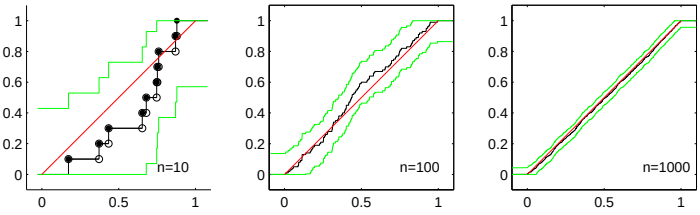
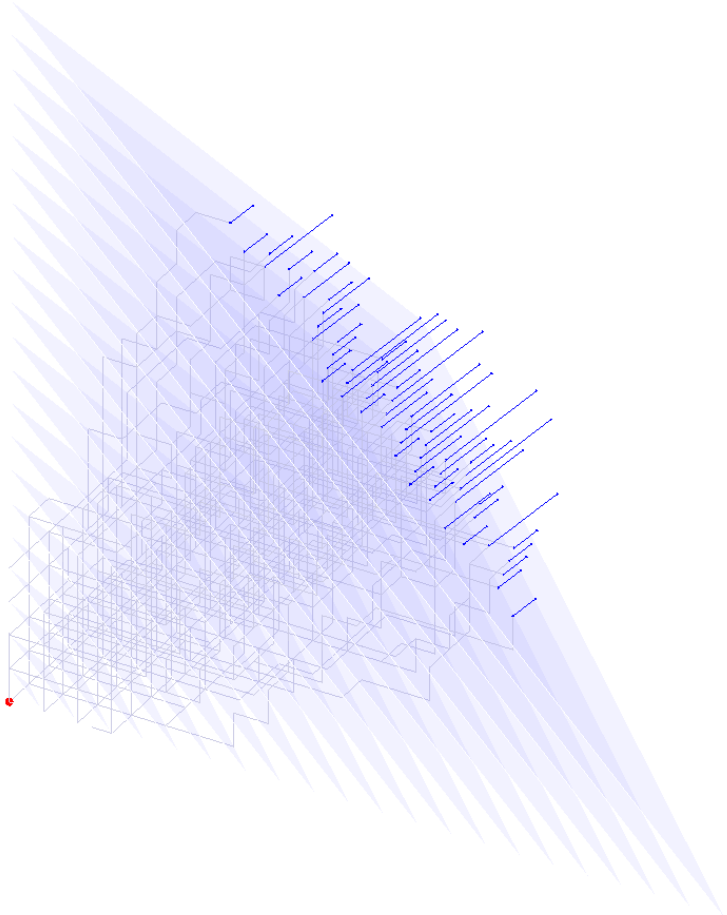
Computational Statistical Experiments I version 0.2SL

Raazesh Sainudiin and Dominic Lee

This work was partially supported by NSF grant DMS-03-06497 and NSF/NIGMS grant DMS-02-01037.





Course Syllabus

STAT 218 - 08S2 (C)	Semester Two 2008
11 points, 0.0917 EFTS	14/07/2008-12/11/2008

Computational Methods in Statistics
Course Syllabus:Wednesday, 12th July 2008

Course Coordinator: Raazesh Sainudiin

0.1 About the Course

The power of modern computers has unleashed new ways of thinking about statistics and implementing statistical solutions. This course introduces the student to computational techniques with uses ranging from exploratory data analysis to statistical inference. These techniques are now widely used and are fast becoming indispensable in the modern statistical toolkit. The course will provide the student with a sound understanding of the computational methods, and with hands-on experience in implementing and using them. Topics include generating random variables, Monte Carlo integration and importance sampling, bootstrap methods, Markov chain Monte Carlo, kernel density estimation and regression, and classification and regression trees.
Pre-requisites: STAT111 or STAT112 or MATH108 or MATH115 or MATH171

0.2 Formal Interactions

Lectures

Days: Mondays and Wednesdays **Times**: 1400 –1500 hours
Place: Room S6, Science Lecture Theatre
Lecturer: Raazesh Sainudiin
Purpose: To communicate the essential concepts and techniques necessary for a sound application of computers to solve statistical problems as described in § 0.1.

You are encouraged to be actively engaged during the lectures to make the most of them. Since it is impossible to impart knowledge to a passive audience, it is in your best interest to think of the lectures as an interactive video game as opposed to a TV show. Please ask questions at anytime during the lectures.

Laboratories

Days: Wednesdays OR Thursdays
Times: 1600 –1700 hours
Place: 035 (Wednesdays) and 036 (Thursdays) Erskine Building
Tutor: Zhu Sha (Joe) **Email**: `szh41@student.canterbury.ac.nz`
Purpose: To give you a taste of implementing code in a personal computer. This activity will reinforce and complement the lectures.

0.3 Individualised Interactions & Course Communications

Lecturer: Raazesh Sainudiin, 724 Erskine Building
Email: `r.sainudiin@math.canterbury.ac.nz`
Office Hours: TBA
Purpose: To clarify specific topics or examples that were unclear to a student or a small group of students during past formal interactions.
Course url: `http://www.math.canterbury.ac.nz/~r.sainudiin/courses/STAT218`

0.4 Student Evaluation

22% Laboratory Attendance and Completion of Labworks and Computer Exercises (2% $\times$ 11 Labwork Submissions).

Labwork submission At the end of the lab, we expect you to hand in some work. Create a diary file named *LabWeek#.txt*, where # is the week number (see column 2 of Table 1), of your laboratory work. The diary file should show what you did in each lab. For full credit you should complete the assigned Labworks and Exercises for the given week and electronically submit the diary file at `http://www.math.canterbury.ac.nz/php/resources/tools/submit/assignment` by the deadline (by midnight on the Friday of the following week).

12% Mid-semester assignment that will involve coding and/or pencil & paper

26% Term project report – **20.0% based on written report & 6% based on partly peer-reviewed oral presentation.** Please visit `http://www.math.canterbury.ac.nz/~r.sainudiin/courses/STAT218/projects/Stat218StudentProjects2007.pdf` to see the term projects completed by students of STAT 218 from 2007.

40% Open book, open notes final exam.

Lecture Notes will be updated to the course web-page regularly about a week after the lecture. Taking your own notes in class is essential. The topics may change as we may not be able to cover such an ambitious survey of computational statistics.

0.5 Time Table & Course Overview

Table 1: Time Table & Course Overview

Lec.	Lab.	Date	Topics	Section/Labworks/Simulations
1		14-07-08	Preliminaries: Set Theory, Numbers, Functions ,...	1.1, 1.2, 1.3
2		16-07-08	Probability	1.4, 2.1
	1	16/17-07-08	Introduction to MATLAB	1, 2, 3, 4, 5, 6
3		21-07-08	Probability and Conditional Probability	2.1,2.2
4		23-07-08	Cond. Probability, Independence & Bayes Theorem	2.2
	2	23/24-07-08	Random variables and Expectations	7, 8, 9, 10
5		28-07-08	Random variables	3.1, 3.2
6		30-07-08	Random variables, Expectations & RNG	3.3, 3.4, 4.1, 4.2
	3	30/31-07-08	Statistics & Visualisation	11, 12, 13, 14, 15, 16, 17, 18
7		04-08-08	IID RVs, Statistics, Common RVs	3.5, 5.1, 5.2, 6.1, 6.2
8		06-08-08	Common RVs & Simulation	6.1, 6.2
	4	06/07-08-08	Simulation of RVs	Sims: 1, 2, 3, 4 & LWs: 19, 20
9		11-08-08	Common RVs & Simulation	6.2, 6.3, 6.4
10		13-08-08	Common RV, R$\vec{\text{V}}$ & Simulation	6.4, 6.5, 6.6
	5	13/14-08-08	Implementation	Sims: 5, 6, 7, 8, 9, 10, 11, 12 LWs: 21, 22, 23, 24
11		18-08-08	Common RV, R$\vec{\text{V}}$ & Simulation	6.4, 6.5, 6.6
12		20-08-08	Rejection Sampling, Other RVs and R$\vec{\text{V}}$s & Experiments	6.7, 6.8, 6.9, 6.10, 6.11
	6	20/21-08-08	Simulation	Sims: 13, 15, 16, 17 & LWs: 26, 27
13		08-09-08	Limits of RVs & Limit Laws in Statistics	7.1, 7.2
14		10-09-08	Limit Laws in Statistics & Point Estimation	7.2, 8.2
	7	10/11-09-08	Point Estimation	LWs: 30, 31
15		15-09-08	Point Estimation & Confidence Sets	8.2, 8.3, 8.4
16		17-09-08	Confidence Intervals & Likelihoods	8.4 , 8.5
	8	17/18-09-08	Implementation	LWs: (31), 32, 33
17		22-09-08	Maximum Likelihood Estimation	9.1
18		24-09-08	Fisher Information	9.3, 9.4
	9	24/25-09-08	Implementation	LWs: 34, 35, 36, 37
19		29-09-08	DF Estimation & Plug-in Estimation	10.1, 10.2
20		01-10-08	Bootstrap	11.1, 11.2
	10	01/02-10-08	Implementation	LWs: 39, 40, 42, 43, 45
21		06-10-08	Hypothesis Testing: size, level, power, Wald Test & p-value	12.1, 12.2, 12.4
22		08-10-08	Permutation Test & Chi-square	12.5, 12.6
	11	08/09-10-08	Oral presentation preparation	email oral presentation
23		13-10-08	Group Project Presentation	
24		15-10-08	Group Project Presentation	
	12	15/16-10-08	Type-setting Written Report	email written report

