLE: PAIRED DATA.This is a variant of Example 7.2 on page 152. Consider again the reduction in run time due to a new compiler option, as given in Figure 2.3 on Page 17. We want to test whether the reduction is significant. We assume the data is iid, but not necessarily normal. The median test gives a confidence interval

$$I(x) = [2.9127; 33.7597]$$

which does not contain 0 so we reject H_0.

Alternatively, let us use the Wilcoxon Signed Rank test. We obtain the p-value

$$p = 2.3103e - 005$$

and thus this test also rejects H_0.

WILCOXON RANK SUM TEST AND KRUSKAL-WALLIS. It is used in unpaired experiments to test (H_0): the two samples come from the same distribution against (H_1) the distributions of the two samples differ by a location shift. It is assumed that the distributions have a density.

Let X_i^1, $i = 1...n_1$ and X_i^2, $i = 1...n_2$ be the two iid sequences that the data is assumed to be a sample of. The *Wilcoxon Rank Sum Statistic* R is the sum of the ranks of the first sample in the concatenated sample.

As for the Wilcoxon rank sum test, its distribution under the null hypothesis depends only on the sample sizes and can be tabulated or, for a large sample size, approximated by a normal distribution. Its mean is

$$\mathbb{E}_{H_0}(R_1) = \frac{n_1(n_1+n_2+1)}{2} \tag{7.54}$$

and its variance is

$$\text{var}_{H_0}(R_1) = \frac{n_1 n_2(n_1+n_2+1)}{2} \tag{7.55}$$

We reject H_0 when the rank sum statistic deviates largely from its expectation under H_0.

EXAMPLE 7.9: NON PAIRED DATA. The Wilcoxon rank sum test applied to Example 7.1 on page 151 gives the following p-values:

```
Parameter Set 1 p = 0.0002854
Parameter Set 2 p = 0.02731
Parameter Set 3 p = 0.6669
```

The results are the same as with ANOVA. H_0 (same distribution) is accepted for the 3rd data set only, at size$= 0.05$.

The *Kruskal-Wallis* test is a generalization of Wilcoxon Rank Sum to more than 2 non paired data series. It tests (H_0): the samples come from the same distribution against (H_1): the distributions may differ by a location shift.

7.7 REVIEW

7.7.1 SUMMARY

TBD

7.7.2 TESTS ARE JUST TESTS

1. The first test to do on any data is a visual exploration. In most cases, this is sufficient.
2. Testing for a 0 mean or 0 median is the same as computing a confidence interval for the mean or the median.
3. Tests work only if the underlying assumptions are verified (for example, iid normal samples). Even then, a test is just a test. Therefore, there are many different tests. Tests can be produced from any pivot: see for example the K-S test for goodness of fit.

4. Some tests work under a larger spectrum of assumptions (for example: even if the data is not normal). They are called robust tests. They should be preferred whenever possible.
5. Test whether the same variance assumption holds, otherwise, use robust tests or asymptotic results.
6. If you perform a large number of different tests on the same data, then the probability of rejecting H_0 is larger than for any single test. So, contrary to non-statistical tests, increasing the number of tests does not always improve the decision.

7.7.3 REVIEW QUESTIONS

QUESTION 7.7.1. *What is the critical region of a test ?* [10]

QUESTION 7.7.2. *What is a type 1 error ? A type 2 error ? The size of a test ?* [11]

QUESTION 7.7.3. *If a test says "do not accept H_0", can we conclude that H_1 is true ?* [12]

7.8 EXERCISES

USEFUL S-PLUS COMMANDS `aov, tables.model, summary, wilcox.test`

USEFUL MATLAB COMMANDS `anova1, anova2, anovan`: (analysis of variance, differ only in the format of the data model); `ttest`: student test; `ranksum, signrank`: Wilcoxon non-parametric tests.

EXERCISE 7.1 (ANOVA). *For which values of the ratio of variation explained by the model do we reject H_0 (i.e. accept H) ? Take the size $\alpha = 0.05$. What happens for large values of $N - k$? Numerical application: Values in [Weber-C11] Example 11.3.*

EXERCISE 7.2. *Consider Example [Weber-C11] Section 11.4 (χ^2-test). What is the p-value of the test ?*

EXERCISE 7.3. *We test H_0 against H_1, using a test of size α, where the rejection region has the form $\{T(X) > k_\alpha\}$. How do the results of the tests compare for $\alpha = 0.05$ and $\alpha = 0.10$? See for example [Weber-C11] Example 11.3.*

---

[10] Call x the data used for the test. The critical region C is a set C of possible values of the observation, such that when the event "$x \in C$" is true we reject H_0.

[11] A type 1 error occurs when the test says "do not accept H_0" whereas the truth is H_0. A type 2 error occurs when the test says "accept H_0" whereas the truth is H_1. The size of a test is $\sup_{\theta \text{ such that } H_0 \text{ is true}} \mathbb{P}_\theta(C)$ (= the worst case probability of a type 1 error).

[12] No, consider Example 7.2 on page 152. The first test says "do not accept $\mu = 0$", from which we cannot conclude that $H_1 = \{\text{"}\mu = 40\text{"}\}$ is true, since the second test says "do not accept $\mu = 40$". A test can only gives an indication that some hypothesis is wrong.

EXERCISE 7.4 (Tests). *We want to evaluate in detail whether Joe's idea, namely to see whether installing more base stations did bring some improvements (Questions 1.4.2 and 1.4.3), using simple tests.*

1. *Import the values of achieved throughput that were used to build Figure 1.1a, Figure 1.1b and Figure 1.1c by copying the file indicated in a complementary document. Note that these are* non-paired *data.*
2. *Using ANOVA, test whether Figure 1.1b is better than Figure 1.1a. Same question when comparing Figure 1.1a and Figure 1.1c on one hand, Figure 1.1b and Figure 1.1c on the other hand. What are the assumptions of ANOVA?*
3. *Same question using Wilcoxon rank-sum tests. What are the assumptions of the Wilcoxon rank-sum test?*
4. *Again using ANOVA, test whether Figure 1.1a, Figure 1.1b and Figure 1.1c have the same mean.*
5. *Same question using Kruskal-Wallis tests. What are the assumptions of the Kruskal-Wallis test?*

CHAPTER 8

LOAD GENERATION WITH SURGE

From PE4 we saw that a proper definition of load is key. We study here an example where the goal is to study the performance of a network and a web server. We study the load generator SURGE developed at Boston University, which, to my knowledge, is the most sophisticated one at the time of writing.

The principle of a load generator are:

- characterize important aspects of the load; produce a stochastic model which reproduces them
- implement an emulator that produces instances of the process, using a random number generator, like a simulator does. It generates real traffic, unlike a simulator.

An important aspect is the choice of distributions of single random variables used to model the load. We also discuss an important feature called *Heavy Tail*.

Contents

8.1 DISTRIBUTIONS

We review a number of attributes of distributions, which is useful in making the right choice.

[McLaughlin97] gives a compendium of distributions. See also [NIST] Section 1.3.6 for an illustration of distributions.

8.1.1 SCALE, LOCATION AND SHAPE PARAMETERS

A distribution can always be scaled and translated, by a transformation of the form $y = \frac{x-m}{s}$. Physically, this corresponds to a change of origin and units. This gives two degrees of freedom called location and scale. In contrast to scale and location, distributions have a shape, which make them unique. The modeler's talent is to pick a distribution that has a shape consistent with the data.

A normal distribution $N(\mu, \sigma^2)$ has location$= \mu$, scale$= \sigma$ and always has the same nice, symmetric bell shape. Other distributions such as Gamma, Beta or Weibull have a shape which depends on the parameter (Figure 8.1). The Weibull distribution has density

$$f(x) = \frac{c(x-a)^{c-1}}{b^c} e^{-\left(\frac{x-a}{b}\right)^c} 1_{\{x>a\}} \tag{8.1}$$

a is a location, and b a scale, parameter.

Consider as another example the effect of a Box-Cox transformation. Let X be a random variable such that $Y := b_s(X) \sim N(\mu, \sigma^2)$ (the distribution of X is *Box-Cox-normal*, with shape parameter s). For $s = 0$ we have the *log-normal* distribution, whose density is (Figure 8.2)

$$f(x) = \frac{1}{x\sqrt{2\pi\sigma^2}} \exp -\frac{(\log x - \mu)^2}{2\sigma^2} \tag{8.2}$$

Here μ and σ both influence shape, mean and variance.

$\frac{cx^{c-1}}{b^c} \exp -(x/b)^c 1_{\{x>0\}}$

QUESTION 8.1.1. *If $X \sim$log-normal, say which one is correct: (1)X is the exponential of a normal random variable; (2) X is the logarithm of a normal random variable.* [1]

[1] (1).

QUESTION 8.1.2. *Give the density of a scaled and translated log-normal distribution. If we require the random variable to have support in* $(0, +\infty)$*, which scalings and translations are possible ?* [2]

QUESTION 8.1.3. *What is the density of a box-cox-normal random variable ?* [3]

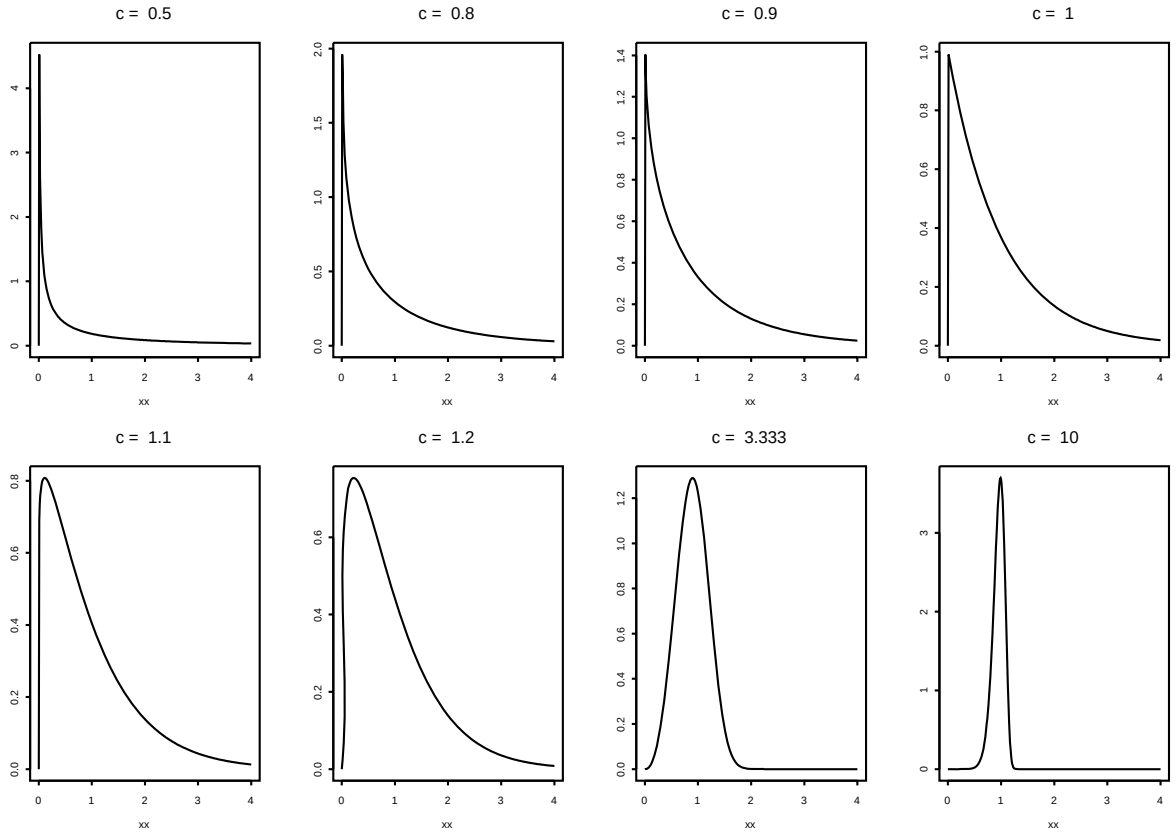


Figure 8.1: Shape of the Weibull distribution for $a = 0$, $b = 1$ and various values of c.

8.1.2 SKEWNESS AND KURTOSIS

Skewness and *Kurtosis* qualify the shape of a distribution that has finite moments up to order 4. They are based on *cumulants*, defined as follows. The *cumulant generating function* of the

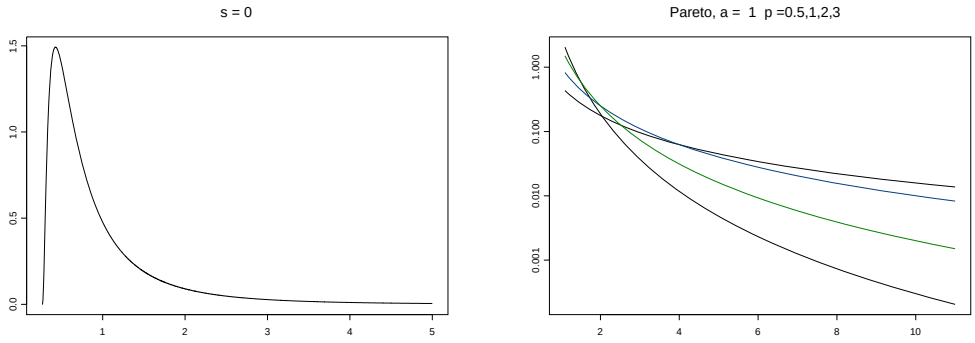
[2] Let $Y = sX + m$ where X is log-normal. The density of Y is

$$\frac{1}{(y-m)\sqrt{2\pi\sigma^2}} \exp - \frac{(\log(y-m) - \mu + \log s)^2}{2\sigma^2}$$

thus the scaling by s is equivalent to changing μ and is not needed. Only location is required. However if we require that Y takes values in $(0, +\infty)$, location is excluded; μ and σ are the only two parameters of a log-normal RV.

[3] For $s = 0$, it is the log-normal density; else

$$sx^{s-1} \exp - \frac{1}{2} \left(\frac{x^s - 1 - s\mu}{s\sigma} \right)^2$$



(a) Log-normal (b) Pareto

Figure 8.2: Shape of the Log-Normal distribution for $\mu = 0$, $\sigma = 1$ and of the Pareto distribution for $a = 1$ and $p = 0.5, 1, 2$ and 3.

distribution of a real random variable X is defined by

$$cgf(s) := \log \hat{f}(-s) = \log \mathbb{E}\left(e^{sX}\right)$$

($\hat{f}$ is the *Laplace transform*). Assume that $\mathbb{E}(e^{s_0|X|}) < \infty$ for some s_0 so that the above is well defined around $s = 0$. This also implies that all moments are finite. Then, by a Taylor expansion:

$$cgf(s) = \kappa_1 s + \kappa_2 \frac{s^2}{2} + \kappa_3 \frac{s^3}{3!} + ... + \kappa_k \frac{s^k}{k!} + ...$$

The coefficient κ_k is, by definition, the cumulant of order k. We have $\kappa_k = \frac{d^k}{ds^k} cgf(0)$. The first four cumulants are :

$$\begin{cases} \kappa_1 = \mathbb{E}(X) \\ \kappa_2 = \mathbb{E}\left(X - \mathbb{E}(X)\right)^2 = \text{var}(X) \\ \kappa_3 = \mathbb{E}\left(X - \mathbb{E}(X)\right)^3 \\ \kappa_4 = \mathbb{E}\left(X - \mathbb{E}(X)\right)^4 - 3\text{var}(X)^2 \end{cases} \tag{8.3}$$

For the normal distribution N_{μ,σ^2}, $cgf(s) = \mu s + \frac{\sigma^2}{2} s^2$ thus all cumulants of order $k \geq 3$ are 0.

SKEWNESS INDEX κ_3 is called skewness. The *skewness index* is

$$\gamma_1 := \kappa_3 / \kappa_2^{3/2} = \kappa_3 / \sigma^3$$

The skewness index is insensitive to changes in scale (by a positive factor) or location. For a density which is symmetric around its mean, $\kappa_{2k+1} = 0$; γ_1 can be taken as a measure of asymmetry of the distribution. When $\gamma_1 > 0$ the distribution is right-skewed, and vice-versa. If ϕ is convex, then $\phi(X)$ has greater skewness index than X.

KURTOSIS INDEX κ_4 is called Kurtosis. The *Kurtosis index* is

$$\gamma_2 := \kappa_4 / \kappa_2^2 = \kappa_4 / \sigma^4$$

The Kurtosis index is insensitive to changes in scale or location. It is used to measure departure from the normal distribution. When $\gamma_2 > 0$, the distribution has a sharper peak around the mean and heavier tail; when $\gamma_2 < 0$, it has a flatter top and decays more abruptly.

QUESTION 8.1.4. *Show that the Kurtosis index is also given by* $\gamma_2 = \frac{\mathbb{E}(X-\mathbb{E}(X))^4}{\sigma^4} - 3$ [4]

QUESTION 8.1.5. *Guess the index of skewness and the sign of the Kurtosis index for a uniform distribution. Check in [McLaughlin97]* [5]

QUESTION 8.1.6. *Show that* $var((X-\mathbb{E}(X))^2) = \kappa_4 + 2\sigma^4 = \sigma^4(2+\gamma_2)$. *Show that* $\gamma_2 \geq -2$. [6]

QUESTION 8.1.7. *Show that the* k*th cumulant of the convolution of two distributions is the sum of the* k*th cumulants* [7]

QUESTION 8.1.8. *Let* $Y = s(X-m)$. *Relate the cumulants of* X *and* Y. [8]

JARQUE-BERA. The *Jarque-Bera* statistic is used to test whether an iid sample comes from a normal distribution. It is equal to $\frac{n}{6}\left(\hat{\gamma}_1^2 + \frac{\hat{\gamma}_2^2}{4}\right)$, the distribution of which is asymptotically χ_2^2 for large sample size n. In the formula, $\hat{\gamma}_1$ and $\hat{\gamma}_2$ are the sample indices of skewness and kurtosis, obtained by replacing expectations by sample averages in Equation (8.3).

EXAMPLE 8.1: APPLICATION TO EXAMPLE 7.8 ON PAGE 175. We would like to test whether the data in Example 7.8 on page 175 and its transform are normal.

```
Original Data          h = 1          p =     0
Transformed Data       h = 0          p =    0.2964
```

The conclusions are the same as in Example 7.8 on page 175, but for the original data the normality assumption is clearly rejected, whereas it was borderline in Example 7.8 on page 175.

8.1.3 POWER LAWS, ZIPF'S LAW AND PARETO DISTRIBUTIONS

The three terms in the title are often read in Internet related performance studies. They are quite related, as we see now.

Power Laws. This is the name for relations of the form

$$y = ax^b$$

i.e., affine relations in xy-log scales: $\log y = b \log x + a$.

[4] Simple calculus.

[5] $\gamma_1 = 0$ and $\gamma_2 = -1.2$.

[6] 1. Simple calculus. 2.The previous is always ≥ 0.

[7] The Laplace transform is the sum of the Laplace transforms.

[8] $\kappa_1(Y) = \kappa_1(X) + m$ and for $k \geq 2$: $\kappa_k(Y) = s^k \kappa_k(X)$. Cumulants other than the mean κ_1 are invariant by translation, so we can study them for centered distributions only.

PARETO DISTRIBUTION. The *Pareto* distribution has density (Figure 8.2)

$$f(x) = pa^p x^{-(p+1)} 1_{\{x>a\}} \tag{8.4}$$

Its complementary distribution $\mathbb{P}(X > x) = \left(\frac{a}{x}\right)^p$ is a power law. Its mean is $\frac{ap}{p-1}$ for $p > 1$ and its variance is $\frac{a^2 p}{(p-2)(p-1)^2}$ for $p > 2$.

QUESTION 8.1.9. *Give a method to generates a random sample of k values from the Pareto distribution with $a = 1$ and index p.* [9]

ZIPF'S LAW. *Zipf's law*, in our context, means that the popularity of an object (for example: a file requested on a server; a server) is approximately inversely proportional to its rank. It has been observed in some cases, and has received much attention.

Formally, call θ_j the probability that object j is selected, and let $\theta_{(1)} \geq \theta_{(2)} \geq ...$ be the collection of θs in decreasing order. Zipf's law means

$$\theta_{(j)} \approx \frac{k}{j}$$

where k is some constant.

Now we show the relation to a Pareto distribution. Assume that we draw the θs at random (as we do in a load generator) by obtaining some random value X_i for object i, and letting $\theta_i = X_i/(\sum_i X_i)$. Assume that the number of objects is large and X_i's marginal distribution is some fixed distribution on $\mathbb{R}^+$, with complementary distribution function $G(x)$. Let $X_{(n)}$ be the reverse order statistic, i.e. $X_{(1)} \geq X_{(2)} \geq ....$ We would like to follow Zip's law, i.e., for some constant c:

$$X_{(n)} \approx \frac{c}{n} \tag{8.5}$$

Now let us look at the empirical complementary distribution $\hat{G}$; it is obtained by putting a point at each X_i, with probability $1/N$, where N is the number of objects. More precisely, let us define it by

$$\hat{G}(x) = \frac{1}{N} \sum_{i=1}^{N} 1_{\{X_i \geq x\}}$$

Thus $\hat{G}(X_{(n)}) = n/N$. Combine with Equation (8.5): Zipf's law mandates that, at every point $x = X_{(n)}$ where we have some data:

$$\hat{G}(x) \approx \frac{k}{x}$$

with $k = c/N$. This means that the empirical distribution of X_i is Pareto with tail index $p = 1$.

In other words, Zipf's law can be interpreted as follows. The probability of choosing object i is itself a random variable, obtained by drawing from a Pareto distribution with tail index $b = 1$, then re-scaling to make the probabilities sum to 1.

[9] Let G be the complementary distribution of Pareto and U uniform on $[0, 1]$; then $G^{-1}(U) \sim$ Pareto: thus $X := \frac{1}{U^{1/p}}$ where U is uniform on $[0, 1]$.

8.1.4 SURVIVAL FUNCTION

A further element to determine which distribution to use is the survival function $h(x) := \frac{\mathbb{P}(X \geq x+\delta)}{\mathbb{P}(X \geq x)}$, where δ is fixed. Interpret X as the duration of a session; then $h(x)$ is the conditional probability that a session, known to have lasted at least x, will live for at least δ more time units.

We compute an asymptotic of $h(x)$, for large x, for Pareto and Weibull. We find, for Pareto

$$h(x) = \left(1 + \frac{\delta}{x}\right)^{-b} \to 1$$

and for Weibull

$$\begin{cases} h(x) \to 0 \text{ for } c > 1 \\ h(x) \to e^{-\frac{\delta}{b}} \text{ for } c = 1 \\ h(x) \to 1 \text{ for } c < 1 \end{cases}$$

Thus for power laws and sub-exponential decay ($c < 1$) the survival function gets close to 1. In contrast, for hyper-exponential decay ($c > 1$) the survival function gets close to 0. And for exponential decay, it converges to a constant < 1 (memoriless property). Think about what it means if X is the level of a flood...

8.1.5 FINDING A DISTRIBUTION THAT FITS SOME CONSTRAINTS

Maximum entropy. to be done

8.1.6 FITTING A DISTRIBUTION

An empirical tool is the qqplot, discussed in Section 2.4.1. It works under the assumption that the sample comes from a common distribution, and is large enough for the empirical distribution to converge to the theoretical one. It does not require the sample to be independent, as long as convergence to the common distribution does occur.

A qqplot is normally done with respect to a standardized distribution (scale =1 and location =0); the actual scale and location are read from the slope and intercept of a regression line, assuming the qq-plot is close to a straight line.

If the sample is known to be iid, formal tests as in Section 7.6.1 can be used.

FITTING LARGE DATA SETS. For very large data sets, it is often reported that tests of fit fail, whereas empirical histograms show a good fit. There are several interpretations to this.

1. a real data set never exactly fits a given, simple distribution. For large data sets, the test becomes accurate and thus rejects the hypothesis of a fit.
2. lack of fit may be related to absence of stationarity: the distribution is no longer the same at the beginning and the end of the measurement period
3. the data does not come from an independent sample. This is almost invariably true in practice, and is sufficient to explain failures of tests. Jan Beran [Bera94-book] reports that long lasting correlation in the data set explains that the level of significance of the classical tests may get close to 1 (instead of α), which makes the test meaningless. We will see an example in Chapter 9 of how to handle this in some cases.

For very large data sets, the problem can be avoided by sub-sampling the data according to a Poisson or Bernoulli process. This will remove the problem if it is due to correlations and the sampling results in few, far apart data points.

8.2 HEAVY TAILS

8.2.1 DEFINITION

Distributions such as $N(\mu, \sigma)$ have a density that decays very fast; thus, large values are very rare. In fact, the normal distribution is often taken to model data with bounded support. In contrast, the log-normal distribution does not decay as fast. We say that log-normal has a *fat tail*. The Pareto distribution is even more pronounced and is said to have *Heavy Tail*, see Figure 8.2.1.

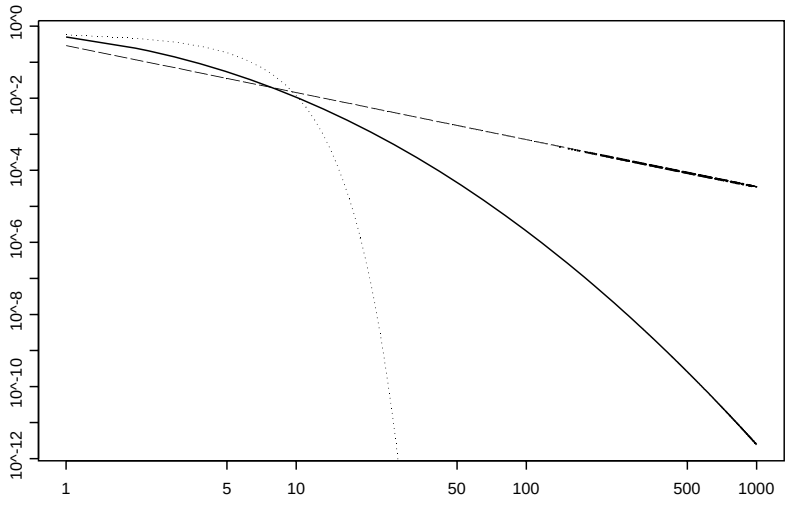


Figure 8.3: $P(x > x)$ versus x on log-log scales, when X is normal (dots), log-normal (solid) or Pareto (dashs). The three distributions have same mean and 99%-quantile.

HEAVY TAIL. We use the following definition (there are more general ones). We say that the distribution F is heavy tailed with index $0 < p \leq 2$ if there is some constant k such that, for large x:

$$1 - F(x) \sim \frac{k}{x^p} \tag{8.6}$$

Here $f(x) \sim g(x)$ means that $f(x) = g(x)(1 + \epsilon(x))$, with $\lim_{x\to\infty} \epsilon(x) = 0$.

A heavy tailed distribution has an infinite variance, and for $p \leq 1$ an infinite mean.

- The Pareto distribution with exponent p is heavy tailed with index p if $0 < p \leq 2$.
- The log-normal distribution is not heavy tailed (its variance is always finite).
- The Cauchy distribution (density $\frac{1}{\pi(1+x^2)}$) is heavy tailed with index 1.

The central limit theorem does not apply to distributions with infinite variance. Instead, if X_i are iid, heavy tailed with index p, then there exist constants c_n and d_n such that

$$\frac{1}{c_n}\sum_{i=1}^{n} X_i + d_n \underset{n \to \infty}{\overset{\text{distrib}}{\to}} S_p$$

where S_p has a "p-stable" distribution. p-stable distributions, for $p \leq 2$, constitute a family of distributions with the following property: if X_i are iid and p-stable then $\frac{1}{n^{\frac{1}{p}}}(X_1 + ... + X_n)$ has the same distribution as the X_is, shifted by some number d_n. The 2-stable distributions are the normal ones. For $p < 2$, p-stable distributions exist and are defined by 3 parameters (in addition to p), called location, scale and skewness. For $p < 2$, stable distributions are either constant or heavy tailed, and p is precisely the heavy tail index. Stable distributions that are not constant have a continuous density, which it is not known explicitly, in general. In contrast, their characteristic functions are known explicitly, see [Crovella99-Method] and [Samorodnistky94-Book]. The Cauchy distribution is 1-stable; Pareto is not stable. Figure 8.5 illustrates the convergence of a sum of iid Pareto random variables.

More precisely, the p-stable distribution with location=μ, skewness=β and scale=σ is defined by its *characteristic function* $\phi(\omega) := \mathbb{E}(e^{i\omega X})$ [Samorodnistky94-Book]. For $p \neq 1$:

$$\phi(\omega) = \exp\left[-\sigma^p|\omega|^p\left(1 - i\beta(\text{sgn}(\omega)\tan\frac{p\pi}{2}\right) + i\mu\omega\right]$$

and for $p = 1$:

$$\phi(\omega) = \exp\left[-\sigma|\omega|\left(1 + \frac{2i\beta}{\pi}\text{sgn}(\omega)\ln|\omega|\right) + i\mu\omega\right]$$

where $\text{sgn}(\omega) = 1$ if $\omega > 0$, $\text{sgn}(0) = 0$, and $\text{sgn}(\omega) = -1$ if $\omega < 0$

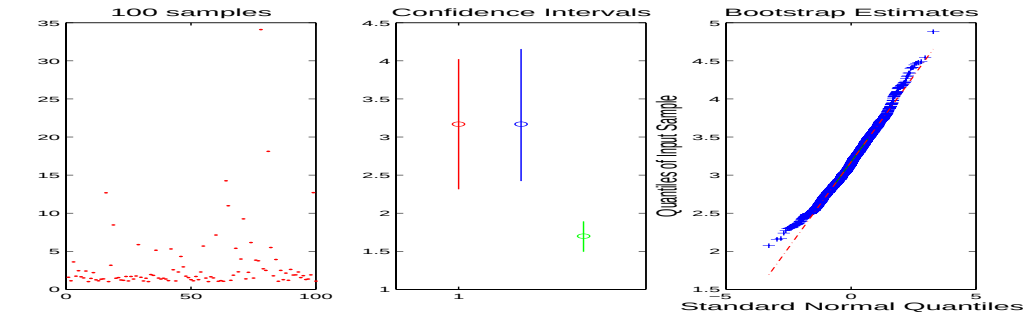
EXAMPLE 8.2: PARETO DISTRIBUTION. We use the Pareto distribution on $[1, +\infty)$ defined by its cdf equal to $F(c) := \mathbb{P}(X > c) = \frac{1}{c^p}$ with $p = 1.25$ (its mean is $= 5$ and it is heavy tailed). Assume we would not know that it comes from a heavy tailed distribution and would like to use the asymptotic result in Theorem 2.3.2 to compute a confidence interval for the mean. We verify convergence to the normal distribution and find on Figure 8.4 that the asymptotic regime does not hold. In contrast, the confidence interval for the median is perfectly correct.

QUESTION 8.2.1. *For which parameters is Weibull heavy tailed ?* [10]

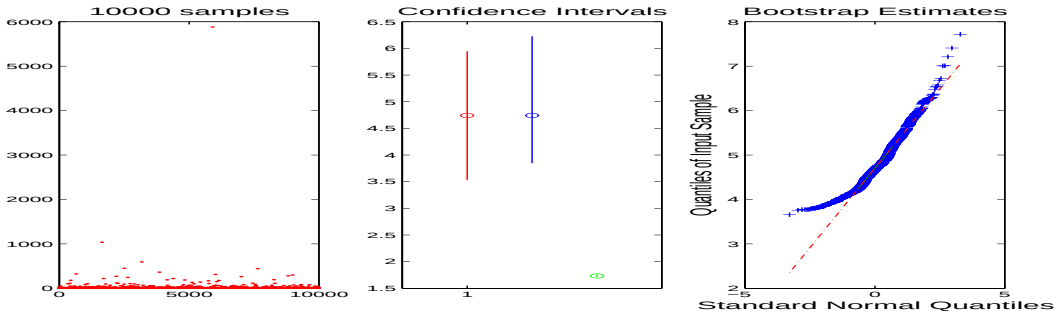
8.2.2 DISCUSSION

THE IMPORTANCE OF THE SECOND MOMENT. Heavy tail means that very large values are not too rare. This is called by Mandelbrot the *Noah effect* (where a large value is a flood). We further illustrate the concept in our context. Consider a server that receives requests for downloading files. Assume the requests arrival times form a Poisson process, and the requested file sizes are iid $\sim F$ where F is some distribution. This is a simplified model, but it will make the point.

[10] None



(a)



(b)

Figure 8.4: (a) Left: Artificially generated sample of 100 values from the Pareto distribution with exponent $p = 1.25$. Center: confidence intervals for the mean computed from Theorem 2.3.2 (left) and the bootstrap percentile estimate (center), and confidence interval for the median (right). Right: qqplot of 999 bootstrap replicates of the mean. The qqplot shows deviation from normality, thus the confidence interval given by Theorem 2.3.2 is not correct. Note that in this case the bootstrap percentile interval is not very good either, since it fails to capture the true value of the mean ($= 5$). In contrast, the confidence interval for the median does capture the true value ($= 1.74$). (b) Same with 10000 samples. The true mean is now within the confidence interval, but there is still no convergence to normality.

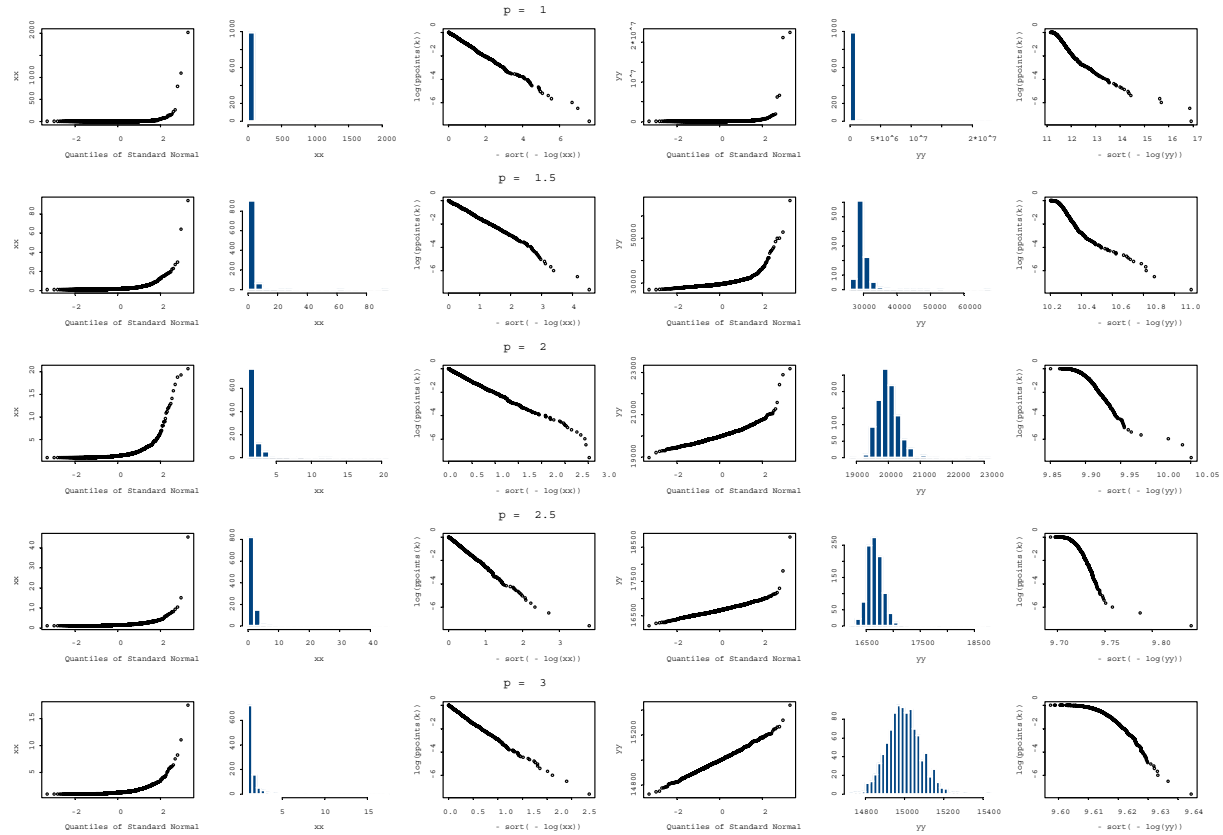


Figure 8.5: Aggregation a sum of iid Pareto random variables ($a = 1$, $p \in \{1, 1.5, 2, 2.5, 3\}$) (simulation in S). On every row: The first three diagrams show the empirical distribution (normal qq-plot, histogram, complementary distribution) of one sample of $n_1 = 10^4$ iid Pareto random variables. The last three show similar diagrams for a sample $(Y_j)_{1 \leq j \leq n}$ of $n = 10^3$ aggregated random variables: $Y_j = \frac{1}{n_1} \sum_{i=1}^{n_1} X_j^i$, where $X_j^i \sim$ iid Pareto. The figure illustrates that for $p < 2$ there is no convergence to a normal distribution, and for $p \geq 2$ there is. It also shows that for $p \geq 2$ the power law behaviour disappears by aggregation, unlike for $p < 2$. Note that for $p = 2$ X_i is heavy tailed but there is convergence to a normal law.

We assume that the server has a unit capacity, and that the time to serve a request is equal to the requested file size. This again is a simplifying assumption, which is valid if the bottleneck is a single, FIFO I/O. From Chapter 6, the mean response time of a request is given by the Pollaczek-Khintchine formula

$$R = \rho + \frac{\rho^2(1+\frac{\sigma^2}{\mu^2})}{2(1-\rho)}$$

where: μ is the mean and σ^2 the variance, of F (assuming both are finite); ρ is the utilization factor (= request arrival rate $\times\mu$). Thus the response time depends not only on the utilization and the mean size of requests, but also on the coefficient of variation $C := \sigma/\mu$. As C grows, the response times goes to infinity. Thus it is vital to capture the second moment. If the real data supports the hypothesis that F is heavy tailed, then the average response time is likely to be high and the estimators of it are unstable.

HEAVY TAIL IN PRACTICE Heavy tail is an asymptotic definition. Since, in practice, all data sets are finite, it is impossible to have a firm answer from statistical inference. In particular, it is often difficult to make a practical difference between log-normal and Pareto. Now since heavy tails introduce much theoretical and practical difficulties, one will often try to avoid heavy tail models.

However, we should be guided by Occam and Dijksta's principle, also called principle of Parsimony in this context. If data is explained by one simple heavy tailed model with few parameters, as well as by a non-heavy tail model with many parameters, then the heavy tail model should be preferred. This is the case of some aspects of SURGE.

To make a simplistic comparison, the use of heavy tail distributions is similar to saying, in optics, that the distance to a remote object is infinite. This is obviously wrong, but if it leads to simpler computations, then it should be used.

QUESTION 8.2.2. *If the tail of the distribution of X follows a power law, can you conclude that X is heavy tailed ?* [11]

8.2.3 TESTING HEAVY TAIL

There is no simple, rigorous test, but there are the following heuristics.

- Plot the empirical distribution in log-log scale and look for a linear relationship, the slope of which will give the exponent p in Equation (8.6). See Figure 1 in [Crovella99-Method].
- The *Hill Plot* and *Hill estimators* are based on the hypothesis that the distribution is Pareto. It has many difficulties in practice. See [Cappe02-SPM] for details and Figure 13 in [Crovella99-Method].
- A tool by Crovella and Taqqu (`aest`) uses the scaling properties and the central limit convergence to stable distributions. Consider X_i iid and heavy tailed, with index p. Call $X_i^{(m)}$ the aggregate sequence, where observations are grouped in bulks of m:

$$X_i^{(m)} := \sum_{j=(i-1)m+1}^{im} X_j$$

[11] No, only if the exponent of the tail is ≤ 2.

For large m_1, m_2, by the weak limit mentioned earlier, we should have approximately the distribution equalities

$$\frac{1}{m_1^{\frac{1}{p}}} X_i^{(m_1)} \sim \frac{1}{m_2^{\frac{1}{p}}} X_j^{(m_2)} \tag{8.7}$$

The idea is now to plot the empirical complementary distributions of $X_i^{(m)}$ for various values of m (Figure 2 in [Crovella99-Method]). Further, the deviation between two curves of the plot is analyzed by means of horizontal and vertical deviations δ and τ as explained in Figure 8.6. We have $\delta = \log x_2 - \log x_1$. By Equation (8.7), we have $x_2 = (m_2/m_1)^{1/p} x_1$ thus

$$\delta = \frac{1}{p} \log \frac{m_2}{m_1}$$

Also, if X_i is heavy tailed, and m is large, then $X_i^{(m)}$ is approximately stable. Thus, if m_2/m_1 is an integer, the distribution of $X_j^{(m_2)}$ (which is a sum of $X_i^{(m_1)}$) is the same as that of $(m_2/m_1)^{1/p} X_i^{(m_1)}$. We should thus have

$$\tau = \log \mathbb{P}(X_i^{(m_2)} > x_1) - \log \mathbb{P}(X_i^{(m_1)} > x_1) \approx \log \frac{m_2}{m_1}$$

The method in `aest` consists in use only the points x_1 where the above holds, then, at such points, estimate p by

$$\hat{p} = \frac{1}{\delta} \log \frac{m_2}{m_1}$$

Then the average of these estimates is used. See Figures 5, 7, 8 and 13 in [Crovella99-Method] for an illustration to some data set.

- Downey proposed in [Downey01-IMW] a test for distinguishing between Pareto tails (hypothesis H_0) and non Pareto tails. It is based on an estimation of the curvature of the complementary distribution (an estimate of the second derivative). If the curvature is large, the Pareto assumption is rejected. See also Exercise 8.5.

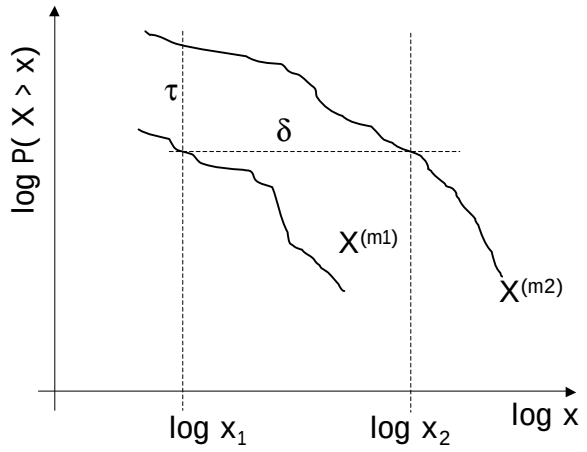


Figure 8.6: Deviations used in the `aest` tool.

8.3 THE WORKLOAD GENERATOR SURGE

This section explains the paper [Barford98-Sigmetrics].

8.3.1 IMPORTANT ASPECTS OF THE LOAD

The following aspects are believed to capture the important characteristics of web traffic.

- The volume on the number of users. In SURGE, this corresponds to the load being generated by atomic entities called *User Equivalents* (UEs). The generated load is an integer number of UEs, each implemented as an independent thread of execution, on one or several machines
- Traffic generated by one UE satisfies a set of constraints on the arrival process, the distribution of request sizes and the correlation of successive requests to the same object, as described below.

The values of the distributions were found by Barford and Crovella [Barford98-Sigmetrics] by fitting measured values.

1. One UE alternates between ON-object periods and "Inactive OFF periods". Inactive OFF periods are iid with a Pareto distribution (Table 8.1).
2. During an ON-object period, a UE sends a request with embedded references. Once the first reference (URL1) is received, there is an "Active OFF period", then the request for the second reference is sent, and so on, until all embedded references are received. There is only one TCP connection at a time per UE, and one TCP connection for each reference (an assumption that made sense with early versions of HTTP).
3. The active OFF times are modelled as iid random variables with Weibull distributions.
4. The number of embedded references is modelled as a set of iid random variables, with a Pareto distribution.

The references are viewed as requests for downloading files. The model is that there is a set of files labeled $i = 1, ..., I$, stored on the server. File i has two attributes: size x_i and request probability θ_i. The distribution of attributes has to satisfy the following conditions.

5. The distribution $H(x)$ of file sizes is a combination of Lognormal and Pareto (Table 8.1).
6. θ_i satisfy Zipf's law
7. The distribution $F(x)$ of requested file sizes satisfy a Pareto distribution (Table 8.1).

There is a relation between those three constraints, which we derive now. Let $I(t)$ be the random variable that gives the index i of the tth file requested. Thus $F(x) = \mathbb{P}(x_{I(t)} = x)$. We can assume that the allocation of file sizes and popularities is done in a preliminary phase, and is independent of $I(t)$. Thus

$$F(x) = \sum_j \mathbb{P}(I(t) = j) 1_{\{x_j \leq x\}} = \sum_j \theta_j 1_{\{x_j \leq x\}} \tag{8.8}$$

Let $x_{(1)} = x_{(2)} = ...$ be the file sizes sorted in increasing order, and let $z(n)$ be the index of the nth file in that order. z is a permutation of the set of indices, such that $x_{(n)} = x_{z(n)}$. By specializing Equation (8.8) to the actual values $x_{(m)}$ we find, after a change of variable $j = z(n)$

$$F(x_{(m)}) = \sum_j \theta_j 1_{\{x_j \leq x_{(m)}\}} = \sum_n \theta_{z(n)} 1_{\{x_{(n)} \leq x_{(m)}\}}$$

thus

$$F(x_{(m)}) = \sum_{n=1}^{m} \theta_{z(n)} \tag{8.9}$$

which gives a constraint between the θ_is and x_is.

The file request references $r(t)$, $t = 1, 2, ...$ are constrained by their marginal distribution (defined by θ_i). Here, we cannot assume that $r(t)$ is an iid sequence, as there is some evidence of correlation in the series (see Chapter 9). The condition taken here is is:

8. For any file index i, define $T_1(i) < T_2(i) < ...$ the successive values of $t \in \{1, 2, ...\}$ such that $i = r(t)$. Assume that $T_{k+1}(i) - T_k(i)$ come from a common distribution, called "temporal locality". The authors find it lognormal, with parameters as indicated in Table 8.1.

QUESTION 8.3.1. *Which of the distributions used in Surge are heavy tailed ?* [12]

8.3.2 BUILDING A PROCESS THAT SATISFIES ALL CONSTRAINTS

It remains to build a generator that produces a random output conformant to all constraints. Constraints 1 to 4 are straightforward to implement, with a proper random number generator. The inactive OFF periods, active OFF periods and number of embedded references are implemented as mutually independent iid sequences.

Constraints 5 to 7 require more care. First, the x_i are drawn from H. Second, the θ_is are drawn (as explained in Section 8.1.3) but not yet bound to the file indexes. Instead, the values are put in a set Θ. In view of Equation (8.9), define

$$\hat{\theta}_{z(m)} = F(x_{(m)}) - \sum_{n=1}^{m-1} \theta_{z(m)}$$

so that we should have $\hat{\theta}_{z(m)} = \theta_{z(m)}$ for all m. If this would be true, it is easy to see that all constraints are satisfied. However, this can be done in [Barford98-Sigmetrics] only approximately. Here is one way to do it. Assume that $z(m) =$, namely, we have sorted the file indices by increasing file size. For $m = 1$ we set θ_1 to the value in Θ which is closest to $\hat{\theta}_1 = F(x_1)$. Then remove that value from Θ, set θ_2 to the value in Θ closest to $\hat{\theta}_2 = F(x_2) - \theta_1$, etc.

Lastly, it remains to generate a time series of file requests $I(t)$ such that the marginal distribution is given by the θ_is and the temporal locality in condition 8 is satisfied. The method in SURGE can be sketched as follows. First, for a given trace size, the number of occurences of file i is drawn at random. This produces a sequence of values N_i (with $\mathbb{E}(N_i) = \theta_i$). A sequence T of values of temporal localities is drawn, using an iid sequence of integers with lognormal distribution

[12] Inactive OFF time, File size, File request size. The number of embedded references is Pareto but not heavy tailed.

	model	density $f(x)$	value of parameters
Inactive OFF time	Pareto	Equation (8.4)	$k = 1$, $p = 1.5$
No of embedded references	Pareto	Equation (8.4)	$a = 1$, $p = 2.43$
Active OFF time	Weibull	Equation (8.1)	$b = 1.46$, $c = 0.382$
File Size	Lognormal	Equation (8.2) $\times 1_{\{x \leq a\}}$	$\mu = 9.357$, $\sigma = 1.318$
	comb. Pareto	Equation (8.4)	$a = 133K$, $p = 1.1$
File Request Size	Pareto	Equation (8.4)	$a = 1000$, $p = 1.0$
Temporal Locality	Lognormal	Equation (8.2)	$\mu = 1.5$, $\sigma = 0.80$

Table 8.1: Distributions and parameters used in SURGE.

(approximately). Then a stack S is created, which initially contains the set $\{1, 2, ..., i, ..., I\}$ (every file name appears exactly once). Let $i = S[T[1]]$. If $N_i > 0$, N_i is decremented, the value of $r(1)$ is moved from its position to the top of S. Else, i is deleted from the stack. Then the operation is repeated (a second value $i = S[T[2]]$ is selected and so on) until the stack is empty. This emulates the behaviour of a stack of least recently used references and provides a process with the required distribution. [Barford98-Sigmetrics] describes a refinement of the method, which produces a more uniform appearance of file request names throughout the sequence.

8.4 FURTHER READING

8.4.1 OTHER SOURCE MODELLING ASPECTS

We have focused in this chapter on distribution of single random variable. In the following chapters we will consider time correlation aspects in more detail. In particular, we will see that there is a relation between heavy tail (a property of a marginal distribution) and long range dependence (a property of the time correlation of a process).

A difficult issue is stationarity. The analyses shown in this chapter assume that the data comes from a stationary random process. If this is not true, then statistical tests are not valid. There are indications that the stationary models as seen in this chapter are valid for relatively short periods of time [Zhang02]. Over longer periods, other models that account for time dependence are needed (see Chapter 9).

A simple, traditional model of request arrivals is the Poisson process. However, for HTTP requests or TCP connection openings (SYN packets), this is not a valid model in general, but it is so for user level session generation [Paxson95-ToN], or also for traffic flows inside an operator backbone (where a very large number of sources is aggregated).

Power laws is a very popular topic in research literature. They were found to hold for topology aspects of the internet [Faloutsos02-].

8.4.2 OTHER LOAD GENERATION TOOLS

There are many other load generation tools. A load generation tool is called a benchmarking tool when it comes with an exact specification of the load to be generated and the experimental procedure. It is used to compare new products such as web server or back-end software. See the

document resource page for more information.

Packet generators are low level load generators that create data packets suitable for testing networks. Primitives include a choice of distribution of inter-packet generation times.

See also [Nahum02-ToN] for a description of performance aspects of web servers.

8.5 EXERCISES

EXERCISE 8.1. *Write an S-program that displays the aggregation of stable random variables, similar to Figure 8.5, for $p = 0.5, 1.0, 1.5, 2$, for skewness = 1.0 (totally skewed stable distribution). For $p < 2$, this distribution is on the set of positive numbers.*

EXERCISE 8.2. *Read [Braford98-Sigmetrics] and answer the following questions.*

1. *What is the difference between request size and file size ? Why is it important to model both ?*
2. *Why is temporal locality important to model ?*
3. *What are the drawbacks of a trace based approach ?*
4. *How is the load generator validated ? The generator is validated against its specification by measuring the empirical distributions. There is no verification against real data.*
5. *Why does the Anderson-Darling test reject the proposed distribution whereas the empirical plot seems to coincide ?*
6. *What are the differences between Surge and SpecWeb96 ?*
7. *Why is there a problem with temporal locality when several Surge clients run in parallel ?*

EXERCISE 8.3. *Read [Downey01-IMW] and answer the following questions.*

1. *What model does Downey propose for HTTP transfer time ? What did Barford and Crovella propose ? How is the difference in conclusions explained ?*
2. *What is the goal of the curvature test ? How does it work ? How is the p-value computed ?*
3. *Which model do you prefer for HTTP transfer times ?*
4. *What is the experimental setup to run Surge ?One server machine (UNIX, C code) and a number of client machines (any OS, Java code). Clients send file requests to the server over a network.*
5. *Can Surge be used to test a server or a network ?*

EXERCISE 8.4. *Homework to be designed in detail*

1. *run surge clients*
2. *verify distribution aspects - use aest to compute heavy tail index of inactive OFF times*
3. *solve a PE question on the performance of wireless LAN*

CHAPTER 9

FORECASTING

In this provisional version, this chapter uses the reference [Weber-TS], an introductory text on time series by Prof. Weber, Cambridge University, that is publicly available (see web site for more information).

From `http://www.perfdynamics.com`:

> Traffic planning is an absolute must, and it's hard to do when you start, because you don't have enough data to predict off of. After building some data, you can use a spreadsheet to create a simple traffic prediction model based on the historical data. Get more sophisticated later as the need demands and time permits. Also, it is good to choose some performance metrics that you will measure over time. One useful metric is the number of Web pages [served] per minute, per CPU. This can be used to predict hardware requirements against the traffic model or to monitor system performance changes over time.

We consider performance evaluation activities which involve forecasting.

- Web site *Capacity Planning*: define how many hosts are needed next quarter to run a web site
- *Software Rejuvenation*: decide when to restart a server program, in order to avoid drawbacks of software aging.
- Dynamic Host *Load Management*: allocate jobs to hosts, in a large scale distributed system, by predicting the available capacity. This example is discussed in Section 9.8.3.

In all cases, forecasting follows the pattern:

1. define a performance metric
2. define measurement methods
3. gather a time series of data
4. forecast from the time series.

In a lab exercise you will experiment with measurement methods. In this chapter we review methods for forecasting from a time series.

9.1 FORECASTING FROM A TIME SERIES

9.1.1 PREDICTION MODELS

Assume time is discrete. The idea is to explain the data by a *Prediction Model* of the form

$$Y_t = f(t, Y_1, ..., Y_{t-1}) + \epsilon_t \tag{9.1}$$

where $f()$ is some non-random function and ϵ_t is a noise sequence, such that $\mathbb{E}(\epsilon_t | Y_1, ..., Y_{t-1}) = 0$. Then, given that we observed $Y_t, Y_{t-1}, ...$ the one-step ahead prediction at time t is $(Y_{t+1}|Y_1, ...Y_t) = f(t+1, Y_1, ..., Y_t)$. A frequent case is when ϵ_t is an iid sequence with 0 expectation.

EXAMPLE 9.1: AVAILABLE CPU. Figure 9.1 shows the available CPU on a host. A prediction model is $Y_t = m(t) + X_t$ with $m(t) = .23697742 + .00045397267t - .000000908222283t^2$ and

$$X(t) = .8226X(t-1) - .01234X(t-2) - .002060X(t-3) - .08239X(t-4) + .6494X(t-5) - .5738X(t-6) + \epsilon_t$$

Here we have

$$f(t, Y_{t-1}, ..., Y_1) = m(t) + .8226X(t-1) - .01234X(t-2) - .002060X(t-3) - .08239X(t-4) + .6494X(t-5) - .5738X(t-6)$$

with $X_{t-j} = Y_{t-j} - m(t-j)$. This is an example of autoregressive process model (see Section 9.7 for more details).

