Applied
Probability and Stochastic Processes

Lecture Notes for 577/578 Class

Włodzimierz Bryc
Department of Mathematics
University of Cincinnati
P. O. Box 210025
Cincinnati, OH 45221-0025
E-mail: `bryc@ucbeh.san.uc.edu`

Created: October 22, 1995
Printed: March 21, 1996

CINCINNATI FREE TEXTS

Contents

Course description

This course is aimed at students in applied fields and assumes a prerequisite of calculus. The goal is a working knowledge of the concepts and uses of modern probability theory. A significant part of such a “working knowledge” in modern applications of mathematics is computer-dependent.

The course contains mathematical problems and computer exercises. Students will be expected to write and execute programs pertinent to the material of the course. No programming experience is necessary or assumed. But the willingness to accept a computer as a tool is a requirement.

For novices, BASIC programming language, (QBASIC in DOS, **Visual Basic** in **Windows**, or BASIC on Macintosh) is recommended. BASIC is perhaps the easiest programming language to learn, and the first programming language is always the hardest to pick.

Programs in QBASIC 4.5 on IBM-compatible PC and, to a lesser extend, programs on Texas Instrument Programmable calculator **TI-85**, and **Windows**$_{TM}$ programs in Microsoft **Visual Basic 3.0** are supported. This means that I will attempt to answer technical questions and provide examples. Other programming languages (`SAS, C, C++, Fortran, Cobol, Assembler, Mathematica, LISP, TeX(!), Excel,` etc.) can be used, but I will not be able to help with the technical issues.

Contents of the course (subject to change)

577 Basic elements of probability. Poisson, geometric, binomial, normal, exponential distributions. Simulations. Conditioning, characterizations.

Moment generating functions, limit theorems, characteristic functions. Stochastic processes: random walks, Markov sequences, the Poisson process.

578 Time dependent and stochastic processes: Markov processes, branching processes. Modeling

Multivariate normal distribution. Gaussian processes, white noise. Conditional expectations. Fourier expansions, time series.

Supporting materials

This text is available through Internet[1] in PostScript, or DVI. A number of other math related resources can be found on WWW[2]. Also available are supporting BASIC[3] program files. Support for `Pascal` is anticipated in the future.

Auxiliary textbooks:

- W. Feller, *An Introduction to Probability Theory*, Vol. I, Wiley 1967. Vol II, Wiley, New York 1966

[1] http://math.uc.edu/b̃rycw/probab/books/

[2] http://archives.math.utk.edu/tutorials.html

[3] http://math.uc.edu/b̃rycw/probab/basic.htm

Volume I is an excellent introduction to elementary and not-that-elementary probability. Volume II is advanced.

- W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical recipes in C*, Cambridge University Press, New York
  A reference for numerical methods: C-language version.

- J. C. Sprott *Numerical recipes Routines and Examples in BASIC*, Cambridge University Press, New York 1992
  A reference for numerical methods: routines in QuickBasic 4.5 version.

- H. M. Taylor & S. Karlin, *An introduction to stochastic modeling*, Acad. Press, Boston 1984
  Markov chains with many examples/models, Branching processes, Queueing systems.

- L. Breiman, *Probability and Stochastic Processes: with a view towards applications*, Houghton Mifflin, Boston 1969.
  includes Markov chains and spectral theory of stationary processes.

- R. E. Barlow & F. Proschan, *Mathematical Theory of Reliability*, SIAM series in applied math, Wiley, New York 1965.
  Advanced compendium of reliability theory methods (mid-sixties).

- S. Biswas, *Applied Stochastic Processes*, Wiley, New York 1995
  Techniques of interest in population dynamics, epidemiology, clinical drug trials, fertility and mortality analysis.

- J. Higgins & S. Keller-McNulty, *Concepts in Probability and Stochastic Modeling* Duxbury Press 1995
  Covers traditional material; computer simulations complement theory.

- T. Harris, *The Theory of Branching Processes* Reprinted: Dover, 1989.
  A classic on Branching processes.

- H. C. Tjims, *Stochastic models. An algorithmic approach*, Wiley, Chichester, 1994.
  Renewal processes, reliability, inventory models, queuieing models.

Conventions

Exercises

The text has three types of “practice questions”, marked as **Problems, Exercises**, and **Projects**. Examples vary the most and could be solved in class by an instructor; Exercises are intended primarily for computer-assisted analysis; Problems are of more mathematical nature. Projects are longer, and frequently open-ended. Exercises, Projects, and Problems are numbered consecutively within chapters; for instance Exercise 10.2 follows Problem 10.1, meaning that there is no Exercise 10.1.

Programming

The text refers to BASIC instructions with the convention that BASIC key-words are capitalized, the names of variables, functions, and the SUBs are mixed case like `ThisExample`. Program listings are typeset in a special "computer" font to distinguish them from the rest of the text.

Copyright and License for 1996 version

Part I

Probability

Chapter 1

Random phenomena

> Definition of a Tree: *A tree is a woody plant with erect perennial trunk of at least 3.5 inches (7.5 centimeters) in diameter at breast height ($4\frac{1}{2}$ feet or 1.3 meters), a definitely formed crown of foliage, and a height of at least 13 feet (4 meters).*
> The Auborn Society Field Guide to North American Trees.

This chapter introduces fundamental concepts of probability theory; events, and their chances. For the readers who are familiar with elementary probability it may be refreshing to see the computer used for counting elementary events, and randomization used to solve a deterministic optimization problem.

The questions are

- What is "probability"?
- How do we evaluate probabilities in real-life situations?
- What is the computer good for?

1.1 Mathematical models, and stochastic models

Every theory has its successes, and its limitations. These notes are about the successes of probability theory. But it doesn't hurt to explain in non-technical language some of its limitations up front. This way the reader can understand the basic premise before investing considerable time.

To begin with, we start with a truism. Real world is complicated, often to a larger degree than scientists will readily admit. Most real phenomena have multi-aspect form, and can be approached in multiple ways. Various questions can be asked. Even seemingly the same question can be answered on many incompatible levels. For instance, the generic question about dinosaur extinction has the following variants.

- Why did Dino the dinosaur die? Was she bitten by a poisonous rat-like creature? Hit by a meteorite? Froze to death?
- Why did the crocodiles survive to our times, and tyranosaurus rex didn't?

- What was the cause of the dinosaur extinction?
- Was the dinosaur extinction an accident, or did it have to happen? (This way, or the other).
- Do all species die out?

Theses questions are ordered from the most individual level to the most abstract. The reader should be aware that probability theory, and stochastic modelling deal only with the most abstract levels of the question. Thus, a stochastic model may perhaps shed some light whether dinosaurs had to go extinct, or whether mammals will go extinct, but it wouldn't go into details of which comet had to be responsible for dinosaurs, or which is the one that will be responsible for the extinction of mammals.

It isn't our contention that individual questions have no merit. They do, and perhaps they are as important as the general theories. For example, a detective investigating the cause of a mysterious death of a young woman, will have little interest in the "abstract statistical fact" that all humans eventually die anyhow. But individual questions are as many as the trees in the forest, and we don't want to overlook the forest, either.

Probabilistic models deal with general laws, not individual histories. Their predictions are on the same level, too. To come back to our motivating example, even if a stochastic model did predict the extinction of dinosaurs (eventually), it would not say that it had to happen at the time when it actually happened, some 60 million years ago. And the more concrete a question is posed, say if we want to know when Dino the dinosaur died, the less can be extracted from the stochastic model.

On the other hand, many concepts of modern science are define in statistical, or probabilistic sense.(If you think this isn't true, ask yourself how many trees do make a forest.) Such concepts are best studied from the probabilistic perspective. An extreme view is to consider everything random, deterministic models being just approximations that work for small levels of noise.

1.2 Events and their chances

Suppose Ω is a set, called the probability, or the sample space. We interpret Ω as a mathematical model listing all relevant outcomes of an experiment.

Let $\mathcal{M}$ be a σ-field of its subsets, called the events. Events $A, B \in \mathcal{M}$ model sentences about the outcomes of the experiment to which we want to assign probabilities. Under this interpretation, the union $A \cup B$ of events corresponds to the *alternative*, the intersection $A \cap B$ corresponds to the *conjunction* of sentences, and the complement A' corresponds to the *negation* of a sentence. For $A, B \in \mathcal{M}$, $A \setminus B := A \cap B'$ denotes the set-theoretical difference.

For an event $A \in \mathcal{M}$ the probability $\Pr(A)$ is a number interpreted as the degree of certainty (in unique experiments), or asymptotic frequency of A (in repeated experiments). Probability $\Pr(A)$ is assigned to all events $A \in \mathcal{M}$, but it must satisfy certain requirements (axioms). A set function $\Pr(\cdot)$ is a probability measure on $(\Omega, \mathcal{M})$, if it fulfills the following conditions:

1. $0 \leq \Pr(A) \leq 1$

2. $\Pr(\Omega) = 1$

3. For disjoint[1] $\Pr(A \cup B) = \Pr(A) + \Pr(B)$.

4. If A_n are such that $\bigcap_{n \geq 1} A_n = \emptyset$ and $A_1 \supset A_2 \supset \ldots \supset A_n \supset A_{n+1} \supset \ldots$ are *decreasing events*, then
$$\Pr(A_n) \to 0. \tag{1.1}$$

Probability axioms do not determine the probabilities in a unique way. The axioms provide only minimal consistency requirements, which are satisfied by many different models.

1.2.1 Uniform Probability

For finite set Ω let
$$\Pr(A) = \frac{\#A}{\#\Omega}. \tag{1.2}$$
This captures the intuition that the probability of an event is proportional to the number of ways that the event might occur.

The uniform assignment of probability involves counting. For small sample spaces this can be accomplished by examining all cases. Moderate size sample spaces can be inspected by a computer program. Counting arbitrary large spaces is the domain of *combinatorics*. It involves combinations, permutations, generating functions, combinatorial identities, etc. Short review in SectionB.2 recalls the most elementary counting techniques.

Problem 1.1 *Three identical dice are tossed. What is the probability of two of a kind?*

The following BASIC program inspects all outcomes when five dice are rolled, and counts how many are “four of a kind”.

```
PROGRAM yahtzee.bas
’


’declare function and  variables
DECLARE FUNCTION CountEq! (a!, b!, c!, d!, e!)

’prepare screen
CLS
PRINT "Listing four of a kind outcomes in Yahtzee..."

’*** go through all cases
FOR a = 1 TO 6: FOR b = 1 TO 6: FOR c = 1 TO 6: FOR d = 1 TO 6: FOR e = 1 TO 6
   IF CountEq(a + 0, b + 0, c + 0, d + 0, e + 0) = 4 THEN
       PRINT a; b; c; d; e; : ct = ct + 1
       IF ct MOD 5 = 0 THEN PRINT :  ELSE PRINT "|";
   END IF
NEXT: NEXT: NEXT: NEXT: NEXT

’print result
```

[1] *Disjoint*, or *exclusive* events $A, B \subset \Omega$ are such that $A \cap B = \emptyset$ is empty.

```
PRINT
PRINT "Total of "; ct; " four of a kind."

FUNCTION CountEq (a, b, c, d, e)
'*** count how many of five numbers are the same
DIM x(5)
x(1) = a
x(2) = b
x(3) = c
x(4) = d
x(5) = e
max = 0
FOR j = 1 TO 5
    ck = 0
    FOR k = 1 TO 5
        IF x(j) = x(k) THEN ck = ck + 1
    NEXT k
    ck = -ck
    IF ck > max THEN max = ck
NEXT j
'assign value to function
CountEq = max
'

END FUNCTION
```

Here is a portion of its output:

```
4 4 4 2 4 | 4 4 4 3 4 | 4 4 4 4 1 | 4 4 4 4 2 | 4 4 4 4 3
4 4 4 4 5 | 4 4 4 4 6 | 4 4 4 5 4 | 4 4 4 6 4 | 4 4 5 4 4
4 4 6 4 4 | 4 5 4 4 4 | 4 5 5 5 5 | 4 6 4 4 4 | 4 6 6 6 6
5 1 1 1 1 | 5 1 5 5 5 | 5 2 2 2 2 | 5 2 5 5 5 | 5 3 3 3 3
5 3 5 5 5 | 5 4 4 4 4 | 5 4 5 5 5 | 5 5 1 5 5 | 5 5 2 5 5
5 5 3 5 5 | 5 5 4 5 5 | 5 5 5 1 5 | 5 5 5 2 5 | 5 5 5 3 5
5 5 5 4 5 | 5 5 5 5 1 | 5 5 5 5 2 | 5 5 5 5 3 | 5 5 5 5 4
5 5 5 5 6 | 5 5 5 6 5 | 5 5 6 5 5 | 5 6 5 5 5 | 5 6 6 6 6
6 1 1 1 1 | 6 1 6 6 6 | 6 2 2 2 2 | 6 2 6 6 6 | 6 3 3 3 3
6 3 6 6 6 | 6 4 4 4 4 | 6 4 6 6 6 | 6 5 5 5 5 | 6 5 6 6 6
6 6 1 6 6 | 6 6 2 6 6 | 6 6 3 6 6 | 6 6 4 6 6 | 6 6 5 6 6
6 6 6 1 6 | 6 6 6 2 6 | 6 6 6 3 6 | 6 6 6 4 6 | 6 6 6 5 6
6 6 6 6 1 | 6 6 6 6 2 | 6 6 6 6 3 | 6 6 6 6 4 | 6 6 6 6 5
Total of 150 four of a kind.
```

The program is written explicitly for tossing five dice, and you may want to modify it to answer similar question for any number of dice, and an arbitrary k-of-a-kind question.

Exercise 1.2 *Run and time* `YAHTZEE.BAS`*. Then estimate how long a similar problem would run if the question involved tossing 15 fair dice. The answer depends on your computer, and the software. Both* `Pascal` *and* `C`*-programs seem to run on my computer about 15 times faster than the (compiled)* `QuickBasic`*.* $\boxed{\text{ANS: Running time would take years!}}$

Exercise 1.2 shows the power of old-fashioned pencil-and-paper calculation.

Problem 1.3 *Continuing Problem 1.1, suppose now n identical dice are tossed. What is the probability of $n-1$ of a kind?* $\boxed{\text{ANS: } \frac{5n}{6^{n-1}}}$.

1.2.2 Geometric Probability

For bounded subsets $\Omega \subset \mathbb{R}^d$, put

$$\Pr(A) = \frac{|A|}{|\Omega|}. \tag{1.3}$$

This captures the intuition that the probability of hitting a target is proportional to the the size of the target.

Geometric probability usually involves multivariate integrals.

Example 1.1 *A point is selected from the 32 cm wide circular dartboard. The probability that it lies within the 8cm wide bullseye is* $\frac{1}{16}$.

Example 1.2 *A needle of length* $2\ell < 2$ *is thrown onto a paper ruled every 2 inches. The probability that the needle crosses a line is* $2\ell/\pi$.

Analysis of Example 1.2 is available on WWW[2].

Exercise 1.4 *Test by experiment (or simulation on the computer) if the above two examples give correct answers. (Before writing a program, you may want to read Section 1.4 first.)*

Example 1.3 *Two drivers arrive at an intersection between 8:00 and 8:01 every day. If they arrive within 15 seconds of each other, both cars have to stop at the stop-sign. How often do the drivers pass through the intersection without stopping?*

Project 1.5 *Continuing Example 1.3: What if there are three cars in this neighborhood? Four? How does the probability change with the number of cars? At what number of users a stop light should be installed?*

1.3 Elementary probability models

1.3.1 Consequences of axioms

Here are some useful formulas that are easy to check with the help of the *Venn diagrams.* For all events A, B

$$\Pr(A \cup B) = \Pr(A) + \Pr(B) - \Pr(A \cap B) \tag{1.4}$$

$$\Pr(A') = 1 - \Pr(A). \tag{1.5}$$

If $A \subset B$ then

$$\Pr(B \setminus A) = \Pr(B) - \Pr(A). \tag{1.6}$$

If A_n are pairwise disjoint events, then

$$\Pr(\bigcup_{n=1}^{\infty} A_n) = \sum_{n=1}^{\infty} \Pr(A_n). \tag{1.7}$$

[2] http://www.mste.uiuc.edu/reese/buffon/buffon.html

1.3.2 General discrete sample space

For a finite or countable set $\Omega \subset \mathbb{N}$ and a given summable sequence of non-negative numbers a_n put

$$\Pr(A) = \frac{\sum_{n\in A} a_n}{\sum_{n\in\Omega} a_n}. \tag{1.8}$$

In probability theory it is customary to denote $p_k = \frac{a_k}{\sum_{n\in\Omega} a_n}$ and rewrite (1.8) as $\Pr(A) = \sum_{n\in A} p_n$.

Formula (1.8) generalizes the uniform assignment (1.2), which corresponds to the choice of equal weights $a_k = 1$. At the same time it is more flexible[3] and applies also to infinite sample spaces.

Example 1.4 *Let $\Omega = \mathbb{N}$ and put $p_k = \frac{1}{2^k}$. Then the probability that an odd integer is selected is* $\Pr(Odd) = \sum_{j=0}^{\infty} 2^{-2j-1} = \frac{2}{3}$. *(Why? See (B.3))*

Table 1.1 list the most frequently encountered discrete probability assignments.

Name	Ω	Probabilities p_k
Binomial	$\{0,\dots,n\}$	$p_k = \binom{n}{k}p^k(1-p)^{n-k}$
Poisson	$\mathbb{Z}_+$	$p_k = e^{-\lambda}\frac{\lambda^k}{k!}$
Geometric	$\mathbb{N}$	$p_k = p(1-p)^{k-1}$
Equally likely outcomes	$\{x_1,\dots,x_k\}$	$p_k = \frac{1}{k}$

Table 1.1: Probability assignments for discrete sample spaces.

Problem 1.6 *For each of the choices of numbers p_k in Table 1.1 verify that (1.8) indeed defines a probability measure.*

The reasons behind the particular choices of the expressions for p_k in Table 1.1 involve modeling.

1.3.3 General continuous sample space

There is no easily accessible general theory for infinite non-countable sample spaces. When $\Omega \subset \mathbb{R}^k$, the generalization of the geometric probability uses a non-negative density function $f(x_1, x_2, \dots, x_k)$

$$\Pr(A) = C \int_A f(x_1, x_2, \dots, x_k)\, dx_1 dx_2 \dots dx_k \tag{1.9}$$

For one-dimensional case $k = 1$, examples of densities are collected in Table 2.2 on page 24.

[3] The price for flexibility is that now we have to decide how to chose p_n.

1.4 Simulating discrete events

The most natural way to verify whether a mathematical model reflects reality is to compare theoretically computed probabilities with the corresponding empirical frequencies. Unfortunately, each part of this procedure is often time consuming, expensive, inconvenient, or impossible. Computer simulations are used as a substitute for both: they provide numerical estimates of theoretical probabilities, and they are closer to the direct experiment.

The first step in a simulation is to decide how to generate "randomness" within a deterministic algorithm in a computer program. Programming languages, and even calculators, provide a method for generating consecutive numbers from the interval $(0,1)$ "at random". For instance, BASIC instruction[4] `PRINT RND(1)` returns different number at **every** use. We shall assume that the reader has access to a method, see Section 6.1, of generating *uniform* numbers from the unit interval $(0,1)$. These are called *pseudo-random numbers*, since the program usually begin the same "random" sequence at every run, unless special precautions[5] are taken.

Once a pseudo-random number from the interval $(0,1)$ is selected, an event that occurs with some known probability p can be simulated by verifying if { `RAND(1)<p`} occurs in the program. For instance, the number of heads in a toss of a $1,000$ fair coins is simulated by the following BASIC program.

```
PROGRAM tosscoin.bas
'


'Simulating 1000 tosses of a fair coins

H = 0
FOR n = 1 TO 1000  'main loop
        IF RND(1) < .5 THEN H = H + 1
NEXT n
'print final message
PRINT "Got "; H; " heads this time"
END
'
```

Here is its output: `Got 520 heads this time`

A simple method for simulating an outcome on a six-face die is to take the integer part of a re-scaled uniformly selected number `INT(1+6*RND(1))`. Can you use this to write a simulation of a roll of 5 dice? Such simulations are often used to evaluate probabilities empirically as a substitute for a real empirical study.

Exercise 1.7 *Write the simulation (as opposed to deterministic inspection of all sample points on page 3) to estimate how often the event "four of a kind" occurs in a roll of five dice.*

[4] Similar instruction on **TI-85** is `Display rand`.

[5] In BASIC, to avoid repetitions use instruction `RANDOMIZE TIMER` at the beginning of a simulation program.

Project 1.8 *Run the (modified) coin tossing program in a loop and to answer the following questions.*

- *In a 100 tosses of a coin, how often does less than 55 heads occur?*
- *In a 100 tosses of a coin, how often does less than 80 heads occur?*
- *Can you sketch the curve*[6] *that represents the probability of less than x coins in $n = 100$ tosses of a fair coin?*

Hint: *Move the main part of* `TOSSCOIN.BAS` *into a* `SUB`*, or a* `FUNCTION`*. This way you can easily use it without cluttering your program with irrelevant details. (See a generic template below.)*

More complicated objects are often of interest in simulations. For instance we may want to draw two cards from the deck of 52. One possible way to do it is to number the cards, and select two numbers a, b.

1. Select the first card `a=INT(RND(1)*52+1)` as a random integer between 1 and 52.
2. Select the second card `b=INT(RND(1)*52+1)` in the same way.
3. Compare a, b
   (a) If $a = b$ then repeat step 2
   (b) Otherwise, got two different cards $a \neq b$ at random

How efficient is this procedure?

Exercise 1.9 *How would you simulate on a computer a* random permutation*? A random subset?*

1.4.1 Generic simulation template

The purpose of simulation is to investigate the unknown values of parameters of interest. In the initial exercises you may want to simulate the events that you know how to compute probabilities of. The purpose of such exercises is to develop intuition about reliability of simulations.

In more advanced exercises you may want to estimate probabilities that aren't known. In such cases it is always a good idea to run simulations of various lengths and compare the results. In this section we briefly discuss how such a simulation can be organized in a way that promotes multiple uses of the same program.

The key is organizing the programs carefully into manageable blocks of small size. Modern BASIc is a structural programming language. The generic program to study the effects of the length of simulation on its output can be written as follows

[6] You need to find out how to handle graphic in BASIC. Otherwise, make a table of values instead.

```
' PROGRAM Generic.bas
'Generic Simulator
'Size is simulation size varied from Min=100 to Max=10000
For Size 100 to 10000 Step 100
Simulate(Size, Result)
Print "Simulation size="; Size ; "Output="; Result
Next Size
End
```

The actual simulation is performed by

```
SUB Simulate (SizeRequested, Result)
'Runs requested number of simulations and returns average
'Trial numbers consecutive simulations from 1 to SizeRequested
For Trial=1 to SizeRequested
SimulateOne(Score)
Result=Result+Score
Next Size
'Most simulations return averages of single trials
Result=Score/Size
End SUB
```

The actual modeling is performed in another SUB, which in the generic program we named `SimulateOne`. This SUB may be as simple as simulating a toss of a single coin

```
SUB SimulateONe(Outcome)
'simulate One occurrence, return numerical outcome
OutCome=0
if RND(1)<1/2 THEN Outcome=1
END SUB
```

Or it can be as complicated as we wish. The example below simulates a toss of five dice, and uses previously introduced function `CountEq(a,b,c,d,e)`. four-of-a-kind.

```
SUB SimulateONe(Outcome)
'simulate One occurrence, return numerical outcome
d1=int(RND(1)*6+1)
d2=int(RND(1)*6+1)
d3=int(RND(1)*6+1)
d4=int(RND(1)*6+1)
d5=int(RND(1)*6+1)
IF CountEq(d1,d2,d3,d4,d5)=4 THEN Outcome=1
END SUB
```

1.4.2 Blind search

Elementary probability when coupled with a fast computer is one of the simplest effective optimization method. The method is the *blind search* – a search for the best answer at random[7]. Pure blind search is usually simple to run, and therefore fast to realize. It often finds answers that are good enough for practical purposes, or at least can serve as the preliminary estimates. Various *ad hoc* modifications increase accuracy and are usually easy to implement, too.

[7] A related method is *brute force* – checking all possible cases.

Project 1.10 *Write a blind-search program to find the maximum of a function.*

- *Organize your program so that the function can easily be changed – but for now use the one you are quite familiar with, like* $100-(x-300)^2$*, or* $300e^{-(x-30)^2}\sin(200x)$*.*
- *If you are looking for more challenge, do the same for three variables. Write a blind-search program to find a maximum of a function like* $300e^{-(x-30)^2}\sin(200x+400y-z)+400e^{-(y-70)^2}\cos(400x+200y+z)$ *over the ball* $x^2+y^2+z^2\leq 1000$*.*
- *As a further complication, try to find a maximum of a function that has two local maxima, and the region isn't convex. (This is an almost hopeless task for gradient methods!)*

1.4.3 Application: Traveling salesman problem

The following program searches for the shortest way to pass through the first n cities[8] in the USA in alphabetic ordering.

Planning such a tour is easy by hand for 3-4 cities. For longer tours some help is needed. To check the performance of the blind search, you may want to know what the usual algorithms involve. A greedy method starts with the shortest distance, and then keeps going to the closest city not yet visited. Another heuristic method is to to select a pair of closest cities and accept the best (shortest) connection among those that do not complete a shorter loop, and do not introduce a spurious third connection to a city. Eventually, the disjoint pieces will form a path that often can be further improved upon inspection.

The program is longer but not at all sophisticated – it just selects paths at random. Notice that a solution to Exercise 1.9 – *how to generate a random permutation* – is given in one of the subprograms (which one?). The method for the latter is simple-minded – the algorithm attempts to place consecutive numbers at random spots until an empty spot is selected.

The following is the main part of the program. You can use it as a template in designing your own version of Blind search programs. The full code with SUBs is online[9] in `RANDTOUR.BAS`.

```
'****
CLS
'**** get number of cities (no choice which) from user
LOCATE 2, 1
INPUT ; "Shortest distance between how many cities?", n

'*** initialize program
CLS
nMax = 19 ' current data size. Make sure not exceeded!
IF n > nMax THEN n = nMax

'declare arrays
DIM SHARED dist(nMax, nMax) ' matrix of distances
DIM SHARED city(nMax) AS STRING
DIM P(n), BestP(n)
```

[8] If you want to include more cities, you have to type the distances in a suitable format. If you embark on this project, try first to implement a method for selecting an arbitrary subset of cities to visit.

[9] http://math.uc.edu/b̃rycw/probab/basic.htm

```
'read distances
CALL AssignDistances(nMax)

'initial permutation
FOR j = 1 TO n
        P(j) = j
        BestP(j) = j
NEXT j
'initial length of trip
MinLen = PathLen(P())

'*** main loop
DO  'infinite loop till user stops
'count trials
no = no + 1
        '*** interacting with user
        'check if user pressed key to stop
        k$ = INKEY$
        IF k$ > "" THEN EXIT DO  'exit infinite loop
        'display currrent progress
        display (no)
        '*** get any path
        CALL GetPermutation(P())
        x = PathLen(P())
        IF x < MinLen THEN
                'Better path found, so memorize and display
                Dlen = MinLen - x
                MinLen = x
                '*Memorize best order and print
                FOR j = 1 TO n
                        BestP(j) = P(j)
                        PRINT city(P(j)); "->";
                NEXT j
                'Finish printing
                PRINT city(P(1))
                PRINT "Best so far: "; MinLen
                PRINT "Progress rate "; Dlen / (no - Slen); " miles per trial"
                Slen = no
        END IF
LOOP
'Print final message
CLS
PRINT "ALPHABETIC TOUR OF FIRST "; n; " CITIES in the USA"
PRINT "Blind Search Recommended Route found in "; Slen; "-th search"
FOR j = 1 TO n - 1
        PRINT city(BestP(j)); "-->";
NEXT j
PRINT city(BestP(n)); "-->"; city(BestP(1))
PRINT "Total distance: "; MinLen
LOCATE 22, 40
PRINT "(Distances subject to change)"
END
```

The program runs in the infinite loop until it is stopped by the user. Once stopped, the program prints the best route it found. For larger sets of cities we may have hard time deciding when to stop it. Here is a sample output (from the improved version, as marked in the actual code):

```
ALPHABETIC TOUR OF FIRST 19 CITIES in the USA
Blind Search Recommended Route found in 151,942 searches.
Chicago----Cincinnati----Buffalo---- Albany----Boston----Augusta----Atlantic City
----Baltimore----Charlotte----Atlanta----Birmingham----Baton Rouge----Austin
----Albuquerque----Cheyenne----Boise----Calgary----Billings----Bismarck----Chicago
Total distance:  8822
Can you find a better one?
```

When you run this program on larger sets of cities, you will notice that the program is not fast. One possible improvement in the design of this program is to modify the randomization to be less likely to pick long paths. For instance, you can attempt to modify paths that are known to be short, or weight the modifications by lengths of resulting paths. Such methods are actually in use in image restoration problems (*simulated annealing*), see page 90.

1.4.4 Improving blind search

A bit of experimenting with various "pure" blind search programs should convince you that

- Blind search programs are easy to write, if you know how to code the main function to randomize.
- Blind search always gives "answers"
- It is difficult to judge how good an answer is.
- In situations that we do know the answer, blind search takes long time to reach it.

It is possible to improve on the last aspect without complicating the program much. The idea is to make random modifications of the currently best found value. For example, in a one-dimensional maximization of a function $f(x)$, we would do the following steps

1. Pick an initial "best-so-far" point x_0 and compute initial value $y_0 = f(x_0)$
2. Select at random x_1 in the "neighborhood" of x_0 and compute $y_1 = f(x_1)$
3. Compare y_0, y_1.
   (a) If $y_1 < y_2$ then repeat Step 2.
   (b) If $y_1 \geq y_0$ then make x_1 the new "best-so-far" $y_0 := y_1, x_0 := x_1$. Then repeat Step 2.
4. Stop the program at user request, or when no changes to y_0 occur for prolonged number of attempts to improve it.

We want to allow for the chance of checking points far away from the "best-so-far" answer. But we don't want this to happen too often, because x_0 might be rather close to the optimum. The tradeoffs are that the program will tend to follow "direct path" to the maximum, but the danger is that it will get stuck longer in local maxima.

Improved blind search with time/state dependent randomization is actually implemented within RANDTOUR. It is commented out in the version on the disk, so that it isn't operational. To make it active, uncomment the call to ImproveBest as a replacement for GetPermutation.

1.4.5 Random permutations

Program RANDTOUR.BAS selects permutations at random only in its "simplest" variant. Here are a few examples of problems that require selecting random permutations.

- Card games:
  - Poker hand: Select 5 cards at random from a deck of cards.
  - Poker (2 players): Select 10 cards at random from a deck of cards.
  - Bridge: Split 52 cards into four groups
- Analyzing statistical experiments:

  Suppose there are 7 items hidden under 12 cups, and a person is allowed to try to find them. How often all seven will be recovered by pure lack (as opposed to, say, parapsychic abilities?

The "naive" selection of random permutation wastes many random numbers. Here is an algorithm that conserves resources better. The basic idea is to select a number from the beginning of the list of all numbers, and move it down to the end. The randomly re-arranged numbers are $P(1), P(2), \ldots, P(n)$

```
SUB GetPermutationFast(P())
'Put random integers into P()
n=UBOUND(P) 'how many entries
'this loop can be omitted is we are sure that P(j) list all the numbers we want
FOR j=1 TO n
P(j)=j
next j
for j=1 to UBOUND(P) 'not n as n changes in the loop
k=INT(RND(1)*n+1)

SWAP P(n), P(k)
n=n-1
next j
```

Exercise 1.11 *There are many incompatible measures of "quality" of an algorithm. One of the "objective" criteria is the number of "If" verifications. Another "objective criterion" might be the number of calls to a function. A "subjective" criterion, which depends on the hardware and circumstances, is timing.*

Does SUB GetPermutationFast *deserve adverb* Fast *in its name?*

Project 1.12 *The Subset-Sum Problem is stated as follows:*

Let S be a set of positive integers and let t be a positive integer. Decide of there is a subset $S' \subset S$ such that the sum of integers in S' is exactly t.

The associated optimization problem is to find a subset $S' \subset S$ whose elements' sum is the largest but doesn't exceed t. This optimization problem is NP-complete, ie it isn't known if there is a polynomial time algorithm (polynomial in the size of S) to find S'.

Investigate how the blind search will do on sets S selected at random, and on sets S constructed in more regular fashion like arithmetic progression $S = \{a, 2a, 3a, \dots\}$, geometric progression $S = a, a^2, a^3, \dots$.

1.5 Conditional probability

In modeling more complicated phenomena we may want to use different probabilities under different circumstances. For instance, in a modified blind search for the minimum of a non-negative function, the randomization strategy might be different when we already made some progress, and it might be different when we are "stuck" in a non-optimal location. Thus we may want to consider probabilities of the same event A (say, hitting a maximum) under different conditions B.

To formalize this idea, suppose B is an event such that $\Pr(B) \neq 0$. The last condition merely says that B is an event that does have some chance of occurring. Conditional probability of event A given event B is denoted by $\Pr(A|B)$. It is defined as

$$\boxed{\Pr(A|B) = \frac{\Pr(A\cap B)}{\Pr(B)}.}$$

Conditional probability satisfies the axioms of probability, and $\Pr(A|B) = 0$ if A, B are disjoint. In particular, $\Pr(A'|B) = 1 - \Pr(A|B)$, $\Pr(B|B) = 1$.

The easiest way to find $\Pr(A|B)$ by simulations is to repeatedly simulate the complete experiment, discarding all the outcomes except the ones resulting in B.

Exercise 1.13 *Suppose we toss repeatedly a fair coin, and the "success" is to get heads.*

Use computer simulations to find the conditional probability that the very first trial was successful, if 10 *consecutive (and independent) trials resulted in* 8 *successes.*

Try to answer the same question under the condition that 50 *independent trials resulted in* 40 *successes.*

You should notice that it takes forever to simulate events that happen rarely. Section 6.7 indicates one possible way out of this difficulty.

1.5.1 Properties of conditional probability

Conditional probability is used in modeling. Often $\Pr(A|B)$ can be assigned by "intuitive" considerations. It can then be used to compute other probabilities. The simplest example is $\Pr(A \cap B) = \Pr(B)\Pr(A|B)$, which is a direct consequence of the definition.

Example 1.5 *Suppose we have a deck of 52 cards numbered 1 through 52. Since it isn't obvious how to simulate selecting 5 cards without replacement, we may want to select them with replacement instead. Let A denote the event that all five cards are different. What is the probability of A?*

We may perform the experiment sequentially, drawing one card at a time. Let A_k *denote the event that the* k *consecutive draws resulted in different cards. Then* $A = A_5 \subset A_4 \subset \ldots \subset A_1$. *Moreover,* $\Pr(A_1) = 1$.

Clearly, $\Pr(A_2|A_1) = \frac{51}{52}$, *so* $\Pr(A_2) = \Pr(A_2 \cap A_1) = \Pr(A_2|A_1)\Pr(A_1) = \frac{51}{52}$. *Similarly,* $\Pr(A_3) = \Pr(A_3 \cap A_2) = \Pr(A_3|A_2)\Pr(A_2) = \frac{50}{52}\frac{51}{52}$. *Continuing this we get* $\Pr(A_5) = \frac{51504948}{52^4} \approx 0.82$.

The following identities are also of interest.

1. Path Probability: $\Pr(\bigcap_{k=1}^n A_k) = \prod_{k=1}^n \Pr(A_k|\bigcap_{j=1}^{k-1} A_j)$
2. Bayes theorem: $\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B)}$
3. Total probability formula: If $\{B_n\}$ are pairwise disjoint and *exhaustive*, ie. $\Pr(A_i \cap B_j) = \emptyset$ for $i \neq j$ and $\bigcup B_n = \Omega$, then
$$\Pr(A) = \sum_n P(A|B_n)\Pr(B_n). \tag{1.10}$$

Exercise 1.14 *What is the probability that in a class of 30 students no matching birthdays occur?*

Example 1.6 *A lake has 200 white fish and a 100 black fish, and a nearby pond contains 20 black fish and 10 white ones. No other fish live there.*

A fish is selected at random from the lake and moved to the pond. Then a fish is selected from the pond and moved back to the lake. What is the probability that all fish in the pond are black?

1.5.2 Sequential experiments

Often the main experiment consist of a sequence of sub-experiments, each depending on the outcome of the previous one. If n such sub-experiments are chained, then the full experiment results in a chain of events, or a path $\mathcal{P} = A_1 \cap A_2 \cap \ldots \cap A_n$. If we assume that k-th experiment depends on the outcome of the $k-1$-th experiment only, then $\Pr(A_k|A_{k-1} \cap \ldots \cap A_1) = \Pr(A_k|A_{k-1})$.

Denoting by $\mathcal{P}$ the generic path $A_1 \cap A_2 \cap \ldots \cap A_n$, and by $\mathcal{P}(k) = A_k$ we have the following *path integral* formula for the probability of an event $\mathcal{F}$ specifying the outcome of the complete experiment, and consisting of many paths $\mathcal{P}$.

$$\Pr(\mathcal{F}) = \sum_{\mathcal{P}\in\mathcal{F}} \Pr(\mathcal{P}) = \sum_{\mathcal{P}\in\mathcal{F}} \prod_{k=0}^{|\mathcal{P}|} \Pr(\mathcal{P}(k+1)|\mathcal{P}(k)). \tag{1.11}$$

Example 1.7 *Suppose that the double transfer operation from Example 1.6 was repeated twice. That is, a random selection was done four times. What is the probability that all fish in the pond are black?*

Exercise 1.15 *Check by simulation how the proportion of black fish in the pond changes when the random transfers from Example 1.6 are performed repeatedly for a long time.*

1.6 Independent events

This section introduces the main modeling concept behind the entries in the Table 1.1.