Contents

List of Tables

List of Figures

Chapter 1

Introduction and Preliminaries

Lecture Notes for STAT 218 S2 2008: Computational Methods in Statistics
Raazesh Sainudiin, Dept. of Maths & Stats, University of Canterbury, Christchurch, NZ.

1.1 Computational Statistical Experiments

A *statistical experimenter* is a person who conducts a *statistical experiment* (for simplicity, from now on, these will be called "experimenters" and "experiments"). Roughly, an experiment is an action with an *empirically observable outcome* (data) that can not necessarily be predicted with certainty, in the sense that a *repetition* of the experiment may result in a different outcome. Most quantitative scientists are experimenters if they apply statistical principles to further their current understanding or *theory* of an *empirically observable real-world phenomenon* (simply, a *phenomenon*). Roughly, 'furthering' your understanding or theory is done by improving your mathematical model ('rigorous cartoon') of the phenomenon on the basis of its compatibility with the *observed data* or *outcome* of the experiment. In this sense, an experimenter attempts to learn about a phenomenon through the outcome of an experiment. An experimenter is often a scientist or engineer, and vice versa.

Technological advances have fundamentally inter-twined computers with most experiments today. First, our instrumentational capacity to observe an empirical phenomenon, by means of automated data gathering (sensing) and representation (storage and retrieval), is steadily increasing. Second, our computational capability to process statistical information or to make decisions using such massive data-sets is also steadily increasing. Thus, our recent technological advances are facilitating computationally intensive statistical experiments based on possibly massive amounts of empirical observations, in a manner that was not viable a decade ago. Hence, a successful scientist or engineer in most specialisations today is a *computational statistical experimenter*, i.e. a statistical experimenter who understands the information structures used to represent their data as well as the statistical algorithms used to process their scientific or engineering decisions. This course is designed to help you take the first steps along this path.

Let us first demonstrate the need for a statistical experiment. Recall that statistical inference or learning is the process of using observations or data to infer some aspect of the distribution function (DF) that generated it. A generic question is:

$$\text{Given realisations from } X_1, X_2, \ldots, X_n \sim \text{some unknown DF } F, \text{ how do we infer } F?$$

Some of the concrete problems involving experiments include:

- **Simulation:** Often, it is necessary to simulate a random variable (RV) with some specific distribution to gain insight into its features or simulate whole systems, such as the air-traffic queues at Heathrow Airport, to make better management decisions.
- **Exploration:** This is the art of (visually) exploring the observed data in order to better understand the empirical phenomenon that generated the data. Visual explorations of simulated data may provide benchmarks against which we may compare and contrast the observed data.
- **Estimation:**
  1. **Parametric Estimation:** Using samples from some unknown DF F that is parameterised by some unknown θ, we can estimate θ from a statistic $\widehat{\Theta}_n$ called the estimator of θ using one of several methods (maximum likelihood, moment estimation or parametric bootstrap).
  2. **Non-parametric Estimation of the DF:** Based on n independent and identically distributed (IID) observations from an unknown DF F, we can estimate it under the general assumption that F belongs to the collection all DFs.
  3. **Confidence Sets:** We can obtain a $1-\alpha$ confidence set for the point estimates, of the unknown parameter θ that belongs to a collection of parameters (also known as parameter space) denoted by Θ or the unknown DF F that belongs to the collection all DFs denoted by $\{\text{all DFs}\}$.
- **Testing:** Based on observations from some DF F that is hypothesised as belonging to a subset Θ_0 of Θ called the space of null hypotheses, we will learn to test (attempt to reject) the falsifiable null hypothesis that F belongs to a smaller collection of parameters denoted by Θ_0. This smaller collection is contained in the parameter space Θ.
- **Learning:** Supervised, unsupervised learning, classification and regression ...

The statistical experiment and the basic objects that make it mathematically definitive and internally consistent are introduced later. The precursor to an experiment is a mathematical model for probability in the 'real world' called the *probability model*. This arises from an axiomatic system based on more fundamental objects, such as sample space, outcomes, events, probability, the addition rule for mutually exclusive events, the definition of conditional probability, limits, real number system, and random variables. These objects in turn build on the set theory, so a refresher in set theory is our first stop. Then, we will introduce the probability model via the intuition behind this construction, and various useful notions associated with continuous and discrete random variables, such as distribution, density & mass functions, and moments. Commonly encountered random variables will serve as examples. We will learn to simulate from these random variables and visualise the realisations. We will visit the most elementary limit theorems before conducting our own computational statistical experiments involving estimation, testing and learning.

1.2 Elementary Set Theory

A **set** is a collection of distinct objects. We write a set by enclosing its elements with curly braces. For example, we denote a set of the two objects $\circ$ and $\bullet$ by:

$$\boxed{\{\circ, \bullet\}}\,.$$

Sometimes, we give names to sets. For instance, we might call the first example set A and write:

$$\boxed{A = \{\circ, \bullet\}} \, .$$

We do not care about the order of elements within a set, i.e. $A = \{\circ, \bullet\} = \{\bullet, \circ\}$. We do not allow a set to contain multiple copies of any of its elements unless the copies are distinguishable, say by labels. So, $B = \{\circ, \bullet, \bullet\}$ is not a set unless the two copies of $\bullet$ in B are labelled or marked to make them distinct, e.g. $B = \{\circ, \tilde{\bullet}, \bullet'\}$. Names for sets that arise in a mathematical discourse are given upper-case letters $(A, B, C, D, \ldots)$. Special symbols are reserved for commonly encountered sets.

Here is the set $\mathfrak{G}$ of twenty two Greek lower-case alphabets that we may encounter later:

$$\mathfrak{G} = \{ \alpha, \quad \beta, \quad \gamma, \quad \delta, \quad \epsilon, \quad \zeta, \quad \eta, \quad \theta, \quad \kappa, \quad \lambda, \quad \mu, \quad \nu, \quad \xi, \quad \pi, \quad \rho, \quad \sigma, \quad \tau, \quad \upsilon, \quad \phi, \quad \chi, \quad \psi, \quad \omega \} \quad .$$

They are respectively named alpha, beta, gamma, delta, epsilon, zeta, eta, theta, kappa, lambda, mu, nu, xi, pi, rho, sigma, tau, upsilon, phi, chi, psi and omega. LHS and RHS are abbreviations for objects on the Left and Right Hand Sides, respectively, of some binary relation. By the notation:

$$\boxed{LHS := RHS} \, ,$$

we mean that LHS **is equal, by definition, to** RHS.

The set which does not contain any element (the collection of nothing) is called the **empty set**:

$$\boxed{\emptyset := \{\,\}} \, .$$

We say an element b **belongs to** a set B, or simply that b belongs to B or that b is an element of B, if b is one of the elements that make up the set B, and write:

$$\boxed{b \in B} \, .$$

When b **does not belong to** B, we write:

$$\boxed{b \notin B} \, .$$

For our example set $A = \{\circ, \bullet\}$, $\star \notin A$ but $\bullet \in A$.

We say that a set C is a **subset** of another set D and write:

$$\boxed{C \subset D}$$

if every element of C is also an element of D. By this definition, any set is a subset of itself.