REMARKS.

- For model 1 (regression model), $f()$ is a function of t only. The h-step ahead prediction prediction is simply $(Y_{t+h}|Y_1, ...Y_t) = f(t+h)$.
- For model 2, function f does depend on the past observed values. Here h-step ahead prediction is done by applying one-step ahead recursively, replacing the future observed values $Y_t, Y_{t+1}, ..., Y_{t+h-2}$ by their predicted values.
- The more correlation there is in the data, the smaller the variance of ϵ_t in proportion to the total variance of Y_t. If the data is iid, the only prediction we can do is to estimate the mean and give it as predicted value.
- A major difficulty is to find a tractable model for the data. The task is facilitated by a number of classical transformations, described below.
- We do not predict the noise when it is iid, but we can give a confidence interval.
- If the data can be modeled by a second order stationary process, then the prediction model is derived from the autocovariance function – see Section 9.7.5.

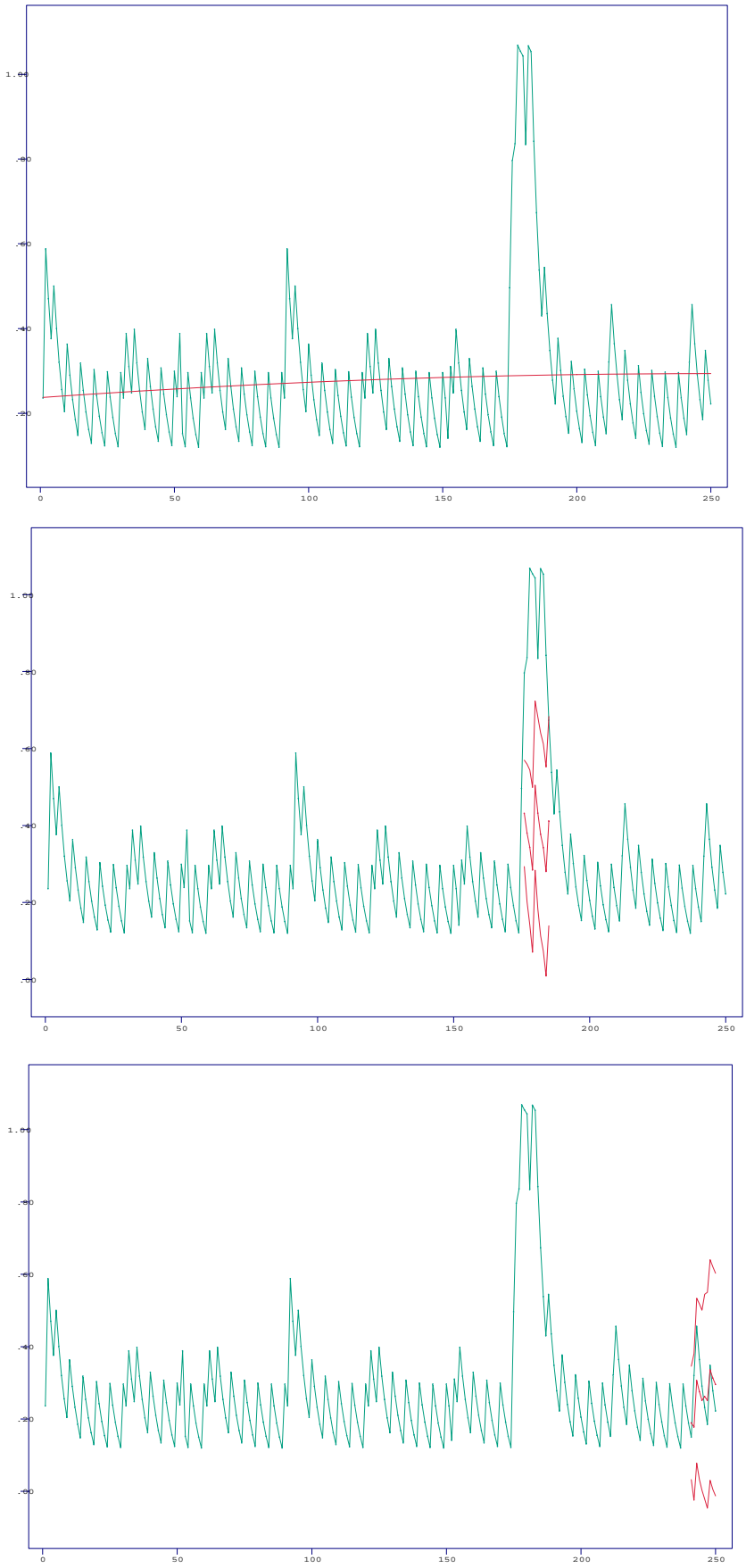


Figure 9.1: Available CPU on a host (courtesy of Peter Dinda). One point every 10 seconds. First graph: raw data and smoothed estimate $m(t)$. Following graphs: 10-step ahead predictions at times 175 and 240, with confidence interval, based on model described in text.

9.1.2 FORECASTING METHODS

There is a large number of forecasting methods. We consider the following ingredients, which are often used in combination.

1. Fit a regression model to the data. If the residuals can be assumed to be iid, then this is sufficient.
2. Transform the data before applying another method, using reversible filters. The transformed data can then be fit to regression model, or to a stationary time series model (see below). This is useful to remove trend and seasonal components and to separate time scales (using multiresolution analysis).
3. Else, fit a stationary process to the residuals, in order to transform them into iid noise. This is the object of Section 9.7.
4. A heuristic, simple method, is the Holt-Winters method. It is in fact a special case of Section 9.7, but is much simpler to use than the general method.

In all cases, scale transformations such as Box-Cox (Equation (2.16)) should be used to make the model look additive.

9.1.3 THE MEANING OF PREDICTION.

Here, prediction is based on a model, fitted from past observation. We can thus extract quantitative information about trends and risks, growth and decline, and use it for resource allocation.

However, remember that the goal is limited to *forecast what can be forecast*. Indeed, forecasting is much like driving a car by looking into the mirror. No automatic software can forecast the unexpected.

Here, confidence intervals are valid only as long as the model, fitted from the past, continues to hold in the future. For example, the confidence intervals on Figure 9.1 may be violated if the process changes suddenly.

9.2 USE OF LINEAR REGRESSION

9.2.1 LINEAR REGRESSION MODELS

Fit the data to a model of the form

$$Y_t = \sum_i x_t(i)\beta(i) + \epsilon_t$$

where ϵ_t is the noise. Here $\beta(i)$ is the regression parameter, estimated by fitting the data on some window $[t - w + 1, t]$. If ϵ_t is iid, the prediction is

$$(Y_{t+h}|Y_1, ...Y_t) = \sum_i x_{t+h}(i)\beta(i)$$

and a confidence interval follows from the general theory of Section 9.7.

If the data shows no periodic behaviour, a simple fit to a function of t with few parameters can be used. The model for Swiss population data in Figure 9.11 is obtained with this method, by fitting a polynomial of degree 2. If we are happy with this forecast, we can use confidence intervals from Section **??** and the method stops here. See Section 9.8.1 and Section 9.8.2 for some examples.

In contrast, if the noise shows some non iid structure, we use Section 9.7.

9.2.2 APPLICATION TO SEASONAL MODELS

Harmonic + Trend model

$$Y_t = f(t) + a_0 + \sum_{j=1}^{h} (a_j \cos(\omega_j t) + b_j \cos(\omega_j t)) + \epsilon_t$$

where $f()$ is a function (for example: polynomial) with k parameters; h is the number of harmonics used – the higher h is, the more accurate, but the less parsimonious the model is. The model has $k+2h+1$ parameters in total, and can be estimated using the linear regression method in Section **??**.

EXAMPLE 9.2: SPRINT TRAFFIC. (Figure 9.2). It has periodicity 16 (= 24 hours). We fit the harmonic plus trend model, using the method in Chapter **??** and obtain

$$\begin{aligned} Y_t = {} & .21818262E+09t^0 + .37733531E+06t^1 - .13294939E+04t^2 \\ & -.87232216E+08\cos(2*\pi*t/16) - .38962764E+07\sin(2*\pi*t/16) \\ & -.21264199E+08\cos(2*\pi*t/8) - .22685501E+08\sin(2*\pi*t/8) + \epsilon_t \end{aligned}$$

Confidence intervals are computed, assuming the normal iid model fits. We derive a prediction with confidence intervals by letting t be outside the measurement interval

---

Season + Trend model This is an alternative, defined by:

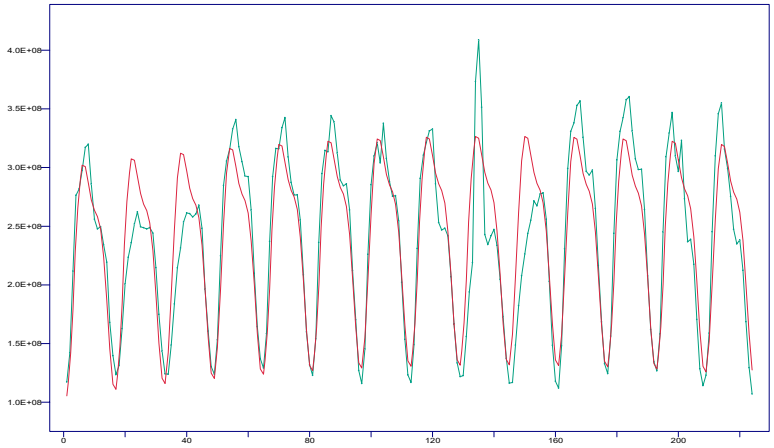
$$Y_t = a_{t \bmod s} + f(t) + \epsilon_t$$

where f has k parameters (for example a polynomial of degree $\leq k-1$). The model has $s+k$ parameters in total, and can be estimated using linear regression method.

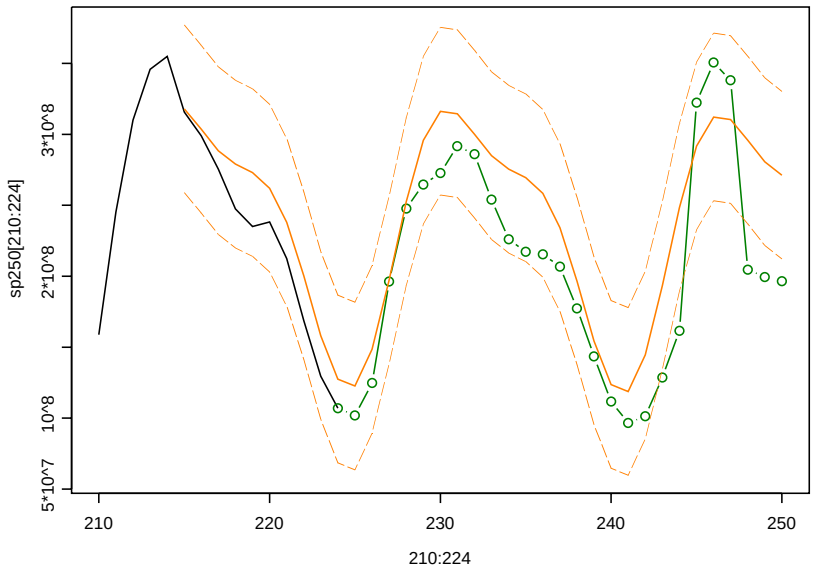
9.3 FINDING PERIODICITIES

A first step is a visual inspection of data, which reveals trends and seasonal components.

A method for mechanically finding a period is the periodogram ([Weber-TS] Chapter 4). The presence and value of a period can be determined with the periodogram.



(a)



(b)

Figure 9.2: (a) traffic volume on an american coast-to coast link (courtesy of Sprintlabs) – one point every 90 mn with fitted harmonic + trend model with 5 parameters; (b) model with confidence interval versus actual data. From 225 to 250, the actual data (not known when the model was fitted) is shown with circles.

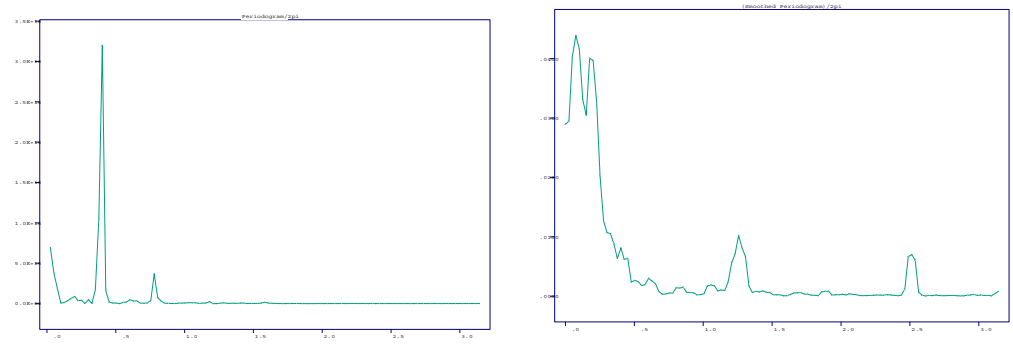


Figure 9.3: Periodograms for Example 9.2 on page 203 (a) and Example 9.1 on page 200 (b).

APPLICATION TO EXAMPLE 9.2 ON PAGE 203 Figure 9.3 (a) shows a periodic behaviour with period $s = 16$. The periodogram has a high peak at $\omega = 0.40343$ (radians), which corresponds to the period $s = \frac{\omega}{2\pi} n \approx 16$, where $n = 250$ is the sample size. Sometimes the periodogram is not as clear: see Figure 9.3.

QUESTION 9.3.1. *In what sense is the periodogram a poor estimator ?* [1]

QUESTION 9.3.2. *How is the periodogram computed in practice ?* [2]

Another method is the autocorrelation, defined in Section 9.7 – see Figure 9.4.

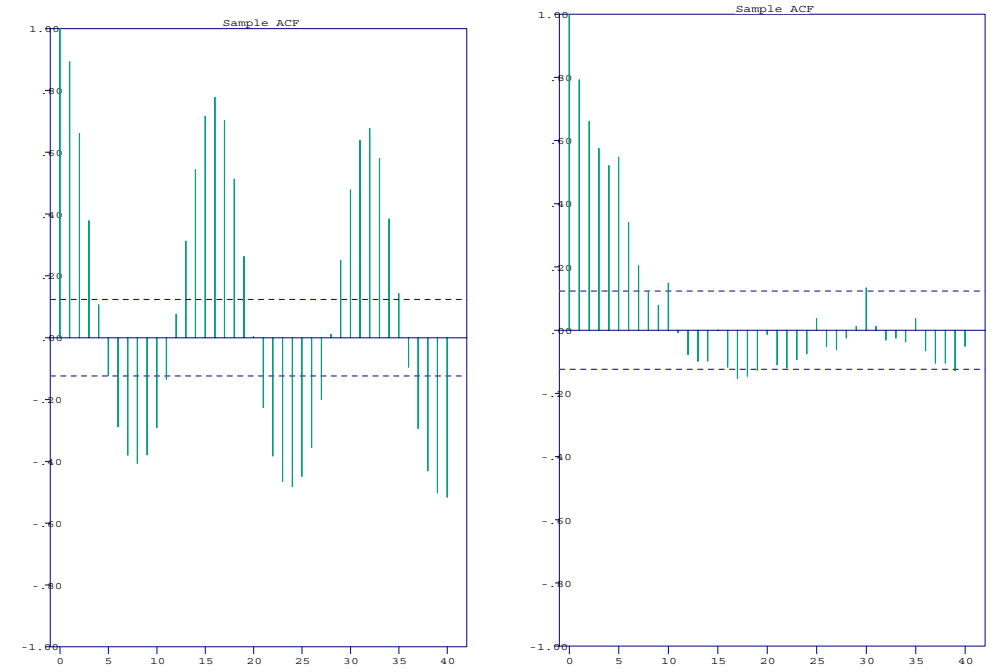


Figure 9.4: Auto-correlation function for Example 9.2 on page 203 (a) (period = 16) and Example 9.1 on page 200 (b).

[1] As the length of the time series increases, the variance of $I(\omega)$ does not decrease. In practice, the noise is high. This is why it is required to look at smoothed versions of the periodogram.

[2] Using a Fast Fourier transform.

9.4 TRANSFORMING THE DATA

9.4.1 THE "CLASSICAL METHOD"

Most time series show both trend and seasonal components. In some cases, we are interested in capturing both. In other cases, we my be interested only in the trend. In this section we review several methods for isolating trends and seasonal components that use filters. The difference with regression models is that filters are fixed and do not depend on the data, therefore, they do not interfere with the computation of confidence intervals. We require that the data tranformation we apply is reversible.

LINEAR FILTERS

Formally, we view a filter $\mathcal{L}$ as a time invariant linear mapping from the set of deterministic sequences $(y_1, ..., y_n, ...)$ onto itself. Such a mapping has the form

$$(\mathcal{L}Y)_t = \sum_{s=-\infty}^{\infty} a_s Y_{t-s}$$

where by convention we let $Y_t = 0$ for $t \leq 0$. In the rest of this lecture we assume that the mapping is regular, i.e. $\sum_{s=0}^{+\infty} |a_s| < +\infty$. The mapping is causal if $a_s = 0$ for $s < 0$.

If you need more background on filters, read [Thiran02-LN] Chapter 4 or [Weber-TS] Chapter 3 and Section 5.1.

MOVING AVERAGES By definition, a *moving average* filter is one such that $\sum_s a_s = 1$.

Moving average filters exist in various flavours, depending on how much weight they put on the past. The window of a filter is the set of s such that $a_s \neq 0$. Classical filters are *moving averages* (finite windows). Let us mention the *deseasonalizing filter*; it aims at separating a seasonal component. For d odd, the simple de-seasonalizing filter is the simple symmetric moving average seen above with $d = 2q + 1$. For d even, the simple de-seasonalizing filter is given by $a_s = 0$ for $|s| > d/2$, $a_{-s} = a_s$ and $a_{d/2} = 0.5/d$, $a_s = 1/d$, for $|s| < d/2$. Variants are the centered moving averages described in [Weber-TS] Section 6.2. $\mathcal{L} - Id$ is a projector (generaly not orthogonal) onto the set of periodic sequences.

A moving average filter is low-pass, i.e., its power transfer function is high for small pulsations ω. For other moving average filters see [Weber-TS] Chapter 6.

APPLICATION TO EXAMPLE 9.2 ON PAGE 203 The data in Figure 9.5 has period 16; we apply the moving average filter defined by $a_s = 0$ except for

$$\begin{cases} a_s = \frac{1}{16} \text{ for } s = -7 \dots 7 \\ a_8 = a_{-8} = \frac{0.5}{16} \end{cases}$$

The result is an estimate of the *deseasonalized* data.

QUESTION 9.4.1. *How are the parameters of a moving average determined ?* [3]

[3] The window size of a MA filter can be estimated by plotting the variance of the differenced time series $\Delta_r Y_t$. The period of a deseasonalizing filter can be found with the periodogram. See [Weber-TS] Chapter 4.

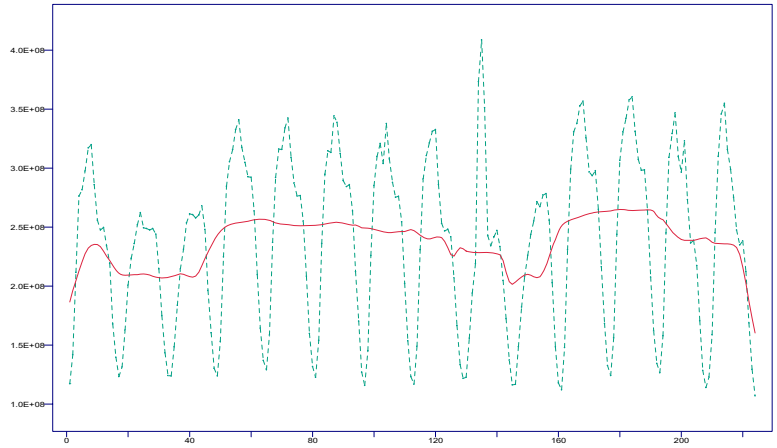


Figure 9.5: Symmetric moving average filter (deseasonalizing filter) applied to Example 9.2 on page 203.

QUESTION 9.4.2. *What is the Slutzky-Yule effect ?* [4]

QUESTION 9.4.3. *Give one line of S code for filtering a time series x into y, with the moving average filter given in [Weber-TS] section 6.1.* [5]

THE "CLASSICAL" FITTING METHOD FOR MODELING BOTH TREND AND SEASONAL COMPONENTS

Assume the periodic component has period s.

1. Estimate the trend $\hat{m}_t$ by applying for example a de-seazonalizing filter, or any other filter deemed appropriate to the problem.
2. Estimate the season component by using some projector of $Y_t - \hat{m}_t$ onto the set of periodic sequences with 0 mean. For example, the orthogonal projector gives $\hat{s}_1, ..., \hat{u}_s$ defined by

$$\hat{u}_j = w_j - \bar{w}$$

   with $w_j := \sum_k (Y_{kd+j} - \hat{m}_{kd+j})$ and $\bar{w} = \sum_j w_j$. Then $\hat{u}_t$ is extended to all t by periodicity.
3. Fit a linear regression model with k parameters to $Y_t - \hat{u}_t$.

This gives a regression model with k parameters instead of $k + s$.

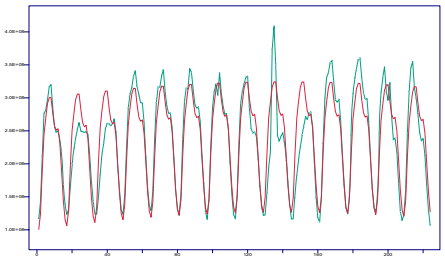
There are other filters that act on the frequency domain.

[4] With some filters such as $\frac{1}{6}[-1, 2, 4, 2, -1]$, a repeated filtering operation makes the filtered time series periodical. This does not happen with the simple moving average filters that we used above – repeated operation simply removes high frequencies.

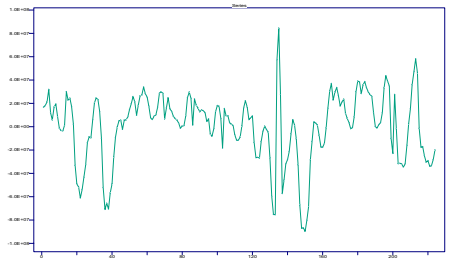
[5] `y <- filter (x, c(-2,3,6,7,6,3,-2)/21, sides=2)`

USE FOR PREDICTION Prediction is performed on the transformed data, using for example a regression model. The final prediction is obtained by inverting the transformation.

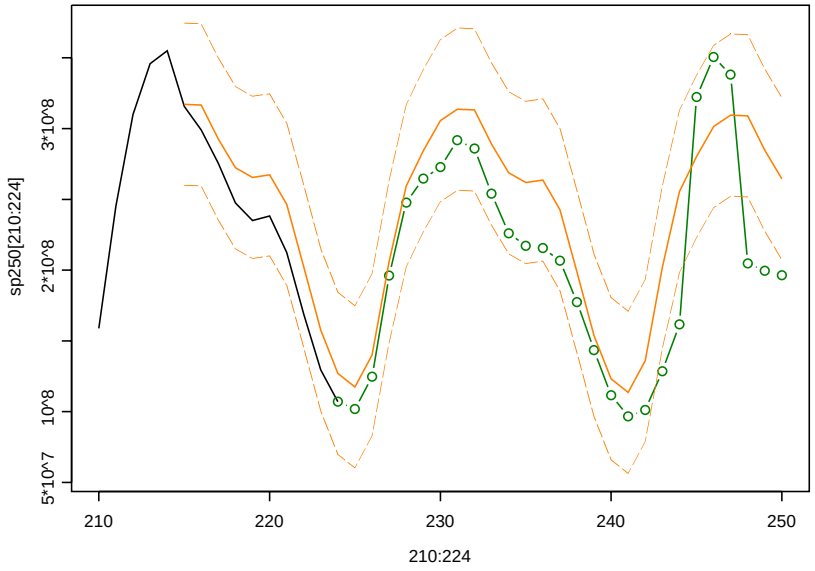
APPLICATION TO EXAMPLE 9.2 ON PAGE 203 Figure 9.6 shows the application to Example 9.2 on page 203. The confidence intervals are obtained assuming the iid noise model fits. The residuals however show that this assumption does not seem to hold. For such cases, we may need more sophisticated models, as in Section 9.7.



(a) Fitted model



(b) Residuals



(c) Fit and Predictions

Figure 9.6: Classical method for trend and season analysis applied to Example 9.2 on page 203. o = actual value of the future (not used for fitting the model) – compare to the forecasts.

9.4.2 DIFFERENCING

An alternative to regression models is to used the *differencing filter* is defined by

$$(\Delta Y)_t = Y_t - Y_{t-1}$$

It is able to remove polynomial trends of any order (by repeated application). For example, if $Y_t = Z_t + at + b$ and Z_t is stationary, then $(\Delta Y)_t = (\Delta Z)_t + a$ does not have a trend anymore. Δ is the discrete time equivalent of a derivative.

A seasonal component with period s can be removed with the lag s differencing filter defined by

$$(\Delta_s Y)_t = Y_t - Y_{t-s}$$

Differencing filters are high-pass filters and thus give an estimate of the noise ϵ_t. They can be used as alternative to moving average filters for isolating trend and seasonal components. See Figure 9.8 for an example.

Differencing does not have the problems of Slutzky-Yule effect mentioned above. Also, it can be inverted and the combination of several differencing filters does not have coefficients that depend on the data.

APPLICATION TO PREDICTION With a simple differencing filter, the reverse filter is given by

$$Y_t = \sum_{s=0}^{+\infty} Z_{t-s}$$

A one step predictor of Y_t is $Y_{t-1} + \hat{m}_t$, where $\hat{m}_t$ is the one step predictor of Z_t. This is applied on Figure 9.8.

The h-step ahead predictor is

$$(Y_{t+h}|Y_1, ...Y_t) = Y_t + \sum_{s=t+1}^{h} \hat{Z}_s$$

where $\hat{Z}_s$ is a predictor of Z_s. If Z_s can be assumed iid white noise, then $\hat{Z}_s = 0$.

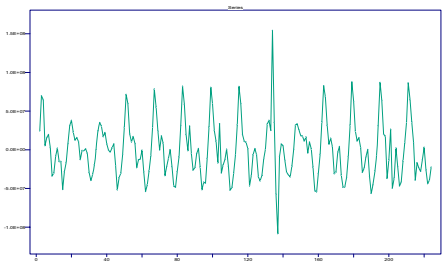
The confidence interval for $(Y_{t+h}|Y_1, ...Y_t)$ is obtained computing the distribution of the sum of h variables. Thus it grows with h. Compare to pure regression methods where this does not happen.

APPLICATION TO EXAMPLE 9.2 ON PAGE 203 We difference at lags 1 and 16 to remove trends and seasonal components, and fit the residuals to iid noise. Note the differences with Figure 9.2 and Figure 9.6:

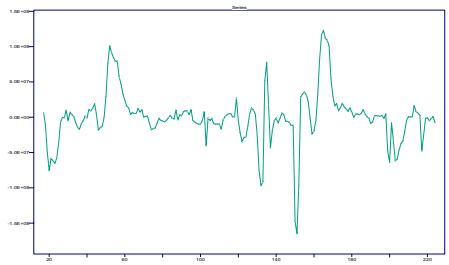
- the confidence interval increases with the prediction horizon
- the prediction is more adaptive in that it starts from the exact value

QUESTION 9.4.4. *Does the order in which differencing at lags* 1 *and* 16 *is performed matter ?* [6]

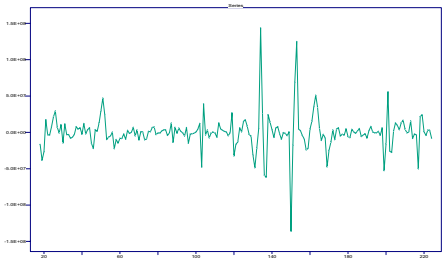
[6] No.



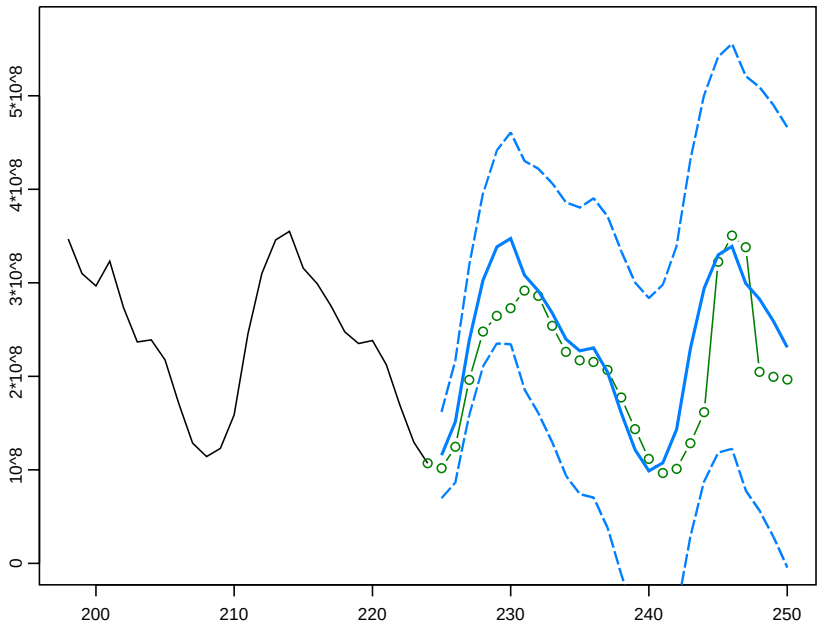
(a) Differencing at Lag 1



(b) Differencing at Lag 16



(c) Differencing at Lags 1 and 16



(d) Prediction at time 224

Figure 9.7: Differencing filters Δ_1 and Δ_{16} applied to Example 9.2 on page 203. The forecasts are made assuming the differenced data is iid gaussian. o = actual value of the future (not used for fitting the model).

QUESTION 9.4.5. *Is there a difference between Δ_s, the lag s differencing filter and Δ^s, the repeated operation of the differencing filter ?* [7]

QUESTION 9.4.6. *Give one line of S code for a differencing filter.* [8]

9.4.3 AD-HOC FILTERS

For data traffic, a common way to remove daily variations is to compute the daily peak (largest value of Y_t in a calendar day). One should be careful about aggregation of data; for very small aggregation intervals and long range dependent data (see Chapter 10) the largest value increases sharply as the aggregation interval decreases. The aggregation interval should be significant to the performance metric we chose. For example, for Internet network engineering, it is of the order of 10 mn.

9.4.4 MULTI-RESOLUTION ANALYSIS

Very long time series may exhibit a mixture of trends and seasonal components at several time scales. It then becomes difficult to define good filters, with the methods seen previously.

A tool of choice for such cases is multi-resolution analysis, based on perfect reconstruction filter banks. First, the data is separated between a smooth part (using a low pass filter such as a moving window average) and a residual. Then the same is applied to the smooth part, but at a double time scale, and the process is continued for a number of steps. The original time series can be perfectly reconstructed from the successive residuals and the last smooth part, using another family of filters. Perfect reconstruction filter banks are built using the theory of wavelets. In many cases, and with properly chosen wavelets, the method identifies the time scales at which a detailed modelling is required, and can be done with independent models. At other time scales, the residuals can be modelled as iid noise. For an example, see [Pappagiannaki03-Infocom] and Exercise 9.9. For more details on multi-resolution analysis, see Chapter 13.

9.5 THE HOLT-WINTERS METHOD

Low pass causal filters can directly be used as heuristic for prediction, without explicit regression model transformation. We present here the *Exponentially weighted moving average* (EWMA)(infinite windows), also called *exponential smoothing*. It is also known as the *Holt-Winters* method.

9.5.1 SIMPLE EXPONENTIAL SMOOTHING

EWMA is a linear filter with infinite window in the past. It is used for smoothing the data, when infinite window is adequate, and for simple one-step ahead forecast.

$$\hat{m}_t := \alpha \sum_{s=0}^{+\infty} (1-\alpha)^s Y_{t-s}$$

[7] Yes, for example for $s = 2$: $\Delta_2 Y_t = Y_t - Y_{t-2}$ whereas $\Delta^2 Y_t = Y_t - 2Y_{t-1} + Y_{t-2}$.

[8] `y <- filter (x, c(1,-1), sides=1)` or `y <- diff (x, lag=1, differences =1)`.

where by convention we let $Y_t = Y_1$ for $t \leq 1$. The implicit prediction model is

$$Y_t = m_t + \epsilon_t \tag{9.2}$$

and the one-step ahead prediction is $(Y_{t+1}|Y_1, ...Y_t) = \hat{m}_t$.

QUESTION 9.5.1. *What is the h-step ahead predictor ?* [9]

The main feature is the recursive computation of $\hat{m}_t$:

$$\hat{m}_t = \alpha Y_t + (1 - \alpha)\hat{m}_{t-1} \tag{9.3}$$

with initial condition $\hat{m}_1 = Y_1$.

EWMA works well when the data has no trend or periodicity, see Figure 9.8.

Simple EWMA has one parameter $\alpha \in [0, 1]$; its value can be determined so as to minimize the one-step forecasting error on some training data.

QUESTION 9.5.2. *What is EWMA for $\alpha = 0$? $\alpha = 1$?* [10]

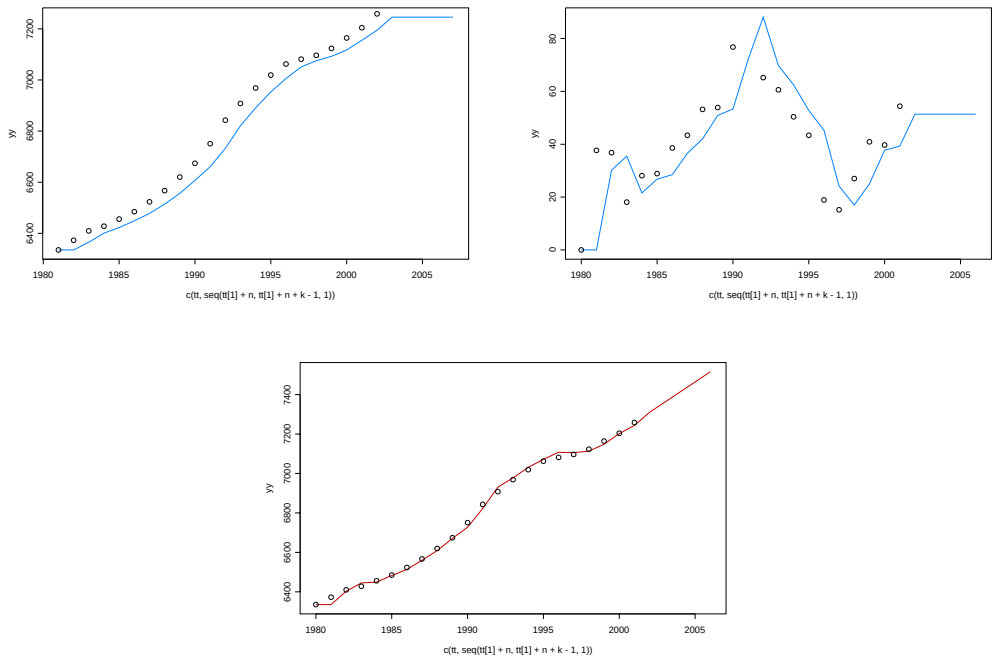


Figure 9.8: First graph: simple EWMA applied to swiss population data Y_t with $\alpha = 0.9$. EWMA is lagging behind the trend. Second graph: simple EWMA applied to the differenced series ΔY_t. Third graph: prediction reconstructed from the previous graph.

QUESTION 9.5.3. *Give the formula for the prediction illustrated in Figure 9.8, third graph.* [11]

[9] $(Y_{t+h}|Y_1, ...Y_t) = \hat{m}_t$, by recursive application.

[10] $\alpha = 0$: a constant, equal to the initial value; $\alpha = 1$: no smoothing, $\hat{m}_t = Y_t$.

[11] Let $Z_t = (\Delta Y)_t$. The h-step ahead predictor for Z_{t+h} is $\hat{m}_t$, obtained recursively by Equation (9.3). The h-step ahread predictor for Y_{t+h} is thus $Y_t + h\hat{m}_t$.

9.5.2 DOUBLE EXPONENTIAL SMOOTHING

It is used when the data has a slow varying trend, and when it is deemed important to keep a memory of the entire sequence. The idea is to apply EWMA to the trend itself. Double EWMA(α, β) is defined as follows. The model is

$$Y_t = a_t + b_t + \epsilon_t$$

where a_t represents the trend level and b_t the trend slope. The filter is defined by

$$\begin{cases} \hat{a}_t = \alpha Y_t + (1-\alpha)(\hat{a}_{t-1} + \hat{a}_{t-1}) \\ \hat{b}_t = \beta(\hat{a}_t - \hat{a}_{t-1}) + (1-\beta)\hat{b}_{t-1} \end{cases}$$

and the h-step predictor is

$$(Y_{t+h}|Y_1, ..., Y_t) = \hat{a}_t + h\hat{b}_t$$

See Figure 9.9 for an example. As for simple EWMA, the parameters are in $[0, 1]$ and need to be fitted on some training data.

QUESTION 9.5.4. *How is double exponential smoothing defined ? When is it used ?* [12]

QUESTION 9.5.5. *How are the parameters of an EWMA filter determined ?* [13]

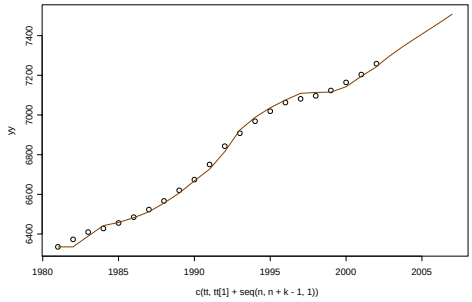


Figure 9.9: Double EWMA with $\alpha = 0.8, \beta = 0.8$. It gives a good predictor; it underestimates the trend in convex parts, overestimates it in concave parts.

QUESTION 9.5.6. *What would the forecast be if we do double EWMA on the differenced series in Figure 9.8 ?* [14]

QUESTION 9.5.7. *Show that simple EWMA(α_1) applied to the differenced series is the same as double EWMA with parameters to be identified.* [15]

[12] tbd. It is used as an alternative to moving averages and is able to model a trend, when it is important to keep a memory of the entire sequence.

[13] By minimizing the forecasting error on the past data.

[14] The trend in the difference would be extrapolated; this is equivalent to assuming that the quadratic growth in the last years will continue. In contrast, simple EWMA applied to the differences assumes that the over linear growth in the last years is a random effect and will not be sustained.

[15] Same as double EWMA with $\alpha = 0, \beta = \alpha_1$.

9.5.3 TRIPLE EXPONENTIAL SMOOTHING

is used when there is both trend and seasonal component. Triple EWMA(α, β, γ) is defined as follows. The model is

$$Y_t = a_t + b_t + c_t + \epsilon_t$$

where a_t is the level of the trend, b_t the slope of the trend, and c_t the correction term for seasonal variation, assumed to have a period s. The filter is

$$\begin{cases} \hat{a}_t = \alpha(Y_t - \hat{c}_{t-d}) + (1-\alpha)(\hat{a}_{t-1} + \hat{b}_{t-1}) \\ \hat{b}_t = \beta(\hat{a}_t - \hat{a}_{t-1}) + (1-\beta)\hat{b}_{t-1} \\ \hat{c}_t = \gamma(Y_t - \hat{a}_t) + (1-\gamma)\hat{c}_{t-d} \end{cases}$$

The h-step ahead prediction is

$$(Y_{t+h}|Y_1, ..., Y_t) = \hat{a}_t + h\hat{b}_t + \hat{c}_{t+h-d\lceil \frac{h}{d} \rceil}$$

In the formula, $\hat{c}_{t+h-d\lceil \frac{h}{d} \rceil}$ is the latest estimate of the seasonal component available at time t, taken at a time instant with the same phase as $t+h$.

As for simple EWMA, the parameters are in $[0, 1]$ and need to be fitted on some training data.

APPLICATION TO EXAMPLE 9.2 ON PAGE 203. The application of triple EWMA is shown on Figure 9.10. The coefficients are obtained by training the estimator over 230 data points. Confidence intervals are obtained based on the fact that EWMA is a special case of seasonal ARMA model, discussed in detail in Section 9.7.

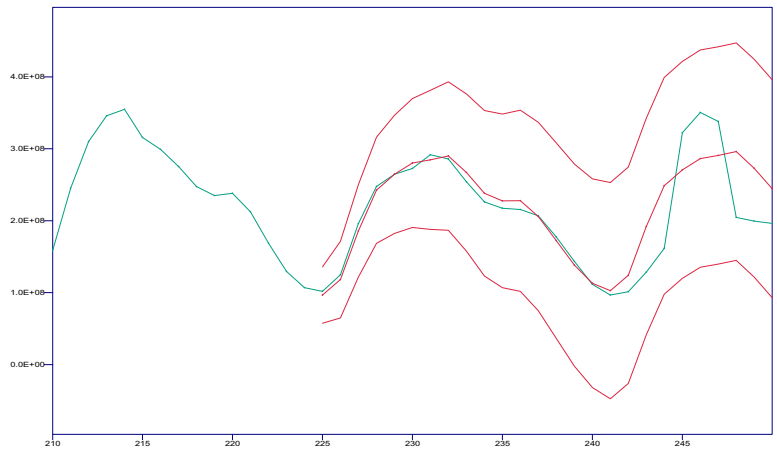


Figure 9.10: Seasonal Holt-Winters prediction applied to Example 9.2 on page 203.

9.5.4 EWMA AND STATE-SPACE APPROCHES

EWMA is in fact a simple example of Kalman filter. Kalman filters are themselves a special case of state-space approach, where the idea is that the observable process is a combination of non-observable ones, with known properties. For triple EWMA, the non-observable processes are the trend and seasonal components a_t, b_t, c_t. The filter equations correspond to maximizing the likelihood, under the assumption of iid, normal noise. The parameters α, β, γ determine the variance matrix of the noise. For details, see [Harvey90-Book]. For a general presentation of Kalman filters, see [Weber-TS] Chapter 8.

9.6 SELECTING A MODEL ORDER

9.6.1 PROBLEMS WITH OVER-FITTING

Common sense tells us that, for equivalent fits, we should pick the simplest model. If the models are nested and are regular, normal with same variance, then we can used ANOVA (Section 4.5) to decide which model explains the data best.

EXAMPLE 9.3: SWISS POPULATION. Figure 9.11 shows the Swiss population fitted to a polynomial of degree 2. The model is $Y_t = f(t) + \epsilon_t$, with $f(t) = 6240.8552 + 48.130583t - .033109825t^2$. The prediction at time t is $f(t+1)$.

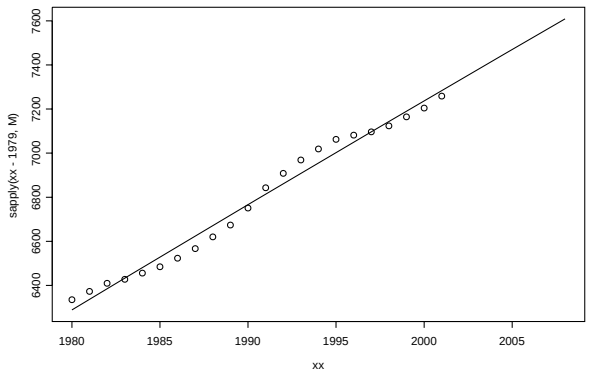


Figure 9.11: Swiss Population forecast for 2005.

We now go one step further: goodness of fit is not an absolute measure of goodness of model. To see why, consider the model in Figure 9.12. A regression model with degree 8 gives a perfect fit. However, its prediction power is ridiculous. At the extreme, a model with absolute best fit has 0 residual error – but it is no longer an explanatory model.

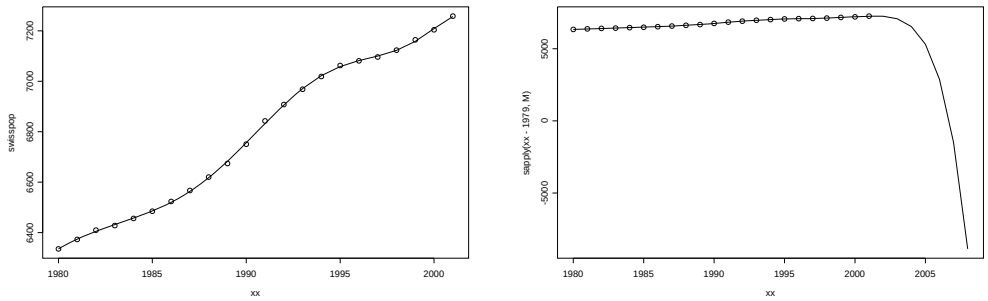


Figure 9.12: Problems with overfitting. First graph: Swiss Population (dots), fitted to a polynom of degree 8 (line). Second graph: prediction based on the polynomial.

Therefore a fitting procedure should use some *information criterion*, which quantifies how much information the model carries. In a parametric family of models, the model with *smallest* information criterion is chosen.

Another family of methods for picking the right model is given by machine learning methods such as artificial neural networks.

9.6.2 AKAIKE'S INFORMATION CRITERION

Akaike's Information Criterion (AIC,) is defined by $AIC = -2l(\hat{\theta}) + 2k$ where k is the dimension of the parameter θ and $l()$ is the log-likelihood.

[Weber-TS] section 7.3 gives an interpretation in terms of entropy, which can be summarized as follows. Consider an independent replication $X = (X_t)_t$ of $Z = (Z_t)_t$. It can be shown that AIC is an approximately unbiased estimator of $\mathbb{E}_\theta(-2\log(f(X|\hat{\theta}(Z)))$. Call $H(\theta)$ the entropy of Z (or X) and $d(\theta||\hat{\theta})$ the Kullback-Leibler distance from the distribution of Z when the true parameter is θ to the distribution with the estimated parameter $\hat{\theta}$. It is known that $\Delta := H(\theta) + d(\theta||\hat{\theta})$ is the number of bits needed by an optimal code to describe X, when the optimal code thinks that the distribution of X is the one estimated from the sample Z (instead of the true one). Δ measures the efficiency of our model to describe the data. We have

$$\begin{cases} d(\theta\|\hat{\theta}) := \int(\log f(x|\theta) - \log f(x|\hat{\theta}))f(x|\theta)dx \\ H(\theta) := -\int \log f(x|\theta)f(x|\theta)dx \end{cases}$$

thus

$$\mathbb{E}_\theta(\Delta) = \mathbb{E}_\theta(-\log(f(X|\hat{\theta}(Z)))$$

and

$$\mathbb{E}_\theta(AIC) \approx 2\mathbb{E}_\theta(\Delta)$$

Thus AIC is a (biased) estimator of the expected value of 2Δ.

This analysis is approximate and the AIC is known to be slightly biased in favour of large model sizes ks. Many variants of AIC exist, with bias corrections that depend on the parametric model. See [BrockwellDavis02-book] or [ShumwayStoffer99-book].

9.6.3 MALLOW'S C_p

This criterion is more specific than AIC. It is used in the context of normal regression models with same variance, as defined in Section 4.3, when the models are not nested. Assume we have a number of models that all are subsets of one same large model M_0. *Mallow's C_p* criterion is defined for every model M by

$$C_p = \frac{SSR(M)}{s^2} - (N - 2p)$$

where $SSR(M)$ is the residual sum of squares for model M, s^2 is the estimator of variance for the full model M_0, N is the total number of samples, p [resp. p_0] is the dimension of model M [resp. M_0].

QUESTION 9.6.1. *Relate $SSR(M_0)$ to s^2* [16]

We can relate C_p to the F-statistic introduced in Theorem **??**, and used to test whether M alone explains the data. We have

$$F = \frac{(SSR(M) - SSR(M_0))/(p_0 - p)}{SSM(M_0)/(N - p_0)} = \frac{SSR(M)}{s^2(p_0 - p)} - \frac{N - p_0}{p_0 - p}$$

thus

$$C_p = (p_0 - p)(F - 1) + p$$

If model M is the true model, then F has a Fisher distribution with denominator degrees of freedom $n = N - p_0$. The expectation of Fisher (m, n) is $\frac{n}{n-2}$. Thus

$$\mathbb{E}(C_p|M) = \frac{2(p_0 - p)}{N - p_0 - 2} + p \approx p$$

where the approximation is for a large sample size N.

If model M is a poor fit, F is probably large and so is C_p. If M is a good fit, C_p is likely to be close to p. Mallow's method is to choose the model with the smallest C_p. Thus, if several models fit well, we will pick the one with the smallest dimension.

9.7 THE LINEAR TIME SERIES METHOD

9.7.1 TESTS FOR STATIONARITY AND WHITE NOISE

Given some time series Y_t, we apply the transformations mentioned earlier; the goal is to obtain a new times series X_t which is stationary. We describe tests and tools that are commonly used for assessing whether a sequence is stationary, or is iid.

[16] By Theorem **??**: $SSR(M_0) = (N - p_0)s^2$

STATIONARITY

A sequence of random variables X_t is *strictly stationary* is the joint distribution of any finite subsequence $(X_{t+t_1}, X_{t+t_2}, ..., X_{t+t_n})$ is independent of the time shift t. It is *second-order stationary* or *weakly stationary* if the first and second moments of any finite subsequence are independent of the time shift; this is equivalent to $\mathbb{E}(X_t)$ and $\mathrm{cov}(X_s, X_{s+t})$ are independent of t. For gaussion processes, both forms of stationarity the two are equivalent, but otherwise not.

ACF AND PACF

For a stationary process, the *auto-covariance function* is $\gamma_t = \mathrm{cov}(X_s, X_{s+t})$. The *auto-correlation function* (ACF) is $\rho_t = \gamma_t/\gamma_0$.

We call *White Noise* a zero-mean, uncorrelated sequence, i.e., with $\rho_t = 0$ for $t \geq 1$.

QUESTION 9.7.1. *Compare γ_k and γ_{-k}* [17]

QUESTION 9.7.2. *What is γ_0 ?* [18]

QUESTION 9.7.3. *Is there a difference between this definition of auto-correlation and the one you may have seen in other courses, for example in [Thiran02-LN] ?* [19]

The *sample auto-covariance*, for a sample $X_1, ..., X_n$ is defined, for $t \geq 0$ by

$$\hat{\gamma}_t = \frac{1}{n}\sum_{s=1}^{n-t}(X_{n+t} - \bar{X})(X_n - \bar{X})$$

where $\bar{X}$ is the sample mean. The *sample ACF* is $\hat{\rho}_t = \hat{\gamma}_t/\hat{\gamma}_0$.

PARTIAL ACF The *Partial Auto-Correlation Function* (PACF) is defined in Chapter **??** as the residual correlation of X_{t+h} and X_t, when $X_{t+1}, ..., X_{t+h-1}$ is known.

ACF OF AN IID SEQUENCE If $X_1, ..., X_n$ is iid with finite variance, then the sample ACF and PACF are asymptotically centered normal with variance $1/n$. ACF and PACF plots usually display the bounds $\pm 1.96/\sqrt{n}$. If the sequence is iid with finite variance, then roughly 95% of the points should fall within the bounds.

TESTS FOR STATIONARITY

- (Visual Test): Plot the sequence and look for trend or seasonal components
- If the sequence is stationary and is not long range dependent (Chapter 10), then ACF and PACF decay fast within the $\pm 1.96/\sqrt{n}$ bounds. If the sequence has a trend, then the sample ACF and PACF may show a very slow decay.

[17] They are equal.

[18] $\gamma_0 = \mathrm{var} X_t$ for all t.

[19] There may be. In [Thiran02-LN], autocorrelation is defined as $\mathbb{E}(X_s X_{s+t}$. It is a different thing. We use the terminology of statisticians.

- The *difference sign test* can be used to detect the presence of an increasing or decreasing test in the data. Let $S_n = \sum_{i=2}^{n} 1_{\{X_i > X_{i-1}\}}$. If X_i is iid, then S_n is approximately normal with mean $\mu_n = \frac{n-1}{2}$ and variance $\sigma_n^2 = \frac{n+1}{12}$. At confidence level 0.95, the difference sign test rejects the hypothesis that the data is iid if $|S_n - \mu_n|/\sigma_n > 1.96$.

TESTS FOR A NOISE SEQUENCE

If the transformed time series X_t can be considered as an iid sequence, then we stop the tranformations. Otherwise, we will model X_t as a more complicated process (see below).

The following tests are usually done [Brockwell-Davis02-book]:

TESTS FOR IID NOISE SEQUENCE. These tests evaluate the hypothesis H_0 that the noise sequence is white and iid

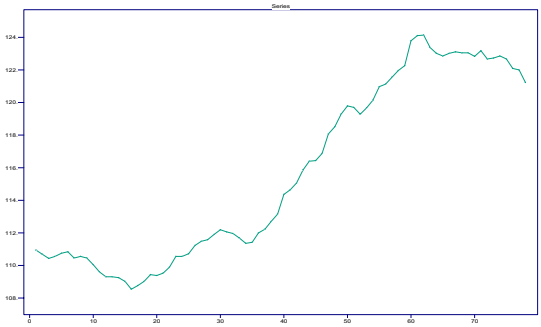
- (Visual Test): lag plot: scatter plot (X_{t+h}, X_t) for various h fixed. If the data is iid, the plot should show no structure. If it is slanted like an ellipsoid, this indicates a correlation in the direction of the principal axis.
- Sample ACF and PACF should fall within the $\pm 1.96/\sqrt{n}$ for most points (visual test). A formal test of this is called *Portmanteau*. The statistic, due to *Ljung and Box*, is $n(n+2)\sum_{t=1}^{h} \hat{\rho}_t^2/(n-t)$, the distribution of which under H_0 is approximately χ_h^2. The lag h has to be chose appropriately (!). A variant is *McLeod and Li*'s portmanteau, where $S' = n(n+2)\sum_{t=1}^{h} \hat{r}_t^2/(n-t)$, where $r()$ is the sample ACF of $X - t^2$. For large n, S' is also χ_h^2 under H_0.
- *Turning Point* Test: see [Weber-TS] Section 1.7
- *Rank Test*: S is the number of couples (s,t) with $s < t$ such that $X_i < X_j$. Under H_0, and for large n, $S \sim N(\mu_n, s_n^2)$ with $\mu_n = n(n-1)/4$ and $s_n^2 = n(n-1)(2n+5)/72$. If $|S - \mu_n|$ is large, we reject H_0. Further, if S is large, the sign of S is an indication of the size of a trend.

The following tests evaluate H_0: the noise is white and normal.