Two events A, B are independent, if the conditional probability is the same as unconditional, $\Pr(A|B) = \Pr(A)$. This is stated in multiplicative form which exhibits symmetry and includes *trivial* events[10]

Definition 1.6.1 *Events A, B are independent if* $\Pr(A \cap B) = \Pr(A)\Pr(B)$.

Independence captures the intuition of non-interaction, and lack of information. In modeling it is often assumed rather than verified. For instance, we shall assume that the events generated by consecutive outputs of the random generator are independent. We also assume that tosses of a coin (fair, or not!) are independent.

Beginners sometimes confuse disjoint versus independent events. Exclusive (ie. disjoint) events capture the intuition of non-compatible outcomes. Not compatible outcomes cannot happen at the same time. This is not the same as independent outcomes. If A, B are disjoint and you know that A occurred, then you do know a lot about B. Namely you know that B cannot occur. Thus there is an interaction between A and B. Knowing whether A occurred influences chances of B, which is not possible under independence.

Independence (or, more properly, *mutual stochastic independence*) of families of events is defined by requesting a much larger number of multiplicative conditions. The reason behind is Theorem 1.6.1, which provides a very convenient tool.

Definition 1.6.2 *Events $A_1, A_2, \ldots, A_n$ are independent, if* $\Pr(\bigcap_{j\in J} A_j) = \prod_{j\in J}\Pr(A_j)$ *for all finite subsets $J \subset \mathbb{N}$.*

Example 1.8 *A coin is tossed repeatedly. Find the probability that heads appears for the first time on the fourth toss.*

Problem 1.16 `SUB GetPermutation` *from the program* `RANDTOUR.BAS` *selects numbers between* 1 *and n at random until it finds a number not yet on the list. Then it ads the number to its list, and repeats the process.*

1. *What is the probability that the second number added to the list required more than k attempts?*
2. *What is the probability that the last number added to the list required more than k attempts?*

Another important concept is the conditional independence. For example, many events in the past and in the future are dependent. But in many mathematical models, past and future are independent conditionally on the present situation. In such a model future depends on past only through present events!

Definition 1.6.3 *Let C be a non-trivial event. Events A, B are C-conditionally independent if* $\Pr(A \cap B|C) = \Pr(A|C)\Pr(B|C)$.

[10] Trivial events are those with probabilities 0, or 1.

1.6.1 Random variables

The general concept of probability space uses "abstract" sets to represent outcomes of an experiment. But many examples considered so far, represented the outcomes in numerical terms.

Random variables are introduced for convenient description of experiments with numerical outcomes. (The other option is to select $\Omega \subset \mathbb{R}$, or $\Omega \subset \mathbb{R}^d$.) If we want to run computer simulations, we need to represent even non-numerical experiments (like tossing coins) in numerical terms anyhow. Thus the language of random variables becomes the natural extension of elementary probability theory, expressing many of the same concepts in a little different language.

A random variable is the numerical quantity assigned to every outcome of the experiment. In mathematical terms, random variable is a function $X : \Omega \to \mathbb{R}$ with the property that sets $\{\omega \in \Omega : X(\omega) \leq a\}$ are events in $\mathcal{M}$ for all $a \in \mathbb{R}$. Recall that the last conditions means that we may talk about probabilities of events $\{\omega \in \Omega : X(\omega) \leq a\}$.

Probabilities for a one-dimensional r. v. are determined by the cumulative distribution function

$$F(x) = \Pr(X \leq x) \tag{1.12}$$

The corresponding tail function $R(x) = 1 - F(x) = \Pr(X > x)$ is sometimes called the *reliability*[11] *function.*

Cumulative distribution function can be used to express probabilities of intervals $\Pr(a < X \leq b) = F(b) - F(a)$. Since probability is continuous, (1.1) we can also compute $\Pr(X = a) = \lim_{b \to a^+} \Pr(a < X \leq b) = F(a^+) - F(a)$. The right hand side limit $F(a^+)$ exists, as F is a non-decreasing function.

Example 1.9 *Suppose* $F(x) = (1 - e^{-x}) \vee 0$. *Then* $\Pr(|X - 2| < 1) = e^{-1} - e^{-3}$.

In probability theory we are concerned with probabilities. Random variables that have the same probabilities are therefore considered equivalent. We write $X \cong Y$ to denote the equality of distributions, ie. $\Pr(X \in U) = \Pr(Y \in U)$ for all *Borel* sets $U \subset \mathbb{R}$ (say, all intervals U).

Vector valued r. v. are measurable the functions $\Omega \to \mathbb{R}^d$. In the vector case we also refer to $\mathbf{X} = (X_1, \dots X_d)$ as the d-variate, or multivariate, random variable.

We will use the ordinary notation for sums and inequalities between random variables. There is however a word of caution. In probability theory, equalities and inequalities between random variables are interpreted almost surely. For instance $X \leq Y + 1$ means $\Pr(X \leq Y + 1) = 1$; the latter is a shortcut that we use for the expression $\Pr(\{\omega \in \Omega : X(\omega) \leq Y(\omega) + 1\}) = 1$.

Problem 1.17 *Show that* $F(x) = \Pr(X \leq x)$ *is right continuous:* $\lim_{x \to a^+} F(x) = F(a)$.

1.6.2 Binomial trials

The statistical analysis of repeated experiments is based on the following.

[11] This terminology arises under the interpretation that X represents failure time.

Theorem 1.6.1 *Suppose that for $j \in \mathbb{N}$ event B_j is either S_j or S'_j, where events $\{S_j\}$ are independent. Then $\{B_j\}$ are independent.*

A binomial experiment, called also binomial trials, consists of the sequence of simpler identical experiments that have two possible outcomes each. The independent events S_j represent *successes* in consecutive experiments. We assume that we have an infinite sequence of events $S_1, S_2, \dots S_k, \dots$ that are independent and have the same probability $p = \Pr(S_j)$. We denote by $F_j = S'_j$ the failure in the j-th experiment, and put $q = 1 - p$.

Two important random variables are associated with the binomial experiment are the number X of successes in n trials, and the number T of trials until first success.

Example 1.10 *The probability that number X of successes in n trials is k is* $\Pr(X = k) = \binom{n}{k} p^k q^{n-k}$. *(Here $k = 0, \dots, n$.)*

Example 1.11 *The probability of more than n attempts needed for the first success is* $\Pr(T > n) = q^n$. *The probability that first success occurs at the n-th trial is* $\Pr(T = n) = pq^{n-1}$ *(geometric).*

Example 1.12 *Geometric distribution has* lack of memory *property:* $\Pr(T > n + k | T > n) = \Pr(T > k)$.

Random variables are often described solely in terms of cumulative distribution function $F(x)$, or formulas for $\Pr(X = x)$ without reference to the underlying probability space Ω. For instance, the number of minutes T that we spend waiting for a bird to come to the bird feeder at the back of my house is random, and I believe $\Pr(T = n) = pq^{n-1}$ because $\Pr(T > n + k | T > n) = \Pr(T > k)$.

It is intuitively obvious that on average we get np successes in n trials. It is perhaps less obvious[12] that on average we need $1/p$ trials to get the first success.

Exercise 1.18 *Write a simulation program to verify the claims about the averages for several values of p.*

Example 1.13 *The probability that in $2n$ tosses of a fair coin, half are heads is* $\frac{(2n)!}{4^n (n!)^2} \approx \frac{1}{\sqrt{\pi n}} \to 0$ *as $n \to \infty$. The latter isn't easy to prove, but the computer printout is quite convincing, see Table 1.2 (note that* $\frac{1}{\sqrt{\pi}} \approx 0.5642$*).*

$2n$	$\Pr(X = n)$	Frequency in 1000 trials	$\sqrt{n}\Pr(X = n)$
100	0.07959	0.08200	0.56278
300	0.04603	0.06100	0.56372
500	0.03566	0.03700	0.56391
700	0.03015	0.02200	0.56399

Table 1.2: Probabilities $\Pr(X = n)$ in $2n$ Binomial trials.

[12] A possible heuristic argument may argue that Tp is on average 1.

1.7 Further notes about simulations

By now you should have written some simple simulation programs, and printed out the results. It is perhaps a good moment to pause and consider what are the aspects of simulations that we are interested in.

In general, we would like to get answers to questions that we don't know how to answer in any other way. But before we do that, we should develop some intuition on the cases that can check the answers. Therefore we begin with simulation of probabilities or averages that are known.

A Simulation of probabilities/ averages that are known should address the following questions.

1. How close the simulation answers are to the theoretical answers? Print them side-by-side.
2. How large the simulation should be? Is it worth to change simulation size from 1,000 to 10,000 trials? In order to answer this question, your simulation has to provide "relative" rather than absolute answers. (Answers of the form "got 32 heads" are meaningless as they depend on simulation size!)
3. How do the answers change as we change the parameters? If you did a simulation of the fair coin, you could change the probability p from the usual value $\frac{1}{2}$.

B The next natural step is to extend models that we know how to handle both theoretically and by simulations to cover aspects that aren't easily accessible by theory. The sample questions involve

1. How would the answers change, if we allow perhaps more realistic assumptions in the model? As an example, suppose that we would like to model the birthday problem with people born non-uniformly throughout the year. Which way would you expect the answer to change?
2. What are typical errors of a simulation of size n? How can we estimate the accuracy of the answer without having the exact answer to compare it to? Chapter 5 gives theoretical basis for such estimates.

1.8 Questions

Problem 1.19 (Exercise) *A family has two children, and one of them is a boy. What is the probability that they have two boys? (If you think this is too hard mathematics, do it as a computer assignment!)* $\boxed{\text{ANS: 1/3}}$

Problem 1.20 (Exercise) *A die is thrown until an ace turns up.*

Assuming the ace doesn't turn up on the first throw, what is the probability that more than three throws (ie. at least four) will be necessary? $\boxed{\text{ANS: 197/198}}$

Suppose that an ace turns up on an even throw. What is the probability that it turned up on the second throw?

(If you think this is too hard mathematics, do it as a computer assignment!)

Project 1.21 *A deck of 52 cards has 4 suits and 13 values per suit.*

1. *Write a program simulating the hand of 5 cards.*
2. *Use your program to answer the following questions:*
   (a) *How often does a pair occur?*
   (b) *How often does a two-pair occur?*
   (c) *How often does a* three of a kind *(three of same value and two different) occur?*
   (d)
   (e) *How often does a four of a kind occur?*
   (f) *How often does a* full house *(2+3) occur?*
   (g) *How often does a* straight *(five cards in a row, not all same suit) occur?*
   (h) *How often does a* flush *(five cards of one suit, not in order) occur?*
   (i) *How often does a* straight flush *(five cards in a row all same suit) occur?*

   If you think this is too difficult on a computer, compute the probabilities by hand.

Exercise 1.22 *A math teacher in a certain school likes to give multiple choice tests, grade them as either right, or wrong, and then lets the students to go over the test and correct the ones they got wrong. This gives them two chances to get a problem right, and the chance of getting a question right increases even if the student just guesses the answers. Suppose a student simply guessed the first time, got the corrections, and guessed differently the second time on the wrong answers. How much his grade improves?*

Project 1.23 *This is the expanded version of Exercise 1.14. In a group of n people, how often at least two have the same birthday?*

1. *Find the formula for the probability $p(n)$ assuming 365 days per year, and equally likely birthdays.*
2. *Compute the probabilities for $n = 20, 30, 40, 50$*
3. *Write a simulation program, and verify if the simulation agrees with the theoretical answers.*
4. *Modify the simulation program to allow for not equal birthdays. Assume January, February, March are less likely then the other days of the year. Change the parameters, and verify how the probabilities $p(n)$ change as you depart from the uniform probabilities. If the change of $p(n)$ is of the magnitude comparable to the simulation accuracy, clearly it is irrelevant.*
5. *A randomly selected person has chance 1/4 to be born on a leap year. How does this affect the answers?*

Chapter 2

Random variables (continued)

> *Tree species are not distributed at random but are associated with special habitats.*
> The Auborn Society Field Guide to North American Trees.

Intuitively, random variables are numerical quantities measured in an experiment. The concept[1] is the core of probability theory; it leads outside of elementary probability and it touches advanced concepts of integration, function transforms and weak limits.

For convenience random variables are split into three groups: continuous, discrete, and the rest.

2.1 Discrete r. v.

Definition 2.1.1 *X is a discrete r. v. if $X(\Omega)$ is countable.*

The definition says that X is a discrete r. v. if there is a finite, or countable set $\mathcal{V}$ of numbers (values) of X such that $p_v = \Pr(X = v) > 0$ and $\sum_{v\in\mathcal{V}} p_v = 1$. The function $f(v) = p_v$ is then called the probability mass function of X. For completeness, the domain of the probability mass function is often extended to all $x \in \mathbb{R}$ (or to $\mathbf{x} \in \mathbb{R}^d$ in the multivariate case) by $f(x) = \Pr(X = x)$.

It is easy to see that if f is a function which satisfies two natural conditions:

$$f(x) \geq 0 \tag{2.1}$$

$$\sum_{x\in\mathbb{R}} f(x) = 1 \tag{2.2}$$

$$\tag{2.3}$$

then there is a probability space with a random variable X such that f is its probability mass function. In modeling random phenomena we can therefore avoid the difficulties of designing appropriate sample spaces, and pick directly relevant densities. The question, if a density does describe the actual outcomes of experiment is to some extend the question of statistics. Properties of various distributions, like lack-of-memory come also handy when selecting appropriate density function.

[1] The precise definition is in Section 1.6.1.

For discrete r. v. the cumulative distribution function (1.12) plays lesser role. It is a discontinuous function given by the expression

$$F(x) = \sum_{v \leq x} p_v. \tag{2.4}$$

This expression does show up in the “generic simulation method in Section 2.1.2.

2.1.1 Examples of discrete r. v.

Table 2.1 list the most frequently encountered discrete distributions.

Name	**Values**	**Probabilities**	**Symbol**	**Parameters**
Binomial	$0,\dots,n$	$\Pr(X=k)=\binom{n}{k}p^k(1-p)^{n-k}$	Bin(n,p)	$0 \leq p \leq 1, n \in \mathbb{N}$
Poisson	$\mathbb{Z}_+$	$\Pr(X=k)=e^{-\lambda}\frac{\lambda^k}{k!}$	$Poiss(\lambda)$	$\lambda > 0$
Geometric	$\mathbb{Z}_+$	$\Pr(X=k)=p(1-p)^{k-1}$		$0 \leq p \leq 1$
Uniform	$\{x_1,\dots,x_k\}$	$\Pr(X=x_j)=\frac{1}{k}$		$k \in \mathbb{N}, x_1,\dots,x_k \in \mathbb{R}$
Hypergeometric				
Negative Binomial				

Table 2.1: Discrete random variables.

Example 2.1 *Let the random variable X denote the number of heads in three tosses of a fair coin.*

Example 2.2 *Let the random variable X denote the score of a randomly selected student on the final exam.*

Problem 2.1 *Let N be $Poiss(\lambda)$, and assume N balls are placed at random into n boxes. Find the probability that exactly m of the boxes are empty.* $\boxed{\text{ANS: } \binom{n}{m}e^{-\lambda m/n}(1-e^{\lambda/n})^{n-m}}$.

2.1.2 Simulations of discrete r. v.

Discrete random variables with finite number of values are simulated by assigning values according to the ranges taken by the (pseudo)random uniform random variable U from the random number generator, `U=Rand(1)`. To decide which value of X should be generated, take a partition $\{0 = a_0 \leq a_1 \leq \dots \leq a_{n-1} \leq a_n \dots \leq 1\}$ of interval $(0,1)$. This means that we simulate $X = f(U)$ using a piecewise constant function f on the interval $(0,1)$. If $f(x) = v_k$ for $x \in (a_k, a_{k+1})$, then $\Pr(X = v_k) = a_{k+1} - a_k$. Therefore we choose $a_1 = 0$, $a_2 = p_1, \dots, a_{k+1} = p_1 + \dots + p_k$. Notice that $a_k = \Pr(X \leq v_k) = F(v_k)$.

Other methods are also available for the distributions from Table 2.1. For example, program `TOSSCOIN.BAS` on page 7 simulates binomial distribution Bin(n=100, p=1/2).

The following exercise provides tools to run more involved simulations.

Exercise 2.2 *Write functions* `SimOneBin(n,p)` *and* `SimOneGeom(p)` *that will simulate a single occurrence of the Bin(n,p) r. v. and the geometric r. v. The sample usage:* `PRINT SimOneBin(15,.5)` *should simulate the number of heads in tossing* 15 *fair coins.*

Also write function `SimGeneric(p())` *which simulates generic r. v. with values* $\{0, 1, \dots, n\}$ *and prescribed probabilities* $p_k = p(k)$.

2.2 Continuous r. v.

Continuous random variables have uncountable sets of values, and the probability of each of them is zero, $\Pr(X = x) = 0$ for all $x \in \mathbb{R}$.

Since probability satisfies continuity axiom (1.1), $\Pr(X \in (a, a+h)) \to 0$ as $h \to 0$ for all a. The main interest in continuous case is that the rate of convergence to 0 is also known, $\Pr(X \in (a, a+h)) \approx f(a)h + o(h)$. Function $f(x)$ in this expansion is called the density function.

In terms of the cumulative distribution function 1.12), the probability is $\Pr(X \in (a, a+h)) = F(a+h) - F(a)$, and thus $f(a) = \lim_{h\to 0} \frac{F(a+h)-F(a)}{h} = F'(a)$ is the derivative of the cumulative distribution function F. Therefore when the Fundamental Theorem of Calculus can be invoked (say, when f is piecewise continuous)

$$F(x) = \int_{-\infty}^{x} f(u)\, du. \tag{2.5}$$

Definition 2.2.1 *Random variable X is (absolutely) continuous, if there is a function f such that $\Pr(X \in U) = \int_U f(x)dx$. Function f is called the probability density function of X.*

It is known that if f is a function which satisfies two natural conditions:

$$f(x) \geq 0 \tag{2.6}$$

$$\int_{\mathbb{R}} f(x)\, dx = 1 \tag{2.7}$$

$$\tag{2.8}$$

then there is a probability space with a random variable X such that f is its density. This is in complete analogy with the discrete case. In modeling random phenomena we can therefore avoid the difficulties of designing appropriate sample spaces, and pick directly relevant densities. The question, if a density does describe the actual outcomes of experiment is to some extend the question of statistics. Properties of various distributions come also handy when selecting appropriate density function.

It is convenient to use the heuristic probability density function in continuous case corresponds to probability mass function in discrete case, and that expressions that involve in discrete case sums are replaced by integrals, compare (2.5) and (2.4).

2.2.1 Examples of continuous r. v.

The following table lists more often encountered densities. Figures 2.1 and 2.2 give the graphs of the normal and exponential densities.

Example 2.3 *A dart is thrown at a circular dart board of radius 6. Let X denote the distance of the dart from the center. Assuming uniform assignment of probability (1.3), the density of X is* $f(x) = \begin{cases} \frac{x}{18} & \textit{if } 0 \leq x \leq 6 \\ 0 & \textit{otherwise} \end{cases}$

Problem 2.3 *Referring to Exercise 1.3, let X, Y denote the arrival times of the two drivers at the intersection. Find the density of the time lapse $|X - Y|$ between their arrivals.*

Name	**Range**	**Density**	**Symbol**	**Parameters**
Normal	$-\infty < x < \infty$	$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(x-\mu)^2}{2\sigma^2}$	$\mathrm{N}(\mu, \sigma)$	$\mu \in \mathbb{R}, \sigma > 0$
Exponential	$x > 0$	$f(x) = \lambda e^{-\lambda x}$		$\lambda > 0$
Uniform	$a < x < b$	$f(x) = \frac{1}{b-a}$	U(a,b)	$a < b$ real
Gamma	$x > 0$	$f(x) = Cx^{\alpha-1}e^{-x/\beta}$	$\mathrm{Gamma}(\alpha, \beta)$	$\alpha > 0, \beta > 0, C = \frac{1}{\beta^\alpha \Gamma(\alpha)}$
Weibull				

Table 2.2: Continuous random variables.

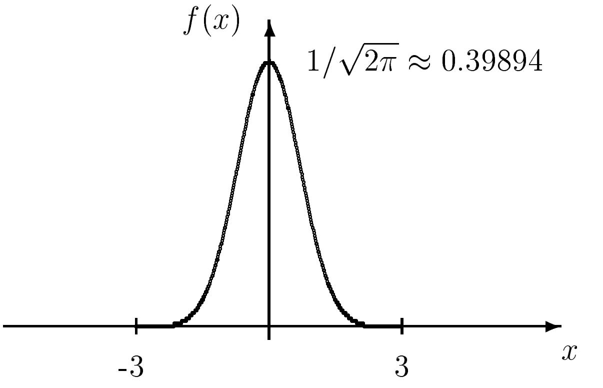

Figure 2.1: Graph of the standard normal $N(0, 1)$ density $f(x) = (2\pi)^{-1/2}e^{-x^2/2}$.

2.2.2 Histograms

Simulations and experiments do not give direct access to the density, but often a histogram will approximate it reasonably well. Histograms are graphical representations of empirical data. A sample histogram is drawn on the side of the square in Figure 5.1 on page 55.

To create a useful histogram, split the range into the finite number of intervals. Then graph over each interval the rectangle with the area equal to the observed frequency. The number, and positioning of intervals depends on the amount of data available, and personal preference.

2.2.3 Simulations of continuous r. v.

The generic method for simulating a continuous random variable is similar to the method used in the discrete case. Namely, we take $X = f(U)$ with the suitable function f.

To find f assume it is increasing and thus invertible with inverse g. Then $\Pr(f(U) < x) = \Pr(U < g(x)) = g(x)$. This completes the generic prescription: take as $f(x)$ the inverse[2] of the cumulative distribution functions $F(x) = \Pr(X \leq x)$.

[2] Actually, we need only a right-inverse, i.e a function such that $F(f(u)) = u$.

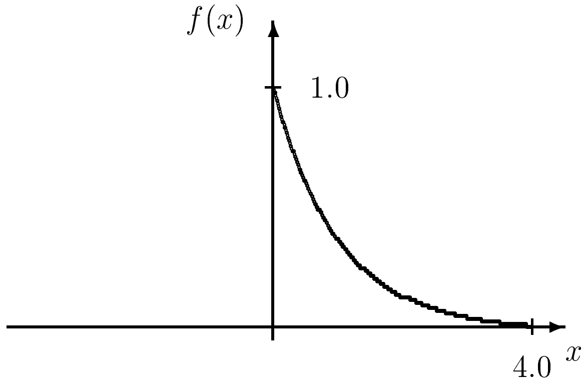

Figure 2.2: Graph of the exponential density $f(x) = e^{-x}$ as the function of $x > 0$.

This method of simulation is quite effective if the inverse of F can be found analytically. It becomes slow when the inverse (or, worse still, cumulative distribution function F itself) is computed by numerical procedures. Since this is the case of the normal distribution, special methods are used to simulate the normal distribution.

Example 2.4 *To simulate X which is exponential with parameter λ, use $X = -\frac{1}{\lambda} \ln U$.*

2.3 Expected values

Expected values are perhaps the single most important numerical characterization of a random phenomenon.

Definition 2.3.1 *For discrete random variable X the expected value EX is given by $EX = \sum_v v \Pr(X = v)$, provided the series converges.*

Expected value captures the intuition of the average of a random quantity. It is also this intuition that leads to estimating the expected value by averaging the outcomes of simulations.

Example 2.5 *If X has values $x_1, \ldots, x_n$ with equal probability, then $EX = \bar{x}$ is the arithmetic mean of $x_1, \ldots, x_n$.*

Simulationsare oten used to get answers that are too difficult to find analytically. The following exercise can be answered by simulation if you figure out how to shuffle cards from a deck at random (Exercise 1.9).

Exercise 2.4 *What is the expected number of cards which must be turned over in order to see each of one suit.*

Example 2.6 *If X takes two values $a < b$ and $\Pr(X = a) = p$, then $EX = pa + (1 - p)b$ is the number in the closed interval $[a, b]$.*

Definition 2.3.2 *For continuous random variable X the expected value EX is given by $EX = \int_{\mathbb{R}} xf(x)\,dx$, provided the integral converges.*

Name	Probability distribution	EX
Normal	$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(x-\mu)^2}{2\sigma^2}$	μ
Exponential	$f(x) = \lambda e^{-\lambda x}$	$\frac{1}{\lambda} > 0$
Uniform	$f(x) = \frac{1}{b-a}$	$\frac{1}{2}(a+b)$ real
Gamma		
Weibull		
Binomial	$\Pr(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$	np
Poisson	$\Pr(X=k) = e^{-\lambda}\frac{\lambda^k}{k!}$	λ
Geometric	$\Pr(X=k) = p(1-p)^{k-1}$	$\frac{1}{p}$
Hypergeometric		
Negative Binomial		

Table 2.3: Expected values of some random variables.

Problem 2.5 *Compute EX for the entries in Table 2.3.*

Exercise 2.6 *Simulate EX for the entries in Table 2.3 (except normal) for different values of parameters involved.*

Problem 2.7 (Exercise) *Referring to Example 2.3, what is the average distance of a dart from the center?* $\boxed{\text{ANS: 4}}$.

If you chose to do the simulations, you should perhaps notice that it is rather difficult to decide how many simulations to take in order to achieve the desired accuracy. Typically, you need to increase the size of a simulation four times to half the error.

Another point to keep in mind is that simulations do return numbers. But from numbers alone it is difficlut to see how parameters of the model change it, and this is a more interesting question.

2.3.1 Tail integration formulas

The following *tail integration formula* is of considerable convenience in theoretical analysis.

Theorem 2.3.1 *For non-negative random variables, both in the discrete, and in the continuous case*

$$EX = \int_0^\infty \Pr(X > t)\,dt \tag{2.9}$$

(The expected value is finite if and only if the corresponding integral converges.)

Proof. To simplify the proof and expose the main idea more clearly, consider the discrete case with a finite number of values. Similarly, to avoid technicalities we consider continuous case with bounded range only.

Discrete case: $EX = x_1p_1 + x_2p_2 + \ldots + x_np_n = x_1(p_1+p_2+\ldots+p_n - (p_2+\ldots+p_n)) + x_2(p_2+\ldots+p_n - (p_3+\ldots+p_n)) + \ldots + x_{n-1}(p_{n-1}+p_n - p_n) + x_np_n = x_1(p_1+p_2+\ldots+p_n) + (x_2-x_1)(p_2+\ldots+p_n - (p_3+\ldots+p_n)) + \ldots + (x_{n-1}-x_{n-2})(p_{n-1}+p_n) + (x_n - x_{n-1})p_n$ The latter is $\int_0^{x_n} \Pr(X.t)\,dt$.

Continuous case: Let f denote the density and F be the cumulative distribution function. Integrating by parts $EX = \int_0^b xf(x)\,dx = b - \int_0^b F(x)\,dx = \int_0^b (1 - F(x)\,dx$. □

If X is discrete integer valued $X \in \{0, 1, \ldots\}$, then (2.9) can be written as

$$EX = \sum_{n=0}^{\infty} \Pr(X > n). \tag{2.10}$$

It is natural to define EX for more general r. v. in the same way through formula (2.9). Write $X = X^+ - X^-$ to decompose X into its non-negative, and non-positive parts, and then define $EX = \int_0^\infty P(X > t)\,dt - \int_0^\infty P(X < -t)\,dt$. Clearly, if one of the integrals diverges, EX is not defined.

Example 2.7 *Suppose X is exponential with the density $f(x) = e^{-x}$. Let Y be X truncated at level 3. That is* $Y = \begin{cases} X & \textit{if } X \le 3 \\ 3 & \textit{if } X \ge 3 \end{cases}$

Clearly, Y is not continuous, as $\Pr(Y = 3) = \Pr(X > 3) = e^{-3} > 0$. *On the other hand, Y is not discrete as it takes uncountable number of values; in fact all the numbers between* 0 *and* 3 *are possible. The definitions of the expected value we gave do not apply, but (2.9) can be used to show that* $E(Y) = 1 - e^{-3}$.

Indeed, $\Pr(Y > t) = \Pr(X > t) = e^{-t}$ *for* $0 < t < 3$, *and* $\Pr(Y > t) = 0$ *for* $t > 3$.

Example 2.8 *To determine the mean of the geometric distribution we can either compute the sum* $p\sum_{n=1}^\infty n(1-p)^{n-1}$, *or use (2.9) and find the easier sum* $\sum_{n=0}^\infty (1-p)^n$.

2.3.2 Chebyshev-Markov inequality

The following inequality is known as Markov's, or Chebyshev's inequality. Despite its simplicity it has numerous non-trivial applications, see eg. Theorem 5.6.1.

Proposition 2.3.2 *If* $U \ge 0$ *then*

$$P(U > t) \le \frac{EU}{t} \tag{2.11}$$

Indeed, by (2.9) we have $EU = \int_0^\infty \Pr(U > x)\,dx \ge \int_0^t \Pr(U > x)\,dx \ge tP(|X| > t)$.

Problem 2.8 *Suppose U is uniform* $U(0,1)$. *Then* $\Pr(U > t) \le \frac{1}{2t}$. *This means that* $1 - t < \frac{1}{2t}$.

1. *Is the above inequality "sharp"? (Graph both curves).*

2. *Is (2.9) sharp? That is, given* $t > 0$, *is there an* $X > 0$ *such that equality occurs?*

2.4 Expected values and simulations

Expected value EX is approximated without much difficulty[3] on a computer by averaging large number of *independent trials.*

Example 2.9 *To find the average of the uniform $U(0,1)$ distribution, take $\frac{1}{n}(U_1 + \ldots + U_n)$.*

This is the basis of *Monte-Carlo* methods , which is a family of related probabilistic methods for computing the integrals. To find $\int_a^b f(x)\,dx$ we simulate $X_j = f(a + bU_j)$. The average $\frac{1}{n}\sum_{j=1}^n X_j$ approximates $\frac{1}{b-a}\int_a^b f(u)\,du$ for large n.

The variance of the n-th approximation is of the order $n^{-1/2}$, which is worse than the *trapezoidal rule* for smooth functions. In return the approximation is insensitive to the smoothness of the integrands, and also to the dimension of the integral. Monte Carlo methods can be used effectively for multiple integrals over irregular domains.

Exercise 2.9 *Use Monte Carlo method to approximate $\pi = \int_{-1}^1 2\sqrt{1-x^2}\,dx$. (You may want to compare the output with numerical procedures described in Section C.1.)*

Another method of similar nature is to pick points (X,Y) at random from the rectangle containing the graph of f and check if $Y < f(X)$ holds. The proportion of "successes" approximates the proportion of the area under the graph of f.

Exercise 2.10 *Approximate $\pi/4$ by selecting points (X,Y) at random from the unit square, and checking if $X^2 + Y^2 < 1$.*

The following sample program computes numerically double integral $\int\int_U \cos(10x + 20y)\,dxdy$ over a circle $x^2 + y^2 = 1$. The only conceptual difficulty as compared to single integrals is how to select points at random from the unit disk. This is done by picking points from a bigger square and discarding those that didn't make it. Can you do this integral analytically? Or by another numerical procedure?

```
PROGRAM dblint.bas

'

'declarations
DECLARE FUNCTION Integrand! (X!, Y!)
DECLARE FUNCTION InDomain! (X!, Y!)

CONST True = -1

' simulation loop
NumTrials = 10000
FOR j = 1 TO NumTrials
        'select random points from the square [-1,1]x[-1,1]
        X = 2 * RND(1) - 1
        Y = 2 * RND(1) - 1
        'check if this is in the domain
```

[3] Provided that limited accuracy is admissible

```
        IF InDomain(X, Y) THEN
                NumTested = NumTested + 1
                Sum = Sum + Integrand(X, Y)
                Var = Var + Integrand(X, Y) ^ 2
        END IF
NEXT j


'Print the answer
PRINT "Examined "; NumTested; " random points"

IF NumTested = 0 THEN END 'nothing found
N = NumTested
PRINT "The integral is approximately "; Sum / N
PRINT "With 95% confidence  the error is less than "; 1.96 * SQR(Var / N - (Sum / N) ^ 2) / SQ

END

FUNCTION InDomain (X, Y)
'This function checks if $x,y$ is in the integration domain
'The definition of the domain can be easily modified here, including
'more complicated domains
IF X ^ 2 + Y ^ 2 < 1 THEN InDomain = True
END FUNCTION

FUNCTION Integrand (X, Y)
'This is the function to be integrated

Integrand = COS(10 * X + 20 * Y)
'

END FUNCTION
```

It is important to have some idea about how accurate the answer is. The program uses the Central Limit Theorem, see Section 5.4, to estimate the magnitude of the error. A less sharp (and thus safer to use!) error estimate can be obtained from (2.11)

Project 2.11 *There are two natural choices to estimate by simulation events of small probability. We can pick a large sample size n, run the simulation experiment and hopefully get several data points. The outcome X of such an experiment is a binomial random variable with unknown probability p of success, and we would estimate $p \approx X/n$. The trouble is that if we don't know how small the chances are, we might get none, estimating probability to be zero. Or we could run the experiment until we get the prescribed number of successes. The observation would then consist of a set of geometric random variables $T_1, \ldots, T_k$ with unknown mean $ET = 1/p$. We could then estimate $p = 1/ET$ by taking the inverse of the arithmetic mean of T_j. In this approach we are guaranteed to get some observations, as long as $p \neq 0$.*

The question is Which of the methods would you recommend to use? *(Of course, you would recommend a better method, but what the word "better" might mean here?)*

2.5 Joint distributions

Often an experiment involves measuring two or more random numbers, say X and Y. The fact that we know the distribution of X, and the distribution of Y separately doesn't determine probabilities of events that involve both X and Y simultaneously.

Example 2.10 *Suppose*

$$\begin{aligned} \Pr(X=1,Y=1) &= \frac{1}{4}+\epsilon \\ \Pr(X=-1,Y=-1) &= \frac{1}{4}+\epsilon \\ \Pr(X=-1,Y=1) &= \frac{1}{4}-\epsilon \\ \Pr(X=1,Y=-1) &= \frac{1}{4}-\epsilon \end{aligned} \tag{2.12}$$

Then $\Pr(X=1)=\Pr(Y=1)=\frac{1}{2}$ *regardless of the value of* ϵ. *On the other hand,* $\Pr(X=Y)=\frac{1}{2}+2\epsilon$ *clearly depends on the value of* ϵ.

It is clear that if X is discrete, and Y is discrete, then (X,Y) is an $\mathbb{R}^2$ valued discrete r. v. That is, the values of the pair are countable. Probabilities $\Pr(X=x,Y=y)$ are called the joint distribution. Corresponding $\Pr(X=x)$ and $\Pr(Y=y)$ are the so called marginals. Example 2.10 points out that marginals do not determine joint probabilities uniquely. But if we know the joint probabilities then we can compute the marginals, eg $\Pr(X=x)=\sum_y \Pr(X=x,Y=y)$.

In contrast to the discrete case, joint continuity can not be recognized from the continuity of the components, and requires full definition.

Definition 2.5.1 *Let* $\mathbf{X}=(X_1,\dots,X_n)$. *Random variables* $X_1,\dots,X_n$ *are jointly (absolutely) continuous, if there is a function* f *such that*

$$\Pr(\mathbf{X}\in U)=\int\dots\int_U f(x_1,\dots x_n)\,dx_1\dots dx_n$$

for all measurable U. *Function* f *is then called the probability density function of* $\mathbf{X}$.

Example 2.11 *Suppose* X,Y *have uniform distribution in the unit disk. Then the joint density is* $f(x,y)=1/\pi$ *for* $x^2+y^2\le 1$ *and the density of* X *is* $f_X(x)=\frac{2}{\pi}\sqrt{1-x^2}$.

The relation between probabilities and the density is

$$f(a,b)=\frac{\partial^2}{\partial a\partial b}\Pr(X\le a,Y\le b). \tag{2.13}$$

Occasionally this can be used to determine the joint density.

2.5.1 Independent random variables

Independence of random variables is defined in terms of joint distributions. The intuition behind this definition is that the events that random variables may generate should be independent. Notice however that the actual definition is simpler than Definition 1.6.2. Can you explain why?

Definition 2.5.2 *Random variables $X_1, \dots, X_n$ are independent, or stochastically independent, if*

$$\Pr(X_1 \in U_1, \dots, X_n \in U_n) = \Pr(X_1 \in U_1) \dots \Pr(X_n \in U_n) \tag{2.14}$$

for all measurable $U_1, \dots, U_n \subset \mathbb{R}$.

(Similarly we define the stochastic independence of random vectors).

We say that $X_1, \dots, X_n$ are *independent identically distributed* (i. i. d) if (2.14) holds and $\Pr(X_i \in U) = \Pr(X_j \in U)$ for all Borel $U \subset \mathbb{R}$.

Proposition 2.5.1 *If X, Y are discrete with the probability mass function $f(x, y)$, then independence of X, Y is equivalent to $f(x, y) = f_X(x) f_Y(y)$.*

If X, Y are continuous with the joint density $f(x, y)$ then independence of X, Y is equivalent to $f(x, y) = f_X(x) f_Y(y)$.

Independence is often part of the model. Independence allows to determine joint distributions from marginals. Thus each independent random variable can be analyzed separately, and then more complex questions can be answered. From the mathematical perspective, under independence we can determine joint distributions if we know marginals.

Example 2.12 *Suppose X is binomial Bin(n,p) and Y is Poiss(λ). If X, Y are independent, then the joint probability mass function of X, Y is given by $f(x, y) = \binom{n}{x} p^n (1-p)^{n-x} e^{-\lambda} \lambda^y / y!$ (or 0).*

Example 2.13 (Example 1.3 continued) *Two drivers arrive at an intersection between 8:00 and 8:01 every day. Their arrival times are independent random variables. Indeed, using formula (2.13) and elementary area computation,* $\Pr(X < a, Y < b) = ab$ *for* $0 < a, b < 1$.