We say that two sets C and D are **equal** (as sets) and write $C = D$ 'if and only if' ($\iff$) every element of C is also an element of D, and every element of D is also an element of C. This definition of set equality is notationally summarised as follows:

$$\boxed{C = D \quad \iff \quad C \subset D, D \subset C} \, .$$

When two sets C and D are not equal by the above definition, we say that C is **not equal** to D and write:

$$\boxed{C \neq D} \, .$$

The **union** of two sets C and D, written as $C \cup D$, is the set of elements that belong to C or D. We can formally express our definition of set union as:

$$\boxed{C \cup D := \{x : x \in C \quad \text{or} \quad x \in D\}} \, .$$

When a colon (:) appears inside a set, it stands for 'such that'. Thus, the above expression is read as 'C union D is equal by definition to the set of all elements x, such that x belongs to C or x belongs to D.'

Similarly, the **intersection** of two sets C and D, written as $C \cap D$, is the set of elements that belong to both C and D. Formally:

$$\boxed{C \cap D := \{x : x \in C \quad \text{and} \quad x \in D\}} .$$

The set-difference or **difference** of two sets C and D, written as $C \setminus D$, is the set of elements in C that do not belong to D. Formally:

$$\boxed{C \setminus D := \{x : x \in C \quad \text{and} \quad x \notin D\}} .$$

When a universal set, e.g. U is well-defined, the **complement** of a given set B denoted by B^c is the set of all elements of U that don't belong to B, i.e.:

$$\boxed{B^c := U \setminus B} .$$

We say two sets C and D are **disjoint** if they have no elements in common, i.e. $C \cap D = \emptyset$.

Figure 1.1: Universal set Ω, a set A such that $A \subset \Omega$ (read: A is a subset of Ω or A is contained in Ω), another set B with $B \subset \Omega$, the set $A \setminus B$ (read: A (set)minus B or A that is not in B), the set $A \cap B$ (read: A intersection B), the set $A \cup B$ (read: A union B) and the set A^c (read: complement of A) which is defined as $\Omega \setminus A$

.

Classwork 1 *Suppose we are given a universal set U, and three of its subsets, A, B and C. Also suppose that $A \subset B \subset C$. Find the circumstances, if any, under which each of the following statements is true (T) and justify your answer:*

(1) $C \subset B$	*T when* $B = C$	*(2)* $A \subset C$	*T by assumption*
(3) $C \subset \emptyset$	*T when* $A = B = C = \emptyset$	*(4)* $\emptyset \subset A$	*T always*
(5) $C \subset U$	*T by assumption*	*(6)* $U \subset A$	*T when* $A = B = C = U$

1.3 Natural Numbers, Integers and Rational Numbers

We denote the number of elements in a set named B by:

$$\boxed{\#B := \text{Number of elements in the set } B}\ .$$

In fact, the Hindu-Arab numerals we have inherited are based on this intuition of the size of a collection. The elements of the set of **natural numbers**:

$\boxed{\mathbb{N} := \{1, 2, 3, 4, \ldots\}}$, may be defined using $\#$ as follows:

$$\begin{aligned} 1 &:= \#\{\star\} = \#\{\bullet\} = \#\{\alpha\} = \#\{\{\bullet\}\} = \#\{\{\bullet, \bullet'\}\} = \ldots, \\ 2 &:= \#\{\star', \star\} = \#\{\bullet, \circ\} = \#\{\alpha, \omega\} = \#\{\{\circ\}, \{\alpha, \star, \bullet\}\} = \ldots, \\ &\vdots \end{aligned}$$

For our example sets, $A = \{\circ, \bullet\}$ and the set of Greek alphabets $\mathcal{G}$, $\#A = 2$ and $\#\mathcal{G} = 22$. The number zero may be defined as the size of an empty set:

$$0 := \#\emptyset = \#\{\}$$

The set of **non-negative integers** is:

$$\boxed{\mathbb{Z}_+ := \mathbb{N} \cup \{0\} = \{0, 1, 2, 3, \ldots\}}\ .$$

A **product set** is the **Cartesian product** ($\times$) of two or more possibly distinct sets:

$$\boxed{A \times B := \{(a, b) : a \in A \text{ and } b \in B\}}$$

For example, if $A = \{\circ, \bullet\}$ and $B = \{\star\}$, then $A \times B = \{(\circ, \star), (\bullet, \star)\}$. Elements of $A \times B$ are called **ordered pairs**.

The binary arithmetic operation of **addition** ($+$) between a pair of non-negative integers $c, d \in \mathbb{Z}_+$ can be defined via sizes of disjoint sets. Suppose, $c = \#C$, $d = \#D$ and $C \cap D = \emptyset$, then:

$$c + d = \#C + \#D := \#(C \cup D)\ .$$

For example, if $A = \{\circ, \bullet\}$ and $B = \{\star\}$, then $A \cap B = \emptyset$ and $\#A + \#B = \#(A \cup B) \iff 2 + 1 = 3$.

The binary arithmetic operation of **multiplication** ($\cdot$) between a pair of non-negative integers $c, d \in \mathbb{Z}_+$ can be defined via sizes of product sets. Suppose, $c = \#C$, $d = \#D$, then:

$$c \cdot d = \#C \cdot \#D := \#(C \times D)\ .$$

For example, if $A = \{\circ, \bullet\}$ and $B = \{\star\}$, then $\#A \cdot \#B = \#(A \times B) \iff 2 \cdot 1 = 2$.

More generally, a product set of $A_1, A_2, \ldots, A_m$ is:

$$\boxed{A_1 \times A_2 \times \cdots \times A_m := \{(a_1, a_2, \ldots, a_m) : a_1 \in A_1, a_2 \in A_2, \ldots, a_m \in A_m\}}$$

Elements of an m-product set are called **ordered m-tuples**. When we take the product of the same set we abbreviate as follows:

$$\boxed{A^m := \underbrace{A \times A \times \cdots \times A}_{m \text{ times}} := \{(a_1, a_2, \ldots, a_m) : a_1 \in A, a_2 \in A, \ldots, a_m \in A\}}$$

Classwork 2 *1. Let $A = \{\circ, \bullet\}$. What are the elements of A^2? 2. Suppose $\#A = 2$ and $\#B = 3$. What is $\#(A \times B)$? 3. Suppose $\#A_1 = s_1, \#A_2 = s_2, \ldots, \#A_m = s_m$. What is $\#(A_1 \times A_2 \times \cdots \times A_m)$?*

Now, let's recall the definition of a function. A **function** is a "mapping" that associates each element in some set $\mathbb{X}$ (the domain) to exactly one element in some set $\mathbb{Y}$ (the range). Two different elements in $\mathbb{X}$ can be mapped to or associated with the same element in $\mathbb{Y}$, and not every element in $\mathbb{Y}$ needs to be mapped. Suppose $x \in \mathbb{X}$. Then we say $f(x) = y \in \mathbb{Y}$ is the **image** of x. To emphasise that f is a **function** from $\mathbb{X} \ni x$ to $\mathbb{Y} \ni y$, we write:

$$\boxed{f(x) = y : \mathbb{X} \mapsto \mathbb{Y}} .$$

And for some $y \in \mathbb{Y}$, we call the set:

$$\boxed{f^{[-1]}(y) := \{x \in \mathbb{X} : f(x) = y\} \subset \mathbb{X}} ,$$

the **pre-image** or **inverse image** of y, and

$$\boxed{f^{[-1]} := f^{[-1]}(y \in \mathbb{Y}) = X \subset \mathbb{X}} ,$$

as the **inverse** of f.

Figure 1.2: A function f ("father of") from $\mathbb{X}$ (a set of children) to $\mathbb{Y}$ (their fathers) and its inverse ("children of").

We motivated the non-negative integers $\mathbb{Z}_+$ via the size of a set. With the notion of two directions ($+$ and $-$) and the magnitude of the current position from the origin zero (0) of a dynamic entity, we can motivate the set of **integers**:

$$\boxed{\mathbb{Z} := \{\ldots, -3, -2, -1, 0, +1, +2, +3, \ldots\}} .$$

The integers with a **minus** or **negative sign** ($-$) before them are called negative integers and those with a **plus** or **positive sign** ($+$) before them are called positive integers. Conventionally, $+$ signs are dropped. Some examples of functions you may have encountered are **arithmetic operations**

such as **addition** $(+)$, **subtraction** $(-)$, **multiplication** $(\cdot)$ and **division** $(/)$ of ordered pairs of integers. The reader is assumed to be familiar with such arithmetic operations with pairs of integers. Every integer is either positive, negative, or zero. In terms of this we define the notion of **order**. We say an integer a is **less than** an integer b and write $a < b$ if $b - a$ is positive. We say an integer a is **less than or equal to** an integer b and write $a \leq b$ if $b - a$ is positive or zero. Finally, we say that a is greater than b and write $a > b$ if $b < a$. Similarly, a is greater than equal to b, i.e. $a \geq b$, if $b \leq a$. The set of integers are **well-ordered**, i.e., for every integer a there is a next largest integer $a + 1$.