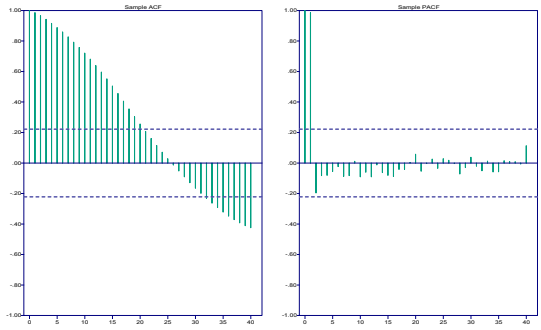
- Jarque-Bera's test (Section 8.1.2)
- For small samples (up to 200), the *correlation coefficient* R^2 of the QQ-plot can be used. In a linear regression model $X_i = aZ_i + b$, the correlation coefficient is $R^2 = S_{ZX}^2/S_{ZZ}S_{ZX}$ (see and Chapter **??**). For normal data, it is related to a student statistic. If the regression is valid, R^2 should be close to 1. Here, we apply R^2 to the regression of the ordered sample $X_{(1)}, ..., X_{(n)}$ regressed on $Z_i = N^{-1}(i - 0.5)/n$, the quantiles of the normal distribution. The distribution of R^2 in this case, and under H_0, is tabulated.
- another method is to fit the data to an ARMA model, defined below, and pick the best model, according to an information criterion. If the best model is the trivial model, then noise is declared white.

EXAMPLES

EXAMPLE 9.4: DOW JONES. Figure 9.13.



(a)



(b)

Figure 9.13: Closing values of the Dow Jones utilities index for 78 days (from [BrockwellDavis02-book]). (a) One value every day. (b) ACF and PACF.

A visual inspection shows that the data is not stationary. The ACF has a very slow decay, which confirms non-stationarity.

We transform the data by the classical method, using a polynomial fit of degree 2 (Figure 9.14). The transformed time series does not look stationary or normal, and does not pass the Ljung - Box,

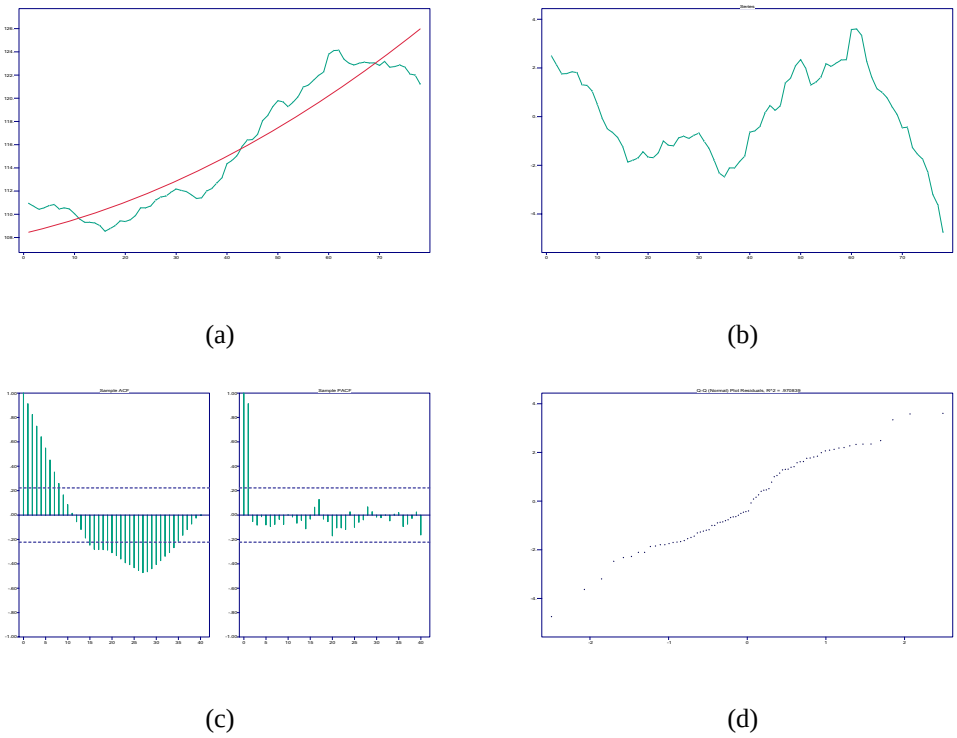


Figure 9.14: Time series in Figure 9.13 transformed with classical method: (a) polynomial fit; (b) tranformed time series; (c) ACF and PACF of transformed time series; (d) QQ-plot of transformed time series.

McLeod - Li and Turning points tests:

```
===========================================
ITSM::(Tests of randomness on residuals)
===========================================

Ljung - Box statistic = .31969E+03 Chi-Square ( 20 ), p-value
= .00000

McLeod - Li statistic = .10240E+03 Chi-Square ( 20 ), p-value
= .00000

# Turning points = 20.000~AN(50.667,sd = 3.6803), p-value =
.00000

# Diff sign points = 34.000~AN(38.500,sd = 2.5658), p-value =
.07946

Rank test statistic = .15340E+04~AN(.15015E+04,sd =
.11589E+03), p-value = .77914
```

```
Jarque-Bera test statistic (for normality) = 1.3573 Chi-Square
(2), p-value = .50731

Order of Min AICC YW Model for Residuals = 1
```

Instead of the classical method, we difference at lag 1 (Figure 9.15). The transformed time series now looks stationary. It passes some tests for iid but not all, which is compatible with the ACF and PACF showing some correlations at small lags. The normality tests does not pass, but the normal qq-plot looks OK.

```
=========================================
ITSM::(Tests of randomness on residuals)
=========================================

Ljung - Box statistic = 46.428 Chi-Square ( 20 ), p-value =
.00070

McLeod - Li statistic = 18.398 Chi-Square ( 20 ), p-value =
.56119

# Turning points = 44.000~AN(50.000,sd = 3.6560), p-value =
.10077

# Diff sign points = 37.000~AN(38.000,sd = 2.5495), p-value =
.69489

Rank test statistic = .15130E+04~AN(.14630E+04,sd =
.11368E+03), p-value = .66006

Jarque-Bera test statistic (for normality) = 6.3217 Chi-Square
(2), p-value = .04239

Order of Min AICC YW Model for Residuals = 1
```

9.7.2 AR, MA, ARMA AND ARIMA MODELS

If we are convinced that, after initial transformations, the noise is not iid, then we can apply an *ARMA* model, which is a generic family of processes. *We assume here that X_t is a 0 mean process.* This can be achieved, if necessary, by differencing, or by removing the sample mean from X.

ARMA models are called *linear models* because the noise X_t is obtained by applying a linear filter to a an iid noise sequence ϵ_t, under the generic form

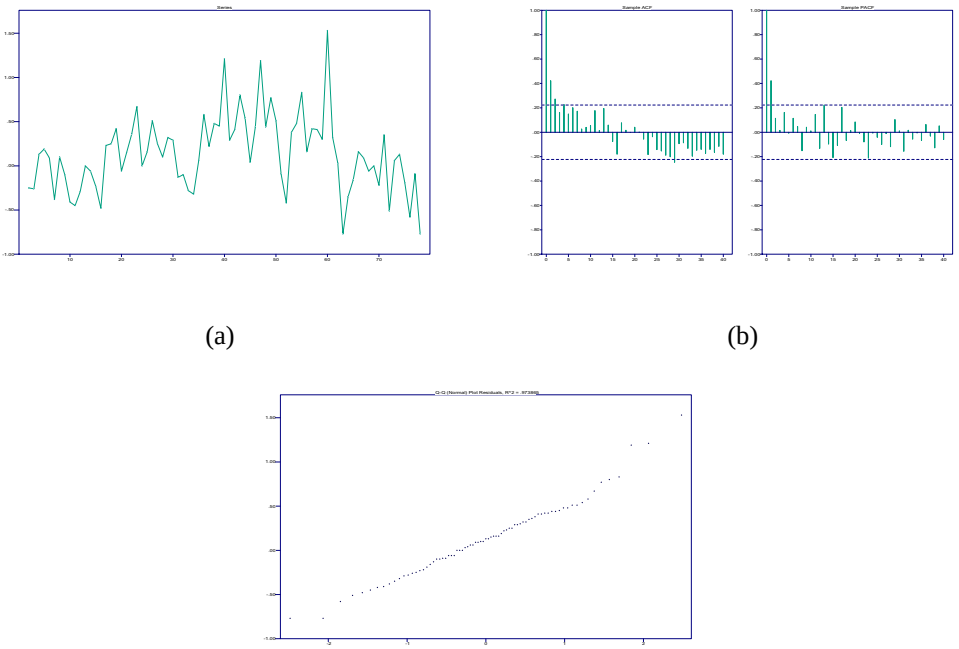
$$X_t - \sum_{r=1}^{p} \phi_r X_{t-r} = \sum_{s=0}^{q} \theta_s \epsilon_{t-s}$$

where ϵ_s is a white noise sequence with variance σ^2. We usually impose $\theta_0 = 1$.

Read [Weber-TS] Sections 1.4, 1.5 and Chapter 2 and answer the following questions.

QUESTION 9.7.4. *What is the variance of an AR(1) process ?* [20]

[20] $\frac{\sigma^2}{1-\phi_1^2}$ for the process $X_t = \phi_1 X_{t-1} + \epsilon_t$, with $\epsilon_t \sim$ WN (σ^2).



(a) (b)

(c)

Figure 9.15: Time series in Figure 9.13 transformed by differencing at lag 1. (a) tranformed time series; (b) ACF and PACF of transformed time series; (c) QQ-plot of transformed time series.

QUESTION 9.7.5. *What is the ACF of an AR(1) process ?* [21]

QUESTION 9.7.6. *What is the PACF of an AR(1) process ?* [22]

QUESTION 9.7.7. *What is the ACF of an MA(1) process ?* [23]

QUESTION 9.7.8. *What can we say about the ACF if the process is AR or MA ? Same question with PACF.* [24]

QUESTION 9.7.9. *What is Levinson-Durbin's recursion ?* [25]

QUESTION 9.7.10. *Is an ARMA process stationary ? An ARIMA process ?* [26]

PARTIAL CORRELATION An AR(p) process is a Markov chain of order p, which explains that the PACF is 0 at lags $> p$. We also know that the PACF can be computed from the inverse of the covariance matrix. For a stationary process, the Toeplitz structure of the covariance matrix allows to do this with the Levinson-Durbin algorithm presented in [Weber-TS] Section 2.6.

[21] $\rho_k = \phi_1^k$ for $k \geq 1$.

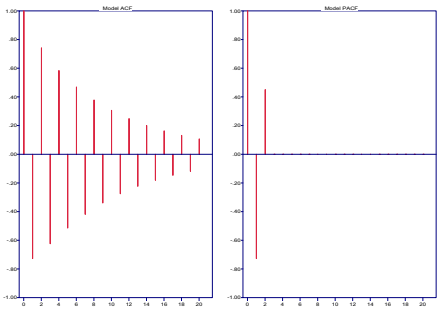
[22] $\rho_1^* = \rho_1 = \phi_1$ and $\rho_k^* = 0$for $k \geq 2$.

[23] $\rho_1 = \theta_1/(1+\theta_1^2)$, $\rho_k = 0$, $k > 1$.

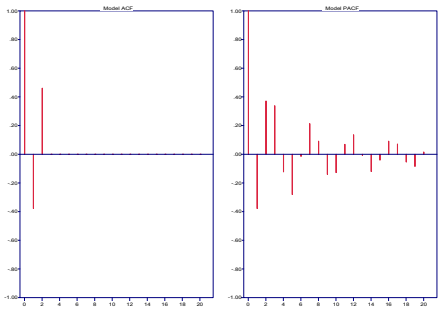
[24] The ACF of an AR(p) process decays to 0 as the lag h goes to infinity. The ACF of an MA(q) process is 0 for $h > q$. The PACF of an AR(p) process is 0 for $h > p$.

[25] An iterative algorithm to compute the PACF. For an AR model, a method to estimate the coefficients based on moment fitting.

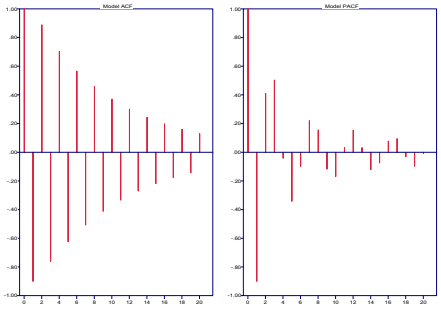
[26] ARMA yes, provided that the polynomial $\Phi(\xi)$ has all roots outside the unit disk; ARIMA no for $d \neq 1$.



(a) AR(2) $X_t = -0.4X_{t-1} + 0.45X_{t-2} + \epsilon_t$



(b) MA(2) $X_t = \epsilon_t - 0.4\epsilon_{t-1} + 0.95\epsilon_{t-2}$



(c) ARMA(2,2) $X_t = -0.4X_{t-1} + 0.45X_{t-2} + \epsilon_t - 0.4\epsilon_{t-1} + 0.95\epsilon_{t-2}$

Figure 9.16: ACF and PACF of various ARMA processes.

WOLD'S DECOMPOSITION Wold's lemma can be formalized as follows. Let $(X_t)_{t\in\mathbb{Z}}$ be a stationary, second-order time series with 0 mean, such that $\sum_k |\gamma_k|^2 < +\infty$. Define P_t as the orthogonal projector on the set of linear combinations of $X_s, s \leq t$ and the constants. In other words, for any random variable Y, $P_t Y = b + \sum_{s\leq t} c_s X_s$ and $\mathbb{E}((Y - P_t Y)^2)$ is minimum among all possible values of the coefficients b and c_s. $P_t Y$ is called the best linear estimation of Y. In signal processing, P_t is a Wiener-Hopf filter. A process is called deterministic if $X_t = P_{t-1}X_t$ for all t.

Call $Z_t = X_t - P_{t-1}X_t$ (the *innovation*). Let $\theta_j = \mathbb{E}(X_t Z_{t-j})/\mathbb{E}(Z_t^2)$. Then a general decomposition is

$$X_t = \sum_{j\in\mathbb{N}} \theta_j Z_{t-j} + V_t$$

where V_t is defined by this equation. Wold's decomposition says that the above equation is well defined, and that V_t is a deterministic process, i.e. $P_s V_t = V_t$ for all s and t. Furthermore, Z_s is a white noise sequence with common variance.

Wold's decomposition is invoked as a justification for the use of ARMA processes when the data comes from a second-order process. V_t is obtained by transforming the process into a stationary one, and an ARMA process is an approximation in the sense that rational fractions such as $\frac{\Theta(\xi)}{\Phi(\xi)}$ can approximate arbitrary power series such as $\sum_{s\in\mathbb{N}} c_s \xi^s$.

MA(Q) PROCESSES are *characterized* by the fact that $\gamma(k) = 0$ for $k > q$. among all stationary second-order processes. This is a reciprocal of [Weber-TS] Section 2.5.

SIMULATED ARMA PROCESS. Look at Figure 9.19 for a simulated sample of the process

$$X_t = -0.4X_{t-1} + 0.45X_{t-2} + \epsilon_t - 0.4\epsilon_{t-1} + 0.95\epsilon_{t-2}$$

.

9.7.3 BARTLETT'S FORMULA

This formula gives an asymptotic distribution of the sample auto-correlation function r_k for large sample sizes. It is used to test a fitted model. The distribution of $\vec{r} = (r_1, ..., r_n)$ is asymptotically normal with mean $\vec{\rho} = (\rho_1, ..., \rho_n)$ and covariance matrix W/n with $W = (w_{i,j}$ given by

$$w_{i,j} = \sum_{k \geq 1} (\rho_{k+i} + \rho_{k-i} - 2\rho_i\rho_k)(\rho_{k+j} + \rho_{k-j} - 2\rho_j\rho_k)$$

See Figure 2.15 for an example.

QUESTION 9.7.11. *What 95%-confidence interval does Bartlett's formula give for ρ_k, $k > 2$, for an MA(1) process ?* [27]

OPERATOR NOTATION

The traditional description of linear processes uses an operator notation. Call B the *back-shift operator*, defined as the one that transforms a sequence $y = (y_t)_{t\in\{1,...,n\}}$ into a new sequence By defined by $By_t = y_{t-s}$ for $t \geq 2$ and $By_1 = 0$. B is a linear mapping, $B^n = 0$. For a polynomial $P(\xi) = p_0 + p_1\xi + ... + p_1\xi^q$ we define by convention $P(B) := \sum_i p_i B^i$.

Polynomials in a fixed operator commute, i.e., for two polynomials $P(\xi)$ and $Q(\xi)$, $P(B)Q(B) = Q(B)P(B)$.

$P(B)$ is invertible iff $p_0 \neq 0$, in which case the inverse $P(B)^{-1} = \sum_{k=1}^{n}(Id - P(B))^k$ is also a polynomial in B. This justifies denoting $P(B)Q(B)^{-1}$ by $\frac{P(B)}{Q(B)}$. This is also equal to $Q(B)^{-1}P(B)$.

If we consider infinite sequences $(y_t)_{t\in\mathbb{Z}}$ the same holds provided that all zeroes of $Q(\xi)$ lie outside the unit disk. Indeed, in such a case, there exists a power series expansion $1/Q(\xi) = \sum_{i=0}^{+\infty} c_i\xi^i$ valid for $|\xi| < \eta$ with $\eta > 1$. The set of regular sequences y_t is endowed with the l_1 norm $\|y\| = \sum_i |y_i|$ and it can easily be seen that the corresponding operator norm gives $\|B\| = 1$. Thus, under the condition that all zeroes of $P(\xi)$ lie outside the unit disk, the series $\sum_{i=0}^{+\infty} c_i B^i$ converges and satisfies $Q(B)\sum_{i=0}^{+\infty} c_i B^i = Id$. Thus we can also write $P(B)Q(B)^{-1} = P(B)Q(B)^{-1} = \frac{P(B)}{Q(B)}$.

QUESTION 9.7.12. *Write dth power of the differencing operator as a polynomial in B.* [28]

QUESTION 9.7.13. *Write the lag-s differencing operator as a polynomial in B.* [29]

---

[27] $w_{k,k} = 1 + 2\rho_1^2$ thus a 95%-confidence interval for ρ_k is $\pm 1.96\sqrt{\frac{1+2\rho_1^2}{n}}$.
[28] $\Delta^d = (1 - B)^d$
[29] $\Delta_s = 1 - B^s$

With this notation, an ARIMA(p, d, q) process is defined by

$$\Phi(B)(1-B)^d Y_t = \Theta(B)\epsilon_t$$

where ϵ_t is iid, normal, with zero mean and finite variance σ^2, and

$$\begin{cases} \Phi(\xi) = 1 - \phi_1\xi - ... - \phi_p\xi^p \\ \Theta(\xi) = \theta_0 + \theta_1\xi + ... + \theta_q\xi^p \end{cases}$$

QUESTION 9.7.14. *Can an AR(p) process be represented as infinite MA ? Conversely ?* [30]

If you are comfortable with linear system theory, read Chapter 5 and answer the following questions.

QUESTION 9.7.15. *How can the auto-covariance of an ARMA process be computed ?* [31]

CONVENTION ON Φ AND Θ. We assume that

- $\Phi(0) = \Theta(0) = 1$. There is no loss of generality, as we can modify σ
- The zeroes of $\Phi(\xi)$ lie outside the unit disk; this is to guarantee that Y_t is stationary when we extrapolate the model to infinite sequences.
- The zeroes of $\Theta(\xi)$ should also lie outside the unit disk; this is to guarantee that ϵ_t is identifiable.
- Both polynomials should have no zero in common; this is also to guarantee that the model is identifiable.

9.7.4 THE BOX-JENKINS METHOD

It is a direct application of the scientific method ! Read [Weber-TS] Sections 7.1 and 7.2.

FITTING AN ARIMA MODEL

Once orders p, d, q are chose, we apply the general principle of MLE and maximize the log-likelihood of the sample. This is a non-linear optimization problem and only approximate solutions exist. The log-likelihood of an ARMA model is

Therefore, some heuristics are often used, with the goal of starting the optimization procedure from an initial value which is not too far from the optimal. For this, we can use an AR model or MA model to obtain an approximate solution first.

[30] Any ARMA process with our convention can be represented as an MA(∞) or AR(∞) process. For the former, see [Weber-TS]. For the latter, write $\frac{\Phi(\xi)}{\Theta(\xi)}$ as a power series with a convergence radius larger than 1:

$$\frac{\Phi(\xi)}{\Theta(\xi)} = 1 + \sum_{i=1}^{+\infty} e_i\xi^i$$

and the ARMA process can be represented as an AR(∞) process:

$$X_t = \sum_{i=1} e_i X_{t-i} + \epsilon_t$$

[31] Expand $\frac{\Theta(\xi)}{\Phi(xi)}$ as a power series $\sum_{n=0}^{+\infty} c_n\xi^n$. The auto-covariance is $\gamma_t = \sum_{n=0}^{+\infty} c_n c_{n+t}\sigma^2$.

SPECIAL CASES.

AR models can be fitted using a *Moment Fitting* heuristic. The idea is based on the observation that the autoregressive coefficients θ_k for $k = 1, ..., p$ are uniquely determined by the auto-covariance function (if the auto-covariance matrix is non-singular) via the *Yule-Walker equations*:

$$\begin{cases} \gamma_k = \sum_{i=1}^{p} \phi_i \gamma_{|k-i|} \text{ for } k = 1...p \\ \sigma^2 = \gamma_0 - \sum_{i=1}^{p} \phi_i \gamma_i \end{cases}$$

The moment fitting method consists in replacing the ACF by the sample ACF in the Yule Walker equations. This gives an estimator which is not the same as the MLE, but is also asymptotically bias-free and consistent (its variance goes to 0).

The Yule-Walker equations can be solved using the iterative method called *Levinson-Durbin* recursion, well known in signal processing, and given in [Weber-TS] Section 2.6. Note that the moment fitting method is equivalent to finding the AR coefficients ϕ_1 that minimize $(X_t - \sum_{i=1}^{p} \phi_i X_{t-i})^2$, a problem known as finding Wiener filtering (see [Thiran02-LN]).

Another method for AR models is Burg's method; for MA models, there also exist some non-MLE methods that are numerically simple: the *innovation* and *Hannan-Rissanen* algorithms (see [BrockwellDavis02-book]). For MA models, these methods give non-consistent estimators.

Read [Weber-TS] Chapter 7 and answer the following questions.

QUESTION 9.7.16. *What can we say about the sample ACF and PACF of an ARMA(p, q) process ?* [32]

QUESTION 9.7.17. *What is AIC ? What is it used for ?* [33]

QUESTION 9.7.18. *How can a confidence interval for the estimated model parameters be obtained ?* [34]

QUESTION 9.7.19. *What tests are performed to validate the model ?* [35]

EXAMPLES

APPLICATION TO EXAMPLE 9.4 ON PAGE 219. We applied the Box-Jenkins procedure to Example 9.4 on page 219 except for the last 10 data points. We looked for the ARMA model for model lagged-1 differenced data that has the least AIC, among all ARMA(p, q) models with $p, q \leq 10$. The result is an AR(1) model, as shown below and in Figure 9.17. The tests on residuals all pass except the test for normality.

```
========================================
ITSM::(Maximum likelihood estimates)
========================================
```

[32] They decay exponentially for lags $> \max(p, q)$.

[33] AIC is $-2\times$ the log-likelihood, plus $2\times$ the model order. It is used to select a model among many; it tries to avoid overfitting.

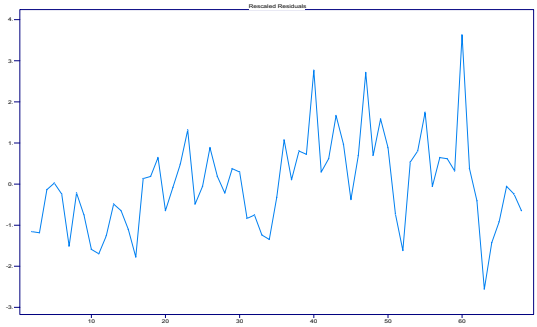
[34] For large samples, confidence Intervals for the maximum likelihood estimators of the model can be found using the Fisher information matrix, which can be computed but is complex to describe. See [BrockwellDavis02-book] Section 5.2 for some details.

[35] The residuals are tested for independence and normality.

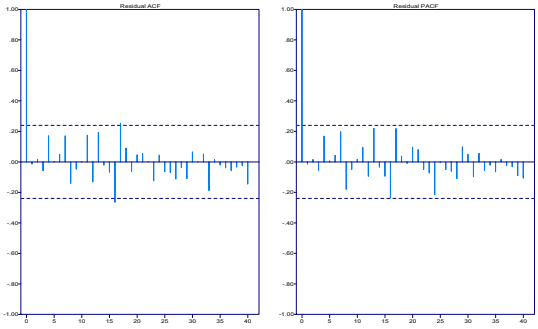
```
Method: Maximum Likelihood

ARMA Model:
X(t) = .4475 X(t-1)
     + Z(t)

WN Variance = .138317

AR Coefficients
       .447470

Standard Error of AR Coefficients
       .109032

(Residual SS)/N = .138317

AICC = 62.008739
BIC  = 61.422006
FPE  = .142508

-2Log(Likelihood) = 57.821239

=========================================
ITSM::(Tests of randomness on residuals)
=========================================

Ljung - Box statistic = 27.996 Chi-Square ( 20 ), p-value = .10950

McLeod - Li statistic = 18.682 Chi-Square ( 21 ), p-value = .60551

# Turning points = 41.000~AN(43.333,sd = 3.4042), p-value = .49308

# Diff sign points = 33.000~AN(33.000,sd = 2.3805), p-value = 1.00000

Rank test statistic = .12360E+04~AN(.11055E+04,sd = 92.395), p-value = .15783

Jarque-Bera test statistic (for normality) = 11.917 Chi-Square (2), p-value = .00258

Order of Min AICC YW Model for Residuals = 0
```

Then we looked for the ARMA model for model lagged-2 differenced data that has the least AIC, among all ARMA(p, q) models with $p, q \leq 10$. The result is an AR(1,1) model, as shown below and in Figure 9.18. The tests on residuals all pass. This may be interpreted as a better model.

```
=======================================
ITSM::(Maximum likelihood estimates)
=======================================

Method: Maximum Likelihood


ARMA Model:
X(t) = .3772 X(t-1)
     + Z(t) - .9297 Z(t-1)

WN Variance = .140684
```



(a) Residuals.



(b) ACF and PACF of residuals.

Figure 9.17: Residuals of best ARMA model for lagged-1 differenced time series in Example 9.4 on page 219 (Dow Jones).

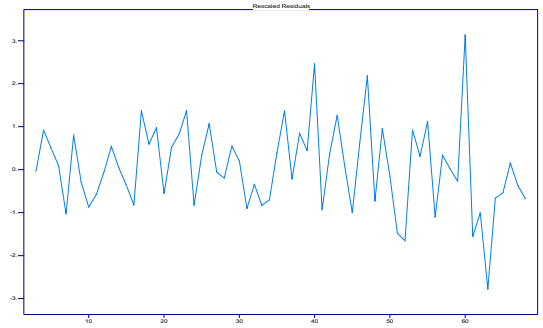
```
AR Coefficients
       .377236

Standard Error of AR Coefficients
       .166161

MA Coefficients
      -.929743

Standard Error of MA Coefficients
       .112450

(Residual SS)/N = .140684

AICC = 65.535640
BIC  = 64.791045

-2Log(Likelihood) = 59.148544

=========================================
ITSM::(Tests of randomness on residuals)
=========================================

Ljung - Box statistic = 27.726 Chi-Square ( 20 ), p-value = .11606

McLeod - Li statistic = 15.466 Chi-Square ( 22 ), p-value = .84148

# Turning points = 41.000~AN(42.667,sd = 3.3780), p-value = .62174

# Diff sign points = 32.000~AN(32.500,sd = 2.3629), p-value = .83242

Rank test statistic = .95200E+03~AN(.10725E+04,sd = 90.349), p-value = .18230

Jarque-Bera test statistic (for normality) = 3.3661 Chi-Square (2), p-value = .18581

Order of Min AICC YW Model for Residuals = 0
```

9.7.5 FORECASTING

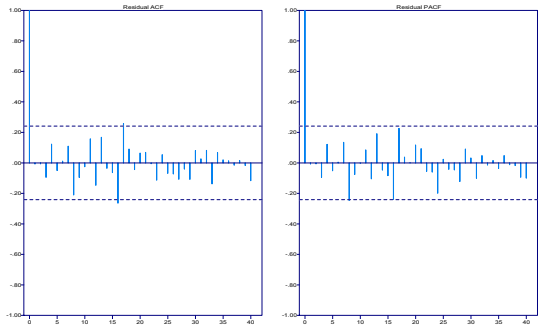
FORECASTING WITH A GENERAL SECOND-ORDER STATIONARY PROCESS

Consider a general second order stationary process. Assume that $E(X_t) = 0$ (if this is not the case, we assume that the mean μ is known and replace X_t by $X_t - \mu$). We want to compute an h-step ahead forecast $\hat{X}_t(h) := (X_{t+h}|X_1, ..., X_t)$. In general, the best one, in least square sense, is the conditional expectation of X_{t+h} given $X_1, ..., X_t$. In practice, it is hard to find except for normal processes. We take instead the *best linear predictor* $\hat{X}_t(h) = P_t X_{t+h}$, defined as the linear combination of $X_1, ..., X_t$ and constants that minimize the mean square forecast error $\mathbb{E}((\hat{X}_t(h) - X_{t+h})^2)$.

THEOREM 9.7.1. *Consider a second-order stationary process X_t with zero mean. Let γ_k be the auto-covariance of X_t at lag k. Let $\Omega(t)$ be the covariance matrix of the vector $(X_1, ..., X_t)^T$, i.e. $\Omega(t)$ is the $t \times t$ symmetric Toeplitz matrix defined by $\Omega(t)_{i,j} = \gamma(|i - j|)$. Assume that $\Omega(t)$ is invertible.*



(a) Residuals.



(b) ACF and PACF of residuals.

Figure 9.18: Residuals of best ARMA model for lagged-2 differenced time series in Example 9.4 on page 219 (Dow Jones).

1. *The best linear predictor at time t and horizon h is*

$$\hat{X}_t(h) = \sum_{i=1}^{t} u_i(h,t) X_i \tag{9.4}$$

with

$$u(h,t) := (u_1(h,t), ..., u_t(h,t)) = (\gamma_{t+h-1}, ..., \gamma_h)\Omega(t)^{-1}$$

2. *The mean square prediction error $MSE_t(h) := \mathbb{E}\left((X_{t+h} - \hat{X}_t(h))^2\right)$ is given by*

$$MSE_t(h) = \gamma_0(1 - R_t^2(h))$$

with $R_t^2(h)$ defined by

$$\gamma_0 R_t^2(h) := (\gamma_{t+h-1}, ..., \gamma_h)\Omega(t)^{-1}(\gamma_{t+h-1}, ..., \gamma_h)^T$$

3. *If the process is gaussian, then $(X_{t+h} - \hat{X}_t(h)) \sim N(0, MSE_t(h))$*

PROOF: Item 1: from the properties of orthogonal projection (Chapter **??**), the coefficients $u_i(h)$ are obtained by expressing that $\mathbb{E}((\hat{X}_t(h) - X_{t+h})X_j) = 0$ for all $j \leq t$.

Item 2: first note that by Pythagoras

$$\text{MSE}_t(h) = \mathbb{E}(X_{t+h}^2) - \mathbb{E}((u(h,t)(X_1, ..., X_t)^T)^2)$$

and by Section 12.5.1:

$$\text{MSE}_t(h) = \gamma_0 - u(h,t)\Omega(t)u(h,t)^T$$

thus

$$\text{MSE}_t(h) = \gamma_0 - (\gamma_{t+h-1}, ..., \gamma_h)\Omega(t)^{-1}(\gamma_{t+h-1}, ..., \gamma_h)^T$$

as required.

Item 3. If the process is gaussian, then by linearity, $X_{t+h} - \hat{X}_t(h)$ is a centered normal random variable. □

REMARKS.

- The forecast at time t depends on the complete past sequence $X_1 \dots X_t$ and the coefficients $u_i(h,t)$ also depend on t. However, for large t, they can be replaced by their limits (see more details for ARMA processes below).
- As $h \to +\infty$, $\hat{X}_t(h) \to 0$ (the process mean). Further, $R_t^2(h) \to 0$ and $\text{MSE}_t(h) \to \gamma_0$ (the process variance).
- If $\Omega(t)$ is not invertible, more complex formulae exist. We will not need them, since for a stationary ARMA process with 0 mean, $\Omega(t)$ is always invertible
  QUESTION 9.7.20. *Prove this statement.* [36]

[36] From Section 12.5.3, the dimension of the space generated by $X_1, ..., X_t$ is the rank of Ω_t. Now X_t is a linear combination of $X_1, ..., X_{t-1}$ and ϵ_t (MA representation of the ARMA process). ϵ_t is orthogonal to $X_1, ..., X_{t-1}$, thus the dimension of the space generated by $X_1, ..., X_t$ is 1 plus the dimension of the space generated by $X_1, ..., X_{t-1}$.

- We interpret $R_t^2(h)$ as the proportion of variance which is predictable. Note that necessarily $0 \leq R_t^2(h) \leq 1$. For a non-correlated process, we have $R_t^2(h) = 0$ and no prediction is possible.
- Instead of the above formula, the coefficients $u_i(h,t)$ can be computed more efficiently using the *innovation algorithm*. We give the details of this algorithm in the case of ARMA processes in the following sections.
- There exist similar formulae for a non-stationary process [BrockwellDavis02-book].

FORECASTING WITH AN AR MODEL

For an AR process the general method gives simple formulae. By a direct application of the method in Section 9.7.5, we find that the one-step ahead forecast is $\hat{X}_t(1) := (X_{t+1}|X_1, ..., X_t)$ is

$$\hat{X}_t(1) = \sum_{i=1}^{p} \phi_i X_{t+1-i}$$

and the 1-step ahead mean square error is

$$\text{MSE}_t(1) = \sigma^2$$

The variance non explained by the prediction is that of the white noise.

QUESTION 9.7.21. *What is $R_t(1)$ for an AR(1) process ?* [37]

FORECASTING WITH AN ARMA MODEL

For an ARMA process, the method in Section 9.7.5 can be made recursive by the innovation algorithm. It is simpler to compute, and is incremental, giving new forecasts as new data becomes available. It computes the one step ahead forecast as a function of past forecast errors [BrockwellDavis02-book].

ONE-STEP AHEAD PREDICTION.

$$\begin{cases} \hat{X}_t(1) = \sum_{j=1}^{t} \theta_{t,j} \left(X_{t+1-j} - \hat{X}_{t-j}(1) \right) \text{ for } 1 \leq n < \max(p,q) \\ \hat{X}_t(1) = \sum_{i=1}^{p} \phi_i X_{t+1-i} + \sum_{j=1}^{t} \theta_{t,j} \left(X_{t+1-j} - \hat{X}_{t-j}(1) \right) \text{ else} \end{cases} \tag{9.5}$$

where $\theta_{s,t}$ are computed by solving for $\theta_{s,.}, \nu_s$ in the equations

$$\begin{cases} r_0 = \Gamma_{1,1} \\ r_k \theta_{t,t-k} = \left(\Gamma_{t+1,k+1} - \sum_{j=0}^{k-1} \theta_{k,k-j} \theta_{t,t-j} r_j \right) \text{ for } 0 \leq k \leq n \\ r_t = \Gamma_{t+1,t+1} - \sum_{j=0}^{t-1} \theta_{t,t-j}^2 r_j \end{cases} \tag{9.6}$$

and Γ is the covariance matrix of $\sigma^{-2}\Phi(B)X_t$, given by

$$\sigma^2 \Gamma_{i,j} = \begin{cases} \gamma(i-j) \text{ for } 1 \leq i,j \leq \max(p,q) \\ \gamma(i-j) - \sum_{r=1}^{p} \phi_r \gamma_{r-|i-j|} \text{ for } \min(i,j) \leq \max(p,q) < \max(i,j) \leq 2\max(p,q) \\ \sum_{r=0}^{q} \theta_r \theta_{r+|i-j|} \text{ for } \max(p,q) < \min(i,j) \\ 0 \text{ else} \end{cases} \tag{9.7}$$

[37] $\frac{|\phi_1|}{\sqrt{1-\phi_1^2}}$

An iterative solution of Equation (9.6) is straightforward, solving in this order for $\theta_{1,1}, r_1, \theta_{2,.}, r_2, ...$ The 1-step ahead mean square error is

$$\text{MSE}_t(1) = \sigma^2 r_t$$

For predictions based on large sample sizes, the values can be approximated by their limits:

$$\begin{cases} \lim_{t\to+\infty} \theta_{t,j} = \theta_j \\ \lim_{t\to+\infty} r_t = 1 \end{cases}$$

This corresponds to backcasting, at the end of [Weber-TS] Section 7.6. Alternatively, we can derive the forecasting equations from an infinite past from the AR(∞) representation of an ARMA process. From Question 9.7.14 we can write $X_t = \sum_{i=1} e_i X_{t-i} + \epsilon_t$. The forecast of X_t based on the infinite past before t is then $\sum_{i=1} e_i X_{t-i}$.

h-STEP AHEAD PREDICTION. The h-step ahead forecast $\hat{X}_t(h) := (X_{t+h}|X_1, ..., X_t)$ is given by

$$\hat{X}_t(h) = \begin{cases} \sum_{j=h}^{t+h-1} \theta_{t+h-1,j}(X_{t+h-j} - \hat{X}_{t+h-j-1}(1)) \text{ for } t+h \leq \max(p,q) \\ \sum_{i=1}^{p} \phi_i \hat{X}_t(t+h-i) + \sum_{j=h}^{q} \theta_{t+h-1,j}(X_{t+h-j} - \hat{X}_{t+h-j-1}(1)) \text{ else} \end{cases} \tag{9.8}$$

The h-step ahead mean square error is

$$\text{MSE}_t(h) = \sigma^2 \sum_{j=0}^{h-1} \left(\sum_{r=0}^{j} \chi_r \theta_{t+h-r,j-r} \right)^2 r_{t+h-j-1} \tag{9.9}$$

where $\chi_0 = 1$ and $\chi_j = \sum_{k=1}^{\min(p,j)} \phi_k \chi_{j-k}$. For large sample sizes, we have

$$\lim_{t\to+\infty} \text{MSE}_t(h) = \sigma^2 \sum_{j=0}^{h-1} c_j^2$$

where $X_t = \sum_{j=0}^{+\infty} c_j \epsilon_{t-j}$ is the MA($+\infty$) representation of the ARMA process, i.e., the coefficients of the Taylor series expansion of $\frac{\Theta(\xi)}{\Phi(\xi)}$ around $\xi = 0$.

REMARK. The confidence intervals obtained with this method do not account for the uncertainty on the parameters ϕ_i, θ_i, which usually have to be estimated. There does not seem to be a simple way to account for both in this framework.

NUMERICAL EXAMPLE. Figure 9.19 shows a numerical example. The values of $\hat{X}_t(h)$ and $\text{MSE}_t(h)$, with $t = 100$ and $h = 1 \ldots 25$ are given below. We see that the forecast rapidly converges towards the mean (0) of the process, while the mean square prediction error converges from the white noise variance towards the variance of the process.

```
====================
ITSM::(ARMA Forecast)
```

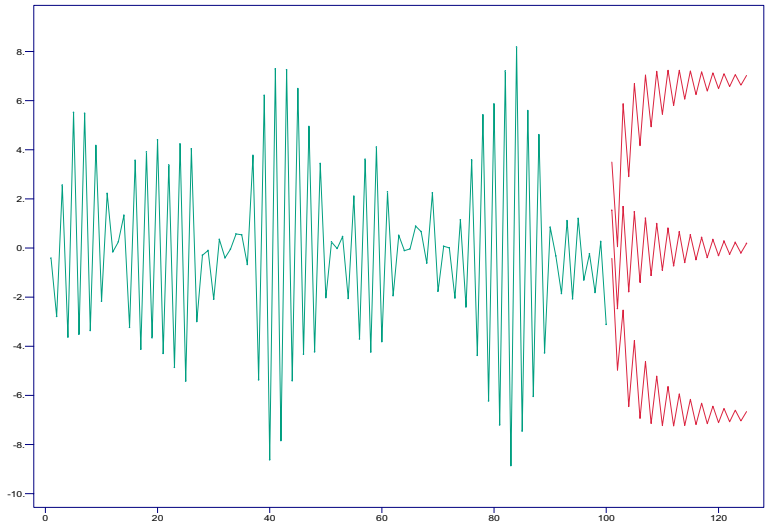


Figure 9.19: Simulated ARMA(2,2) process in Figure 9.16, with white noise variance $\sigma^2 = 1$. Right part: prediction with confidence interval.

```
====================
```

			Approximate 95 Percent Prediction Bounds	
Step	Prediction	sqrt(MSE)	Lower	Upper
1	1.53498	1.00028	-.42553	3.49550
2	-2.46289	1.28092	-4.97344	.04766
3	1.67590	2.14444	-2.52712	5.87892
4	-1.77866	2.38684	-6.45678	2.89945
5	1.46562	2.66841	-3.76438	6.69562
6	-1.38664	2.83212	-6.93750	4.16421
7	1.21419	2.97669	-4.62001	7.04838
8	-1.10966	3.08067	-7.14766	4.92833
9	.99025	3.16593	-5.21486	7.19536
10	-.89545	3.23178	-7.22961	5.43872
11	.80379	3.28484	-5.63439	7.24197
12	-.72447	3.32689	-7.24505	5.79611
13	.65149	3.36071	-5.93539	7.23837
14	-.58661	3.38778	-7.22654	6.05332
15	.52782	3.40960	-6.15488	7.21051
16	-.47510	3.42714	-7.19217	6.24197
17	.42756	3.44130	-6.31726	7.17238
18	-.38482	3.45272	-7.15202	6.38239
19	.34633	3.46194	-6.43895	7.13160
20	-.31170	3.46940	-7.11160	6.48820
21	.28053	3.47542	-6.53117	7.09222
22	-.25248	3.48030	-7.07374	6.56879
23	.22723	3.48424	-6.60175	7.05620
24	-.20450	3.48743	-7.03974	6.63073
25	.18405	3.49001	-6.65624	7.02435

FORECASTING WITH AN ARIMA MODEL

Consider an ARIMA process Y_t: $\Phi(B)(1-B)^d Y_t = \Theta(B)\epsilon_t$. Call $X_t = (1-B)^d Y_t$ the underlying ARMA process. The differencing filter can be inverted to

$$Y_t = X_t - \sum_{j=1}^{d} \left(\begin{array}{c} d \\ j \end{array} \right) (-1)^j Y_{t-j}$$

from where the predictor equations follow:

$$\hat{Y}_t(h) = \hat{X}_t(h) - \sum_{j=1}^{d} \left(\begin{array}{c} d \\ j \end{array} \right) (-1)^j \hat{Y}_t(h-j)$$

The predictors can be recursively computed from the above, taking into account that $\hat{Y}_t(h-j) = Y_{t+h-j}$ for $h - j \leq 0$. It follows that $\hat{Y}_t(1) - Y_t = \hat{X}_t(1) - X_t$. This can be used to get innovation formulae, similar to the ARMA case:

$$\hat{Y}_t(h) = \sum_{j=1}^{p+d} \phi_j^* \hat{Y}_t(h-j) + \sum_{j=h}^{t+h-1} \theta_{t+h-1,j}(Y_{t+h-j} - \hat{Y}_{t+h-j-1}(1)) \tag{9.10}$$

where $\theta_{s,j}$ is defined by Equation (9.6) and ϕ_j^* is the jth coefficient of $\Phi^*(\xi) := (1-\xi)^d \Phi(\xi)$.

Confidence intervals are computed with formulae similar to the Equation (9.9), with ϕ replaced by ϕ^*.

For Seasonal ARIMA models, there are analog formulae, see [BrockwellDavis02-book] Section 6.5.1.

QUESTION 9.7.22. *Consider the forecasting equation $\hat{X}_t(1) = 1/p(X_t + ... + X_{t-p+1})$. What process model does this correspond to ?* [38]

APPLICATION TO EXAMPLE 9.4 ON PAGE 219.

We applied the two ARIMA models (with differencing at lags $d = 1$ and $d = 2$) for the Dow Jones trace identified in Section 9.7.4. The models are identified over the first 68 data values. The forecasts for times 69 to 78 are plotted on Figure 9.20, together with the actual data. Both models give similar forecasts, and both have the actual data in the 95% confidence intervals.

[38] It is the forecasting equation for the AR(p) model with polynomial $\Phi(\xi) = 1 - 1/p \sum_{i=1}^{p} \xi^p$. However, Φ has root 1, thus this AR model is not stationary and does not fit our framework. Since 1 is root of Φ, we can factor by $1 - \xi$ and write

$$\Phi(\xi) = (1-\xi)\Psi(\xi)$$

After some algebra it comes

$$\Psi(\xi) = cst \times (1 - \sum_{i=1}^{p-1} u_i \xi^i)$$

with $u_i = (-1)^{i+1} \frac{2}{(i+1)(i+2)} \left(\begin{array}{c} p-1 \\ i \end{array} \right)$. It can be seen that Φ does not have any other root in the unit disk than 1, by convexity arguments, and 1 is a root with multiplicity 1 (because $\Phi'(1) \neq 0$). Thus Ψ has no root in the unit disk. By Equation (9.10), we have the forecasting equation of an ARIMA($(p-1, d, 0)$ model with regression coefficients u_i.

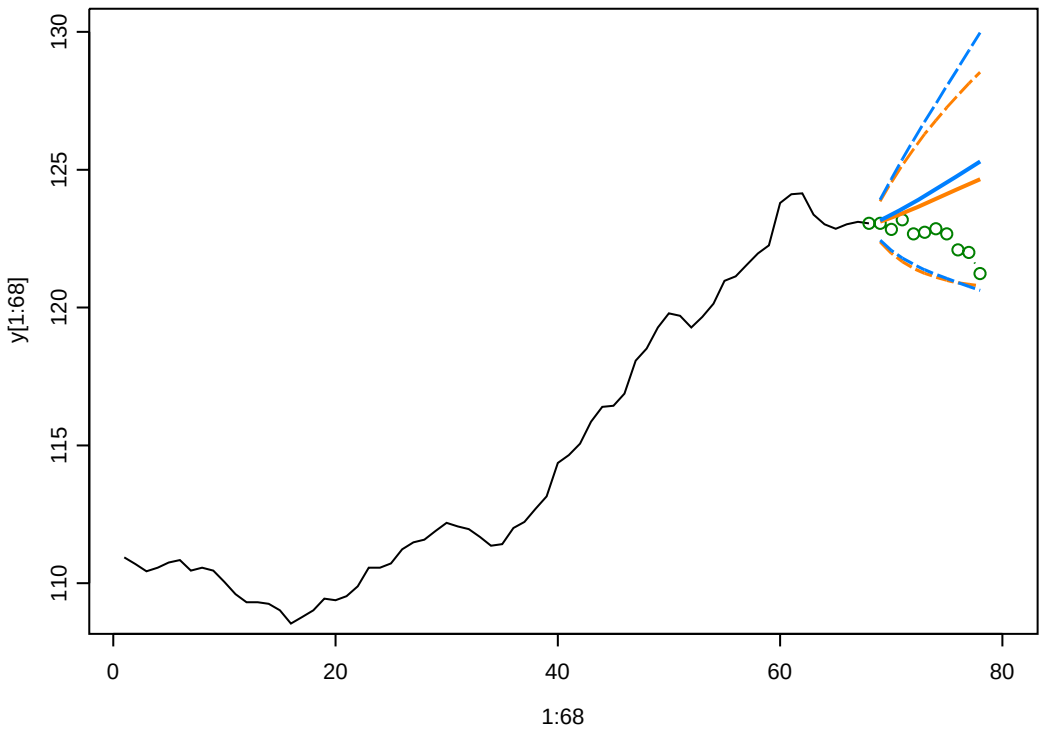


Figure 9.20: Forecasts obtained by both ARIMA models ($d = 1$, $d = 2$) for Example 9.4 on page 219 (Dow Jones), with confidence intervals. The ARIMA forecasts with $d = 1$ are slightly less than for $d = 2$. The actual data, not used at time of forecasting, is shown with circles

VALIDATION

As usual, the model must be validated if confidence intervals based on the normal noise assumption are used. Validation uses the tests in Section 9.7.1. See Section 9.7.4 for some examples

There is a fundamental difference with the principles applied in Chapter 2. There, we want confidence intervals for the parameters of a model assumed to explain the data. The confidence intervals, and the resulting diagnostic, are valid only if the model assumption are correct.

In contrast, when we forecast some data, the formulae for prediction are valid as long as the noise is white, even if the noise is not normal. If the noise is not normal, they do not correspond to the best predictors, but simply to the best linear predictors, as explained above. The variance of the prediction error is still valid, but cannot be used to obtain confidence intervals from the normal distribution.

The following method can be used to obtain confidence intervals. First, note that prediction based on large samples, the prediction formulae can be interpreted as follows: the noise ϵ_t is estimated by the one-step prediction error $r_t = X_t - \hat{X}_t(1)$, also called residuals. Second, the distribution of the noise can be approximated by the empirical, observed distribution of the residuals. See [Basu96-infocom] for an example with non gaussian noise.

One can even go further by using the *bootstrapping* methods. This method consists in sampling random values from the distribution of the residuals, in order to obtain a confidence interval for the prediction. Consider Equation (9.8), which, for large sample sizes, can be written as

$$\hat{X}_t(h) = \sum_{i=1}^{p} \phi_i \hat{X}_t(h-i) + \sum_{j=h}^{q} \theta_j (X_{t+h-j} - \hat{X}_{t+h-j}(1)) = \sum_{i=1}^{p} \phi_i \hat{X}_t(h-i) + \sum_{j=h}^{q} \theta_j r_{t+h-j}$$

Assume the residuals r_t are iid. We can easily check this with the standard methods. The distribution of r_t is independent of t; we obtain its empirical value by keeping a database of all values r_t (there are N such values). Now we do a simulation as follows. We pick N numbers out of the database, *with replacement*, and re-construct the time series and the forecast, using the ARMA definition, starting from arbitrary initial values, and the prediction equation. We repeat this M times; this gives M values of the forecast. We compute the lower and upper percentiles of this set of M values and use them as confidence interval.

9.7.6 SEASONAL ARIMA MODELS

ARIMA models are able to fit seasonal behaviour, depending on the ACF. See Figure 9.19 for an example. Thus, we can model time series with seasonal components with ARIMA processes. An MA(q) process can thus model a time series with period q.

However, if the period is not very small, we may need a large model order, which is not good. A Seasonal ARIMA can be used instead. It is a subset of ARIMA, where we impose constraints on the parameters, in order to reach high model orders, while having few parameters in total. The general model, called *Seasonal ARIMA* or *Multiplative ARIMA*, with parameters (p, d, q, P, D, Q, s) is

$$\Phi(B)\Psi(B^s)(1-B)^d(1-B^s)^D Y_t = \Theta(B)\Lambda(B^s)\epsilon_t$$

where Φ, Θ, Ψ and Λ are polynomials of degree p, q, P and Q. This allows to model the process as a superposition of a seasonal components with period s and a general trend component.

In order to apply Seasonal ARIMA model, the Box-Jenkins method needs to be complemented with

- identification of period s
- make the process stationary by differencing at lags $1, 2, ..$ and $s, 2s, ...$

Seasonal ARIMA models allow for randomness in both the seasonal pattern, unlike the classical method approach based on linear regression.

APPLICATION TO EXAMPLE 9.2 ON PAGE 203 The ACF and PACF of the Sprint time series (Figure 9.21) show some correlation at lags up to 5, around 16 and 32. This suggests a SARIMA model with $p, q \leq 5$, $P, Q \leq 2$, $d = D = 1$. We fitted on the first 224 data points, using the AIC criterion, and obtained that the best model is for $p = 4, q = 0, P = 2, Q = 2$. The resulting forecasts are shown on Figure 9.22. Compare to the model based on white noise given in Figure 9.7: the SARIMA model fits slightly better and gives a smaller confidence interval. The model diagnostic on the figure shows that the residues do not pass the test for normality (p-value in Box-Ljung portmanteau test is small) and there is one large residual correlation at lag ≈ 140. Thus the model cannot be invoked as an explanation for the data, but it may be used for forecasting.

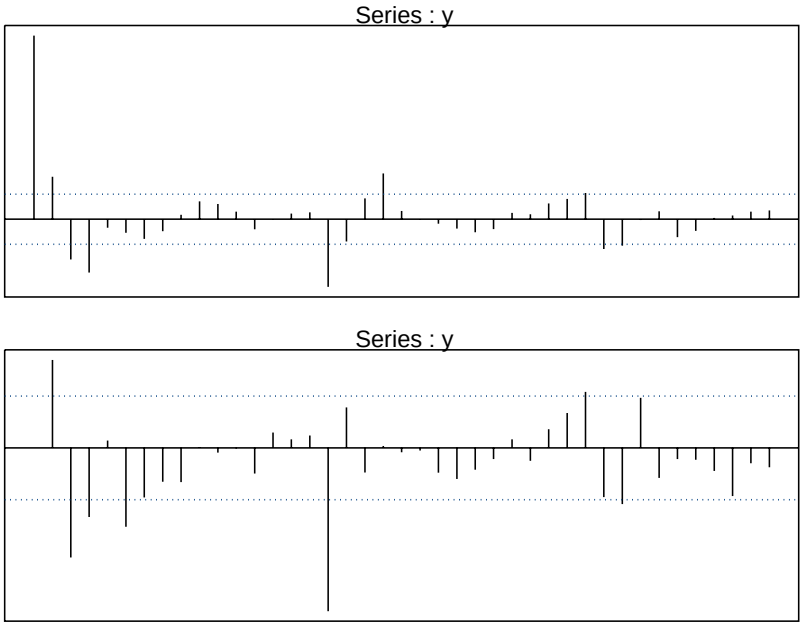
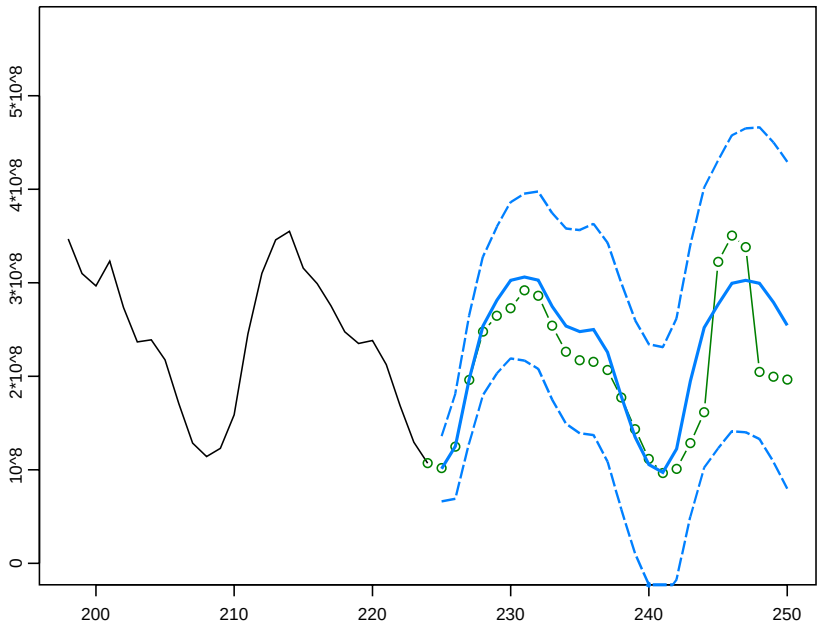
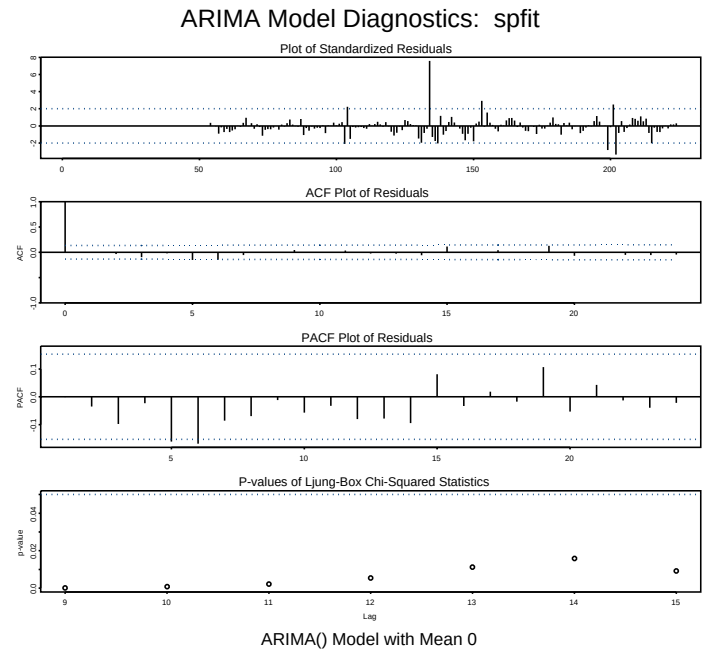


Figure 9.21: ACF and PACF of Sprint data (Example 9.2 on page 203), differenced at lags 1 and 16.