2.6 Functions of r. v.

Some random variables are obtained by taking functions of another ones, possibly multidimensional. In the notes we often limit our attention to a single random variable Z given by a function $Z = \phi(X, Y)$ of two arguments; this is convenient for notation and exhibits most of the interesting techniques.

Sums and linear combinations, medians, maxima, and minima are perhaps the most often encountered functions of multidimensional random variables. Methods to compute the distribution, or the expected value of such a function are of considerable practical significance.

2.7 Moments of functions

If $Z = \phi(X, Y)$ has the expected value, then EZ can be computed directly without computing the density, or probability mass function of Z. The following identity is useful.

If X, Y are discrete and EZ exists, then

$$E\phi(X,Y) = \sum_{x,y} \phi(x,y)\Pr(X = x, Y = y). \tag{2.15}$$

If X, Y are jointly continuous then Z might be continuous, discrete, or say of mixed type. Regardless of the case

$$E\phi(X,Y) = \int\int_{\mathbb{R}^2} \phi(x,y) f(x,y)\,dxdy, \tag{2.16}$$

and the double integral converges if EZ exists. Conversely, if the integral converges, then EZ exists and is given by formula (2.16).

In particular the expected value is linear

$$E(aX + bY + c) = aEX + bEY + c. \tag{2.17}$$

This can be easily verified using (2.16), but the identity (2.16) is beyond the scope of this notes, as we do not want to dwell on the general definition of EZ that would encompass all cases.

The fact that expected value is linear provides a simple method of computing some otherwise difficult sums.

Example 2.14 *Suppose X is Binomial $Bin(n,p)$. Then $X = X_1 + \ldots + X_n$, where X_j is the number of successes in j-th trial. Clearly each X_j is 0, or 1, and $EX_j = p$.*

Exercise 2.12 (Example 1.3 continued) *Two drivers arrive at an intersection between 8:00 and 8:01 every day. On average how much time lapses between their arrivals?*

Definition 2.7.1 *Let $m = EX$. The variance $Var(X)$ is defined as $Var(X) = E(X - m)^2$.*

Notice that

$$Var(aX + b) = a^2 Var(X). \tag{2.18}$$

In particular, $Var(X) = 0$ when $X = const$. Sometimes a more convenient expression for the variance is

$$Var(X) = EX^2 - (EX)^2 = E(X - EX)^2.$$

The standard deviation is $\sigma = \sqrt{Var(X)}$.

Problem 2.13 *Compute variances $Var(X)$ for the entries in Table 2.4. (Some of these are a real challenge to your computational skills, so you may safely give up. Another method will make it easier in Chapter 3).*

Since $Var(X) = EX^2 - (EX)^2 \geq 0$, the following inequality follows

$$(EX)^2 \leq EX^2 \tag{2.19}$$

Name	Probability distribution	$Var(X)$
Normal	$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{(x-\mu)^2}{2\sigma^2}$	σ^2
Exponential	$f(x) = \lambda e^{-\lambda x}$	$\frac{1}{\lambda^2}$
Uniform	$f(x) = \frac{1}{b-a}$	$\frac{1}{12}(b-a)^2$
Gamma		
Weibull		
Binomial	$\Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$	$np(1-p)$
Poisson	$\Pr(X = k) = e^{-\lambda}\frac{\lambda^k}{k!}$	λ
Geometric	$\Pr(X = k) = p(1-p)^{k-1}$	$\frac{1}{p^2}$
Hypergeometric		
Negative Binomial		

Table 2.4: Variances of some random variables.

Definition 2.7.2 *The covariance of random variables X, Y with expected values m_X, m_Y is defined as $cov(X,Y) = E(X - m_X)(Y - m_Y)$.*

Clearly $Cov(X,X) = Var(X)$ and $Var(X+Y) = Var(X) + Var(Y) + 2cov(X,Y)$.

Theorem 2.7.1 *If X, Y are independent, then $Var(X+Y) = Var(X) + Var(Y)$.*

Example 2.15 *let $X_1, X_2, \ldots, X_n$ be independent $\{0,1\}$-valued random variables, and suppose that $\Pr(X_j = 1) = p$. Then $Var(\sum_{j=1}^n X_j) = np(1-p)$. What is the distribution of $\sum_{j=1}^n X_j$?*

Tail integration formula revisited

Example 2.16 *If $X > 0$ then $Ee^X = 1 + \int_0^\infty e^t P(X > t)\, dt$*

Problem 2.14 *Show that $EX^2 = \int_0^\infty tP(|X| > t)\, dt$.*
Generalize this formula to $E|X|^p$.

Chebyshev-Markov inequality

Special cases of Chebyshev's inequality (2.11) are:

$$\Pr(|X| > t) < \frac{1}{t} E|X| \tag{2.20}$$

$$\Pr(|X - \mu| > t) < \frac{1}{t^2} Var(X) \tag{2.21}$$

$$\Pr(|X| > t) < e^{-at} Ee^{aX} \tag{2.22}$$

Chebyshev's inequality is one reason we often strive for small average quadratic error. If $E|X - X_0|^2 < \epsilon$ then we can be sure that $\Pr(|X - X_0| > \sqrt[3]{\epsilon}) < \sqrt[3]{\epsilon}$. The following may be used (with some caution) in computer programs to asses error in estimating probabilities by sampling.

Example 2.17 *If X is $Bin(n,p)$ then* $\Pr(|\frac{X}{n} - p| > \frac{1}{\sqrt[3]{n}}) \leq \frac{1}{4\sqrt[3]{n}}$

Exercise 2.15 *Run a simulation of the event that you know probability of, printing out the error estimate.*

Do the same for the event that you don't know the probability analytically. (Use programs written for previous exercises)

2.7.1 Simulations

The unknown variance σ^2 of a sequence of simulated random variables X_j can be approximated by $\frac{1}{n}\sum_{j=1}^{n}(X_j - \bar{X})^2$, where $\bar{X}$ is the arithmetic mean of $X_1, \dots, X_n$. Thus can also use (2.21) and Theorem 2.7.1 to asses errors in estimating variances. Another more accurate method is presented later on in Chapter 5, but it also requires estimating variances.

From now on, in the output of your simulation programs you should provide some error estimates.

2.8 Application: scheduling

A critical path analysis involves estimating time of completing a project consisting of many tasks of varying lengths. Some of the tasks can be done concurrently, while other may begin only after other preliminary tasks are completed. This is modeled by the dependency graph together with the estimated times.

2.8.1 Deterministic scheduling problem

As an example of simple scheduling problem consider the following.

Example 2.18 *Suppose that we want to bake a batch of chocolate-chip cookies. The tasks and their (estimated) times are:*

T1 Bake at 350F (40 min)

T2 Make batter (5 min)

T3 Pre-heat oven to 350F (10 min)

T4 Find and grease pan (2 min)

T5 Find a cookie-tray to serve cookies (2 min)

T6 Take cookies out, cool and serve (5 min)

The dependency graph is quite obvious here; for instance, we cannot start baking before batter is ready. What is the shortest time we can eat the cookies?

2.8.2 Stochastic scheduling problem

In some projects, the actual numbers are only the averages, and the actual completion times of the projects may be random. The distribution of the actual completion time, or even its average may be difficult to compute analytically. Nevertheless, simulations let us estimate the average and analyze the probabilities of events.

Exercise 2.16 *Suppose for the sake of this exercise that the numbers presented in the cookie-baking example are just the average values of the exponential random variables.*

- *What will be the average completion time then? Will it be larger, smaller, or about equal to the previous answer?*
- *How often will we finish the process before the previously estimated (deterministic) time?*

2.8.3 More scheduling questions

In more realistic analysis of production processes we also have to decide how to split the tasks between available personnel.

Example 2.19 *This exercise refers to the tasks presented in Example 2.18. On average, how fast can a single person bake chocolate-chip cookies? What if there are two people?*

2.9 Distributions of functions

Distributions of functions of random variables are often difficult to compute explicitly. Special methods deal with more frequent cases.

Sums of discrete r. v.

Sums can be handled directly, but a more efficient method uses generating functions of Chapter 3.

Suppose X, Y are discrete and $f(x,y) = \Pr(X = x, Y = y)$ is their joint probability mass function. Then $Z = X + Y$ is discrete with values $z = x + y$. Therefore

$$f_Z(z) = \sum_x f(x, z - x) \tag{2.23}$$

For independent random variables X, Y this takes a slightly simpler form.

$$f_Z(z) = \sum_x f_X(x) f_Z(z - x) \tag{2.24}$$

Formula (2.24) can be used to prove the so called summation formulas.

Theorem 2.9.1 *If X, Y are independent Binomial with the same parameter p, ie. X is Bin(n,p) and Y is Bin(m, p), then $X + Y$ is Binomial Bin(n+m,p).*

If X, Y are independent Poisson Poiss(λ_X) and Poiss(λ_Y), then $X + Y$ is Poisson Poiss$(\lambda_X + \lambda_Y)$.

Sums of continuous r. v.

One can prove that if X, Y are independent and continuous than $X + Y$ is continuous with the density

$$f(z) = \int_{-\infty}^{\infty} f_X(u) f_Y(z-u)\, du \tag{2.25}$$

Formula (2.25) defines the *convolution* $f_X * f_Y$. It can be used to prove the so called summation formulas.

Theorem 2.9.2 *If X, Y are independent Normal, then $X + Y$ is Normal.*

If X, Y are independent Gamma with the same parameter β, then $X + Y$ is Gamma($\alpha_X + \alpha_Y, \beta$).

Example 2.20 (Example 1.3 continued) *Two drivers arrive at an intersection between 8:00 and 8:01 every day. What is the density of the time that lapsed between their arrivals?*

Example 2.21 *Suppose X, Y are independent $U(0,1)$. The density of $Z = X + Y$ is* $f(z) = \begin{cases} z & \text{if } 0 \le z \le 1 \\ 2-z & \text{if } 1 \le z \le 2 \end{cases}$.

Minima, maxima

Minima and maxima occur for instance if we are waiting for one of the independent events, and then we follow the first one (minimum), or the last one (maximum). Embedded Markov chains construction in Section 11.2 are based on minima of independent exponential r. v.

Suppose X, Y are independent. If $U = \min\{X, Y\}$ then $\Pr(U > t) = \Pr(X > t)\Pr(Y > t)$. Therefore the reliability function of U can be computed from the two given ones.

Example 2.22 *If X, Y are independent exponential, then $U = \min\{X, Y\}$ is exponential.*

If $U = \max\{X, Y\}$ then $\Pr(U < t) = \Pr(X < t)\Pr(Y < t)$. Therefore the cumulative distribution function of U can be computed from the two given ones.

Example 2.23 *Suppose X, Y are independent uniform U(0,1). Then $U = \max\{X, Y\}$ has the density $f(u) = 2u$ for $0 < u < 1$.*

Problem 2.17 *Let $U_1, \dots, U_n$ be independent uniform $U(0,1)$. Find the density of*

- $X = \min_j U_j$
- $Y = \max_j U_j$

Problem 2.18 *If X, Y are independent exponential random variables with parameters λ, μ, show that $\Pr(X < Y) = \frac{\lambda}{\lambda+\mu}$.*

Order statistics

Order statistics generalize minima and maxima. Their main use is in (robust) statistics, and this section can be safely skipped. Let $X_1, \dots, X_n$ be independent continuous random variables with the cumulative distribution function G and density $g = G'$.

Let $R_1, \dots, R_n$ be the corresponding *order statistics.* This means that at the end of each experiment we re-arrange the numbers $X_1, \dots X_n$ into the increasing sequence $R_1, \dots, R_n$. This means that $R_1 = \min_j X_j$ is the smallest, $R_2 = \max_i \min_{j \neq i} X_j$ is the second largest, etc.

The density of R_k can be found by the following method. In order for the inequality $R_k > x$ to hold, there must be at least k values among the X_j above level x. Since X_j are independent and have the same probability $p = \Pr(X_j > x)$ of "success" in crossing over the x-level, this means that $\Pr(R_k > x)$ is given by the binomial formula with n trials and probability of success $p = 1 - G(x)$.

$$\Pr(R_k > x) = \sum_{j=k}^{n} \tbinom{n}{j} (1 - G(x))^j (G(x))^{n-j} \tag{2.26}$$

When the derivative is taken, the sum collapses into just one term, giving the elegant answer $r_k(x) = \frac{n!}{(k-1)!(n-k)!} (G(x))^k (1 - G(x))^{n-k} g(x)$.

2.10 L_2-spaces

Inequalities related to expected values are best stated in the geometric language of norms and normed spaces. We say that $X \in L_2$, if X is *square integrable*, ie. $EX^2 < \infty$.

The L_2 norm is

$$\|X\|_2 = \sqrt{E|X|^2}.$$

Notice that $\|X - EX\|_2$ is just another notation for the standard deviation. Thus standard deviation is the L_2 distance of X from a constant.

We say that X_n converges to X in L_2, if $\|X_n - X\|_2 \to 0$ as $n \to \infty$. We shall also use the phrase *sequence X_n converges to X in mean-square.* An example of the latter is Theorem 5.2.1.

Several useful inequalities are collected in the following[4].

Theorem 2.10.1 *For all square-integrable X, Y*

- *Cauchy-Schwarz inequality:*

$$EXY \leq \|X\|_2 \|Y\|_2. \tag{2.27}$$

- *Jensen's inequality:*

$$E|X| \leq E\|X\|_2. \tag{2.28}$$

- *Triangle inequality:*

$$\|X + Y\|_2 \leq \|X\|_2 + \|Y\|_2. \tag{2.29}$$

[4] Theorem A.1.1 gives a more general statement.

Proof. Proof of (2.27): Quadratic function $f(t) = E|X+tY|^2 = EX^2 + 2tEXY + t^2EY^2$ is non-negative for all t. Therefore its determinant $\Delta \geq 0$. (Compute Δ to finish the proof.)

Proof of (2.28): Use (2.27) with $Y = 1$.

Proof of (2.29): By (2.27) $E|X+Y|^2 \leq \|X\|_2^2 + \|Y\|_2^2 + 2\|X\|_2\|Y\|_2$. □

2.11 Correlation coefficient

Correlation is a concept deeply rooted in statistics. The correlation coefficient $corr(X,Y)$ is defined for square-integrable non-degenerate r. v. X, Y by the formula

$$\rho = corr(X,Y) = \frac{EXY - EXEY}{\|X - EX\|_2 \|Y - EY\|_2}.$$

The Cauchy-Schwarz inequality (2.27) implies that $-1 \leq corr(X,Y) \leq 1$.

Random variables with $\rho = 0$ are called *uncorrelated.* Correlation coefficient close to one of the extremes ±1 means that there is a strong linear relation between X, Y; this is stated more precisely in (2.31).

Theorem 2.7.1 states that independent random variables with finite variances are uncorrelated.

Problem 2.19 *Give an example of dependent random variables that are uncorrelated.*

2.11.1 Best linear approximation

Suppose we would like to approximate random variable Y by another quantity X that is perhaps better accessible. Of all the possible ways to do it, linear function $Y \approx mX + b$ is perhaps the simplest. Of all such linear functions, we now want to pick the best. In a single experiment, the error is $|Y - mX - b|$. We could minimize the average empirical error over many experiments $\frac{1}{n}\sum_j |Y_j - mX_j - b|$. This approximates the average error $E|Y - mX - b|$. Let us agree to measure the error of the approximation by a quadratic error $E(Y - mX - b)^2$ instead. (This choice leads to simpler mathematics.)

Question: For what values of m, b the error $E(Y - mX - b)^2$ is the smallest? When is it 0?

Let $H(m,b) = E(Y - mX - b)^2$. Clearly, H is a quadratic function of $m, b \in \mathbb{R}$. The unique minimum is determined by the system of equations

$$\begin{aligned} \frac{\partial H}{\partial m} &= 0 \\ \frac{\partial H}{\partial b} &= 0. \end{aligned} \tag{2.30}$$

The answer is $m = \frac{cov(X,Y)}{VarX}$, $b = EY - mEX$, and the minimal error is

$$(1 - \rho^2)Var(Y). \tag{2.31}$$

2.12 Application: length of a random chain

A chain in the x, y-plane consists of n links each of unit length. The angle between two consecutive links is $\pm\alpha$, where $\alpha > 0$ is a constant. Assume the sign is taken and random, with probability $\frac{1}{2}$ for each. Let L_n be the distance from the beginning to the end of the chain. The angle between the k-th link and the positive x-axis is a random variable S_{k-1}, where we may assume (why?) $S_0 = 0$ and $S_k = S_{k-1} + \xi_k\alpha$, where $\xi = \pm 1$ with probability $\frac{1}{2}$. The following steps determine the average length of the random chain.

1. $L_n^2 = (\sum_{k=0}^{n-1} \cos S_k)^2 + (\sum_{k=0}^{n-1} \sin S_k)^2$.
2. $E \cos S_n = \cos^n \alpha$
3. $E \sin S_n = 0$
4. $E \cos S_m \cos S_n = \cos^{n-m} \alpha E \cos^2 S_m$ for $m < n$
5. $E \sin S_m \sin S_n = \cos^{n-m} \alpha E \sin^2 S_m$ for $m < n$
6. $EL_n^2 - L_{n-1}^2 = 1 + 2\cos\alpha \frac{1-\cos^{n-1}\alpha}{1-\cos\alpha}$
7. $EL_n^2 = n\frac{1+\cos\alpha}{1-\cos\alpha} - 2\cos\alpha \frac{1-\cos^n\alpha}{(1-\cos\alpha)^2}$

2.13 Conditional expectations

2.13.1 Conditional distributions

For discrete X, Y the conditional distribution of variable Y given the value of X is just the conditional probability, $\Pr(Y = y|X = x)$. In jointly continuous case, define the conditional density

$$f(y|X = x) = \frac{f(x, y)}{f_X(x)}.$$

Conditional density $f(y|X = x)$ is defined only for x such that $f_X(x) > 0$; this is a reasonable approach for the most often encountered continuous, or piecewise continuous densities. Since the densities are actually the elements of L_1 space rather than functions, special care is needed in the definition of the conditional density. In fact the theory of probability is often developed without the reference to conditional distributions.

Definition 2.13.1 *The conditional expectation $E\{X|Y = y\}$ is defined as $\sum x \Pr(X = x|Y = y)$ in discrete case, and as $\int_{\mathbb{R}} x f(x|Y = y)\, dx$ in the continuous case. One can show that the expected values exist, when $E|X| < \infty$.*

Example 2.24 *A game consists of tossing a die. If the face value on the die is X then a coin is tossed X times. Let Y be the number of heads. Then $E(Y|X = x) = \frac{1}{2}x$.*

2.13.2 Conditional expectations as random variables

Since $E(X|Y=y)$ depends on the actual value of Y, and Y is random, the conditional expectation is a random variable itself. We shall write $E\{X|Y\}$ or $E^Y X$ for the random variable defined by the conditional expectation $E\{X|Y=y\}$.

Example 2.25 *Suppose Y is a discrete with different values on the events $A_1, A_2, \dots, A_n$ which form a non-degenerate disjoint partition of the probability space Ω. Then*

$$E\{X|Y\}(\omega) = \sum_{k=1}^{n} m_k I_{A_k}(\omega),$$

where $m_k = \int_{A_k} X\,dP/P(A_k)$. In other words, on A_k we have $E\{X|\mathcal{F}\} = \int_{A_k} X\,dP/P(A_k)$. In particular, if X is discrete and $X = \sum x_j I_{B_j}$, then we get intuitive expression

$$E\{X|\mathcal{F}\} = \sum x_j P(B_j|A_k) \text{ for } \omega \in A_k.$$

Example 2.26 *Suppose that $f(x,y)$ is the joint density with respect to the Lebesgue measure on $\mathbb{R}^2$ of the bivariate random variable (X,Y) and let $f_Y(y) \neq 0$ be the (marginal) density of Y. Put $f(x|y) = f(x,y)/f_Y(y)$. Then $E\{X|Y\} = h(Y)$, where $h(y) = \int_{-\infty}^{\infty} x f(x|y)\,dx$.*

Total probability formula for conditional expectations is as follows.

$$EY = E(E(Y|X)) \tag{2.32}$$

compare (1.10).

Example 2.27 *In Example 2.24, $EY = 3/2$.*

2.13.3 Conditional expectations (continued)

In discrete case conditional expectations of functions are given by,

$$E(g(X)|Y=y) = \sum_x g(x)\Pr(X=x|Y=y) \tag{2.33}$$

The following version of total probability formula is often useful.

$$Eg(X) = E(E(g(X)|Y)) \tag{2.34}$$

Example 2.28 *Suppose N is Binomial $Bin(m,q)$ and given the value of N, r. v. X is Bin(N,p). What is the distribution of X?*

Similar question can be solved for N having a Poisson distribution.

Example 2.29 *What is the distribution of a geometric sum of i. i. d. exponential r. v.?*

Example 2.30 *Stock market fluctuations can be modelled by $Z = \xi_1 + \ldots + \xi_N$, where N, the number of transactions, is Poisson(λ) and ξ are normal $N(0,\sigma)$. There is no explicit formula for the density of Z, but there is one for the moment generating function. Thus Chebyshev inequality gives bounds of the form $\Pr(Z > t) \leq \exp \ldots$.*

2.14 Best non-linear approximations

This section explains the relevance of conditional expectations to the problem of best mean-square approximation.

Theorem A.2.1 gives geometric interpretation of the conditional expectation $E\{\cdot|Z\}$; for square integrable functions $E\{.|Z\}$ is just the orthogonal projection of the Banach (normed) space L_2 onto its closed subspace $L_2(Z)$, consisting of all 2-integrable random variables of the form $f(Z)$.

Theorem 2.14.1 *For square integrable Y the quadratic error $E(Y-h(X))^2$ among all square integrable functions $h(X)$ is the smallest if $h(x)=E(Y|X=x)$.*

Theorem 2.14.1 implies that the linear approximation from Section 2.11.1 is usually less accurate. In Chapter 10 we shall see that linear approximation is the best one can get in the all important normal case. Even in non-normal case linear approximations offer quick solutions based on simple second order statistics. In contrast, the non-linear approximations require elaborate numerical schemes to process the empirical data.

2.15 Lack of memory

Conditional probabilities help us to arrive at important classes of densities in modeling. In this section we want to analyze an non-aging device, which characteristics do not change with time.

Suppose T represents a failure time of some device. If the device is working at time t, then the probability of surviving additional s seconds is $\Pr(T>t+s|T>t)$. For a device that doesn't exhibit aging this probability should be the same as for the brand new device.

$$\Pr(T>t+s|T>t)=\Pr(T>s) \tag{2.35}$$

Problem 2.20 *Show that exponential T satisfies (2.35).*

Problem 2.21 *Show that geometric T satisfies (2.35) for integer t,s.*

Equation (2.35) implies $\Pr(T>t+s)=\Pr(T>t)\Pr(T>s)$, an equation that can be solved.

Theorem 2.15.1 *If $T>0$ satisfies (2.35) for all $t,s>0$, then T is exponential.*

Proof. The tail distribution function $N(x)=P(T>x)$ satisfies equation

$$N(x+y)=N(x)N(y) \tag{2.36}$$

for arbitrary $x,y>0$. Therefore to prove the theorem, we need only to solve functional equation (2.36) for the unknown function $N(\cdot)$ under the conditions that $0\leq N(\cdot)\leq 1$, $N(\cdot)$ is left-continuous, non-increasing, $N(0^+)=1$, and $N(x)\to 0$ as $x\to\infty$.

Formula (2.36) implies that for all integer n and all $x\geq 0$

$$N(nx)=N(x)^n. \tag{2.37}$$

Since $N(0^+) = 1$ and $N(\cdot)$, it follows from (2.37) that $r = N(1) > 0$. Therefore (2.37) implies $N(n) = r^n$ and also $N(1/n) = r^{1/n}$ (to see this, plug in (2.37) values $x = 1$ and $x = 1/n$ respectively). Hence $N(n/m) = N(1/m)^n = r^{n/m}$ (by putting $x = 1/m$ in (2.37)).

This shows that for rational $q > 0$

$$N(q) = r^q. \tag{2.38}$$

Since $N(x)$ is left-continuous (why?), $N(x) = \lim_{q \nearrow x} N(q) = r^x$ for all $x \geq 0$. It remains to notice that since $N(x) \to 0$ as $x \to \infty$, we have $r < 1$. Therefore $r = \exp(-\lambda)$ for some $\lambda > 0$ and $N(x) = \exp(-\lambda x), x \geq 0$. □

Remark 1 *Geometric distribution also has the lack of memory property. If equation (2.36) is assumed to hold for integer values of x, y only, and $T > 0$ is integer valued, then T is geometric.*

2.16 Intensity of failures

The intuitive lack-of-memory, or non-aging property of the exponential distribution can be generalized to include simple models of aging. We may want to assume that a component analyzed becomes less reliable, or more reliable with time. An example of the first one is perhaps a brand new car. An example of the latter is perhaps a software operating system when updates are installed promptly.

Let $T > 0$ be a continuous r. v. interpreted as a failure time of a certain device. If the device is in operational condition at time t, then the probability that it will fail immediately afterwards may be assumed negligible. The probability of failing within h units of time is $\Pr(T < t + h|T > t)$. The failure rate at time t is defined as

$$\lambda(t) = \lim_{h \to 0} \frac{1}{h} \Pr(T < t + h|T > t) \tag{2.39}$$

Example 2.31 *If T is exponential then the failure rate is constant.*

A family of failure rates that exhibit interesting aging patterns is provided by the family of power functions $\lambda(t) = t^a$.

Theorem 2.16.1 *If T is continuous with failure rate $\lambda(t) = t^a$, where $a > 0$ then T has the Weibull density:*

$$f(t) = Ct^{a-1}e^{-bt^a} \text{ for } t > 0. \tag{2.40}$$

(Here $C = ab$ is the normalization constant).

2.17 Poisson approximation

Of the discrete distributions, the formula for the Poisson distribution is perhaps mysterious. Poisson distribution is often called *the law of rare events.*

Theorem 2.17.1 *Suppose X_n are $Bin(n, p_n)$ and $np_n \to \lambda$. Then* $\Pr(X_n = k) \to e^{-\lambda}\lambda^k/k!$.

Proof. Rewrite the expression $\binom{n}{k}\frac{\lambda^k}{n^k}(1-\frac{\lambda}{n})^{n-k} = \lambda^k/k!(1-\lambda/n)^n/(1-\lambda/n)^k\prod_{j=0}^{k}(1-j/n)$. □

Example 2.32 *How many raisins should a cookie have on average so that no more than one cookie in a hundred has no raisins? So that no more than one cookie in a thousand has no raisins?*

Example 2.33 *A bag of chocolate chip cookies has 50 cookies. The manufacture claims there are a 1,000 chips in a bag. Is it likely to find a cookie with 15 or less chips in such a bag?*

2.18 Questions

Problem 2.22 *The performance of the algorithm for selecting a random permutation in* `GetPermutation` *SUB of* `RANDTOUR.BAS` *can be estimated by the following.*

From numbers $1, \ldots n$*, select at random* $k > n$ *numbers. On average, how many of these numbers repeat? (and hence should be thrown out)*

Problem 2.23 *The performance of the algorithm for selecting a random permutation in* `GetPermutation` *SUB of* `RANDTOUR.BAS` *can be estimated by analyzing the following "worst-case" scenario.*

When the algorithm attempts to select <u>*last*</u> *of the random numbers* $1, \ldots n$*, then*

1. *What is the probability if will find the "right number" on first attempt?*

2. *How many attempts on average does the algorithm take to find the last random number?*

Exercise 2.24 *What is the probability that in the group of* 45 *people one can find two born on the same day of the year? (Compare Example 1.14).*

Problem 2.25 *For a group of n person, find the expected number of days of the year which are birthdays of exactly k people. (Assume 365 days in a year and that all birthdays are equally likely.)*

Problem 2.26 *A large number N of people are subject to a blood test. The test can be administered in one of the two ways:*

(i) Each person can be tested separately (N tests are needed).

(ii) The blood sample of k people can be pooled (mixed) and analyzed. If the test is positive, each of the k people must be tested separately and in all $k + 1$ *tests are then required for k people.*

Assume the probability p that the test is positive is the same for all people and that the test results for different people are stochastically independent.

1. *What is the probability that the test for a pooled sample of k people is positive?*
2. *What is the expected number of tests necessary under plan (ii)?*
3. *Find the equation for the value of k which will minimize the expected number of tests under plan (ii).*
4. *Show that the optimal k is close to $\frac{1}{\sqrt{p}}$ and hence that the minimum expected number of tests is on average about $\frac{2N}{\sqrt{p}}$.*

Exercise 2.27 *Solve Exercise 1.3 on page 5 assuming that the arrival times are exponential rather than uniform. Assume independence.*

Exercise 2.28 *There are 3 stop-lights spaced within 1km of each other and operating asynchronously. (They are reset at midnight.) Assuming each is red for 1 minute and then green for one minute, what is the average time to pass through the three lights by a car that can instantaneously accelerate to 60km/h.*

This exercise can be developed into a simulation project that may address some of the following questions

- *How does the speed change with the number of lights?*
- *How does the answer change if a car has finite acceleration?*
- *Are the answers different, if each green light lasts random amount of time, 1min on average?*
- *How to model/simulate more than one car?*
- *Can you simulate car traffic on a square grid with stop-lights at intersections?*

More theoretical questions

Problem 2.29 (Hoeffding) *Show that if XY, X, Y are discrete, then*

$$EXY - EXEY = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (P(X \geq t, Y \geq s) - P(X \geq t)P(Y \geq s))\, dt\, ds.$$

Problem 2.30 *Let $X \geq 0$ be a random variable and suppose that for every $0 < q < 1$ there is $T = T(q)$ such that*

$$P(X > 2t) \leq qP(X > t) \text{ for all } t > T.$$

Show that all the moments of X are finite.

Problem 2.31 *If ξ, U are independent, $\Pr(\xi = 0) = \Pr(\xi + 1) = \frac{1}{2}$ and U is uniform $U(0,1)$. What is the distribution of $U + \xi$?*

Chapter 3

Moment generating functions

3.1 Generating functions

Properties of a sequence $\{a_n\}$ are often reflected in properties of the *generating function* $h(z) = \sum_n a_n z^n$.

3.2 Properties

The moment generating function of a real-valued random variable X is defined by $M_X(t) = E\exp(tX)$. If $X > 0$ has the density $f(x)$, the moment generating function is its Laplace transform: $M(t) = \int_0^\infty e^{tx} f(x)\, dx$.

A moment generating function is non-negative, and convex (concave up). The typical example is the moment generating function of the $\{0, 1\}$-valued random variable.

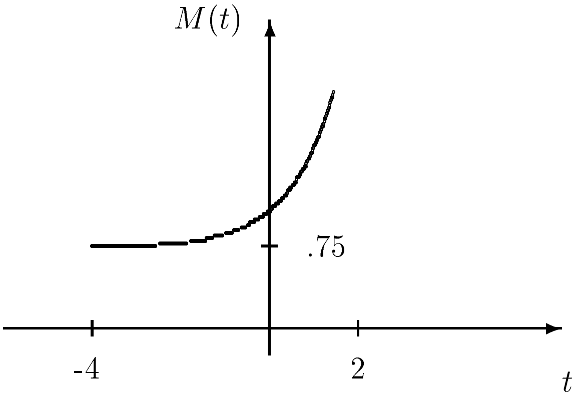

Figure 3.1: Graph of the moment generating function $M(t) = \frac{3}{4} + \frac{1}{4}e^t$.

A linear transformations of X changes the moment generating function by the following formula.

$$M_{aX+b}(t) = e^{tb} M_X(at). \tag{3.1}$$

Important properties of moment generating functions are proved in more theoretical probability courses[1].

Theorem 3.2.1 *(i) The distribution of X is determined uniquely by its moment generating function $M(t)$.*

(ii) If X, Y are independent random variables, then $M_{X+Y}(t) = M_X(t) M_Y(t)$ for all $t \in \mathbb{R}$.

(iii) $M(0) = 1, M'(0) = EX, M''(0) = EX^2$

Name	Distribution	Moment generating function
Normal N(0,1)	$f(x) = \frac{1}{\sqrt{2\pi}} \exp - \frac{x^2}{2}$	$M(t) = e^{t^2/2}$
Exponential	$f(x) = \lambda e^{-\lambda x}$	$M(t) = \frac{\lambda}{\lambda - t}$
Uniform $U(-1,1)$	$f(x) = \frac{1}{2}$ for $-1 \leq x \leq 1$	$M(t) = \frac{1}{t} \sinh t$
Gamma	$f(x) = 1/, (\alpha) \beta^{-\alpha} x^{\alpha - 1} \exp -x/\beta$	$M(t) = (1 - \beta t)^{-\alpha}$
Binomial	$\Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$	$M(t) = (1 - p + pe^t)^n$
Poisson	$\Pr(X = k) = e^{-\lambda} \frac{\lambda^k}{k!}$	$M(t) = \exp \lambda(e^t - 1)$
Geometric	$\Pr(X = k) = p(1-p)^{k-1}$	$M(t) = \frac{pe^t}{1-(1-p)e^t}$

Table 3.1: Moment generating functions.

Problem 3.1 *Find moment generating functions for each of the entries in Table 3.1.*

Problem 3.2 *Use moment generating functions to compute $EX, Var(X)$ for each of the entries in Table 2.4.*

Problem 3.3 *Prove the summation formulas stated in Theorems 2.24 and 2.25*

For a d-dimensional random variable $\mathbf{X} = (X_1, \dots, X_d)$ the moment generating function $M_{\mathbf{X}} : \mathbb{R}^d \to \mathbb{C}$ is defined by $M_{\mathbf{X}}(\mathbf{t}) = E\exp(\mathbf{t} \cdot \mathbf{X})$, where the dot denotes the dot (scalar) product, ie. $\mathbf{x} \cdot \mathbf{y} = \sum x_k y_k$. For a pair of real valued random variables X, Y, we also write $M(t, s) = M_{(X,Y)}((t, s))$ and we call $M(t, s)$ the joint moment generating function of X and Y.

The following is the multi-dimensional version of Theorem 3.2.1.

Theorem 3.2.2 *(i) The distribution of $\mathbf{X}$ is determined uniquely by its moment generating function $M(\mathbf{t})$.*

(ii) If $\mathbf{X}, \mathbf{Y}$ are independent $\mathbb{R}^d$-valued random variables, then

$$M_{\mathbf{X}+\mathbf{Y}}(\mathbf{t}) = M_{\mathbf{X}}(\mathbf{t}) M_{\mathbf{Y}}(\mathbf{t})$$

for all $\mathbf{t}$ in $\mathbb{R}^d$.

[1] See eg. W. Feller, *An Introduction to Probability Theory*, Vol II, Wiley, New York 1966.

3.2.1 Probability generating functions

For $\mathbb{Z}_+$-valued random variables it is convenient to consider the so called *generating function* $G(z) = M(\ln z)$. In this case Theorem 3.2.1(i) is elementary, as $G(z) = \sum_{k=0}^{\infty} p_k z^k$ determines uniquely its Taylor series coefficients $p_k = \Pr(X = k)$.

3.3 Characteristic functions

The major nuisance in using the moement generating functions is the fact that the moment generating functions may not exist, when the definiting integral diverges.

For this reason it is more preferable to use an expression that is always bounded, and yet has the same convenient algebraic properties. The natural candidate is

$$e^{ix} = \cos x + i \sin x.$$

The *characteristic function* is accordingly defined as

$$\phi_X(t) = Ee^{itX}. \tag{3.2}$$

For symmetric random variables complex numbers can be avoided at the expense of trigonometric identities.

Example 3.1 *If X is $-1, 1$ valued,* $\Pr(X = 1) = \frac{1}{2}$, *then* $\phi(t) = \cos t$.

Example 3.2 *The characteristic function of the normal $N(0, 1)$ distribution is $\phi(t) = e^{-t^2/2}$.*

3.4 Questions

Problem 3.4 *Let $S_n = X_1 + \ldots + X_n$ be the sum of mutually independent random variables each assuming the values $1, 2, \ldots, a$ with probability $\frac{1}{a}$.*

1. *Show that* $Ee^{uS_n} = \left(\frac{e^u(1-e^{au})}{a(1-e^u)}\right)^n$.
2. *Use the above identity to show that for $k \geq n$*

$$\Pr(S_n = k) = a^{-n} \sum_{j=0}^{\infty} (-1)^j \binom{n}{j} \binom{k-aj-1}{n-1}$$

(For $a = 6$ $\Pr(S_n = k)$ is the probability of scoring the sum $k + n$ in a throw with n dice. The solution of this problem is due to de Moivre.)

Problem 3.5 *Suppose the probability p_n that a family has exactly n children is αp^n when $n \geq 1$ and suppose $p_0 = 1 - \alpha\frac{p}{1-p}$. (Notice that this is a constaint on the admissible values of α, p since $p_0 \geq 0$.*

Suppose that all distributions of the sexes for n children are equally likely. Find the probability that a family has exactly k girls.

Hint: The answer is at first as the infinite series. To find its sum, use generating function, or negative binomial expansion: $(1 + x)^{-k} = 1 + \frac{k+1}{1!}x + \frac{(k+1)k+2}{2!}x^2 + \ldots$.

$\boxed{\text{ANS: } \frac{2\alpha p^k}{(2-p)^{k+1}}}$.

Problem 3.6 *Show that if $X \geq 0$ is a random variable such that*

$$P(X > 2t) \leq 10\,(P(X > t))^2 \text{ for all } t > 0,$$

then $E\exp(\lambda|X|) < \infty$ for some $\lambda > 0$.

Problem 3.7 *Show that if $E\exp(\lambda X^2) = C < \infty$ for some $a > 0$, then*

$$E\exp(tX) \leq C\exp(\frac{t^2}{2\lambda})$$

for all real t.

Problem 3.8 *Prove that function $\phi(t) := E\max\{X, t\}$ determines uniquely the distribution of an integrable random variable X in each of the following cases:*

(a) If X is discrete.

(b) If X has continuous density.