Classwork 3 *Consider the set of integers* $\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$. *Try to set up the arithmetic operation of addition as a function. The domain for addition is the Cartesian product of* $\mathbb{Z}$*:*

$$\mathbb{Z}^2 := \mathbb{Z} \times \mathbb{Z} := \{(a, b) : a \in \mathbb{Z}, b \in \mathbb{Z}\}$$

What is its range ?

$$+ : \mathbb{Z} \times \mathbb{Z} \mapsto$$

Figure 1.3: A pictorial depiction of addition and its inverse. The domain is plotted in orthogonal **Cartesian coordinates**.

If the magnitude of the entity's position is measured in units (e.g. meters) that can be rationally divided into q pieces with $q \in \mathbb{N}$, then we have the set of rational numbers:

$$\mathbb{Q} := \{p/q : p \in \mathbb{Z}, q \in \mathbb{Z} \setminus \{0\}\}$$

The expressions p/q and p'/q' denote the same rational number if and only if $p \cdot q' = p' \cdot q$. Every rational number has a unique irreducible expression p/q, where q is positive and as small as possible. For example, 1/2, 2/4, 3/6, and 1001/2002 are different expressions for the same rational number whose irreducible unique expression is 1/2.

Addition and multiplication are defined for rational numbers by:

$$\frac{p}{q} + \frac{p'}{q'} = \frac{p \cdot q' + p' \cdot q}{q \cdot q'} \qquad \text{and} \qquad \frac{p}{q} \cdot \frac{p'}{q'} = \frac{p \cdot p'}{q \cdot q'} \ .$$

The rational numbers form a **field** under the operations of addition and multiplication defined above in terms of addition and multiplication over integers. This means that the following properties are satisfied:

1. Addition and multiplication are each **commutative**

$$a + b = b + a, \qquad a \cdot b = b \cdot a \ ,$$

and associative

$$a + (b + c) = (a + b) + c, \qquad a \cdot (b \cdot c) = (a \cdot b) \cdot c \ .$$

2. Multiplication **distributes** over addition

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c) \ .$$

3. 0 is the **additive identity** and 1 is the multiplicative identity

$$0 + a = a \qquad \text{and} \qquad 1 \cdot a = a \ .$$

4. Every rational number a has a negative, $a + (-a) = 0$ and every non-zero rational number a has a reciprocal, $a \cdot 1/a = 1$.

The field axioms imply the usual laws of arithmetic and allow subtraction and division to be defined in terms of addition and multiplication as follows:

$$\frac{p}{q} - \frac{p'}{q'} := \frac{p}{q} + \frac{-p'}{q'} \qquad \text{and} \qquad \frac{p}{q} / \frac{p'}{q'} := \frac{p}{q} \cdot \frac{q'}{p'}, \quad \text{provided } p' \neq 0 \ .$$

We will see later that the theory of finite fields is necessary for the study of pseudo-random number generators (PRNGs) and PRNGs are the heart-beat of randomness and statistics with computers.

1.4 Real Numbers

Unlike rational numbers which are expressible in their reduced forms by p/q, it is fairly tricky to define or express real numbers. It is possible to define real numbers formally and constructively via equivalence classes of Cauchy sequence of rational numbers. For this all we need are notions of (1) infinity, (2) sequence of rational numbers and (3) distance between any two rational numbers in an infinite sequence of them. These are topics usually covered in an introductory course in real analysis and are necessary for a firm foundation in computational statistics. Instead of a formal constructive definition of real numbers, we give a more concrete one via decimal expansions. See Donald E. Knuth's treatment [*Art of Computer Programming, Vol. I, Fundamental Algorithms*, 3rd Ed., 1997, pp. 21-25] for a fuller story. A **real number** is a numerical quantity x that has a decimal expansion:

$$x = n + 0.d_1 d_2 d_3 \ldots \ , \text{ where, each } d_i \in \{0, 1, \ldots, 9\}, n \in \mathbb{Z} \ ,$$

and the sequence $0.d_1 d_2 d_3 \ldots$ does not terminate with infinitely many consecutive 9s. By the above decimal representation, the following arbitrarily accurate enclosure of the real number x by rational numbers is implied:

$$n + \frac{d_1}{10} + \frac{d_2}{100} + \cdots + \frac{d_k}{10^k} =: \underline{x}_k \leq x < \overline{x}_k := n + \frac{d_1}{10} + \frac{d_2}{100} + \cdots + \frac{d_k}{10^k} + \frac{1}{10^k}$$

for every $k \in \mathbb{N}$. Thus, rational arithmetic $(+, -, \cdot, /)$ can be extended with arbitrary precision to any ordered pair of real numbers x and y by operations on their rational enclosures $\underline{x}, \overline{x}$ and $\underline{y}, \overline{y}$.

Some examples of real numbers that are not rational (**irrational numbers**) are:

$$\begin{aligned}\sqrt{2} &= 1.41421356237309\ldots\text{the side length of a square with area of 2 units}\\ \pi &= 3.14159265358979\ldots\text{the ratio of the circumference to diameter of a circle}\\ e &= 2.71828182845904\ldots\text{Euler's constant}\end{aligned}$$

We can think of π as being enclosed by the following pairs of rational numbers:

$$\begin{aligned}3+\frac{1}{10} =: \underline{\pi}_1 \leq \pi < \overline{\pi}_1 &:= 3+\frac{1}{10}+\frac{1}{10^1}\\ 3+\frac{1}{10}+\frac{4}{100} =: \underline{\pi}_2 \leq \pi < \overline{\pi}_2 &:= 3+\frac{1}{10}+\frac{4}{100}+\frac{1}{100}\\ 3+\frac{1}{10}+\frac{4}{100}+\frac{1}{10^3} =: \underline{\pi}_3 \leq \pi < \overline{\pi}_3 &:= 3+\frac{1}{10}+\frac{4}{100}+\frac{1}{10^3}+\frac{1}{10^3}\\ &\vdots\\ 3.14159265358979 =: \underline{\pi}_{14} \leq \pi < \overline{\pi}_{14} &:= 3.14159265358979+\frac{1}{10^{14}}\\ &\vdots\end{aligned}$$

Think of the real number system as the continuum of points that make up a line, as shown in Figure 1.4.

Figure 1.4: A depiction of the real line segment $[-10, 10]$.

Let y and z be two real numbers such that $y \leq z$. Then, the **closed interval** $[y, z]$ is the set of real numbers x such that $y \leq x \leq z$:

$$[y, z] := \{x : y \leq x \leq z\} .$$

The **half-open interval** $(y, z]$ or $[y, z)$ and the **open interval** (y, z) are defined analogously:

$$\begin{aligned}(y, z] &:= \{x : y < x \leq z\} ,\\ [y, z) &:= \{x : y \leq x < z\} ,\\ (y, z) &:= \{x : y < x < z\} .\end{aligned}$$

We also allow y to be **minus infinity** (denoted $-\infty$) or z to be **infinity** (denoted ∞) at an open endpoint of an interval, meaning that there is no lower or upper bound. With this allowance we get the set of **real numbers** $\mathbb{R} := (-\infty, \infty)$, the **non-negative real numbers** $\mathbb{R}_+ := [0, \infty)$ and the **positive real numbers** $\mathbb{R}_{>0}(0, \infty)$ and .

$$\begin{aligned}\mathbb{R} := (-\infty, \infty) &= \{x : -\infty < x < \infty\} ,\\ \mathbb{R}_+ := [0, \infty) &= \{x : 0 \leq x < \infty\} ,\\ \mathbb{R}_{>0} := [0, \infty) &= \{x : 0 < x < \infty\} .\end{aligned}$$