(a) forecasts



(b) model diagnostic

Figure 9.22: SARIMA model with best AIC for Example 9.2 on page 203 (Sprint).

9.8 CASE STUDIES

9.8.1 WEB SITE PLANNING

CAPACITY PLANNING

The problem is to plan for adequate capacity. Here, in addition to a prediction model, as defined above, we need a *capacity model*, which links the predicted load to some required hardware and software. A capacity model can be derived from a queuing or bottleneck analysis of the system.

For example, assume you are planning a video on demand center and we look in detail at the server hardware. A prediction model gives us the forecast penetration in number of residential and business users. A capacity model could be as described in Chapter **??**:

> For your video on demand application, the number of required servers is given by $N_1 = \lceil \frac{R}{59.3} + \frac{B}{3.6} \rceil$ and the number of disk units by $N_2 = \lceil \frac{R}{19.0} + \frac{B}{2.4} \rceil$, where R [resp. B] is the number of residential [resp. business] customers.

WEB SITE CAPACITY PLANNING

Read [Gunther01-LNCS] and answer the following questions.

QUESTION 9.8.1. *What is the performance metric ? How is it measured ? Were there any difficulties ?* [39]

QUESTION 9.8.2. *What data transformations are applied ?* [40]

QUESTION 9.8.3. *What is the prediction model ?* [41]

QUESTION 9.8.4. *What is the window used to produce a forecast* [42]

QUESTION 9.8.5. *What is the capacity model ?* [43]

QUESTION 9.8.6. *How are both models validated ?* [44]

QUESTION 9.8.7. *What is the doubling period ?* [45]

QUESTION 9.8.8. *What confidence intervals are given ?* [46]

---

[39] CPU utilization, measured every "few minutes". It was collected using a data collection tool installed by the server vendor. Another system was put in place by the site operators, but it aggregated all data into 8 hour summaries, which made it impossible to do peak dimensioning.

[40] First, the *effective server demand* is derived. It is defined as the hypothetical CPU utilization if there would be enough capacity. The idea is to use indicators of non saturated resources, model the CPU utilization as a breakpoint + linear model (as we did with Example 4.3 on page 94), and keep only the first linear part. It is deduced from a linear regression model, with 6 unspecified factors. Further research in the references tells us that these factors are related to queue lengths. Second, the peak effective server utilization is used, which leaves one data point per day

[41] $\log Y_t = \log Y_0 + b(t - t_0) + \epsilon_t$. An implicit assumption is that ϵ_t is centered normal iid.

[42] 5 weeks.

[43] The number p of CPUs is related to the effective demand C by

$$C(p) = \frac{p}{1 + \sigma(p-1)(1 + \lambda p)}$$

where σ and λ are parameters, estimated from previous experience, that account for contention and stale cache delays.

[44] The prediction model is not validated. The capacity model seems to be validated in previous references.

[45] The time it takes to double the load, according to the prediction model. Here: $\frac{\log b}{2}$.

[46] None.

9.8.2 SOFTWARE REJUVENATION

SOFTWARE AGING

Due to imperfections, some (if not all) server programs do not always release all resources they used, such as memory, kernel data objects ("committed bytes"), or processes in zombie state. As such programs are meant to be running by all times, it is not possible to simply restart when needed as you do with you PC. Restarting the program (*rejuvenation*) is the solution, but it has a cost (service interruption), therefore it should be performed only when necessary. Some software aging systems predict when a restart is necessary.

The framework is the same as with capacity planning, except that a capacity model is not needed. Instead, we use exhaustion thresholds.

PROACTIVE MANAGEMENT OF SOFTWARE AGING

Read [Castelli01-IBM] Sections 1, 3 from "Predictions Algorithm" to Appendix A. Then answer the following questions.

QUESTION 9.8.9. *What is the performance metric ?* [47]

QUESTION 9.8.10. *What transformation is applied to the data ?* [48]

QUESTION 9.8.11. *What is the prediction model ?* [49]

QUESTION 9.8.12. *How is a model selected ?* [50]

QUESTION 9.8.13. *Why is the breakpoint model appropriate ?* [51]

QUESTION 9.8.14. *Are there implicit assumptions in the model ?* [52]

QUESTION 9.8.15. *Is the model validated ?* [53]

QUESTION 9.8.16. *What is the window used for prediction ?* [54]

9.8.3 DYNAMIC LOAD SCHEDULING IN DISTRIBUTED SYSTEMS

Read [Dinda99-HPDC] and answer the following questions.

[47] Memory (available bytes), committed bytes, used I-nodes.

[48] The data is smoothed by an ad-hoc filter: the medians over non overlapping time windows are taken. This removes outliers and reduces the size of the time series.

[49] There are six models: linear regressions with 1, 2 or 3 breakpoints; same with the log of the data. The breakpoint models are the same as Example 3 in Chapter **??**.

[50] Two methods are presented: one is Mallow's C_p, the other is ad-hoc (based on the prediction capability tested on recent data).

[51] It removes transients.

[52] Yes, regular normal model with same variance.

[53] Not directly, but the complete system is validated experimentally. In particular, the regular normal model assumption is indirectly checked by comparing the impact of the two model selection methods.

[54] One third of the time horizon over which a prediction is required. It is of the order of one day.

QUESTION 9.8.17. *What is the goal of host load prediction ?* [55]

QUESTION 9.8.18. *What is the general method ?* [56]

QUESTION 9.8.19. *How is host load measured ?* [57]

QUESTION 9.8.20. *How does this load measure relate to execution time.* [58]

QUESTION 9.8.21. *What is the time horizon of the prediction ?* [59]

QUESTION 9.8.22. *What are the prediction models used ?* [60]

QUESTION 9.8.23. *Is the BM model a regular AR(p) model ?* [61]

QUESTION 9.8.24. *What is the criterion of fit ?* [62]

QUESTION 9.8.25. *What is the criterion of evaluation ?* [63]

QUESTION 9.8.26. *What is the fit interval ? The test interval ?* [64]

QUESTION 9.8.27. *What does stepping the model mean ?* [65]

QUESTION 9.8.28. *How are the various models evaluated ?* [66]

QUESTION 9.8.29. *What are the best models for prediction ?* [67]

QUESTION 9.8.30. *Why is AR(p) preferred by the authors ?* [68]

QUESTION 9.8.31. *What is a fractional ARIMA model ?* [69]

QUESTION 9.8.32. *What are the tools used in the factorial analysis ?* [70]

---

[55] In a distributed system, schedule a task on a processor that is less loaded, in order to improve response time.

[56] Host load is monitored and predicted. For a given task, the host with a predicted load compatible with the delay requirement is selected.

[57] The load figure is the number of UNIX processes ready to run. It is smoothed by the UNIX OS. It is polled by the prediction application every second.

[58] Almost linearly (from empirical measurements), which is interpreted as an indication that the system behaves roughly in processor sharing mode.

[59] One step is one second. The prediction horizon h is 1 to 30 seconds.

[60] Simple models: MEAN is the sample mean, used as predictor. BM is the predictor of an AR(p) model with fixed coefficients $\phi_i = 1/p$, namely $\hat{X}_t(1) = 1/p \sum_{s=t-p+1}^{t} X_s$. It is a moving average of the data, causal, with window size p and equal coefficients. Other models are AR(p) models with $p = 1...32$, MA(q) with $q = 1...8$, ARMA(p,q) with $p = 1...4, q = 1...4$, ARIMA with same p, q and $d = 1, 2$ and fractional ARIMA with same p, q and d in the interval $(0, 0.5)$.

[61] No, see Question 9.7.22 on page 236.

[62] Models are fitted using the standard method in this lecture.

[63] The 1-, 15- and 30-step-ahead prediction errors. They are analyzed visually with box-plots.

[64] The fit interval is a subset of the data used for fitting the model. We could call it a training sequence. The test interval is the subset of data, following the fit interval, that is used for doing predictions and comparing with the real value.

[65] Once a model is fitted, keep the model constant but apply the forecasting formulae such as Equation (9.5) to new data.

[66] A randomized set of experiments is done, with the following factors. The fit and test intervals are between 5mn and 3 hours. The model is as described before.

[67] All models give equivalent results for one-step ahead prediction. For larger prediction horizon, ARMA, AR, ARIMA and BM are doing well. MA and MEAN are doing poorly. BM is slightly less good in some cases.

[68] Because model identification is simpler, due to the Levinson-Durbin algorithm

[69] A long range dependent linear model, which is not a second order process in the sense of the Wold decomposition. See Chapter 10.

[70] Box-plots of the results for every model and every time horizon.

9.9 AN OUTLOOK ON FORECASTING METHODS

In this chapter, we have seen both simple heuristic methods and the classical methods based on fitting a linear time series model. Again, it is important to notice the difference between fitting a model for explanatory purposes, or for prediction. In the former case, the fit has to be correct for the interpretation to be valid. In the latter case, the fit has to be best in the sense of prediction – an impossible challenge in all rigor.

Cross-correlation between time series can often be used for prediction, if we believe that one time series anticipates the others. Joint, multi-dimensional ARIMA models are used, as an extension of teh one-dimension ARIMA models seen in this chapter, see [BrockwellDavis02-book] for some examples. A simpler alternative is *regressing on a lagged time series*. For example, if Y_t is the mortgage rate of your bank on day t, take x_t to be the stock market index at times $(t-2, t-1)$ and assume the model $Y_T = x_{t-1}\beta_1 + x_{t-2}\beta_2 + \epsilon_t$.

The GARCH family of models aims at capturing the fact that some time series have a very large volatility, which is expressed by the fact that the variability is much larger than for ARIMA models. A GARCH(p, q) model has the form $Z_t = \sqrt{H_t}\epsilon_t$ where ϵ_t is iid, white noise with a specified distribution (normal or other), and $H_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i Z_{t-i}^2 + \sum_{j=1}^{q} \beta_j H_{t-j}^2$. GARCH models can be fitted numerically using MLE. See [Davison02-book] for some examples. GARCH models were applied primarily to financial data; such models are not able to forecast sudden changes, but they do account for the extreme volatility that follows such events.

An alternative set of methods consists in keeping some of the data as training data, and fit the model that gives the least prediction error on that data (in contrast to the method seen in this chapter which uses the complete set of data for fitting the model). This poses many problems on how to choose the training data, but simplifies the fitting problem, and opens up new heuristic methods, like artificial neural networks or genetic algorithms.

Last but not least, we considered only "forecasting what can be forecast". A complete forecasting method requires the qualitative analysis of external factors.

9.10 EXERCICES

USEFUL MATLAB COMMANDS :

- `Y = filter(P,Q,X)` computes the output $Y = [y_1\ y_2\ y_3 ... y_n]$ of the filter, where $P = [P_0\ P_1\ P_2 ... P_p]$, $Q = [1\ Q_1\ Q_2 ... Q_q]$ are the filter coefficients and $X = [x_1\ x_2\ x_3 ...]$ is the input. The filter is defined by the relation

$$y_k + Q_1 y_{k-1} + ... + Q_q y_{k-p} = P_0 x_k + P_1 x_{k-1} + ... + P_q x_{k-q}$$

  where we set $x_i = 0$ and $y_i = 0$ when $i < 0$ or $i > n$.
  The polynomial $P(\xi) = P_0\xi^p + P_1\xi^{q-1} + ... + P_q$ is called the *numerator polynomial* and $Q(\xi) = \xi^q + Q_1\xi^{q-1} + ... + Q_q$ the *denominator polynomial*.
  In our terminology, this filter is the mapping

$$\begin{array}{rcl} \mathbb{R}^n & \rightarrow & \mathbb{R}^n \\ X & \rightarrow & Y = \dfrac{\sum_{i=0}^{p} P_p B^p}{Id + \sum_{j=1}^{q} Q_j B}(X) = \dfrac{P(B)}{Q(B)} \cdot X \end{array}$$

i.e. the filter is the mapping $\frac{P(B)}{Q(B)}$ (where B is the back shift operator).

- The *reverse filter* is always defined (because we have a finite n and we impose the first coefficient of Q to be non zero). The reverse filter is obtained by transposing P and Q:

```
X = filter(Q,P,X)
```

  Although always defined, the reverse filter might be numerically unstable; this happens when the corresponding infinite discrete filter (i.e. defined for infinite input sequences) is unstable.
- A filter of this form is stable if and only if all its *poles* are inside the unit disk (their modulus is less than 1). The poles are the (usually complex) roots of the denominator polynomial $Q(\xi) = \xi^q + Q_1\xi^{q-1} + ... + Q_q$. The *zeroes* of the filter are the roots of the numerator polynomial $P(\xi) = P_0\xi^p + P_1\xi^{q-1} + ... + P_q$. They are the poles of the reverse filter. Thus, the reverse filter is stable if the zeroes of the original filter are all inside the unit disk.
  `zplane(P,Q)` plots the zeroes and the poles of the filter, together with the unit circle.
- The *impulse response* of the filter is $h = [h_0\ h_1\ ...\ h_n]$ such that

$$y_k = \sum_{i=0}^{k-1} x_{k-i}h_i$$

  It is obtained by applying the filter to the impulse sequence `imp =[1 0 0 ...]`:

$$h = \texttt{filter(P,Q,imp)}$$

- `filter`: `Y = filter(P,Q,eps)` simulates an ARMA process when `eps` is iid white noise; `c = filter(P, Q, imp)` with `imp=[1 0 0 0 0 ...]` computes the coefficients c_k of the MA(∞) representation of the ARMA process
- The convention in Matlab is different from others. Matlab uses: $\Phi(\xi) = 1 + \phi_1\xi + ... + \phi_p\xi^p$.
- `predict` (system identification toolbox): gives predictors for ARMA models
- `armax` (system identification toolbox): fits an ARMA model

USEFUL S-PLUS COMMANDS :

- `x <- arima.mle()` MLE fit of a seasonal ARIMA model, the resulting object `x` contains all information. `x$loglik` is $-2\times$ the log-likelihood.
- `arima.diag()`: plots diagnostics
- `arima.sim` simulate an ARIMA process
- `acf()` computes covariance at all lags
- The convention in S-PLUS is different from ours and from Matlab's. S-PLUS uses: $\Theta(\xi) = 1 - \theta_1\xi - ... - \theta_q\xi^q$.

EXERCISE 9.1. *Assume $X_t = at + Z_t$ where Z_t is stationary. What is the asymptotic behaviour of the sample ACF of X_t ?*

EXERCISE 9.2. *Homework to be designed in detail*

1. *propose a forecasting method for one month ahead that works on the trace `epfl`.*

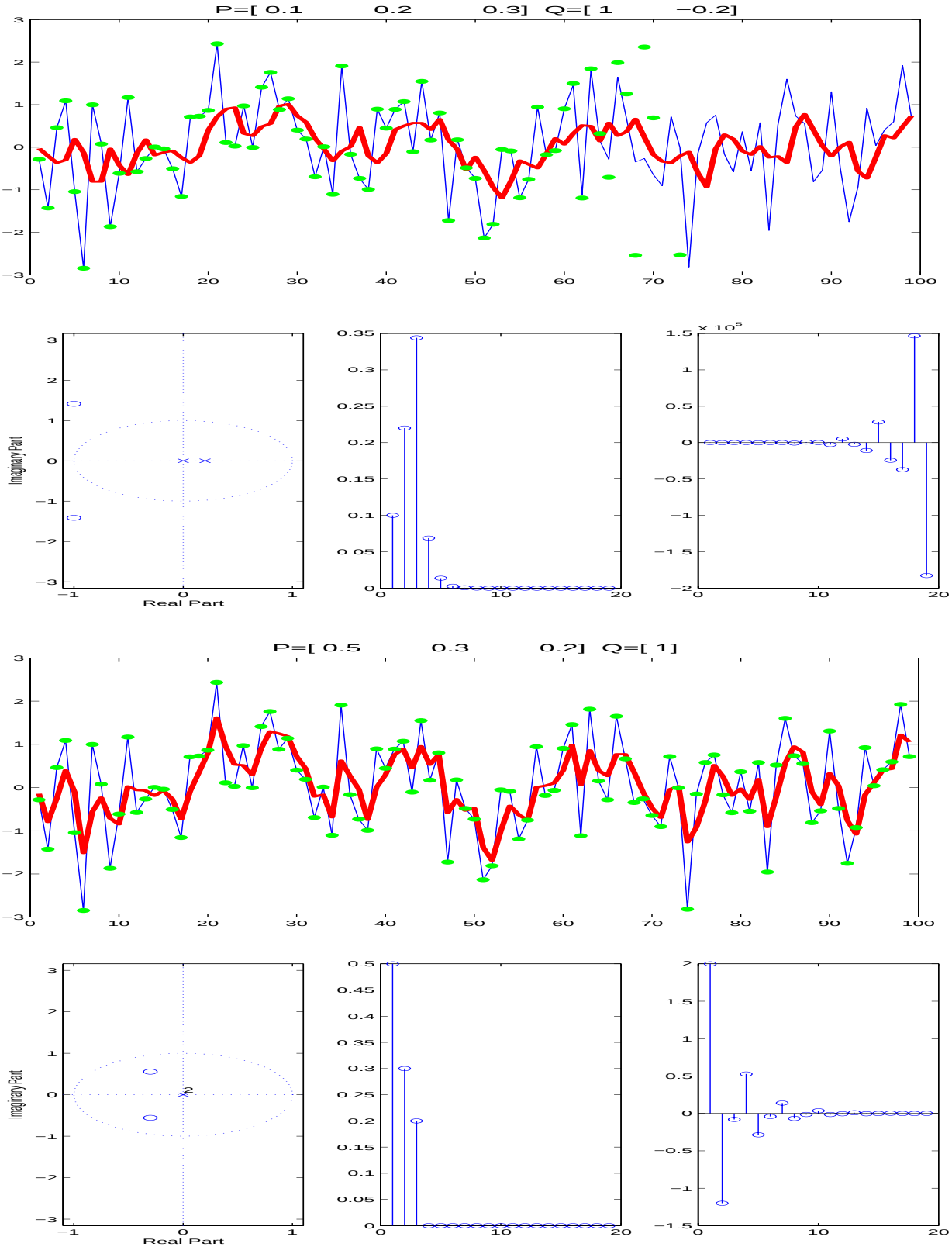


Figure 9.23: Top : time plot of X (thin line) and Y (thick line) for P and Q as indicated, i.e. for $Y_k = 0.1X_k + 0.2X_{k-1} + 0.3X_{k-2} + 0.2Y_{k-1}$. $Z = \texttt{filter}(Q, P, Y)$ (dots) is equal to X in theory, but for large values of the time, the accumulated numerical errors make a difference. Next panels: the zeroes (o) are outside the unit disk, so the inverse filter is unstable; impulse response of filter and of inverse. Bottom: same with the filter $Y_k = 0.5X_k + 0.3X_{k-1} + 0.2X_{k-2}$. The inverse is stable.

2. *apply your prediction algorithm to predict the traffic in the month of the exam. Give a confidence interval. The best prediction is the one which is closest to the true value, while having the smallest confidence interval (details to be finalized).*

EXERCISE 9.3. *Homework to be designed in detail*

1. *propose a forecasting method for one step ahead that works on the traces* `dinda1a`, `dinda2a`, `dinda3a` *and* `dinda4a`. *Try: AR, MA, ARIMA, and Holt-Winters.*
2. *Select the best prediction algorithm by testing your model on* `dinda1b`, `dinda2b`, `dinda3b` *and* `dinda4b` *What is the best model in each case ?*
3. *Apply your best algorithm to predict the load 30 steps after the end of each test sequence. Compare your result to the real value. (we need to find a way such that you can do it only once). The best prediction (maximum error, mean square error) wins.*

EXERCISE 9.4.
- *Show that simple EWMA is equivalent to long-range forecasting with an ARIMA(0,1,1) model.What is the correspondence between α and the ARIMA parameters ?*
- *Show that double EWMA is equivalent to long-range forecasting with an ARIMA(0,2,2) model. What is the correspondence between α, β and the ARIMA parameters ?*

EXERCISE 9.5. *(Homework) Simulation study of the power required for a short file transfer. Do a long simulation. Remove transients. Compute confidence interval using sub-sampling.*

EXERCISE 9.6. *Complete Exercises 9.9 and 9.10 before this one.*

1. *Read [Pappagiannaki03-Infocom]. What are the ARIMA models used for forecasting $l(t)$ and $dt_3(t)$?*
2. *Fit the best ARIMA models to traces* `sprint1a`, `sprint5a` *and* `sprint6a`. *Do you confirm the conclusions of the paper ?*

EXERCISE 9.7. *(Theory)*

1. *What is the orthogonal projection on the set of periodical sequences, with period s ? What is the corresponding de-seasonalizing filter ?*

EXERCISE 9.8. *In TCP, the round trip time is estimated by the following code.*

```
sampleRTT = last measured round trip time
estimatedRTT = last estimated average round trip time
deviation = last estimated round trip deviation

initialization (first sample):
estimatedRTT = sampleRTT + 0.5s; deviation = estimatedRTT/2

new value of sampleRTT available ->
    Err = sampleRTT - estimatedRTT
    estimatedRTT = estimatedRTT + 0.125 * Err
    deviation = deviation + 0.250 * (|Err|- deviation)
    RTO = estimatedRTT + 4*deviation
```

What kind of filter is used for `estimatedRTT` *? for* `deviation` *?*

EXERCISE 9.9. *Reading Assignment. Read [Pappagiannaki03-Infocom] and answer the following questions.*

1. *What is the goal of the forecasting study ?*
2. *Are seasonal variations modeled ?.*
3. *What are the long and short term effects that affect capacity ?*
4. *How is data collected ?*
5. *How many traces are analyzed ?*
6. *Are there seasonal components ?*
7. *What does the 1st, 3rd and 4th time scales correspond to ?*
8. *How are outliers excluded ?*
9. *What is the model resulting from the wavelet analysis ? An empirical statement that an upper bound on used capacity is* $l(t) + 3dt_3(t)$ *where* $l(t)$ *is the weekly average of* $c_6(t)$ *and* $dt_3(t)$ *is the weekly standard deviation of* $d_3(t)$*.*
10. *What is the forecasting method ?*
11. *Do the forecasting models depend on the traces ?*
12. *How are the models validated ?*
13. *What is the filter that maps* c_{j-1} *to* c_j *?*

EXERCISE 9.10. *Homework: traces to be taken from Dina's email (part a is all but last 6 months. Wavelet analysis to be checked with WaveThresh.*

1. *Implement the a-trou wavelets on the trace* `sprint1a`*,* `sprint5a` *and* `sprint6a`*. Apply Holt-Winters to the two series* c_6 *and* dt_3*.*
2. *apply your prediction algorithm to* `sprint1b`*,* `sprint5b` *and* `sprint6b`*. How does the forecast compare to the one in [Pappagiannaki03-Infocom] for the first trace ?*

CHAPTER 10

LONG RANGE DEPENDENCE

10.1 INTRODUCTION

Since [Leland94-ToN], models for data traffic have to incorporate an important feature called long range dependence, which we introduce now. Consider a stationary second-order process X_t with auto-covariance function γ_k. In Chapter 9 we saw that, if the series γ_k is absolutely summable, then Wold's decomposition applies and we can reasonably hope to fit an ARMA model. We call such processes *short range dependent*. In this chapter, we examine the case where this assumption does not hold, i.e. $\sum_{k\in\mathbb{N}} |\gamma_k| = +\infty$. It turns out that this has many practical implications that hold for traffic data sets that both have a high resolution (of the order of seconds) and a very long time span.

The general theory of processes such that $\sum_{k\in\mathbb{N}} |\gamma_k| = +\infty$ is well beyond the scope of this course. Instead, we consider processes fro which $|\gamma_k|$ decays hyperbolically, i.e., is of the order of $\frac{1}{k^\alpha}$, with $0 < \alpha < 1$.

10.2 LONG RANGE DEPENDENCE

10.2.1 DEFINITION

Consider a stationary second-order process X_t, $t = 1, 2, ...$ We say that X_t is *Long-Range Dependent* or has *Long Memory* with order $0 < \alpha < 1$ iff there exists some constant c_1 such that

$$\gamma_k \sim \frac{c_1}{k^\alpha} \tag{10.1}$$

where the equivalence means that the limit of the ratio is 1 when k grows to $+\infty$.

For reasons that become obvious later, the parameter $H = 1 - \frac{\alpha}{2}$ (the *Hurst parameter*) is used instead of α. We consider only cases with $\frac{1}{2} < H < 1$. The value $\frac{1}{2}$ is the boundary between long and short range dependence. The effect of long range dependence is higher for H close to 1.

10.2.2 EXAMPLES

EXAMPLE 10.1: NILE RIVER MINIMA AND THE JOSEPH EFFECT. There exist statistics for the level of the for the period 622–1284. See Figure 10.1 and Figure 10.2. The time series shows periods of increase followed by periods of increase. The series seems non-stationary. These apparent trends are called the *Joseph Effect*, from [Bible, Genesis 41]:

> Joseph said to Pharaoh [..] "Behold, there come seven years of great plenty throughout all the land of Egypt. There will arise after them seven years of famine, and all the plenty will be forgotten in the land of Egypt."

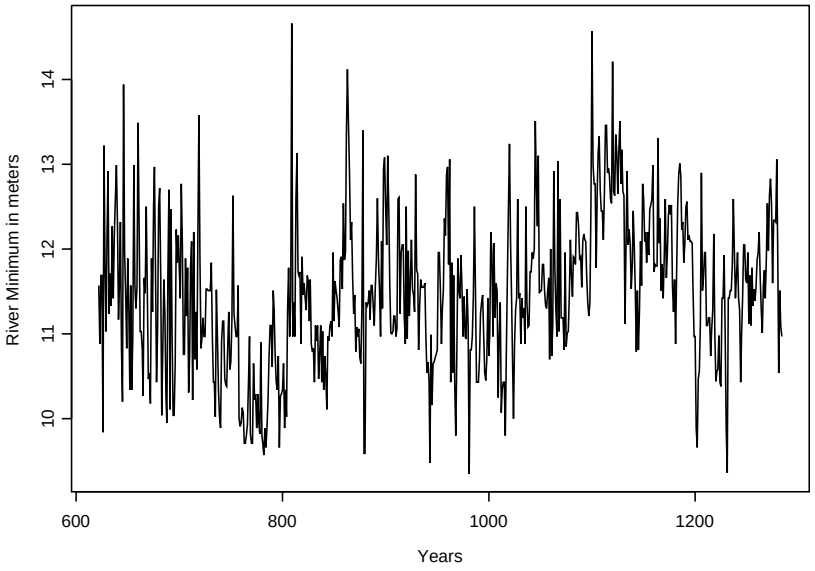


Figure 10.1: Nile River Minima.

EXAMPLE 10.2: ETHERNET DATA. Figure 10.3 and Figure 10.4 . The number of bytes (Figure 10.3) or packets Figure 10.4 also shows a very irregular pattern. The ACF decays slowly. The figures show aggregation at different time scales. The aggregate data does not seem to look more like normal iid noise, as the central limit theorem would say. See also Figure 10.5 and Figure 10.6.

EXAMPLE 10.3: Counter-Example: for an ARMA process, it can be shown that there always exists some $r > 0$ such that $\rho_k = o(r^k)$, thus an ARMA process is always short-range dependent. See Figure 10.7 and Figure 10.8.

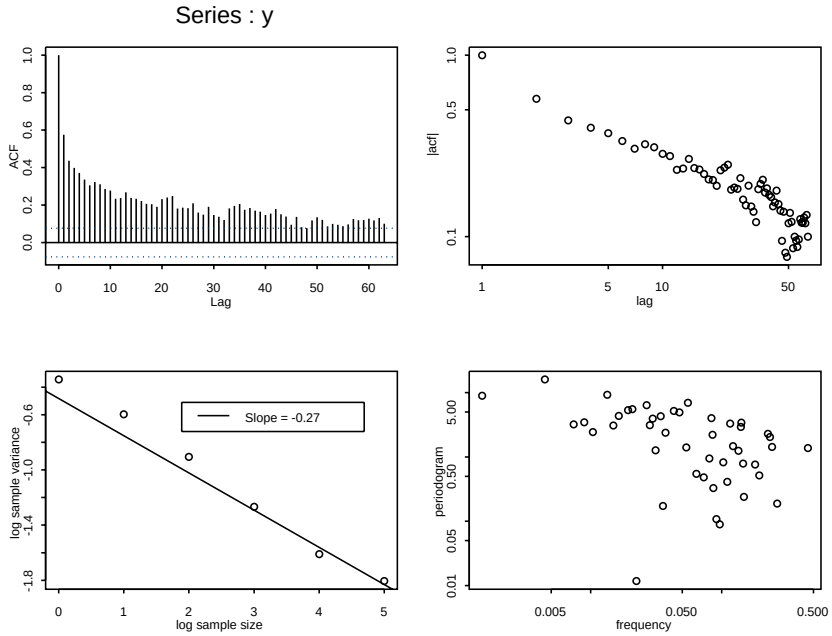


Figure 10.2: Top row: ACF of Nile data in natural scale (with 95% confidence limits about zero) and in log-scale. Bottom row: variance time plot with slope α; estimation of Hurst parameter is $H = 1 - \frac{\alpha}{2}$ ($H = 0.865$); and periodogram.

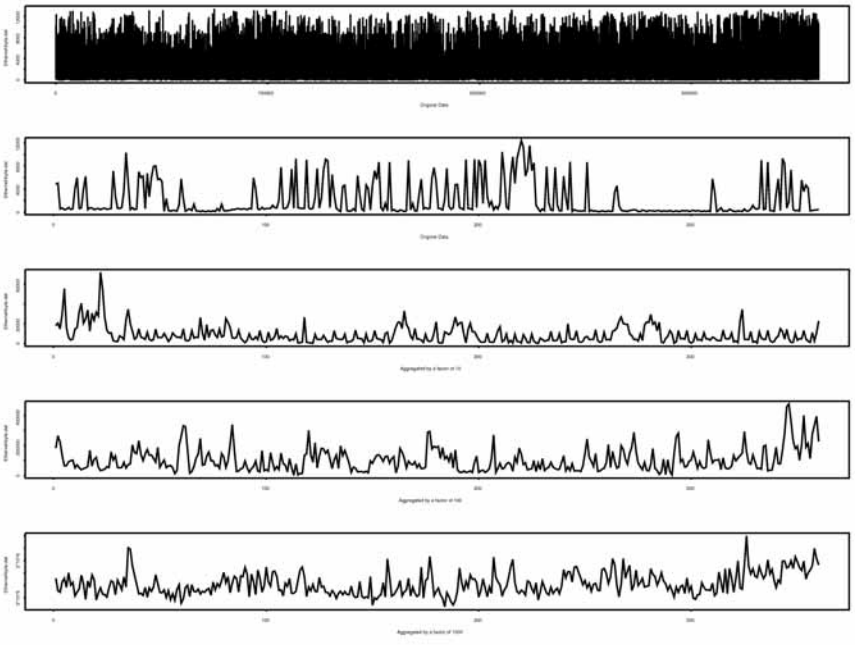


Figure 10.3: Ethernet Data, in Bytes, aggregated at different time scales. All graphs are truncated to have the same number of points except the top one which is the original data. [Leland94-ToN]

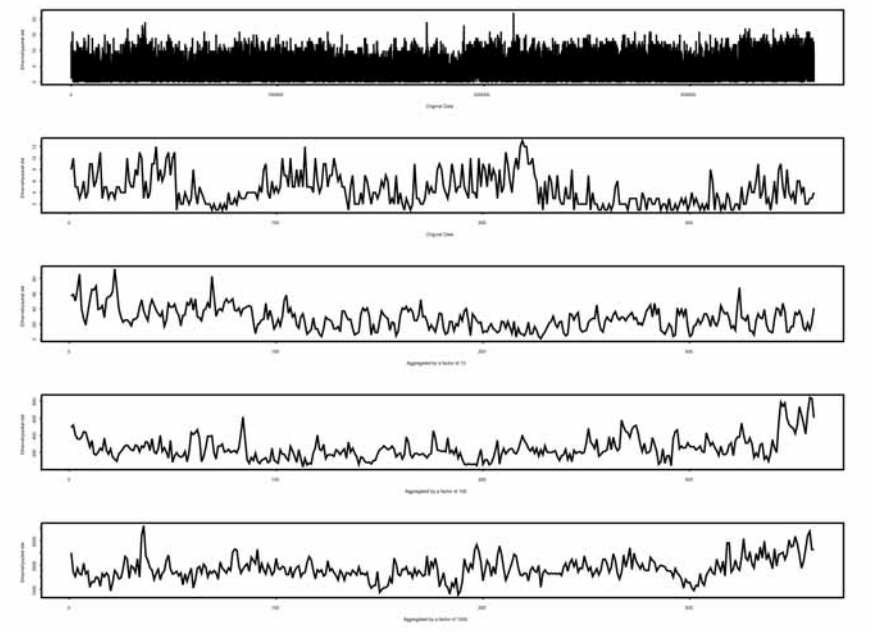


Figure 10.4: Ethernet Data, in Packets.

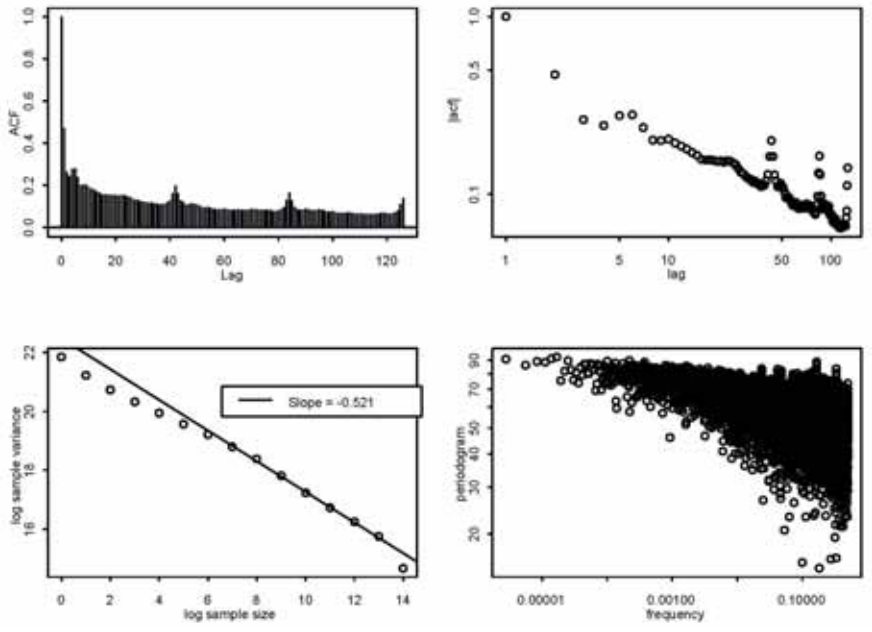


Figure 10.5: ACF, variance time plot, and periodogram of Ethernet byte data (confidence interval about 0 is indistinguishable from 0). Estimated $H = 0.740$.

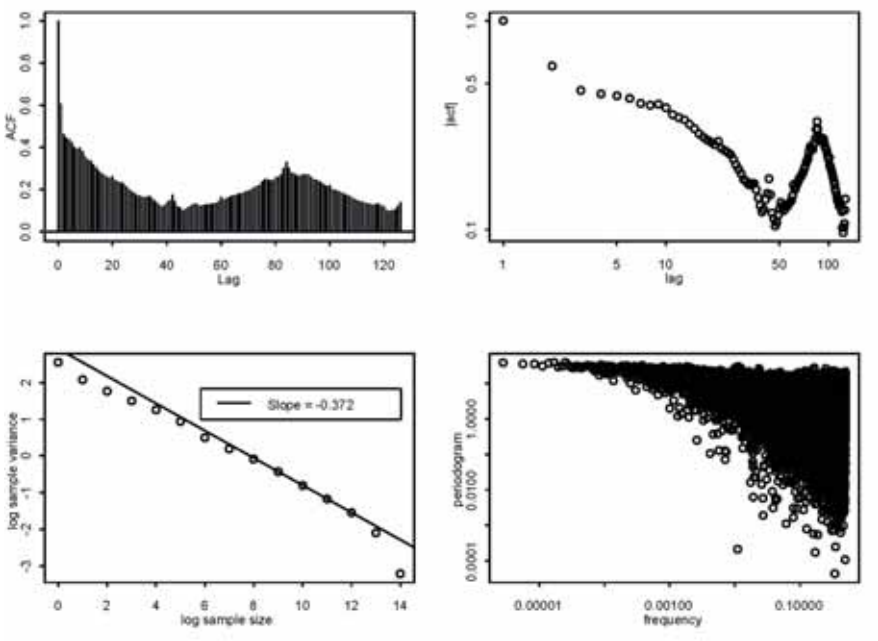


Figure 10.6: ACF, variance time plot, and periodogram of Ethernet packet data. Estimated $H = 0.814$

10.2.3 PROPERTIES

VARIANCE OF SAMPLE MEAN. For a short range dependent process, the variance of the sample mean $\bar{X}_n = 1/n \sum_{t=1}^{n} X_n$ decays as $1/\sqrt{n}$. For a long range dependent process, the decay is slower:

THEOREM 10.2.1 (Beran94-book). *Let X_t be long range dependent. For $n \to +\infty$:*

$$\mathrm{var}\bar{X}_n \sim \frac{c_1}{H(2H-1)} \frac{1}{n^{2(1-H)}}$$

QUESTION 10.2.1. *What is the order of the variance of the partial sum $S_n = \sum_{t=1}^{n} X_t$?* [1]

SPECTRAL DENSITY. The *spectral density* $f()$ of the time series X_t is defined as the Fourier transform of the auto-covariance:

$$f(\omega) = \sum_{k \in \mathbb{Z}} \gamma_k e^{-ik\omega}$$

(This is well defined only if we accept f to be a Distribution rather than a standard function). Since $\gamma_{-k} = \gamma k$, the spectral density is even and real: $f(\omega) = f(-\omega) \in \mathbb{R}$. Conversely, the autocovariance is retrieved by the inverse Fourier transform:

$$\gamma_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\omega) e^{i\omega k} d\omega = \frac{1}{\pi} \int_{0}^{\pi} \cos(\omega k) f(\omega) d\omega$$

[1] $\frac{c_1}{H(2H-1)} n^H$

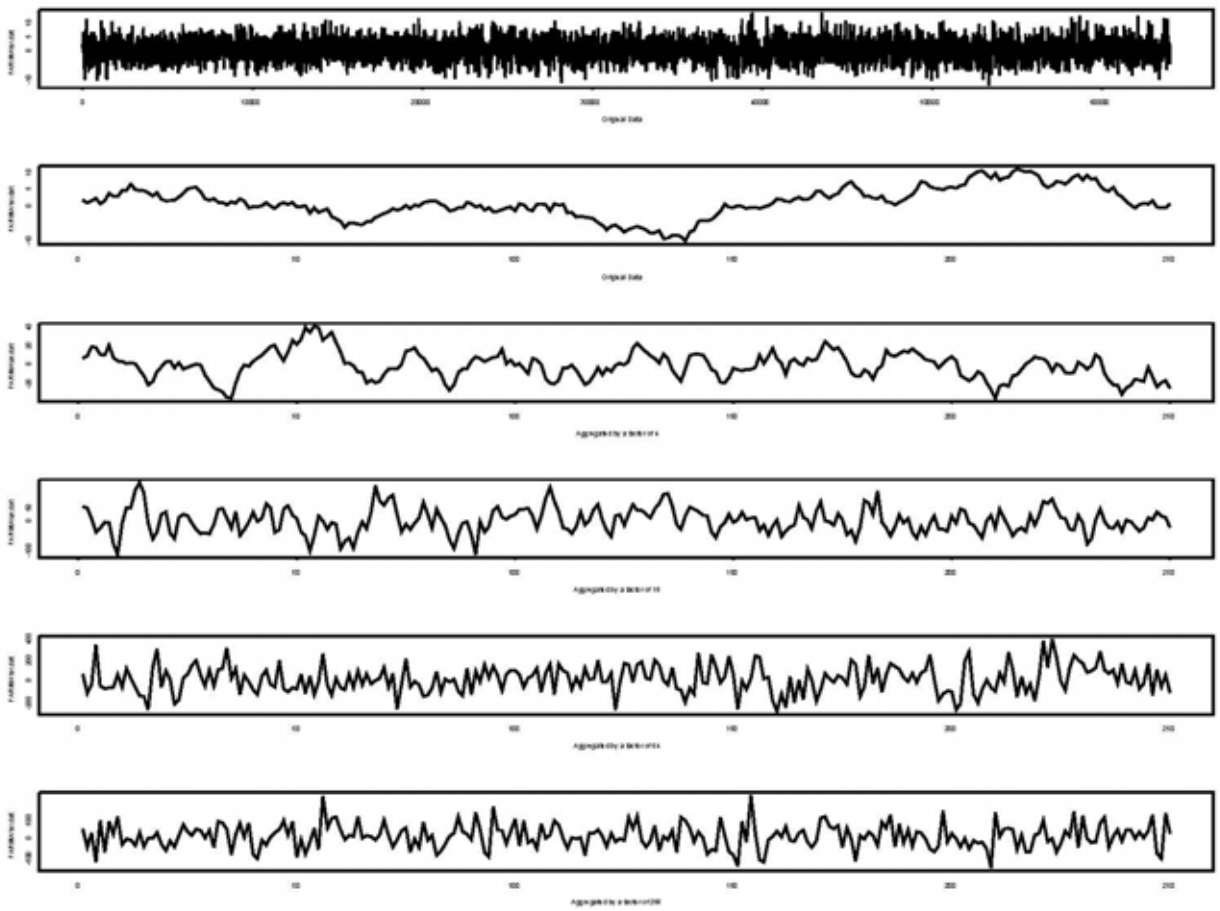


Figure 10.7: An ARMA process with $p = 1$, $q = 0$, $\phi_1 = 0.95$. The top two graphs are the original time series. Other graphs are aggregated and re-scaled. The time series has some local trends at the original time scale, due to the auto-regressive component, but they disappear by aggregation. In the aggregation limit, we have white noise.

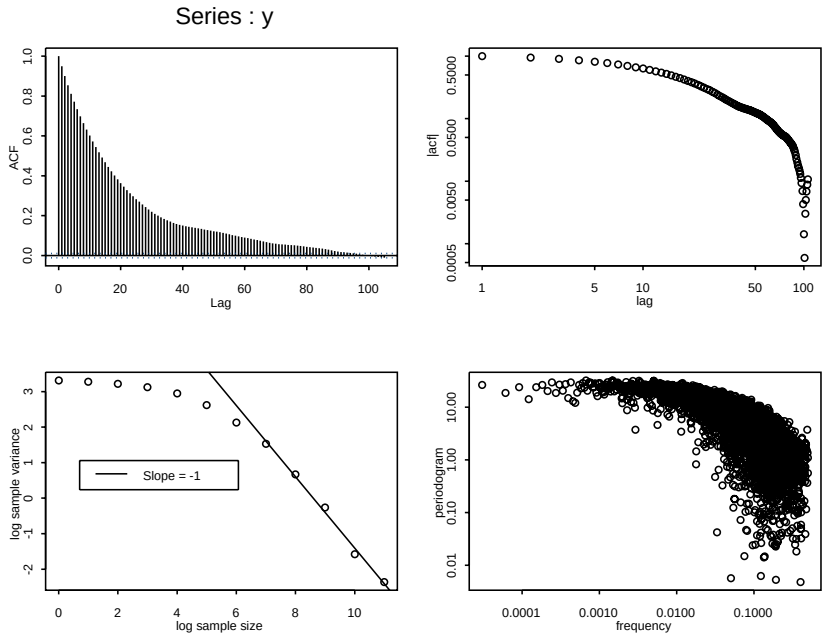


Figure 10.8: ACF, variance time plot, and periodogram of simulated ARMA data. Estimated $H = 0.5$

Long range dependence is visible in the frequency domain by a pole at the origin ("$1/f$-noise").

THEOREM 10.2.2 (Spectral Density of LRD). *If X_t is long range dependent, the spectral density $f(\omega)$ is defined for $\omega \neq 0$; for ω in the neighbourhood of 0:*

$$f(\omega) \sim c_2 |\omega|^{1-2H}$$

Conversely, this property implies long range dependence. The constants c_1 and c_2 are related by

$$c_2 = 2c_1 \Gamma(2H-1)\sin((1-H)\pi)$$

In contrast, with short range dependence, the spectral density is defined and continuous for $\omega = 0$. In the theorem, $\Gamma()$ is *Euler's integral*, defined by $\Gamma(x) = \int_0^{+\infty} t^{x-1}e^{-t}$. If x is a positive integer, then $\Gamma(x) = (x-1)!$. Γ is defined everywhere except at negative integers and 0.

10.2.4 HURST PARAMETER

Hurst is a famous hydrologist who, like many, was interested in Egypt and The Nile. Hurst found in 1951 that the level of the Nile was a long range dependent sequence. He formulated it as follows. Assume you build a reservoir of capacity B_1. At time t_0, the reservoir has initial content B_0. Hurst was interested in the value of B_1 that, over some time interval $[t_0, t_0+h]$ would guarantee a constant output rate and no overflow. Call Y_s the cumulative input of water into the reservoir, minus evaporation and leaks. The conditions are

$$\begin{cases} c = \frac{Y_{t_0+h} - Y_{t_0}}{h} \\ B_0 + Y_s - Y_{t_0} - cs \geq 0 \text{ for all } s \in [t_0, t_0+h] \\ B_0 + Y_s - Y_{t_0} - cs \leq B_1 \text{ for all } s \in [t_0, t_0+h] \end{cases}$$

It can easily be shown that it is necessary and sufficient that

$$B_1 \geq R(t_0, h) = \max_{s \in [t_0, t_0+h]} (Y_s - Y_{t_0} - cs) - \min_{s \in [t_0, t_0+h]} (Y_s - Y_{t_0} - cs)$$

$R(t_0, h)$ is called the "range statistic". It is the required capacity of the reservoir. A scale-free version of it, called *rescaled range statistic* is $R(t_0, h)/S(t_0, h)$, with

$$S(t_0, h)^2 = \frac{1}{k} \sum_{s \in [t_0, t_0+h]} (X_s - \bar{X}(t_0, h))^2$$

with $X_s = Y_s - Y_{s-1}$ and $\bar{X}(t_0, h)$ is the sample average of X_s over $[t_0, t_0 + h]$.

It turns out that for short range dependent processes, for large h, $R(t_0, h)/S(t_0, h)$ should be of the order of $\sqrt{h}$. Hurst plotted $R(t_0, h)/S(t_0, h)$ in log-log scale for various values of t_0 and h and, in contrast, found that the regression line always tended to have a slope greater than $1/2$. For a long range dependent process with order α, the slope of this line is precisely $H = 1 - \frac{\alpha}{2}$, hence the name.

10.2.5 REMARKS ON TERMINOLOGY.

A slightly more general definition is that $\gamma_k = \frac{L(k)}{k^\alpha}$, where $L()$ is a slow varying function at infinity, which means that $\lim_{k \to +\infty} L(kx)/L(k) = 1$ for any fixed $x > 0$. This allows more general decays than exponential, for example $\gamma_k = \frac{\log k}{k^\alpha}$. We do not use this slightly more general definition for simplicity, as this does not impact the results in this chapter.

Some authors call *fractional process* a stationary process that satisfies our definition or its variant with slowly varying functions, leaving the concept of long range dependence for the general case $\sum_{k \in \mathbb{N}} |\gamma_k| = +\infty$.

Finally, note that we focus on processes with finite variance γ_0.

10.3 FRACTIONAL ARIMA PROCESSES

Fractional ARIMA processes (FARIMA) are generalizations of ARIMA processes that have long range dependence.

FRACTIONAL DIFFERENCE OPERATOR. An ARIMA process is defined by (Section 9.7.2)

$$\Phi(B)(Id - B)^d Y_t = \Theta(B)\epsilon_t$$

First note that

$$(Id - B)^d = \sum_{k=0}^{d} \binom{d}{k} (-1)^k B^k$$

with

$$\binom{d}{k} = \frac{d!}{k!(d-k)!} = \frac{\Gamma(d+1)}{\Gamma(k+1)\Gamma(d-k+1)}$$

The function Γ is defined in Section 10.2.3. Γ is defined and $\neq 0$ for all real numbers except for integers that are ≤ 0. The binomial coefficient $\begin{pmatrix} d \\ k \end{pmatrix}$ can thus be extended by the above formula to all positive, real values of d and all integer values of k. If d is integer and $k \geq d+1$ then $\Gamma(d-k+1) = \infty$ and $\begin{pmatrix} d \\ k \end{pmatrix} = 0$. Thus, we can define, at least formally,

$$(1-\xi)^d = \sum_{k=0}^{+\infty} \begin{pmatrix} d \\ k \end{pmatrix} (-1)^k \xi^k$$

and the definition coincides with the usual one if $d \in \mathbb{N}$. We call it the *fractional difference operator*. It is well defined for finite time series, and it can be shown that the convergence occurs in l^2 sense for infinite time series.

$$(Id-B)^d = \sum_{k=0}^{+\infty} \begin{pmatrix} d \\ k \end{pmatrix} (-1)^k B^k \tag{10.2}$$

Note that $\begin{pmatrix} d \\ k \end{pmatrix} (-1)^k$ is also simply equal to $\prod_{j=1}^{k} \frac{j-1-d}{j}$. The z-transform of the fractional difference operator is $H(z) = \sum_{k=0}^{+\infty} \begin{pmatrix} d \\ k \end{pmatrix} (-1)^k B^k = (1-z)^d$. The linear filter theorem continues to apply: if $Y_t = (Id-B)^d X_t$ with $-1/2 < d < 1/2$ then the spectral density relation holds

$$f_Y(\omega) = \left|1 - e^{i\omega}\right|^{2d} f_X(\omega)$$

It follows that, if a process X_t is LRD with Hurst parameter H, then $Y = (1-B)^d X$ is SRD (with $d = H - \frac{1}{2}$). See Figure 10.9. This suggests the following family of models.

FRACTIONAL ARIMA. A fractional ARIMA process Y_t has parameters p, d, q, Φ, Θ, F, where p, q are integers, $-1/2 < d < 1/2$, Φ, Θ are polynomials with the same restrictions as for ARMA processes, and F is a probability distribution with 0 mean. It is defined as the stationary solution to

$$\Phi(B)(Id-B)^d Y_t = \Theta(B)\epsilon_t$$

where $\epsilon_t \sim$ iid F. Unless otherwise specified, we take $F = N(0, \sigma^2)$.

For $d = 0$, the process is simply ARMA.

The definition is equivalent to

$$\Phi(B)Y_t = (Id-B)^{-d}\Theta(B)\epsilon_t$$

which shows that the process is well defined. The commutativity of power series in B also implies that

$$\Phi(B)Y_t = \Theta_t W_t$$

where

$$(Id-B)^d W_t = \epsilon_t \tag{10.3}$$

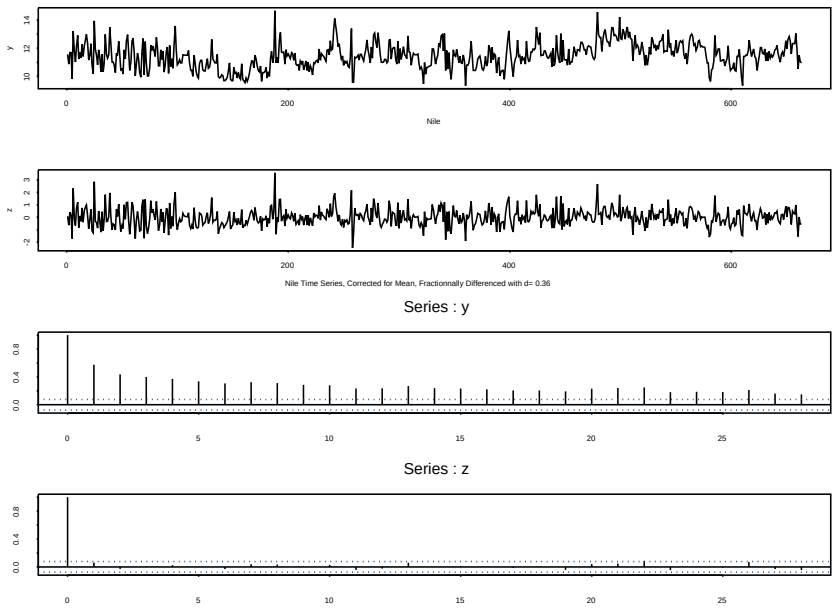


Figure 10.9: Fractional Difference Operator $(Id - B)^d$ applied to the mean corrected time series "Nile" Y_t ($d = 0.36$) (bottom), compared to original time series (top) The transformed time series is SRD.