Problem 3.9 *Let $p > 2$. Show that $\exp(|t|^p)$ is <u>not</u> a moment generating function.*

Chapter 4

Normal distribution

> *Next to a stream in a forest you see a small tree with tiny, bell-shaped, white flowers in dropping clusters.*
> The Auborn Society Field Guide to North American Trees.

The predominance of normal distribution is often explained by the Central Limit Theorem, see Section5.4. This theorem asserts that under fairly general conditions the distribution of the sum of many independent components is approximately normal. In this chapter we give another reason why normal distribution might occur. The usual definition of the standard normal variable Z specifies its density $f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$, see Figure 2.1 on page 24. In general, the so called $N(m, \sigma)$ density is given by

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m)^2}{2\sigma^2}}.$$

By completing the square one can check that the moment generating function $M(t) = Ee^{tZ} = \int_{-\infty}^{\infty} e^{itx} f(x)\, dx$ of the standard normal r. v. Z is given by

$$M(t) = e^{\frac{t^2}{2}}.$$

In multivariate case it is more convenient to use moment generating functions directly. For consistency we shall therefore adopt the following definition.

Definition 4.0.1 *A real valued random variable X has the normal $N(m, \sigma)$ distribution if its moment generating function has the form*

$$M(t) = \exp(tm + \frac{1}{2}\sigma^2 t^2),$$

where m, σ are real numbers.

From Theorem 3.2.1 one can check by taking the derivatives that $m = EX$ and $\sigma^2 = Var(X)$. Using (3.1) it is easy to see that every univariate normal X can be written as

$$X = \sigma Z + m, \tag{4.1}$$

where Z is the standard $N(0,1)$ random variable with the moment generating function $e^{\frac{t^2}{2}}$. This is perhaps the most convenient representation[1] of the general univariate normal distribution. Traditionally, it was used to answer questions like

> If X has given mean $\mu = 123$ and given variance $\sigma^2 = 456$, for what values of a we have $\Pr(|X| > a) = .8$?

with the help of tabularized values of the cumulative distribution function of standard normal Z.

Exercise 4.1 (Basketball coach's problem) *Collect data on the heights of 25 to 50 randomly selected males. Make a histogram of the data, compute the empirical mean and standard deviation.*

- *Does it appear that the normal distribution is a good probability distribution for these heights?*
- *There are about 200,000 males in Cincinnati area. Assuming normal distribution of heights with the mean and variance as you obtained from the data, estimate the number of males taller than $7'$.*

4.1 Herschel's law of errors

The following narrative comes from J. F. W. Herschel[2].

> *"Suppose a ball is dropped from a given height, with the intention that it shall fall on a given mark. Fall as it may, its deviation from the mark is error, and the probability of that error is the unknown function of its square, ie. of the sum of the squares of its deviations in any two rectangular directions. Now, the probability of any deviation depending solely on its magnitude, and not on its direction, it follows that the probability of each of these rectangular deviations must be the same function of its square. And since the observed oblique deviation is equivalent to the two rectangular ones, supposed concurrent, and which are essentially independent of one another, and is, therefore, a compound event of which they are the simple independent constituents, therefore its probability will be the product of their separate probabilities. Thus the form of our unknown function comes to be determined from this condition..."*

Ten years after Herschel, the reasoning was repeated by J. C. Maxwell[3]. The fact that velocities are normally distributed is sometimes called Maxwell's theorem.

The beauty of the reasoning lies in the fact that the interplay of two very natural assumptions: of independence and of rotation invariance, gives rise to the *normal law of errors* — the most important distribution in statistics.

[1] For the multivariate analog, see Theorem 10.4.2

[2] J. F. W. Herschel, *Quetelet on Probabilities*, Edinburgh Rev. 92 (1850) pp. 1–57

[3] J. C. Maxwell, *Illustrations of the Dynamical Theory of Gases*, Phil. Mag. 19 (1860), pp. 19–32. Reprinted in *The Scientific Papers of James Clerk Maxwell*, Vol. I, Edited by W. D. Niven, Cambridge, University Press 1890, pp. 377–409.

Theorem 4.1.1 *Suppose random variables X, Y have joint probability distribution $\mu(dx, dy)$ such that*

(i) $\mu(\cdot)$ is invariant under the rotations of $\mathbb{R}^2$;

(ii) X, Y are independent.

Then X, Y are normal.

The following technical lemma asserts that moment generating function exists.

Lemma 4.1.2 *If X, Y are independent and $X+Y, X-Y$ are independent, then $E \exp aX < \infty$ for all a.*

Proof. Consider a real function $N(x) := P(|X| \geq x)$. We shall show that there is x_0 such that

$$N(2x) \leq 8(N(x-x_0))^2 \tag{4.2}$$

for each $x \geq x_0$. By Problem 3.6 this will end the proof.

Let X_1, X_2 be the independent copies of X. Inequality (4.2) follows from the fact that event $\{|X_1| \geq 2x\}$ implies that either the event $\{|X_1| \geq 2x\} \cap \{|X_2| \geq 2x_0\}$, or the event $\{|X_1 + X_2| \geq 2(x-x_0)\} \cap \{|X_1 - X_2| \geq 2(x-x_0)\}$ occurs.

Indeed, let x_0 be such that $P(|X_2| \geq 2x_0) \leq \frac{1}{2}$. If $|X_1| \geq 2x$ and $|X_2| < 2x_0$ then $|X_1 \pm X_2| \geq |X_1| - |X_2| \geq 2(x-x_0)$. Therefore using independence and the trivial bound $P(|X_1 + X_2| \geq 2a) \leq P(|X_1| \geq a) + P(|X_2| \geq a)$, we obtain

$$P(|X_1| \geq 2x) \leq P(|X_1| \geq 2x)P(|X_2| \geq 2x_0)$$

$$+P(|X_1 + X_2| \geq 2(x-x_0))P(|X_1 - X_2| \geq 2(x-x_0))$$

$$\leq \frac{1}{2}N(2x) + 4N^2(x-x_0)$$

for each $x \geq x_0$. □

Proof of Theorem 4.1.1. Let $M(u) = E^{uX}$ be the moment generating function of X. Since $Ee^{u(X+Y)+v(X-Y)} = Ee^{(u+v)X+(u-v)Y}$

$$M(\sqrt{2}u)M(\sqrt{2}v) = M(u+v)M(u-v) \tag{4.3}$$

This implies that $Q(x) = \ln M(x)$ satisfies

$$Q(\sqrt{2}u) + Q(\sqrt{2}v) = Q(u+v) + Q(u-v) \tag{4.4}$$

Differentiating (4.4) with respect to u and then v we get

$$Q''(u+v) = Q''(u-v)$$

Therefore (take $u = v$) the second derivative $Q''(u) = Q''(0) = const \geq 0$. This means $M(u) = \exp(\alpha u + \beta u^2)$. □

4.2 Bivariate Normal distribution

Definition 4.2.1 *We say that X, Y have jointly normal distribution (bivariate normal), if $aX + bY$ is normal for all $a, b \in \mathbb{R}$.*

If $EX = EY = 0$, the moment generating function $M(t, s) = Ee^{tX+sY}$ is given by $M(t, s) = e^{\frac{1}{2}(\sigma_1^2 t^2 + 2ts\rho\sigma_1\sigma_2 + s^2 t^2)}$

Clearly $\sigma_1^2 = Var(X), \sigma_2^2 = Var(Y), \rho = Cov(X, Y)$.

When $\rho \neq \pm 1$ the joint density of X, Y exists and is given by

$$f(x, y) = \frac{1}{\sqrt{2\pi(1-\rho^2)\sigma_1\sigma_2}} \exp -\frac{x^2 - y^2 - 2\rho\sigma_1\sigma_2 xy}{2\sigma_1^2\sigma_2^2(1-\rho^2)}. \tag{4.5}$$

Example 4.1 *If X, Y are jointly normal with correlation coefficient $\rho \neq \pm 1$ then the conditional distribution of Y given X is normal.*

4.3 Questions

Problem 4.2 *Show that if X, Y are independent and normal, then $X + Y$ is normal. (Hint: moment generating functions are easier than convolution formula (2.24).)*

Problem 4.3 *If X, Y are independent normal $N(0, 1)$, find the density of $X^2 + Y^2$. (Hint: compute cumulative distribution function, integrating in polar coordinates.)*

Problem 4.4 *For jointly normal X, Y show that $E(Y|X) = aX + b$ is linear.*

Problem 4.5 *If X, Y are jointly normal then $Y - \rho X\sigma_Y/\sigma_X$ and X are independent.*

Problem 4.6 *If X, Y are jointly normal with variances σ_X^2, σ_Y^2 and the correlation coefficient ρ, then $X = \sigma_X(\gamma_1 \cos\theta + \gamma_2 \sin\theta$, $Y = \sigma_Y(\gamma_1 \sin\theta + \gamma_2 \cos\theta$, where γ_j are independent $N(0, 1)$ and $\sin 2\theta = \rho$.*

Chapter 5

Limit theorems

This is a short chapter on asymptotic behavior of sums and averages of independent observations. Theorem 5.2.1 justifies simulations as the means for computing probabilities and expected values. Theorem 5.4.2 provides error estimates.

5.1 Stochastic Analysis

There are several different concepts of convergence of random variables.

Definition 5.1.1 *$X_n \to X$ in probability, if* $\Pr(|X_n - X| > \epsilon) \to 0$ *for all* $\epsilon > 0$

Example 5.1 *Let X_n be $N(0, \sigma = \frac{1}{n})$. Then $X_n \to o$ in probability.*

Definition 5.1.2 *$X_n \to X$ almost surely, if* $\Pr(X_n \to X) = 1$.

Example 5.2 *Let U be the uniform U(0,1) r. v. Then $\frac{1}{n}U \to 0$ almost surely.*

Example 5.3 *Let U be the uniform U(0,1) r. v. et* $X_n = \begin{cases} 0 & \text{if } U > \frac{1}{n} \\ 1 & \text{otherwise} \end{cases}$. *Then $X_n \to 0$ almost surely.*

Definition 5.1.3 *$X_n \to X$ in L_2 (mean square), if $E|X_n - X|^2 \to 0$ as $n \to \infty$.*

Remark 2 *If $X_n \to X$ in L_2, then by Chebyshev's inequality $X_n \to X$ in probability.*
If $X_n \to X$ almost surely, then $X_n \to X$ in probability.

5.2 Law of large numbers

Each law of large numbers (there are many of them) states that empirical averages converge to the expected value. In statistical physics the law of large numbers impies that trajectory averages and population averages are asymptoticaly the same. In simulations, it provides a theoretical foundation, and connects the frequency with the probability of an event.

Theorem 5.2.1 *Suppose X_k are such that $EX_k = \mu, Var(X_k) = \sigma^2$, and $cov(X_i, X_j) = 0$ for all $i \neq j$. Then $\frac{1}{n}\sum_{j=1}^n X_j \to \mu$ in L_2*

Proof. To show that $\frac{1}{n}\sum_{j=1}^n X_j \to \mu$ in mean square, compute the variance. □

Corollary 5.2.2 *If X_n is Binomial $Bin(n,p)$ then $\frac{1}{n}X_n \to p$ in probability.*

5.2.1 Strong law of large numbers

The following method can be used to prove strong law of large numbers for Binomial r. v.

1. If X is $Bin(n,p)$, use moment generating function to show that $E(X-np)^4 \leq Cn^2$
2. Use Chebyshev's inequality and fourth moments to show that $\sum_n \Pr(|X_n/n - p| > \epsilon) < \infty$.
3. Use the convergence of the series to show that $\lim_{N\to\infty} \Pr(\bigcup_{n\geq N}\{|X_n/n - p| > \epsilon\}) = 0$.
4. Use the continuity of the probability measure to show that $\Pr(\bigcap_N \bigcup_{n\geq N}\{|X_n/n - p| > \epsilon\}) = 0$.
5. Show that with probability one for every rational $\epsilon > 0$ there is $N = N(\epsilon)$ such that for all $n > N$ the inequality $|X_n/n - p| < \epsilon$ holds. Hint: if $\Pr(A_\epsilon) = 1$ for rational ϵ, then $\Pr(\bigcap_\epsilon A_\epsilon) = 1$.

5.3 Convergence of distributions

In addition to the types of convergence introduced in Section 5.1, we also have the *convergence in distribution.*

Definition 5.3.1 *X_n converges to X in distribution, if $Ef(X_n) \to Ef(X)$ for all bounded continuous functions f.*

Theorem 5.3.1 *If $X_n \to X$ in distribution, then $\Pr(X_n \in (a,b)) \to \Pr(X \in (a,b))$ for all $a < b$ such that $\Pr(X = a) = \Pr(X = b) = 0$.*

Theorem 5.3.2 *If $M_{X_n}(u) \to M_X(u)$ for all u, then $X_n \to X$ in distribution.*

Theorem 2.17.1 states convergence in distribution to Poisson limit. Here is a proof that uses moment generating functions.
Proof of Theorem 2.17.1. $M_{X_n}(u) = (1 + p_n(e^u - 1))^n \to e^{\lambda(e^u - 1)}$. □

5.4 Central Limit Theorem

We state first normal approximation to binomial.

Theorem 5.4.1 *If X_n is Binomial $Bin(n,p)$ and $0 < p < 1$ is constant, then $\frac{X_n - np}{\sqrt{npq}} \to Z$ in distribution to $N(0,1)$ random variable Z.*

Proof. For $p = 1/2$ only. The moment generating function of $\frac{X_n - np}{\sqrt{npq}} = \frac{2X_n - n}{\sqrt{n}}$ is $M_n(u) = e^{-\sqrt{n}u}(\frac{1}{2} + \frac{1}{2}e^{-2u/\sqrt{n}})^n = (\frac{e^{-u/\sqrt{n}} + e^{u/\sqrt{n}}}{2})^n = \cosh^n(u/\sqrt{n}) \to e^{u^2/2}$. (Can you justify the limit?) □

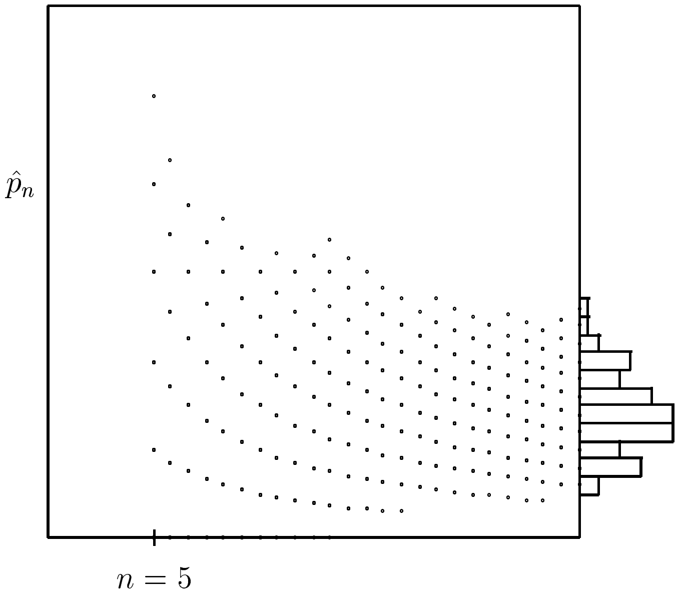

Figure 5.1: Empirical proportions $\hat{p}_n$ as the function of sample size. Output of `LIMTHS.BAS` for $p = .25$ and $5 \leq n \leq 30$

Limit theorems are illustrated by the program `LIMTHS.BAS`. This program graphs empirical proportions $\hat{p}_t$ as the (random) function of $t \leq n$, makes a histogram, and compares it with the Normal and Poisson histograms. By trying various lengths of path n, one can see the almost sure Law of Large Numbers, and for moderate n see the normal approximation.

Problem 5.1 *The graph of $\hat{p}_n$ as the function of n as given by* `LIMTHS.BAS` *suggests a pair of "curves" between which the averages are "squeezed". What are the equations of these curves?*

Exercise 5.2 *Suppose a poll of size n is to be taken, and the actual proportion of the voters supporting an issue in question is $p = \frac{1}{2}$. Determine the size n such that the observed proportion $\hat{p} = \frac{1}{n}X$ satisfies $\Pr(\hat{p} > .8) \leq .01$.*

Exercise 5.3 *Plot the histogram for a 100 independent Binomial $Bin(n = 100, p = .5)$ random variables.*

Theorem 5.4.2 *If X_j are i.i.d. with $EX = \mu, Var(X) = \sigma^2$ then* $\frac{\sum_{j=1}^n X_j - n\mu}{\sigma\sqrt{n}} \to Z$

Proof. For $\mu = 0, \sigma = 1$ only.

The moment generating function is $M_n(u) = (M_1(u/\sqrt{n}))^n = (1 + \frac{u}{\sqrt{n}}M'(0) + \frac{u^2}{2n}M''(0) + O(\frac{1}{n^2}))^n \to e^{u^2/2}$ □

A lesser known aspect of the central limit theorem is that one can actually simulate paths of certain continuous time processes by taking $X_n(t) = \frac{1}{\sqrt{n}}\sum_{k\leq nt}\xi_k$, where ξ_k are independent mean-zero r. v.

The following program uses this to simulate random curves. The program requires a graphics card on the PC.

```
PROGRAM firewalk.bas
'

'This program simulates random walk paths (with uniform incerments)
'reflected at the boundaries of a region
'
'declarations of subs
DECLARE SUB CenterPrint (Text$)

' minimal error handling - graphics card is required
ON ERROR GOTO ErrTrap

CLS
'request good graphics (some cards support SCREEN 7, etc)
SCREEN 9

LOCATE 1, 1 'title
CenterPrint "Path of reflected random walk"
LOCATE 9, 1  'timer location
PRINT "Timer"

scale = 10 '
WINDOW (0, 0)-(scale, scale): VIEW (150, 100)-(600, 300)
LINE (0, 0)-(scale, scale), 10, B': LINE (scale, scale)-(2 * scale, 0), 11, B
FOR j = -4 TO 4
LINE (0, scale / 2 + j)-(scale / 50, scale / 2 + j), 2
NEXT j
X = scale / 2
Y = scale / 2
dead = 0
T = 0
col = 14 * RND(1)
speed = scale / 100
WHILE INKEY$ = "" 'infinite loop until a key is pressed
T = T + 1
X0 = X
Y0 = Y
X = X + (RND(1) - 1 / 2) * speed
Y = Y + (RND(1) - 1 / 2) * speed
IF X < 0 THEN X = -X: col = 14 * RND(1)
IF Y < 0 THEN Y = -Y: col = 14 * RND(1)
IF X > scale THEN X = 2 * scale - X: col = 14 * RND(1)
IF Y > scale THEN Y = 2 * scale - Y: col = 14 * RND(1)
IF X > 2 * scale THEN X = 4 * scale - X: col = 14 * RND(1)
LINE (X0, Y0)-(X, Y), col
LOCATE 10, 1
PRINT T
WEND

END

ErrTrap: 'if there are errors then quit
CLS
PRINT "This error requires graphics card VGA"
PRINT "Error running program"
PRINT "Press any key ...'"
WHILE INKEY$ = ""
WEND
END

SUB CenterPrint (Text$)
' Print text centered in 80 column screen
offset = 41 - LEN(Text$) \ 2
IF offset < 1 THEN offset = 1
LOCATE , offset
PRINT Text$
'

END SUB
```

5.5 Limit theorems and simulations

One role of limit theorems is to justify the simulations and provide error estimates. The simulation presented on page 28 uses the Central Limit Theorem to print out the so called 95% confidence interval.

Central limit theorem is also a basis for a fast simulation of the normal distribution, see Section 6.3.3.

5.6 Large deviation bounds

We begin with the bound for binomial distribution.

Recall that the relative entropy is $H(q|p) = -q\ln q/p - (1-q)\ln\frac{1-q}{1-p}$.

Theorem 5.6.1 *Suppose X_n is Binomial $Bin(n,p)$. Then for $p' \geq p$*

$$\Pr(\frac{1}{n}X_n \geq p') \leq \exp nH(p'|p) \tag{5.1}$$

Proof. Use Chebyshev's inequality (2.22) and the moment generating function. $\Pr(X_n \geq np') = \Pr(uX_n \geq unp') \leq e^{-np'u}E^{uX_n} = \left(e^{-p'u}(1-p+pe^u)\right)^n = \left((1-p)e^{-p'u} + pe^{(1-p')u}\right)^n$.

Now the calculus question: for what $u \geq 0$ the function $f(u) = (1-p)e^{-p'u} + pe^{(1-p')u}$ attains a minimum? The answer is $u = 0$ if $p' \leq p$, or $u = \ln(\frac{1-p}{p}\frac{p'}{1-p'})$. Therefore the minimal value is $f(u_{min}) = \frac{p}{p'}e^{(1-p')u} = \exp(-p' \ln p'/p - (1-p')\ln(1-p')/(1-p))$ □

Corollary 5.6.2 *If $p = 1/2$ then*

$$\Pr(|\frac{1}{n}X_n - \frac{1}{2}| > t) \leq e^{-nt^2/2} \tag{5.2}$$

Proof. This follows from the inequality $(1+x)\ln(1+x) + (1-x)\ln(1-x) \geq x^2$. □

5.7 Conditional limit theorems

Suppose X_j are i.i.d. Conditional limit theorems say what is the conditional distribution of X_1, given the value of the empirical average $\frac{1}{n}\sum_{j=1}^n h(X_j)$. Such probabilities are difficult to simulate when $\frac{1}{n}\sum_{j=1}^n h(X_j)$ differs significantly from $Eh(X)$.

5.8 Questions

Exercise 5.4 *A 400-point multiple choice test has four possible responses, one of which is correct.*

- *What proportion of students that just guess the answer gets the score of more than 100? Of more than 110? Of more than 130?*
- *What proportion of students that know how to answer corectly 20% of question gets the score of more than 100? of more than 130? Of more than 180?*
- *What proportion of problems you know how to answer corectly in order to have a fair shot at A, which requires a score of at least 361? (For the purpose of this exercise a fair shot is 75% chance.)*

Part II

Stochastic processes

Chapter 6

Simulations

This chapter collects information about how to simulate special distributions.

Considerations of machine efficiency, in particular convenience of doing fixed precision arithmetics, make the uniform $U(0,1)$ the fundamental building block of simulations. We do not consider in much detail how such sequences are produced – efficient methods are hardware-dependent. We also assume these are i. i. d., even though the typical random number generator returns the same pseudo-random sequence for each value of the seed, and often the sequence is periodic and correlated. This is again a question of speed and hardware only.

6.1 Generating random numbers

Suppose x_0 is an arbitrary number between 0 and 1 with 5 decimal places or more. Let $x_1 = \{147x_0\}$, and $x_{n+1} = \{147x_n\}$, where $\{a\} = a - [a]$ denotes the fractional part. Here are the questions: Is x_n a random sequence? Does it have "enough" propertries of a random sequence to be used for simulations?

6.1.1 Random digits

The classical Peano curve actually maps the unit interval onto the unit square preserving the Lebesgue measure. Thus two independent $U(0,1)$ are as "random" as one!

Experiments with discrete outcomes aren't necessarily less random than continuous models. Expansion into binary fractions connects infinite tosses of a coin with a single uniform r. v.

Example 6.1 *Let U be uniform $U(0,2\pi)$. Random variables $X_k = sign(\sin(2^k U))$ are i. i. d. symmetric independent.*

Theorem 6.1.1 *If ξ_j are independent identically distributed discrete random variables with values $\{0,1\}$ and $\Pr(\xi = 1) = 1/2$ then $\sum_{k=1}^{\infty} \frac{1}{2^k}\xi_k$ in uniform $U(0,1)$.*

Proof. We show by induction that if U is independent of $\{\xi_j\}$ uniform $U(0,1)$ r. v., then $\frac{1}{2^n}U + \sum_{k=1}^{n} \frac{1}{2^k}\xi_k$ is uniform for all $n \geq 0$. For induction step, notice that in distribution $\frac{1}{2^n}U + \sum_{k=1}^{n} \frac{1}{2^k}\xi_k \cong \frac{1}{2}\xi_1 + \frac{1}{2}U$. This reduces the proof to $n = 1$ case.

The rest of the proof is essentially the solution of Problem 2.31. Clearly $\Pr(\frac{1}{2}\xi_1+\frac{1}{2}U < x) = \Pr(\xi_1 = 0)\Pr(U/2 < x) + \Pr(\xi_1 = 1)\Pr(\frac{1}{2} + U/2 < x)$ Expression $\Pr(U/2 < x)$ is $0, 2x$, or 1. Expression $\Pr(\frac{1}{2} + U/2 < x)$ is $0, 2x - 1$, or 1. Their average (check carefully ranges of x!) is $\begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } 0 \leq x \leq 1 \\ 1 & \text{if } x > 1 \end{cases}$

□

Coefficient 2 does not play any special role. The same fact holds true in any number system - if $N > 1$ is fixed, and $U = \sum X_j N^{-j}$ is expanded in pase N, then X_j are independent uniform $\{0, 1, \ldots, N-1\}$-valued discrete random variables.

From the mathematical point of view all of the simulations can be based on a single $U(0,1)$ random variable. In particular, to generate independently uniform integer numbers[1] in the prescribed range $0 \ldots N$ we need to pick any $u \in (0,1)$, and define $X_j = N^j u \bmod N$.

Clearly X_j is the integer part of NU_j, where U_j solve the recurrence

$$U_{j+1} = \{U_j N\} \tag{6.1}$$

and $\{\}$ denotes the fractional part. A computer realization of this construction would use $U_j = n_j/M$ with some $M > N$, Thus $n_{j+1} = Nn_j \bmod M$.

Many programs provide access to uniform numbers. These however might be platform-dependent, and are often of "low quality". Often a person performing simulation may want to use the code they have explicit access to. What is "random enough" for one application may not be random enough for another.

6.1.2 Overview of random number generators

There is good evidence, both theoretical (see (6.1)) and empirical, that the simple multiplicative congruential algorithm

$$n_{j+1} = an_j \pmod N \tag{6.2}$$

can be as good as the more general linear congruential generator. Park & Miller propose a minimal standard generator based on the choices $a = 7^5, N = 2^{31} - 1$. The computer implementation of this method is not obvious due to overflows when multippying large integers in finite computer arithmetics, see Schrage's algorithm in [23].

The linear congruential method has the advantage of being very fast. It has the disadvantage that it is not free of sequential correlations between successive outputs. This shows up clearly in the fact that the consecutive k-points lie in at most $N^{1/k}$ subspaces of dimension $k - 1$.

Many system-supplied random number generators are *linear congruential generators*, which generate a sequence of integers $n_1, n_2, \ldots$ each between 0 and $N-1$ by the recurrence relation

$$n_{j+1} = an_j + b \bmod N \tag{6.3}$$

[1] This is all we can hope for in the finite computer arithmetics. Whenever $U_j \in (0,1)$ is selected and finite approximation is chosen, $N_j = NU_j$ is an integer, where $N = 2^b$ is the characteristic of the operating system.

The value of N is the largest integer representable in the machine. This is 2^16 on 16-bit machines, unless long integers are used (then it is 2^{32}). Value used in C-supplied generator are $a = 1103515245, b = 12345, N = 2^{32}$.

6.2 Simulating discrete r. v.

6.2.1 Generic Method – discrete case

Section 2.1.2 described the generic method for simulating a discrete random variables with finite number of values. Namely, take $X = f(U)$, where f is a suitable piecewise constant functions on the interval $(0, 1)$. Let $\Pr(X = v_k) = p_k$. Then $f(x) = v_k$ for $x \in (\sum_{j=1}^{k} p_j, \sum_{j=1}^{k+1} p_j)$.

This is rather easy to convert into the computer algorithm.

6.2.2 Geometric

A simple method for simulating geometric distribution is to simulate independent binomial trials until the first success.

6.2.3 Binomial

A simple method for simulating Binomial $Bin(n, p)$ random variable is to simulate binomial trials with the prescribed probability of success. For a sample program illustrating this method, see `TOSSCOIN.BAS` page 7.

6.2.4 Poisson

An exact method to simulate Poisson distribution is based on the fact that it occurs the Poisson process, and that sojourn times in the Poisson process are exponential (see Theorem 11.1.1 on page 103). Therefore, to simulate$X$ with $Poiss(\lambda)$ distribution, simulate independent exponential r. v. $T_1, T_2, \ldots,$ with parameter $\lambda = 1$ and put as the value of X the first value of n such that $T_1 + \ldots + T_n > \lambda$.

A reasonable approximation to Poisson $Poiss(\lambda)$ random variable X is obtained by simulating binomial $Bin(n, p)$ random variable X' with $\lambda = np$. Since $X' \leq n$, use n large enough to exceed any realistic values of X. Run program `LIMTHMS.BAS` to compare the histograms – why is Poisson distribution more spread out than the binomial?

6.3 Simulating continuous r. v.

6.3.1 Generic Method – continuous case

Section 2.2.3 described the generic method for simulating a continuous random variable, similar to the method used in the discrete case. Namely, take $X = f(U)$ where f is the inverse[2] of the cumulative distribution functions $F(x) = \Pr(X \leq x)$.

[2] Actually, we need only a right-inverse, i.e a function such that $F(f(u)) = u$.

Problem 6.1 *Given a cumulative distribution function $G(x)$ with inverse $H()$, and density $g(x) = G'(x)$, let U_1, U_2 be two independent uniform $U(0,1)$ random variables. Show that $X = H(U_1), Y = U_2 g(X)$ are uniformly distributed in the region $\{(x,y) : 0 < y < g(x)\}$.*

6.3.2 Randomization

If the conditional density of Y given $X = x$ is $f(y|x)$, and the density of X is $g(x)$, then the density of Y is $\int (f(y|x)g(x)\,dx$.

As an example, suppose Y has density $cye^{-y} = C\sum_{n=0}^{\infty} y(1-y)^n/n!$. Let $\Pr(X = n) = c/n!$. Then the conditional distribution is $f(y|X = n) = Cy(1-y)^n$, which is the distribution of a median from the sample of size $2n$.

6.3.3 Simulating normal distribution

By the central limit theorem (Theorem 5.4.2), if $U_1, \dots, U_n$ are i. i. d. uniform $U(0,1)$ then $\sqrt{\frac{12}{n}}\sum_{k=1}^{n}(U_k - \frac{1}{2})$ is asymptotically normal $N(0,1)$. In particular a computer-efficient approximation to normal distribution is given by

$$\sum_{k=1}^{12} U_k - 6.$$

The exact simulation of normal distribution uses the fact that an independent pair X_1, X_2 of normal $N(0,1)$ random variables can be written as

$$X_1 = R\cos\Theta \tag{6.4}$$

$$X_2 = R\sin\Theta \tag{6.5}$$

where Θ is uniform $U(0, 2\pi)$, $R = \sqrt{X^2 + Y^2}$ is exponential (see Problem 4.3) with parameter $\lambda = \frac{1}{2}$, and random variables Θ, R are independent. Clearly $R = \sqrt{-2\ln U}$, see Example 2.4.

We simulate two independent normal $N(0,1)$ r. v. from two independent uniform r. v. U_1, U_2 by taking $\Theta = 2\pi U_1, R = \sqrt{-2\ln U_2}$ and using formulas (6.4)-(6.5).

6.4 Rejection sampling

The idea of rejection method is very simple. In order to simulate a random variable with the density $f(x)$, select a point X, Y at random uniformly from the region $\{(x,y) : y < g(x)\}$. Then $\Pr(X < x) = \int_{-\infty}^{x} f(t)\,dt$ is the cumulative distribution function, which we do not have to know analytically.

The name comes from the technique suggested by Problem 6.1. Instead of selecting points under the graph of $f(x)$, we pick another function $g(x) > f(x)$, which has known antiderivative with explicitly available inverse. We pick points under the graph of $g(x)$, and "reject" those that didn't make it below the graph of $f(x)$.

Often used density $g(x) = c/(1 + x^2)$ leads to $X = H(U_1)$, where $H(u) = \tan(\pi u/c)$.

Rejection sampling can be used to simulate continuous or discrete distributions. The idea behind using it in discrete case is to convert discrete distribution to a function of continous random variable. For example, to use rejection sampling for Poisson distribution, simulate the density $f(x) = e^{-\lambda}\lambda^{[x]}/[x]!$ and take the integer part $[X]$ of the resulting random variable.

6.5 Simulating discrete experiments

6.5.1 Random subsets

To simulate uniformly selected random subsets of $\{1, \ldots, n\}$, define sets by a sequences $S(j) \in \{0, 1\}$ with the interpretation $j \in S$ if $S(j) = 1$. Now select independently 0 or 1 with probablity $\frac{1}{2}$ for each of the values of $S(j), j = 1, \ldots, n$.

```
SUB RandomSubset( S())
n = UBOUND(S) ' read out the size of array
FOR j = 1 TO n
'select entries at random
S(j) = INT(RND(1) + 1)
NEXT j
```

For subsets of low dimension, a sequence of $0, 1$ can be identified with binary numbers. Set operations are then easily converted to binary operations on integers/long integers.

6.5.2 Random Permutations

Suppose we want to re-arrange elements $a(1), \ldots, a(n)$ into a random order. A quick method to accomplish this goal is to pick one element at a time, and set it aside. This can be easily implemented within the same sequence.

```
SUB Rearrange( A())
n = UBOUND(A) ' read out the size of array
ns = n 'initial size of randomization
FOR j = 1 TO n
'take a card at random and put it away
k = INT(RND(1) * ns + 1)
SWAP A(k), A(ns)
ns = ns - 1 'select from remaining a(j)
NEXT j
```

Problem 6.2 *Let $a_1, \ldots a_n$ be numbers such that $\sum_j a_j = 0, \sum_j a_j^2 = 1$ Let X denote the sum of the first half of those numbers, after a random rearrangement.*

- *Find $E(X)$, $Var(X)$.*
- *Under suitable conditions, as $n \to \infty$, the distribution of X is asymptotically normal. Verify by simulations.*
- *Let Y be the sum of the first $\frac{1}{4}$ of a_j after random rearrangement. Find $E(X|Y)$ and $E(Y|X)$.*

6.6 Integration by simulations

Program listing on page 28 explains how to perform double integration by simulation. Here we concentrate on refining the procedure for single integrals by reducing the variance.

To evaluate $J = \int_a^b f(x)\,dx$ we may use the approximation $\frac{1}{N}\sum_{j=1}^N f(X_j)/g(X_j)$, where X_j are i.i.d. with the density $g(x)$ such that $\int_a^b g(x)\,dx = 1$.

The error, as measured by the variance[3], is

$$Var\left(\frac{1}{N}\sum_{j=1}^N f(X_j)/g(X_j)\right) = \frac{1}{\sqrt{N}}\int_a^b \left(\frac{f(x)}{g(x)} - J\right)^2 dx.$$

The variance is the smallest if $g(x) = C|f(x)|$, therefore a good approximation g to function f will reduce the variance[4]. This procedure of reducing variance is called *importance sampling.*

For smooth functions, a better approximation is obtained by selecting points more uniformly than the pure random choice. The so called *Sobol sequences* are based on sophisticated mathematics. Cookbook prescriptions can be found eg in *Numerical recipes*.

Note: The reliability of the Monte Carlo Method, and the associated error bounds depends on the quality of the random number generator. It is rather unwise to use an unknown random number generator in questions that require large number of randomizations.

Example 6.2 *The following problem is a challenge to any numerical integration method due to rapid oscillations,* $\int_0^1 2\sin^2(1000x)\,dx = 1 - \frac{1}{2000}\sin 2000 \approx 1.0004367$

6.6.1 Stratified sampling

The idea of stratified sampling is to select different number of points from different subregions. As a simple example, suppose we want to integrate a smooth function over the interval $[0, 1]$ using n points. Instead of following with the standard Monte Carlo prescription, we can divide $[0, 1]$ into k non-overlapping segments and choose n_j points from the j-th subinterval I_j. An extreme case is to take $k = n$ and $n_j = 1$ – this becomes a variant of the trapezoidal method. The optimal choice of n_j is to select them proportional to the local standard deviation of the usual Monte Carlo estimate of $\int_{I_j} f(x)\,dx$. Indeed, denoting by F_j the estimator of the integral over I_j, the variance of the answer is $Var(\sum_{j=1}^k F_j) = \sum_j Var(F_j) = \sum_j \sigma_j^2/n_j$. The minimum under the constraint $\sum_j n_j = n$ is $n_j \approx \sigma_j$.

A simple variant of recursive stratified sampling is to generate points and subdivisions based on estimated values of the variances.

6.7 Monte Carlo estimation of small probabilities

Unlikely events happen too rarely to have any reasonable hope of simulating them directly. Under such circumstances a special method of *selective sampling*[5] was developed.

[3] Why variance?

[4] The smallest possible variance is 0! Why doesn't this happen in the actual application??

[5] See J. S. Sadowski, IEEE Trans. Inf. Th. IT-39 (1993) pp. 119–128, and the references therein.

Example

Suppose we want to find $\Pr(X_1 + \ldots + X_n > \alpha n)$ for large n and a given density $f(x)$ of independent r. v. X. Consider instead independent random variables Y_j with the "tilted density" $Ce^{\lambda x} f(x)$, where C is the normalizer, and λ is such that $EY = \alpha$. By the law of large numbers (Theorem 5.2.1), the event $Y_1 + \ldots + Y_n > \alpha n$ has large probability, and $\Pr(X_1 + \ldots + X_n > \alpha n) = \int_{y_1+\ldots+y_n>\alpha n} e^{-\lambda y_1} \ldots e^{-\lambda y_n} f(y_1) \ldots f(y_n)\, dy_1 \ldots dy_n$. This leads to the following procedure.

- Simulate N independent realizations of the sequence $Y_1, \ldots, Y_n$.
- Discard those that do not satisfy the constraint $y_1 + \ldots + y_n > \alpha n$.
- Average the expression $e^{-\lambda Y_1} \ldots e^{-\lambda Y_n}$ over the remaining realizations to get the desired estimate.