For a positive real number $b \in \mathbb{R}_{>0}$ and an integer $n \in \mathbb{Z}$, the n-th **power** or **exponent** of b is:

$$b^0 = 1, \qquad b^n = b^{n-1} \cdot b \quad \text{if } n > 0, \qquad b^n = b^{n+1}/b \quad \text{if } n < 0 \ .$$

The following **laws of exponents** hold by mathematical induction when $m, n \in \mathbb{Z}$:

$$b^{m+n} = b^m \cdot b^n, \qquad (b^m)^n = b^{m \cdot n} \ .$$

If $y \in \mathbb{R}$ and $m \in \mathbb{N}$, the unique positive real number $z \in \mathbb{R}_{>0}$ such that $z^m = y$ is called the **m-th root of** y and denoted by $\sqrt[m]{y}$, i.e.,

$$z^m = y \implies z = \sqrt[m]{y} \ .$$

For a rational number $r = p/q \in \mathbb{Q}$, we define the r-th power of $b \in \mathbb{R}$ as follows:

$$b^r = b^{p/q} := \sqrt[q]{b^p}$$

The laws of exponents hold for this definition and different expressions for the same rational number $r = ap/aq$ yield the same power, i.e., $b^{p/q} = b^{ap/aq}$. Recall that a real number $x = n + 0.d_1d_2d_3 \ldots \in \mathbb{R}$ can be arbitrarily precisely enclosed by the rational numbers $\underline{x}_k := n + \frac{d_1}{10} + \frac{d_2}{100} + \cdots + \frac{d_k}{10^k}$ and $\overline{x}_k := n + \frac{d_1}{10} + \frac{d_2}{100} + \cdots + \frac{d_k}{10^k} + \frac{1}{10^k}$ by increasing k. Suppose first that $b > 1$. Then, using rational powers, we can enclose b^x,

$$b^{n+\frac{d_1}{10}+\frac{d_2}{100}+\cdots+\frac{d_k}{10^k}} =: b^{\underline{x}_k} \leq b^x < b^{\overline{x}_k} := b^{n+\frac{d_1}{10}+\frac{d_2}{100}+\cdots+\frac{d_k}{10^k}+\frac{1}{10^k}} \ ,$$

within an interval of width $b^{n+\frac{d_1}{10}+\frac{d_2}{100}+\cdots+\frac{d_k}{10^k}}\left(b^{\frac{1}{10^k}} - 1\right) < b^{n+1}(b-1)/10^k$. By taking a large enough k we can evaluate b^x to any accuracy. Finally, when $b < 1$ we define $b^x := (1/b)^x$ and when $b = 0$, $b^x := 1$.

Suppose $y \in \mathbb{R}_{>0}$ and $b \in \mathbb{R} \setminus \{1\}$ then the real number x such that $y = b^x$ is called the **logarithm of y to the base** b and we write this as:

$$y = b^x \iff x = \log_b y$$

The definition implies:

$$x = \log_b(b^x) = b^{\log_b x} \ ,$$

and the laws of exponents imply:

$$\begin{aligned} &\log_b(xy) = \log_b x + \log_b y, \quad \text{if} \quad x > 0,\ y > 0 \ \text{ and} \\ &\log_b(c^y) = y \log_b c, \quad \text{if} \quad c > 0 \ . \end{aligned}$$

The **common logarithm** is $\log_{10}(y)$, the **binary logarithm** is $\log_2(y)$ and the **natural logarithm** is $\log_e(y)$, where e is the Euler's constant. Since we will mostly work with $\log_e(y)$ we use $\log(y)$ to mean $\log_e(y)$. You are assumed to be familiar with trigonometric functions ($\sin(x)$, $\cos(x)$, $\tan(x)$, ...). We sometimes denote the special power function e^y by $\exp(y)$.

Familiar extremal elements of a set of real numbers, say A, are the following:

$$\boxed{\max A := \text{greatest element in } A}$$

For example, $\max\{1, 4, -9, 345\} = 4$, $\max[-93.8889, 1002.786] = 1002.786$.

$$\boxed{\min A := \text{least element in } A}$$

For example, $\min\{1, 4, -9, 345\} = -9$, $\min[-93.8889, 1002.786] = -93.8889$. We need a slightly more sophisticated notion for the extremal elements of a set A that may not belong to A. We say that a real number x is a **lower bound** for a non-empty set of real numbers A, provided $x \leq a$ for every $a \in A$. We say that the set A is **bounded below** if it has at least one lower bound. A lower bound is the **greatest lower bound** if it is at least as large as any other lower bound. The greatest lower bound of a set of real numbers A is called the **infimum** of A and is denoted by:

$$\boxed{\inf A := \text{greatest lower bound of } A}$$

For example, $\inf(0, 1) = 0$ and $\inf\{10.333 \cup [-99, 1001.33)\} = -99$. We similarly define the **least upper bound** of a non-empty set of real numbers A to be the **supremum** of A and denote it as:

$$\boxed{\sup A := \text{least upper bound of } A}$$

For example, $\sup(0, 1) = 1$ and $\sup\{10.333 \cup [-99, 1001.33)\} = 1001.33$. By convention, we define $\inf \emptyset := \infty$, $\sup \emptyset := -\infty$. Finally, if a set A is not bounded below then $\inf A := -\infty$ and if a set A is not bounded above then $\sup A := \infty$.

Symbol	Meaning
$A = \{\star, \circ, \bullet\}$	A is a set containing the elements $\star$, $\circ$ and $\bullet$
$\circ \in A$	$\circ$ belongs to A or $\circ$ is an element of A
$A \ni \circ$	$\circ$ belongs to A or $\circ$ is an element of A
$\odot \notin A$	$\odot$ does not belong to A
$\#A$	Size of the set A, for e.g. $\#\{\star, \circ, \bullet, \odot\} = 4$
$\mathbb{N}$	The set of natural numbers $\{1, 2, 3, \ldots\}$
$\mathbb{Z}_+$	The set of non-negative integers $\{0, 1, 2, 3, \ldots\}$
$\emptyset$	Empty set or the collection of nothing or $\{\}$
$A \subset B$	A is a subset of B or A is contained by B, e.g. $A = \{\circ\}, B = \{\bullet\}$
$A \supset B$	A is a superset of B or A contains B e.g. $A = \{\circ, \star, \bullet\}, B = \{\circ, \bullet\}$
$A = B$	A equals B, i.e. $A \subset B$ and $B \subset A$
$Q \implies R$	Statement Q implies statement R or If Q then R
$Q \iff R$	$Q \implies R$ and $R \implies Q$
$\{x : x$ satisfies property $R\}$	The set of all x such that x satisfies property R
$A \cup B$	A union B, i.e. $\{x : x \in A$ or $x \in B\}$
$A \cap B$	A intersection B, i.e. $\{x : x \in A$ and $x \in B\}$
$A \setminus B$	A minus B, i.e. $\{x : x \in A$ and $x \notin B\}$
$A := B$	A is equal to B by definition
$A =: B$	B is equal to A by definition
A^c	A complement, i.e. $\{x : x \in U,$ the universal set, but $x \notin A\}$
$A_1 \times A_2 \times \cdots \times A_m$	The m-product set $\{(a_1, a_2, \ldots, a_m) : a_1 \in A_1, a_2 \in A_2, \ldots, a_m \in A_m\}$
A^m	The m-product set $\{(a_1, a_2, \ldots, a_m) : a_1 \in A, a_2 \in A, \ldots, a_m \in A\}$
$f := f(x) = y : \mathbb{X} \mapsto \mathbb{Y}$	A function f from domain $\mathbb{X}$ to range $\mathbb{Y}$
$f^{[-1]}(y)$	Inverse image of y
$f^{[-1]} := f^{[-1]}(y \in \mathbb{Y}) = X \subset \mathbb{X}$	Inverse of f
$\mathbb{Z} := \{\ldots, -2, -1, 0, 1, 2, \ldots\}$	Integers
$a < b$ or $a \leq b$	a is less than b or a is less than or equal to b
$a > b$ or $a \geq b$	a is greater than b or a is greater than or equal to b
$\mathbb{Q}$	Rational numbers
(x, y)	the open interval (x, y), i.e. $\{r : x < r < y\}$
$[x, y]$	the closed interval (x, y), i.e. $\{r : x \leq r \leq y\}$
$(x, y]$	the half-open interval $(x, y]$, i.e. $\{r : x < r \leq y\}$
$[x, y)$	the half-open interval $[x, y)$, i.e. $\{r : x \leq r < y\}$
$\mathbb{R} := (-\infty, \infty)$	Real numbers, i.e. $\{r : -\infty < r < \infty\}$
$\mathbb{R}_+ := [0, \infty)$	Real numbers, i.e. $\{r : 0 \leq r < \infty\}$
$\mathbb{R}_{>0} := (0, \infty)$	Real numbers, i.e. $\{r : 0 < r < \infty\}$

Table 1.1: Symbol Table: Sets and Numbers

1.5 Introduction to MATLAB

We use MATLAB to perform computations and visualisations. MATLAB is a numerical computing environment and programming language that is optimised for vector and matrix processing. STAT 218/313 students will have access to Maths & Stats Department's computers that are licensed to run MATLAB. You can remotely connect to these machines from home by following instructions at `http://www.math.canterbury.ac.nz/php/resources/compdocs/remote`.