We can interpret Y_t as an ARMA process where the noise is W_t. Such a noise is called *fractionally integrated white noise*. Its variance is

$$\gamma_W(0) = \sigma^2 \frac{\Gamma(1-2d)}{\Gamma^2(1-d)}$$

Its spectral density is

$$f_W(\omega) = \left|1 - e^{i\omega}\right|^{-2d} \frac{\sigma^2}{2\pi}$$

Thus, by Theorem 10.2.2, it is long range dependent for $0 < d < 1/2$, with

$$H = \frac{1}{2} + d$$

The same holds for FARIMA processes in general.

For $-1/2 < d < 1/2$ the operator $(Id - B)^d$ can be inverted and its inverse is $(Id - B)^{-d}$. Thus, following Equation (10.3):

$$W_t = (Id - B)^{-d} \epsilon_t$$

It can be shown that for $0 \leq d < \frac{1}{2}$, the FARIMA model can be assumed stationary, just like an ARMA process (i.e. if the auto-regressive polynomial has all roots outside the unit disk, and we pick as initial condition the stationary distribution).

The ACF of fractionally integrated white noise is

$$\rho_W(k) = \frac{\Gamma(k+d)\Gamma(1-d)}{\Gamma(k-d+1)\Gamma(d)} = \prod_{j=1}^{k} \frac{j-1+d}{j-d}$$

The autocovariance of a FARIMA process is

$$\gamma_h = \sum_{j,k\in\mathbb{N}} \psi_j \psi_k \gamma_W(h+j-k)$$

where $\sum_{j\in\mathbb{N}} \psi_j \xi^j = \Theta(\xi)/\Phi(\xi)$. It can also be computed from the spectral density relation

$$f_Y(\omega) = \left|\frac{\Theta(e^{i\omega})}{\Phi(e^{i\omega})}\right|^2 f_W(\omega) = \left|\frac{\Theta(e^{i\omega})}{\Phi(e^{i\omega})}\right|^2 \left|1-e^{i\omega}\right|^{-2d} \frac{\sigma^2}{2\pi}$$

See Figure 10.10, Figure 10.14 and Figure 10.7 for simulations of FARIMA processes.

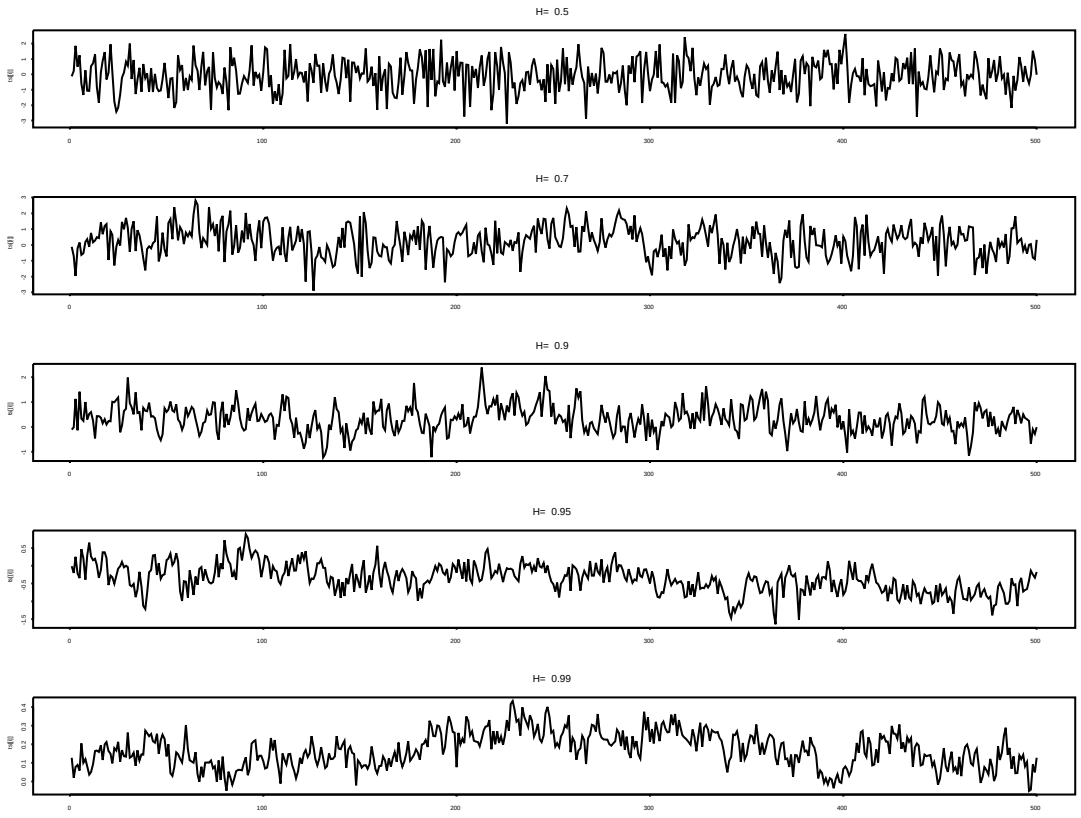


Figure 10.10: Simulations of fractionally integrated white noise, with unit variance. Top: $H = 0.5$ (iid normal noise). Long range dependence increases as H becomes close to 1. We see that for H close to 1 the time series exhibits apparent local trends, typical of long range dependence.

QUESTION 10.3.1. *Is a FARIMA process stationary for $0 < d < 1/2$? Same question for an ARIMA process with $d \in \mathbb{N}$.* [2]

[2]FARIMA is stationary for $0 < d < 1/2$, unlike ARIMA which is not stationary, except for $d = 0$.

10.4 LRD AND SELF-SIMILARITY

10.4.1 SELF-SIMILAR TIME SERIES

Self-similarity was introduced by Kolmogorov and Mandelbrot. A deterministic geometrical object is self-similar if it repeats the same pattern, independent at the distance from which we look at it. A stochastic process is self-similar if the sample paths "look the same", independent of the distance, but are not a repetition of a pattern.

Formally, consider a stationary time series X_t. For all m, define

$$X_t^{(m)} = \frac{1}{m}\left(X_{(t-1)m+1} + ... + X_{tm}\right)$$

$X_t^{(m)}$ is obtained by aggregating the data in X_t by blocks of size m, and averaging.

DEFINITION 10.4.1. *X_t is a self-similar time series iff for all m, X_t and $X_t^{(m)}$ have the same distribution, up to a scaling factor.*

If a time series is the limit of normalized partial sums of a stationary time series, then it is self-similar. Thus the role of self-similar time series among stationary time series is the same as stable distributions among univariate distributions (Section 8.2).

The factor in Definition 10.4.1 necessarily has the form $\frac{1}{m^{1-H}}$ for some H (called the Hurst parameter). If we assume that the time series has second moments and the autocorrelation decays to 0 then the only possible cases are $0 < H < 1$. For $0 < H < \frac{1}{2}$ the process is short-range dependent and has the property that all correlations are negative and $\sum_{k\geq 1}\rho_k = \frac{-1}{2}$ – a case that we will not consider in practice. Thus we will consider only the case $H \in [1/2, 1)$. The only self-similar time series we will encounter is a Gaussian times series called fractional Gaussian noise, defined later.

A second order stationary time series is called a (second order) self-similar time series with Hurst parameter $H \in [1/2, 1)$ if for all m, X_t and $\frac{1}{m^{1-H}}X_t^{(m)}$ have the same second order characteristics (mean and auto-covariance).

A second order stationary time series has the following properties.

- Its ACF is
$$\rho_k = \frac{1}{2}\left((k+1)^{2H} - 2k^{2H} + (k-1)^{2H}\right)$$
- For $H = 1/2$, the time series is non correlated. For $H > 1/2$, a limited development of $(1+x)^{2H}$ shows that, for large k
$$\rho_k \sim H(2H-1)k^{2H-2}$$
thus $\sum_{k>0}\rho_k = +\infty$ and the series is long range dependent.
- the spectral density for $\omega \neq 0$ is
$$f(\omega) = c(1-\cos\omega)\sum_{j\in\mathbb{Z}}|2\pi j + \omega|^{-2H-1}$$
with $c = 2\sigma^2\sin(\pi H)\Gamma(2H+1)$. Further, for $\omega \to 0$:
$$f(\omega) \sim \frac{c}{2|\omega|^{2H-1}}$$

Iid time series are self-similar iff the distribution of X_i is p-stable. Iid normal time series are self-similar with Hurst parameter $H = 1/2$.

QUESTION 10.4.1. *What is the Hurst parameter of a stable iid time series ?* [3]

The only other example we will encounter is Fractional Gaussian noise. Fractional ARIMA processes are long range dependent but not self-similar.

QUESTION 10.4.2. *Show that FARIMA$(0, d, 0)$ is not self-similar.* [4]

10.4.2 FRACTIONAL GAUSSIAN NOISE

DEFINITION 10.4.2. *Fractional Gaussian noise (fGn) is the only self-similar time series X_t, $t \in \mathbb{N}$, that is gaussian and such that*

- $\mathbb{E}(X_t) = 0$
- $\text{var}(X_t) = \sigma^2$ *for some fixed* $\sigma^2 > 0$
- *It is second order stationary and its auto-covariance function is*

$$\gamma_k = \frac{\sigma^2}{2}\left((k+1)^{2H} - 2k^{2H} + (k-1)^{2H}\right)$$

where $H \in [\frac{1}{2}, 1)$ is a fixed parameter (called the Hurst parameter).

For $H = 0.5$, $\gamma_k = 0$ and fractional Gaussian noise is the usual normal white noise. For $H > 0.5$, fGn is *not* white noise.

QUESTION 10.4.3. *Is fGn stationary ?* [5]

.

QUESTION 10.4.4. *How would you simulate fGn ?* [6]

Efficient simulations of fGn are based on the fact that the discrete Fourier transform $\hat{X}(\omega)$ of a stationary time series X_t with auto-covariance γ_k is non-stationary white noise (i.e. $\mathbb{E}(\hat{X}(\omega)) = 0$ and $\hat{X}(\omega), \hat{X}(\omega')$ are independent) with variance $v(\omega) =$ discrete fourier transform of γ_k. See Figure 10.11 for an example.

QUESTION 10.4.5. ***Trouvez l'intrus.*** *Three simulated and aggregated time series are shown on Figures 10.12 to 10.14. The first two graphs are the original time series (all samples, 250 first samples), the following graphs are 250 samples of the aggregated time series, aggregated 4 times each.*
The three time series are one of the following:

[3] $H = 1/p$.

[4] Plot the auto-covariance function and see that it does not have the proper form.

[5] Yes: it is second order stationary and normal

[6] For $t = 0$, draw a normal random variable with variance σ^2; this gives a number x_0. For $t = 1$, compute the conditional distribution of X_1 given $X_0 = x_0$. From Theorem 12.5.4, it is normal with mean $\rho_1 x_0$ and variance $\sigma^2(1 - \rho_1^2)$. Draw a random normal variable with these parameters and obtain x_1. Iterate: x_2 is obtained by sampling the distribution of X_2 conditional to $X_{=}0 = x_0, X_1 = x_1$, and so on.

- *ARIMA process with* $p = 1$, $d = 0$, $q = 0$,$\phi_1 = 0.95$
- *FARIMA process with* $H = 0.9$, $p = 1$, $q = 0$, $\phi_1 = 0.95$,
- *fractional gaussian noise with* $H = 0.9$,
- *fractional gaussian noise with* $H = 0.5$.

Say which is what. [7]

The rest of this section explains how fractional Gaussian noise can be mathematically constructed from a continuous time process called *Fractional Brownian motion* (fBm). You can skip it a first reading.

DEFINITION 10.4.3. *A continuous time process* Y_t *is self-similar with stationary increments iff*

- *For any stretching factor* c*, the process* $Y(ct)$ *has the same distribution as* $Y(t)$*, up to some scaling factor (which depends on* c*).*
- *The distribution of* $Y(t + k) - Y(t)$ *is independent of* t.

Necessarily (apart from pathological processes), the scaling factor must be of the form c^{-H} for some $H > 0$. If the process has finite second moments then $0 < H < 1$. We consider in this lecture only the case $\frac{1}{2} \leq H < 1$.

A zero mean self-similar process with stationary increments and with second moments necessarily has a covariance function given by

$$\Gamma_{s,t} = \frac{\sigma^2}{2}\left(t^{2H} - (t-s)^{2H} + s^{2H}\right) \tag{10.4}$$

where $\sigma^2 = \mathrm{var}(Y(1))$. Thus for $H = \frac{1}{2}$ the process is uncorrelated.

DEFINITION 10.4.4. *Fractional Brownian motion* $B_{H,\sigma^2}(t)$*, with Hurst parameter* $H \in [1/2, 1)$ *and variance parameter* σ^2 *is the only process with the following properties.*

1. $B_H(t)$ *is gaussian with 0 mean.*
2. $B_H(t)$ *is a self-similar process with stationary increments.*
3. $B_H(0) = 0$ *a.s.*
4. $\mathrm{var}(B_H(1)) = \sigma^2$

For $\sigma^2 = 1$ *we call it the standard fractional Brownian motion* $B_H(t) = B_{H,1}(t)$

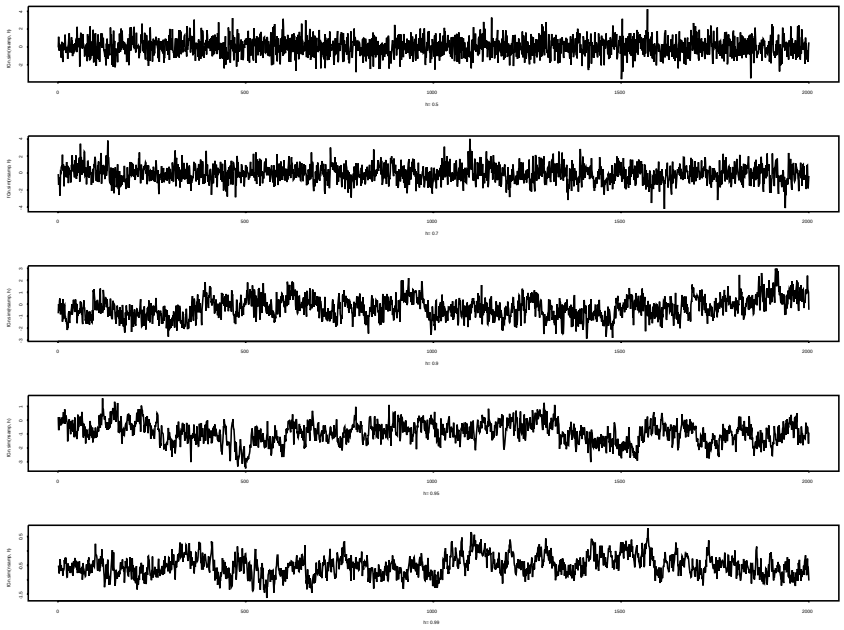
The covariance function of the fractional Brownian motion is given by Equation (10.4). $B_H(t)$ is a convenient mathematical abstraction, but it has non smooth properties: its sample paths are continuous but nowhere differentiable.

Note that $B_H(t)$ is *not* stationary.

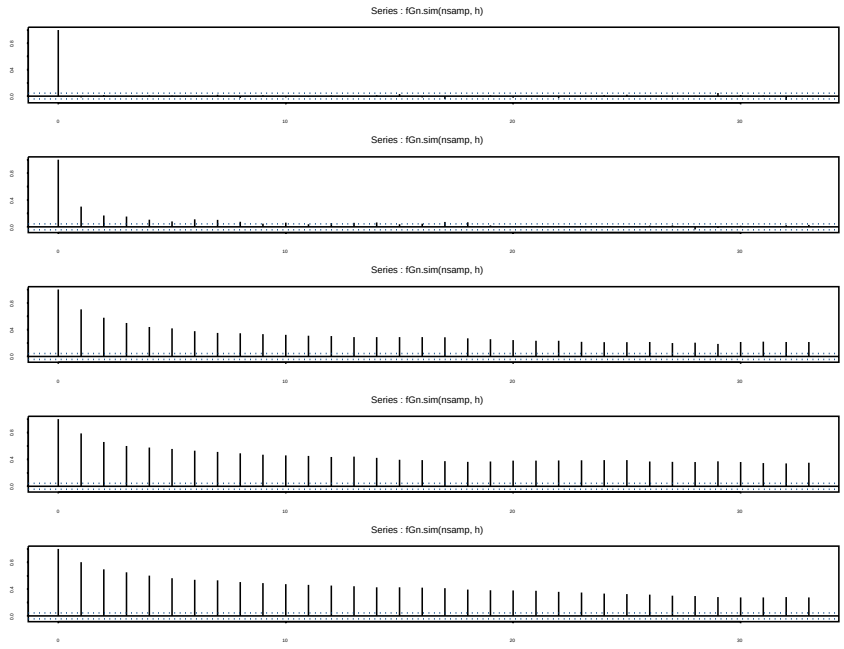
For $H = \frac{1}{2}$, we have the ordinary Brownian motion. It is the only one for which the increments are independent.

DEFINITION 10.4.5. *Fractional Gaussian noise is the time series of increments of a Fractional Brownian motion:* $X_t = B_H(t) - B_H(t-1)$*, for* $t = 1, 2, ...$

[7] (a)=fGn 0.5, (b)=fGn 0.9, (c)=FARIMA.



(a) Simulated Time Series



(b) ACF

Figure 10.11: Simulated fractional Gaussian noise for $H = 0.5$ (top) to $H = 0.99$ (bottom) with sample ACF.

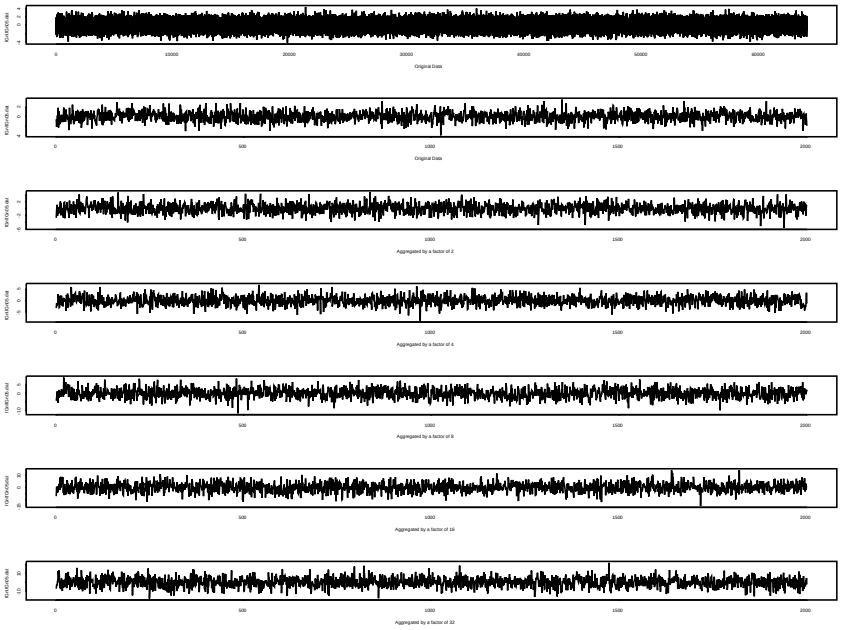


Figure 10.12: Series (a) of the game in Question 10.4.5

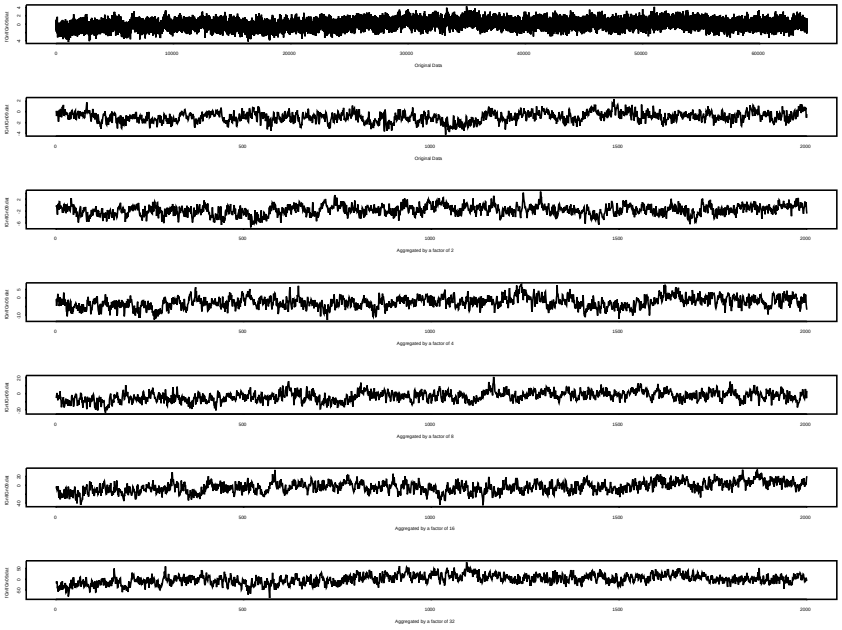


Figure 10.13: Series (b) of the game in Question 10.4.5

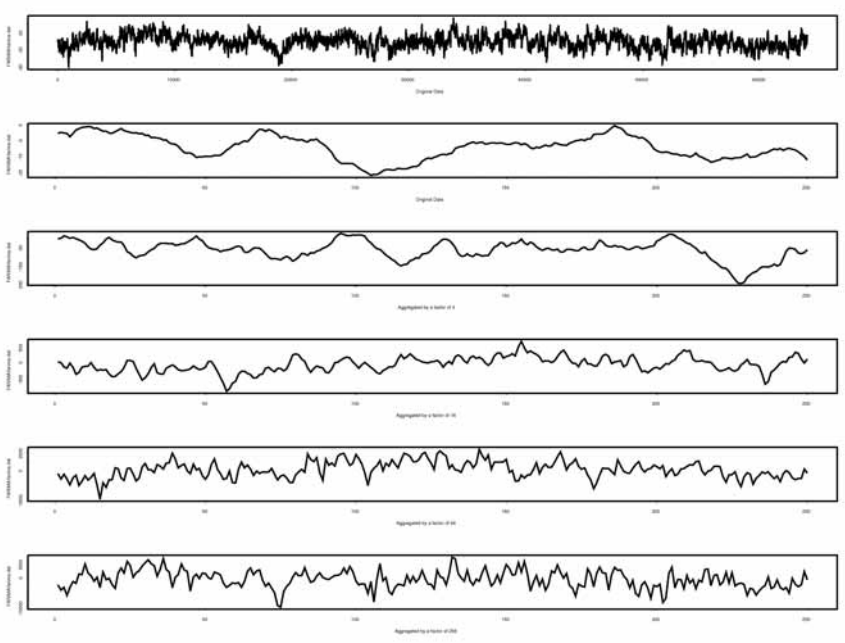


Figure 10.14: Series (c) of the game in Question 10.4.5

Fractional Gaussian noise is a Gaussian time series with the following properties:

- it has zero mean
- it is stationary
- it is self-similar
- For $H = \frac{1}{2}$, fGn is the ordinary sequence of iid noise: $X_t \sim iidN(0, \sigma^2)$. In all other cases, it is not iid and is long range dependent

This gives a useful intuitive representation of fractional Gaussian noise. For $H = 1/2$ we have the usual Brownian motion

The definition of fGn can be extended to other self-similar time series, with stable marginal distribution. Such time series have infinite variance and no auto-covariance function can be defined.

10.4.3 ASYMPTOTIC SELF-SIMILARITY AND LRD

Long range dependence and fractional Gaussian noise are related as follows. A general result is that (under some mild conditions), the partial sums of a long range dependent time series, re-scaled by $1/n^H$, converges in distribution to a fractional Gaussian noise. For a short range dependent time series, this is the usual central limit theorem. See Figure 10.7 and Figure 10.14

This illustrates an important visual aspect of long range dependence. For a short range dependent time series with finite variance, the partial sums converge to a memoriless process. In contrast, with long range dependence, the limits of partial sums are still correlated: short memory disappears with aggregation, while long memory resists.

This also gives a useful intuitive representation of fractional Brownian motion. For $H = 1/2$ we have the usual Brownian motion $B(t)$. Intuitively, think of the Brownian motion as the limit of a

random walk when the time step is very small: $B((t+1)\delta) = B(t\delta) + \epsilon_{t+1}$ where ϵ_t is iid white noise. By the central limit theorem, the increments of $B(t)$ are normal. Thus the random walk ARIMA$(0,1,0)$ gives an approximation of brownian motion.

In contrast, for $H \in (1/2, 1)$, think of $B_H(t)$ as the limit of a random walk where the increments ϵ_t are long range dependent, with Hurst parameter H.

10.5 STRUCTURAL MODELS WITH LRD

Structural models try to reproduce the essential features of a system (as opposed to black-box models such as time series or regression models). Some models are able to explain long range dependence by heavy tails.

FLUID MODEL WITH HEAVY TAILED INTER-ARRIVAL TIMES. The following model is adapted from [Grossglauser96-sigcomm]. It represents traffic intensity on a network link or a web server, as follows. There is a sequence of *rate change epochs* τ_n, such that the sequence $T_n = \tau_{n+1} - \tau_n$ is iid, with complementary distribution function $F(t) = \mathbb{P}(T_n > t)$. At time τ_n, a rate $\lambda(n)$ is picked at random, independently of the past and present of the system, from a finite set of rates $\{\lambda_1, ...\lambda_I\}$. Let $\pi_i = \mathbb{P}(\lambda(n) = \lambda_i)$. Let $X(t)$ be the rate at time t. We assume that the system has been running for a long time and is in stationary regime, which means that X_t is a stationary sequence.

Assume that the distribution of T_n is heavy tailed and has a finite mean, i.e.

$$F(t) \sim ct^{-p}$$

with $1 < p < 2$.

We now compute the auto-covariance of X_t. The mean is

$$\mu = E(X_t) = \sum_{i=1}^{I} \pi_i \lambda_i$$

and for $h \geq 1$ we compute

$$r(h) = \text{cov}(X_{t+h}, X_t) = \mathbb{E}((X_{t+h} - \mu)(X_t - \mu))$$

We condition with respect to the event

$$A(t,h) := \{\text{No arrival occurs in} [t+1, t+h]\}$$

Conditional to $A(t,h)$, $X_{t+h} = X_t$, and conditional to non-$A(t,h)$, the rates X_t and X_{t+h} are independent, by construction. Thus

$$r(h) = \mathbb{E}((X_t - \mu)^2 | A(t,h)) \mathbb{P}(A(t,h))$$

Now by construction, X_t is independent of $A(t,h)$. Thus

$$r(h) = \mathbb{E}((X_t - \mu)^2) \mathbb{P}(A(t,h)) = r(0) \mathbb{P}(A(0,h))$$

By Palm's inversion formula (Section 11.3):

$$\mathbb{P}(A(0,h) = \frac{1}{\mathbb{E}(T_n)} \int_{t \geq h} F(t)dt \sim c' h^{-(p-1)}$$

Since $0 < p - 1 < 1$, it follows that X_t is LRD.

Conversely, if $F(t) \leq ct^{-p}$ for some $p > 2$ and for t large enough (fast decay), then T_n is light tailed and $\sum |r(h)| < \infty$ and the process X_t is short range dependent.

This example provides an intuitive explanation for long range dependence, rooted in heavy tail. It remains to see whether this is really an explanation, and whether we need an explanation at all (the quest for an explanation lies on the assumption that SRD would be normal). For an attempt to explain heavy tail in session duration and in file size distributions, see [Downey01-IMW].

OTHER STRUCTURAL MODELS. A similar structural model is the ON-OFF source model, where the on and off periods are iid and mutually independent. Consider the superposition of M such sources and let X_t be the number of sources that are ON at time t. This represents traffic generated by the superposition of unit rate sources. It is shown in [Leland94-ToN] that if either the ON or the OFF period is heavy tailed, then X_t is LRD. This holds for any value of M.

Another class of structural models tries to explain LRD by fractal processes, namely, patterns that are reproduced identically at every scale. For an introduction to such constructions, see [Cappe02-SPM] and [Abry02-SPM].

STRUCTURAL MODELS FOR SRD. For user level sessions, i.e., on-off models of human behaviour, there are some indications that simple, SRD models such as Poisson processes fit well [Paxson95-ToN]. In fact, such models were (successfully) used for dimensioning telephone networks for almost a century.

10.6 TESTS FOR LRD

LRD is tested by estimating the Hurst parameter. A large number of methods exist, see [Taqqu02-html] for an exhaustive list with examples. Many of the methods do not work well. We focus here on two, which do work.

10.6.1 VARIANCE TIME PLOT

This is a simple method, which is easy to understand, but may give some rough results. It consists in verifying asymptotic self-similarity by plotting an estimator $v(m)$ of the variance of the aggregated process:

$$v(m) := \frac{m}{N} \sum_{t=1}^{N/m} \left(X_t^{(m)} - \bar{X}) \right)^2$$

For large enough m, we should find $v(m) \sim c \times m^{-2(1-H)}$, for some constant c. This can easily be verified in log-scale. The corresponding diagram is called a *variance time plot*. See Figures 10.2, 10.5, 10.6 and 10.8.

In practice, the plot may be difficult to interpret because, for large m, where the scaling occurs, we have few data blocks. For example, on Figure 10.18 we find a slope largely less than 1, which is impossible in theory.

10.6.2 LOG-SCALE DIAGRAM

The *Log Scale Diagram* is based on a wavelet analysis of the time series. See Section 13.2 for background information on wavelets. Roughly speaking, for a fixed j, the series of wavelet coefficients d_j, k represent the difference between the time series aggregated by factors of 2^{j-1} and 2^j. The method is based on the fact that wavelet coefficients are short range dependent, even for LRD time series. More precisely, we have [Abry00-book]

THEOREM 10.6.1. *Let X_t be a long range dependent time series. Let $d_{j,k}$ be the wavelet coefficients at octave j (as defined in Section 13.2). If the mother wavelet has N vanishing moments and its Fourier transform is N times differentiable at the origin, then*

- *For any fixed j, $d_{j,k}$ is 2nd order stationary*
- $\mathbb{E}(d_{j,k}) \sim c_1' 2^{j(2H-1)}$ *as* $j \to \infty$
- *For any fixed j, the auto-correlation function of $d_{j,k}$, γ_j satisfies* $\gamma_j(h) \sim h^{2(H-N-1)}$

The second item expresses that the wavelet coefficients reproduce a power law behaviour. Since the number of vanishing moments N is at least 1, the third item means that the wavelet coefficients are short range dependent. The assumption in the theorem are true for all the wavelets usually used.

The log-scale diagram is as follows [Abry00-book]. Let n_j be the number of wavelet coefficients available at octave j. An estimator of $\mathbb{E}(d_{j,k})$ is

$$\mu_j = \frac{1}{n_j} \sum_{k=1}^{n_j} d_{j,k}^2$$

An estimator of $\log \mathbb{E}(d_{j,k})$ is

$$s_j = \log \mu_j - \frac{1}{n_j \log 2}$$

where the last term is an attempt to cancel the bias due to the non-linearity of $\log$. A plot of j versus s_j is called the log-scale diagram. If the points are close to aligned for large j, the slope is an estimate of $\alpha = 2H - 1$.

A confidence interval may be obtained as follows. If the data comes from a normal process (such as FARIMA) the estimator $\hat{\alpha}$ obtained by a least square fit of the s_j to a straight line, over the range $[j_1, j_2]$ where scaling occurs, is approximately normal, with zero mean and variance

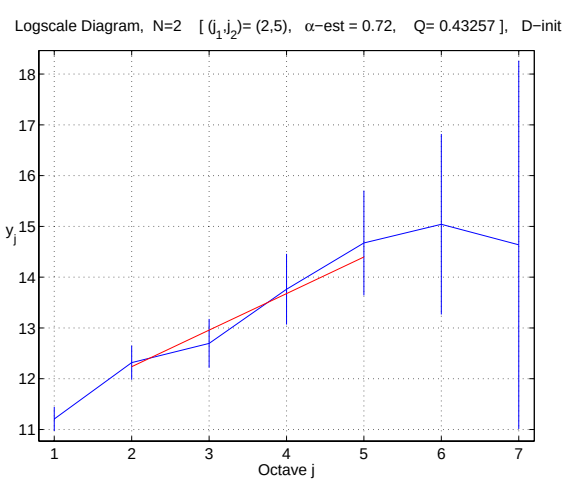
$$v \approx \frac{1}{n} \frac{1 - 2^{-J}}{F}$$

with $J = j_2 - j_1 + 1$, $n = \sum_{j=j_1}^{j_2} n_j$ and $F = (\log 2)^2 2^{1-j_1}(1 - (J^2/2 + 2)2^{-J} + 2^{-2J})$.

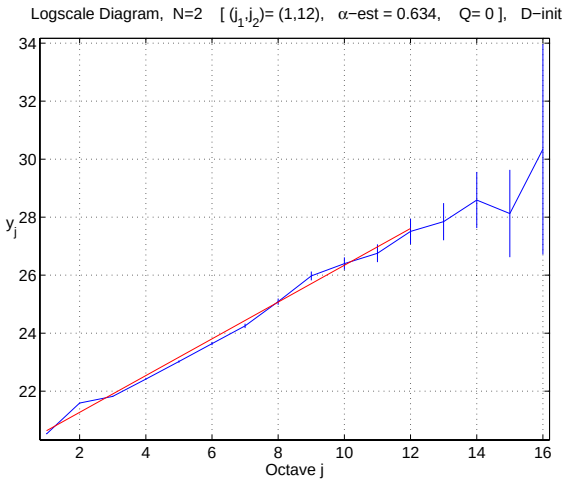
See Figure 10.15 for an application to Ethernet and Nile examples.

QUESTION 10.6.1. *Compare the log-scale diagram estimates to the variance time plot estimates in Figures 10.2, 10.5 and 10.6.* [8]

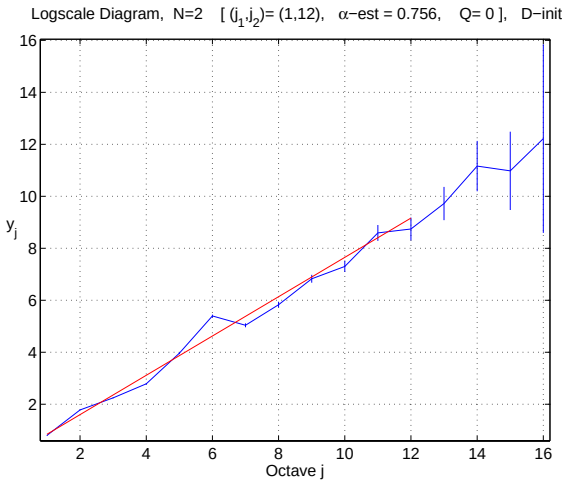
[8]



(a) $[0.722, 0.998]$



(b) $[0.815, 0.819]$



(c) $[0.876, 0.881]$

Figure 10.15: Logscale Diagram for (a) Nile data, (b) Ethernet byte data and (c) Ethernet packet data. LRD is found with confidence intervals for Hurst parameter as shown.

10.6.3 NON-STATIONARITY VERSUS LRD

Non-stationarity and LRD look the same in some respects.

ERRATIC TRENDS that remain after aggregation are a common feature of LRD time series, but also of integrated, non stationary processes such as ARIMA. For example, consider an ARIMA$(0,1,0)$ process, which is not stationary (Figure 10.16). The original time series is

$$X_t = \sum_{n=1}^{t} \epsilon_t$$

where ϵ_t is iid centered normal with variance σ^2. We have $\mathrm{var}(X_t) = t\sigma^2$ and X_T is not stationary. For the aggregated time series, we have

$$\mathrm{var}\frac{1}{\sqrt{m}}X_t^{(m)} = \frac{1}{m}mt\sigma^2 = t\sigma^2$$

thus the aggregated time series remains non-stationary at all aggregation time scales. See also Figure 10.17) for a more sophisticated ARIMA example.

NON-STATIONARITY MAY BE INTERPRETED AS LRD . Remember that LRD is defined for a stationary process.

Consider the Sprint data in Example 9.2 on page 203, (with 6400 data points instead of 250). Figure 10.19 shows the data, the variance and ACF diagrams. The slow decay in ACF and the slope of the variance time plot suggest that LRD is present. Consider now the differenced time series, at lags 1 and 16 (Figure 10.19). The diagrams clearly indicate short range dependence.

Thus, if we have a data set that obviously does not look stationary, considering it as a sample path generated by a stationary process may lead to the conclusion that the process is LRD. If we remove trends from the data set (by differencing), the conclusion may be opposite. Always analyze trends and seasonality before anything else !

WAVELET COEFFICIENTS have the property that polynomial components of degree $\leq N-1$ are cancelled (but not in the coarse approximations). More precisely, if $X_t = Y_t + P(t)$, where $P(t)$ is a deterministic polynomial of degree $\leq N-1$ and Y_t is stationary, then for any j, $d_{j,k}$ is a stationary sequence. The same holds if $\Delta^d X_t = Y_t$, with $d \leq N-1$.

The number of vanishing moments is $N = 1$ for the Haar wavelet and ≥ 2 for other wavelets used in practice. Thus for all j $d_{j,k}$ is a zero mean time series, even if X_t is not. See Figure 10.20.

Thus wavelet based methods are more robust against trends. The same holds for seasonal components, which are removed by the low pass filtering performed when computing the coefficients. Indeed, the log-scale diagram method does not find LRD even if the Sprint data is not differenced (Figure 10.21).

	Log scale diagram estimate	variance time plot estimate
Nile	$[0.722, 0.998]$	0.865
Ethernet Byte	$[0.815, 0.819]$	0.740
Ethernet Packet	$[0.876, 0.881]$	0.814

The estimates are not too far away, but the point estimates are not in the confidence intervals.

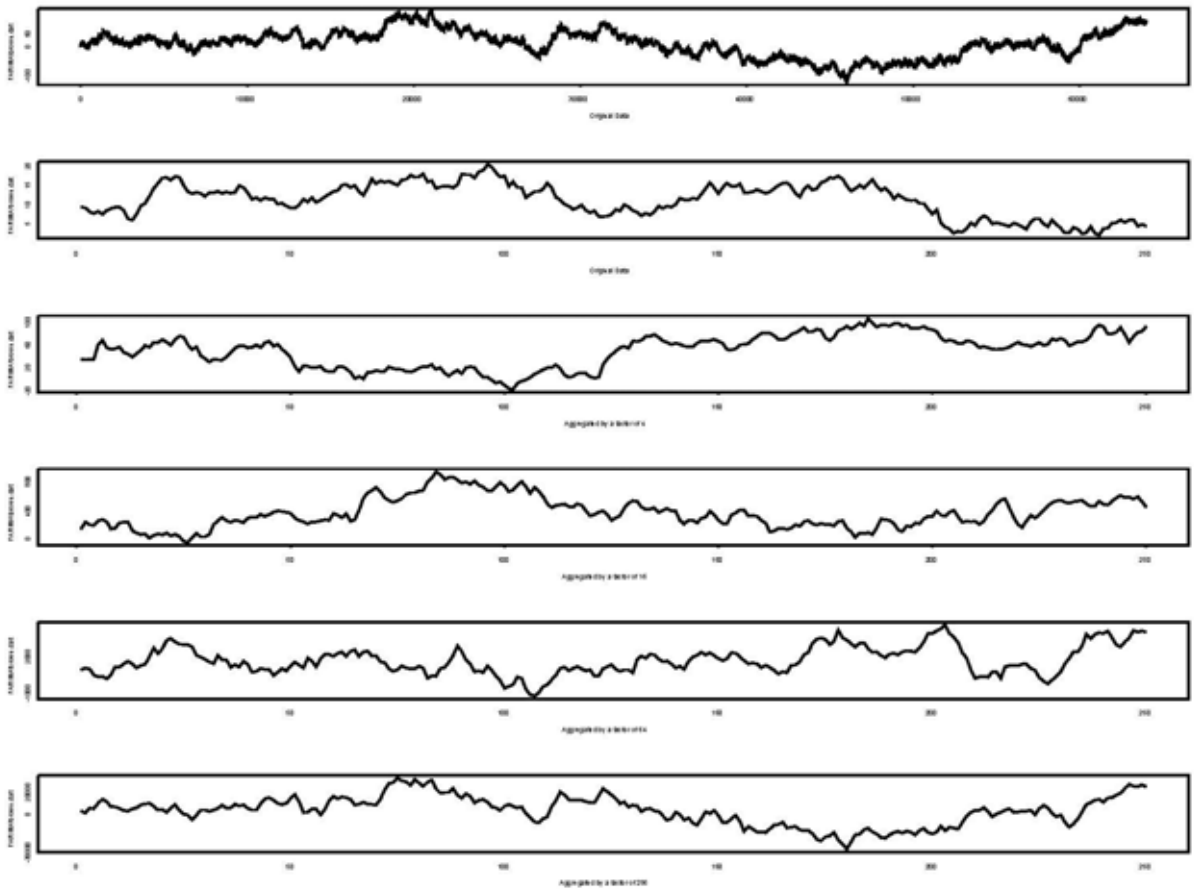


Figure 10.16: Simulation of an ARIMA$(0,1,0)$ model (random walk, discrete time approximation of standard brownian motion. The process is non stationary and remains so after multiple aggregations. It is a self-similar non stationary (short range dependent) process.

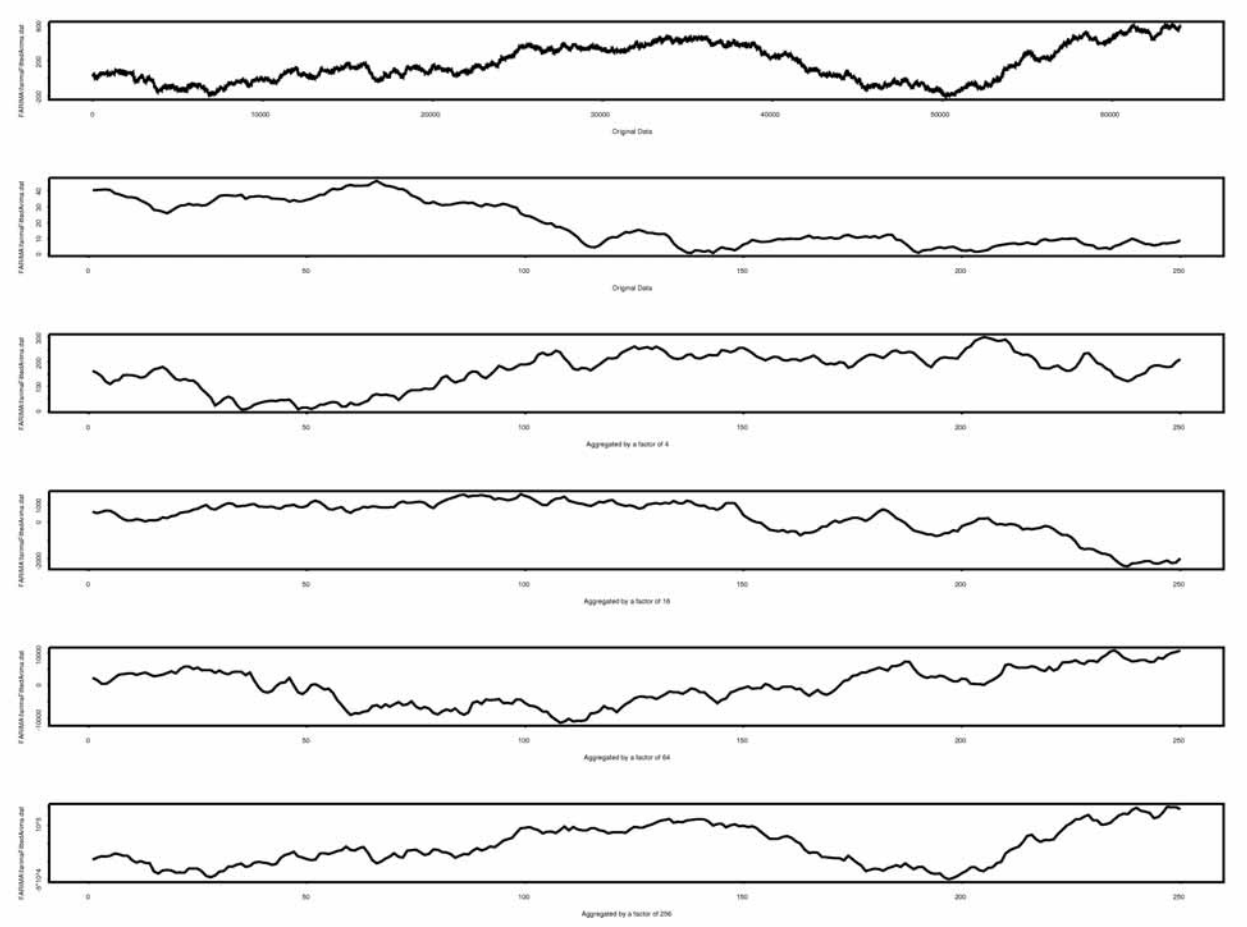


Figure 10.17: Simulation of a (non-stationary) ARIMA$(3, 1, 3)$ model fitted to the (stationary) fractional ARIMA series of Figure 10.14, aggregated at several time scales. We see that the non-stationary time series exhibits the same apparent trend behaviour that resists aggregation.

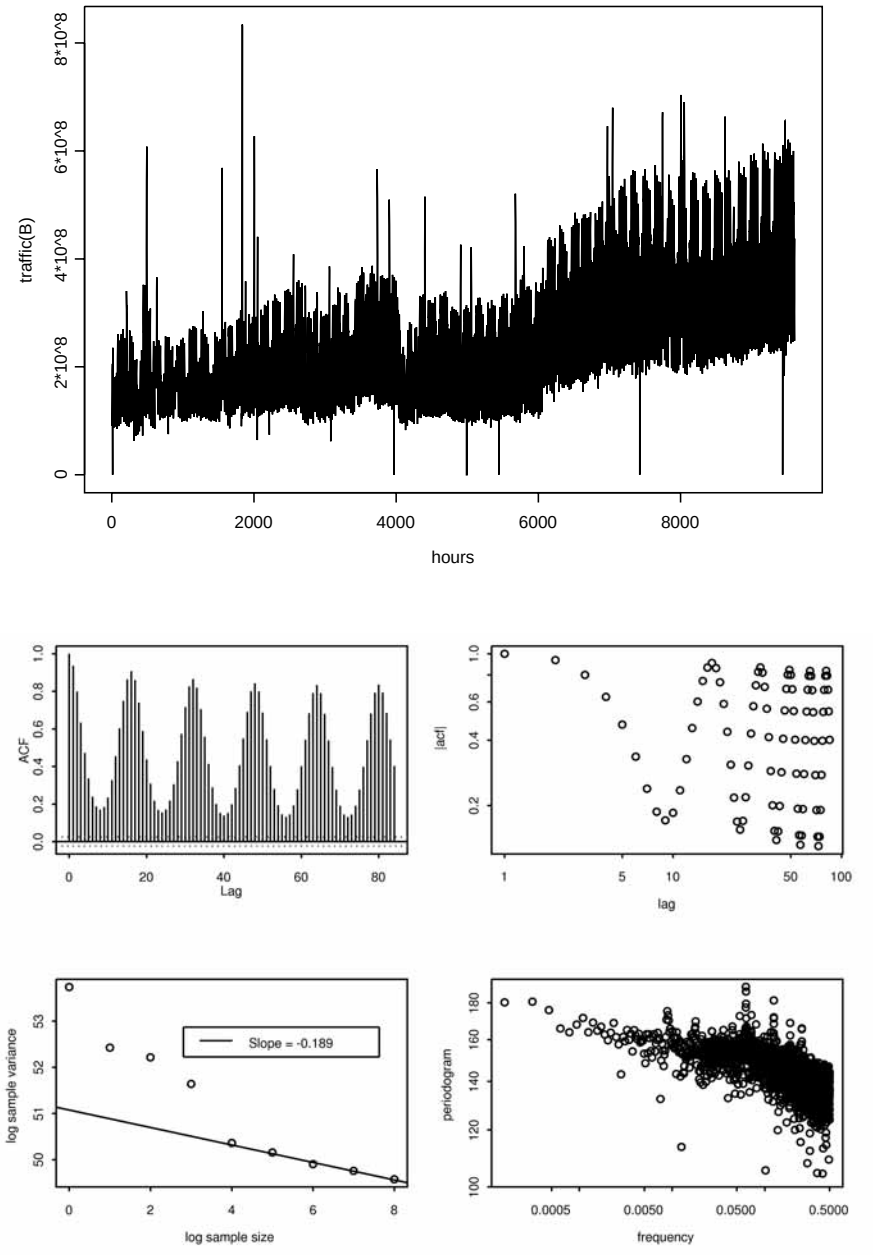


Figure 10.18: Sprint data in Example 9.2 on page 203, original and differenced at lags 1 and 16. The variance analysis of the original time series suggests LRD with $H = 0.905$.

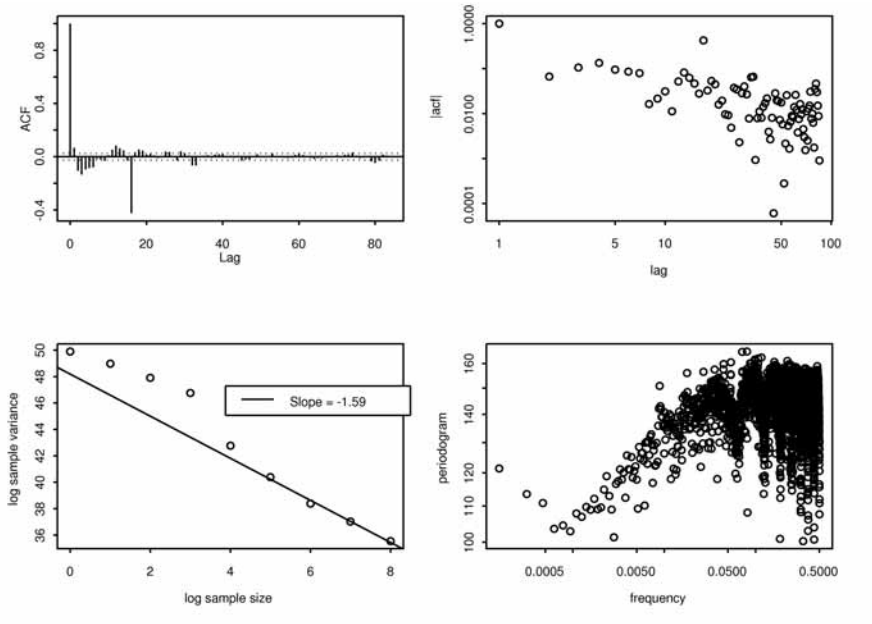


Figure 10.19: Variance analysis of Sprint data in Example 9.2 on page 203, differenced at lags 1 and 16. shows that the time series is *not* LRD. The data in Figure 10.18 should not be assumed to come from a stationary model.

STATIONARY OR NOT ? For a given data set, it often possible to fit a stationary model or a non-stationary one. Consider for example Figure 9.11. We fit a non stationary data model, and this gives us useful information about the growth pattern. In fact, we are interested there in the non stationary part of the data.

In contrast, if the data shows erratic trends, we may not be interested in modeling the trends explicitly, but rather, have a model that will incorporate such trends as random events. ARIMA processes are such models, as well as LRD processes (Figure 10.16).

Distinguishing between ARIMA and LRD models is not easy if the number of points is small. In contrast, if it is large, then we can apply log-scale diagrams. Indeed, polynomial non-stationary components are mostly cancelled by wavelet analysis. If the estimated Hurst parameter is not equal to 0.5, then a long range dependent model should be assumed.

Remember that stationarity is a property of an abstract process, not of the data itself...

10.7 APPLICATIONS

10.7.1 SIMULATION AND CONFIDENCE INTERVALS

REFAIRE EN UTILISANT MON TUTORIAL SIGMETRICS

Assume we want to compute a confidence interval of the mean of some stationary process in one long run, and the data appears to be long range dependent. We cannot apply the sub-sampling method described in Section **??** since correlation persists across long time intervals.

A possible method uses the ACF. We can estimate the ACF by the sample ACF for small lags,

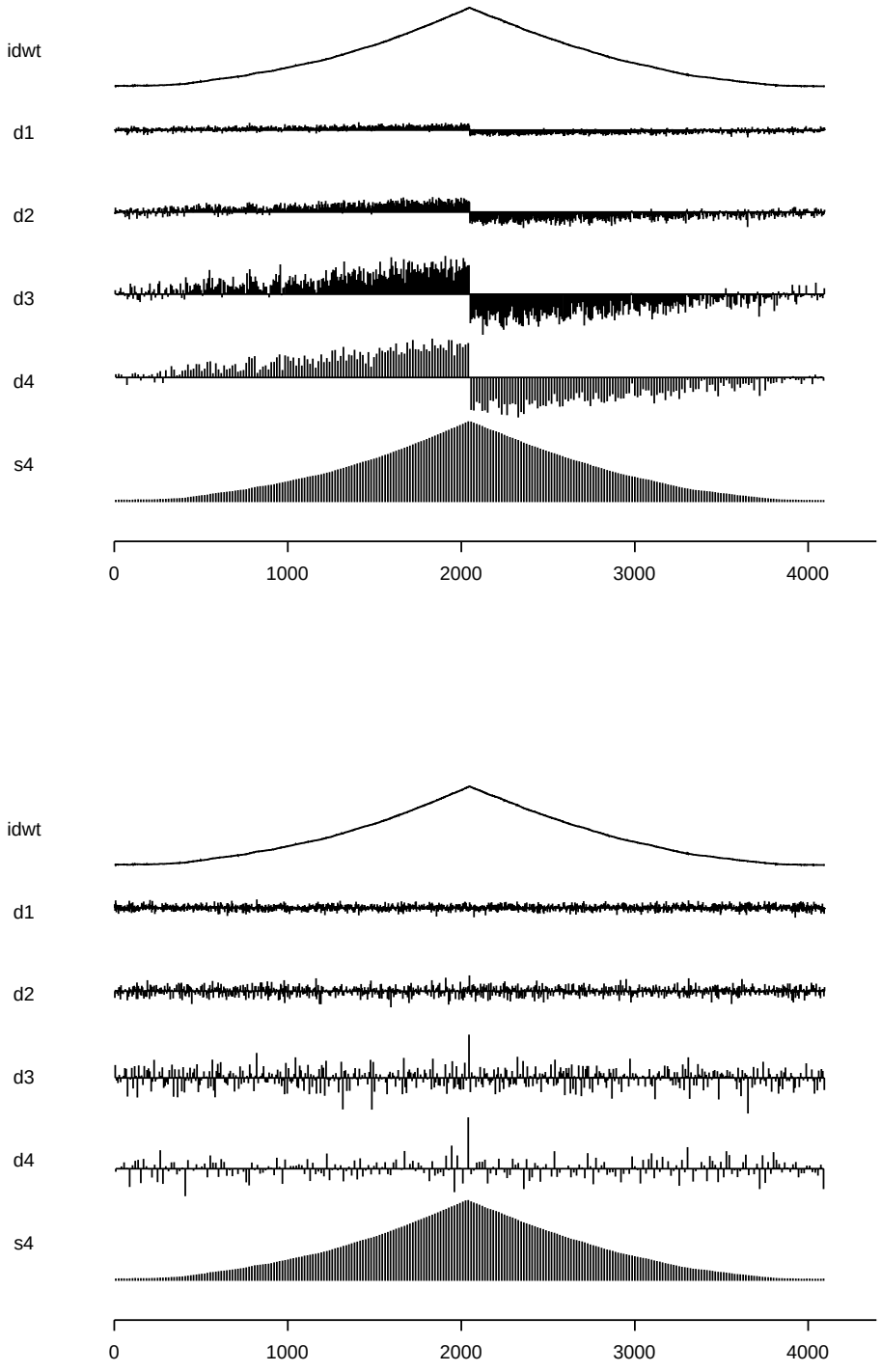
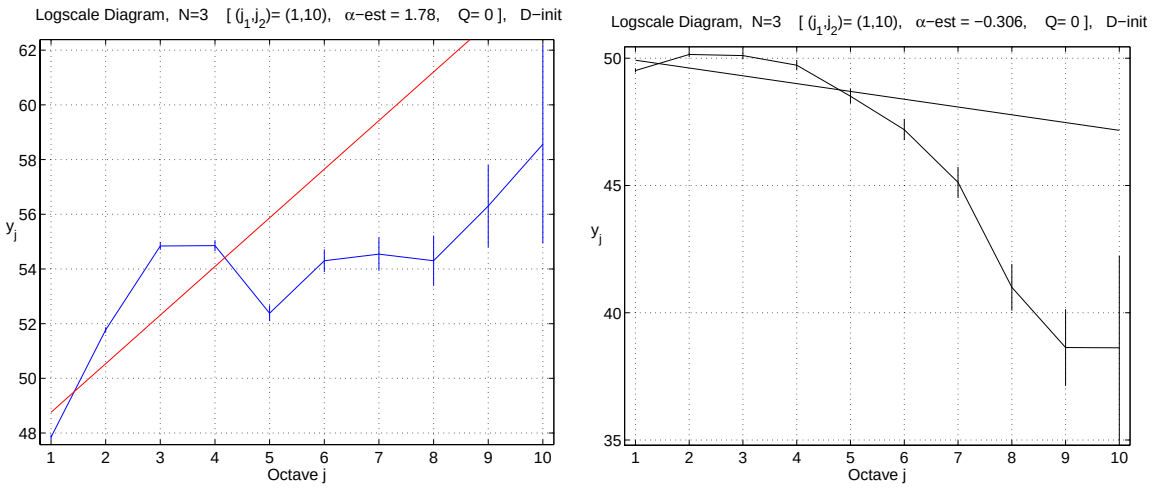


Figure 10.20: Wavelet and scaling coefficients of a process with polynomial trend. First graph: Haar wavelet, which has 1 vanishing moment. The wavelet coefficients are zero mean, but not stationary. Second graph: Daublet-4, which has 2 vanishing moments. The wavelet coefficients look stationary.