Example 6.3 *Suppose X_j are $\{0, 1\}$-valued so that $X_1 + \ldots X_n$ is binomial $Bin(n, p)$. What is the distribution of Y_j? Simplify the expression $e^{-\lambda Y_1} \ldots e^{-\lambda Y_n}$.*

Exercise 6.3 *Write the program computing by simulation the probability that in a $n = 10$ tosses of a fair coin, at least 8 heads occur. Once you have a program that does for $n = 10$ a comparable job to the "naive" program below, try Exercise 1.13 on page 14.*

Here is a naive program, that does the job for $n = 10$, but not for $n = 100$. It should be used to test the more complex "tilted density" simulator.

```
PROGRAM heads.bas
'

'Simulating N fair coins
' declarations
DECLARE FUNCTION NumHeads% (p!, n%)
DEFINT I-N  ' declare integer variables

'prepare screen
CLS

PRINT "Simulating toss of n fair coins"

'get users input
n = 10 'number of trials
INPUT "Number of coins n=", n
pr = .5 ' fairness of a coin
frac = .8 ' percentage of heads seeked
' get the frac from the user
PRINT " How often we get more than f heads? (where 0<f<"; n; ")"
INPUT "f=", frac
IF frac >= 1 THEN frac = frac / n 'rescale if too large
'tell user what is going on
PRINT "Hit any key to see the final answer"
LOCATE 20, 10
PRINT "With some patience you may see digits stabilize"

DO 'main loop
```

```
        T = T + 1
        IF NumHeads(pr, n) > frac * n THEN s = s + 1
        IF INKEY$ > "" THEN EXIT DO
        LOCATE 10, 10
        PRINT "Trial #"; T
        PRINT "Current estimate"; USING "##.#####"; s / T
LOOP
'print the answer
PRINT
PRINT "Prob of more then "; INT(frac * n); " heads in "; n; " trials is about "; s / T
END

DEFINT A-H, O-Z
FUNCTION NumHeads (p!, n)
'simulate a run of n coins
 h = 0
FOR k = 1 TO n
        IF RND(1) < p! THEN h = h + 1
NEXT k
NumHeads = h
'

END FUNCTION
```

Chapter 7

Introduction to stochastic processes

> **stochastic**, *a.* conjectural; able to conjecture
> *Webster's New Universal Unabridged Dictionary*

Stochastic processes model evolution of systems that either exhibit inherent randomness, or operate in an unpredictable environment. This unpredictability may have more than one form, see Section 7.3.

Probability provides models for analyzing random or unpredictable outcomes. The main new ingredient in stochastic processes is the explicit role of time. A stochastic process is described by its position $X(t)$ at time $t \in [0,1]$, $t \in [0,\infty)$, or $t \in \{0,1,\dots\}$.

From the conceptual point of view, stochastic processes that use discrete moments of time $t \in \{0,1,\dots\}$ are the simplest. Since the discrete moments of time can represent arbitrarily small time increments, discrete time models are rich enough to model real-world phenomena. Continuous time versions are convenient mathematical idealizations.

7.1 Difference Equations

Mathematical models of time evolution of deterministic systems often involve differential equations.

Difference equations are discrete analogues of the differential equations. Difference equations occur in applied problems, and also when solving differential equations by series expansions, or by Euler's method. The concepts of numerical versus analytical solution, initial values, boundary values, linearity, and superposition of solutions occur in the discrete setup in complete analogy with the theory of differential equations.

A difference equation determines unknown sequence (y_n) through a recurrence relation that specifies the pattern. We will consider only special cases of classes of equation

$$y_{n+1} = f(n, y_n, y_{n-1}, \dots, y_{n-k+1}).$$

Here coefficient k is the *order* of the equation. For instance, $y_{n+1} = f(n, y_n)$ is an equation of order 1, $y_{n+1} = f(n, y_n, y_{n-1})$ is an equation of order 2, etc.

7.1.1 Examples

Example 7.1 *Suppose a sequence y_n is to satisfy*

$$y_{n+1} = \cos(y_n), \tag{7.1}$$

where the cosine function is in radians.

It is easy to write a short program that computes the values of y_n. But to compute the actual sequence we need to specify the *initial value* y_0. Table 7.1 gives sample outputs of such a program for several choices of y_0.

y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
-1	.5403023	.8575532	.6542898	.7934803	.7013688	.7639596	.7221025	.7504177
-.5	.8775826	.6390125	.8026851	.694778	.7681958	.7191654	.7523558	.7300811
0	1	.5403023	.8575532	.6542898	.7934803	.7013688	.7639596	.7221025
.5	.8775826	.6390125	.8026851	.694778	.7681958	.7191654	.7523558	.7300811
1	.5403023	.8575532	.6542898	.7934803	.7013688	.7639596	.7221025	.7504177
1.5	.0707372	.9974992	.542405	.8564697	.6551088	.7929816	.7017242	.7637303
2	-.4161468	.9146533	.6100653	.8196106	.6825058	.7759946	.7137247	.7559287

Table 7.1: Sequences y_n satisfying equation (7.1) with different initial values y_0.

Similar numerical procedures show up in differential equations, where they are used to approximate continuous solutions by discretizing time. The procedure is called *Euler method*, or *tangent line method.*

Example

Some difference equations are simpler than others. Suppose a sequence y_n is to satisfy

$$y_{n+1} = y_n + d, \tag{7.2}$$

where d is a given number. It is easy to write a short program that computes values of y_n. To compute them we again need to specify the initial value y_0.

On the other hand, we may notice that equation (7.2) defines the arithmetic progression. Instead of a table like Table 7.1, we can write down the *solution* for all possible values of y_0 and for all n. Namely,

$$y_n = y_0 + dn. \tag{7.3}$$

In general, an *analytical solution* of the difference equation is the formula that expresses y_n as the function of n. This should be contrasted with the *numerical solution* which is an algorithm, or computer program, that computes values of y_n. The *general solution* is the function of both y_0 and n, as contrasted with a *particular solution* that works for a prescribed initial value[1] y_0 only.

[1] Such as $y_0 = 0$.

Example 7.2 *Here is another well known difference equation with obvious solution. Suppose*

$$y_{n+1} = ry_n, \tag{7.4}$$

where r *is a given number. Equation (7.4) defines a geometric progression and its solution is.*

$$y_n = y_0 r^n \tag{7.5}$$

Examples 7.1.1 and 7.2 are deceptively simple. Not every difference equation has a simple or easy to guess answer.

Example

The following equation defines the Fibonacci sequence

$$y_{n+1} = y_n + y_{n-1}. \tag{7.6}$$

In a typical application, y_n denotes the number of rabbits at the end of the nth month. In particular, if we buy one newborn rabbit at the beginning of the first month then the first terms of the sequence are easy[2] to write down:
`1, 1, 2, 3, 5, 8, 13, 21, ...`
Without much difficulty this can be converted to a computer program and used to answer questions like *When will the population of rabbits exceed 1 million?* The general expression (solution) of the equation corresponding to this situation is given by the following formula.

$$y_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

This is the outcome of the standard computation!

7.2 Linear difference equations

Many interesting difference equations, including (7.2), (7.4), (7.6) fall into the category of linear difference equations with constant coefficients. The first order linear difference equations with constant coefficients has the form

$$y_{n+1} + ay_n = g(n).$$

The second order linear difference equations with constant coefficients has the form

$$y_{n+2} + ay_{n+1} + by_n = g(n).$$

A method to solve difference equation of order 2 consists of the following steps:

[2] Following good mathematical practise, we simplify the real world and assume that a single mature rabbit will produce one offspring every month.

- First we solve the "homogeneous equation" $y_{n+2} + ay_{n+1} + by_n = 0$. This is accomplished through the substitution $y_n = r^n$,which leads to the characteristic equation $r^2 + ar + b = 0$ for r. Once two roots are found, the general solution is $y_n = C_1 r_1^n + C_2 r_2^n$. The notable exception when the roots are equal is $y_n = C_1 r_0^n + C_2 n r_0^n$.
- Secondly, we find any solution of the non-homogeneous equation, disregarding initial conditions. This can be accomplished by the method of varying parameters: try $y_n = C_1(n) r_1^n + C_2(n) r_2^n$. Or by guessing. In the often encountered polynomial case $g(x) = n^\ell$ the trial solution $y_n = n^s(u_0 + u_1 n + \ldots + u_\ell n^\ell)$ will work ($s = 0$ except when $y_n = n$ solves the homogeneous equation, in which case $s = 1$).
- In the final step the general solution to homogeneous equation is combined with the particular solution to non-homogeneous equation, and the initial value problem is solved.

Example

Here is an example of the applied problem that leads to a natural, but not obviously solvable difference equation.

Suppose you borrow y_0 dollars on fixed monthly interest rate r. If you do not make any payments on your loan, then your balance will "balloon" exponentially. Formula $y_n = (1 + r)^n y_0$ expresses monthly balance after n months when no loan payments are made. Some people prefer to pay a fixed amount of p dollars at the end of each month. If p is large enough, then the balance may even shrink down! This situations is easily described by a difference equation. To write it down, compute the next month balance, if the previous month balance is known:

$$y_{n+1} = (1 + r)y_n - p. \tag{7.7}$$

Equation (7.7) and its general solution are of interest to bank officers and to their customers. From the formula for y_n, they can determine monthly payments that will pay a loan off in the prescribed amount of time.

In differential equations, we often solve a similar problem in continuous time, with continuous compounding and continuous payments schedule. This is a mathematical simplification of the actual banking situation, but the answers are reasonably close, and they are easier to get.

Here is an example of the mathematical problem that leads to a natural, but nontrivial[3] difference equation.

Example 7.3 *Suppose we want to find the formula for the sum of all consecutive integers from* 1 *to* n. *Let the answer by* y_n. *Then the recurrence formula*

$$y_{n+1} = y_n + n + 1 \tag{7.8}$$

holds.

[3] Most likely, you know the solution to this one, but the method is useful for other problems as well.

Now (7.8) can be written as $y_{n+1} - y_n = n + 1$ and actually this is why we use the name *difference equations* rather than recurrence relations. Notice that we do know the solution of its continuous analogue. Equation $y' = t + 1$ for unknown function $y = y(t)$ resembles (7.8). Its solution[4] is $y(t) = \frac{1}{2}t^2 + t$. So to find the answer to (7.8) we may try substituting $u_n = \frac{1}{2}n^2 + n$ into the equation. Unfortunately, simple arithmetics shows that we don't get the right answer, as

$$u_{n+1} = u_n + n + \frac{3}{2}. \tag{7.9}$$

But then, we are not that far off the target. Let $v_n = u_n - y_n$. Subtracting equation (7.8) from equation (7.9) we get the differential equation for (v_n)

$$v_{n+1} = v_n + \frac{1}{2}.$$

This is the special case of the arithmetic progression equation (7.2). From Example 7.1.1 equation (7.3) we know that $v_n = n/2$. Therefore $y_n = \frac{1}{2}n^2 + n - n/2 = \frac{n(n+1)}{2}$

The method we used – subtracting the equations – works for the so called *linear difference equations* (and also linear differential) equations. It is closely related to the *Principle of Superposition* for linear differential equations.

7.2.1 Problems

1. Check if the given sequence solves the difference equation.

   (a) Equation: $y_{n+1} = -y_n$. Sequence $y_n = \cos n\pi$.

   (b) Equation: $y_{n+1} = y_n + 2y_{n-1}$. Sequence $y_n = 2^n$.

   (c) Equation: $y_{n+1} = y_n + 2y_{n-1}$. Sequence $y_n = -\frac{1}{2}n^3 + 3n^2 - \frac{5}{2}n + 1$.

   (d) Equation: $y_{n+1} = 2y_n - y_{n-1} + 2$. Sequence $y_n = n^2$.

2. Write down the difference equation that you need to solve each of the following problems (**do not solve the equation, nor the problem**).

   (a) A loan of \$1000 has interest rate that varies with time as follows: first month interest is 0%, second month interest is $\frac{1}{12}$%, third month interest is $\frac{2}{12}$%, etc with monthly interest increasing by $\frac{1}{12}$% every month. Determine the fixed monthly payment p that will pay this loan within one year.

   (b) A loan of \$1000 has constant monthly interest rate of $\frac{1}{12}$%. I arranged my monthly payments in the following fashion: *at the end of the first month I will pay nothing, at the end of the second month I will pay \$10, at the end of the third month I will pay \$20, etc, with monthly payments increasing by \$10 every month.* Determine when I will pay off this loan and how much money I will pay in total.

3. Find the general solution of the following difference equations.

   (a) $y_{n+1} = \frac{1}{n}y_n$

[4] Clearly, we request $y_0 = 0$.

(b) $y_{n+1} = \frac{n}{n+1} y_n$

(c) $n(n+1)y_{n+1} = y_n$

4. Solve the given initial value problem for the difference equation.

   (a) $y_{n+1} = \frac{1}{n} y_n$; y_0=1

   (b) $y_{n+1} = \frac{n}{n+1} y_{n-1}$; $y_0 = 1, y_1 = 0$

   (c) $n(n+1)y_{n+1} = 2y_n$;$y_0 = 1, y_1 = 0$

5. Find the formula for

   (a) $y_n = 1^2 + 2^2 + \ldots + n^2$

   (b) $y_n = 1^3 + 2^3 + \ldots + n^3$

   (c) $y_n = \frac{1}{2} + \frac{1}{6} + \ldots + \frac{1}{n(n+1)}$

   (d) $y_n = 1 + r + r^2 + \ldots + r^{n-1}$

   (e) $y_n = r + 2r^2 + 3r^3 + \ldots + nr^n$

6. Each solution of equation (7.1) has the limit $\lim_{n\to\infty} y_n$. Show that the first digits of this limit are .739085133.

7.3 Recursive equations, chaos, randomness

Before jumping to models that use explicit randomness in their evolution, it is quite illuminating to analyze first some mathematically simple and well defined "deterministic evolutionary processes". The following set of examples describes deterministic evolution in discrete time of a system described by a single numerical parameter $x \in (0,1)$. All of the examples fall into the category of (non-linear) difference equations: given initial value $x_0 \in (0,1)$ and a simple evolution equation of the form $x_{n+1} = g(x_n)$, we are supposed to make inferences about the behavior of the solutions.

There is no randomness in the evolution itself. But since we are allowed to choose any initial condition, and no initial condition can be measured exactly, we may as well consider the initial value to be random.

Example 7.4 *Equation*

$$x_{n+1} = \cos(x_n)$$

is analyzed numerically in Example 7.1. A useful technique to analyze such equations is to graph function $y = g(x)$ together with line $y = x$, and represent the sequence x_k by points (x_k, x_k) on the line. The actual proof that $x_n \to x_*$ may perhaps be not that obvious, but the geometric argument seems to be quite convincing.

Example 7.5 *Let $\{x\}$ denote the fractional part of x. Equation*

$$x_{n+1} = \{2x_n\}$$

uses discontinuous function $g(x)$. The previous graphical technique is more difficult to apply here, but the reason for this difficulty might be not apparent. Special initial points, *like $x_0 = 1/2, x_0 = 1/4, x_0 = 1/8, x_0 = 3/4, \dots$ are relatively easy to analyze. But these are exceptional - the majority of the initial points actually doesn't follow this pattern.*

Here is a rather surprising fact. Suppose x_0 is selected at random. Since x_k is a function of x_0, and x_0 is random, x_k becomes a random variable. It may happen that $x_k < 1/2$. Call this event A_k. The reasoning presented in Section 6.1.1 implies that events $\{A_k\}$ are independent and have the same probability $\Pr(A_k) = 1/2$. Therefore the deterministic evolution equation contains at least as much randomness as the tosses of a coin.

The last example may perhaps give the impression that it is the discontinuity of the evolution equation that is the source of difficulties. This is not at all the case.

Example 7.6 *Equation*

$$x_{n+1} = 4x_n(1 - x_n)$$

is another example of the "chaotic equation" with solutions exhibiting as much irregularity as the tosses of a coin. Attempts at graphing its solutions do no indicate any patterns. Tiny differences in the choice of initial value x_0 significantly change the evolution within short time.

Example 7.6 indicates that "naive" prediction of the future of a system is unreliable even within the realm of deterministic evolution equations.

On the other hand, there are aspects of the evolution that we can analyze reliably. These deal with the average behavior of the evolution.

There are two classes of questions that we can answer, but both deal with statistical nature of the evolution:

- what happens if the same experiment is repeated many times (with slightly different initial conditions)?
- What is the average behavior of the system over long periods of time?

For instance, we can ask and get reliable answers to questions like:

- What is the average $\frac{1}{n}\sum_{j=1}^{n} x_j$?
- What is $\frac{1}{n}\sum_{j=1}^{n} U(x_j)$ for a given function U?
- How often $x_j < 1/2$? (Meaning - what proportion of $j \leq T$ satisfies the condition for large T.)
- How often $x_3 < 1/2$ when x_0 is selected according to density $g(x)$? (Meaning - what proportion of x_0 satisfies the condition for large number of initial points x_0.)

Theory of stochastic processes uses descriptive rather than casual models. Its primary goal is to isolate methods that answer questions that can be answered - about the averages and chances of events. It is setup in the form that makes it more natural to ask the "correct" questions. But in real life, and in simulations, we do have access to aspects of the phenomenon than what the theory does not expose. In analyzing simulations it is important to keep in mind the examples above. Avoid collecting data that deal with instances rather than statistical phenomena. Print out well defined statistics of the simulation only. Do not clutter your simulations with irrelevant details.

7.4 Modeling and simulation

The quick way to get insights into operation of real systems is to model their behavior. Here are examples of enterprises that operate under randomness. Mathematicians devised methods of modelling each of these. But it is interesting also to simulate their behavior. Notice that simulations require assumptions, but so do the analytical methods. Regardless of the method, we have to be careful about what are the questions we can answer.

Example 7.7 *Suppose we want to study how much capital is needed to run a casino. We need to answer the following.*

- *How often do people win?*
- *How likely is the casino to loose money in a day? In a month?*
- *How much capital should be kept on hand to cover the losses?*

There is also a number of questions that we do not want to answer.

Example 7.8 *Suppose we want to study how much capital is needed to insure cars. We need to answer the following.*

- *How often do accidents occur? How expensive are repairs/medical costs?*
- *How likely is that the insurance company looses money in a day? In a month?*
- *How much capital should be kept on hand to cover losses?*

There is also a number of questions that we do not want to answer. For instance we do not want to predict whether I'll file an insurance claim today, driving back home on I-74 without enough sleep since I was preparing this class until 3AM.

Example 7.9 *A construction company has n jobs to be performed in the future. For each of these jobs, experts provide estimate of the cost. Then, after questioning, they complement the estimates by the lower/upper bound for the costs. How much money should be allotted?*

Example 7.10 *A store averages $\lambda(t)$ customers per hour at time t of the day. Each customer brings some (random) profit. However, a customer may just leave the store without shopping, if the lines are too long.*

- *How many cashiers should be available for each shift (time) t?*
- *What are the profits on average?*

7.5 Random walks

A random walk is a process of the form $X_n = \sum_{j=1}^{n} \xi_j$, where ξ_j are i. i. d. Random walks have *independent increments* , and describe the accumulation of independent contributions over time.

Random walks are examples of Markov processes which will be studied in detail in Chapter 8. Their special structure allows to analyze them independently of the general theory.

7.5.1 Stopping times

The stopping times are random variables that describe phenomena which depend on the trajectory of a random walk. The definition captures the intuition that their values are determined by the history of a Markov chain.

Definition 7.5.1 *$T : \Omega \to \mathbb{N} \cup \infty$ is a stopping time, if the event $\{T = n\}$ is independent of $\xi_{n+1}, \xi_{n+2}, \ldots$.*

This definition is specialized to random walks. In a more general Markov case the definition is less transparent but captures the same idea.

The most important example of a stopping time is the first entrance time $T = \inf\{k : X_k \in A\}$. An example that is **not** a stopping time is the last exit from a set.

When $T < \infty$ we define random sums $X_T = \sum_{j \leq T} \xi_j$. The following theorem is an exercise when T is independent of ξ.

Theorem 7.5.1 *If ξ_j are i. i. d., $E\xi = \mu$, and $ET < \infty$ is a stopping time then*

$$EX_T = \mu ET \tag{7.10}$$

Proof. $EX_T = \sum_n EX_n \Pr(T = n) = \sum_n \sum_{k=1}^n E\xi_k \Pr(T \geq k)$. Since ξ_k and $\{T \geq k\} = \{T < k\}'$ are independent, therefore $ES_T = \mu \sum_n \Pr(T \geq n) = \mu ET$ by tail integration formula (2.9). □

Theorem 7.5.2 *If ξ_j are i.i.d., $E\xi = \mu$, $Var(\xi) = \sigma^2 < \infty$, and $ET < \infty$ then*

$$E(S_T - T\mu)^2 = \sigma^2 ET \tag{7.11}$$

(These formulas are of interest in branching processes, and in chromatography.)

Example 7.11 *The number of checks cashed at a bank per day is Poisson random variable N with mean $\lambda = 200$. The amount of each check is a random variable with a mean of \$30 and a standard deviation of \$5. If the bank has \$6860 on hand, is the demand likely to be met?*

Problem 7.1 *Suppose ξ_k, T are independent. Find the variance of X_T in terms of the first two moments of ξ, T.*

7.5.2 Example: chromatography

Chromatography is a technique of separation mixtures into compounds. One of its uses is to produce the DNA bands.

The sample is injected into a column, and the molecules are transported along the length by electric potential, flow of gas, or liquid. The basis for chromatographic separation of a sample of molecules is difference in their physical characteristics. The molecules switch between two phases: mobile, and stationary, and the separation of compounds is caused by the difference in times spend in each of the phases.

Suppose that the molecules of a compound spend random independent amounts of time U_k in mobile phase and random amount of time W_k in the stationary phase. Thus at time t the position of a molecule is given by a random sum $v\sum_{j=1}^{T(t)} U_j$, where $T(t) = \inf\{k : \sum_{j=1}^{k} U_j + W_j > t\}$.

Section 7.5.1 gives formulas for the mean and the variance of the position. Since the number of transitions T is likely to be large for a typical molecule, it isn't surprising that the actual position has (asymptotically) normal distribution. (The actual Central Limit Theorem for random sums is not stated in these notes.)

Exercise 7.2 *Simulate the output of the chromatography column of fixed length separating a pair of substances that have different distributions of mobile and stationary phases* U_k, W_k*. Select a hundred particles of each substance, and measure the degree of separation at the end of the column.*

7.5.3 Ruining a gambler

The following model is a reasonable approximation to some of the examples in Section 7.4.

Suppose a gambler can afford to loose amount $L > 0$, while the casino has capital $C < 0$. Let $\xi_j = \pm 1$ be i. i. d. random variables modelling the outcomes of consecutive games, S_n be the partial sums (representing gains of the gambler), and let $T = \inf\{k : S_k \geq L \text{ or } S_k \leq C\}$ be the total number of games played. Then $\Pr(T > k) = \Pr(C < S_k < L)$ and thus $ET = \sum_k \Pr(C < S_k < L)$.

The special case $\Pr(\xi = \pm 1) = 1/2$ is easily solved, since in this case $ES_T = E\xi ET = 0$. Let $p = \Pr(S_T = C)$ denote the probability of ruining the casino. Since S_T is either C, or L we have $0 = ES_T = pC + (1-p)L$, giving $p = L/(L-C)$. This formula means that a gambler has a fair chance of ruining a casino in a fair game, provided he brings with him enough cash L.

For more general random walks (and less fair games) probability of gambler's ruin can be found explicitly using the one-step-analysis (Section 8.3). It is also interesting to find how long a game like that would last on average. (The expression for ET given above is not explicit.)

7.5.4 Random growth model

The following models various growth phenomena like the spread of a disease, where the infected individual may either die, or infect a number of other individuals. Here we concentrate on bacteria which have simple reproduction mechanism, and all spatial relations are neglected.

Let X_t denote the number of bacteria in t-th generation, with $X_0 = 1$. Assume that a bacteria can die with probability $q > 0$, or divide into two cells with probability $p = 1 - q$, and that all deaths occur independently. Our goal here is to find the average number of bacteria $m(t) = E(X_t)$ in the t-th generation. This can be recovered from Theorem 7.5.1. Instead, we show another method based on conditioning.

The number of bacteria in the next generation is determined by binomial probabilities: $\Pr(X_{t+1} = 2k | X_t = n) = \binom{n}{k} p^k q^{n-k}$. Therefore $E(X_{t+1}|X_t) = 2pX_t$ and the average population size $m(t) = E(X_t)$ satisfies difference equation $m(t+1) = 2pm(t)$. We have $m(t) = (2p)^t$. In particular, the population on average grows exponentially when $p > 1/2$.

It is perhaps surprising that a population of bacteria with $p = 3/4$, which on average grows by 50% per generation, has still a $\frac{1}{3}$ chance of going extinct. One way to interpret this is to say that infections by a "deadly" and rapidly developing desease may still have a large survival rate without any intervention of medicine, or immune system. (The methods to compute such probabilities will be introduced in Section 8.3. The answer above assumes infection by a single cell.)

Problem 7.3 *Find the formula for the variance $Var(X_t)$ of the number of bacteria in t-th generation.*

Problem 7.4 *What is the probability that an infection by 10 identical bacteria with the doubling probability $p = 3/4$ dies out?*

Chapter 8

Markov processes

> **evolution**, *n.* [L. *evolutio (-onis)*, an unrolling or opening...
> *Webster's New Universal Unabridged Dictionary*

Markov processes are perhaps the simplest model of a random evolution without long-term memory.

Markov process is a sequence X_t of random variables indexed by discrete time $t \in \mathbb{Z}_+$, or continuous $t \geq 0$ that satisfies the so called Markov property. The set of all possible values of variables X_t is called the *state space* of the Markov chain. Typical examples of state spaces are $\mathbb{R}$, $\mathbb{N}$, the set of all non-negative pairs of integers, and finite sets.

Markov chains are Markov processes with discrete time. Thus a Markov chain is an infinite sequence $\{X_k\}_{k\in\mathbb{Z}_+}$ of (usually, dependent) random variables with short-term (one-step) memory.

8.1 Markov chains

The formal definition of the Markov property is as follows.

Definition 8.1.1 *A family of discrete r. v. $\{X_k\}_{k\in\mathbb{Z}_+}$ is a Markov chain, if*

$$\Pr(X_{k+1} \in U|X_0, \ldots, X_k) = \Pr(X_{k+1} \in U|X_k)$$

depends only on the present value X_k.

Examples of Markov chains are:

- A sequence of independent r. v.
- A constant random sequence $X_k = \xi$.
- Random walks (sums of independent random variables).

Examples of non-Markov processes are easy to construct, but lack of Markov propertry is not obvious to verify. In general, if X_k is a Markov process, $Y_k = f(X_k)$ may fail to be Markov.

8.1.1 Finite state space

If a Markov chain has a finite state space, we can always assume[1] it consists of integers.

Markov condition

$$\Pr(X_{k+1} = x|X_0, \ldots, X_k) = \Pr(X_{k+1} = x|X_k) \tag{8.1}$$

implies that probability of reaching x in the next step depends only on the present value X_k. The probabilistic behavior of such a chain is completely determined by the initial distribution $p_k = \Pr(X_0 = k)$ and the transition matrices $P_n(i, j) = \Pr(X_n = j|X_{n-1} = i)$, see formula (1.11) on page 15. For mathematical convenience we shall assume that one step transition matrices $P_t = P$ do not depend on time t. Such Markov chains are called *homogeneous*. This assumption isn't realistic, nor always convenient. For instance, the Markov simulation in Section 8.5 uses a Markov chain with transitions that vary with time. But homogeneous Markov chains are still flexible enough to handle some time dependencies efficiently through modifications to the state space.

Example 8.1 *Suppose X_n is a Markov chain with periodic transition probabilities $P_n = P_{n+T}$. Then $Y_n = (X_{n+1}, X_{n+2}, \ldots, X_{n+T})$ is a homogeneous Markov chain.*

Problem 8.1 *Suppose ξ_j are independent $\{0, 1\}$-valued with $\Pr(\xi = 1) = p$. Let $X_n = a\xi_n + b\xi_{n+1}$, where $ab \neq 0$.*

Explain why X_n is a Markov chain.

Write the transition matrix for the Markov chain X_n.

Proposition 8.1.1 *The probabilities $\Pr(X_n = j|X_0 = i)$ are given by the i, j-entries of the matrix P^n*

Proof. This is the consequence of Markov property (8.1) and the total probability formula (1.10). □

Powers of moderately sized matrices are easy to compute on the computer. Section B.9 indicates a mathematical method of computing P^n for small dimensions using the Cayley-Hamilton theorem. Under certain conditions the powers converge.

Exercise 8.2 *Find $\lim_{n\to\infty} P^n$ for the matrix from Problem 8.1.*

Stationary Markov processes

Suppose $\Pr(X_0 = k) = p_k$, where p_k solve stationarity equations

$$\sum_k p_k = 1 \tag{8.2}$$

$$[p_1, \ldots, p_d] = [p_1, \ldots, p_d] \times P \tag{8.3}$$

$$\tag{8.4}$$

[1] Notice that this is a mathematical simplification that might be not worth pursuing if the actual state space has some convenient interpretation.

Then the resulting process is stationary: the distribution of each k-tuple $(X(t_1),\dots,X(t_k))$ is invariant under shifts in time, $(X(t_1+s),\dots,X(t_k+s)) \cong (X(t_1),\dots,X(t_k))$. This is interpreted as "equilibrium", or "steady state". Notice that "steady state" is a statistical concept, and is not easily visible in a simulation of the single trajectory. In order to be able to see it, one has to simulate a large number of independently evolving Markov chains that begin with the same initial distribution and have the same transition matrix.

If X_t is a stationary Markov process and f is a function on its state space, then $Y(t) = f(X_t)$ is also stationary, although not necessarily Markov.

If $X_j(t)$ are independent realizations of the same Markov process and f is a function on their state space then $Y_n(t) = n^{-1/2}\sum_{j=1}^n (f(X_j(t)) - \mu)$ is stationary and approximately normal random sequence.

8.1.2 Markov processes and graphs

The states of a Markov chain may be represented by vertices of a graph, and one step transitions may be described by directed edges with weights. Such representation of a markov chain aids in visualizing a Markov chain.

Classification of states

Graph notions have bearing on properties of the Markov chain. In particular, Markov chain is irreducible, if the corresponding graph is connected. Markov chain is periodic, if there is $N > 1$ (the period) such that all cycles of the graph are multiples of N. If there is no such N then Markov chain is called *aperiodic*.

Finite state Markov chain is regular, if there is deterministic number N such that all states are connected by paths of length at most N.

Problem 8.3 *Show that regular Markov chain is aperiodic and irreducible.*

Trajectory averages

Additive functionals of a Markov process are expressions of the form $\frac{1}{n}\sum_{j=0}^{n-1} f(X_j)$. Under certain conditions, the averages converge and the limit doesn't depend on the initial distribution. Under certain conditions, partial sums are approximately normal.

Problem 8.4 *Let X_k be an irreducible $\{0,1\}$-valued Markov chain with invariant initial distribution.*

- *Show that there is $C > 0$ such $Var(\sum_{t=0}^T X_t) \le CT$.*
- *Use the above to show that the law of large numbers holds.*

Asymptotic probabilities

Let $p_k(n) = \Pr(X_n = k)$, and suppose that the limit $p_k(\infty) = \lim_{n\to\infty} p_k(n)$ exists. Since $p_k(n+1) = \sum_{j=1}^{d} p_j(n)P(j,k)$, therefore the limit probabilities satisfy the *stationarity equations* (8.2)

For regular (finite state space) Markov chains the limit actually exists independently of the initial state. Therefore the stationarity equations can be used to find the limit.

Problem 8.5 *Let* $P = \begin{bmatrix} 1/4 & 3/4 \\ 3/4 & 1/4 \end{bmatrix}$.

- *Find the initial distribution of* X_0 *that results in a stationary process.*
- *Find the limiting distribution* $\lim_{n\to\infty} p_k(n)$.

Problem 8.6 *Let* $P = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

- *Find the initial distribution of* X_0 *that results in a stationary process.*
- *Explain why* $\lim_{n\to\infty} p_k(n)$ *does not exist.*

Problem 8.7 *Let* $P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

- *Find the initial distribution of* X_0 *that results in a stationary process.*
- *Find the limiting distribution* $\lim_{n\to\infty} p_k(n)$.

Example: two-state Markov chain

Suppose X_k is a Markov chain with transition matrix $P = \begin{bmatrix} 1-a & a \\ b & 1-b \end{bmatrix}$. Then

$$P^n = \frac{1}{a+b}\begin{bmatrix} b & a \\ b & a \end{bmatrix} + \frac{(1-a-b)^n}{a+b}\begin{bmatrix} a & -a \\ -b & b \end{bmatrix}.$$

If $0 < a, b < 1$

$$P^n \to \begin{bmatrix} \frac{b}{a+b} & \frac{a}{a+b} \\ \frac{b}{a+b} & \frac{a}{a+b} \end{bmatrix}$$

and the rate of convergence is exponentially fast.

Problem 8.8 *Suppose* X_k *is a Markov chain with transition matrix* $P = \begin{bmatrix} a & 1-a \\ 1-b & b \end{bmatrix}$. *Then* $Y_n = (X_n, X_{n+1})$ *is also a Markov process. Find its transition matrix and the stationary distribution.*

8.2 Simulating Markov chains

Homogeneous and non-homogeneous Markov chains with finite state space are fairly straightforward to simulate. A generic prescription is to simulate X_{n+1} using the conditional distribution determined by X_n. This simulation differs very little from the generic method described in Section 2.1.2

Example: how long a committee should discuss a topic?

This example involves simulation of a Markov chain. For Markov chains many theoretical results are available, but simulation is often the most expedient way to study it.

Exercise 8.9 *Suppose n people on a committee discuss a certain issue. When one person finishes speaking, we assume that it is equally likely that any of the other $n-1$ begins. We also assume that each person speaks for exponentially distributed random time. How long does it take on average for all participants to take part in the discussion?*

8.2.1 Example: soccer

The soccer game[2] is played by two teams, each with 10 players in the field and a goalkeeper. A modern line-up split the players into between the zones of defence, center, and attack. Thus a configuration (3-4-4) means 3 defenders (backs), 4 midfield link men and 4 strikers (forwards). In the Markov model of soccer we will just watch the position of the ball, and we assume it can be only in one of the five positions: left goal, left defence, midfield, right defence, right goal. Wee shall assume that at every unit of time the ball has to move left or right with chances proportional to the number of players lined-up for a given position.

For each possible configurations of teams, we could determine the average score, and thus determine which player distribution is the best, if there is one. If there is no best single arrangement that wins against all other arrangements, then we should still be able to find the optimal *mixed strategy.*

Exercise 8.10 *For the Markov chain defined above, select a pair of configurations and determine the following.*

- *Average time to hit the goal*
- *Probability of scoring left before right goal*
- *Long run behavior of the ball.*

[2] Source: **football** in A. Hornby, *Oxford Advanced Learner's Dictionary of Current English* , Oxford University Press 1974

A brief review of game theory

In *game theory*, a rational player is supposed to minimize her losses against the best choice of the opponent. The terminology uses minimax= minimize maximal loss, maximin= maximize minimal gain; the amazing fact is that these are equal and often the optimal strategies are randomized (mixed).

The optimal strategy in a game is the best (in the above minimax sense) randomized strategy against every deterministic strategy of the opponent.

Problem 8.11 *What is the "optimal" arrangement of players in soccer?*

Modifications

As given above, the model of the soccer game is simple enough to be analyzed without computer. In a more realistic model one could consider additional factors.

- The field can be partitioned into more pieces
- The field is a two-dimensional rectangle.
- Players can move between neighboring portions of the game field.
- Players react to the position of the ball
- Players react to the positions of other players
- Some soccer players are more skilled.

8.3 One step analysis

For homogeneous Markov chains a surprising number of quantities of interest can be computed using the so called *one step analysis.* The idea is to find out what happens to the quantity of interest within one step of the evolution, and solve the resulting difference equation.

Example 8.2 *In the setup of Section 7.5.3, suppose* $\Pr(\xi = 1) = p = 1 - \Pr(\xi = -1)$. *Let T be the first time random walk reaches $L > 0$ or $C < 0$. Find the probability of winning (ruining the casino)* $\Pr(X_T = C|X_0 = 0)$.

Note that evven though we are interested in a single number $\Pr(X_T = C|X_0 = 0)$, *the fist-step analysis requires computing probabilities* $p(x) = \Pr(X_T = C|X_0 = x)$ *for all initial positions x.*

Example 8.3 *Suppose* $\Pr(\xi = 1) = p, \Pr(\xi = -1)1 - p$. *Let T be the first time random walk reaches $L > 0$ or $C < 0$. Find the average length of the game $E_0(T)$.*

Note that the fist-step analysis requires computing $m(x) = E_x(T)$ for all initial positions x.

Problem 8.12 *On average, how long does it take for a symmetric random walk on a square lattice to exit from a $d \times d$ square?*

Problem 8.13 *For the Markov chain defined in Section 8.2.1, select a pair of player configurations and determine the following.*

- *Average time to hit the goal*
- *Probability of scoring left before right goal*
- *Long run behavior of the ball.*

Problem 8.14 *A fair coin is tossed repeatedly until $k = 2$ successive heads appear. What is the average number of tosses required? (Hint: see Problem 8.1, or run a simulation.)*

Problem 8.15 *One of the quality control rules is to stop the process when on $k = 8$ successive days the process runs above, or below, specification line. On average, how often a process that does not need any adjustment will be stopped by this rule?*

8.3.1 Example: vehicle insurance claims

Suppose that you have an insurance contract for a vehicle. The premium payment is due yearly in advance and there are four levels $P_1 > P_2 > P_3 > P_4$. The basic premium is P_1, unless no claims were made in the previous year. If no claims were made in a given year, then the premium for the following year is upgraded to the next category.