Labwork 1 (Basics of MATLAB) *Let us familiarize ourselves with* MATLAB *in this session. First, you need to launch* MATLAB *from your terminal. Since this is system dependent, ask your tutor for help. The command window within the* MATLAB *window is where you need to type commands. Here is a minimal set of commands you need to familiarize yourself with in this session.*

1. *Type the following command to add 2 numbers in the command window right after the command prompt* `>>` .

```
>> 13+24
```

Upon hitting `Enter` *or* `Return` *on your keyboard, you should see:*

```
ans =

    37
```

The summand 37 of 13 and 24 is stored in the default variable called `ans` *which is short for answer.*

2. *We can write* **comments** *in* MATLAB *following the* `%` *character. All the characters in a given line that follow the percent character* `%` *are ignored by* MATLAB. *It is very helpful to comment what is being done in the code. You won't get full credit without sufficient comments in your coding assignments. For example we could have added the comment to the previous addition. To save space in these notes, we suppress the blank lines and excessive line breaks present in* MATLAB*'s command window.*

```
>> 13+24 % adding 13 to 24 using the binary arithmetic operator +
ans =    37
```

3. *You can* **create or reopen a diary file** *in* MATLAB *to record your work. Everything you typed or input and the corresponding output in the command window will be recorded in the diary file. You can create or reopen a diary file by typing* `diary filename.txt` *in the command window. When you have finished recording, simply type* `diary off` *in the command window* **to turn off the diary file**. *The diary file with* **.txt** *extension is simply a text-file. It can be edited in different editors after the diary is turned off in* MATLAB. *You need to type* `diary LabWeek1.txt` *to start recording your work for electronic submission (see Section 0.4).*

```
>> diary blah.txt % start a diary file named blah.txt
>> 3+56
ans =    59
>> diary off % turn off the current diary file blah.txt
```

```
>> type blah.txt % this allows you to see the contents of blah.txt
3+56
ans =    59
diary off
>> diary blah.txt % reopen the existing diary file blah.txt
>> 45-54
ans =    -9
>> diary off % turn off the current diary file blah.txt again
>> type blah.txt % see its contents
3+56
ans =    59
diary off
45-54
ans =    -9
diary off
```

4. *Let's learn to store values in variables of our choice. Type the following at the command prompt :*

```
>> VariableCalledX = 12
```

Upon hitting enter you should see that the number 12 *has been assigned to the variable named* `VariableCalledX` *:*

```
VariableCalledX =    12
```

5. MATLAB *stores default value for some variables, such as* `pi` *(π),* `i` *and* `j` *(complex numbers).*

```
>> pi
ans =    3.1416
>> i
ans =        0 + 1.0000i
>> j
ans =        0 + 1.0000i
```

All predefined symbols (variables, constants, function names, operators, etc) in MATLAB *are written in lower-case. Therefore, it is a good practice to name the variable you define using upper and mixed case letters in order to prevent an unintended overwrite of some predefined* MATLAB *symbol.*

6. *We could have stored the sum of 13 and 24 in the variable* `X`*, by entering:*

```
>> X = 13 + 24
X =    37
```

7. *Similarly, you can store the outcome of multiplication (via operation* `*` *), subtraction (via operation* `-`*), division (via* `/` *) and exponentiation (via* `^` *)of any two numbers of your choice in a variable name of your choice. Evaluate the following expressions in* MATLAB*:*

$p = 45.89 * 1.00009$ $\qquad$ $d = 89.0/23.3454$

$m = 5376.0 - 6.00$ $\qquad$ $p = 2^{0.5}$

8. *You may compose the elementary operations to obtain rational expressions by using parenthesis to specify the order of the operations. To obtain* $\sqrt{2}$*, you can type the following into* MATLAB*'s command window.*

```
>> 2^(1/2)
ans =    1.4142
```

The omission of parenthesis about $1/2$ *means something else and you get the following output:*

```
>> 2^1/2
ans =     1
```

MATLAB *first takes the* 1*st power of* 2 *and then divides it by* 2 *using its default precedence rules for binary operators in the absence of parenthesis. The order of operations or default precedence rule for arithmetic operations is 1.* **b***rackets or parentheses; 2.* **e***xponents (powers and roots); 3.* **d***ivision and* **m***ultiplication; 4.* **a***ddition and* **s***ubtraction. The mnemonic* **bedmas** *can be handy. When in doubt, use parenthesis to force the intended order of operations.*

9. *When you try to divide by 0, matlab returns* `Inf` *for infinity.*

```
>> 10/0
ans =   Inf
```

10. *We can clear the value we have assigned to a particular variable and reuse it. We demonstrate it by the following commands and their output:*

```
>> X
X =    37
>> clear X
>> X
??? Undefined function or variable 'X'.
```

Entering `X` *after* `clear`*ing it gives the above self-explanatory error message preceded by* `???`*.*

11. *We can suppress the output on the screen by ending the command with a semi-colon. Take a look at the simple command that sets* X *to* $\sin(3.145678)$ *with and without the '*`;`*' at the end:*

```
>> X = sin(3.145678)
X =   -0.0041
>> X = sin(3.145678);
```

12. *If you do not understand a* MATLAB *function or command then type* `help` *or* `doc` *followed by the function or command. For example:*

```
>> help sin
 SIN    Sine of argument in radians.
    SIN(X) is the sine of the elements of X.
    See also asin, sind.
    Overloaded methods:
       darray/sin
    Reference page in Help browser
       doc sin
>> doc sin
```

It is a good idea to use the help files before you ask your tutor.

13. *Set the variable* `x` *to equal* 17.13 *and evaluate* $\cos(x)$, $\log(x)$, $\exp(x)$, $\arccos(x)$, $\text{abs}(x)$, $\text{sign}(x)$ *using the matlab commands* `cos`, `log`, `exp`, `acos`, `abs`, `sign`, *respectively. Read the help files to understand what each function does.*

14. *When we work with real numbers (floating-point numbers) or really large numbers, we might want the output to be displayed in concise notation. This can be controlled in* MATLAB *using the* `format` *command with the* `short` *or* `long` *options with/without* `e` *for scientific notation.* `format compact` *is used for getting compacted output and* `format` *returns the default format. For example:*

```
>> format compact
>> Y=15;
>> Y = Y + acos(-1)
Y =   18.1416
>> format short
>> Y
Y =   18.1416
>> format short e
>> Y
Y =  1.8142e+001
>> format long
>> Y
Y =  18.141592653589793
>> format long e
>> Y
Y =    1.814159265358979e+001
>> format
>> Y
Y =   18.1416
```

15. *Finally, to quit from* MATLAB *just type* `quit` *or* `exit` *at the prompt.*

```
>> quit
```

16. *An* **M-file** *is a special text file with a* `.m` *extension that contains a set of code or instructions in* MATLAB. *In this course we will be using two types of M-files:* **script** *and* **function** *files. A script file is simply a list of commands that we want executed and saves us from retyping code modules we are pleased with. A function file allows us to write specific tasks as functions with input and output. These functions can be called from other script files, function files or command window. We will see such examples shortly.*

By now, you are expected to be familiar with arithmetic operations, simple function evaluations, format control, starting and stopping a diary file and launching and quitting MATLAB.