(a) (b)

Figure 10.21: Logscale Diagram for Sprint data (a) without differencing (b) with differencing. No LRD is found.

and by its asymptotic expression in Equation (10.1) otherwise. An estimator of the exponent α is given by the log-scale diagram in Section 10.6.2. An extension of the same method can be used to compute the intercept in the log scale diagram, which gives an estimator of $\log c_1$. See [Veitch01-2parms] for a detailed analysis and implementation.

10.7.2 FORECASTING WITH LONG RANGE DEPENDENCE

For LRD processes with finite variance and known auto-covariance function, the forecasting method is essentially the same as for classical time series (Section 9.7.5). In practice, the following methods can be used.

- (FARIMA) Compute MLE as for ARMA. This part is computationally expensive. Forecasts are done as usual.
- (FARIMA with known Hurst paramter) First identify the Hurst parameter. If the confidence interval is small, then fit a fractional ARIMA model to the fractionnally differenced time series. Use the classical methods for prediction.
- (ARIMA) An ARIMA model is non stationary and, over short period of times, may be able to track the apparent trends of an LRD process. The method here consists in fitting an ARIMA model to the recent data and use the classical method for prediction. The model is fitted again periodically. See [Bansali] for details.
- (Wavelet Analysis) Decompose the time series into a multi-resolution analysis: model the details by a joint, multi-dimensional ARMA process or Kalman filter. Model the coarse approximation by a simple regression model. See [Pappagiannaki03-infocom] for an example in that direction.

Peter Dinda reports in [Dinda99-HPDC] that, for host load prediction, FARIMA models perform marginally better than ARIMA models. Figure 10.22 confirms that fact: the predictions for both

FARIMA and ARIMA are close (and almost equal to the mean), but the confidence interval is smaller with FARIMA. [Beran-94] finds that this is a general finding: long memory helps finding smaller confidence intervals.

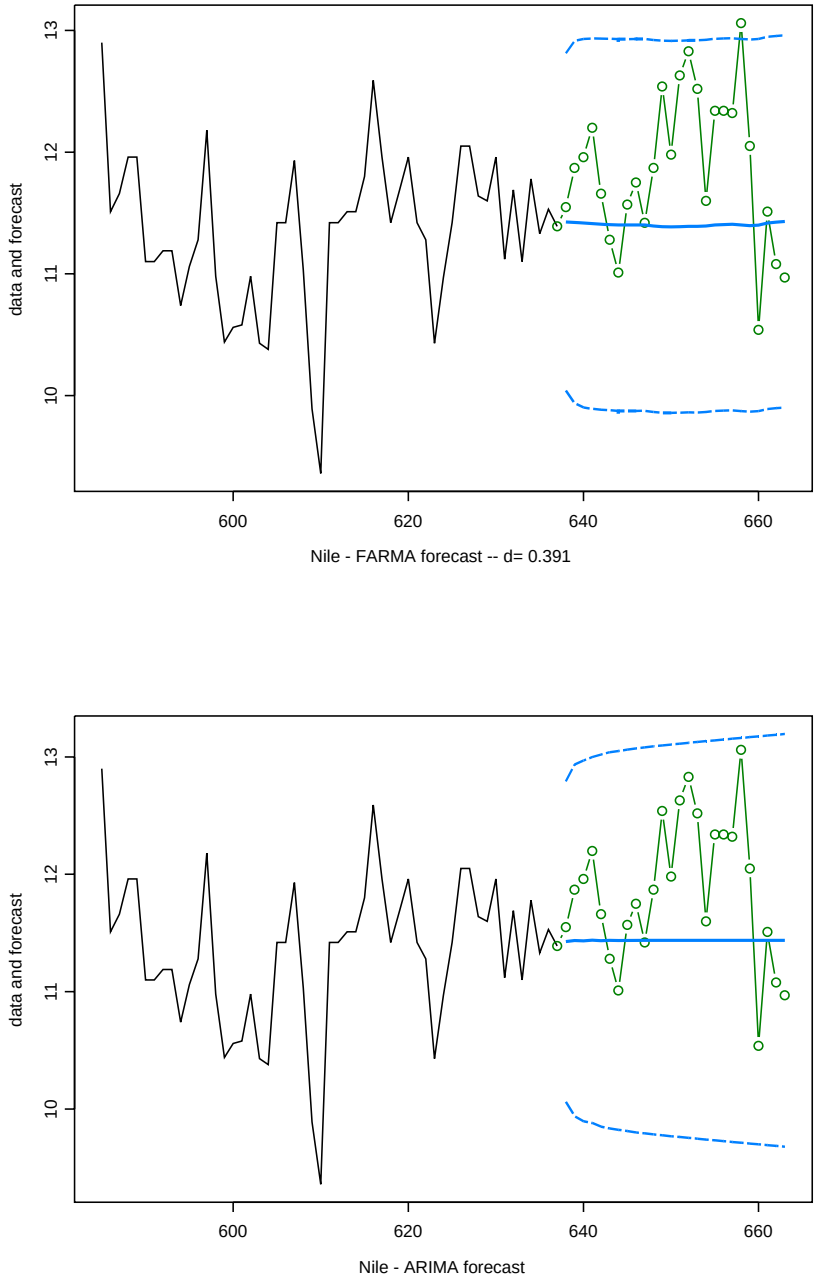


Figure 10.22: Forecasts for Nile data, fitted on the time series minus the last 26 data points. Top: Best FARIMA(p,d,q) model for $p, d \leq 2$. The estimation found $d = 0.391$ ($H = 0.891$). Bottom: best ARIMA(p,1,q) model for $p, d \leq 5$.

10.8 REVIEW AND QUESTIONS

Hurst Parameter	$0.5 \leq H < 1$	$H = 0.5$ means SRD	$H = 1 - \frac{\alpha}{2}$
Decay Exponent of ACF	$0 < \alpha \leq 1$	$\alpha = 1$ means SRD	$\alpha = 2(1 - H)$
FARIMA(p, d, q)	$0 \leq d < 0.5$	$d = 0$ means SRD	$d = H - \frac{1}{2} = \frac{1-\alpha}{2}$

QUESTION 10.8.1. *For each of the following process type, say if it is stationary (assume that the auto-regressive polynomials have all roots outside the unit disk):*

1. *ARMA*
2. *ARIMA(p, d, q) (assume that $d \geq 1$ and $d \in \mathbb{N}$)*
3. *FARIMA(p, d, q) (assume that $0 < d\frac{1}{2}$)*
4. *fGn(H) (assume that $\frac{1}{2} \leq d < 1$)*

[9]

QUESTION 10.8.2. *Same question with : "self-similar" instead of "stationary"* [10]

QUESTION 10.8.3. *Same question with : "long-range dependent" instead of "stationary"* [11]

QUESTION 10.8.4. *What is the difference between a FARIMA process and fGn ?* [12]

QUESTION 10.8.5. *Is there a difference between fractionally integrated noise and fGn ?* [13]

QUESTION 10.8.6. *What are the differences between LRD and self-similarity ?* [14]

QUESTION 10.8.7. *What is the difference between heavy tail and LRD ?* [15]

QUESTION 10.8.8. *How can I know if my time series is LRD ?* [16]

QUESTION 10.8.9. *If a time series is non-stationary, does this go away by aggregation ?* [17]

QUESTION 10.8.10. *Is it possible to have a stationary model and a non-stationary one for the same data ?* [18]

[9] ARMA, FARIMA and fGn are stationary. ARIMA is not.

[10] fGn is self-similar; ARMA, FARIMA and ARIMA are not (except for special cases)

[11] ARMA is not LRD. ARIMA is not stationary so the question of LRD does not apply. FARIMA and fGn are LRD for $H \neq 0$.

[12] FARIMA has both LRD and a short range structure that can be exploited to fit some data that is not strictly self-similar.

[13] Yes. The former is not self-similar (its auto-covariance function does not have the right form, whereas the latter is. Both are noise models with LRD.

[14] Self similarity is a property of aggregated processes. Aggregation tends to produce self-similarity. If the original data is SRD, the aggregation limit is white noise (=fGn 0.5)(after proper re-scaling); if the original data is LRD, the aggregated data is fGn with same hurst parameter.

[15] Heavy tail is a property of the distribution of one random variable. LRD is a property of the ACF of a second order process. If we build a processes by superposing indepedent on-off sources that have independent on and off period, when the On (or Off) duration is heavy tailed, the process is LRD, and conversely.

[16] First make sure it looks stationary. Then look at variance time plots in log-log scales, or (better) use the scalogram method.

[17] In general no. There are exceptions: seasonal components go away by aggregation, but trends do not. Random trends like in ARIMA(0,1,0) do not go away.

[18] Yes, if the data set is large and exhibits apparent trends. An ARIMA-like model (non-stationary) and a LRD, stationary model (for example: FARIMA) may both explain the data well.

10.9 EXERCICES

EXERCISE 10.1. *Compute the Hurst parameter of the Sprint, Dinda and Ethernet traces. Cut the traces in two and re-do the computations. What do you find ? Use confidence intervals (see Darryl Veitch's tools).*

EXERCISE 10.2. *Compute the Hurst parameter for the traffic load generated by Surge. Explain the result.*

EXERCISE 10.3. *Read [grossglauser96-sigcomm] and answer the following questions.*

- *What is the main finding of the paper ?*
- *What is the model ? How is LRD created ?*
- *How is the queuing system solved ?*

EXERCISE 10.4. *Compute confidence intervals for the bottleneck utilisation in your SURGE experiment, using a single long run.*

EXERCISE 10.5. *Forecast the Dinda time series using methods that account for LRD. Compare to methods that do not. Same question for the Sprint traces.*

USEFUL S-PLUS COMMANDS :

- `arima.fracdiff` fit a FARIMA model
- `arima.fracdiff.sim` simulate a FARIMA process
- `gamma(), lgamma`: $\Gamma()$ and $\log \Gamma()$

CHAPTER 11

PALM CALCULUS OR THE IMPORTANCE OF THE VIEWPOINT

11.1 INTRODUCTION

11.1.1 THE IMPORTANCE OF THE VIEWPOINT

EXAMPLE 11.1: VIDEO SERVER. Consider the question mentioned in Chapter **??**: "A video server starts the film on a channel three times per hour. Is it fair to say that the average waiting time is $60mn/3/2 = 10mn$?". The operator viewpoint may be that a performance metric is that $\lambda = 3$ films per hour are started. A customer may have a different viewpoint. If she connects to the system at a random instant, she will have to wait until the next movie starts, and we may take as performance metric the average waiting time. Assume for example that films are started at the hour, the hour plus 5mn, and the hour plus 20 mn. We can compute the average waiting time by assuming that our customer picks a minute at random uniformly in the hour. She will thus experience an average waiting equal to

$$W_c = \frac{5}{60} \times 2.5mn + \frac{15}{60} \times 7.5mn + \frac{40}{60} \times 20mn = 15mn25s$$

In an attempt to become customer oriented, the provider might change his performance metric and compute the average time between films, as follows.

$$X_p = \frac{1}{3}5mn + \frac{1}{3}15mn + \frac{1}{3}40mn = 20mn$$

and since the average time between films is 20 mn, the average waiting time is estimated by the provider to $W_p = X_p/2 = 10mn$.

This shows the importance of the viewpoint. Both computations may look reasonable, in some sense. In this chapter, we give a framework to analyze the two viewpoints and how they relate.

This example illustrates the use of conditional probability. More generally, it is important to scrutinize the model used, whenever a probability or an average is used as performance metric. Any probabilistic result, must come with model assumptions.

11.1.2 PALM CALCULUS

Consider now another example. Assume you are simulating a complex system (for example a rate control protocol). You can sample the system at an arbitrary time instant (an external observer comes at random, hits the stop key and looks at the system state), or at arbitrary points of interest (arrivals of feedback messages in the protocol). How do the two relate ?

We will see that Palm Calculus can be use to relate the two viewpoints.

We give the Palm calculus results for both discrete and continuous times, but expose the theory for discrete time only. The continuous time framework is obscured by constructions that are needed for existence and stability, but which make the theory difficult to access. In contrast, computations are sometimes a little more cumbersome in discrete time. We also leave out proofs of stationarity and ergodicity, which are often difficult problems.

Note that the same framework as we show here is used in stochastic geometry [Stoyan]. After reading this chapter, the alert reader will find it considerably easier to understand stochastic geometry concepts (see also exercise 11.7).

11.2 STATIONARITY

WHAT IS STATIONARITY ? Stationarity is a property of a model. A stochastic model X_t is *stationary* if, for any finite sequence of times $t_1, t_2, ..., t_n$ and for any time offset v the joint distribution of $X_{t_1+v}, X_{t_2+v}, ..., X_{t_n+v}$ is independent of v.

It means in practice that the system does not become older: a stationary system is one for which there is no way to gain any information about the age of the system by by looking at its output.

Examples of *non stationary* models fall in the two broad following categories.

- **unstable** models: observe the buffer length in a queuing system where the input rate is larger than the service capacity. The longer the simulation is run, the larger the queue length is.
- models with **seasonal or growth** components, or more generally, time dependent inputs; for example: internet traffic grows month after month and is more intense at some times of the day

Figure 3.1 on Page 64 illustrates the simulation of stationary and non-stationary models.

In many practical cases, we want to separately analyze the effect of time varying inputs (such as seasonal variations) and of the internal dynamics of the system; we then use a stationary model of the system.

WHEN IS A SYSTEM STATIONARY ? There is no formal answer to this question. Informally, we think of a system as being non-stationary if the well accepted models of the system are stationary. A model is well accepted if it provides useful answers to some questions. In Chapter 10 we see some cases where both stationary and non-stationary models can apply to the same data.

ASYMPTOTIC STATIONARITY AND MARKOV CHAINS Most simulated models can be interpreted, at least theoretically, as Markov chains. Thus it is important to understand stationarity for Markov chains.

Consider a Markov chain on a enumerable state space (we recall in appendix the basic properties of Markov chains). A Markov chain is *stable* when

- it is irreducible (any state can be reached from any state)
- there exists a non zero solution to the balance equation

For a stable Markov chain, a solution of the balance equation is unique (up to a multiplicative constant). The only solution that sums to 1 is called the *stationary probability*.

Now the distribution of an ergodic Markov chain converges exponentially fast to the stationary distribution.

This explains why we can think of stationarity as the regime obtained after running a simulation long enough.

A Markov chain is strictly stationary if the distribution of the state space at time $t = 0$ is the stationary distribution.

QUESTION 11.2.1. *Can a model be non-stationary while all of its inputs have a time independent distribution ?* [1]

11.3 PALM PROBABILITY

11.3.1 STATIONARY POINT PROCESS

We wish to model a sequence of events that is in stationary regime. Think of it as as a simulation that has been running for a long time. The mathematical framework for that is the *Stationary Point Process* [Baccelli88-book].

A point process in discrete time is a sequence of random time instants $T_n \in \mathbb{Z}$ with $n \in \mathbb{Z}$. It is also convenient to use the counting time series instead of the sequence of points T_n.

DEFINITION 11.3.1. *In discrete time, the counting time series N associated with a point process is the random sequence defined by $N(t) = \sum_{n\in\mathbb{Z}} 1_{\{T_n=t\}}$, i.e. $N(t)$ is the number of points at t.*

A point process is defined either by the sequence of time instants T_n or the counting time series N. In continuous time, the counting time series is replaced by the random counting *measure* defined by $N(I) =$ the number of points in the interval I.

DEFINITION 11.3.2. *A stationary point process in discrete time is a sequence of time instants $T_n \in \mathbb{Z}$ with $n \in \mathbb{Z}$, such that*

- *(stationarity) the corresponding counting time series $N(t)$ is stationary*
- *(simple point process) for any $t \in \mathbb{Z}$, $\mathbb{P}(N(i) > 1) = 0$*

[1] Yes, if it is unstable.

- *(liveness) with probability 1, there are infinitely many points in any unbounded interval.*

We consider only simple point processes. This means that the counting time series $N(t)$ is always equal to 0 or 1. Stationarity means that for any finite sequence $t_1 < t_2 < ... < t_n$ and for any u the distribution of $N(t_1 + u), N(t_2 + u), ..., N(t_n + u))$ is independent of u. See Chapter 9 for testing whether a time series is stationary.

EXAMPLE 11.2: BUSES AT SAINT-FRANÇOIS. You stand at the bus stop and observe buses passing by. T_n is the sequence of bus arrival instants. If you arrive at an arbitrary time and board the next bus, you experience only one point T_n. A bus inspector who measures all bus arrival epochs is able to give an estimate of the time series $N(t)$. We do *not* assume here that the bus interarrival times $T_n - T_{n-1}$ are iid.

EXAMPLE 11.3: RENEWAL SOURCE MODEL. Consider one infinite iid sequence of positive numbers U_n, $n \in \mathbb{Z}$. Run a simulation as follows. Draw a point at $T_1 = U_1$, then at $T_2 = U_1 + U_2$, etc. Run the simulation long enough for it to reach steady state. (We will see a more rigorous solution later). For any interval I, $N(I)$ is the number of points in that interval. Such a sequence is used in some traffic models, where T_n is the time at which a source changes its rate (Section 10.5).

EXAMPLE 11.4: POISSON PROCESS. A Poisson process is a point process in continuous time. For any interval $[a, b]$, $N[a, b]$ is a Poisson random variable, i.e. $P([a, b] = k) = \frac{\lambda^k}{k!} e^{-k\lambda}$ for some $\lambda > 0$. If two intervals I and J are disjoint, then $N(I)$ and $N(J)$ are independent.

11.3.2 INTENSITY

DEFINITION 11.3.3. *By stationarity, $\lambda = \mathbb{E}(N(t))$ is independent of t and is called the intensity of the point process.*

Consider now an arbitrary subset I of time instants. $\mathbb{E}(N(I)) = \sum_{i \in I} \mathbb{E}(N(i)) = \lambda \sum_{i \in I} 1$. Thus

$$\mathbb{E}(N(I)) = \lambda |I|$$

where $|I|$ is the number of elements in I. In continuous time, the formula is the same, with $|I|$ equal to the "length" (Lebesgue measure) of I, and we usually require that I is measurable. Thus

- in continuous time: $\mathbb{E}(N[a, b]) = \lambda(b - a)$

- in discrete time: $\mathbb{E}(N(a,b]) = \lambda(b-a)$

QUESTION 11.3.1. *In continuous time, what is* $\mathbb{E}(N(a,b])$ *?* [2]

QUESTION 11.3.2. *In discrete time, what is* $\mathbb{E}(N[a,b])$ *?* [3]

For a Poisson process, λ is the usual one. For the renewal source model, we will see below that $\lambda = 1/(\mathbb{E}(U_n)$.

In a simulation, λ is estimated by the number of points per time unit during steady-state.

We assume the following [Baccelli88-book]:

- $\lambda > 0$

For the Poisson process, this is naturally true. For the renewal source model, this corresponds to $\mathbb{E}(U_n) < +\infty$.

QUESTION 11.3.3. *What is the intensity for the video server example in Section 11.1.1 ?* [4]

11.3.3 PALM PROBABILITY

THE ARBITRARY TIME INSTANT. In the rest of this chapter we use the following convention.

- The time instants T_n are such that $... < T_{-2} < T_{-1} < T_0 \leq 0 < T_1 < T_2 < ...$

In other words, we call by convention T_0 the time instant just before or at time 0. This convention is the one used by mathematicians to give a meaning to "a random time instant": we regard $t = 0$ as our random time instant, in some sense, we fix the time origin arbitrarily.

This differs from the convention used in many simulations, where $t = 0$ is the beginning of the simulation. Our convention, in this chapter, is that $t = 0$ is the beginning of the observation period for a simulation that has run long enough to be in steady state.

PALM PROBABILITY AND EXPECTATION.

DEFINITION 11.3.4.

Time is discrete. Given a point process T_n*, the Palm probability* P^0 *is the conditional probability, given that* $T_0 = 0$ *(i.e., given that there is a point at time* 0*).*

There is a similar definition for the *Palm expectation*. The Palm probability represents the point of view obtained by sampling a system at times T_n.

Why use a special notation ($\mathbb{P}^0$) instead of the classical conditional probability notation $\mathbb{P}(...|T_0 = 0)$? The reason is that, in continuous time, the conditional probability is not defined, since the probability that a point occurs exactly at time 0 is 0. However, the Palm probability still exists and all properties are the same as in discrete time. The rigorous definition is complicated: see [Baccelli88-book].

[2] $\lambda(b-a)$
[3] $\lambda(b-a+1)$
[4] $\lambda = 3$ per hour.

EXAMPLE: BUSES AT SAINT-FRANÇOIS. $\mathbb{E}^0(T_1)$ is the average time between buses, seen by an inspector standing at the bus stop and who spends the hour counting intervals from bus to bus. $\mathbb{E}(T_1)$ is the average waiting time experienced by you and me when we come to the bus stop at some arbitrary time instant and wait for the next bus. We will see that $\mathbb{E}^0(T_1) = \frac{1}{\lambda}$.

EXAMPLE: RENEWAL SOURCE MODEL. Consider the following special case. Assume U_n is constant, equal to the same value u. Let X_t be the time duration from t to the next point. Given that there is a point at $t = 0$, the time duration until the next point is x, thus $\mathbb{E}^0(X_0) = u$. In contrast, if we pick a random instant as beginning of observation period, we should fall anywhere between two points, thus we expect to have $\mathbb{E}(X_0) = \frac{u}{2}$. We will give a formal proof later.

EXAMPLE: POISSON PROCESS. Let X_t be the time duration from t to the next arrival. Then $P^0(X_0 > x) = P(X_0 > x) = e^{-\lambda x}$, in other words, X_t is an exponential random variable, both under P and P^0.

REMARK. One should be careful with the convention that $T_0 \leq 0 < T_1$. Indeed, once we accept it, $T_{n+1} - T_n$ is no longer the interval between two arbitrary consecutive points. In contrast, it is the nth interval that follows an arbitrary point in time. In this framework, the distribution of the interval between two arbitrary consecutive points is the Palm distribution of $T_1 - T_0$ (= the Palm distribution of T_1). For example, for the renewal source model mentioned above, we should now write $\lambda = \frac{1}{\mathbb{E}^0(U_1)}$.

QUESTION 11.3.4. *For the video server example in Section 11.1.1, what is (1) the Palm expectation of the time between films (2) the expected time from an arbitrary instant to the start of the next film ?* [5]

QUESTION 11.3.5. *Under P^0, what is the probability that $T_0 = 0$?* [6]

11.3.4 JOINT STATIONARITY.

Consider both a point process T_n with counting time series $N(t)$, and some process X_t, on the same probability space (i.e., both T_n and X_t are observed during the same simulation). X_t can take values in any space.

[5] (1) $20mn$ (2) $15mn25s$.
[6] 1.

DEFINITION 11.3.5. *We say that T_n, X_t are jointly stationary iff the process $(N(t), X_t)$ is strictly stationary. This means that for any finite sequence of times $t_1 < t_2 < ... < t_n$ the distribution of*

$$(N(t_1 + u), X_{t_1+u}, N(t_2 + u), X_{t_2+u}, ..., N(t_n + u), X_{t_n+u})$$

is independent of u.

If T_n, X_t are jointly stationary, then X_t is strictly stationary. Intuitively, the process X_t moves with T_n whenever we change the time origin. We will freely use informal synonyms such as "X_t is jointly stationary with T_n".

PROPOSITION 11.3.1. *If T_n, X_t are is jointly stationary then for any bounded deterministic function $f()$: $E^t(f(X_t)) = E^0(f(X_0))$*

Proof. By definition, $E^t(f(X_t)) = \mathbb{E}(f(t)|N(t) = 1) = \frac{1}{\lambda}\mathbb{E}(f(X_t)1_{\{N(t)=1)\}})$ which, by stationarity, is independent of t.

□

Thus, both $E(f(X_t))$ and $E^t(f(X_t))$ are independent of t, for any bounded f.

EXAMPLE 11.5: ELAPSED AND RUNNING TIMES. We use the following notation. Let $T^+(t)$ [resp. $T^-(t)$] be the first point after [resp. before or at] t. Thus, for example, $T^+(0) = T_1$ and $T^-(0) = T_0$.

Let $X_t = T^+(t) - t$ (time until next point), $Y_t = t - T^-(t)$ (time since last point), $Z_t = T^+(t) - T^-(t)$ (duration of current interval). Then (X_t, Y_t, Z_t) is jointly stationary with T_n.

QUESTION 11.3.6. *Is $T^+(t)$ stationary ?* [7]

EXAMPLE 11.6: MARKOV CHAIN. Consider an irreducible finite Markov chain in discrete time and $T_n = n$. Joint stationarity is true iff the initial distribution is the stationary distribution of the Markov chain.

THINNING. At every arrival T_n of a stationary point process we associate a type $I_n \in \{1, 2, , , M\}$. Consider the *thinned* point process T_n^i obtained by selecting those points for which $I_n = i$. If I_n is the value of the type at time T_n, then T_n^i is stationary. Let λ_i be its intensity.

QUESTION 11.3.7. *Show (in discrete time) that $\lambda = \sum_{i=1}^{M} \lambda_i$.* [8]

[7] No, it is not stationary – its mean is larger for large t.

[8]

$$\lambda = \mathbb{P}(N(t) = 1) = \sum_{i=1}^{M} \mathbb{P}(N(t) = 1 \text{ and } I_n = i)$$

Now $\lambda_i = \mathbb{P}(N(t) = 1 \text{ and } I_n = i)$.

11.3.5 ERGODIC INTERPRETATION OF PALM PROBABILITY.

We say that X_t is *ergodic* if the sample path averages of any bounded function of X_t converge to a non-random number. If, in addition, X_t is strictly stationary, then this limit is necessarily the expectation of $f(X_t)$ (which is independent of t). In particular, this implies, in discrete time:

$$\mathbb{E}(f(X_0)) = \mathbb{E}(f(X_t)) = \lim_{T\to+\infty} \frac{1}{T}\sum_{s=1}^{T} f(X_s)$$

and in continuous time

$$\mathbb{E}(f(X_0)) = \mathbb{E}(f(X_t)) = \lim_{T\to+\infty} \frac{1}{T}\int_{s=0}^{T} f(X_s)ds$$

For a stationary ergodic system, we can thus interpret a stationary probability $\mathbb{P}(X_t \in A)$ as a *time average*.

The strong law of large numbers says that an iid sequence with finite mean is ergodic. An irreducible, finite, aperiodic Markov chain is ergodic. If we remove the finite assumption, ergodicity requires that the chain is positive (i.e. there exists a stable solution to the Kolmogorov equations), which is a stability argument. This is quite general: a process is ergodic if it is stable and mixes well (any state can be reached from any state).

We say that X_t, T_n constitute an *ergodic-stationary* system if they are jointly stationary and the process $X_t, N(t)$ is ergodic. For an ergodic-stationary system, we have the following result:

$$\mathbb{E}^0(f(X_0)) = \lim_{N\to+\infty} \frac{1}{N}\sum_{n=1}^{N} f(X_{T_n})$$

This gives the interpretation of Palm probability as an *event average*. Note that the various formulae (direct, inversion, Campbell) do not require ergodicity; but their interpretation is simple if the system is ergodic.

EXAMPLE 11.7: STOP AND GO PROTOCOL. A source sends packets to a destination. Error recovery is done by the stop and go protocol, as follows. When a packet is sent, a timer, with fixed value S_1, is set. If the packet is acknowledged before S_1, transmission is successful. Otherwise, the packet is re-transmitted. The packet plus acknowledgement transmission and processing have an constant duration equal to $S < S_1$. The proportion of successful transmissions (fresh or not) is α. We assume that the source is greedy, i.e., always has a packet ready for transmission. Can we compute the throughput of this protocol without further information ?

An ergodic interpretation gives the answer. Call $N(t)$ the number of successfully transmitted packets over some long period of time $[0, t]$ and $S_{av}(t)$ the average time to successfully transmit a packet, measured over this interval. Thus $S_{av}(t) = t/N(t)$. Call λ the throughput and assume the system is stationary ergodic. We have $\lim_{t\to+\infty} N(t)/t = \mathbb{E}(N(0)) = \lambda$ and thus

$$S_{av} := \lim_{t\to+\infty} S_{av}(t) = 1/\lambda$$

Let $N_1(t)$ be the number of timeouts occurring during the same time. We have

$$t = N(t)S + N_1(t)S_1 + \epsilon$$

where $epsilon$ is an error term, bounded by S_1. Call λ_1 the intensity of the timeout process. Divide the above by t, let t go to infinity and obtain:

$$1 = \lambda S + \lambda_1 S_1$$

We need one more equation, in order to compute λ_1. Call $A(t)$ the fraction of packets or acknowledgements lost in $[0, t]$; we have

$$A(t)\left(N(t) + N_1(t)\right) = N_1(t) \pm 1$$

Further, $\lim_{t\to+\infty} A(t) = \alpha$, thus, dividing the above by t gives, at the limit

$$\alpha(\lambda + \lambda_1) = \lambda_1$$

Combining the equations gives

$$S_{av} = S + \frac{\alpha}{1-\alpha} S_1$$

and the throughput is $\lambda = 1/S_{av}$.

11.3.6 RYLL-NARDZEWSKI AND SLIVNYAK'S INVERSION FORMULA

THEOREM 11.3.1 (Inversion Formula). *If T_n, X_t is jointly stationary, then, in discrete time*

$$\mathbb{E}(X_0) = \lambda \mathbb{E}^0\left(\sum_{s=1}^{T_1} X_s\right) = \lambda \mathbb{E}^0\left(\sum_{s=0}^{T_1-1} X_s\right)$$

and in continuous time

$$\mathbb{E}(X_0) = \lambda \mathbb{E}^0\left(\int_0^{T_1} X_s ds\right)$$

Proof. (discrete time) We show first that $\mathbb{E}(X_0) = \lambda \mathbb{E}^0\left(\sum_{s=1}^{T_1} X_s\right)$. Condition the main term in the right hand-side with respect to $T_1 = t_1$:

$$\mathbb{E}^0\left(\sum_{s=1}^{T_1} X_s | T_1 = t_1\right) = \sum_{s=1}^{t_1} \mathbb{E}^0(X_s | T_1 = t_1) = \sum_{s=1}^{t_1} \frac{\mathbb{E}(X_s 1_{\{T_1=t_1\}} 1_{\{T_0=0\}})}{\mathbb{P}(T_1 = t_1, T_0 = 0)}$$

thus

$$\mathbb{E}^0\left(\sum_{s=1}^{T_1} X_s\right) = \sum_{t_1=1}^{+\infty} \sum_{s=1}^{t_1} \frac{\mathbb{E}(X_s 1_{\{T_1=t_1\}} 1_{\{T_0=0\}}) \mathbb{P}(T_1 = t_1, T_0 = 0)}{\mathbb{P}(T_1 = t_1, T_0 = 0)\mathbb{P}(T_0 = 0)}$$

Multiply by λ and obtain, for the right-handside:

$$RHS = \sum_{(s,t_1):1\le s\le t_1} \mathbb{E}(X_s 1_{\{T_1=t_1\}} 1_{\{T_0=0\}})$$

Re-arrange the summation by summing first with respect to t_1 and obtain

$$RHS = \sum_{s=1}^{+\infty} \sum_{t_1=s}^{+\infty} \mathbb{E}(X_s 1_{\{T_1=t_1\}} 1_{\{T_0=0\}}) = \sum_{s=1}^{+\infty} \mathbb{E}(X_s 1_{\{T_1\ge s\}} 1_{\{T_0=0\}}) = \sum_{s=1}^{+\infty} \mathbb{E}(X_s 1_{\{N[1,s)=0\}} 1_{\{N(0)=1\}})$$

By joint stationarity of X_t and N:

$$\begin{aligned} RHS &= \sum_{s=1}^{+\infty} \mathbb{E}(X_0 1_{\{N[1-s,0)=0\}} 1_{\{N(-s)=1\}}) \\ &= \sum_{s=1}^{+\infty} \mathbb{E}(X_0 1_{\{T^-(-1)=-s\}}) = \mathbb{E}\left(\sum_{s=1}^{+\infty} X_0 1_{\{T^-(-1)=-s\}}\right) = \mathbb{E}(X_0) \end{aligned}$$

This shows the first formula. The proof for the second formula, $\mathbb{E}(X_0) = \lambda \mathbb{E}^0(\sum_{s=0}^{T_1-1})$ is similar.

□

REMARK. There are many variants of the inversion formula. For example, one can similarly show that

$$\mathbb{E}(X_0) = \lambda \mathbb{E}^0\left(\sum_{s \in V_0} X_s\right)$$

where V_n (*Voronoi cell*) is the set of times that are closest to T_n, with rounding by excess:

$$V_n = \left(\frac{T_{n-1}+T_n}{2}, \frac{T_n+T_{n+1}}{2}\right]$$

QUESTION 11.3.8. *What does Palm's formula give for $X_t = T^+(t)$?* [9]

11.3.7 APPLICATION TO INTENSITY.

Apply the inversion formula to $X_t = 1$ and obtain

PROPOSITION 11.3.2.

For a stationary point process with intensity λ:

$$\frac{1}{\lambda} = \mathbb{E}^0(T_1 - T_0) = \mathbb{E}^0(T_1)$$

This formula is well known for a Poisson process, but we now know that it is true for *any* stationary point process.

11.3.8 APPLICATION: RESIDUAL LIFETIME AND FELLER'S PARADOX

Consider a stationary point process T_n, with intensity λ. Let $X_t = T^+(t) - t$ be the time from t until the next point, $Y_t = t - T^-(t)$ the time since the last point, and $Z_t = X_t + Y_t$ the interval seen at a random instant.

QUESTION 11.3.9. *What is X_0 ? Y_0 ? Z_0 ?* [10]

[9] Nothing, because X_t, T_n is not jointly stationary.

[10] $X_0 = T_1$ and $Y_0 = -T_0$, $Z_0 = T_1 - T_0$.

Note that, with our convention, $X_t > 0$ and $Y_t \geq 0$.

THEOREM 11.3.2. *For any t, the distributions of X_t and Y_t have densities given by*

$$\begin{cases} f_X(s) = \lambda \mathbb{P}^0(T_1 > s) \\ f_Y(s) = \lambda \mathbb{P}^0(T_1 \geq s) \end{cases}$$

The distribution of Z_t is given by

$$dF_Z(s) = \lambda s dF_T(s)$$

where F_T is the Palm distribution of $T_1 - T_0$.

The Palm probability $\mathbb{P}^0(T_1 \geq s)$ is the complementary distribution of the time between points. In discrete time, the theorem means that $\mathbb{P}(X_t = s) = \lambda \mathbb{P}^0(T_1 > s)$, $\mathbb{P}(Y_t = s) = \lambda \mathbb{P}^0(T_1 \geq s)$ and $\mathbb{P}(Z_t = s) = \lambda s \mathbb{P}^0(T_1 = s)$.

In continuous time, if the Palm distribution of $T_1 - T_0$ has a density f_T, (i.e. $dF_T(s) = f_T(s)ds$) then X_t and Y_t both have a density equal to

$$f_X(s) = f_Y(s) = \lambda \int_s^{+\infty} f_T(u)du$$

and Z_t has density

$$f_Z(s) = \lambda s f_T(s)$$

Proof. (discrete time) X_t is jointly stationary with T_n, thus its distribution is independent of t, and we can apply the inversion formula. For any $s \geq 0$ we have

$$\mathbb{P}(X_0 = s) = \mathbb{E}(1_{\{X_0 = s\}}) = \lambda \mathbb{E}^0 \left(\sum_{u=0}^{T_1 - 1} 1_{\{X_u = s\}} \right)$$

Given that there is a point at 0 and $0 \leq u \leq T_1 - 1$, we have $X_u = T_1 - u$, thus

$$\mathbb{P}(X_0 = s) = \lambda \mathbb{E}^0 \left(\sum_{u=0}^{T_1 - 1} 1_{\{T_1 = u + s\}} \right)$$

Now the sum in the formula is 1 if $T_1 > s$ and 0 otherwise. Thus

$$\mathbb{P}(X_0 = \tau) = \lambda \mathbb{E}^0 \left(1_{\{T_1 > s\}} \right) = \lambda \mathbb{P}^0(T_1 > s)$$

which shows the formula for X_t. The formula for Y_t is similar, using $Y_u = u$ for $0 \leq u \leq T_1 - 1$.

For Z_t, apply the inversion formula and obtain

$$\mathbb{P}(Z_0 = s) = \lambda \mathbb{E}^0 \left(\sum_{u=0}^{T_1 - 1} 1_{\{Z_u = s\}} \right)$$

Now under P^0, $Z_u = T_1$ does not depend on u for $0 \leq u \leq T_1 - 1$ thus

$$\mathbb{P}(Z_0 = s) = \lambda \mathbb{E}^0 \left(1_{\{T_1 = s\}} \sum_{u=0}^{T_1 - 1} 1 \right) = \lambda \mathbb{E}^0 \left(T_1 1_{\{T_1 = s\}} \right) = \lambda s \mathbb{P}^0(T_1 = s)$$

□

EXAMPLE: POISSON PROCESS.We have $f_T(t) = \lambda e^{-\lambda s}$ and $\mathbb{P}^0(T_1 > s) = \mathbb{P}^0(T_1 \geq s) = e^{-\lambda s}$ thus $f_X(s) = f_Y(s) = f_T(s)$, as expected: the time until the next arrival, or since the last arrival, has the same distribution as the time between arrivals. The distribution of Z_t has density

$$f_T(s) = \lambda^2 s e^{-\lambda s}$$

i.e., it is an Erlang-2 distribution.

COROLLARY 11.3.1 (Mean Residual Times and Mean Interval). *With the notation of Theorem 11.3.2:*

$$\mathbb{E}(Z_t) = \lambda \mathbb{E}^0(T_1^2))$$

Further, in discrete time

$$\begin{cases} \mathbb{E}(X_t) = \frac{\lambda}{2}\mathbb{E}^0(T_1(T_1+1)) \\ \mathbb{E}(Y_t) = \frac{\lambda}{2}\mathbb{E}^0(T_1(T_1-1))) \end{cases}$$

and in continuous time

$$\mathbb{E}(X_t) = \mathbb{E}(Y_t) = \frac{\lambda}{2}\mathbb{E}^0(T_1^2)$$

Proof. Apply Theorem 11.3.2 or apply the inversion formula directly.

□

EXAMPLE: BUSES AT SAINT-FRANÇOIS. The average time until next bus, seen by you and me, is

$$\mathbb{E}(X_t) = \frac{1}{2}\left(\frac{1}{\lambda} + \lambda \text{var}^0(T_1 - T_0)\right)$$

where $\text{var}^0(T_1 - T_0)$ is the variance, under Palm, of the time between buses. It is the variance estimated by the inspector. The expectation $\mathbb{E}(X_t)$ is minimum, equal to $\frac{1}{2\lambda}$ when the buses are absolutely regular ($T_1 - T_0$ is constant). If the interval between buses $T_1 - T_0$ seen by the inspector is heavy tailed, then $\mathbb{E}(X_t)$ is infinite. Thus, when the inspector should report not only the mean time between buses, but also its variance.

QUESTION 11.3.10. *For the video server example in Section 11.1.1, verify the value found for the expected time from an arbitrary instant to the next film, by applying the corollary.* [11]

QUESTION 11.3.11. *If T_n is a Poisson process, what is $\mathbb{E}(T^-(0))$?* [12]

[11] Time unit is a slot of $5mn$: $\mathbb{E}^0(T_1^2) = \frac{1}{3}(1^2 + 3^2 + 8^2) = 74/3$; $\lambda = 1/4$; $\mathbb{E}(X_t) = \lambda/2\mathbb{E}^0(T_1^2) = 74/24$ time units $= 15mn25s$ as found earlier.

[12] $\frac{1}{\lambda}$

FELLER'S PARADOX. Apply now the corollary to Z_t, the duration of the interval, seen at an arbitrary instant. We find

$$\mathbb{E}(Z_t) = \frac{1}{\lambda} + \lambda \text{var}^0(T_1 - T_0)$$

thus, except for constant bus interarrival times, the mean interval, seen at an arbitrary instant, is always larger than the mean interval between buses $\mathbb{E}^0(T_1 - T_0) = \frac{1}{\lambda}$, measured by the inspector ! This is called Feller's paradox, and, as we have shown, it holds for *any* stationary point process (in particular, whatever the correlation between successive intervals is). An intuitive explanation is that if we pick a random time interval, we are more likely to fall in a large one.

QUESTION 11.3.12. *For the video server example in Section 11.1.1, what is (1) the Palm expectation of the time between films (2) the expected time between films measured from an arbitrary instant ?* [13]

QUESTION 11.3.13. *Is it fair to say that the average waiting time is the average interval between evants, divided by 2 ?* [14]

QUESTION 11.3.14. *Answer the question asked in Chapter* **??** *about the video server example: "Is it fair to say that the average waiting time is* $60mn/3/2 = 10mn$ *"?* [15]

QUESTION 11.3.15. *Does Feller's paradox apply to a Poisson process ?* [16]

QUESTION 11.3.16. *How do you interpret the difference between the discrete time and continuous time results in Corollary 11.3.1 ?* [17]

QUESTION 11.3.17. *Prove the statements for the renewal source model in Section 11.3.3.* [18]

11.3.9 WHEN CAN A SEQUENCE OF TIME INSTANTS BE CONSIDERED A STATIONARY POINT PROCESS ?

In some cases, such as the random waypoint model in Example 3.5 on page 68, we are given the state of the system at transition instants. In this section we examine whether it is formally possible to build a point process for which the states sampled at transition instants correspond to the Palm viewpoint.

To understand why this is an issue, consider the following example.

EXAMPLE 11.8: RANDOM WAYPOINT. The random waypoint model is defined in Example 3.5 on page 68. The intensity of the process of transitions (or "waypoints") is given by the Palm inversion formula:

$$\lambda^{-1} = \mathbb{E}^0(T_1) = \mathbb{E}^0(\frac{D_1}{V_0})$$

---

[13] (1) $20mn$ (2) $\mathbb{E}(Z_t) = 2\mathbb{E}(X_t) = 30mn50s$.

[14] Yes, if averages are at arbitrary instants, since (at least in continuous time) $\mathbb{E}(X_t) = \frac{1}{2}\mathbb{E}(W_t)$.

[15] No, this performance metric does not represent customer waiting times. It is fine to divide by 2 for expectations at arbitrary time points, but it does not represent customer waiting time if we apply it to Palm averages.

[16] For a Poisson process, $\mathbb{E}(W_t) = 2/\lambda = 2\mathbb{E}^0(T_1)$, so the answer is yes. The average at a random instant is twice as large as seen at a random point of the Poisson process.

[17] An observer arriving at an arbitrary discrete time instant samples the system slightly differently than one arriving at an arbitrary continuous time instant. If the time unit is very small, the difference can be neglected.

[18] Direct application of Corollary 11.3.1, in continuous time, taking $dF_T =$ Dirac mass at u.

with $D_1 := \|M_1 - M_0\|$. By construction, D_1 and V_0 are independent. Thus

$$\lambda^{-1} = \mathbb{E}^0(D_1)\mathbb{E}^0(\frac{1}{V_0})$$

It is finite if and only if $\mathbb{E}^0(\frac{1}{V_0})$ is finite, which means $v_{\min} > 0$.

If $v_{\min} = 0$, the formula would give $\lambda = 0$, which means that there is a problem.

In the case $v_{\min} = 0$, Palm calculus does not apply, i.e. we cannot consider the transitions as the transition instants of a stationary process. A simulation study shows that in fact, the system "freezes": as you run the simulation longer and longer, it becomes more likely to draw a very small speed V_n. When such a small speed is drawn, the system stays with a very long time at this speed. In contrast, for $v_{\min} > 0$, it does. We give in this section a theorem that states such a result.

THEOREM 11.3.3. *Consider a sequence of wide sense increasing, random, times $T_0 = 0 \leq T_1 \leq T_2 \leq ...$, and of random variables $Y_0, Y_1, ...$ such that*

C1 *$(T_n - T_{n-1}, Y_n)$ is stationary with respect to the index n.*

Then T_n can be considered as the points of a stationary, marked point process, observed conditional to the event "there is an arrival at time 0" if and only if

C2 $\mathbb{E}^0(T_1) < \infty$
C3 $\mathbb{P}^0(T_1 > 0) = 1$

Further, define the process Z_t by $Z_t = Y_n$ with n such that $T_n \leq t < T_{n+1}$. Then T_n, Z_t are jointly stationary.

The proof is complex – see [Baccelli88-book]. In continuous time, there is an additional condition:

C4 $E^0(N(0,t)) < \infty$ for all $t \geq 0$.

REMARK. A process Z_t such that

- T_n, Z_t are jointly stationary
- Z_t is constant on intervals of the form $[T_n, T_{n+1})$

is called a *mark* of the point process.

EXAMPLE 11.9: RENEWAL SOURCE MODEL (I.I.D INTERARRIVAL TIMES). Let $T_0 = 0, T_1 = U_1, ..., T_n = U_1 + U_2 + ... + U_n$ where U_n is iid ≥ 0. Under which conditions can we consider these points as the realization of a stationary point process with a point at time $t = 0$?

The answer is: the inter-arrival time S_1 is not identically zero (with probability 1) and has finite mean.

Proof.

Here condition C1 is obviously true. We now apply the remaining conditions in the theorem.

Conditions C2 and C3 mean that the inter-arrival time S_1 is not identically zero (with probability 1) and and has finite mean.

Condition C4 is always true if U_1 is not identically 0. We prove this now.

First note that

$$E^0(N(0,t)) = \sum_{k\geq 1} k\mathbb{P}^0(N(0,t)=k) = \sum_{k\geq 1} \mathbb{P}^0(N(0,t)\geq k) = \sum_{k\geq 1} \mathbb{P}^0(T_k \leq t) \tag{11.1}$$

Pick some arbitrary, fixed $s > 0$; by Markov's inequality:

$$\mathbb{P}^0(T_n \leq t) \leq e^{st}\mathbb{E}^0\left(e^{-sT_n}\right)$$

Now $\mathbb{E}^0\left(e^{-sT_n}\right) = \mathbb{E}\left(e^{-s(U_1+\ldots+U_n)}\right)$ is the Laplace-Transform of the convolution of n independent random variables. Thus

$$\mathbb{P}^0(T_k \leq t) \leq e^{st}G^k(s)$$

where $G(s) := \mathbb{E}\left(e^{-sU_1}\right)$ is the Laplace-Transform of U_1. We have $G(s) = 1$ if and only if $sU_1 = 0$ with probability 1. Thus, by hypothesis, $G(s) < 1$ since $s > 0$. By Equation (11.1):

$$E^0(N(0,t)) \leq e^{st}\sum_{k\geq 1} G^k(s) < \infty$$

□

EXAMPLE 11.10: RANDOM WAYPOINT. The random waypoint model is defined in Example 3.5 on page 68. We apply the theorem to the sequence of times T_n and marks (M_n, V_n). We obtain that the random waypoint is stationary if and only if $v_{\min} > 0$.

Proof. Condition C1 is true by construction since speed and next position at transition instants are iid.

Condition C2 follows from :

$$\mathbb{E}^0(T_1) = \mathbb{E}^0(D_1)\mathbb{E}^0(\frac{1}{V_0})$$

It is bounded if and only if $\mathbb{E}^0(\frac{1}{V_0}$ is finite, which means here $v_{\min} > 0$.

Condition C3 is obviously true.

Condition C4 : The inter-transition times $S_n = T_n - T_{n-1}$ are not all independent, but S_m and S_n are independent if $n - m \geq 2$. The rest of the proof is similar to the proof in Example 11.9 on page 294

□

11.4 A MENAGERIE OF PALM CALCULUS FORMULAE.

We give here a few useful theorems which hold in discrete and continuous times. We give the proofs in discrete time only.

11.4.1 CAMPBELL'S FORMULA

THEOREM 11.4.1 (Campbell's Formula). *let T_n be a stationary point process with intensity λ and $F(t)$ a bounded, random (not necessarily stationary) process.*

$$\mathbb{E}\left(\sum_{n\in\mathbb{Z}} F(T_n)\right) = \lambda \sum_{t\in\mathbb{Z}} \mathbb{E}^t(F(t))$$

where E^t is the conditional expectation, given that there is a point at t.

Proof. The left handside of the equation in the theorem is

$$\mathbb{E}\left(\sum_{t\in\mathbb{Z}} F(t)1_{\{N\{t\}=1\}}\right) = \sum_{t\in\mathbb{Z}} \mathbb{E}\left(F(t)1_{\{N\{t\}=1\}}\right) = \sum_{t\in\mathbb{Z}} \mathbb{E}^t(F(t))\mathbb{P}(N\{t\}=1) = \lambda \sum_{t\in\mathbb{Z}} \mathbb{E}^t(F(t))$$

□

In continuous time, E^t is not strictly speaking a conditional expectation, and a rigorous statement requires a complex formalism, which we do not develop here. Assuming such a formalism, Campbell's formula is

$$\mathbb{E}\left(\sum_{n\in\mathbb{Z}} F(T_n)\right) = \lambda \int_{t\in\mathbb{R}} \mathbb{E}^t(F(t))dt \tag{11.2}$$

SPECIAL CASE If $F(t) = f(t)$ is non-random, Campbell's formula gives, in discrete time

$$\mathbb{E}\left(\sum_{n\in\mathbb{Z}} f(T_n)\right) = \lambda \sum_{t\in\mathbb{Z}} f(t) \tag{11.3}$$

and in continuous time

$$\mathbb{E}\left(\sum_{n\in\mathbb{Z}} f(T_n)\right) = \lambda \int_{t\in\mathbb{R}} f(t)dt$$

EXAMPLE 11.11: SHOT NOISE WITH DETERMINISTIC SHOTS. A *shot noise* process is defined by $X(t) = \sum_{n\in\mathbb{Z}} h(t - T_n)$, where T_n is a stationary point process ("shot epochs") and $h(t)$ a function such that $h(t) = 0$ for $t < 0$. Shot noise is used to model traffic in a backbone network in [Barakat02-infocom], where T_n represents the beginning of sessions and $h(t)$ the bit rate generated by one session that would start at time 0. We say here that the shots are deterministic to express that $h(t)$ is non-random.

A direct application of Campbell's formula to $f(t) = h(\tau - t)$ gives, for any time τ:

$$\mathbb{E}(X(\tau)) = \mathbb{E}\left(\sum_{n\in\mathbb{Z}} f(T_n)\right) = \lambda \sum_{t\in\mathbb{Z}} h(t-\tau)$$

thus

$$\mathbb{E}(X(\tau)) = \mathbb{E}(X(0)) = \lambda \sum_{t\in\mathbb{Z}} h(t) \tag{11.4}$$

which is also known under the name of Campbell formula. In continuous time, we have

$$\mathbb{E}(X(\tau)) = \mathbb{E}(X(0)) = \lambda \int_0^{+\infty} h(t)dt$$

APPLICATION: SHOT NOISE A general *shot noise* process is defined by $X(t) = \sum_{n\in\mathbb{Z}} h(t - T_n, Z_{T_n})$, where T_n is a stationary point process ("shot epochs"), Z_t a mark, and $h(t,z)$ a function such that $h(t,z) = 0$ for $t < 0$. For the example of internet traffic, T_n is the beginning of a session, Z_{T_n} is the random parameter chosen for session n and $h(t,z)$ the bit rate generated by one session that would start at time 0, and has parameter z. A direct application of Campbell's formula to $F(t) = h(-t, Z_t)$ gives

$$\mathbb{E}(X(0)) = \lambda \sum_{t\in\mathbb{Z}} \mathbb{E}^t(h(-t, Z_t)) = \lambda \sum_{t\in\mathbb{Z}} \mathbb{E}^t(h(-T^-(-t), Z_t))$$

now $t - T^-(t), T_n$ is jointly stationary. Thus, by Proposition 11.3.1

$$\mathbb{E}^t(h(-T^-(t), Z_t) = \mathbb{E}^t(h(t - T^-(t) - t, Z_t) = \mathbb{E}^0(h(-t, Z_0))$$

and

$$\mathbb{E}(X(0)) = \lambda \sum_{t\in\mathbb{Z}} \mathbb{E}^0(h(-t, Z_0)) = \lambda \mathbb{E}^0\left(\sum_{t\in\mathbb{Z}} h(-t, Z_0)\right)$$

which can be re-written as

$$\mathbb{E}(X(0)) = \lambda \sum_{t\in\mathbb{N}} \mathbb{E}^0(H(t)) \tag{11.5}$$

where $H(t) := h(t, Z_0)$ is the value at time t of the random function chosen for a typical shot. There is an equivalent formula in continuous time:

$$\mathbb{E}(X(0)) = \lambda \int_0^{+\infty} \mathbb{E}^0(H(t))dt$$

Equation (11.5) is also known as a Campbell formula for shot noise. Note that the stationarity assumption implies that $X(t)$ is stationary and thus $\mathbb{E}(X(\tau)) = \mathbb{E}(X(0))$ for any time τ. Compare to Equation (11.4): the expected value of the shot noise is the same as if we replace the random shot $H(t)$ by its expected value. We have shown that this holds quite generally, whether the shot epochs are a Poisson process or not.