Suppose that the probability of the damage larger than s hundred dollars is e^{-s} (substitute your favorite density for the exponential density used in this example). Because of the incentive of lowered premium, not all damage should be reported. The goal of this example is to determine numbers s_1, s_2, s_3, s_4 above which the claims should be filed; s_j is a cutoff used in a year when premium P_j is paid.

Let X_t be the type of premium paid in t-th year. Clearly, the transitions are $i \mapsto 1$ with probability e^{-s_i}, $1 \mapsto 2$ with probability $1 - e^{-s_1}$, etc.

In the long run, the yearly cost is

$$C(s_1, s_2, s_3, s_4) = \sum \pi_j (P_j + m_j), \tag{8.5}$$

where π_j are the equilibrium probabilities and m_j are average un-reimbursed costs: $m_j = 100 \int_0^{s_j} s e^{-s}\, ds$. The optimal claim limits follow by minimizing expression (8.5).

Exercise 8.16 *Find optimal s_j when $P_1 = 800, P_2 = 720, P_3 = 650, P_4 = 600$ (about 10% discount).*

8.3.2 Example: a game of piggy

In a game of piggy, each player tosses two dice, and has an option of adding the outcome to his score, or rolling again. The game ends when the first player exceeds 100 points.

Each player has to toss the dice at least once per turn. The player can choose to toss the dice as many times as (s)he wishes as long as no ace occurs. However the current total is added to player's score only when the player ends his turn voluntarily.

If an ace occurs, then the player's turn is over, and his score is not increased. If two aces occur then the score is set to zero.

A player chooses the following strategy: toss the dice for as long as an ace occurs, or the sum of current subtotal+score exceeds number K. (If her score exceeds K she tosses the dice once, as this is required by the rules.) The quantity to optimize is the average number of turns that takes the player to reach the score of a hundred. The number of turns under this strategy is the number of aces when the score is less than K, plus the number of throws when the score is larger than K.

If X_k denotes players score after the k-th throw, then clearly X_k is a Markov chain with the finite number of states, and with chance $\frac{1}{36}$ of returning to zero at each turn.

Which value of K minimizes the average number of turns that takes the player to reach a hundred?

A simple-minded computation would just take into account the average gain and disregard the score completely. If the players subtotal is t then after the additional throw the total is on average $23/36 \times (t+7)$. This is less then t for $t \geq 12$, thus there is no point continuing beyond 12. Is this conclusion correct? Should the player choose to stop once the total on the dice exceeds 12?

Exercise 8.17 *What is the average number of turns it takes to reach a 100 under this strategy?*

A less simple-minded computation would take into account the average gain: If the players score is s then after the additional throw his score on average is $1/36 \times 0 + 1/3 \times s + 23/36 \times (s+7)$ which is more then s for $s < 7 \times 23$. Is this conclusion correct? Should the player always choose to toss again?

Exercise 8.18 *What is the average number of turns it takes to reach a 100 under this strategy?*

Exercise 8.19 *What is the average number of turns it takes to reach a 100 under the cautious strategy of no additional tosses?*

Another computation would take into account the current score s and current subtotal t. After the additional throw the numbers on average are $s_1 = 1/36 \times 0 + 35/36 \times s$, $t_1 = 23/36 \times (t+7)$. On average, $t_1 + s_1 > t + s$ when $t < 12 - s/13$.

More complicated strategies could depend on current totals of other players, current score of the player, and the subtotal in some more complicated fashion. For instance, if s denotes the score, t denotes current subtotal, one could stop using two-parameter criterion $t > A - Bs$. This may have seemingly different effects than the previous strategy: when A is large, and score s is close to 0 there is no need for stopping; but if accumulated score s is large, the player may behave more cautiously.

One could optimize the probability of winning against $k = 3, 4$ players instead of just minimizing the average number of tosses. The latter puts this problem into game theory framework. Simulations seem to indicate that the strategy based on $t < 25 - \frac{1}{9}s$ works well against inexperienced human players.

8.4 Recurrence

A state x of Markov chain is recurrent, if with probability one the chain returns back to x. Otherwise it is called transient. If X_t is irreducible then all states are simultaneously either recurrent, or transient.

Theorem 8.4.1 *State x is recurrent iff $\sum P_{x,x}(t) = \infty$.*

Proof. Let M be the number of times X_t returns to state x.

Let f be the probability of returning to x. State x is recurrent iff $f = 1$. Suppose $f < 1$ and let M be the number of returns. Clearly $\Pr_x(M \geq k) = f^k$, thus $E_x(M) = f/(1-f) < \infty$. Since $M = \sum_t I_{\{X_t=x\}}$, $E_x(M) = \sum P_{x,x}(t)$.

□

Interesting fact: simple random walks in R^d are recurrent when $d < 3$, transient when $d \geq 3$. However the return times have infinite average.

Theorem 8.4.2 *Let T be a return time, and suppose $m(x) = E_x T < \infty$. Then $P_{x,x}(t) \to 1/m(x)$ as $t \to \infty$.*

Problem 8.20 *Suppose Markov chain X_t moves to the right $k \mapsto k+1$ with probability $1/k$ or returns to 1 with probability $1 - \frac{1}{k}$, $k = 1, 2, \dots$. Find its stationary distribution, and the average return time to state k.*

8.5 Simulated annealing

This sections describes a more refined method for randomized search of minima of functions.

Suppose a finite set V is given, and we are to minimize a function $U : V \to \mathbb{R}$.

The first step of design is to specify a *connected* directed graph $\mathcal{G} = (V, E)$. In other words, for every point $u \in V$ we need to pick the set of directed edges (u, v) for the markov chain to follow from state u. (This step is usually performed for computational efficiency; theoretically, all possible transitions could be admissible.)

The second step is to choose "control parameter" θ that will vary as the program is running.

The third step is to define transition probabilities:

$$P(u, v) = C(u) e^{-\theta(U(v)-U(u))^+}, \tag{8.6}$$

where $C(u, v) = \sum_v e^{-\theta(U(v)-U(u))^+}$ is the norming constant, and the only v's considered are those with $(u, v) \in E$. (Can you explain now why $\mathcal{G}$ shouldn't be the complete graph).

Theorem 8.5.1 *The invariant measure of transition matrix (8.6) is $p_\theta(u) = \frac{1}{Z}e^{\theta U(u)}$*

Proof. To check the invariance condition, denote $\mathcal{N}(u) = \{v : (u, v) \in E\}$.
□

An efficient realization of the above is to pick $v \in \mathcal{N}_u$ at random and accept it with probability $\begin{cases} 1 \text{ if } U(v) \leq U(u) \\ e^{-\Theta U(v)} \text{ otherwise.} \end{cases}$

8.5.1 Program listing

The program is available online, or on the disk.

8.6 Solving difference equations through simulations

In the example below we limit our attention to the one-dimensional problem. This simplifies the notation, while the basic ideas are the same.

Let $u(\mathbf{x}, t)$ be an unknown function of $\mathbf{x} \in \mathbb{R}^d, t \geq 0$. The difference equation we may want to solve is the following discrete analog of the diffusion equation.

$$u(\mathbf{x}, t+1) = u(\mathbf{x}, t) + A\frac{1}{2^d}\sum_{\mathbf{v}=\pm e_k} u(\mathbf{x} + \mathbf{v}, t) \tag{8.7}$$

$$u(\mathbf{x}, o) = u_0(\mathbf{x}) \tag{8.8}$$

The solution is $u(x, t) = E(u_0(S_t))$, where $S_t = \sum_{j \leq t} \epsilon_j$ is the sum of independent random variables with 2^d equally likely values $\pm e_k \in \mathbb{R}^d$.

8.7 Markov Autoregressive processes

Suppose ξ_k is a stationary Markov chain and let X_n be the solution of the difference equation $X_{n+1} - aX_n = \xi_{n+1}$. One can write the transition Matrix for Markov process X_t and try find the stationary distribution for X_0.

A more direct method is based on the fact that the solution of the difference equation is $X_t = a^t X_0 + a^{t-1}\xi_1 + \ldots + a\xi_{t-1} + \xi_t$. Therefore if $|a| < 1$, the stationary initial distribution is $X_0 = \sum a^k \xi_k$. Thus $X_t = \sum_{k=0}^{\infty} a^k \xi_{t-k}$

Problem 8.21 *Suppose ξ_k are i. i. d. Find the covariance EX_0X_k.*

Problem 8.22 *Suppose ξ_k are i. i. d. Find $E(X_k|X_0)$.*

Solutions of higher order difference equations can be easily out into the Markov framework, too. If $X_{n+2} + aX_{n+1} + bX_n = \xi_{n+1}$ then $Y_n = (X_{n+1}, X_n)$ is Markov and satisfies the corresponding equation in matrix form: $Y_{n+1} = AY_n + \Xi_{n+1}$. Therefore the stationary solution exist provided that the eigenvalues of A satisfy inequality $|\lambda_j| < 1$.

8.8 Sorting at random

Efficiency of sorting algorithms is often measured by the number of comparisons required.

To sort efficiently a set S of numbers into ascending order we should find a number y such that about half of the elements of S is below y. Then the total number of steps required is $T(n) \leq T(/n/2)+n+C(n)$, where $C(n)$ is the number of comparisons required to find y.

Random Quick Sort is based on an idea that a random choice of y is good enough on average.

Theorem 8.8.1 *The expected number of comparisons in random quick sort is at most* $2n\ln(n+1)$.

Here is one possible realization of the subprogram:

```
PROGRAM qsrt.bas
'

SUB QuickSort (Index(), Aux(), First%, Last%)
'sorts two matrices in increasing order by the valuies of Index() from pocz to kon
' Note: mixes order of indices corresponding to equal Index(j)

  '** Quick-sort (ascending) the fields in Array$(), from field First% thru Field Last%
 IF First% >= Last% THEN EXIT SUB
 CONST max = 30
 DIM Lft%(max + 1), Rght%(max + 1)
 Temp% = 1
 Lft%(1) = First%
 Rght%(1) = Last%
 DO
   Start% = Lft%(Temp%)
   Ende% = Rght%(Temp%)
   Temp% = Temp% - 1
   DO
    IndexLft% = Start%
    IndexRght% = Ende%
    x = Index((Start% + Ende%) \ 2)
    DO
     WHILE Index(IndexLft%) < x AND IndexLft% < Ende%
     IndexLft% = IndexLft% + 1
     WEND
     WHILE x < Index(IndexRght%) AND IndexRght% > Start%
     IndexRght% = IndexRght% - 1
     WEND
     IF IndexLft% > IndexRght% THEN EXIT DO
     SWAP Index(IndexLft%), Index(IndexRght%)     '** switch elements
     SWAP Aux(IndexLft%), Aux(IndexRght%)
      IndexLft% = IndexLft% - (IndexLft% < Ende%)
      IndexRght% = IndexRght% + (IndexRght% > Start%)
    LOOP
    IF IndexRght% - Start% >= Ende% - IndexLft% THEN
      IF Start% < IndexRght% THEN
        Temp% = Temp% + 1
        Lft%(Temp%) = Start%
        Rght%(Temp%) = IndexRght%
      END IF
      Start% = IndexLft%
    ELSE
      IF IndexLft% < Ende% THEN
        Temp% = Temp% + 1
        Lft%(Temp%) = IndexLft%
        Rght%(Temp%) = Ende%
      END IF
       Ende% = IndexRght%
    END IF
   LOOP WHILE Start% < Ende%
  IF Temp% > max THEN Temp% = 0
 LOOP WHILE Temp%
'

END SUB
```

8.9 An application: find k-th largest number

The following theorem occasionally helps to estimate the average time of accomplishing a numerical task.

Theorem 8.9.1 *Suppose $g(x)$ is increasing (non-decreasing) function and X_t is a Markov chain on $\mathbb{N}$ that moves left only and $E(X_{t+1}|X_t = m) \leq m + g(m)$. Let T be the time of reaching 1. Then $E_n(T) \leq \int_1^n \frac{1}{g(x)}\,dx$.*

Proof. By induction $E_n(T) \leq 1 + E\int_1^X \frac{dx}{g(x)} = 1 + \int_1^n \frac{dx}{g(x)} - E\int_X^n \frac{dx}{g(x)} \leq 1 + \int_1^n \frac{dx}{g(x)} - E\int_X^n \frac{dx}{g(n)} \leq 1 + \int_1^n \frac{dx}{g(x)} + E\frac{X-n}{g(n)}$ □

As an application we consider the following algorithm to pick the k-th number in order from a set S of n numbers.

1. Initialize $S_1 = S, S_2 = \emptyset$.

2. Pick y at random from S_1.

3. Revise sets $S_1 = \{x : x < y\}, S_2 = \{x : x > y\}$.

4. If $|S_1| = k - 1$ then y was found.

5. If $|S_1| > k$ then repeat the process with new S_1.

6. If $|S_1 < k-1$ then swap S_1 and S_2, replace k by $k - |S_1| - 1$, and repeat the process.

Problem 8.23 *Estimate the average running time of the above algorithm. (ANS: $ET \leq 4\ln n$).*

Chapter 9

Branching processes

Suppose certain objects multiply independently and in discrete time intervals. Each object at the end of every period produces a random number ξ of descendants (offspring) with the probability distribution $p_k = \Pr(\xi = k)$. Let X_t be the total number of objects at t-th generation.

Then in distribution $X_{t+1} = \sum_{j=1}^{X_t} \xi_j$. The three questions of interest are the average size of the population, its variance, and the probability of extinction.

Definition 9.0.1 *By extinction we mean the event that the random sequence $\{X_t\}$ consists of zeros for all but the finite number of values of $t \in \mathbb{N}$.*

Probability of extinction by time n can be found directly from the first-step-analysis: numbers $u_n = \Pr(X_n = 0)$ satisfy

$$u_{n+1} = \sum_{k=0}^{\infty} p_k (u_n)^k. \tag{9.1}$$

9.1 The mean and the variance

Let $\mu_n = E(X_n), V_n = Var(X_n)$. By Theorem 7.5.1 and induction we have

$$\mu_n = \mu^n \tag{9.2}$$

$$V_n = \sigma^2 \mu^{n-1}(1-\mu^n)/(1-\mu). \tag{9.3}$$

9.2 Generating functions of Branching processes

Let $g(z) = \sum_{k=0}^{\infty} p_k z^k$ be the generating function. Clearly $g(z) \approx p_0 + \mu z$ for small z. Equation (9.1) for probabilities u_n of extinction by the n-th generation is $u_{n+1} = g(u_n)$.

Theorem 9.2.1 (Watson(1874)) *The generating function $H_n(z)$ of X_n is the n-fold composition (iteration) of g.*

Proof. $E(z^{X_{n+1}}|X_n = k) = (g(z))^k$. Thus by total probability formula (2.34) $H_{n+1}(z) = Ez^{X_{n+1}} = \sum_k (g(z))^k \Pr(X_n = k) = H_n(g(z))$. □

In particular, $EX_n = \frac{d}{dz} g^{\circ(n)}(z)|_{z=1} = \mu^n$ and $u_n = \Pr(X_n = 0) = g^{\circ(n)}(0)$.

Problem 9.1 *Prove (9.2) using moment generating function directly.*

9.2.1 Extinction

Notice that if there is a limit of $q = \lim_{n\to\infty} g^{\circ(n)}(0)$, then it has to satisfy the equation

$$g(s) = s. \tag{9.4}$$

Since X_t is integer valued and the events $A_n = \{X_t = 0\}$ are decreasing, by continuity (1.1) the probability of extinction $q = \lim_{n\to\infty} \Pr(X_n = 0)$ exists.

Theorem 9.2.2 *If $EX_1 \leq 1$, the extinction probability $q = 1$. If $EX_1 > 1$, the extinction probability is the unique nonnegative solution less than 1 of the equation (9.4).*

Proof. This is best understood by graphing $g(s)$ and marking the iterates on the diagonal.

Check by induction that $g^n(0) < 1$ for all $n \in \mathbb{N}$. If there is a solution $s_0 < 1$ of $g(s) = s$, then it is the "attractive point" of the iteration.

□

9.3 Two-valued case

Below we re-analyzes the growth model presented in Section 7.5.4. Suppose that the probabilities of offspring are $p_0 = \theta, p_2 = 1 - \theta$.

The generating function is $g(z) = \theta + (1-\theta)z^2$. Asymptotic probability of extinction solves quadratic equation $(1-\theta)z^2 - z + \theta = 0$. The roots are $z_1 = 1$ and $z_2 = \frac{\theta}{1-\theta}$. In particular, probability of extinction is 1 when $\theta \geq \frac{1}{2}$.

When $\theta = 1/4$ probabilities of extinction at $n-th$ generation are $u_0 = 0, u_1 = .25, u_2 = .296875, u_3 = .3161, u_4 = .3249, u_5 = .33127, u_6 = .3323$.

9.4 Geometric case

Suppose that the probabilities of offspring are $p_0 = 1 - \rho\theta, p_k = \theta(1-\rho)\rho^k$.

The generating function is $g(z) = (1 - \rho\theta) + \theta(1-\rho)\rho\frac{z}{1-\rho z}$. The most interesting feature of this moment generating function is that it can be readily composed.

Lemma 9.4.1 *The composition of fractional linear functions $f(z) = \frac{a+bz}{c+dz}$ and $g(z) = \frac{A+Bz}{C+Dz}$ is a fractional linear function (with the coefficients given by the matrix multiplication!).*

Asymptotic probability of extinction has to solve quadratic equation $1-\rho\theta+\theta(1-\rho)\rho\frac{z}{1-\rho z}=z$. The roots are $z_1=1$ and $z_2=\frac{\theta}{1-\theta}$. In particular, probability of extinction is 1 when $\theta\geq\frac{1}{2}$.

Since the iterates of the generating function can actually be written down explicitly, in geometric case $\Pr(X_n=0)=1-m^n(1-p_e)/(m^n-p_e)$ is explicit. Here $p_e=z_2$, $m=....$

Problem 9.2 *Suppose that in a branching process the number of offspring of the initial seedling has a distribution with generating function $F(z)$. Each member of the next generation has the number of offspring whose distribution has generating function $G(z)$. The distributions alternate between generations.*

- *Find the extinction probability in terms of F, G.*
- *What is the average population size?*

Problem 9.3 *In a simple model of linear immunological response, the doubling probability p of the population of bacteria changes with time due to the increased number of lymphocytes. If there are $X(t)$ bacteria at t-th generation, then assume $p=a/(t+a)$. Find the probability u_t of extinction by t-th generation for infection by 10 bacteria. What is the average length of the disease?*

Chapter 10

Multivariate normal distribution

Univariate normal distribution, standardization, and its moment generating function were introduced in Chapter 4. Below we define multivariate normal distribution.

10.1 Multivariate moment generating function

We follow the usual linear algebra notation. Vectors are denoted by small bold letters $\mathbf{x}, \mathbf{v}, \mathbf{t}$, matrices by capital bold initial letters $\mathbf{A}, \mathbf{B}, \mathbf{C}$ and vector-valued random variables by capital boldface $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$; by the dot we denote the usual dot product in $\mathbb{R}^d$, ie. $\mathbf{x} \cdot \mathbf{y} := \sum_{j=1}^d x_j y_j$; $\|\mathbf{x}\| = (\mathbf{x} \cdot \mathbf{x})^{1/2}$ denotes the usual Euclidean norm. For typographical convenience we sometimes write $(a_1, \dots, a_k)$ for the vector $\begin{bmatrix} a_1 \\ \vdots \\ a_k \end{bmatrix}$. By $\mathbf{A}^T$ we denote the transpose of a matrix $\mathbf{A}$.

Below we shall also consider another scalar product $\langle \cdot, \cdot \rangle$ associated with the normal distribution; the corresponding semi-norm will be denoted by the triple bar $|||\cdot|||$.

Definition 10.1.1 *An $\mathbb{R}^d$-valued random variable $\mathbf{Z}$ is multivariate normal, or Gaussian (we shall use both terms interchangeably; the second term will be preferred in abstract situations) if for every $\mathbf{t} \in \mathbb{R}^d$ the real valued random variable $\mathbf{t} \cdot \mathbf{Z}$ is normal.*

Example 10.1 *Let $\xi_1, \xi_2, \dots$ be i. i. d. $N(0,1)$. Then $\mathbf{X} = (\xi_1, \xi_2, \dots, \xi_d)$ is multivariate normal.*

Example 10.2 *Let ξ be $N(0,1)$. Then $\mathbf{X} = (\xi, \xi, \dots, \xi)$ is multivariate normal.*

Example 10.3 *Let $\xi_1, \xi_2, \dots$ be i. i. d. $N(0,1)$. Then $\mathbf{X} = (X_1, X_2, \dots, X_T)$, where $X_k = \sum_{j=1}^k \xi_j$ are partial sums, is multivariate normal.*

Clearly the distribution of univariate $\mathbf{t} \cdot \mathbf{Z}$ is determined uniquely by its mean $m = m_{\mathbf{t}}$ and its standard deviation $\sigma = \sigma_{\mathbf{t}}$. It is easy to see that $m_{\mathbf{t}} = \mathbf{t} \cdot \mathbf{m}$, where $\mathbf{m} = E\mathbf{Z}$. Indeed, by linearity of the expected value $m_{\mathbf{t}} = E\mathbf{t} \cdot \mathbf{Z} = \mathbf{t} \cdot E\mathbf{Z}$. Evaluating the moment generating function $M(s)$ of the real-valued random variable $\mathbf{t}\cdot\mathbf{Z}$ at $s=1$ we see that the moment generating function of $\mathbf{Z}$ can be written as

$$M(\mathbf{t}) = \exp(\mathbf{t}\cdot\mathbf{m} + \frac{\sigma_{\mathbf{t}}^2}{2}).$$

10.2 Bivariate normal distribution

In this section we consider a pair of (jointly) normal random variables X_1, X_2. For simplicity of notation we suppose $EX_1 = 0, EX_2 = 0$. Let $Var(X_1) = \sigma_1^2, Var(X_2) = \sigma_2^2$ and denote $corr(X_1, X_2) = \rho$. Then the covariance matrix is $\mathbf{C} = \begin{bmatrix} \sigma_1^2 & \rho \\ \rho & \sigma_2^2 \end{bmatrix}$ and the joint moment generating function is

$$M(t_1, t_2) = \exp(\frac{1}{2}t_1^2\sigma_1^2 + \frac{1}{2}t_2^2\sigma_2^2 + t_1t_2\rho).$$

If $\sigma_1\sigma_2 \neq 0$ we can normalize the variables and consider the pair $Y_1 = X_1/\sigma_1$ and $Y_2 = X_2/\sigma_2$. The covariance matrix of the last pair is $\mathbf{C_Y} = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$; it generates scalar product given by

$$\left\langle \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \right\rangle = x_1y_1 + x_2y_2 + \rho x_1y_2 + \rho x_2y_1$$

and the corresponding (semi)-norm is $\| \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \| = (x_1^2 + x_2^2 + 2\rho x_1x_2)^{1/2}$. Notice that when $\rho = \pm 1$ the semi-norm is degenerate and equals $|x_1 \pm x_2|$.

Denoting $\rho = \sin 2\theta$, it is easy to check that

$$Y_1 = \gamma_1\cos\theta + \gamma_2\sin\theta,$$
$$Y_2 = \gamma_1\sin\theta + \gamma_2\cos\theta$$

for some i.i.d normal $N(0,1)$ r. v. γ_1, γ_2. One way to see this, is to compare the variances and the covariances of both sides.

This implies that the joint density of Y_1 and Y_2 is given by

$$f(x,y) = \frac{1}{2\pi\cos 2\theta}\exp(-\frac{1}{2\cos^2 2\theta}(x^2 + y^2 - 2xy\sin 2\theta)) \tag{10.1}$$

which is a variant of (4.5).

Another representation

$$Y_1 = \gamma_1,$$
$$Y_2 = \rho\gamma_1 + \sqrt{1-\rho^2}\gamma_2$$

illustrates non-uniqueness of the linear representation.

The latter representation makes the following Theorem obvious in the bivariate case.

Theorem 10.2.1 *Let* $\mathbf{X}, \mathbf{Y}$ *be jointly normal.*

(i) If $\mathbf{X}, \mathbf{Y}$ *are uncorrelated, then they are independent.*

(ii) $E(\mathbf{Y}|\mathbf{X}) = \mathbf{m} + A\mathbf{X}$ *is linear*

(iii) $\mathbf{Y} - A\mathbf{X}$ *and* $\mathbf{X}$ *are independent.*

Returning back to random variables X_1, X_2, we have $X_1 = \gamma_1\sigma_1\cos\theta + \gamma_2\sigma_1\sin\theta$ and $X_2 = \gamma_1\sigma_2\sin\theta + \gamma_2\sigma_2\cos\theta$; this representation holds true also in the degenerate case.

10.2.1 Example: normal random walk

In this example we analyze a discrete time Gaussian random walk $\{X_k\}_{0\leq k\leq T}$. Let $\xi_1, \xi_2, \ldots$ be i. i. d. $N(0,1)$. We are interested in explicit formulas for the moment generating function and for the density of the $\mathbb{R}^T$-valued random variable $\mathbf{X} = (X_1, X_2, \ldots, X_T)$, where

$$X_k = \sum_{j=1}^{k} \xi_j \tag{10.2}$$

are partial sums.

Clearly, $\mathbf{m} = 0$. Equation (10.2) expresses $\mathbf{X}$ as a linear transformation $\mathbf{X} = A\mathbf{g}$ of the i. i. d. standard normal vector with

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 1 & 1 & \ldots & 0 \\ \vdots & & \ddots & \vdots \\ 1 & 1 & \ldots & 1 \end{bmatrix}.$$

Therefore from (10.5) we get

$$M(\mathbf{t}) = \exp\frac{1}{2}(t_1^2 + (t_1+t_2)^2 + \ldots + (t_1+t_2+\ldots+t_T)^2).$$

To find the formula for joint density, notice that $\mathbf{A}$ is the matrix representation of the linear operator, which to a given sequence of numbers $(x_1, x_2, \ldots, x_T)$ assigns the sequence of its partial sums $(x_1, x_1+x_2, \ldots, x_1+x_2+\ldots+x_T)$. Therefore, its inverse is the finite difference operator $\Delta : (x_1, x_2, \ldots, x_T) \mapsto (x_1, x_2 - x_1, \ldots, x_T - x_{T-1})$. This implies

$$\mathbf{A}^{-1} = \begin{bmatrix} 1 & 0 & 0 & \ldots & \ldots & 0 \\ -1 & 1 & 0 & \ldots & \ldots & 0 \\ 0 & -1 & 1 & \ldots & \ldots & 0 \\ 0 & 0 & -1 & \ldots & \ldots & 0 \\ \vdots & \ddots & & \ddots & & \vdots \\ 0 & \ldots & 0 & \ldots & -1 & 1 \end{bmatrix}.$$

Since $\det\mathbf{A} = 1$, we get

$$f(\mathbf{x}) = (2\pi)^{-n/2}\exp -\frac{1}{2}(x_1^2 + (x_2-x_1)^2 + \ldots + (x_T - x_{T-1})^2). \tag{10.3}$$

Interpreting $\mathbf{X}$ as the discrete time process $X_1, X_2, \ldots$, the probability density function for its trajectory $\mathbf{x}$ is given by $f(\mathbf{x}) = C\exp(-\frac{1}{2}\|\Delta\mathbf{x}\|^2)$.

Expression $\frac{1}{2}\|\Delta\mathbf{x}\|^2$ can be interpreted as proportional to the kinetic energy of the motion described by the path $\mathbf{x}$; assigning probabilities by $Ce^{-Energy/(kT)}$ is a well known practice in statistical physics. In continuous time, the derivative plays analogous role.

10.3 Simulating a multivariate normal distribution

To simulate any d-dimensional normal distribution we need only to simulate d independent $N(0,1)$ random variables and use linear representations like in Theorem 10.4.2. For such simulation the covariance matrix needs to be inverted and diagonalized, a numerical nuisance in itself. When the multivariate normal distribution occurs as the so-called time series, a method based on Fourier expansion is then convenient, see Section 11.4, or the introductory examples in Chapter 12.

10.3.1 General multivariate normal law

The linear algebra results imply that the moment generating function corresponding to a normal distribution on $\mathbb{R}^d$ can be written in the form

$$M(\mathbf{t}) = \exp(\mathbf{t}\cdot\mathbf{m} + \frac{1}{2}\mathbf{C}\mathbf{t}\cdot\mathbf{t}). \tag{10.4}$$

10.4 Covariance matrix

Theorem 3.2.2 identifies $\mathbf{m} \in \mathbb{R}^d$ as the mean of the normal random variable $\mathbf{Z} = (Z_1, \dots, Z_d)$; similarly, double differentiation $M(\mathbf{t})$ at $\mathbf{t} = 0$ shows that $\mathbf{C} = [c_{i,j}]$ is given by $c_{i,j} = Cov(Z_i, Z_j)$. This establishes the following.

Theorem 10.4.1 *The moment generating function corresponding to a normal random variable $\mathbf{Z} = (Z_1, \dots, Z_d)$ is given by (10.4), where $\mathbf{m} = E\mathbf{Z}$ and $\mathbf{C} = [c_{i,j}]$, where $c_{i,j} = Cov(Z_i, Z_j)$, is the covariance matrix.*

From (10.4) and linear algebra we get also

$$M(\mathbf{t}) = \exp(\mathbf{t}\cdot\mathbf{m} + \frac{1}{2}(\mathbf{A}\mathbf{t})\cdot(\mathbf{A}\mathbf{t})). \tag{10.5}$$

We have the following multivariate generalization of (4.1).

Theorem 10.4.2 *Each d-dimensional normal random variable $\mathbf{Z}$ has the same distribution as $\mathbf{m} + \mathbf{A}\mathbf{g}$, where $\mathbf{m} \in \mathbb{R}^d$ is deterministic, $\mathbf{A}$ is a (symmetric) $d \times d$ matrix and $\mathbf{g} = (\gamma_1, \dots, \gamma_d)$ is a random vector such that the components $\gamma_1, \dots, \gamma_d$ are independent $N(0,1)$ random variables.*

Proof. Clearly, $E\exp(\mathbf{t}\cdot(\mathbf{m} + \mathbf{A}\mathbf{g})) = \exp(\mathbf{t}\cdot\mathbf{m})E\exp(\mathbf{t}\cdot(\mathbf{A}\mathbf{g}))$. Since the moment generating function of $\mathbf{g}$ is $E\exp(\mathbf{x}\cdot\mathbf{g}) = \exp\frac{1}{2}\|\mathbf{x}\|^2$ and $\mathbf{t}\cdot(\mathbf{A}\mathbf{g}) = (\mathbf{A}^T\mathbf{t})\cdot\mathbf{g}$, therefore we get $E\exp(\mathbf{t}\cdot(\mathbf{m} + \mathbf{A}\mathbf{g})) = \exp\mathbf{t}\cdot\mathbf{m}\exp+\frac{1}{2}\|\mathbf{A}^T\mathbf{t}\|^2$, which is another form of (10.5). □

10.4.1 Multivariate normal density

Now we consider the multivariate normal density. The density of independent $\gamma_1, \dots,$ in Theorem 10.4.2 is the product of the one-dimensional standard normal densities, ie.

$$f_{\mathbf{g}}(\mathbf{x}) = (2\pi)^{-d/2} \exp(-\frac{1}{2}\|\mathbf{x}\|^2).$$

Suppose that $\det \mathbf{C} \neq 0$, which ensures that $\mathbf{A}$ is nonsingular. By the change of variable formula, from Theorem 10.4.2 we get the following expression for the multivariate normal density.

Theorem 10.4.3 *If $\mathbf{Z}$ is centered normal with the nonsingular covariance matrix $\mathbf{C}$, then the density of $\mathbf{Z}$ is given by*

$$f_{\mathbf{Z}}(\mathbf{x}) = (2\pi)^{-d/2}(\det \mathbf{A})^{-1} \exp(-\frac{1}{2}\|\mathbf{A}^{-1}\mathbf{x}\|^2),$$

or

$$f_{\mathbf{Z}}(\mathbf{x}) = (2\pi)^{-d/2}(\det \mathbf{C})^{-1/2} \exp(-\frac{1}{2}\mathbf{C}^{-1}\mathbf{x}\cdot\mathbf{x}),$$

where matrices $\mathbf{A}$ and $\mathbf{C}$ are related by $C = A \times A^T$.

In the nonsingular case the density expression implies strong integrability.

Theorem 10.4.4 *If $\mathbf{Z}$ is normal, then there is $\epsilon > 0$ such that*

$$E\exp(\epsilon\|\mathbf{Z}\|^2) < \infty.$$

Remark 3 Theorem 10.4.4 holds true also in the singular case and for Gaussian random variables with values in infinite dimensional spaces.

10.4.2 Linear regression

For general multivariate normal random variables $\mathbf{X}$ and $\mathbf{Y}$ we have the following linearity of regression result.

Theorem 10.4.5 *If $(\mathbf{X}, \mathbf{Y})$ has jointly normal distribution on $\mathbb{R}^{d_1+d_2}$, then*

$$E\{\mathbf{X}|\mathbf{Y}\} = \mathbf{a} + \mathbf{Q}\mathbf{Y}; \tag{10.6}$$

Random vectors $\mathbf{Y} - \mathbf{Q}\mathbf{Y}$ and $\mathbf{X}$ are stochastically independent.

Vector $\mathbf{a} = \mathbf{m}_{\mathbf{X}} - \mathbf{Q}\mathbf{m}_{\mathbf{Y}}$ and matrix $\mathbf{Q}$ are determined by the expected values $\mathbf{m}_{\mathbf{X}}, \mathbf{m}_{\mathbf{Y}}$ and by the (joint) covariance matrix $\mathbf{C}$ (uniquely if the covariance $\mathbf{C}_{\mathbf{Y}}$ of $\mathbf{Y}$ is nonsingular). To find $\mathbf{Q}$, multiply (10.6) (as a column vector) from the right by $(\mathbf{Y} - E\mathbf{Y})^T$ and take the expected value. By Theorem A.2.1(i) we get $\mathbf{Q} = \mathbf{R} \times \mathbf{C}_{\mathbf{Y}}^{-1}$, where we have written $\mathbf{C}$ as the (suitable) block matrix $\mathbf{C} = \begin{bmatrix} \mathbf{C}_{\mathbf{X}} & \mathbf{R} \\ \mathbf{R}^T & \mathbf{C}_{\mathbf{Y}} \end{bmatrix}$.

Problem 10.1 *For the random walk from Section 10.2.1, what is $E(X_k|X_1, \dots, X_{k-1}, X_{k+1}, \dots, X_n)$?*

Problem 10.2 *Suppose $X_1, \dots, X_d$ are jointly normal, $EX_j = 0, EX_j^2 = 1$ and all covariances $EX_iX_j = \rho$ are the same for $i \neq j$.*

Find $E(X_1|X_2, X_3, \dots, X_d)$. (Notice that in this example $\rho > -1/d$.)

10.5 Gaussian Markov processes

Suppose $(X_t)_{t=0,1,\dots}$ is a Markov chain with multivariate normal distribution. That is, suppose X_0 is normal, and the transition probabilities are normal, too.

Without loss of generality we assume $EX_t = 0, EX_t^2 = 1$ and let $EX_0X_1 = \rho$. Then $E(X_{t+1}|X_t) = \rho X_t$ and therefore $EX_0X_t = \rho^t$. This means that the covariance matrix of the Markov Gaussian process depends on one parameter ρ only. Comparing the answer with Section 8.7 we find out that all homogeneous Markov Gaussian processes have the form $X_t = \sum_{k=0}^{\infty} \gamma_{k+t}\rho^k$, where γ_k are independent normal r. v.

Chapter 11

Continuous time processes

Continuous time processes are the families of random variables X_t, with $t \geq 0$ interpreted as time.

11.1 Poisson process

Poisson distribution occurs as an approximation to binomial. Another reason for its occurrence is related to exponential random variables and counting customers in queues. The latter is perhaps the most efficient way of simulating Poisson random variables, see Section 6.2.4. A related reason for the frequent use of Poisson distribution in modeling is the law of rare events.

Definition 11.1.1 *A Poisson process of intensity $\lambda > 0$ is an integer-valued stochastic process $\{N_t :\}$ with the following properties.*

- $N_0 = 0$
- *For $s > 0, t > 0$ random variable $N_{t+s} - N_t$ has Poisson distribution with parameter λs.*
- *N_t has independent increments*

Suppose cars pass by an intersection and the times between their arrivals are independent and exponential. Thus we are given i. i. d sequence T_j of exponential r. v. (with parameter λ). The number of cars that passed by within time interval $(0, t)$ is a random variable N_t. Clearly, N_t is the first integer k such that $T_1 + \ldots T_k > t$.

Theorem 11.1.1 *For $t > 0, s > 0$ random variable $N(t+s) - N(t)$ is independent of N_t and has Poisson distribution $Poiss(s\lambda)$.*

Proof. We will prove only that N_t has the Poisson distribution. To simplify notation assume that exponential r.'v. have parameter $\lambda = 1$. We prove the formula $\Pr(N_t = k) = \frac{t^k}{k!}e^{-t}$ by induction.

$k = 0$: $\Pr(N_t = 0) = \Pr(T > t) = \int_t^\infty e^{-x}\,dx = e^{-t}$

Suppose the formula is true for k. Then

$$\Pr(N_t = k+1) = \Pr(T_1 + \ldots + T_{k+1} > t) = \int_t^\infty 1/,\ (k+1) x^k e^{-x}\, dx.$$

Integrating by parts we check that

$$\Pr(N_t = k+1) = \frac{t}{k+1} \Pr(N_t = k).$$

□

Similar processes are considered in reliability theory, and in queueing theory, also for non-exponential sojourn times T_j.

Problem 11.1 *Assume a device fails when a cumulative effect of k shocks occurs. If the shocks occur according to the Poisson process with parameter λ, what is the density function for the life T of the device?*

Problem 11.2 *Let $f(x,t) = Ef(x+N_t)$, where N_t is the Poisson process of intensity $\lambda = 1$. Show that $\frac{\partial f}{\partial t} = f(x+1) - f(x)$.*

In particular, $p_t(k) = \Pr(N_t = k)$ satisfies $\frac{\partial p_t(k)}{\partial t} = p_t(k+1) - p_t(k)$.