1.6 Permutations, Factorials and Combinations

Definition 1 (Permutations and Factorials) *A* **permutation** *of n objects is an arrangement of n distinct objects in a row. For example, there are 2 permutations of the two objects $\{1, 2\}$:*

$$12, \qquad 21 \ ,$$

and 6 permutations of the three objects $\{a, b, c\}$:

$$abc, \quad acb, \quad bac, \quad bca, \quad cab, \quad cba \ .$$

Let the number of ways to choose k objects out of n and to arrange them in a row be denoted by $p_{n,k}$. For example, we can choose two ($k = 2$) objects out of three ($n = 3$) objects, $\{a, b, c\}$, and

arrange them in a row in six ways ($p_{3,2}$):

$$ab, \quad ac, \quad ba, \quad bc, \quad ca, \quad cb \ .$$

Given n objects, there are n ways to choose the left-most object, and once this choice has been made there are $n-1$ ways to select a different object to place next to the left-most one. Thus, there are $n(n-1)$ possible choices for the first two positions. Similarly, when $n > 2$, there are $n-2$ choices for the third object that is distinct from the first two. Thus, there are $n(n-1)(n-2)$ possible ways to choose three distinct objects from a set of n objects and arrange them in a row. In general,

$$p_{n,k} = n(n-1)(n-2)\dots(n-k+1)$$

and the total number of permutations called 'n **factorial***' and denoted by $n!$ is*

$$n! := p_{n,n} = n(n-1)(n-2)\dots(n-n+1) = n(n-1)(n-2)\dots(3)\ (2)\ (1) =: \prod_{i=1}^{n} i \ .$$

Some factorials to bear in mind

$$0! := 1 \quad 1! = 1, \quad 2! = 2, \quad 3! = 6, \quad 4! = 24, \quad 5! = 120 \quad 10! = 3,628,800 \ .$$

When n is large we can get a good idea of $n!$ without laboriously carrying out the $n-1$ multiplications via Stirling's approximation (*Methodus Differentialis (1730), p. 137*) :

$$n! \approxeq \sqrt{2\pi n}\left(\frac{n}{e}\right)^n \ .$$

Definition 2 (Combinations) *The combinations of n objects taken k at a time are the possible choices of k different elements from a collection of n objects, disregarding order. They are called the k-combinations of the collection. The combinations of the three objects $\{a, b, c\}$ taken two at a time, called the 2-combinations of $\{a, b, c\}$, are*

$$ab, \quad ac, \quad bc \ ,$$

and the combinations of the five objects $\{1, 2, 3, 4, 5\}$ taken three at a time, called the 3-combinations of $\{1, 2, 3, 4, 5\}$ are

$$123, \quad 124, \quad 125, \quad 134, \quad 135, \quad 145, \quad 234, \quad 235, \quad 245, \quad 345 \ .$$

The total number of k-combination of n objects, called a **binomial coefficient***, denoted $\binom{n}{k}$ and read "n choose k," can be obtained from $p_{n,k} = n(n-1)(n-2)\dots(n-k+1)$ and $k! := p_{k,k}$. Recall that $p_{n,k}$ is the number of ways to choose the first k objects from the set of n objects and arrange them in a row with regard to order. Since we want to disregard order and each k-combination appears exactly $p_{k,k}$ or $k!$ times among the $p_{n,k}$ many permutations, we perform a division:*

$$\binom{n}{k} := \frac{p_{n,k}}{p_{k,k}} = \frac{n(n-1)(n-2)\dots(n-k+1)}{k(k-1)(k-2)\dots 2\ 1} \ .$$

Binomial coefficients are often called "Pascal's Triangle" (Blaise Pascal, Traité du Triangle Arithmétique 1653), but they have many "fathers". There are earlier treatises of the binomial coefficients including Szu-yüan Yü-chien ("The Precious Mirror of the Four Elements") by the Chinese mathematician Chu Shih-Chieh in 1303, and in an ancient Hindu classic, Piṅgala's Chandaḥśāstra, due to Halāyudha (10-th century AD).

1.7 Array, Sequence, Limit, ...

In this section we will study a basic data structure in MATLAB called an **array** of numbers. Arrays are finite sequences and they can be processed easily in MATLAB. The notion of infinite sequences lead to **limits**, one of the most fundamental concepts in mathematics.

For any natural number n, we write

$$\boxed{\langle x_{1:n} \rangle := x_1, x_2, \ldots, x_{n-1}, x_n}$$

to represent the **finite sequence** of real numbers $x_1, x_2, \ldots, x_{n-1}, x_n$. For two integers m and n such that $m \leq n$, we write

$$\boxed{\langle x_{m:n} \rangle := x_m, x_{m+1}, \ldots, x_{n-1}, x_n}$$

to represent the **finite sequence** of real numbers $x_m, x_{m+1}, \ldots, x_{n-1}, x_n$. In mathematical analysis, finite sequences and their countably infinite counterparts play a fundamental role in limiting processes. Given an integer m, we denote an **infinite sequence** or simply a sequence as:

$$\boxed{\langle x_{m:\infty} \rangle := x_m, x_{m+1}, x_{m+2}, x_{m+3}, \ldots \ .}$$

Given index set $\mathcal{I}$ which may be finite or infinite in size, a sequence can either be seen as a set of ordered pairs:

$$\{(i, x_i) : i \in \mathcal{I}\} \ ,$$

or as a function that maps the index set to the set of real numbers:

$$x(i) = x_i : \mathcal{I} \mapsto \{x_i : i \in \mathcal{I}\} \ ,$$

The finite sequence $\langle x_{m:n} \rangle$ has $\mathcal{I} = \{m, m+1, m+2, m+3, \ldots, n\}$ as its index set while an infinite sequence $\langle x_{m:\infty} \rangle$ has $\mathcal{I} = \{m, m+1, m+2, m+3, \ldots\}$ as its index set. A **sub-sequence** $\langle x_{j:k} \rangle$ of a finite sequence $\langle x_{m:n} \rangle$ or an infinite sequence $\langle x_{m:\infty} \rangle$ is:

$$\langle x_{j:k} \rangle = x_j, x_{j+1}, \ldots, x_{k-1}, x_k \quad \text{where,} \quad m \leq j \leq k \leq n < \infty \ .$$

A rectangular arrangement of $m \cdot n$ real numbers in m rows and n columns is called an $m \times n$ **matrix**. The '$m \times n$' represents the **size** of the matrix. We use bold upper-case letters to denote matrices, for e.g:

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \ldots & x_{1,n-1} & x_{1,n} \\ x_{2,1} & x_{2,2} & \ldots & x_{2,n-1} & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m-1,1} & x_{m-1,2} & \ldots & x_{m-1,n-1} & x_{m-1,n} \\ x_{m,1} & x_{m,2} & \ldots & x_{m,n-1} & x_{m,n} \end{bmatrix}$$

Matrices with only one row or only one column are called **vectors**. An $1 \times n$ matrix is called a **row vector** since there is only one row and an $m \times 1$ matrix is called a **column vector** since there is only one column. We use bold-face lowercase letters to denote row and column vectors.

$$\text{A row vector } \mathbf{x} = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \end{bmatrix} = (x_1, x_2, \ldots, x_n)$$

$$\text{and a column vector } \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{m-1} \\ y_m \end{bmatrix} = \begin{bmatrix} y_1 & y_2 & \ldots & y_m \end{bmatrix}' = (y_1, y_2, \ldots, y_m)' \ .$$

The superscripting by $'$ is the transpose operation and simply means that the rows and columns are exchanged. Thus the transpose of the matrix $\mathbf{X}$ is:

$$\mathbf{X}' = \begin{bmatrix} x_{1,1} & x_{2,1} & \dots & x_{m-1,1} & x_{m,1} \\ x_{1,2} & x_{2,2} & \dots & x_{m-1,2} & x_{m,2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{1,n-1} & x_{2,n-1} & \dots & x_{m-1,n-1} & x_{m,n-1} \\ x_{1,n} & x_{2,n} & \dots & x_{m-1,n} & x_{m,n} \end{bmatrix}$$

In linear algebra and calculus, it is natural to think of vectors and matrices as points (ordered m-tuples) and ordered collection of points in Cartesian co-ordinates. We assume that the reader has heard of operations with matrices and vectors such as matrix multiplication, determinants, transposes, etc. Such concepts will be introduced as they are needed in the sequel.

Finite sequences, vectors and matrices can be represented in a computer by an elementary data structure called an **array**.