11.4.2 LITTLE'S FORMULA.

THEOREM 11.4.2 (Little). *Consider a sequence T_n, Y_n stationary with respect to index n. Assume that T_n can be viewed as a stationary point process, with intensity λ (see Theorem 11.3.3). Let $X(t) = \sum_{n\in\mathbb{Z}} 1_{\{T_n\le t<Y_n+T_n\}}$ and define Z_t by $Z_t = Y_n$ if and only if $T_n < leqt < T_{n+1}$. We interpret T_n as customer arrival times, Y_n as the residence time of the nth customer, Z_t as the residence time of the last customer who arrived before or at t and $X(t)$ as the number of customers present in the system at time t. Then for any t*

$$\mathbb{E}(X(0)) = \lambda\mathbb{E}^0(Z_0)$$

The theorem relates the average number of customers at an arbitrary instant to the average residence time seen by an arbitrary customer.

Proof. Apply Campbell's formula, with $F(t) = 1_{\{t\le 0<Z_t+t\}}$. The left-handside in Campbell's formula is $\mathbb{E}(X(0))$.

Let us compute the right-handside. If $t > 0$ then $F(t) = 0$. Else $\mathbb{E}^t(F(t)) = \mathbb{P}^t(Z_t > -t)$. Now by joint stationarity, $\mathbb{P}^t(Z_t > u) = \mathbb{P}^0(Z_0 > u)$ for any $u \ge 0$. Thus $\mathbb{E}^t(F(t)) = \mathbb{P}^0(Z_0 > -t)$ and the right handside is

$$\lambda\sum_{t\le 0}\mathbb{P}^0(Z_0 > -t) = \lambda\sum_{u\ge 0}\mathbb{P}^0(Z_0 > u) = \lambda\mathbb{E}^0(Z_0)$$

□

Note that, by stationarity, we also have $\mathbb{E}(X(t)) = \lambda\mathbb{E}^t(Z_t)$ for any t.

11.4.3 NEVEU'S EXCHANGE FORMULAE

THEOREM 11.4.3 (Exchange Formula). *Consider two jointly stationary point processes T_n and T'_n, with counting time series N and N', and intensities λ, λ'. Let X_t be jointly stationary with them. This means that the joint process $N(t), N'(t), X_t$ is strictly stationary. Call $\mathbb{E}^0_N$ [resp. $\mathbb{E}^0_{N'}$] the Palm expectation with respect to the first [resp. second] point process. Time is either discrete or continuous. Then:*

$$\lambda\mathbb{E}^0_N(X_0) = \lambda'\mathbb{E}^0_{N'}\left(\sum_{m\in\mathbb{Z}} X_{T_m}1_{\{0<T_m\le T'_1\}}\right) = \lambda'\mathbb{E}^0_{N'}\left(\sum_{m\in\mathbb{Z}} X_{T_m}1_{\{0\le T_m<T'_1\}}\right)$$

Proof. (discrete time): Apply the inversion formula to $X_sN(s)$ and note that $\sum_{m\in\mathbb{Z}} X_{T_m}1_{\{0<T_m\le T'_1\}} = \sum_{s=1}^{T'_1} X_sN(s)$.

□

COROLLARY 11.4.1.

$$\lambda = \lambda'\mathbb{E}^0_{N'}\left(N(0,T'_1]\right) = \lambda'\mathbb{E}^0_{N'}\left(N[0,T'_1)\right)$$

Proof. apply the theorem with $X_s = 1$

□

COROLLARY 11.4.2 (*Wald's Identity*).

$$\mathbb{E}_N^0(X_0) = \frac{\mathbb{E}_{N'}^0\left(\sum_{m\in\mathbb{Z}} X_{T_m} 1_{\{0<T_m\leq T_1'\}}\right)}{\mathbb{E}_{N'}^0\left(N(0,T_1']\right)}$$

Proof. First Corollary 11.4.1, then apply Theorem 11.4.3

□

REMARKS. The exchange formulae do not make any assumption about the joint behaviour of the two point processes, other than stationarity. Wald's identity is often shown under restrictive assumptions (for example, when X_s is an iid sequence), but we have shown here that it is generally true.

The exchange formula can also be applied to Voronoi cells.

APPLICATION: THE STOP AND GO PROTOCOL. We re-visit the computation of the stop and go protocol given in Section 11.3.5. Apply the corollary with the first process equal to arrivals of fresh packets, and second process equal to all retransmission attempts. Thus the second process contains all points of the first process, and more. We have, by Theorem 11.4.3 with $X_s = 1$:

$$\lambda = \lambda' \mathbb{E}_{N'}^0\left(N(0,T'1]\right) = \lambda'(1-\alpha)$$

where the last equality is by definition of α. We compute λ' from Proposition 11.3.2:

$$\frac{1}{\lambda'} = (1-\alpha)S + \alpha S_1$$

Combining the two gives $\frac{1}{\lambda} = S + \frac{\alpha}{1-\alpha}S_1$ as already found.

11.4.4 MATTHES' DIRECT FORMULA

The direct formula gives an intuitive interpretation of Palm probability:

THEOREM 11.4.4 (Direct Formula). *If T_n, X_t is jointly stationary, then for any interval I:*

$$\mathbb{E}^0(X_0) = \frac{1}{\lambda|I|}\mathbb{E}\left(\sum_{n \text{ such that } T_n\in I} X_{T_n}\right)$$

Proof. Apply Campbell's formula to $F(t) = X_{T^-(t)}1_{\{T^-(t)\in I\}}$:

$$\mathbb{E}\left(\sum_{n \text{ such that } T_n\in I} X_{T_n}\right) = \mathbb{E}(\sum_n F(T_n)) = \lambda \sum_{t\in\mathbb{Z}} \mathbb{E}^t\left(X_{T^-(t)}1_{\{T^-(t)\in I\}}\right)$$

Now, by stationarity

$$\mathbb{E}^t(X_{T^-(t)}1_{\{T^-(t)\in I\}}) = \mathbb{E}^0(X_{T^-(0)}1_{\{T^-(0)\in I\}}) = \mathbb{E}^0(X_0 1_{\{0\in I-t\}}) = 1_{\{t\in I\}}\mathbb{E}^0(X_0)$$

where $I-t$ is the set $\{i-t, i\in I\}$. Thus

$$\mathbb{E}\left(\sum_{n \text{ such that } T_n\in I} X_{T_n}\right) = \lambda\sum_{t\in\mathbb{Z}} 1_{\{t\in I\}}\mathbb{E}^0(X_0) = \lambda|I|\mathbb{E}^0(X_0)$$

□

EXAMPLE 11.12: ELAPSED AND RUNNING TIMES. Consider some stationary point process T_n and let $Z_t = T^+(t) - T^-(t)$. (Z_t, T_n) is jointly stationary and thus we can apply the direct formula. Let us take $I = [0,t]$. On one side we have:

$$\mathbb{E}^0(Z_0) = \mathbb{E}^0(T_1) = \frac{1}{\lambda}$$

On the other side, we find

$$\frac{1}{\lambda t}\mathbb{E}\left(\sum_{n:0\le T_n\le t} Z_{T_n}\right) = \frac{1}{\lambda t}\mathbb{E}\left(\sum_{n:0\le T_n\le t} T_{n+1}-T_n\right) = \frac{1}{\lambda t}\mathbb{E}\left(T^+(t)-T^-(0)\right)$$

Thus

$$\mathbb{E}(T^+(t)) - t = \mathbb{E}(T^-(0)) = \mathbb{E}(t - T^-(t))$$

which expresses that, in average, the time from an arbitrary instant to or from the next point are equal.

11.5 CASE STUDY: THROUGHPUT OF TCP

TCP is considered to be the reference protocol in the Internet, and any session should have a throughput not exceeding that of TCP. Therefore, there is interesting in understanding the performance of TCP. TCP controls its sending rate by increasing it when there is no congestion,a nd reducing it when it receives a congestion signal. A congestion signal is a packet loss in the current internet, or a bit in an acknowledgement packet in the future. The following paper relates the throughput of a TCP connections to the characteristics of the loss process.

Read [Altman00-Sigcomm] and answer the following questions.

QUESTION 11.5.1. *What is the performance metric used for a TCP connection ?* [19]

QUESTION 11.5.2. *What is the equation describing the evolution of the rate of the TCP connection ?* [20]

---

[19] The throughput $X(t)$, assumed to be represented by a stationary process.

[20]

$$X_{n+1} = \nu X_n + \alpha S_n$$

where ν is the decrease factor and α the linear increase term.

QUESTION 11.5.3. *What are the values of α and ν?* [21]

QUESTION 11.5.4. *What is the stationary point process used in the paper ? What is its intensity called ?* [22]

QUESTION 11.5.5. *In our framework, how would we write $\mathbb{E}(X_n^*))$ and $\mathbb{E}((X_n^*)^2$?* [23]

QUESTION 11.5.6. *How are $\mathbb{E}(X_n^*))$ and $\mathbb{E}^0(X(0)^2)$ computed ?* [24]

QUESTION 11.5.7. *How is the throughput related to X_n ?* [25]

QUESTION 11.5.8. *Which jointly stationary process in the inversion formula applied to ?* [26]

QUESTION 11.5.9. *What is the difference between loss event rate and the intensity of the loss process ?* [27]

QUESTION 11.5.10. *What is the loss-throughput formula obtained in the paper ?* [28]

QUESTION 11.5.11. *What is the "famous square root formula" ? Under which assumptions is it valid ?* [29]

QUESTION 11.5.12. *What is the conservative timeout model ?* [30]

QUESTION 11.5.13. *How do the authors derive the throughput formula for the conservative timeout model ?* [31]

---

[21] $\alpha = 1/(bRTT^2)$. In fact, $b = 1/MSS^2$ (maximum TCP segment size). $\nu = 0.5$.

[22] The sequence of loss events. It is not necessarily stationary, but the authors show that it converges to a stationary point process and place themselves at the limit. $\lambda = 1/\mathbb{E}^0(S_1)$.

[23] These are Palm probabilities. We would write them $\mathbb{E}^0(X(0))$ and $\mathbb{E}^0(X(0)^2)$.

[24] First, an EWMA representation of the stationary sequence X_n^* is given. Second, this is used to derive the mean and second moment, using a direct computation.

[25] By the inversion formula.

[26] $X(t)$.

[27] The loss event rate, p, is defined by an ergodic interpretation as the long term average of the number of losses per data unit sent. It is equal to $\frac{\lambda}{\mathbb{E}(X(0))}$.

[28]

$$RTT\sqrt{p}\sqrt{\frac{1+\nu}{2(1-\nu)} + \frac{1}{2}\hat{C}(0) + \sum_{k=1}^{\infty}\nu^k\hat{C}(k)}$$

where $\hat{C}(k)$ is the normalized auto-covariance function (covariance/square of mean), under Palm, of S_n.

[29] It is the formula obtained for a deterministic S_n.

[30] A more accurate model that accounts for the fact that loss events are of two types: with timeout (TO) or without (TD). With TO, the dynamic of the system is slightly different. During some period, in average equal to Z, the rate is set to 0

[31] First, they derive the throughput $\bar{X}'$ of the virtual system where the idle periods are deleted. This is the same as in the original case. Second, they use an ergodic interpretation to show that the real throughput is related by

$$\bar{X} = (1 - \lambda_{TO}Z)\bar{X}'$$

It is in fact a thinning formula and could be obtained by reasoning with conditional probabilities.

QUESTION 11.5.14. *What is the model with transmission rate limitation ? How is it solved ?* [32]

QUESTION 11.5.15. *How is the model validated ?* [33]

QUESTION 11.5.16. *How do the authors compare their model to competing ones ?* [34]

11.5.1 MODULATED MODELS.

A powerful, generic family of models can be built using modulators, as follows. Consider a stationary point process T_n with a mark Z_t that takes a finite number of values $1, ..., I$, called "states". Given a function y_i of the state i, we are interested in the process $Y_t = y_{Z_t}$. We say that Z_t is a *modulating process* and Y_t is a modulated process.

There is a discrete time, finite space, ergodic Markov chain X_n. At step n we draw a random number S_n according to a distribution F_i, with $i = X_n$, independently of all past, given that we are in state i. A continuous time process Z_t, called the modulator, stays in state i for a duration equal to S_n. Call $T_n = S_1 + ... + S_n$. We have thus $Z_t = X_n$ iff $T_n \leq t < T_{n+1}$. We assume that the system is stationary, thus T_n is a stationary point process, and Z_t is a mark.

PROPOSITION 11.5.1. *Let π_i^0 be the probability that the modulator is in state i at an arbitrary transition and $\bar{T}_i = \mathbb{E}^0(T_1 - T_0 | Z_0 = i)$ the expected duration of an inter-transition time, starting from state i. We have*

$$\begin{cases} \lambda = \sum_i \pi_i^0 \bar{T}_i \\ \mathbb{E}(Y_t) = \lambda \sum_i \pi_i^0 y_i \bar{T}_i \end{cases}$$

Proof. Apply the inversion formula to 1 then to Y_t

□

[32] In many cases, the rate $X(t)$ cannot grow indefinitely, but is limited to a maximum M. This changes the dynamics of the system to

$$X_{n+1} = M \wedge (\nu X_n + \alpha S_n)$$

The new system is harder to study. The authors take $\nu = 1/2$ use a bounding technique and min-plus algebra, they re-write the dynamics as

$$X_{n+1} = \left[(M - \frac{1}{2} X_n) \wedge (\alpha S_n) \right] + \frac{1}{2} X_n$$

which can be used to show that an upper bound is $\hat{X}_n$ with

$$\hat{X}_{n+1} = [(M \wedge (\alpha S_n)] + \frac{1}{2} \hat{X}_n$$

and a lower bound is $\check{X}_n$ with

$$\check{X}_{n+1} = \left[\frac{M}{2} \wedge (\alpha S_n) \right] + \frac{1}{2} \check{X}_n$$

Then the same method is used to obtain the palm expectations and the throughput.

[33] By a limited number of measurements.

[34] Their model makes no specific assumption on the loss process. Competing works assume the loss process is either deterministic or Poisson. The authors find that this introduces some errors, but in some cases the errors are largely cancelled by inaccuracies in the model (such as absence of TO or rate limitation effects).

EXAMPLE 11.13: LOSS CHANNEL MODEL. A path on the internet is often model as a loss system, where the packet loss ratio P_t depends on the state of a hidden modulator. Assume that when the modulator Z_t is in state i, the loss ratio is p_i. We have $P_t = p_{Z_t}$.

We find that the time average loss rate is

$$\bar{p} = \frac{\sum_i \pi_i^0 p_i \bar{T}_i}{\sum_i \pi_i^0 \bar{T}_i}$$

See exercise 11.7 for an application to the Internet.

QUESTION 11.5.17. *What is* $\mathbb{P}(Z_t = i)$ *?* [35]

REMARK. MARKOV MODULATED PROCESS. A special case is when Z_t is a markov process. The modulated process Y_t is then called a markov modulated process. Note that Proposition 11.5.1 does not assume any Markov property.

11.6 APPLICATION TO MARKOV MODELING

For a quick review of Markov chains, see Section 11.8.

to be completed– add Erlang

11.6.1 EMBEDDED SUB-CHAIN

If we observe a Markov chain at selected transitions, we obtain an *embedded sub-chain*. We explain in this section how to compute all elements of the embedded subchain.

Consider first a discrete time chain. Let C be a matrix such that $A - C$ is wide-sense positive. We consider that C defines a process of *selected transition*, as follows. Whenever a transition i, j of the markov chain occurs, we draw a random number, independent of all past, and with with probability $Ci, j/Q_{i,j}$ decide that the transition is "selected". To gain some intuition, consider the simple case where $C_{i,j} = A_{i,j}$ or 0. Define $F = \{(i, j) \in E^2 : C_{i,j} = Q_{i,j}\}$; a transition is selected if it is in F. In continuous time, the definition is the same with A in lieu of Q.

Call T_n the point process of selected transitions. Then X_{T_n} is itself a markov chain, since the knowledge of the state at the nth transition is sufficient to compute the probabilities of future events (this is the strong markov property). The sequence $Y_n = X_{T_n}$ is called the *embedded sub-chain* and we say that C is the *matrix of selected transitions*.

THEOREM 11.6.1 (LeBoudec84-diss). *Consider an ergodic, stationary markov chain* X_t, $t \in \mathbb{Z}$, *with stationary probability* π. *Consider an embedded sub-chain* Y_n *with matrix of selected transitions* $C_{i,j}$.

[35] Apply the proposition to $y_j = 1_{\{i=j\}}$ and find $\mathbb{P}(Z_t = i) = \lambda \pi_i^0 \bar{T}_i$.

1. *The transition matrix J of the embedded sub-chain Y_n satisfies $(Id - Q + C)J = C$ (discrete time) or $(C - A)J = C$ (continuous time).*
2. *The intensity of the point process of selected transitions is $\eta = \sum_{i,j} \pi_i C_{i,j}$*
3. *The probability that a selected transition is (i, j) is $\mathbb{P}^0(X_{-1} = i, X_0 = j) = \frac{1}{\eta}\pi_i C(i, j)$.*
4. *The probability to be in state j just after a selected transition is $\pi_j^0 := \mathbb{P}^0(X_0 = j) = \frac{1}{\eta}\sum_i \pi_i C(i, j)$. The probability to be in state i just before a selected transition is $\mathbb{P}^0(X_{-1} = i) = \frac{1}{\eta}\pi_i \sum_j C(i, j)$.*

Proof. By the strong markov property:

$$J_{i,j} = \mathbb{P}^0(X_{T_1} = j | X_{T_0} = i) = \mathbb{P}(X_{T^+(0)} = j | X_0 = i)$$

Condition with respect to the next transition, selected or not:

$$J_{i,j} = \sum_{k:(i,k)\in F} Q_{i,k} + \sum_{k:(i,k)\notin F} Q_{i,k}\mathbb{P}(X_{T^+(0)} = j | X_1 = k \text{ and } X_0 = i)$$

Now, for $(i, k) \notin F$, given that $X_0 = i, X_1 = k$, we have $T^+(0) = T^+(1)$. Thus, the last term in the previous equation is

$$\sum_{k:(i,k)\notin F} Q_{i,k}\mathbb{P}(X_{T^+(1)} = j | X_1 = k \text{ and } X_0 = i) = \sum_{k:(i,k)\notin F} Q_{i,k} J_{k,i}$$

Combining the two gives $J = C + (Q - C)J$ which shows item 1.

Now, by definition of an intensity, $\eta = \sum_{(i,j)\in F} \mathbb{P}(X_0 = j, X_{-1} = i)$ and $\mathbb{P}(X_0 = j, X_{-1} = i) = \pi_i Q_{i,j}$, which shows item 2.

By application of Matthes's direct formula

$$\mathbb{P}^0(X_{-1} = i, X_0 = j) = \frac{1}{\eta}\mathbb{E}(1_{\{X_{-1}=j\}} 1_{\{X_0=i\}} 1_{\{(i,j)\in F\}}) = \frac{1}{\eta}\mathbb{P}(X_{-1} = j, X_0 = i) 1_{\{(i,j)\in F\}}$$

which shows item 3. Item 4 follows immediately.

□

QUESTION 11.6.1. *Is the embedded sub-chain irreducible if the original one is ?* [36]

EXAMPLE 11.14: ARP REQUESTS WITHOUT REFRESHES. IP packets delivered by a host are produced according to a Point process with λ packets per second in average. The packet delivery is a renewal source model, with the time between packet arrivals having a phase type distribution. This models an almost constant inter-arrival time. When a packet is delivered, if an ARP request was emitted not more than t_a seconds ago, no ARP request is generated. Else, an ARP request is generated. (t_a is the ARP timer). What is the probability p that an arriving packet causes an ARP request to be sent ?

Consider discrete time. We can model the system as a Markov chain X_t with state (i, s) where i is the phase of the modulator of the inter-arrival time model and $s \in$

[36] Not necessarily, it may have states that are never reached. For example, take $F = \{(0, 1)\}$; all states other than 1 are never reached.

$\{0, 1, ..., t_a\}$ is the remaining lifetime of the timer. Let $Q_{i,j}$ be the transition matrix of the modulator and θ_i the arrival rate given that the modulator is in state i. The transitions of X_t are

$$\begin{cases} (i,s) \to (j, s-1) \text{ with probability } Q_{i,j} \text{ for } s \neq 0 \\ (i,0) \to (j,0) \text{ with probability } Q_{i,j}(1-\theta_i) \\ (i,0) \to (j,t_a) \text{ with probability } Q_{i,j}\theta_i \end{cases}$$

We can thus compute the stationary probability $\pi_s(i) := \mathbb{E}(X_t = (i,s))$ from the steady-state equations (π_s is a row matrix and $\Theta = \mathrm{diag}(\theta_i)$) :

$$\begin{cases} \pi_s = \pi_{s+1} Q \;\; 0 < s < t_a \\ \pi_{t_a} = \pi_0 \Theta Q \\ \pi_0 = \pi_1 Q + \pi_0 (Id - \Theta) Q \end{cases}$$

which solves into

$$\pi_s = \pi_0 \Theta Q^{t_a + 1 - s}$$

and

$$\pi_0 = \pi_0 \Theta Q^{t_a+1} + \Pi_0 (Id - \Theta) Q$$

The last equation gives π_0 up to a multiplicative constant.

Now we apply Theorem 11.6.1 with selected transitions corresponding to a packet arrival. Call $q(i,s)$ the probability that an arriving packet sees the system in a state (i,s). We have

$$p = \sum_i q(i, t_a)$$

Now $q(i,s) = \eta^{-1}(\pi_{s+1}\Theta Q)[i]$ for $s \neq t_a$ and $q(i,t_a) = \eta^{-1}(\pi_0 \Theta Q)[i] = \eta^{-1}(\pi_{t_a})[i]$. Thus $p = \eta^{-1} \sum_i \pi_{t_a}[i]$. We compute η by the normalizing condition $\sum_{i,s} q(i,s) = 1$.

Numerical App to Erlang-k

to be completed– add Erlang

"OBSERVABLE TRANSITIONS" OF A DISCRETE TIME CHAIN. Consider a chain with more than 1 state, such that $Q_{i,i} > 0$ for some i, i.e., there are some looping states. Let C be the set of non-looping transitions: $C_{i,j} = Q_{i,j}$ for $i \neq j$ and $C_{i,i} = 0$. The embedded sub-chain is the chain that is observable. Its transition matrix is $J = D^{-1}C = \mathrm{diag}((1 - Q_{i,i})^{-1})(Q - \mathrm{diag}(Q_{i,i})$.

QUESTION 11.6.2. *Why is* $1 - Q_{i,i} \neq 0$ *in this example ?* [37]

11.6.2 DISCRETE TIME CHAIN EMBEDDED IN A CONTINUOUS TIME CHAIN.

Consider a stationary ergodic continuous time chain X_t with generator A. Let T_n be the point process of transition epochs. The embedded sub-chain has transition matrix $J = D^{-1}(A + D)$ where $D = -\mathrm{diag} A_{i,i}$ is the diagonal matrix whose ith element is the rate of transition out of state

[37] Because $Q_{i,i} < 1$ otherwise the chain is not irreducible.

i. Put differently, this says that the probability that the next transition leads to state j, starting from i, is $\frac{A_{i,j}}{D_{i,i}}$.

Define $\bar{T}_i = \mathbb{E}^0(T_1|X_0 = i)$ the mean sojourn time is state i. We know that $\bar{T}_i = \frac{1}{D_{i,i}}$. By application of Proposition 11.5.1 we find the relation between state probabilities at an arbitrary time and at an arbitrary transition:

$$\pi_i = \eta \frac{\pi_i^0}{D_{i,i}}$$

The rate of transitions η is obtained by expressing that $\sum_i \pi_i = 1$:

$$\frac{1}{\eta} = \sum_i \frac{\pi_i^0}{D_{i,i}}$$

11.6.3 PASTA

THEOREM 11.6.2 (PASTA). *Consider a system that can be modeled by a stationary, ergodic Markov chain. We are interested in a matrix of $C \geq 0$ of selected transitions such that*

- *For any state i, $\sum_j C_{i,j} = \lambda$ is a constant.*

The point process of selected transitions is a Bernoulli process (discrete time) or Poisson process (continuous time) with intensity λ. The Palm probability to be in state i just before a transition is the stationary probability.

A *Bernoulli process* is a Point process in discrete time such that $N(t)$ is an iid sequence.

Proof. (discrete time) The probability that there is a transition at time 1, given that $X_0 = i$, is λ, independent of i. Thus $N(1)$ is independent of the state at time 0. Since we have a Markov chain, the state at time 1 depends on the past only through the state at time 0. Thus $N(1)$ is independent of $N(t0)$ for all $t \geq 0$. By stationarity, it follows that $N(t)$ is iid, i.e. is a Bernoulli process.

The relation between Palm and stationary probabilities follows from Theorem 11.6.1, item 4. The Palm probability to be in state i just before a transition is

$$\frac{1}{\eta}\pi_i \sum_i C(i,j) = \frac{\lambda}{\eta}\pi_i$$

The sum of probabilities is 1, thus necessarily $\frac{\lambda}{\eta} = 1$.

□

INTERPRETATION The condition that $\sum_j C_{i,j}$ is a constant is called the "Lack of Anticipation Assumption". Another way to view the theorem is to say that a Poisson process of events sees the system as an observer at an arbitrary point in time, provided that the future of the event process is independent of the state of the system.

Interpret C as external arrivals into a queuing system. The theorem is known as "Poisson Arrivals See Time Averages", hence the acronym.

EXAMPLE 11.15: ARP REQUESTS WITHOUT REFRESHES. Consider the example in Example 11.14 on page 304, but assume that the IP packets delivered by a host are produced according to a Poisson process with intensity λ. What is the probability p that an arriving packet causes an ARP request to be sent ?

Call T_n the point process of ARP request generations, and μ its intensity. First, let p be the probability that an arriving packet causes an ARP request to be sent. We have

$$\mu = p\lambda \tag{11.6}$$

(to see why, assume time is discrete and apply the definition of intensity).

Second, let $Z_t = 1$ if the ARP timer is running, 0 if it has expired. Thus p is the probability that an arriving packet sees $Z_t = 0$. To see why the PASTA property applies, think in discrete time. The system can be modeled by a Markov chain with $X_t =$ the residual value of the timer. We have $Q_{i,i-1} = 1$ for $i > 0$, $Q_{0,t_a} = \lambda$, $Q_{0,0} = 1 - \lambda$. The selected transitions are packet arrivals, corresponding to $C_{i,i} = C_{0,t_a} = \lambda$ and $C_{i,j} = 0$ otherwise. Thus we can apply Theorem 11.6.2 in discrete time, and we extrapolate that we can do the same in continuous time. Thus $p = \mathbb{P}(Z_t = 0)$.

By the inversion formula:

$$p = \mathbb{P}(Z_t = 0) = \mu\mathbb{E}^0(T_1 - t_a) = \mu\left(\frac{1}{\mu} - t_a\right) = 1 - \mu t_a \tag{11.7}$$

Combining the two equations gives $p = \frac{1}{\lambda t_a + 1}$ (and $\mu = \frac{\lambda}{1+\lambda t_a}$).

EXAMPLE 11.16: M/GI/1 QUEUE. A similar reasoning shows that for a queuing system with Poisson arrivals, an arriving customer sees the system (just before its own arrival) in the same way as an external observer arriving at an arbitrary instant.

EXAMPLE 11.17: A POISSON PROCESS THAT DOES NOT SATISFY PASTA. The PASTA theorem requires the event process to be Poisson or Bernoulli *and* the lack of anticipation assumption. Here is an example of Poisson process that does not satisfy the lack of anticipation assumption, and does not have the PASTA property.

Construct a simulation as follows. Requests arrive as a Poisson process of rate λ into a single server queue. The service time of the request that arrives at time T_n is $\frac{1}{2}(T_{n+1} - T_n)$. The service times are exponential with mean $\frac{1}{2\lambda}$, but not independent of the arrival process. The system has exactly one customer during half of the time, and 0 customer otherwise. Thus the stationary distribution of queue length X_t is given by $\mathbb{P}(X_t = 0) = \mathbb{P}(X_t = 1) = 0.5$ and $\mathbb{P}(X_t = k) = 0$ for $k \geq 2$. In contrast, the queue is always empty when a customer arrives. Thus the Palm distribution of queue length just before an arrival is different from the stationary distribution of queue length.

The arrival process does not satisfy the lack of anticipation assumption: at a time t where the queue is not empty, we know that there cannot be an arrival; thus the probability that an arrival occurs during a short time slot depends on the state of the process.

APPLICATION TO MEASUREMENTS. The PASTA property shows that sampling a system at random observation instants, distributed like a Poisson or Bernoulli process, provides an unbiased estimator of the stationary distribution.

11.6.4 APPLICATION: PERFECT SIMULATION

In some cases, it is possible to start a simulation immediately in the stationary regime (*Perfect Simulation*). Theoretically, if we know the stationary distribution of a Markov chain, all we need to do is draw a state at random according to this distribution and run the program. In practice, this may not be as simple.

Consider the Renewal Source Model defined in Section 11.3.1. Think of it as a Markov chain, with $X_t = T^+(t) - t$, ie. the state of the model is the time until the next arrival. The stationary distribution is given by the residual life results in Theorem 11.3.2. Thus we can start the simulation immediately in steady state by drawing a random number V_1 that follows the distribution of the residual time, which has a density $f(t) = \lambda \mathbb{P}^0(U_1 > t)$. Then set $T_1 = V_1$ and from there on run the simulation as before.

11.7 EXERCICES

EXERCISE 11.1 (Residual Time). *Consider the notation of Theorem 11.3.2. Is the distribution of Z_t equal to the convolution of those of X_t and Y_t ?*

EXERCISE 11.2. *A distributed protocol establishes consensus by periodically having one host send a message to n other hosts and wait for an acknowledgement. Assume the times to send and receive an acknowledgement are iid, with distribution $F(t)$. What is the number of consensus per time unit achieved by the protocol ? Give an approximation using the fact that the mean of the kth order statistic in a sample of n is approximated by $F^{-1}(\frac{k}{n+1})$. Compare to [Bakr02-PODC].*

EXERCISE 11.3 (File Distributions). *Packets arriving at a router belong to flows. Let $P(x)$ be the probability that, for an arbitrary packet, its flow is of size x packets. Let $F(x)$ be the probability that an arbitrary flow is of length x packets. Show that there is a necessary relation between $P()$ and $F()$. Verify this relation on Figure 2 in [Anees99-Sigcomm].*

EXERCISE 11.4 (ARP protocol with refreshes). *IP packets delivered by a host are produced according to a stationary point process with λ packets per second in average. Every packet causes the emission of an ARP if the previous packet arrived more than t_a seconds ago (t_a is the ARP timer). What is the average number of ARP requests generated per second ?*

EXERCISE 11.5. *Read [Rougier00-PE] and answer the following questions. To be done.*

EXERCISE 11.6. *(Rekeying for Multicast) Read [Zhang02-Perf] and answer the following questions. To be done.*

EXERCISE 11.7 (Rate Control in the Internet). *Read [Vojnovic02-Sigcomm] and answer the following questions.*

1. *What is a TCP friendly rate control ?*
2. *Why is there a difference between TCP friendly and conservative ?*
3. *What is the loss parameter $\bar{p}$?*
4. *What is the "loss event interval" θ_n ? What is the estimator $\hat{\theta}_n$?*
5. *What is the basic control ? What is the relation between average rate and $\hat{\theta}_n$?*
6. *What is the main argument in the proof of Theorem 1 ?*
7. *What is the main conclusion of Section 5.1 ?*
8. *Do we need the Markov chain property in Section 5.1 ?*
9. *Why do the authors expect TCP to see a higher loss parameter than a rate controlled application ?*
10. *Why does a Poisson source experience the stationary loss estimator ?*

EXERCISE 11.8. *Consider the Aloha with a finite number of stations. More precisely, we consider a set of m stations running the slotted Aloha protocol. Assume they operate as follows:*

- *fresh arrivals to a station is according to a Bernoulli process, with 0 or 1 packet arrival per time slot per station. q_a is the probability of 1 arrival during one time slot. All arrival processes are independent. All packets have a transmission time equal to one time slot and all stations are synchronized.*
- *when a station experiences a collision, it becomes backlogged and remains so until the packet is successfully transmitted. Backlogged stations attempt to retransmit according to mutually independent Bernoulli trials; call q_r the probability that a given sbacklogegd station attempts to retransmit during one time slot. When a station is backlogged, all arriving (fresh) packets are simply discarded.*

1. *Compute $a(k,i)$, the probability that there are k fresh arrivals in one time slot given that there are i backlogged stations; compute $r(k,i)$, the probability that there are k retransmission attempts in one time slot given that there are i backlogged stations.*
2. *Give a discrete time Markov chain model with $(m+1)$ states for this system. Write the transition probabilities $Q(i,j)$. Express $Q(i,j)$ by using $a()$ and $r()$.*
3. *Call $p(i)$ the steady-state probability of state n; find a method to compute $p(i)$. You can use a mathematical package such as Mathematica.*
4. *Call $S(i)$ the expected number of succesful transmissions in one time slot given that there are i backlogged stations at the beginning of the slot. Show that the following holds for all i:*
$$\sum_i p(i)S(i) = \sum_i p(i)(m-i)q_a$$
(find an interpretation).
5. *In the rest of the exercise, we consider the following combinations of parameters:*
   - *$q_a = \frac{1}{m^2}$ to $\frac{1}{m}$ by increments of $\frac{1}{m}$*
   - *$q_r = q_a$; $q_r = 2q_a$; $q_r = 4q_a$*

   Compute the steady state probability p for all cases with $m = 5$ and $m = 10$.
   Do the following verifications for all *numerical cases:*

(a) *Verify that Q is a stochastic matrix*
(b) *Verify that p satisfies $pQ = p$*
(c) *Verify the equality of the previous question*

6. *For all numerical cases, compute :*
   - *the offered load λ*
   - *the throughput θ*
   - *the average transmission and retransmission rate G*
   - *the average loss ratio L*
   - *the average delay (not including the 1 slot transmission delay) for a packet which is not lost.*

   Put all these results on one diagram by plotting the results as a function of λ. Also plot θ as a function of G and comment on the results.
7. *(This question is optional) If you have numerical problems or do not want to use a numerical package, you can write an algorithm to compute $p(i)$ iteratively. For that purpose, use the cut lemma to obtain $p(i)$ as a function of $p(j)$, $j \leq i-1$.*

EXERCISE 11.9. *(Continuation of Exercise 11.7)*

1. *Consider now the cases $m = 10$. Does the previous method work ? Analyze why. How can you obtain a solution ?*
2. *An alternative is to use an ad-hoc solution method, which exploits the fact that the Markov chain does not have skips to the right, namely, transitions $n \to n-k$ are possible only for $k \leq 1$. The method below is called the Hessenberg method.*
   The idea is to compute ratios instead of the steady state probabilities.
   Define r_i by $p(i) = r_i p(i+1)$.
   (a) *Show that*
   $$p(i)\left(1 - Q(i,i) - \sum_{j \leq i-1} r_j r_{j+1} \dots r_{i-1} Q(j,i)\right) = p(i+1)Q(i+1,i)$$
   (b) *Show that*
   $$\sum_{j \leq i-1} r_j r_{j+1} \dots r_{i-1} Q(j,i) = r_0(P(0,i) + r_1(P(0,1) + \dots (r_{i-1}P(i-1,i))))$$
   (c) *Derive from the previous equations an algorithm to compute r_i iteratively, starting from r_0.*
   (d) *Solve again the case $m = 10$*

EXERCISE 11.10. *(Double Campbell)*

1. *Let T_n be a stationary point process in discrete time. Show that, for any bounded random function $F(s,t)$:*
   $$\mathbb{E}\left(\sum_{(m,n)\in\mathbb{Z}^2} F(T_m, T_n)\right) = \lambda\left(\sum_{t\in\mathbb{Z}} \mathbb{E}^t(F(t,t)) + \sum_{(s,t)\in\mathbb{Z}^2, s\neq t} \phi(t-s)\mathbb{E}^{s,t}(F(s,t))\right) \tag{11.8}$$
   where $\phi(u) = \mathbb{E}^0(N(u))$ is the probability that there is a point at time u, given that there is one at time 0. (Hint: try to extend the Proof of Campbell's formula.)

2. *Assume that T_n is a Bernoulli process, i.e., $N(t)$ is an iid sequence. What does Equation (11.8) give ?*
3. *Give the corresponding formula in continuous time.*
4. *Consider a shot noise process with a Poisson point process for the shot epochs:*

$$X(t) = \sum_{n \in \mathbb{Z}} h(t - T_n, Z_{T_n})$$

Compute the variance of $X(0)$.

11.8 APPENDIX: QUICK REVIEW OF MARKOV CHAINS

For more details see for example [Thiran02-LN] or [Bremaud01-book]. We consider a markov chain on some enumerable set E. In discrete time, the chain is given by a transition matrix Q, with $Q_{i,j} = \mathbb{P}(X_{t+1} = j | X_t = i)$. Q is a stochastic matrix, i.e. $Q(i,j) \geq 0$ and $\sum_j Q_{i,j} = 1$. In continuous time, the chain is determined by the generator matrix A, where $A_{i,j}$ is the rate of transition from state i to j; it is such that $\mathbb{P}(X_{t+dt} = j | X_t = i) = A_{i,j}dt + o(dt)$ for $i \neq j$. A has non-negative entries everywhere except on the diagonal and $\sum_j A_{i,j} = 0$.

QUESTION 11.8.1. *What is $A_{i,i}$?* [38]

Call $\pi(t)$ the row vector of probabilities at time t, i.e. $\pi_i(t) = \mathbb{P}(X_t = i)$. We have $\pi(t) = \pi(0)Q^t$ in discrete time, and $\pi(t) = \pi(0)e^{-tA}$ in continuous time. The exponential of a matrix is defined like for complex numbers by $e^A = \sum_{n=0}^{\infty} A^n/n!$.

A stationary probability is a row vector π that satisfies $\pi Q = \pi$ (discrete time) or $\pi A = 0$ (continuous time), is wide-sense positive, and sums to 1. For a finite state space E, there is always at least one stationary probability. There may be several if the chain branches into subsets of state spaces from which it cannot exit. For an infinite state space, there may not exist a stationary probability (the chain "escapes to infinity").

The chain is *stationary* if $\pi(t)$ is independent of t. For a chain starting at time 0, this is true iff the initial probability distribution $\pi(0)$ is a stationary probability.

The chain is irreducible if any state can be reached from any state. The chain is positive is the steady-state equation $\mu Q = \mu$ (discrete time) or $\mu A = 0$ (continuous time) has at least one solution μ with finite sum (μ is a row vector). If the chain is not irreducible, there may be some states such that $\pi_i = 0$ for a stationary probability π.

The chain is *ergodic* if it is irreducible positive, and for discrete time, aperiodic. If so, the stationary probability is defined as the only solution of the steady-state equation that sums to 1. Such a solution is necessarily positive. For an ergodic chain, we have $\lim_{t \to +\infty} \pi(t) = \pi$ where π is the unique stationary probability.

An ergodic chain is also stationary iff the initial probability $\pi(0)$ is the stationary probability π. Otherwise, it becomes stationary for t large enough.

For a continuous time markov chain, the time until the next transition given that $X_t = i$ is an exponential random variable with parameter $d(i) = -A_{i,i}$.

[38] $A_{i,i} = -\sum_{j:j \neq i} A_{i,j}$.

A *phase type* distribution is the lifetime of a finite, transient markov chain X_t, defined as follows. There are $I+1$ states labeled $0, 1, ..., I$; state 0 is the final state. The random variable T is the first time $t \geq 0$ for which $X_t = 0$. Let $A_{i,j}$ be the rate of transition from state $i \neq 0$ to state j, $d(i) = \sum_{j:i \neq i} A_{i,j}$ (departure rate), and α_i the probability that the chain in in state i at time 0. We assume that $\alpha_0 = 0$. The moment generating function of T, $m(s) := \mathbb{E}(e^{sT})$, is obtained by solving the set of linear equations, defined for all $i = 1...I$:

$$m_i(s) = \left(\sum_{j:j \neq 1} \frac{A_{i,j}}{d(i)} m_j(s) + \frac{A_{i,1}}{d(i)} \right) \frac{d(i)}{d(i) - s} = m_i(s)$$

which are obtained by letting $m_i(s) := \mathbb{E}(e^{sT} | X_0 = i)$.

Special cases often used are

- the hypo-exponential distribution, for which $A_{i,j} = 0$ except for $i < I, j = i + 1$ or $i = I, j = 0$. If the non-zero rates $A_{i,j}$ are all the same, this is the Erlang-I distribution.
- the hyper-exponential distribution, for which $A_{i,j} = 0$ except for $j = 0$

PH-type distributions have a rational moment generating function (quotient of two polynoms). They can approximate any distribution, in some sense.

PART III

APPENDIX

CHAPTER 12

PROBABILITY THEORY AND TABLES

Probability derives the properties of models. A model is, in general in our framework, a collection of random variables (independent or not). It is a branch of pure maths: given a model, we can derive proven properties and do computations. In contrast, statistics starts when the model itself is not known. The problem of statistics is to infer a model from the data and return something useful about the data. Determining a model is not a pure mathematical exercise, in the sense that it is not possible to prove formally whether a model is appropriate or not – though in many cases some models are obviously wrong. Much of this book is about finding the right model for the right situation. In this chapter, we give the results of probability theory that we used throughout the book.

Contents

12.1 RANDOM VARIABLES AND DISTRIBUTIONS

A real random variable is a mapping from the set of randomness Ω to $\mathbb{R}$, i.e. the output of a random generator that produces a real number. For a real random variable X:

- The *Cumulative Distribution Function* (*cdf*) is the function $F : \mathbb{R} \to [0, 1]$ defined by $F(x) = \mathbb{P}(X \leq x)$. The distribution of a random variable is entirely defined by its cdf. A cdf is always right-continuous, i.e. $F(x) = \lim_{x \to c^+}(x)$ for all $c \in \mathbb{R}$.
- A *probability density function* (*pdf*) of X is a function $f : \mathbb{R} \to \mathbb{R}^+$ such that for any subset $A \subset \mathbb{R}$: $\mathbb{P}\{X \in A\} = \int_{x \in A} f(x)dx$, if it exists. f is defined up to a zero mass set, which means in particular that you can change the value of f on a enumerable set of points and still obtain a density. The distribution of a random variable that has a density is entirely defined by its pdf. There are random variables that do not have a density, for example the (degenerate) random variable that is deterministically equal to some (non random) value x_0. Its cdf is called the Dirac mass at x_0.

12.1.1 RANDOM VECTORS

Covariance matrix

12.1.2 CHANGE OF VARIABLE

12.2 CONVERGENCE RESULTS

Def of convergence in distrib and in proba and in l^2

THEOREM 12.2.1 (Slutzky's Lemma). *Slutzky's lemma (conv en proba + en loi implique en loi*
If X_n converges in distribution to X and Y_n converges in probability to a constant c, then X_n/Y_n converges in distribution to X/c

CLT

12.3 ORDER STATISTIC

$(X_{(j)}, X_{(k)})$ has a density.
$\mathbb{E}(X_{(j)}) = \frac{j}{n+1}$ if distrib is uniform

12.4 LINEAR ALGEBRA AND NOTATION

12.4.1 GENERAL NOTATION

We consider in this chapter a linear space E of finite dimension. We assume the reader is familiar with the definitions of linear space, linear mapping, dimension, and coordinate system.

BASIS, COORDINATES Any linear space has a basis, and all bases have the same number of elements. If this number is finite, it is the dimension of the linear space. Any vector $\vec{x}$ can be written uniquely as a linear combination of elements of the basis.

Examples:

- $E = \mathbb{R}^I \times \mathbb{R}^J \times \mathbb{R}^K$. A vector is a triple (X, Y, Z) with $X \in \mathbb{R}^I, Y \in \mathbb{R}^J, Z \in \mathbb{R}^K$. The dimension is $I + J + K$.
- $E = \mathbb{R}[I, J, K]$. A vector is an array $a[,,]$ with three indices (more generally, n indices). This set is called the set of "tensors" in physics. For $1 \leq i \leq I, 1 \leq j \leq J, 1 \leq k \leq K$, $a[i, j, k]$ is a real number. The dimension is IJK.
  The tensor $a[,,]$ can be written

$$a[,,] = \sum_{i,j,k} a[i, j, k] z_{i,j,k}$$

  where $z_{i,j,k}$ is the tensor defined by $z_{i,j,k}[i', j', k'] = 1_{\{i=i'\}} 1_{\{j=j'\}} 1_{\{k=k'\}}$. The list

$$(z_{i,j,k})_{1 \leq i \leq I, 1 \leq j \leq J, 1 \leq k \leq K}$$

  is a basis of the space of tensors and $a[i, j, k]$ is the coordinate of $a[,,]$ attached to the element $z_{i,j,k}$ of the basis.

It is traditional to identify a vector $\vec{x}$ and its $p \times 1$ matrix of coordinates X in some well-know basis (p is the dimension of the linear space). However, this may sometimes be counter-productive, for example for reasoning about tensors. There are many ways to map an array $a[,,]$ to a column matrix of coordinates. This is unpleasant to write and is best left to the statistical software to handle. An example where this occurs is given in Section **??**. Another example is with wavelet analysis in Chapter 13.

LINEAR MAPPING AND MATRIX A *linear mapping* f from a space E with dimension p to a space F with dimension q is a mapping such that $f(\alpha\vec{x} + \beta\vec{y}) = \alpha f(\vec{x}) + \beta f(\vec{y})$ for all real numbers α, β and all vectors $\vec{x}, \vec{y} \in E$. Let $(\vec{e}_i)_{1 \leq i \leq p}$ [resp. $(\vec{f}_j)_{1 \leq j \leq q}$] be a basis of E [resp. F]. The matrix A associated with f is the two-dimensional array defined by

$$A[r, s] = r\text{th coordinate of } f(\vec{e}_s)$$

A linear mapping is commonly identified with its matrix, assuming there is a non-ambiguous, well defined basis. However, it is sometimes useful to make the distinction. See Section **??** for a place where such a viewpoint is helpful.

- The *null space* of f is the set of $\vec{x}$ such that $f(\vec{x}) = \vec{0}$.
- The *image* of f is the set of $\vec{x}$ that can be written $\vec{x} = f(\vec{y})$ for some $\vec{y}$
- Both null space and image are linear subspaces of (E and F respectively). The dimension theorem says that the sum of their dimensions is the dimension of E.

OTHER NOTATION For two arbitrary sets E, F:

- $E \times F$ is the set of couples (e, f) where $e \in E$ and $f \in F$.
- E^F is the set of mappings from F to E. When F is finite, this is the same as the set of arrays indexed by F.

12.4.2 DIRECT SUMS

Let E be a linear space and E_i, $i = 1...k$ sub-linear spaces of E. If any $\vec{x} \in E$ can be decomposed in a unique way as

$$\vec{x} = \vec{x}_1 + ... + \vec{x}_k$$

where $\vec{x}_i \in \mathbb{E}_i$, then we say that E is the *direct sum* of the E_is and we write

$$E = E_1 \oplus E_2 ... \oplus E_k$$

Example: let $E = \mathbb{R}[I, J]$. Let E_1 be the set of constant arrays, E_2 [resp. E_3] the set of arrays that depend only on i [resp. j] and that sum to 0, and E_4 the set of arrays $a[,]$ such that $\sum_i a[i, j] = \sum_j a[i, j] = 0$.

QUESTION 12.4.1. *Show that* $E = E_1 \oplus E_2 \oplus E_3 \oplus E_4$ [1]

12.4.3 PROJECTOR

A *projector* is a linear mapping f from E to E such that $f \circ f = f$, i.e. $f(f(\vec{x})) = f(\vec{x})$ for all $\vec{x}$. Then E is the direct sum of the null space of f and the image of f.

Conversely, consider a direct sum $E = E_1 \oplus E_2$ and let $\vec{x} = \vec{x}_1 + \vec{x}_2$ be the corresponding decomposition. The mapping from $\vec{x}$ to $\vec{x}_1$ is a projector, with null space E_2 and image E_1. Thus a projector is entirely defined by its null space and its image.

We have the following characterization. For any $\vec{x} \in E$, $\Pi_{E_1}(\vec{x})$ is the unique vector such that

$$\begin{cases} \Pi_{E_1}(\vec{x}) \in E_1 \\ \vec{x} - \Pi_{E_1}(\vec{x})) \in \mathbb{E}_2 \end{cases} \tag{12.1}$$

[1] Hint: write

$$a[i, j] = \bar{a} + (\bar{a}[i, .] - \bar{a}) + (\bar{a}[., j] - \bar{a}) + (a[i, j] - \bar{a}[i, .] - \bar{a}[., j] + \bar{a})$$

with $\bar{a} = \frac{1}{IJ} \sum_{i,j} a[i, j], \bar{a}[i, .] = \frac{1}{J} \sum_{i,j} a[i, j], \bar{a}[., j] = \frac{1}{J} \sum_{i,j} a[i, j]$.

12.4.4 INNER PRODUCT, ISOMETRY

Notation: *inner product* $\vec{u} \cdot \vec{v}$. It is equal to $u^T v$ where u, v is the column vector of coordinates in some orthonormal basis.

$$\|\vec{x}\| := \sqrt{\vec{x} \cdot \vec{x}}$$

Two subspaces E_1 and E_2 are orthogonal iff $\vec{x}_1 \cdot \vec{x}_2 = 0$ for all $\vec{x}_1 \in E_1, \vec{x}_2 \in E_2$.

An *isometry* is a mapping that preserves the norm. It is necessarily linear.

MAPPING A VECTOR TO ITS COORDINATES Let $(\vec{e}_i)i \in I$ an orthornormal basis of E, $\vec{x} \in E$ and X^T the column vector of coordinates of $\vec{x}$ in this basis. The mapping $\vec{x} \to X$ is an isometry from E to $\mathbb{R}^I$

12.4.5 ORTHOGONAL PROJECTORS

Given a sub-space E_1, the set of vectors $\vec{y}$ that are orthogonal to *all* vectors in E_1 is called the *orthogonal* of E_1. E_1 and its orthogonal are in direct sum. The projector with image E_1 and null space the orthogonal of E_1 is called the *orthogonal projector* on to E_1 and is denoted with Π_{E_1}.

The following characterization follows from Equation (12.1). For any $\vec{x} \in E$, $\Pi_{E_1}(\vec{x})$ is the unique vector such that

$$\begin{cases} \Pi_{E_1}(\vec{x}) \in E_1 \\ \text{for all } \vec{y} \in E_1 \ : (\vec{x} - \Pi_{E_1}(\vec{x})) \cdot \vec{y} = 0 \end{cases}$$

The following theorem relates minimization of sums of squares to orthogonal projectors.

THEOREM 12.4.1. *The optimization problem (where $\vec{y}$ is the unknown)*

$$\begin{array}{c} \textit{minimize } \|\vec{x}_0 - \vec{y}\|^2 \\ \textit{subject to } \vec{y} \in E_1 \end{array}$$

has a unique solution, equal to $\vec{y} = \Pi_{E_1}(\vec{x}_0)$

Example: Haar function

12.5 NORMAL VECTORS

Let $\vec{X}$ be a random vector in a finite dimension. Then if h is linear, non random:

$$\mathbb{E}(h(\vec{X})) = h(\mathbb{E}(\vec{X}))$$

In matrix form, for any non-random matrix H, $\mathbb{E}(HX) = H\mathbb{E}(X)$

12.5.1 COVARIANCE FORM

For a random vector $\vec{X}$ such that all first and second moments are defined, the *covariance form* ω is a symmetric bilinear form defined by

$$\omega(\vec{u}, \vec{v}) = \mathrm{cov}(\vec{u} \cdot \vec{X}, \vec{v} \cdot \vec{X})$$

or equivalently

$$\omega(\vec{u}, \vec{u}) = \mathrm{var}(\vec{u} \cdot \vec{X})$$

In some orthonormal basis where we identify $\vec{X}$ with a column vector X of coordinates, the matrix Ω of ω is called the *covariance matrix*. It is given by

$$\Omega := \mathbb{E}((X - \mu)(X - \mu)^T)$$

with $\mu = \mathbb{E}(X)$. In matrix form:

$$\mathrm{cov}(u^T X, v^T X) = u^T \Omega v$$

$$\mathrm{var}(u^T X) = u^T \Omega u$$

Now consider a new basis, where the coordinates of $\vec{X}$ is X'. Let A be the square matrix defined by:

$$A[r, s] = r\text{th coordinate, in the old basis, of the } s\text{th vector of the new basis.}$$

Then $X = AX'$. It follows that $\mathrm{cov}(u^T X, v^T X) = u^T \Omega v = u'^T A^T \Omega A v'$ thus the covariance matrix of $\vec{X}$ in the new basis is

$$\Omega' = A^T \Omega A$$

ω is obviously a wide-sense positive form, i.e. $\omega(\vec{u}, \vec{u}) \geq 0$. From the general theory of bilinear forms, we know that there exists an orthonormal basis $\vec{f_1}, ..., \vec{f_n}$ in which the matrix of ω is diagonal with diagonal terms $\lambda_i \geq 0$. If we call X_i the ith coordinate of $\vec{X}$ in the basis $\vec{f_1}, ..., \vec{f_n}$, then the collection of random variables X_i is non-correlated (i.e. $\mathrm{cov}(X_i, X_j) = 0$).

The *null space* of the random vector X is the space generated by those vectors $\vec{f_i}$ for which $\lambda_i = 0$. Its dimension is n minus the rank of the matrix of ω in any basis. It can be computed by solving $u^T \Omega = 0$ where u is the column vector of coordinates of $\vec{u}$ is some basis, and Ω the matrix of ω in the same basis. Ω is invertible iff the null space is $\{0\}$.

The null space is also the set of vectors $\vec{u}$ such that $\mathrm{var}(\vec{u} \cdot \vec{X}) = 0$, i.e., $\vec{u} \cdot \vec{X}$ is a.s. constant. In other words, X takes its values in the affine sub-space orthogonal to the null space that contains the mean $\vec{\mu}$. The dimension of this sub-space is the rank of ω. In any basis, the direction of this sub-space is the linear space generated by the columns of Ω.