Problem 11.3 *Customers arrive at a facility at random according to a Poisson process of rate λ. The customers are dispatched (processed) in groups at deterministic times $T, 2T, 3T, \ldots$.*

There is a waiting time cost c per customer per unit of time, and a dispatch cost K.

- *What is the mean total cost (customer waiting+dispatch cost) per unit of time during the first cycle?*
- *What value of T minimizes the mean cost per unit of time?*

Problem 11.4 *Find the mean EN_t, variance $Var(N_t)$ and the covariance $cov(N_t, N_s)$.*

Problem 11.5 *Let $X(t) = (-1)^{N_t}$. Find the mean EX_t, variance $Var(X_t)$ and the covariance $cov(X_t, X_s)$.*

The rate λ in the Poisson process has probabilistic interpretation:

$$\lambda = \lim_{h \to 0} \frac{\Pr(N_{t+h} - N_t = 1)}{h} \tag{11.1}$$

In many applications we wish to consider rates $\lambda(t)$ that vary with time. The corresponding process is just a time-change of the Poisson process $X(t) = N_{\Lambda(t)}$, where $\Lambda(t) = \int_0^t \lambda(s) ds$.

11.1.1 The law of rare events

Let $N(I)$ denote the number of events that occur in interval I. We make the following postulates.

1. If intervals $I_1, \dots I_r$ are disjoint, then random variables $\{N(I_j)\}$ are independent.
2. For any t and $h > 0$ the probability distribution of $N((t, t+h])$ does not depend on t.
3. $\frac{\Pr(N(I_h)\geq 2)}{h} \to 0$ as $h \to 0$
4. $\frac{\Pr(N(I_h)=1)}{h} \to \lambda$ as $h \to 0$

Theorem 11.1.2 *$N_t = N((0,t])$ is the Poisson process of intensity λ.*

Example 11.1 *A careless programmer assigns memory locations to the variables in his program[1] at random . Suppose that there are $M \to \infty$ locations and $N = \lambda M$ variables. Let X_i be the number of variables assigned to each location. If each location is equally likely to be chosen, show that*

- $\Pr(X_i = k) \to e^{-\lambda}\lambda^k/k!$ *as $N \to \infty$*
- *X_i and X_j are independent in the limit for $i \neq j$.*

In the limit, what fraction of memory locations has two or more variables assigned?

Example 11.2 *While testing a program, the number of bugs discoverd in the program follows the Poisson process with intensity $\lambda = 5$ errors per hour. Tester's fiance enters the test area and agrees to wait for the tester to find just one more bug. How long will she wait on average: 12 minutes, or 6 minutes?*

11.1.2 Compound Poisson process

The Poisson process N_t counts the number of events. If each event results in a random (and independent) outcome ξ_j, then the total is the compound Poisson process $Z(t) = \sum_{j=1}^{N_t} \xi_j$.

The moments and also the moment generating function of $Z(t)$ can be determined through conditioning. Section 7.5.1 implies that if $E\xi = \mu$, $Var(\xi) = \sigma^2$ then $E(Z(t)) = \lambda\mu t$, $Var(Z(t)) = \lambda(\sigma^2 + \mu^2)t$.

[1] Variable aliasing is the mistake of assigning the same location to two or more variables in a program.

Examples

1. *Risk assessment:* Insurance company has M customers. Suppose claims arrive at an insurance company in accordance with the Poisson process with rate λM. Let Y_k be the magnitude of the k-th claim. The net profit of the company is then $Z(t) - M\theta t$, where θ is the (fixed in this example) premium.

2. *A shock model:* Let N_t be the number of shocks to a system up to time t and let ξ_k denote the damage or wear incurred by the k-th shock.

   If the damage accumulates additively, then $Z(t)$ represents the total damage sustained up to time t. Suppose that the system continues to operate until this total damage is less than some critical value c and fails otherwise. Then the (random) failure time T satisfies $T > t$ if and only if $Z(t) < c$. Therefore $\Pr(T > t) = \sum_{n=0}^{\infty} \Pr(\sum_{k=1}^{n} \xi_k \leq z | N_t = n)(\lambda t)^n e^{-\lambda t}/n!$. Thus $ET = \frac{1}{\lambda}\sum_{n=0}^{\infty}\Pr(\sum_{j=1}^{n}\xi_j \leq c)$. In particular if ξ_k are exponential $ET = \frac{1+c\mu}{\lambda}$.

11.2 Continuous time Markov processes

Given a discrete-time Markov chain, there are many ways of "running it" in continuous time. One particular method is to make the moves at random moments of time. If these instances are exponential, then the resulting continuous-time process is Markov, too. This is the so-called *embedded Markov chain.*

Non-pathological continuous time Markov processes with countable state space have *embedded Markov chain* representation. In such representation we run a continuous time clock based on the independent exponential random variables. Once the time comes, we select the next position according to the transition probabilities of a discrete-time Markov chain.

The theory of continuous time Markov chains (that is — processes with countable state space) is similar to discrete time theory. The linear first-order difference equations for probabilities are replaced by the systems of first-order differential equations.

Example 11.3 *Suppose X_n is a two-state Markov chain with the following transitions: $0 \mapsto 1$ with probability a, $1 \mapsto 0$ with probability b. From Section 8.1 we know that $P(X_k = 1) \to a/(a + b)$ as $k \to \infty$.*

Let T_k be i. i. d. exponential r. v. and let $Y(t) = X_k$ when $T_1 + \ldots + T_k < t < T_1 + \ldots + T_{k+1}$.

Function $p(t) = \Pr(Y(t) = 1)$ satisfies differential equation: $p'(t) = -\lambda p(t) + \lambda b p(t) + \lambda a(1 - p(t)$. Indeed, $\Pr(Y(t + h) = 1) \approx \Pr(Y(t) = 1, T > h) + \Pr(Y(t) = 0, T < h)$.

Therefore $p(t) = a/(a + b) + b/(a + b)\exp(-\lambda(1 - b + a)t)$.

Here is a slightly different method to run the continuous time finite-valued Markov chain.

Pick the initial value according to prescribed distribution. Then select an exponential random variable for each of the possible transitions. Each of these can have different parameter λ_k. Then select the smallest, $T = \min T_j$. It can be shown that T is exponential with parameter $\sum \lambda_j$ and that $\Pr(T = T_j) = \frac{\lambda_j}{\sum_k \lambda_k}$.

11.2.1 Examples of continuous time Markov processes

Despite many similarities, continuous models differ from discrete ones in their predictions.

Growth model

In Section 7.5.4 we considered discrete-time growth model which assumed that bacteria divide at fixed time intervals. This assumption is well known not to be satisfied — mitosis is a process that consists of several stages of various lengths, of which the longest may perhaps be considered to have exponential density. In this section we shall assume that a colony of bacteria consists of $X(t)$ cells which divide at exponential moments of time, or die. We assume individual cells multiply at a rate a and die at a rate b. One way to interpret these numbers is to assume that there are two competing effects: extinction or division. When there are k such cells, the population grows one cell at a time, rate of growth is ak, rate of death is kb. We assume $a > b$.

Let $p_k(t) = \Pr(X(t) = k)$. Then $p_k'(t) = -(a+b)kp_k + (k+1)bp_{k+1} + a(k-1)p_{k-1}$. Population average $m(t) = \sum_k kp_k(t)$ satisfies $m'(t) = (a-b)m(t)$, thus $m(t) = e^{(a-b)t}$.

Problem 11.6 *Suppose X_t is a Markov process whose birth rate is $an + \alpha$ and death rate is bn with $b > a$. This describes a population of species that die out in a given habitat, but have a constant rate of "invasion". One would expect that such competing effects will result in some sort of equilibrium. Find the average population size as $t \to \infty$.*

Example 11.4 *Suppose $X_t = (-1)^{N_t}$, where N_t is the Poisson Process. Is X_t Markov?*

Exercise 11.7 *Customers arrive at a burger outlet at a rate λ, and after exponential service time with mean $1/\mu$ leave.*

- *What is the average number $m(t)$ of customers inside the building t minutes after opening?*
- *On your next visit to a burger outlet, estimate all three averages.*

Exercise 11.8 *Customers arrive at a burger outlet at a rate λ, and after exponential service time with parameter μ leave. If there second cashier is opened, the service time will be reduced twice on average, but the cost of hiring the second cashier is \$5 per hour. A customer purchases of average \$5, with the profit of \$2 over the costs. If k customers are waiting in the line, the next person driving by will stop with probability 2^{-k}.*

- *What rate λ (if any) will justify hiring the second cashier?*
- *What is the average number $m(t)$ of customers inside the building t minutes after opening?*

11.3 Gaussian processes

Continuous-time Markov processes with uncountable state space require more advanced mathematical tools. Only Gaussian case can be briefly mentioned here.

Definition 11.3.1 *A stochastic process $\{X_t\}_{0 \leq t < \infty}$ is Gaussian, if the n-dimensional r. v. $(X_{t_1}, \dots, X_{t_n})$ has multivariate normal distribution for all $n \geq 1$ and all $t_1, \dots, t_n \in [0, \infty)$.*

11.4 The Wiener process

The simplest way to define the Wiener process is to list its properties as follows.

Definition 11.4.1 *The Wiener process $\{W_t\}$ is a Gaussian process with continuous trajectories such that*

$$\begin{aligned} W_0 &= 0; && (11.2)\\ EW_t &= 0 \textit{ for all } t \geq 0; && (11.3)\\ EW_tW_s &= \min\{t, s\} \textit{ for all } t, s \geq 0. && (11.4) \end{aligned}$$

A stochastic process $\{X_t\}_{t \in [0,1]}$ has continuous trajectories if it is defined by a $C[0,1]$-valued random vector, or if all of its paths are continuous[2]. For infinite time interval $t \in [0, \infty)$, a stochastic process has continuous trajectories if its restriction to $t \in [0, N]$ has continuous trajectories for all $N \in \mathbb{N}$.

Conditions (11.2)–(11.4) imply that the Wiener process has independent increments, ie. $W_0, W_t - W_0, W_{t+s} - W_t, \dots$ are independent.

Series expansions for the Wiener process are available in the literature. One way to obtain these is from Fourier expansion for the covariance function.

Problem 11.9 *Let $u(x,t) = Ef(W_t + x)$, where f is a smooth function. Show that u satisfies the parabolic equation $\frac{\partial u}{\partial t} = \frac{1}{2}\frac{\partial^2 u}{\partial x^2}$.*

Problem 11.10 *What partial differential equation is solved by $u(x,t) = Ef(aW_t + x + bt)$ when a, b are non-zero constants?*

Scaled Wiener process is a good model of diffusion. Use the two-dimensional Wiener process to model a two-dimensional diffusion. The two-dimensional Wiener process is obtained from two independent one-dimensional components. The diffusion coefficient a^2 has units [*length*^{2}*/time*] and is implemented by scaling the Wiener process: aW_t has diffusion coefficient a^2.

Exercise 11.11 *An eye-irritant pollutant is emitted from a factory chimney located at $x = 0, y = 10$ on the xy plane, and the wind blows left-to-right with velocity $v(y) = y$ at height y. At a distance $L = 100$ down-wind, there is a residential building of height 15. Which floor of the building is polluted the most? Is there a significant difference in pollution level between the floors? (Assume that the distances are in units such that the diffusion coefficient is 1.)*

[2] This is a very imprecise statement!

Chapter 12

Time Series

This chapter contains additional topics on discrete time processes. We begin with several examples of possible simulations. The resulting random curves are very different, but they have the same "mean-square" behavior.

Example 12.1 *Suppose we wish to pick a curve at random. In other words, we need a random function* $X(t)$ *of integer parameter* t*. Here is one way to do it: take* $X(t) = \sum_k \gamma_k a_k \cos(kt)$*, where* γ_k *are i. i. d. normal* $N(0,1)$*. To ensure the series is convergent we need* $\sum_k a_k^2 < \infty$*; thus we can assume* $a_k = \int_0^{2\pi} g(\theta)\cos(k\theta)\,d\theta$*. The theory of Fourier series tells us that if* $\int g^2(\theta) < \infty$*, then the coefficients* a_k *are square-summable.*

Example 12.2 *Suppose we wish to pick a curve at random. Here is one way to do it: take* $X(t) = \sum c_j \cos(2\pi t + \Phi_j)$*, where* Φ_j *are independent and uniformly distributed on the interval* $(0, 2\pi)$*. Again select* $c_k = \int_0^{2\pi} g(\theta)\cos(k\theta)\,d\theta$*.*

Exercise 12.1 *Write simulations of the curves as described. Pick as* g *a trigonometric polynomial, say* $g(\theta) = 1 + \cos(\theta)\cos(2\theta)$*.*

12.1 Second order stationary processes

Stationary processes are those that their probabilistic characteristics (distributions, conditional distributions, moments, covariances) do not change with time.

Example 12.3 *Suppose* X_n *is a Markov chain, and* $\Pr(X_0 = j) = \pi(j)$*, where* π_j *is its invariant distribution. Then* $(X_0, X_k) \cong (X_n, X_{n+k})$*.*

For second-order stationary processes only means, variances, and covariances do not change with time. That is, $m(t) = EX(t) = const, cov(X(t), X(s)) = K(t-s)$. The second-order theory of processes is a very coarse theory. Nevertheless, it does solve the best linear prediction problem.

Proposition 12.1.1 *In each of the introductory Examples,* $EX(t) = 0$ *and* $cov(X(t), X(s)) = K(t-s)$*, where* $K(t) = \int_0^{2\pi} \cos(\theta t) f(\theta)\,d\theta$*.*

12.1.1 Positive definite functions

The covariance $K(t)$ of the weakly stationary process is a *positive definite* function. That is, $\sum c_i c_j K(t_i - t_j) = E|\sum_j c_j X(t_j)|^2 \geq 0$ for all $c_j \in \mathbb{C}, t_j \in \mathbb{R}$. In addition, $K(t) = K(-t)$.

Theorem 12.1.2 *Given a positive definite even function $K : \mathbb{Z} \to \mathbb{R}$, there is a $(0, 2\pi)$-valued random variable Θ such that for all integer t*

$$K(t) = K(0)E\cos(t\Theta). \tag{12.1}$$

Proposition 12.1.3 *Suppose Θ from Theorem 12.1.2 has density $f(\theta)$. Let $g(\theta) = \sqrt{f(\theta)}$ and define X_t as in Example 12.1. Then X_t is Gaussian, mean zero, with covariance (12.1).*

12.2 Trajectory averages

When a time series X_t is observed, $\frac{1}{n}(X_1 + \ldots + X_n)$ is the "trajectory average". It is interesting how does this compare to the "probabilistic" average EX.

The following theorem follows immediately from the proof of Theorem 5.2.1. The assumption holds true in particular when X_n has spectral density.

Theorem 12.2.1 *Suppose X_n is weakly stationary. If $cov(X_0, X_n) \to 0$ as $n \to \infty$ then $\frac{1}{n}(X_1 + \ldots + X_n) \to EX$ in L_2-norm.*

Proof. Compute the variance, and use the Cesaro summability result (Theorem B.3.1). □

The covariance argument can be rewritten in spectral notation. Suppose $EX_0X_k = (2\pi)^{-1}\int_{-\pi}^{\pi} e^{iks} f(s)\,ds$. Then $E(X_1 + \ldots + X_n)^2 = (2\pi)^{-1}\int_{-\pi}^{\pi} |\sum_{k=1}^{n} e^{iks}|^2 f(s)\,ds$, so $Var(\frac{1}{n}(X_1 + \ldots + X_n)) = (2\pi)^{-1}\frac{1}{n^2}\int_{-\pi}^{\pi} \frac{\sin^2(\frac{1}{2}ns)}{\sin^2(\frac{1}{2}s)} f(s)\,ds \to 0$ as $n \to \infty$.

12.3 The general prediction problem

The basic problem in filtering and prediction is as follows. Given variables $X_1, \ldots, X_n$, find the estimator of the value of Y with smallest quadratic error. Case $n = 1$ is presented is Sections 2.11.1 and 2.14 for the linear and non-linear case.

In the linear prediction problem we deal with linear estimators $a_0 + \sum_j a_j X_j$. The quadratic error involves variances, covariances and averages only. Thus it is appropriate to handle this in through the second order processes, and the solution should depend on the density of Θ - the so called spectral density only.

The general Hilbert space theory tells us that the best linear prediction of X_{t+1} based on the past is $\sum_{j=0}^{t} a_j X_j$ with coefficients a_j such that $EX_j(X_{t+1} - \sum_{j=0}^{t} a_j X_j) = 0$ for all $0 leq j \leq t$. This is a system of $t+1$ linear equations for $t+1$ unknown coefficients. Gramm-Schmidt orthogonalization allows to replace X_j by orthonormal ξ_j. Optimal prediction uses $\sum_{j=0}^{t} \alpha_j \xi_j$ with $\alpha_j = EX_0\xi_{t-j}$.

12.4 Autoregressive processes

Suppose ξ_k are i. i. d. A stochastic process X_t is an autoregressive process, if it satisfies a linear difference equation

$$X_{t+1} = \sum_{j=0}^{d} a_j X_{t-j} + \xi_{t+1} \tag{12.2}$$

with random coefficients ξ_{t+1}.

Example 12.4 *Autoregressive process $X_{t+1} = aX_t + \xi_{t+1}$ is Markov. For $|a| < 1$ it has a stationary distribution. What is it? What happens for $|a| > 1$?*

Example 12.5 *A moving average $X_{t+1} = \frac{1}{d}\sum_{j=1}^{d} \xi_{t-j}$ is an autoregressive process. What is the corresponding difference equation (12.2)?*

For autoregressive process the optimal one-step prediction of X_{t+1} is

$$\sum_{j=0}^{d} \frac{a_j}{a_0} X_{t-j}. \tag{12.3}$$

The spectral theory asserts that this is best linear prediction. However if ξ_j are i. i. d., then this is actually optimal non-linear prediction as well.

More general autoregressive sequences are defined as solutions of

$$\sum_{j=0}^{d} a_j X_{t-j} = \sum_{i} b_i \xi_{t-i} \tag{12.4}$$

These generalize simultaneously autoregressive and moving average processes.

Chapter 13

Additional topics

13.1 A simple probabilistic modeling in Genetics

First we describe the population at a single instance. We consider the model in which each hereditary character is carried by a pair of genes. For simplicity, we assume only one (pair) of a gene, and only one hereditary character corresponding to one *locus*. There are several possible alleles (categories) in a locus – we assume there are only two, denoted by a, A, of which A is dominant. Each individual thus is described by one of the pairs AA, Aa, aa, the so called *phenotype* . However, a is obstructed from the view by A, thus for an outside observer individuals AA and Aa are indistinguishable. A statistical study of such a population can only yield the proportion P_A of individuals with "A-feature", and not the actual probabilities of the three possible phenotype.

Now we turn to the modelling of the generation change. Under simple assumptions we shall be able to find out what are the frequencies of phenotype and genotypes. Usually this information is not directly available. We assume that the next generation occurs by *random mating*. Let $p_{AA}(0), p_{Aa}(0), p_{aa}(0)$ be the actual (and as yet unknown) probabilities of the phenotype. Under random mating with independent selection of parents, the probability that an offspring has phenotype AA is $p_{AA}(1) = (p_{AA}(0) + \frac{1}{2}p_{Aa}(0))^2$.

Denoting by $p_A(t) = p_{AA}(t) + \frac{1}{2}p_{Aa}(t)$ we get the Hardy-Weinberg equilibrium: after one generation the proportions $p_A(t)$ of genotypes stabilize and the phenotype frequencies become

$$P_{AA} = p_A^2 \tag{13.1}$$
$$P_{Aa} = 2p_a p_A \tag{13.2}$$
$$P_{aa} = p_a^2 \tag{13.3}$$
$$\tag{13.4}$$

This determines the actual proportions of the phenotype from the proportion of observed A-carriers and a-carriers.

Problem 13.1 *Show that* $P_{Aa} = 2(\sqrt{P_{aa}} - P_{aa})$.

The next question is to study the effects of the selection, where, say phenotype aa has different chance of survival. This leads to total probability formula and Markov chains.

13.2 Application: verifying matrix multiplication

Suppose one has an algorithm to multiply large matrices and we want to check if the output is correct. A possible method is to pick the vector $\mathbf{X}$ of $0, 1$ and check if $AB\mathbf{X} = C\mathbf{X}$. This is the so called *Freivalds technique*

Theorem 13.2.1 *If A, B, C are $n \times n$ matrices such that $AB \neq C$ then* $\Pr(AB\mathbf{X} = C\mathbf{X}) \leq \frac{1}{2}$.

Proof. For a non-zero $\mathbf{v}$ we have $\Pr(|\mathbf{v} \cdot \mathbf{X}| > 0) < \frac{1}{2}$ □

13.3 Exchangeability

Definition 13.3.1 *A sequence (X_k) of random variables is* exchangeable*, if the joint distribution of $X_{\sigma(1)}, X_{\sigma(2)}, \dots, X_{\sigma(n)}$ is the same as the joint distribution of $X_1, X_2, \dots, X_n$ for all $n \geq 1$ and for all permutations σ of $\{1, 2, \dots, n\}$.*

The following beautiful theorem due to B. de Finetti points out the role of exchangeability as a substitute for independence.

Theorem 13.3.1 *Suppose that $X_1, X_2, \dots$ is an infinite exchangeable sequence. Then there exist a σ-field $\mathcal{N}$ such that $X_1, X_2, \dots$ are $\mathcal{N}$-conditionally i. i. d., that is*

$$P(X_1 < a_1, X_2 < a_2, \dots, X_n < a_n|\mathcal{N})$$

$$= P(X_1 < a_1|\mathcal{N})P(X_1 < a_2|\mathcal{N}) \dots P(X_1 < a_n|\mathcal{N})$$

for all $a_1, \dots, a_n \in \mathbb{R}$ and all $n \geq 1$.

We will use the following (weak) version of the martingale[1] convergence theorem.

Theorem 13.3.2 *Suppose $\mathcal{F}_n$ is a decreasing family of σ-fields, ie. $\mathcal{F}_{n+1} \subset \mathcal{F}_n$ for all $n \geq 1$. If X is integrable, then $E\{X|\mathcal{F}_n\} \to E\{X|\mathcal{F}\}$ in L_1-norm, where $\mathcal{F}$ is the intersection of all $\mathcal{F}_n$.*

Proof. Suppose first that X is square integrable. Subtracting $m = EX$ if necessary, we can reduce the convergence question to the centered case $EX = 0$. Denote $X_n = E\{X|\mathcal{F}_n\}$. Since $\mathcal{F}_{n+1} \subset \mathcal{F}_n$, by Jensen's inequality $EX_n^2 \geq 0$ is a decreasing non-negative sequence. In particular, EX_n^2 converges.

Let $m < n$ be fixed. Then $E(X_n - X_m)^2 = EX_n^2 + EX_m^2 - 2EX_nX_m$. Since $\mathcal{F}_n \subset \mathcal{F}_m$, by Theorem A.2.1 we have

$$EX_nX_m = EE\{X_nX_m|\mathcal{F}_n\} = EX_nE\{X_m|\mathcal{F}_n\}$$

[1] A martingale with respect to a family of increasing σ-fields $\mathcal{F}_n$ is and integrable sequence X_n such that $E(X_{n+1}|\mathcal{F}_n) = X_n$. The sequence $X_n = E(X|\mathcal{F}_n)$ is a martingale. The sequence in the theorem is of the same form, except that the σ-fields are decreasing rather than increasing.

$$= EX_nE\{E\{X|\mathcal{F}_m\}|\mathcal{F}_n\} = EX_nE\{X|\mathcal{F}_n\} = EX_n^2.$$

Therefore $E(X_n - X_m)^2 = EX_m^2 - EX_n^2$. Since EX_n^2 converges, X_n satisfies the Cauchy condition for convergence in L_2 norm. This shows that for square integrable X, sequence $\{X_n\}$ converges in L_2.

If X is not square integrable, then for every $\epsilon > 0$ there is a square integrable Y such that $E|X - Y| < \epsilon$. By Jensen's inequality $E\{X|\mathcal{F}_n\}$ and $E\{Y|\mathcal{F}_n\}$ differ by at most ϵ in L_1-norm; this holds uniformly in n. Since by the first part of the proof $E\{Y|\mathcal{F}_n\}$ is convergent, it satisfies the Cauchy condition in L_2 and hence in L_1. Therefore for each $\epsilon > 0$ we can find N such that for all $n, m > N$ we have $E\{|E\{X|\mathcal{F}_n\} - E\{X|\mathcal{F}_m\}|\} < 3\epsilon$. This shows that $E\{X|\mathcal{F}_n\}$ satisfies the Cauchy condition and hence converges in L_1.

The fact that the limit is $X_\infty = E\{X|\mathcal{F}\}$ can be seen as follows. Clearly X_∞ is $\mathcal{F}_n$-measurable for all n, ie. it is $\mathcal{F}$-measurable. For $A \in \mathcal{F}$ (hence also in $\mathcal{F}_n$), we have $EXI_A = EX_nI_A$. Since $|EX_nI_A - EX_\infty I_A| \leq E|X_n - X_\infty|I_A \leq E|X_n - X_\infty| \to 0$, therefore $EX_nI_A \to EX_\infty I_A$. This shows that $EXI_A = EX_\infty I_A$ and by definition, $X_\infty = E\{X|\mathcal{F}\}$. □

Proof of Theorem 13.3.1. Let $\mathcal{N}$ be the tail σ-field, ie.

$$\mathcal{N} = \bigcap_{k=1}^{\infty} \sigma(X_k, X_{k+1}, \ldots)$$

and put $\mathcal{N}_k = \sigma(X_k, X_{k+1}, \ldots)$. Fix bounded measurable functions f, g, h and denote

$$F_n = f(X_1, \ldots, X_n);$$

$$G_{n,m} = g(X_{n+1}, \ldots, X_{m+n});$$

$$H_{n,m,N} = h(X_{m+n+N+1}, X_{m+n+N+2}, \ldots),$$

where $n, m, N \geq 1$. Exchangeability implies that

$$EF_nG_{n,m}H_{n,m,N} = EF_nG_{n+r,m}H_{n,m,N}$$

for all $r \leq N$. Since $H_{n,m,N}$ is an arbitrary bounded $\mathcal{N}_{m+n+N+1}$-measurable function, this implies

$$E\{F_nG_{n,m}|\mathcal{N}_{m+n+N+1}\} = E\{F_nG_{n+r,m}|\mathcal{N}_{m+n+N+1}\}.$$

Passing to the limit as $N \to \infty$, see Theorem 13.3.2, this gives

$$E\{F_nG_{n,m}|\mathcal{N}\} = E\{F_nG_{n+r,m}|\mathcal{N}\}.$$

Therefore

$$E\{F_nG_{n,m}|\mathcal{N}\} = E\{G_{n+r,m}E\{F_n|\mathcal{N}_{n+r+1}\}|\mathcal{N}\}.$$

Since $E\{F_n|\mathcal{N}_{n+r+1}\}$ converges in L_1 to $E\{F_n|\mathcal{N}\}$ as $r \to \infty$, and since g is bounded,

$$E\{G_{n+r,m}E\{F_n|\mathcal{N}_{n+r+1}\}|\mathcal{N}\}$$

is arbitrarily close (in the L_1 norm) to

$$E\{G_{n+r,m}E\{F_n|\mathcal{N}\}|\mathcal{N}\} = E\{F_n|\mathcal{N}\}E\{G_{n+r,m}|\mathcal{N}\}$$

as $r \to \infty$. By exchangeability $E\{G_{n+r,m}|\mathcal{N}\} = E\{G_{n,m}|\mathcal{N}\}$ almost surely, which proves that

$$E\{F_n G_{n,m}|\mathcal{N}\} = E\{F_n|\mathcal{N}\}E\{G_{n,m}|\mathcal{N}\}.$$

Since f, g are arbitrary, this proves $\mathcal{N}$-conditional independence of the sequence. Using the exchangeability of the sequence once again, one can see that random variables $X_1, X_2, \dots$ have the same $\mathcal{N}$-conditional distribution and thus the theorem is proved. □

13.4 Distances between strings

A string is a sequence of *letters*, or symbols from the finite alphabet. For the purpose of computer modelling, we can assume that a string is a sequence of natural numbers $\{1, \dots d\}$, parameter d being the size of the alphabet.

Three simple examples of strings are the words, sentences in, say, English, and DNA molecules. Here $d = 26$ (for lower-case words), $d = 94$ for sentences, and $d = 6$ for the DNA (there are only four proteins, but extra symbols are used to mark various undecided cases).

The question of comparing two strings for similarities arises in molecular biology and in designing a spell-checker, or a speech recognition system. Accordingly, one would like to say which strings are similar, and how likely it is that they are similar due to chance only. Additional complications arise from the fact that two strings compared do not necessarily have the same length.

A simple way to compare two strings is to measure the number of symbols that don't match (the hamming distance). For instance `abbacd` and `babacd` would then have distance 2. But `abbacd` and `bbacda` would have distance 6, even though they differ just by one transposition.

A less obvious way to compare two strings is to measure the *edit distance*: how many *editing* operations are needed to transform one of the strings into another. Usually the editing operations are:

- insert a symbol
- delete a symbol
- replace a symbol
- transpose two consecutive symbols

These are suitable for spell-checkers, where it is known that about 80% of typing errors are of the above form, thus most of mistyped words have edit-distance 1 from the original.

Accordingly, the edit distance is set to 0, if the words are identical, 1 if they differ by a single error of one of the listed types, 2 if there were two such errors, etc. Formally, it is defined as the smallest number of elementary "TE" transformations required to transform one of the words into another.

The method of computation is based on recurrence. It is easy to see that a number of transformations between and empty word and another one is exactly the length[2] of the

[2] Only deletions are required.

word. If the two words U, W are formed from shorter ones $U0, W0$ by adding letters at the end, then the distance $Dist(U, W)$ is the smallest of the numbers

- $Dist(U0, W) + 1$ (delete)
- $Dist(U, W0) + 1$ (add)
- $Dist(U0, W0) + 0$ or 1 if same letter is added different letter (swap)
- $Dist(U0', V0') + 1$, if the last two letters are identical (transpose), where $U0', V0'$ are $U0, W0$ with last letter removed.

Here is a complete VB-listing:

```
Function Dist (U$, V$) As Integer
'Returns the edit distance
'(number of elementary changes: replacement, deletion, insertion,transposition)
'that are required to transform word U$ into V$
'
'Declare auxiliary variables
Dim m As Integer, n As Integer, j As Integer, i  As Integer
Dim x As Integer, y As Integer, z As Integer, A$, B$
If Len(U$) < Len(V$) Then A$ = U$: B$ = V$:  Else A$ = V$: B$ = U$
m = Len(A$)
n = Len(B$)
'Declare matrix of distances between substrings of i,j characters
ReDim D(m, n) As Integer
'Assign boundary values: distances from empty string
 For i = 0 To m: D(i, 0) = i: Next i
 For j = 1 To n: D(0, j) = j: Next j
'
'Main recurrence: Compute next distance D(i,j) from previously found values
  For i = 1 To m
        For j = 1 To n
          x = D(i - 1, j) + 1   'delete character
          y = D(i, j - 1) + 1   'inserte character
          x = Intmin(x, y)         'choose better (Integer Minimum)
          y = D(i - 1, j - 1) - (Mid$(A$, i, 1) <> Mid$(B$, j, 1)) 'swap characters i
          x = Intmin(x, y)        ' choose better
          z = 0
          If i > 1 And j > 1 Then
              'If Mid$(A$, i, 1) <> Mid$(B$, j, 1) Then
              z = (Mid$(A$, i, 1) = Mid$(B$, j - 1, 1)) * (Mid$(A$, i - 1, 1) = Mid$(
              'End If
          y = (1 + D(i - 2, j - 2)) * z + x * (1 - z)
          x = Intmin(x, y)
          End If
          D(i, j) = x
```

```
    Next j
  Next i
Dist = D(m, n)'current value
End Function
```

The main problem with edit distance to analyze DNA molecules is processing time.

Exercise 13.2 *Write a program computing the edit distance between strings, and another one, which does the editing by simulation. (Try the randomization based on random number of edit operations from the currently best candidate)*

13.5 A model of cell growth

A cell in its growth goes through several phases, which have different probabilistic characteristics. In a simple model, the cell doubles after a random time, which is the sum of the exponential and deterministic portion. The average of the exponential phase can be assumed to depend on the external circumstances.

Questions of interest

How does the growth of cells affect other cells? How to control mixed populations of cells to stay within prescribed limits?

13.6 Shannon's Entropy

Let $X_1, \dots X_n$ be independent identically distributed (i. i. d.) discrete r. v with k values, say $\{v_1, \dots, v_k\}$. Put $\mathbf{X} = (X_1, \dots, X_k)$.

For a fixed vector $\mathbf{y}$ we have thus the joint probability mass function $f(\mathbf{x}) = \Pr(\mathbf{X} = \mathbf{x})$. The average information $H(\mathbf{X})$ contained in $\mathbf{X}$ is defined as

$$H(\mathbf{X}) = -E \log f(\mathbf{X}) = -\sum_{\mathbf{x}} f(\mathbf{x}) \log f(\mathbf{x}) \tag{13.5}$$

Problem 13.3 *Prove Gibbs' inequality $\sum_j p_j \log p_j \geq \sum p_j \log q_j$ for all $q_j > 0, \sum q_j = 1$.*

Notice that $H(\mathbf{X}) \geq 0$ and $H(\mathbf{X}) \leq \log k$

13.6.1 Optimal search

Coding

Huffman's code

Problem 13.4 *Suppose you have 15 identical in appearance coins, except that one of them has a different weight. Find the optimal[3] weighting strategy to identify the odd coin by using a scale.*

[3] That is, find the strategy that costs the least if you have to pay for each use of the scale.

13.7 Application: spread of epidemic

Modeling of the spread of disease is complicated due to multiple factors that influence its development. The birth-and-death process does not seem to be a good model for the spread of an epidemic in a finite population, since when a large proportion of the population has been infected, we cannot suppose that the rate of infections is independent of past history.

Appendix A

Theoretical complements

A.1 L_p-spaces

Inequalities related to expected values are best stated in geometric language of norms and normed spaces. We say that $X \in L_p$, if X is p-integrable, i.e. $E|X|^p < \infty$. In particular, X is *square integrable* if $EX^2 < \infty$.

The L_p norm is

$$\|X\|_p = \begin{cases} \sqrt[p]{E|X|^p} & \text{if } p \geq 1; \\ \text{ess sup}|X| & \text{if } p = \infty. \end{cases}$$

Notice that $\|X - EX\|_2$ is just the standard deviation.

We say that X_n converges to X in L_p, if $\|X_n - X\|_p \to 0$ as $n \to \infty$. If X_n converges to X in L_2, we shall also use the phrase *sequence X_n converges to X in mean-square.* An example of the latter is Theorem 5.2.1.

Several useful inequalities are collected in the following.

Theorem A.1.1 *(i) for $1 \leq p \leq q \leq \infty$ we have Minkowski's inequality*

$$\|X\|_p \leq \|X\|_q. \tag{A.1}$$

(ii) for $1/p + 1/q = 1$, $p \geq 1$ we have Hölder's inequality

$$EXY \leq \|X\|_p \|Y\|_q. \tag{A.2}$$

(iii) for $1 \leq p \leq \infty$ we have triangle inequality

$$\|X + Y\|_p \leq \|X\|_p + \|Y\|_p. \tag{A.3}$$

Special case $p = q = 2$ of Hölder's inequality (A.2) reads $EXY \leq \sqrt{EX^2EY^2}$. It is frequently used and is known as the *Cauchy-Schwarz inequality.*

For the proof of Theorem A.1.1 we need the following elementary inequality.

Lemma A.1.2 *For $a, b > 0, 1 < p < \infty$ and $1/p + 1/q = 1$ we have*

$$ab \leq a^p/p + b^q/q. \tag{A.4}$$

Proof. Function $t \mapsto t^p/p + t^{-q}/q$ has the derivative $t^{p-1} - t^{-q-1}$. The derivative is positive for $t > 1$ and negative for $0 < t < 1$. Hence the maximum value of the function for $t > 0$ is attained at $t = 1$, giving

$$t^p/p + t^{-q}/q \geq 1.$$

Substituting $t = a^{1/q}b^{-1/p}$ we get (A.4). □

Proof of Theorem A.1.1 (ii). If either $\|X\|_p = 0$ or $\|Y\|_q = 0$, then $XY = 0$ a. s. Therefore we consider only the case $\|X\|_p\|Y\|_q > 0$ and after rescaling we assume $\|X\|_p = \|Y\|_q = 1$. Furthermore, the case $p = 1, q = \infty$ is trivial as $|XY| \leq |X|\|Y\|_\infty$. For $1 < p < \infty$ by (A.4) we have

$$|XY| \leq |X|^p/p + |Y|^q/q.$$

Integrating this inequality we get $|EXY| \leq E|XY| \leq 1 = \|X\|_p\|Y\|_q$. □

Proof of Theorem A.1.1 (i). For $p = 1$ this is just Jensen's inequality; for a more general version see Theorem A.2.1. For $1 < p < \infty$ by Hölder's inequality applied to the product of 1 and $|X|^p$ we have

$$\|X\|_p^p = E\{|X|^p \cdot 1\} \leq (E|X|^q)^{p/q}(E1^r)^{1/r} = \|X\|_q^p,$$

where r is computed from the equation $1/r + p/q = 1$. (This proof works also for $p = 1$ with obvious changes in the write-up.) □

Proof of Theorem A.1.1 (iii). The inequality is trivial if $p = 1$ or if $\|X + Y\|_p = 0$. In the remaining cases

$$\|X + Y\|_p^p \leq E\{(|X| + |Y|)|X + Y|^{p-1}\} = E\{|X||X + Y|^{p-1}\} + E\{|Y||X + Y|^{p-1}\}.$$

By Hölder's inequality

$$\|X + Y\|_p^p \leq \|X\|_p\|X + Y\|_p^{p/q} + \|Y\|_p\|X + Y\|_p^{p/q}.$$

Since $p/q = p - 1$, dividing both sides by $\|X + Y\|_p^{p/q}$ we get the conclusion. □

A.2 Properties of conditional expectations

In more advanced courses conditional expectation $E(X|Y)$ is defined as a random variable $\phi(Y)$ that satisfies $EXf(Y) = E\phi(Y)f(Y)$ for all bounded measurable (or just continuous) functions f.