EXAMPLE. In $\mathbb{R}^3$, let the covariance matrix be

$$\Omega = \begin{pmatrix} a & 0 & a \\ 0 & b & b \\ a & b & a+b \end{pmatrix}$$

The rank is 2. The linear space generated by the columns of Ω is the plane defined by $x+y-z = 0$. Thus the random vector takes its values almost surely in the plane defined by $x+y-z = x_0+y_0-z_0$ where $\mu = (x_0, y_0, z_0)$.

12.5.2 NORMAL VECTOR

A random vector $\vec{X}$ is normal iff for any $\vec{u} \in \mathbb{R}^n$, the real random variable $\vec{u} \cdot \vec{X}$ has a normal distribution. The expectation and the covariance matrix completely characterize a normal distribution.

DENSITY If Ω has full rank, $\vec{X}$ has a density, given by

$$f_{\vec{X}}(\vec{x}) = \frac{1}{\sqrt{(2\pi)^n \det \Omega}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu})^T \Omega^{-1} (\vec{x}-\vec{\mu})}$$

(in the above, we identified a vector and its coordinates). Else, $\vec{X}$ spans an affine sub-space of dimension equal to the rank of Ω.

CHARACTERISTIC FUNCTION In all cases, the characteristic function is

$$\mathbb{E}(e^{iu^T X}) = e^{iu^T \mu - \frac{1}{2} u^T \Omega u}$$

For any normal vector $\vec{X}$, there exists an orthonormal basis $(\vec{f_1}, \vec{f_2}..., \vec{f_n})$ in which the r first coordinates of $\vec{X}$ are mutually independent, and the $n-r$th others are almost surely constant. Here, r is the rank of Ω. This follows from diagonalization of the covariance matrix Ω.

If $X_1, ..., X_n$ is normal and is a sequence of independent variables, then a change of coordinate system basis will, in general, not keep independence (except for homoscedastic vectors, see below).

12.5.3 THE EUCLIDIAN SPACE OF A NORMAL PROCESS

Given a normal process, the linear combinations of it form a Hilbert space. Homoscedasticity means that it is the same as normal geometry.

Otherwise, the rank of Ω_n is the dimension of the space generated by $X_1, ...X_n$.

12.5.4 HOMOSCEDASTIC VECTOR

THEOREM 12.5.1. *If the matrix of the covariance form of a random vector is $\sigma^2 Id$ in one orthonormal basis, with $\sigma \in \mathbb{R}^+$, then the same holds in any other orthonormal basis.*

DEFINITION 12.5.1. *A normal vector is called Homoscedastic if its covariance matrix in one basis is $\sigma^2 Id$ for some $\sigma > 0$.*

Thus if $X_1, X_2, ..., X_n$ is jointly normal, saying that it is homoscedastic means that $X_i = \mu_i + \epsilon_i$, with μ_i non-random and ϵ_i normal iid with.

A homoscedastic normal vector always has a density (since its covariance matrix is invertible), given by

$$f_X(\vec{x}) = \frac{1}{(2\pi)^{\frac{n}{2}} \sigma^n} e^{-\frac{1}{2\sigma^2} \|\vec{x}-\vec{\mu}\|^2}$$

THEOREM 12.5.2. *Consider an homoscedastic normal vector $\vec{X}$ with values in a space E and let $\vec{\mu} = \mathbb{E}(\vec{X})$.*

- *For any orthogonal transformation U of E, $U(\vec{X})$ is also homoscedastic (with same common variance σ^2).*
- *Let Π_M be the orthogonal projector on some linear sub-space M. $\Pi_M(\vec{X})$ and $\vec{Y} = \vec{X} - \Pi_M(\vec{X})$ are independent, $\|\Pi_M(\vec{X}) - \Pi_M(\vec{\mu})\|^2 \sim \chi^2_m$ and $\|\vec{Y} - \vec{\mu} + \Pi_M(\vec{\mu})\|^2 \sim \chi^2_{n-m}$ where $n = dim E$ and $m = \dim M$.*

MLE FOR HOMOSCEDASTIC NORMAL VECTORS

THEOREM 12.5.3. *Consider a vector $\vec{X}$ of* independent, normal *random variables X_r with common variance σ^2, where the index r is in some finite set R (N is the number of elements in R). Assume that $\vec{\mu} := (\mu_r)_{r\in R}$ is restricted to a linear subspace M of $\mathbb{R}^R$. Let $k = \dim M$.*

- *The MLE of $(\vec{\mu}, \sigma)$ is given by*
$$\hat{\mu} = \Pi_M(\vec{X})$$
$$\hat{\sigma}^2 = \frac{1}{N}\|\vec{X} - \hat{\mu}\|^2$$
- $\mathbb{E}_{\vec{\mu},\sigma}(\hat{\mu}) = \vec{\mu} = \mathbb{E}_{\vec{\mu},\sigma}(\vec{X})$
- *Under $\vec{\mu}, \sigma$: $\vec{X} - \hat{\mu}$ and $\hat{\mu}$ are independent normal vectors. Further*
$$\|\vec{X} - \vec{\mu}\|^2 = \|\vec{X} - \hat{\mu}\|^2 + \|\vec{\mu} - \hat{\mu}\|^2$$
- *Under $\vec{\mu}, \sigma$: $\|\vec{X} - \hat{\mu}\|^2 \sim \chi^2_{N-k}\sigma^2$ and $\|\hat{\mu} - \vec{\mu}\|^2 \sim \chi^2_k\sigma^2$*
- *(Fisher distribution) Under $\vec{\mu}, \sigma$:*
$$\frac{\frac{\|\hat{\mu}-\vec{\mu}\|^2}{k}}{\frac{\|\vec{X}-\hat{\mu}\|^2}{N-k}} \sim F_{k,N-k}$$

Proof. The log likelihood of an observation $(x_r)_{r\in R}$ is

$$l_x(\vec{\mu}, \sigma) = -\frac{1}{2}\ln(2\pi) - N\ln(\sigma) - \frac{1}{2\sigma^2}\sum_{r\in R}(x_r - \mu_r)^2 = -\frac{1}{2}\ln(2\pi) - N\ln(\sigma) - \frac{1}{2\sigma^2}\|\vec{x} - \vec{\mu}\|^2 \quad (12.2)$$

For a given σ, by Theorem 12.4.1, the log-likelihood is maximized for $\vec{\mu} = \hat{\mu} = \Pi_M(\vec{x})$, which is independent of σ. Let $\vec{\mu} = \hat{\mu}$ in Equation (12.2) and maximize with respect to σ, this gives the first item in the theorem. The rest follows from Theorem 12.5.2.

□

COROLLARY 12.5.1. *Let $(X_i)_{i=1\dots n} \sim N(\mu, \sigma^2)$.*

- *The MLE of $(\vec{\mu}, \sigma)$ is given by*
$$\hat{\mu} = \bar{X} := \frac{1}{n}\sum_{i=1}^{n} X_i$$
$$\hat{\sigma}^2 = \frac{1}{n}S_{XX} := \frac{1}{N}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

- *Under μ, σ: S_{XX} and $\bar{X}$ are independent. Further*

$$\sum_i (X_i - \mu)^2 = S_{XX} + n(\bar{X} - \mu)^2$$

- *Under $\vec{\mu}, \sigma$: $S_{XX} \sim \chi^2_{n-1}\sigma^2$ and $\bar{X} \sim N(\mu, \frac{\sigma^2}{n})$.*
- *(Student distribution): Under $\vec{\mu}, \sigma$*

$$\frac{\sqrt{n}\bar{X}}{\sqrt{\frac{S_{XX}}{n-1}}} \sim t_{n-1}$$

12.5.5 CONDITIONAL NORMAL DISTRIBUTION

Assume we have a decomposition of the linear space into two orthogonal sub-spaces. Let $\vec{X} = \vec{X}_1 + \vec{X}_2$ be the corresponding decomposition of a normal vector $\vec{X}$. In matrix form: $X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$, if we take a basis compatible with the decomposition.

The covariance matrix of X can be decomposed into blocks as follows.

$$\Omega = \begin{pmatrix} \Omega_{1,1} & \Omega_{1,2} \\ \Omega_{2,1} & \Omega_{2,2} \end{pmatrix}$$

where $\Omega_{i,j}$ (*cross-covariance matrix*) is defined by

$$\Omega_{i,j} = \mathbb{E}((X_i - \mathbb{E}(X_i))(X_j - \mathbb{E}(X_j))^T)$$

X_1 and X_2 are independent iff $\Omega_{1,2} = 0$. Note that $\Omega_{1,2} = \Omega_{2,1}^T$

THEOREM 12.5.4 ([Davison02-book]). *If $\Omega_{2,2}$ is invertible, the conditional distribution of X_1 given that $X_2 = x_2$ is well defined and is normal, with mean*

$$\mu_1 + \Omega_{1,2}\Omega_{2,2}^{-1}(x_2 - \mu_2)$$

and covariance matrix

$$\Omega_{1,1} - \Omega_{1,2}\Omega_{2,2}^{-1}\Omega_{2,1}$$

Note that the covariance matrix is independent of x_2. This is true only for normal vectors.

12.5.6 PARTIAL CORRELATION

[Davison02-book]

Consider the case $\vec{X}_1 = (X_1, 0, ..., 0, X_n)^T$ and $\vec{X}_2 = (0, X_2, ..., X_{n-1}, 0)^T$. The conditional covariance matrix of $\vec{X}_1$ given $\vec{X}_2$ is a 2×2 matrix. Let it be $\begin{pmatrix} \gamma_{1,1} & \gamma_{1,n} \\ \gamma_{1,n} & \gamma_{n,n} \end{pmatrix}$. The number $\gamma_{1,n}$ is called the *partial covariance* and $r_{1,n} = \gamma_{1,n}/\sqrt{\gamma_{1,1}\gamma_{n,n}}$ the *partial correlation* of X_1 and X_n. The partial correlation expresses the residual correlation between X_1 and X_n when we know the other variables $X_2, ..., X_{n-1}$.

THEOREM 12.5.5 ([Davison02-book]). *When Ω (the covariance matrix of the joint vector $(X_1, X_2, ..., X_{n-1}, X_n)^T$ has full rank, the partial correlation of X_1 and X_n is given by the relation*

$$r_{1,n} = \frac{\tau_{1,n}}{\sqrt{\tau_{1,1}\tau_{n,n}}}$$

where $\tau_{i,j}$ is the (i, j)th term of Ω^{-1}.

If $X_1, ..., X_n$ is a Markov chain, and $n > 1$, then X_n is independent of X_1, given $X_2, ..., X_{n-1}$. In such a case, the partial correlation of X_1 and X_n is 0 (but the covariance of X_1 and X_n is not 0). Partial correlation can be used to test if a Markov chain model is adequate.

CHAPTER 13

ORTHOGONAL WAVELETS AND MULTIRESOLUTION ANALYSIS

We give a short summary of key facts related to orthogonal wavelets. For a more general theory, including non-orthogonal wavelets (called "bi-orthogonal") see the course page at `lcavwww.epfl.ch` and [Vetterli95-book].

13.1 HILBERT SPACES

transposition and scalar product

Notation: inner product $\vec{u} \cdot \vec{v}$. It is equal to $u^T v$ where u, v is the column vector of coordinates in some orthonormal basis.

Define inner product and Hilbert space E. Linear form: def, matrix, continuous, representation in Hilbert spaces.

Linear combinations and series in case of Hilbert.

13.2 MULTI-RESOLUTION ANALYSIS

Wavelets are defined for functions of continuous time (but we will apply them to time series, i.e. functions of discrete time, see later). We consider functions that are square integrable, thus we are in the Hilbert space $L^2(\mathbb{R})$. Orthogonal wavelets come in pairs: a *father wavelet* $\phi(t)$, also called *scaling function* and a *mother wavelet*, or just "wavelet", $\psi(t)$. They are such that

$$\int_{t\in\mathbb{R}} \phi(t)dt = 1 \text{ and } \int_{t\in\mathbb{R}} \psi(t)dt = 0$$

Orthogonal wavelets are required to have some other special properties, some of them are mentioned as needed in the rest of this document.

Examples: Haar wavelet. Put a figure. Other orthogonal wavelets: Daublets, Symmlets, Coiflets. Different wavelet families give slightly different decompositions. The only important aspect we will use is the number of vanishing moments and the regularity (see below).

For $j \in \mathbb{Z}$ (*octave*) and $k \in \mathbb{Z}$ (*location*) define the *dilatations* and *translations* of the wavelet functions by

$$\phi_{j,k}(t) = 2^{-j/2}\phi(2^{-j}(t - 2^j k)) \quad \psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}(t - 2^j k))$$

2^j is also called the *scale* parameter: think of it as the inverse of a frequency parameter, roughly speaking.

QUESTION 13.2.1. *Draw $\phi_{2,10}$ and $\psi_{3,-5}$ for the Haar wavelets.* [1]

Call V_j the space generated by $\phi_{j,k}$, $k \in \mathbb{Z}$. The orthogonal wavelets are such that the sequence $\phi_{0,k}$ constitute an orthonormal basis of V_0, i.e.

$$\int_{t\in\mathbb{R}} \phi(t-k)\phi(t-h)dt = 0 \text{ if } k \neq h \text{ and } 1 \text{ if } k = h$$

and similarly the set of $\phi_{j,k}$, $k \in \mathbb{Z}$ constitute an orthonormal basis of V_j.

QUESTION 13.2.2. *Verify that $\phi_{0,k}$ constitute an orthonormal basis of V_0 for the Haar wavelet* [2]

QUESTION 13.2.3. *What is V_0 for for the Haar wavelet ?* [3]

QUESTION 13.2.4. *Does a high octave number j correspond to a high frequency ?* [4]

MULTI-RESOLUTION ANALYSIS, STEP 0. Consider a fixed function $f(t)$. We call $C_0 := \Pi_{V_0}(f)$ the projection of f on V_0. It follows that

$$C_0(t) = \sum_{k\in\mathbb{Z}} c_{0,k}\phi_{0,k}(t)$$

with

$$c_{0,k} = \int_{t\in\mathbb{R}} f(t)\phi(t-k)dt$$

$C_0(t)$ is a smooth approximation of $f(t)$. The difference $f(t) - C_0(t)$ is the initial detail, in practice we expect it to be negligible (but see below for a discussion).

MULTI-RESOLUTION ANALYSIS, STEP n. Multi-resolution analysis is based on coarser and coarser approximations of $C_0(t)$. First, the wavelets are such that $\phi(t/2) \in V_0$. In other words, there exist a sequence u_k such that $\phi(t/2) = \sum_{k\in\mathbb{Z}} u_k\phi(t-k)$ (equality is in the mean square sense).

QUESTION 13.2.5. *Find u_k for the Haar wavelets* [5]

[1] tbd
[2] tbd
[3] The set of functions $f(t)$ that are constant between integers.
[4] No, with our convention, it is the opposite. Negative octaves correspond to high frequencies.
[5] tbd

Now consider V_1, the space generated by the sequence $\phi_{1,k}$. It follows that $V_1 \subset V_0$. Let $C_1 := \Pi_{V_1}(C_0)$ be the projection of C_0 on V_1 and $D_1 := C_0 - C_1$. In other words

$$C_0(t) = D_1(t) + C_1(t)$$

$C_1(t)$ is a coarser approximation of $f(t)$ than $C_0(t)$.

The step can be iterated by considering V_n, the space generated by $\phi_{n,k}$, $k \in \mathbb{Z}$, C_n the projection of C_{n-1} on V_n and

$$D_n(t) = C_n(t) - C_{n-1}(t)$$

for $n = 1$ to some integer J. Thus we have $V_0 \supset V_1 \supset ... \supset V_J$ and

$$C_0(t) = D_1(t) + D_2(t) + ... + D_n(t) + C_J(t) \tag{13.1}$$

$C_J(t)$ is called the *coarse* approximation of C_0 at octave J, and D_n the *detail* at octave n. C_J is a coarser approximation than C_0 (Figure 13.1 and Figure 13.2). Equation (13.1) is called a *multi-resolution analysis* of $f(t)$ at octaves 0 to J.

ENERGY AT OCTAVE j By construction, the details D_j and C_n are mutually orthogonal. Thus ("conservation of energy"):

$$\|C_0\|^2 = \|D_1\|^2 + ... + \|D_n\|^2 + \|C_n\|^2$$

Also, the smooth approximation C_0 and f are orthogonal, thus

$$\|f\|^2 = \|C_0\|^2 + \|f - C_0\|^2$$

QUESTION 13.2.6. *Write the conservation of energy in terms of integrals* [6]

NEGATIVE OCTAVES In practice (see below) the function $f(t)$ is very close to its projection on V_0, and multi-resolution analysis works as explained above. A general property of wavelets is that the sequence of V_j, when j goes to $-\infty$, is dense in $L^2(\mathbb{R})$, in other words, any function can be approximated by its projection on V_j for some j.

If the difference between $f(t)$ and $C_0(t)$ is not negligible, multi-resolution should be started at some negative octave J_0, instead of at octave $J_0 = 0$. The rest is without change. Note that the functions C_j and D_j are always the same, independent of the octave J_0 at which we start the multi-resolution.

13.3 THE SCALING AND WAVELET COEFFICIENTS

The orthogonal wavelets are such that the sequence $\phi_{j,k}$, $k \in \mathbb{Z}$, is an orthonormal basis of V_j, and the sequence $\psi_{j,k}$, $k \in \mathbb{Z}$, is an orthonormal basis of the orthogonal of V_j in V_{j-1}. Thus, we can write

$$\begin{cases} C_j(t) = \sum_{k\in\mathbb{Z}} c_{j,k}\phi_{j,k}(t) \\ D_j(t) = \sum_{k\in\mathbb{Z}} d_{j,k}\psi_{j,k}(t) \end{cases}$$

[6] tbd

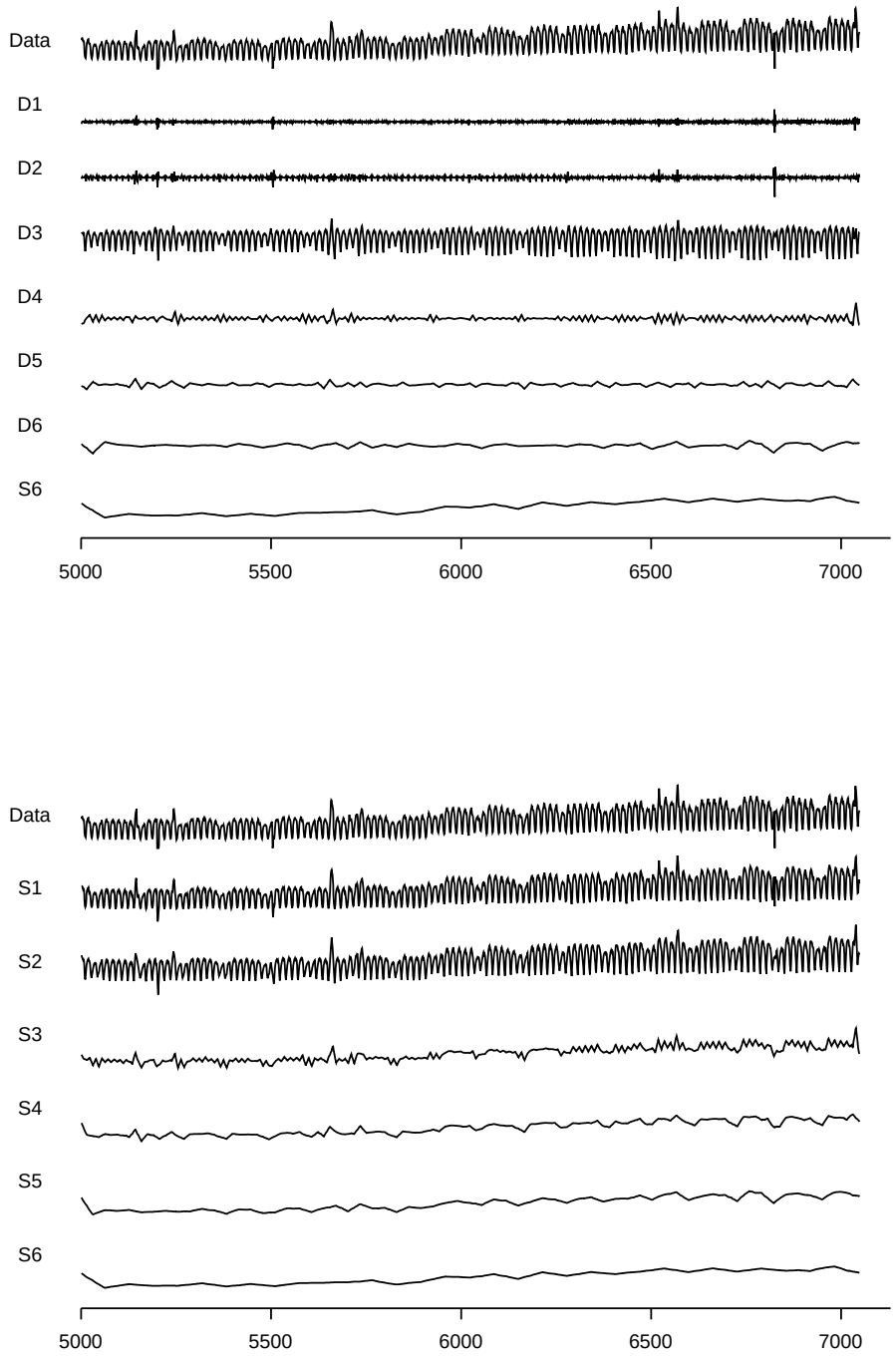
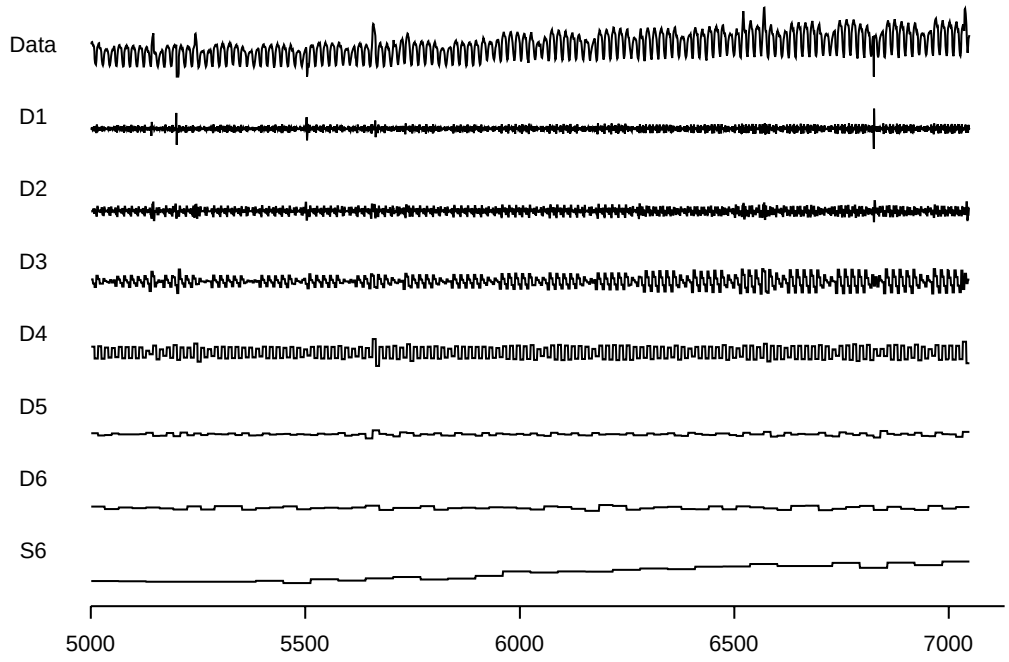


Figure 13.1: First graph: Decomposition of $C_0(t)$ into a coarse approximation $C_6(t)$ and successive details $D_j(t)$. Second graph: the successive coarse approximations $C_i(t)$. The data is equal to $C_0(t)$ and is shown at the top of each graph; it is the amount of internet traffic in bytes on a backbone link of the American operator SPRINT; one point is the aggregate over 90 mns. Wavelet basis: Daubechies 6. The details are high at octaves 3 and 4, which corresponds to timescales of 12 hours and 24 hours.



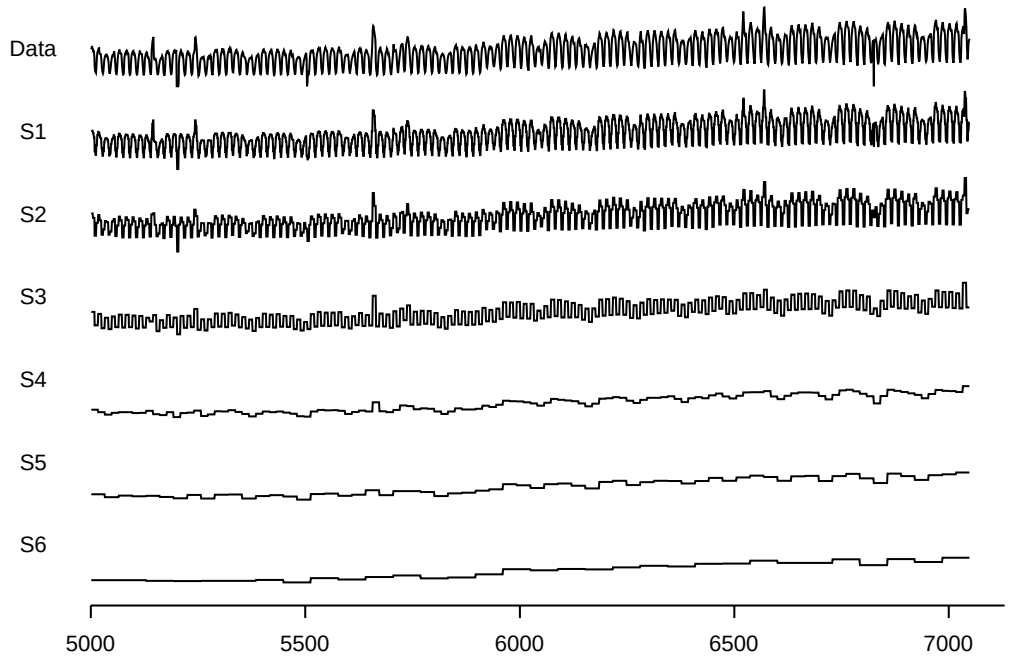


Figure 13.2: Same as Figure 13.1 but with Wavelet = Haar.

with

$$\begin{cases} c_{j,k} = \int_{t\in\mathbb{R}} f(t)\phi_{j,k}(t)dt \\ d_{j,k} = \int_{t\in\mathbb{R}} f(t)\psi_{j,k}(t)dt \end{cases} \tag{13.2}$$

$c_{j,k}$ is called a *scaling coefficient* and $d_{j,k}$ a *wavelet coefficient*.

THE PYRAMIDAL ALGORITHM. The coefficients $c_{j,k}, d_{j,k}$ are not computed by means of Equation (13.2). Instead, a *discrete wavelet transform* (DWT) is used, based on the *pyramidal algorithm*.

It computes $c_{j,k}$ and $d_{j,k}$, assuming that the coefficients at scale 0, $c_{0,k}$, are known. It is given by

$$\begin{cases} c_{j,k} = \sqrt{2}\sum_{n\in\mathbb{Z}} u_n c_{j-1,2k+n} \\ d_{j,k} = \sqrt{2}\sum_{n\in\mathbb{Z}} v_n c_{j-1,2k+n} \end{cases}$$

where u_n, v_n are equal the coordinates of $\phi_{1,0}, \psi_{1,0}$ in the basis $\phi_{0,n}$:

$$\begin{cases} \phi(t/2) = \sum_{n\in\mathbb{Z}} u_n \phi(t-n) \\ \psi(t/2) = \sum_{n\in\mathbb{Z}} v_n \phi(t-n) \end{cases}$$

IDWT is the inverse transformation

NUMBER OF COEFFICIENTS Assume we are given only a finite number of coefficients $c_{0,k}$. At every octave, the number of scaling and wavelet coefficients is divided by 2. If we have $2^J n$ coefficients at step 0, then at octave j, $1 \leq j \leq J$, we have $2^{J-j}n$ coefficients. The complexity of the pyramidal algorithm is $O(N)$, where N is the total number of coefficients computed.

ATOMS A multi-resolution analysis at octaves 0 to J can be written as

$$C_0(t) = \sum_{k\in\mathbb{Z}} c_{J,k}\phi_{j,k}(t) + \sum_{j=1}^{J}\sum_{k\in\mathbb{Z}} d_{j,k}\psi_{j,k}(t)$$

The individual terms in this summation are called *atoms*. The $J+1$ sequences $(c_{J,k})_{k\in\mathbb{Z}}$ and $(d_{j,k})_{k\in\mathbb{Z}}$ for $j = 1$ to J are called *crystals*.

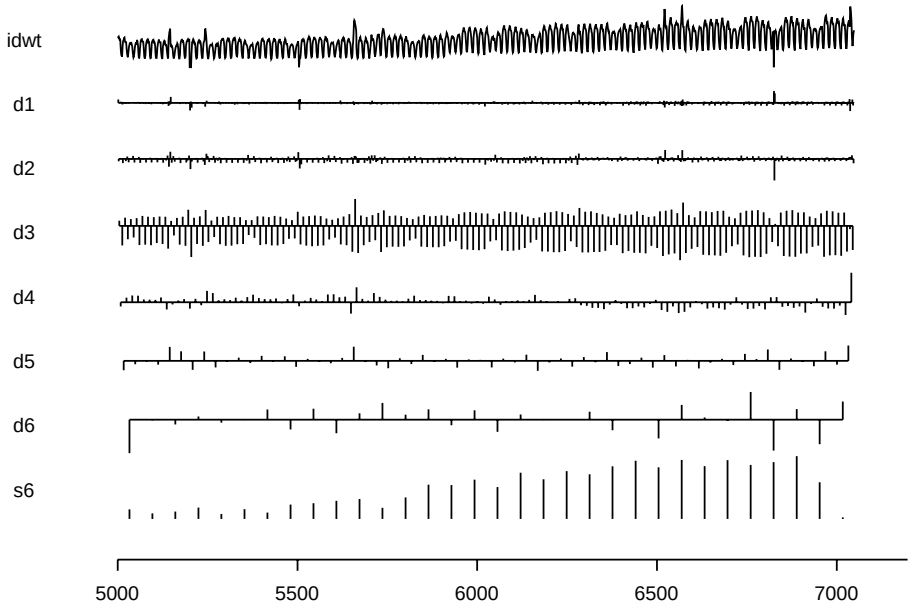
ENERGY AT OCTAVE j The mapping from a vector to its coordinates in an orthonormal basis is an isometry, thus

$$\begin{cases} \|C_J\|^2 = \sum_{k\in\mathbb{Z}} c_{J,k}^2 \\ \|D_j\|^2 = \sum_{k\in\mathbb{Z}} d_{j,k}^2 \end{cases}$$

and we can re-write the conservation of energy as

$$\int_{t\in\mathbb{R}} C_0(t)^2 dt = \sum_{k\in\mathbb{Z}} c_{J,k}^2 + \sum_{j=1}^{J}\sum_{k\in\mathbb{Z}} d_{j,k}^2$$

and similarly if we start at octave $J_0 < 0$ instead of 0.



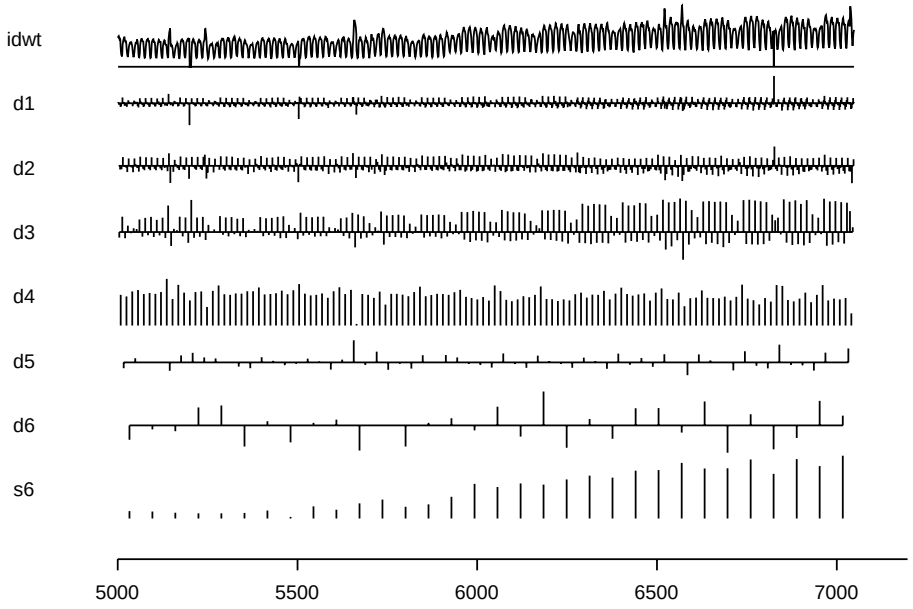


Figure 13.3: Scaling and Wavelet coefficients for the example of Figure 13.1 with wavelet = Daubechies 6 (first graph) and wavelet = Haar (second graph). The variability of the time series is mainly at octave 3, (first graph) or 3 and 4 (second graph).

VANISHING MOMENTS The number of *vanishing moments* is the largest M such that

$$\int_{t\in\mathbb{R}} t^m \psi(t)dt = 0 \text{ for } m = 0, 1, ..., M-1$$

If $f(t)$ is a polynomial of degree $\leq M-1$, the wavelet coefficients are 0. If $f(t)$ is well approximated by the first M terms of its Taylor expansion, the wavelet coefficients are small.

13.4 TIME SERIES

Consider x_n, $n \in \mathbb{Z}$. Wavelets apply to functions of continuous time – this shows up in particular for the computation of coefficients as integrals. Thanks to the pyramidal algorithm, the plays a role only for the computation of the initial coefficients $c_{0,k}$.

DETERMINISTIC CASE If x_n is non random, all we need is a mapping $x_n \to X(t)$ that preserves the norms, i.e. $\|x\|^2 = \|X\|^2$ (the mapping is an *isometry*). A generic form is

$$X(t) = \sum_{n\in\mathbb{Z}} x_n g(t-n)$$

A simple function is $g_0(t) = 1_{\{0\leq t<1\}}$. A better one is $g_1(t) = \text{sinc}(t) = \frac{sin\pi t}{\pi t}$. By Shannon's sampling theorem, the resulting $X(t)$ is the band-limited process which can be perfectly reconstructed from the samples x_n. In either case we have $X(n) = x_n$ for $n \in \mathbb{Z}$.

The initial coefficients can be computed as follows.

$$\begin{aligned} c_{0,k} &= \sum_{n\in\mathbb{Z}} x_n \int_{t\in\mathbb{R}} \phi(t-k)g(t-n)dt = \sum_{n\in\mathbb{Z}} x_n \int_{t\in\mathbb{R}} \phi(t-k+n)g(t)dt \\ &= \sum_{n\in\mathbb{Z}} x_n I_{k-n} = (x * I)_k \end{aligned} \tag{13.3}$$

with

$$I_k = \int_{t\in\mathbb{R}} \phi(t-k)g(t)dt$$

Thus we should apply a convolution filter to the time series before taking its wavelet transform. In practive, there is only a small number of coefficients I_k that are non 0 or non-negligible.

Some packages do not do the correct initialization; instead, they initialize multi-resolution analysis with $c_{0,k} = x_k$. This is equivalent to applying the wavelet analysis to $Y(t) = \sum_{k\in\mathbb{Z}} \phi(t-k)x_k$. This mapping of x_n to Y_n is an isometry because $\phi_{0,n}$ is an orthornormal system, however, it does not seem natural. This is because the time series which is analyzed in reality is $y_n = \sum_{k\in\mathbb{Z}} x_k \phi(n-k)$, which, in general, is not equal to x_n. This may introduce some distortion into the coefficients at lower octaves.

QUESTION 13.4.1. *Compute y_n for the Haar wavelet.* [7]

The mapping from the time series x_n to the coefficients so obtained is called discrete time wavelet transform. It is an orthogonal transformation, and is used by some authors in replacement of the true DWT.

[7] $Y(t)$ is the natural extrapolation $\sum_{n\in\mathbb{Z}} x_n 1_{\{n\leq t<n+1\}}$ and $y_n = x_n$. There is no distortion in this specific case.

IMPACT OF FIRST OCTAVE J_0. In multi-resolution analysis, we often limit ourselves in practice to $J_0 = 0$, i.e. we assume that $C_0(t)$, the projection on V_0 of $X(t)$, is close to $X(t)$. This is particularly true if we take $X(t) = \sum_{n\in\mathbb{Z}} x_n \mathrm{sinc}(t-n)$, due to the spectral properties of wavelets; it is more accurate for wavelets with higher degree of regularity.

For the time series in Figure 13.1 and Figure 13.3, the error is negligible (of the order of computation errors).

QUESTION 13.4.2. *How can we verify whether this approximation holds ?* [8]

In the rare cases where this approximation is not valid, this does not impact the values of coefficients obtained with the pyramidal algorithm. It simply means that the coefficients for negative octaves are not negligible and should be computed as well. This can be done with the pyramidal algorithm, starting with $c_{J_0,k}$ instead of $c_{0,k}$. It is equivalent to replacing the original time series with the up-sampled time series

$$x^*_n = 2^{J_0/2} X(2^{J_0} n)$$

where $X(t)$ is the continuous time interpolation of x_n (remember that J_0 is negative; we have $x^*_{2^{-J_0}n} = x_n$.

STOCHASTIC CASE Assume now that x_n is a random sequence, and we are interested in second order properties of x_t. Then we should use only the mapping

$$X(t) = \sum_{n\in\mathbb{Z}} x_n \mathrm{sinc}(t-n)$$

This definition is shown to be valid in the mean square sense in [Veitch00-Init], provided that the father wavelet ϕ is bounded, which is the case for the ones we use. It can be shown that the second-order properties of $X(t)$ and x_n are the same. We use this property for analyzing the auto-covariance functions of time series.

PADDING AND BOUNDARY CONDITIONS See S+Wavelet tutorial.

USEFUL S-PLUS COMMANDS

- `make.signal`: out of a data frame, make an object that wavelet functions can use
- `ca <- dwt(ic)` perform DWT on initial coefficients; returns scaling and wavelet coefficients; `plot(ca)` displays the scaling and wavelet coefficients
- `eda(ca)` plots distribution of energy and other summary data
- `ca <- mrd(ic)` multi-resolution analysis (decomposition) (returns the coarse approximation and details); `ca <- mra(ic)` multi-resolution approximation (returns the successively coarser approximations; `plot(ca)` plots the results.
- `reconstruct`: returns the time series $C_0[t]$; `top.atoms`: returns the largest coefficients; `decompose` returns the atoms.
- `dwt.matrix` the discrete time wavelet transform

[8] By plotting $x_n - C_0(n)$, where C_0 is the sum of crystals, or by comparing the sum of squares of x_n with that of the coefficients.

CHAPTER 14

TABLES AND DISTRIBUTIONS

A good list of distributions can be found in [McLaughlin97], a compendium of distributions by Michael P. McLaughlin that is publicly available (see web site for more information). We add here the tables and concepts that could not be found there.

14.1 CATALOG OF DISTRIBUTIONS

We give a list of commonly used distributions and the notation for their cdf. For more details see [McLaughlin97].

14.1.1 *Binomial*

$B_{n,p}$

14.1.2 *Multinomial* $M_{n,\vec{p}}$

A sequence $N_1, ..., N_k$ in $\mathbb{N}^k$ has the multinomial distribution $M_{n,\vec{p}}$ if and only if

$$\begin{cases} \sum_{i=1}^{k} N_i = n \\ \mathbb{P}\{N_1 = n_1, ..., N_k = n_k\} = \begin{pmatrix} n! \\ n_1!...n_k! \end{pmatrix} p_1^{n_1}...p_k^{n_k} \end{cases} \tag{14.1}$$

Assume n random variables X_j are iid, take values in the finite set $\{1, 2, ..., k\}$ and $\mathbb{P}(X_j = i) = p_i$. Let $N_i = \sum_{j=1}^{n} 1_{\{X_j = i\}}$ (number of observations that are equal to i). Then the distribution of the vector $(N_1, ..., N_k)$ is $M_{n,\vec{p}}$.

14.1.3 *Geometric*

Geom(θ)

14.1.4 *Normal*

also called *gaussian* N_{μ,σ^2}

14.1.5 *Chi-Square*

χ^2_n is the distribution of the sum of the squares of n independent random variables with distribution $N_{0,1}$. Expectation: n; Variance: $2n$

14.1.6 *Fisher*

$F_{m,n}$ is the distribution of

$$Z = \frac{X/m}{Y/n}$$

where $X \sim \chi^2_m$, $Y \sim \chi^2_n$ and X and Y are independent.
If $F \sim F_{m,n}$ then $\frac{1}{F} \sim F_{n,m}$, thus if $F_{m,n}(\eta) = \gamma$ then $F_{n,m}(1/\eta) = \gamma$
If $T \sim t_n$ then $T^2 \sim F_1, n$.

14.1.7 *Student*

t_n is the distribution of

$$Z = \frac{X}{\sqrt{Y/n}}$$

where $X \sim N_{0,1}$, $Y \sim \chi^2_n$ and X and Y are independent.

14.2 CONFIDENCE INTERVALS FOR QUANTILES

The following tables can be used to determine confidence intervals for quantiles (including median), as follows (see Theorem 2.2.1 for more details).

For a sample of n iid data points $x_1, ..., x_n$, the tables give a confidence interval at the confidence level $\gamma = 0.95$ or 0.99 for the q-quantile with $q = 0.5$ (median), $q = 0.75$ (quartile) and $q = 0.95$. The confidence interval is $[x_{(j)}, x_{(k)}]$, where $x_{(j)}$ is the jth data point in increasing order.

The confidence intervals for $q = 0.05$ and $q = 0.25$ are not given in the tables. They can be deduced by the following rule. Let $[x_{(j)}, x_{(k)}]$ be the confidence interval for the q-quantile given by the table. A confidence interval for the $1 - q$-quantile is $[x_{(j')}, x_{(k')}]$ with

$$j' = n + 1 - k$$
$$k' = n + 1 - j$$

For example, with $n = 50$, a confidence interval for the third quartile ($q = 0.75$) at confidence level 0.99 is $[x_{(29)}, x_{(45)}]$, thus a confidence interval for the first quartile ($q = 0.25$) at confidence level 0.99 is $[x_{(6)}, x_{(22)}]$.

The tables give p, the actual confidence level obtained (it is not possible to obtain a confidence interval at exactly the required confidence levels). For small values of n no confidence interval is possible. For large n, an approximate value is given.

n	j	k	p
$n \leq 5$: no confidence interval possible.			
6	1	6	0.969
7	1	7	0.984
8	1	7	0.961
9	2	8	0.961
10	2	9	0.979
11	2	10	0.988
12	3	10	0.961
13	3	11	0.978
14	3	11	0.965
15	4	12	0.965
16	4	12	0.951
17	5	13	0.951
18	5	14	0.969
19	5	15	0.981
20	6	15	0.959
21	6	16	0.973
22	6	16	0.965
23	7	17	0.965
24	7	17	0.957
25	8	18	0.957
26	8	19	0.971
27	8	20	0.981
28	9	20	0.964
29	9	21	0.976
30	10	21	0.957
31	10	22	0.971
32	10	22	0.965
33	11	23	0.965
34	11	23	0.959
35	12	24	0.959
36	12	24	0.953
37	13	25	0.953
38	13	26	0.966
39	13	27	0.976
40	14	27	0.962
41	14	28	0.972
42	15	28	0.956
43	15	29	0.968
44	16	29	0.951
45	16	30	0.964
46	16	30	0.960
47	17	31	0.960
48	17	31	0.956
49	18	32	0.956
50	18	32	0.951
51	19	33	0.951
52	19	34	0.964
53	19	35	0.973
54	20	35	0.960
55	20	36	0.970
56	21	36	0.956
57	21	37	0.967
58	22	37	0.952
59	22	38	0.964
60	23	39	0.960
61	23	39	0.960
62	24	40	0.957
63	24	40	0.957
64	24	40	0.954
65	25	41	0.954
66	25	41	0.950
67	26	42	0.950
68	26	43	0.962
69	26	44	0.971
70	27	44	0.959
$n \geq 71$	$\approx \lfloor 0.50n - 0.980\sqrt{n} \rfloor$	$\approx \lceil 0.50n+1+ 0.980\sqrt{n} \rceil$	0.950

n	j	k	p
$n \leq 7$: no confidence interval possible.			
8	1	8	0.992
9	1	9	0.996
10	1	10	0.998
11	1	11	0.999
12	2	11	0.994
13	2	12	0.997
14	2	12	0.993
15	3	13	0.993
16	3	14	0.996
17	3	15	0.998
18	4	15	0.992
19	4	16	0.996
20	4	16	0.993
21	5	17	0.993
22	5	18	0.996
23	5	19	0.997
24	6	19	0.993
25	6	20	0.996
26	7	20	0.991
27	7	21	0.994
28	7	21	0.992
29	8	22	0.992
30	8	23	0.995
31	8	24	0.997
32	9	24	0.993
33	9	25	0.995
34	10	25	0.991
35	10	26	0.994
36	10	26	0.992
37	11	27	0.992
38	11	27	0.991
39	12	28	0.991
40	12	29	0.994
41	12	30	0.996
42	13	30	0.992
43	13	31	0.995
44	14	31	0.990
45	14	32	0.993
46	15	33	0.992
47	15	33	0.992
48	15	33	0.991
49	16	34	0.991
50	16	35	0.993
51	16	36	0.995
52	17	36	0.992
53	17	37	0.995
54	18	37	0.991
55	18	38	0.994
56	18	38	0.992
57	19	39	0.992
58	20	40	0.991
59	20	40	0.991
60	20	40	0.990
61	21	41	0.990
62	21	42	0.993
63	21	43	0.995
64	22	43	0.992
65	22	44	0.994
66	23	44	0.991
67	23	45	0.993
68	23	45	0.992
69	24	46	0.992
70	24	46	0.991
71	25	47	0.991
72	25	47	0.990
$n \geq 73$	$\approx \lfloor 0.50n - 1.288\sqrt{n} \rfloor$	$\approx \lceil 0.50n+1+ 1.288\sqrt{n} \rceil$	0.990

Table 14.1: Quantile $q = 50\%$, Confidence Levels $\gamma = 95\%$ (left) and 0.99% (right)

n	j	k	p
$n \leq 10$: no confidence interval possible.			
11	5	11	0.950
12	6	12	0.954
13	7	13	0.952
14	7	14	0.972
15	8	15	0.969
16	9	16	0.963
17	9	17	0.980
18	9	17	0.955
19	10	18	0.960
20	12	20	0.956
21	12	20	0.960
22	13	21	0.956
23	13	22	0.974
24	14	23	0.970
25	14	24	0.982
26	15	24	0.959
27	16	25	0.958
28	17	26	0.954
29	17	27	0.971
30	17	27	0.954
31	18	28	0.958
32	20	30	0.956
33	20	30	0.958
34	21	31	0.955
35	22	32	0.950
36	22	33	0.968
37	22	34	0.979
38	23	34	0.961
39	24	35	0.960
40	25	36	0.958
41	25	37	0.972
42	25	37	0.961
43	26	38	0.963
44	28	40	0.961
45	28	40	0.963
46	28	40	0.951
47	29	41	0.953
48	31	43	0.952
49	31	43	0.954
50	32	44	0.952
51	32	45	0.966
52	33	46	0.964
53	33	47	0.975
54	34	47	0.959
55	35	48	0.959
56	36	49	0.957
57	36	50	0.969
58	37	50	0.951
59	38	51	0.951
60	39	53	0.961
61	39	53	0.963
62	39	53	0.954
63	40	54	0.956
64	42	56	0.955
65	42	56	0.956
66	43	57	0.955
67	44	58	0.952
68	44	59	0.966
69	44	60	0.975
70	45	60	0.962
71	46	61	0.961
72	47	62	0.960
73	47	63	0.971
74	48	63	0.956
75	49	64	0.956
$n \geq 76$	$\approx \lfloor 0.75n - 0.849\sqrt{n} \rfloor$	$\approx \lceil 0.75n+1+ 0.849\sqrt{n} \rceil$	0.950

n	j	k	p
$n \leq 16$: no confidence interval possible.			
17	7	17	0.992
18	8	18	0.993
19	9	19	0.993
20	10	20	0.993
21	11	21	0.991
22	11	22	0.995
23	12	23	0.994
24	13	24	0.992
25	13	25	0.996
26	13	25	0.993
27	15	27	0.992
28	15	27	0.993
29	16	28	0.992
30	16	29	0.995
31	17	30	0.994
32	18	31	0.993
33	18	32	0.996
34	19	32	0.991
35	20	33	0.990
36	21	35	0.991
37	21	35	0.993
38	21	35	0.990
39	23	37	0.990
40	23	37	0.991
41	23	39	0.997
42	24	39	0.994
43	25	40	0.993
44	26	41	0.992
45	26	42	0.995
46	27	42	0.990
47	28	44	0.993
48	29	45	0.991
49	29	45	0.993
50	29	45	0.990
51	31	47	0.990
52	31	47	0.991
53	31	49	0.996
54	32	49	0.993
55	33	50	0.993
56	34	51	0.992
57	34	52	0.995
58	35	52	0.991
59	36	53	0.990
60	37	55	0.992
61	37	55	0.993
62	37	55	0.991
63	39	57	0.991
64	39	57	0.991
65	40	58	0.991
66	41	59	0.990
67	41	60	0.993
68	42	61	0.993
69	42	62	0.995
70	43	62	0.992
71	44	63	0.991
72	45	64	0.991
73	45	65	0.994
74	45	65	0.992
75	47	67	0.992
76	48	68	0.991
77	48	68	0.992
78	48	68	0.991
79	50	70	0.991
80	50	70	0.991
81	51	71	0.990
$n \geq 82$	$\approx \lfloor 0.75n - 1.115\sqrt{n} \rfloor$	$\approx \lceil 0.75n+1+ 1.115\sqrt{n} \rceil$	0.990

Table 14.2: Quantile $q = 75\%$, Confidence Levels $\gamma = 95\%$ (left) and 0.99% (right)

n	j	k	p
$n \leq 58$: no confidence interval possible.			
59	50	59	0.951
60	52	60	0.951
61	53	61	0.953
62	54	62	0.955
63	55	63	0.957
64	56	64	0.958
65	57	65	0.959
66	58	66	0.961
67	59	67	0.962
68	60	68	0.963
69	61	69	0.964
70	62	70	0.964
71	63	71	0.965
72	64	72	0.965
73	65	73	0.966
74	66	74	0.966
75	67	75	0.966
76	68	76	0.966
77	69	77	0.966
78	70	78	0.966
79	71	79	0.966
80	72	80	0.965
81	73	81	0.964
82	74	82	0.964
83	75	83	0.963
84	76	84	0.962
85	77	85	0.961
86	78	86	0.960
87	79	87	0.959
88	80	88	0.957
89	81	89	0.956
90	82	90	0.954
91	83	91	0.952
92	84	92	0.950
93	84	93	0.974
94	85	94	0.973
95	86	95	0.972
96	87	96	0.971
97	88	97	0.970
98	89	98	0.969
99	90	99	0.967
100	91	100	0.966
101	91	100	0.952
102	92	101	0.953
103	93	102	0.953
104	94	103	0.954
105	95	104	0.954
106	96	105	0.954
107	97	106	0.954
108	98	107	0.954
109	99	108	0.954
110	100	109	0.954
111	101	110	0.954
112	102	111	0.953
113	103	112	0.953
114	104	113	0.952
115	105	114	0.951
116	106	115	0.950
117	107	117	0.965
118	108	118	0.963
119	109	119	0.961
120	110	120	0.959
121	110	120	0.967
122	111	121	0.966
123	112	122	0.966
$n \geq 124$	$\approx \lfloor 0.95n - 0.427\sqrt{n} \rfloor$	$\approx \lceil 0.95n+1+ 0.427\sqrt{n} \rceil$	0.950

n	j	k	p
$n \leq 89$: no confidence interval possible.			
90	76	90	0.990
91	79	91	0.990
92	80	92	0.990
93	81	93	0.991
94	82	94	0.991
95	83	95	0.991
96	84	96	0.992
97	85	97	0.992
98	86	98	0.992
99	87	99	0.992
100	88	100	0.993
101	89	101	0.993
102	90	102	0.993
103	91	103	0.993
104	92	104	0.993
105	93	105	0.993
106	94	106	0.993
107	95	107	0.993
108	96	108	0.993
109	97	109	0.993
110	98	110	0.993
111	99	111	0.993
112	100	112	0.993
113	101	113	0.993
114	102	114	0.992
115	103	115	0.992
116	104	116	0.992
117	105	117	0.992
118	106	118	0.991
119	107	119	0.991
120	108	120	0.991
121	109	121	0.990
122	109	122	0.995
123	110	123	0.995
124	111	124	0.995
125	112	125	0.994
126	113	126	0.994
127	114	127	0.994
128	115	128	0.994
129	116	129	0.993
130	117	130	0.993
131	118	131	0.993
132	119	132	0.992
133	120	133	0.992
134	121	134	0.992
135	122	135	0.991
136	123	136	0.991
137	124	137	0.990
138	124	138	0.995
139	125	139	0.995
140	126	140	0.995
141	127	141	0.994
142	127	141	0.992
143	128	142	0.992
144	129	143	0.992
145	130	144	0.992
146	131	145	0.992
147	133	147	0.992
148	134	148	0.992
149	135	149	0.992
150	136	150	0.991
151	137	151	0.991
152	138	152	0.990
153	138	152	0.992
154	139	153	0.992
$n \geq 155$	$\approx \lfloor 0.95n - 0.561\sqrt{n} \rfloor$	$\approx \lceil 0.95n+1+ 0.561\sqrt{n} \rceil$	0.990

Table 14.3: Quantile $q = 95\%$, Confidence Levels $\gamma = 95\%$ (left) and 0.99% (right)

Bibliography

[1] D. Bertsekas and R. Gallager. *Data Networks*. Prentice-Hall, Englewood Cliffs, New Jersey, 1992.

[2] F.P. Kelly, A. K. Maulloo, and D.K.H. Tan. Rate control for commmunication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49, 1998.

[3] L. Massoulie and J. Roberts. Fairness and quality of service for elastic traffic. In *Proceedings of Infocom*, 1999.

[4] Mazumdar R., Mason L.G., and Douligeris C. Fairness in network optimal flow control: Optimality of product form. *IEEE Transactions on Communication*, 39:775–782, 1991.

[5] Peter Whittle. *Optimization under Constraints*. Wiley and Sons, Books on Demand, `www.umi.com`, 1971.

[6] Bernard Ycart. *Modèles et algorithmes markoviens*, volume 39. Springer Verlag, 2002.

In this temporary version, further citations are found on the web page of the course under “Documents”