The next theorem lists useful properties of conditional expectations.

Theorem A.2.1 *(i) If $Y = f(Z)$ is such that X and XY are integrable, then $E\{XY|Z\} = YE\{X|Z\}$;*

(ii) $E^X E^{X,Y} = E^X$;

(iii) *If* (X, Y) *and* Z *are independent, then* $E\{X|Y, Z\} = E\{X|Y\}$;

(iv) *If* $g(x)$ *is a convex function and* $E|g(X)| < \infty$, *then* $g(E\{X|Y\}) \leq E\{g(X)|Y\}$;

(v) *If* Y *is non-random, then* $E\{X|Y\} = EX$;

(vi) *If* X, Y *are integrable and* $a, b \in \mathbb{R}$ *then* $E\{aX + bY|Z\} = aE\{X|Z\} + bE\{Y|Z\}$;

(vii) *If* X *and* Y *are independent, then* $E\{X|Y\} = EX$.

Remark 4 Inequality (iv) is known as Jensen's inequality and this is how we shall refer to it.

The abstract proof uses the following[1].

Lemma A.2.2 *If* Y_1 *and* Y_2 *are* $\mathcal{F}$*-measurable and* $\int_A Y_1\,dP \leq \int_A Y_2\,dP$ *for all* $A \in \mathcal{F}$, *then* $Y_1 \leq Y_2$ *almost surely. If* $\int_A Y_1\,dP = \int_A Y_2\,dP$ *for all* $A \in \mathcal{F}$, *then* $Y_1 = Y_2$.

Proof. Let $A_\epsilon = \{Y_1 > Y_2 + \epsilon\} \in \mathcal{F}$. Since $\int_{A_\epsilon} Y_1\,dP \geq \int_{A_\epsilon} Y_2\,dP + \epsilon P(A_\epsilon)$, thus $P(A_\epsilon) > 0$ is impossible. Event $\{Y_1 > Y_2\}$ is the countable union of the events A_ϵ (with ϵ rational); thus it has probability 0 and $Y_1 \leq Y_2$ with probability one.

The second part follows from the first by symmetry. □

Proof of Theorem A.2.1.

(i) This is verified first for $Y = I_B$ (the indicator function of an event $B \in \mathcal{F}$). Let $Y_1 = E\{XY|\mathcal{F}\}, Y_2 = YE\{X|\mathcal{F}\}$. From the definition one can easily see that both $\int_A Y_1\,dP$ and $\int_A Y_2\,dP$ are equal to $\int_{A\cap B} X\,dP$. Therefore $Y_1 = Y_2$ by the Lemma A.2.2.

For the general case, approximate Y by simple random variables and use (vi).

(ii) This follows from Lemma A.2.2: random variables $Y_1 = E\{X|\mathcal{F}\}$, $Y_2 = E\{X|\mathcal{G}\}$ are $\mathcal{G}$-measurable and for A in $\mathcal{G}$ both $\int_A Y_1\,dP$ and $\int_A Y_2\,dP$ are equal to $\int_A X\,dP$.

(iii) Let $Y_1 = E\{X|\mathcal{N} \bigvee \mathcal{F}\}, Y_2 = E\{X|\mathcal{F}\}$. We check first that

$$\int_A Y_1\,dP = \int_A Y_2\,dP$$

for all $A = B \cap C$, where $B \in \mathcal{N}$ and $C \in \mathcal{F}$. This holds true, as both sides of the equation are equal to $P(B) \int_C X\,dP$. Once equality $\int_A Y_1\,dP = \int_A Y_2\,dP$ is established for the generators of the σ-field, it holds true for the whole σ-field $\mathcal{N} \bigvee \mathcal{F}$; this is standard measure theory[2].

(iv) Here we need the first part of Lemma A.2.2. We also need to know that each convex function $g(x)$ can be written as the supremum of a family of affine functions $f_{a,b}(x) = ax + b$. Let $Y_1 = E\{g(X)|\mathcal{F}\}, Y_2 = f_{a,b}(E\{X|\mathcal{F}\}), A \in \mathcal{F}$. By (vi) we have

$$\int_A Y_1\,dP = \int_A g(X)\,dP \geq f_{a,b}(\int_A X)\,dP = f_{a,b}(\int_A E\{X|\mathcal{F}\})\,dP = \int_A Y_2\,dP.$$

Hence $f_{a,b}(E\{X|\mathcal{F}\}) \leq E\{g(X)|\mathcal{F}\}$; taking the supremum (over suitable a, b) ends the proof.

(v), (vi), (vii) These proofs are left as exercises. □

[1] Readers not familiar with measure theory should skip the proofs.

[2] See $\pi - \lambda$ Theorem (Theorem 3.3) P. Billingsley, *Probability and measure*, Wiley, New York 1986.

Problem A.1 *Prove part (v) of Theorem A.2.1.*

Problem A.2 *Prove part (vi) of Theorem A.2.1.*

Problem A.3 *Prove part (vii) of Theorem A.2.1.*

Problem A.4 *Prove the following conditional version of Chebyshev's inequality: if* $E|X| < \infty$*, then*

$$P(|X| > t|Y) \leq E\{|X|\ |Y\}/t$$

almost surely.

Problem A.5 *Show that if* (U, V, X) *are such that in distribution* $(U, X) \cong (V, X)$ *then* $E\{U|X\} = E\{V|X\}$ *almost surely.*

Problem A.6 *Show that if* X, Y *are integrable non-degenerate random variables, such that*

$$E\{X|Y\} = aY,\ E\{Y|X\} = bX,$$

then $|ab| \leq 1$.

Problem A.7 *Show that if* X, Y *are integrable such that* $E\{X|Y\} = Y$ *and* $E\{Y|X\} = X$*, then* $X = Y$ *a. s.*

Appendix B

Math background

The following sections are short reference on material from general math (calculus, linear algebra, etc).

B.1 Interactive links

The following links are operational as of March 21, 1996. Please note that these may change at any time.

- real analysis[1] is available online.

B.2 Elementary combinatorics

The art of counting is called *combinatorics.* Here is a short listing of the formulas. All are the consequences of the product rule of counting.

Permutations:
Permutation is an arrangement (ordering) of k out of n distinct objects without repetitions. The number of permutations is $\frac{n!}{(n-k)!}$. In particular, the number of ways to order n objects is $n!$

Combinations: Combinations are k-element subsets of n distinct elements. The number of combinations is $\binom{n}{k}$.

Variations: Variations are arrangements of k out of n distinct objects with repetitions allowed. The number of variations is n^k

B.3 Limits

The following limit can be computed by L'Hospital rule.

$$\lim_{n\to\infty}(1+a/n)^n = e^a \tag{B.1}$$

The Cesaro summability formula is

Theorem B.3.1 *If $a_n \to a$ then $\frac{1}{n}(a_1 + \ldots + a_n) \to a$.*

[1] http://www.shu.edu/projects/reals/reals.html

B.4 Power series expansions

The following power series expansions are of interest in this course.

$$e^x = \sum_{k=0}^{\infty} x^k/k! \tag{B.2}$$

$$\frac{1}{1-x} = \sum_{k=0}^{\infty} x^k \tag{B.3}$$

They give immediately the expansions

$$\ln 1+x = \sum_{k=0}^{\infty} (-1)^k \frac{x^{k+1}}{k+1} \tag{B.4}$$

$$e^{x^2/2} = \sum_{k=0}^{\infty} \tag{B.5}$$

$$\int_0^x e^{-t^2/2}\, dt = \sum_{k=0}^{\infty} \tag{B.6}$$

$$\tag{B.7}$$

In particular for $x > 0$ we have $\ln 1+x < x$ and $\ln 1+x > x - x^2/2$.

B.5 Multivariate integration

Suppose $x = x(u,v), y = y(u,v)$. The change of variables formula in multivariate case is

$$\int\int_U f(x,y)\, dxdy = \int\int_V f(x(u,v), y(u,v))|J(u,v)|\, dudv, \tag{B.8}$$

where the Jacobian J is given by

$$J = \det \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{bmatrix} \tag{B.9}$$

B.6 Differential equations

The solution of the linear differential equation $y' + ay = g(x)$ with $y(0) = y_0$ is given by $y(x) = y_0 e^{-ax} \int_0^x e^{at} g(t)/, dt$.

Second order linear equation $y'' + ay' + by = g(x)$ is often solved by the method of varying a constant. The first step is to solve the homogenous equation $y'' + ay' + by = 0$ first using $y = e^{rx}$. The general solution of homogenous equation is $y = C_1 e^{r_1 x} + C_2 e^{r_2 x}$, or $y = C_1 e^{rx} + C_2 x e^{rx}$ if the characteristic equation has double root.

B.7 Linear algebra

Definition B.7.1 *A scalar product of vectors in* V *is a bilinear, positive definite, non-degenerate mapping* $\langle\cdot|\cdot\rangle : \mathsf{V} \times \mathsf{V} \to \mathbb{R}$.

Definition B.7.2 *Vectors* $\mathbf{x}, \mathbf{y}$ *are orthogonal with respect to scalar product* $\langle|\rangle$, *if* $\langle \mathbf{x}|\mathbf{y}\rangle = 0$.

The length of a vector is $||\mathbf{x}|| = \sqrt{\langle \mathbf{x}|\mathbf{x}\rangle}$. Orthogonal vectors of unit length are called *orthonormal.*. If $\mathbf{e}_j$ are the orthonormal vectors and $\mathbf{x}$ is in their linear span, then the coefficients in the expansion $\mathbf{x} = \sum_j a_j \mathbf{e}_j$ are given by $a_j = \langle \mathbf{x}|\mathbf{e}_j\rangle$.

Example B.1 *Let* V *be the vector space of all continuous functions on the interval* $[-\pi, \pi]$. *In the scalar product* $\langle f|g\rangle = \int_{-\pi}^{\pi} f(t)g(t)\, dt$ *the functions* $\{\sin k\}_{k\in\mathbb{N}}, \{\cos kt\}_{k\in\mathbb{N}}, 1$ *are orthogonal.*

B.8 Fourier series

The Fourier series for a function $f(x)$ is the expansion $f(x) = \sum_n a_n \sin \pi n x + b_n \cos \pi n x$. Every periodic continuous function f can be expanded in the Fourier series. The coefficients are

$$b_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x)\, dx \tag{B.10}$$

$$b_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) \cos n\pi x\, dx \text{ for } n \neq 0 \tag{B.11}$$

$$a_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) \sin n\pi x\, dx \tag{B.12}$$

$$\tag{B.13}$$

Example B.2 *Expand $|x|$ into the Fourier series, and graph the corresponding partial sums.*

B.9 Powers of matrices

The Cayley - Hamilton theorem asserts that each $d \times d$ matrix A satisfies the polynomial equation $Q_d(A) = 0$, where $Q_d(x) = \det(A - xI)$ is the characteristic polynomial of degree d.

This implies that $A^n = a_0(n)I + a_1(n)A + \ldots + a_{d-1}(n)A^{d-1}$, where $x^n = D(x)Q(x) + a_0(n) + a_1(n)x + \ldots + a_{d-1}(n)x^{d-1}$. If A has n distinct eigenvalues λ_j, the coefficients $a_j(n)$ can be found by solving the system of equations $\lambda_j^n = a_0(n) + a_1(n)\lambda_j + \ldots + a_{d-1}(n)\lambda_j^{d-1}$.

A quick method that finds a characteristic polynomial due to K... is to solve the system of linear equations for the coefficients: pick a vector $\mathbf{X}$ at random and solve[2] for $a_0, \ldots, a_{d-1}$ the equations $a_0\mathbf{X} + a_1 A\mathbf{X} + \ldots + a_{d-1}A^{d-1}\mathbf{X} = 0$. If the resulting matrix is singular, re-sample $\mathbf{X}$ until a non-singular matrix is found.

[2] Use Gaussian elimination.

Appendix C

Numerical Methods

Calculators and more sophisticated math-geared software have efficient numerical methods built-in. Here are short prescriptions that may be used by general programmer. A more complete source is e.g.*Numerical Recipes: The Art of Scientific Computing* series.

C.1 Numerical integration

To approximate the integral $\int_a^b f(x)\,dx$ in calculus we use left and right sums: $L_n = \frac{b-a}{n}\sum_{k=0}^{n-1} f(a+\frac{b-a}{n}k)$, $R_n = \frac{b-a}{n}\sum_{k=1}^{n} f(a+\frac{b-a}{n}k)$

The more exact *trapezoidal rule* uses $\int_a^b f(x)\,dx \approx S_n = \frac{1}{2}(L_n + R_n)$.

Still more powerful and easy to program integration method is the *Simpson rule*: $\int_a^b f(x)\,dx \approx \frac{4}{3}S_{2n} - \frac{1}{3}S_n$. The Simpson rule is **exact** for cubic polynomials. Typical way to program it is to call the subroutine performing trapezoid method integration twice. Many calculators have Simpson rule build in. Before you use it, be sure to check if it is reliable enough. A simple test that catches some poorly written routines is $\int_0^{500} e^{-x^2}\,dx \approx \sqrt{\pi/2}$.

C.2 Solving equations

The fast and powerful *bisection method* requires correct end-points, and finds one solution only. But it has virtually no assumptions, except the continuity: If $f(x)$ is continuous and $f(a) < 0, f(b) > 0$ then there is $a < x < b$ such that $f(x) = 0$ and the interval length can be halved by computing $f(\frac{a+b}{2})$.

Three difficulties arise in real applications.

- How to find a "correct" initial pair
- How to find more than one solution
- How to solve systems of equations in several unknowns

The second point has satisfactory answer for polynomial equations. The third point can be tackled through search for minimum. Namely, if equations are $f(x,y) = 0, g(x,y) = 0$ then we need to find minima of $f^2(x,y) + g^2(x,y)$.

C.3 Searching for minimum

The analog of the bisection method for finding minima is to begin with three points $a < b < c$ such that $f(b)$ is the smallest of the three. The next triple is produced by partitioning the larger of the two segments in proportion[1] $\frac{3-\sqrt{5}}{2} \approx 0.38197$, and comparing the resulting four values to pick the narrower triplet.

[1] This is golden section. The explanation why it occurs here can be found in numerical methods books.

Appendix D

Programming Help

Programming help is grouped by the type of software. Currently available (as of March 21, 1996) are preliminary versions of:

- Help for TI85 Programmable calculator[1]

D.1 Introducing BASIC

Any DOS-running PC comes also with BASIC. You can start it with the command

```
    QBASIC
```

from the DOS command line, or from **Windows** *File Manager* (click on an entry `QBASIC.EXE`, usually located in). If you are a devoted Windows user, you may install an *Icon* in *Program Manager* to run BASIC with a click of the mouse.

Correctly written programs halt at the end. But not all programs do that, so an "emergency exit" is build in.

To stop the execution of any program, press simultaneously $\boxed{\boxed{\texttt{Ctrl}}}+\boxed{\boxed{\texttt{Break}}}$**.**

how to use this chapter

This text was written for a complete novice to BASIC. If you fit the description, read the pages below, and **type in** each of the sample programs. Once you have them in the QBASIC, run them to see the effects of various instrunctions.

There is no possibility that by running these programs you will do any harm to your equipment. Experiment, and if something goes realy wrong, you can always turn OFF or restart the computer.

D.2 Computing

The mathematical conventions in QBASIC are `2^(1/2)` for $2^{1/2}$, `SQR(13)` for $\sqrt{13}$, $LOG(2)$ for the natural logarithm $\ln 2$, etc. With these, one can use QBASIC as a calculator. For instance the instruction

```
PRINT LOG(1+2^(1/2))
```

will print the decimal value of the expression $\ln(1+\sqrt{2})$. This probably is the simplest program to begin with. It is so short that there is no point in saving it.

The real power comes from repetitive operations explained in Section D.4.

[1] http://math.uc.edu/b̃rycw/preprint/ti85

D.3 The structure of a program

BASIC programs are actually text files. The instructions are read consecutively by the BASIC interpreter. The QBASIC interpreter that comes with DOS is sufficient for our purposes. Sample programs below introduce certain more exotic build-in functions.

Example D.1 *The following simple program prints current date & time*

```
PRINT "HELLO"
PRINT "Today is "; DATE$
PRINT TIME$
```

Besides the actual code, programs should contain comments with explanations of instructions and their purpose. Comments and explanations in the program can be hidden from the processor through `REM` command. Everything in a line after this command is skipped (the only exception being metacommands, which we don't have to concern ourselves with here). The sample[2] programs, use a shortcut `'` instead of the full command `REM`. This is faster to type and equivalent in action.

Example D.2 *Here is the previous program with comments*

```
'Prints current date & time
PRINT "HELLO" 'greet the user
PRINT "Today is "; DATE$ 'print date
PRINT TIME$ ' print time could have printer TIMER=# seconds since 1980
```

The typical program consists of the main module with fairly general instructions that call subroutines to perform specific jobs, and a number of subprograms that are marked in text by `SUB ... END SUB`, and are displayed separately by the QBASIC Editor. Subprograms make it easier to write, test, and maintain the program.

Within `QBasic` the `SUB`s are separated, each in each own text window, but the final program is saved as one file with `SUB`s following the main module. To create a SUB, choose **New SUB** from the **Edit** menu. More details follow in Section D.9.

Larger projects often use several modules[3] that are compiled and run together.

Example D.3 *Here is the revision of the previous program with a simple but useful SUB.*

```
'Prints current date & time in the middle of the screen
CLS 'clear display
CenterPrint "HELLO" 'greet the user
CenterPrint "Today is " & DATE$ 'print date - string has to be concatenated
CenterPrint TIME$ ' print time could have printer TIMER=# seconds since 1980

SUB CenterPrint (Text$)
'**** print text Text$ centered on screen
l=41-LEN(Text$)/2
if l<0 then l=0 'too long text cannot be centered
print TAB(l); Text$
```

New things: CLS, concatenation of strings, calling SUB, screen positioning by TAB, LEN("hello")

Every subprogram should begin with a (commented) short description of its purpose, and the meaning of parameters.

[2] The actual sample programs on the disk also contain commented LaTeX typesetting instructions at the beginning and at the end. Their sole purpose is to insert the listings into this text.

[3] For example, in many applied math programs there is a need for integration routines. These perhaps would be collected in a separate module to facilitate repeated usage.

D.4 Conditionals and loops

Loops of fixed size are best handled by

```
FOR j=1 to 10 STEP .5 ...  NEXT j.
```

`STEP` is optional, and if none is given, then 1 is used by default.

Example D.4 *Program*

```
FOR j=1 TO 100
S=S+j
NEXT J
PRINT S
```

computes and prints the total of the first 100 integers (5050).

Conditional action is accomplished by

```
IF ...  THEN ...  ELSE ...  END IF
```

`END IF` is required only when multiple lines of instructions are to be executed.

Selection from several cases is perhaps easiest through

```
   SELECT var ...
CASE 0 ...
CASE .5 ...
CASE ELSE
END SELECT
```

Conditional loops (the ones that last indefinitely, unless special circumstance is encountered) can be programmed through

```
WHILE cond ...  WEND
```

or through

```
DO
....
IF cond THEN EXIT DO
...
LOOP
```

Other ways of breaking out of `DO ...  LOOP` are available, too.

Example D.5 *The following program illustrates several conditional instructions. It finds consecutive max* = 30000 *prime numbers larger than n* = 1

```
n=1
max = 30000
nc = n - 1
WHILE k < max
    DO
        nc = nc + 1
        prime = -1    'TRUE
        FOR j = 2 TO SQR(nc)
        IF (nc MOD j) = 0 THEN prime = 0: EXIT FOR
        NEXT j
        IF prime THEN EXIT DO
    LOOP
    k = k + 1
    Print k;"-th prime is "; nc
WEND
```

Exercise D.1 *Write the program solving recurrence (7.1).*

D.5 User input

To stop the execution of any program, press simultaneously `Ctrl`+`Break`.

To have user supply a value for variable X, write `INPUT "Real number=";X`

Note: instruction `INPUT` stops the program. The user has to hit `Enter` for the program to continue.

To scan for the key pressed by the user without stopping the program, use `INKEY$` function instead of `INPUT`. This allows the user to control the program by simple menu functions. For example, embed the following lines within `DO ... LOOP` and stop the program by pressing KeyQ.

```
Key$=INKEY$
SELECT CASE Key$
     CASE "Q"
     END 'stop the program
     CASE "q"
     END 'stop the program
     CASE "?"
     PRINT "Did you ask for help?"
     CASE ELSE
      BEEP
END SELECT
```

A simple way to let the user know that something went wrong is to `BEEP`.

Exercise D.2 *Write a program that complains (beeps) when user presses any key.*

Exercise D.3 *Write a program that will type the text provided by user from the keyboard to screen in upper-case regardless what the user selected. (Hint: Function* `T2$=UCASE$(T1$)` *converts to upper-case)*

D.6 More about printing

Instruction `PRINT` is used to print to the screen, and in a slightly modified version, to the file on the disk.

To print to the printer, use `LPRINT` instead of `PRINT`. In the latter case, to force the page out of the printer, end every printing job with `LPRINT CHR$(12)`.

QBASIC provides sophisticated ways of controlling the text output by format, colors, location on the screen. In addition it does have graphic statements, as long as the computer has graphics card. But the only thing needed for us is the regular

```
PRINT "New value is X=";X
```

which outputs the string (in quotation marks) and the value `New value is X=1.234`

Occasionally we may want to do minimal “format” through the semi-colon, or `TAB()`.

```
PRINT ".";
```

To see the effect of the semi-colon, run the above statement in a loop[4]. Then delete the semicolon, and run it again.

For professional formatting of output, look up the instruction `PRINT USING "###.##"` in any BASIC textbook (your public library is a good source!).

[4]
```
FOR j=1 TO 1000
PRINT ".";
NEXT J
```

D.7 Arrays and matrices

Vectors and matrices are handled as arrays. They need to be declared to reserve room in memory. This is accomplished by the dimensioning statement
tt DIM ArrayName (Size1, Size 2, Size 3).
For instance `DIM A(100)` specifies a vector, `DIM A(20,50)` defines a 20×50 matrix. Arrays use up a lot of memory, so don't declare arrays larger than what you need, and pay attention to data type, see Section D.8.

D.8 Data types

QBASIC supports many built in data types. Use declaration like `DIM n AS INTEGER`, or append the name by the corresponding marker `n%`.

- Integer (%)
- Long (&)
- Single (!) – default
- Double (#)
- String ($)

These can be combined into more complex user defined data structures through `Type` declaration.

If variables are not declared, and the name isn't appended with the symbol indicating type, the variables are treated as `single` precision. This is admissible[5] in small programs where speed and memory aren't of major concern.

D.9 User defined FUNCTIONs and SUBs

To create a SUB, choose **New SUB** from the **Edit** menu. To create a FUNCTION, choose **New FUNCTION** from the **Edit** menu.

User defined functions are listed as separate windows within QBASIC. Select **View** to switch between various functions.

Each user=defined function starts with `BEGIN FUNCTION FunctName (x, y,z)` and ends with `END FUNCTION`. The code between these two lines is executed whenever the function is envoked from main program, from another function, or SUB, or from itself. `FunctName` is a name for your function (choose a descriptive one). Arguments `(x,y,z)` are the variables passed to the function.

Example D.6 *The following function soves the linear equation $ax + b = 0$*

```
FUNCTION Solution(a, b)
Solution=-b/a
END FUNCTION
```

Exercise D.4 *Write a function that solves quadratic equation $ax^2 + bx + c = 0$.*

There are just a couple things to remember when making functions:

- The function has to be assigned output by `FunctionName=value`.
- Variables used within function are different from those in the rest of the program, except those passed to it as an argument. Thus you can use the same index j for sums in the program, and in the function. But if a value of a parameter passed is changed within the function, it is changed throughout the program.

[5] Use `DEFINT I-L` to over-ride the default and force all the variables with names that begin with `I` though `L` to be of `Integer` type.

- Function can do more than just return the "value". For instance, it can print, and change values of all/some/none of its variables, and return value.

Once a function is created, you can use it from within a program in the same way as the build in functions. The name of the function carries its value, eg `x=FunctName(23)+FunctName(24)`. Notice that the function can also change values of the variables in its arguments, and perform any other tasks – this behavior is like that of any program.

`SUB`s act like functions, except that the name doesn't carry "value" and the only change (if any) is to the variables passed. To invoke `SUB` from the program, use `CALL SubName(Parameters)`. This method is recommended as it is more resistant to typographical errors.

You can pass a whole array as an argument of a function. The arrays are recognized by the parentheses: `FuncName(A())` is a function of array `A()`.

Exercise D.5 *Write functions* `SimulateBinomial (p,n), SimulateNormal(n,sigma), SimulateExponential(m)` *that simulate a single instance of the Binomial, Normal, Exponential distribution with given parameters.*

D.10 Graphics

Graphic commands are available only on computers with a graphics card (all PC's that run Windows have a graphics card).

It is nice to be able to make simple graphs, but the full topic is beyond the scope of this introduction. Program `LIMTHS.BAS` draws lines and boxes of various colors, in the display rectangle covering a portion of the screen.

If you are seriously interested in graphing the results of your computations, and printing the outcomes, you may want to switch to **Visual Basic** in **Windows**.

D.11 Binary operations

Long integers can be used to represent subsets of an n-element set, for n up to $n = 15$ (Why?). Single number k corresponds to long integer 2^k, the set $\{j, k\}$ is represented by $2^j + 2^k$, etc.

If `S, T` are integers representing sets, then `S AND T` represents the intersection of sets, `S OR T` is the union, and `S AND NOT T` is the difference of sets.

To check if $j \in S$, verify if `2^j AND T` is non-zero.

D.12 File Operations

Beginners in BASIC need no file operations to solve the exercises.

If you want to save your printout to file, print to the ASCII file sequentially. This is slower and less versatile than binary, but easier to master.

The syntax is quite rigid. The following example contains the basic idea:

```
OPEN FileName FOR OUTPUT as #17

PRINT #17, "HELLO"

CLOSE #17
```

You can print to file text, numbers, etc. If the file with the sam name as output file already exists, it is replaced by the new one (overwritten). To add rows to an existing file without loosing its contents use `OPEN FileName FOR APPEND as #17`

You can read input from programs back into the program by the corresponding `INPUT` statements. Beware that the file has to have the format expected by the `INPUT`.

The following simple program reads entries from the file `FileName` and prints it onto the screen, one by one.

```
OPEN FileName FOR INPUT as #17
WHILE NOT EOF(17)
INPUT #17, H$
PRINT H$
WEND
CLOSE #17
PRINT H$
```

Change `INPUT #17, H$` to `LINE INPUT #17, H$` to get full text rather than words.

D.13 Good programming

Adhering to good programming principles pays in clarity of programs, and facilitates debugging (finding errors).

- Get into the habit of structural programming.
    - Be aware of the distinction between main program, and subprograms. The main program should play different role - it should direct the course of action, not do the actions. Avoid formulas, computations, algorithms in the main part of the program. Have the "subordinate" subprograms do the tasks
    - Split longer subprograms into smaller blocks (also subprograms, or modules), preferably the ones you can re-use. As a rule of thumb - subprogram listing should fit within one typed page (60 lines).
    - Generalize! If you want to average 3 numbers, you can use `FUNCTION Average(x1,x2,x3)`. But you should write tt FUNCTION Average (X()) that averages as many numbers as you ask.
- Use comments! Write the purpose of each subprogram before writing the code. Test the operation of the program with "empty", or test subprograms before you write the actual code for your subprograms.
- Avoid `Label` and `GoTo` instructions. Whenever possible, use `While ... End`, `Do ... Loop`, or `For ... ... End` constructions.
- Test each sub-program separately, ne at a time, and use only well-tested modules.

Adhering to the principles below is not a guarantee that the programs will work. It is also possible to write programs that execute correctly without any of the below. Nevertheless, it is a good habit to follow these recommendation. The gain is in clarity of the program, readability of its portions. Consequently, you will be able to design more complex programs that execute as expected. You will also re-use components easier.

If you wrote a program that uses `GOTO` statement(s), it is a good exercise to re-write it without a single `GOTO` instruction!

D.14 Example: designing an automatic card dealer

Modern BASIC is a structural language. The objective of this example is to show how various features of BASIC interplay in a design of a card-dealing application.

Here is the description of the situation.

> *A deck of cards consists of 52 cards. Each card has two attributes:*
>
> - *Suit (hearts, spades diamonds, and clubs)*
> - *Value (1 through 13)*

Card games require shuffling the deck, and dealing some number of cards from the top of the deck. The objective of this example is to write a program that will print out the number of requested cards twice. That is, the first player will get as many cards as she requests, and then the second player will get as many cards as he requests (from the rest of the deck!

D.14.1 First Iteration

Once we realize what are the natural steps the real person would go through, the program is very easy to design. Here is the program(!)

```
'PROGRAM: GIVECARD.BAS
InitializeDeck
ShuffleDeck
'ask first player
n=HowManyCards
DealCards(n)
'ask second player
n=HowManyCards
DealCards(n)
```

What remains is only to decide what each step should do, and how the information about the cards will be stored. Since storing the information determines how it is passed between subprograms, we begin with determining this part.

We may want to use an array `Deck(52)` which will be of "user defined" type. The advantage of this approach is that we may modify the information we w "store" with each card with minimal changes in the program itself.

```
DEF TYPE Cards
Suite as string
Value as integer ' we may want string here, too!
End type
```

Afterwards we may declare two shared arrays

```
Dim Shared Deck(52) as cards
DIM Shared Order(52)
```

`Shared` means that every SUB in the program can access values of `Order(j)`, and `Deck(j)`. The first card dealt will be `Deck(Order(1))`. The definition of type says that its value is `Deck(Order(1)).value` and its suit is `Deck(Order(1)).suit`.

SUBS

The simplest way to begin designing SUBS is to describe the purpose of each function/SUB with no code.

```
SUB InitializeDeck
'Initialize cards to their values.
'
END SUB
```

The next routine is perhaps not easy to write for a beginner, but we have a good example in the book.

```
SUB ShuffleDeck
'Make a random permutation
'Store it in shared array Order()
'Order(1), Order(2) are distinct random numbers range 1,...n,
END SUB
```

The next routine interacts with the user. User interaction should ALWAYS be implemented as a separate SUB. As straightforward as it seems, reliable coding requires extensive checking for errors resulting from unpredicted user reactions.

```
'ask first player
FUNCTION HowManyCards
' Ask user how many cards (s)he requests
' store in variable.
'  Check for possible errors in input
' return value if enough cards are left.
END FUNCTION
```

The following routine is in charge of giving out cards. Since the order of cards was already determined, it seems straightforward. But again complications arise, and this portion will be much easier to handle if coded as a separate SUB

```
SUB DealCards(n)
' Remember how many cards are left
' Check how many cards are left
' Print out Error message if not enough cards
'Print next n cards
' You have to decide here HOW the cards will appear on screen:
'words?  pictures?  numbers?
'  (In this example, it will be numbers)
END SUB
```

D.14.2 Second Iteration

Rather than beginning to code the actual functions, we may want to double check that the "flow" of our program is as we expect it. We may write a "dummy" versions of the more difficult parts of the program, and test its operation. Only after we are sure that the program behaves as expected, we can invest more time into coding more difficult parts.

Here is a test program. It was produced from the previously described code; all newly added parts are clearly marked so that they can be removed once not needed.

```
DEF TYPE Cards
'***test***
REM Suite as string
Suite as integer
'*** end test ***
Value as integer ' we may want string here, too!
End type

Dim Shared Deck(52) as cards
DIM Shared Order(52)

InitializeDeck
ShuffleDeck
'ask first player
n=HowManyCards
DealCards(n)
'ask second player
n=HowManyCards
DealCards(n)

SUB InitializeDeck
'Initialize cards to their values.
'*** test ***
```

```
for j = 1 to 52
Deck(j).value=j mod 13 +1
Deck(j).suit=j mod 4 +1
next j
'*** end test ***
END SUB

SUB ShuffleDeck
'Make a random permutation
'Store it in Order()
'Order(1), Order(2) are distinct random numbers range 1,...n,
'*** test ***
'Factory order
For j= 1 to 52
Order(j)=j
NEXT j
'*** end test ***
END SUB

FUNCTION HowManyCards%
' Ask user how many cards (s)he requests
' store in variable.
'  Check how many cards are left
' Print out Error message if not enough cards
' return value if enough cards are left.
'*** test ***
INPUT ``How many cards"; x
' should check for "crazy" answers here
HowManyCards=x
'*** end test ***
END FUNCTION

SUB DealCards(n)
' Remember how many cards are left
'Print next n cards
'*** test ***
' just print first n cards for now
Print "Your cards are:"
FOR j=1 TO n
Print Deck(Order(j)).Value ; " of suit No" ;Deck(Order(j)).Suit
NEXT j
'*** end test ***
END SUB
```

With this “skeleton” program we can check the following things:

1. Does the program do what we wanted? Are shared variables shared between subprograms?
2. Are there any preliminary coding mistakes/typos? Are the variables of correct type (as declared in each SUB/FUNCTION)?
3. Does the output routine operate correctly? (Ask for various numberst of cards. Reverse the order in `SUB ShuffleDeck`.)

D.14.3 Third iteration

Now we are ready to design/code each SUB. This is left for the reader to do. Here are some hints.

- You can use `SELECT CASE ... END SELECT` in `SUB InitializeDeck` to assign words to suits, or even characters `CHR$(3) -- CHR$(6)`.
- Use `STATIC` variable to remember which card to deal from in `SUB DealCards(n)`
- Since we have only 52 cards, randomization in `SUB ShuffleDeck` doesn't have to be fast. But you may want to implement there a more realistic simulation of shuffling (cutting deck in half, mixing the halves, etc.)
- To implement a reasonable error detection in `FUNCTION HowManyCards%` you may just reject requests for negative number of cards, and for more than 52 cards. `INPUT` statement has some protection built in – fractional numbers will go through, but strings at least will be stopped.
- `SUB DealCards(n)` is the best place to check if there is enough cards left. (If not, request `FUNCTION HowManyCards%` from there again).

List of Tables

Bibliography

[1] R. Motwani & P. Raghavan, *Randomized Algorithms*, Cambridge University Press, Cambridge 1995.

[2] N. I. Akhiezer, *The Classical Moment Problem*, Oliver & Boyd, Edinburgh, 1965.

[3] P. Billingsley, *Probability and measure*, Wiley, New York 1986.

[4] P. Billingsley, *Convergence of probability measures*, Wiley New York 1968.

[5] W. Bryc, *Normal Distribution, characterizations with applications*, Lecture Notes in Statist. v. 100 (1995).

[6] A. Dembo & O. Zeitouni, *Large Deviation Techniques and Applications*, Jones and Bartlett Publ., Boston 1993.

[7] R. Durrett, *Probability: Theory and examples*, Wadsworth, Belmont, Ca 1991.

[8] K. T. Fang & T. W. Anderson, *Statistical inference in elliptically contoured and related distributions*, Allerton Press, Inc., New York 1990.

[9] K. T. Fang, S. Kotz & K.-W. Ng, *Symmetric Multivariate and Related Distributions*, Monographs on Statistics and Applied Probability 36, Chapman and Hall, London 1990.

[10] W. Feller, *An Introduction to Probability Theory*, Vol. I, Wiley 196. Vol II, Wiley, New York 1966.

[11] J. F. W. Herschel, Quetelet on Probabilities, *Edinburgh Rev.* 92 (1850) pp. 1–57.

[12] K. Ito & H. McKean *Diffusion processes and their sample paths,* Springer, New York 1964.

[13] N. L. Johnson & S. Kotz, *Distributions in Statistics: Continuous Multivariate Distributions*, Wiley, New York 1972.

[14] N. L. Johnson & S. Kotz & A. W. Kemp, *Univariate discrete distributions*, Wiley, New York 1992.

[15] A. M. Kagan, Ju. V. Linnik & C. R. Rao, *Characterization Problems of Mathematical Statistics*, Wiley, New York 1973.

[16] A. N. Kolmogorov, *Foundations of the Theory of Probability*, Chelsea, New York 1956.

[17] H. H. Kuo, *Gaussian Measures in Banach Spaces*, *Lecture Notes in Math.*, Springer, Vol. 463 (1975).

[18] J. C. Maxwell, *Illustrations of the Dynamical Theory of Gases*, Phil. Mag. 19 (1860), pp. 19–32. Reprinted in *The Scientific Papers of James Clerk Maxwell*, Vol. I, Edited by W. D. Niven, Cambridge, University Press 1890, pp. 377–409.

[19] W. Magnus & F. Oberhettinger, *Formulas and theorems for the special functions of mathematical physics*, Chelsea, New York 1949.

[20] K. S. Miller, *Multidimensional Gaussian Distributions*, Wiley, New York 1964.

[21] J. K. Patel & C. B. Read, *Handbook of the normal distribution*, Dekker, New York 1982.

[22] B. L. S. Prakasa Rao, *Identifiability in Stochastic Models* Acad. Press, Boston 1992.

[23] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical recipes in C*, Cambridge University Press, New York 1992.

[24] M. Rosenblatt, *Stationary sequences and random fields*, Birkhäuser, Boston 1985.

[25] R. Sikorski, *Advanced Calculus*, PWN, Warsaw 1969.

[26] Y. L. Tong, *The Multivariate Normal Distribution*, Springer, New York 1990.

Index