Labwork 2 (Sequences as arrays) *Let us learn to represent, visualise and operate finite sequences as* MATLAB *arrays. Try out the commands and read the comments for clarification.*

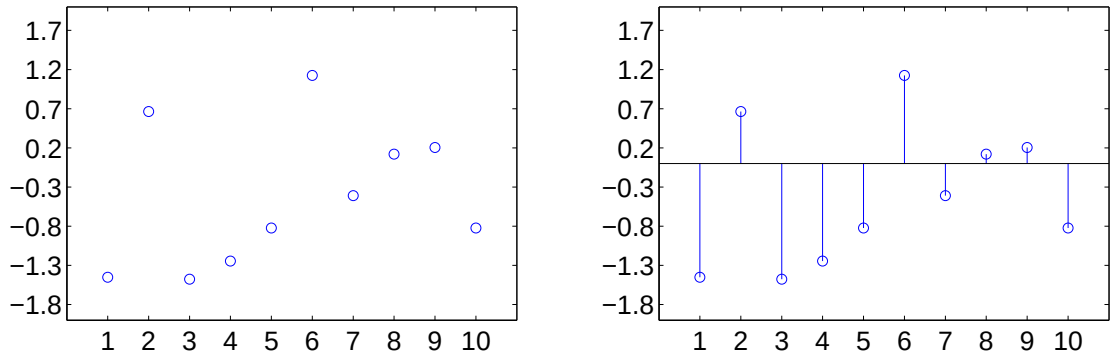
```
>> a = [17]                        % Declare the sequence of one element 17 in array a
a =    17
>> % Declare the sequence of 10 numbers in array b
>> b=[-1.4508 0.6636 -1.4768 -1.2455 -0.8235 1.1254 -0.4093 0.1199 0.2043 -0.8236]
b =
   -1.4508    0.6636   -1.4768   -1.2455   -0.8235    1.1254   -0.4093    0.1199    0.2043   -0.8236
   >> c = [1 2 3]                 % Declare the sequence of 3 consecutive numbers 1,2,3
z =     1     2     3
>> % linspace(x1, x2, n) generates n points linearly spaced between x1 and x2
>> r = linspace(1, 3, 3)                % Declare sequence r = c using linspace
r =     1     2     3
>> s1 = 1:10 % declare an array s1 starting at 1, ending by 10, in increments of 1
s =     1     2     3     4     5     6     7     8     9    10
>> s2 = 1:2:10  % declare an array s2 starting at 1, ending by 10, in increments of 2
s =     1     3     5     7     9
>> s2(3) % obtain the third element of the finite sequence s2
ans =     5
>> s2(2:4) % obtain the subsequence from second to fourth elements of the finite sequence s2
ans =     3     5     7
```

We may visualise (as per Figure 2) the finite sequences $\langle b_{1:n} \rangle$ stored in the array `b` *as the set of ordered pairs $\{(1, b_1), (2, b_2), \dots, (10, b_n)\}$ representing the function $b(i) = b_i : \{1, 2, \dots, n\} \mapsto \{b_1, b_2, \dots, b_n\}$ via* **point plot** *and* **stem plot** *using* `Matlab`*'s* `plot` *and* `stem` *commands, respectively.*

```
>> display(b) % display the array b in memory
b =
   -1.4508    0.6636   -1.4768   -1.2455   -0.8235    1.1254   -0.4093    0.1199    0.2043   -0.8236
>> plot(b,'o') % point plot of ordered pairs (1,b(1)), (2,b(2)), ..., (10,b(10))
>> stem(b) % stem plot of ordered pairs (1,b(1)), (2,b(2)), ..., (10,b(10))
>> plot(b,'-o') % point plot of ordered pairs (1,b(1)), (2,b(2)), ..., (10,b(10)) connected by lines
```

Labwork 3 (Vectors and matrices as arrays) *Let us learn to represent, visualise and operate vectors as* MATLAB *arrays. Syntactically, a vector is stored in an array exactly in the same way we stored a finite sequence. However, mathematically, we think of a vector as an ordered m-tuple that can be visualised as a point in Cartesian co-ordinates. Try out the commands and read the comments for clarification.*

Figure 1.5: Point plot and stem plot of the finite sequence $\langle b_{1:10} \rangle$ declared as an array.




```
>> a = [1 2]        % an 1 X 2 row vector
>> z = [1 2 3]         % Declare an 1 X 3 row vector z with three numbers
z =     1     2     3
>> % linspace(x1, x2, n) generates n points linearly spaced between x1 and x2
>> r = linspace(1, 3, 3)               % Declare an 1 X 3 row vector r = z using linspace
r =     1     2     3
>> c = [1; 2; 3]          % Declare a 3 X 1 column vector c with three numbers.  Semicolons delineate columns
c =
     1
     2
     3
>> rT = r'        % The column vector (1,2,3)' by taking the transpose of r via r'
rT =
     1
     2
     3
>> y = [1 1 1]                         % y is a sequence or row vector of 3 1's
y =     1     1     1
>> ones(1,10)        % ones(m,n) is an m X n matrix of ones.  Useful when m or n is large.
ans =     1     1     1     1     1     1     1     1     1     1
```

We can use two dimensional arrays to represent matrices. Some useful built-in commands to generate standard matrices are:

```
>> Z=zeros(2,10) % the 2 X 10 matrix of zeros
Z =
     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0
>> O=ones(4,5) % the 4 X 5 matrix of ones
O =
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
>> E=eye(4) % the 4 X 4 identity matrix
E =
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1
```

We can also perform operations with arrays representing vectors, finite sequences, or matrices.

```
>> y % the array y is
y =     1     1     1
>> z % the array z is
z =     1     2     3
>> x = y + z                            % x is the sum of vectors y and z (with same size 1 X 3)
x =     2     3     4
>> y = y * 2                             % y is updated to 2 * y (each term of y is multiplied by 2)
y =     2     2     2
>> p = z .* y                            % p is the vector obtained by term-by-term product of z and y
p =     2     4     6
>> d = z ./ y                             % d is the vector obtained by term-by-term division of z and y
d =    0.5000    1.0000    1.5000
>> t=linspace(-10,10,4)                 % t has 4 numbers equally-spaced between -10 and 10
t =  -10.0000   -3.3333    3.3333   10.0000
>> s = sin(t)                            % s is a vector obtained from the term-wise sin of the vector t
s =    0.5440    0.1906   -0.1906   -0.5440
>> sSq = sin(t) .^ 2         % sSq is an array obtained from term-wise squaring ( .^ 2) of the sin(t) array
sSq =    0.2960    0.0363    0.0363    0.2960
>> cSq = cos(t) .^ 2 % cSq is an array obtained from term-wise squaring ( .^ 2) of the cos(t) array
cSq =    0.7040    0.9637    0.9637    0.7040
>> sSq + cSq % we can add the two arrays sSq and cSq to get the array of 1's
ans =     1     1     1     1
>> n = sin(t) .^2 + cos(t) .^2         % we can directly do term-wise operation sin^2(t) + cos^2(t) of t as well
n =     1     1     1     1
>> t2 = (-10:6.666665:10)        % t2 is similar to t above but with ':' syntax of (start:increment:stop)
t2 =  -10.0000   -3.3333    3.3333   10.0000
```

Similarly, operations can be performed with matrices.

```
>>  (O+O) .^ (1/2) % term-by-term square root of the matrix obtained by adding O=ones(4,5) to itself
ans =
    1.4142    1.4142    1.4142    1.4142    1.4142
    1.4142    1.4142    1.4142    1.4142    1.4142
    1.4142    1.4142    1.4142    1.4142    1.4142
    1.4142    1.4142    1.4142    1.4142    1.4142
```

We can access specific rows or columns of a matrix as follows:

```
>> % declare a 3 X 3 array A of row vectors
>> A = [0.2760    0.4984    0.7513; 0.6797    0.9597    0.2551; 0.1626    0.5853    0.6991]
A =
    0.2760    0.4984    0.7513
    0.6797    0.9597    0.2551
    0.1626    0.5853    0.6991
>> A(2,:) % access the second row of A
ans =
    0.6797    0.9597    0.2551
>> B = A(2:3,:) % store the second and third rows of A in matrix B
B =
    0.6797    0.9597    0.2551
    0.1626    0.5853    0.6991
>> C = A(:,[1 3]) % store the first and third columns of A in matrix C
C =
    0.2760    0.7513
    0.6797    0.2551
```

Labwork 4 (Plotting a function as points of ordered pairs in two arrays) *Next we plot the function $sin(x)$ from several ordered pairs $(x_i, sin(x_i))$. Here x_i's are from the domain $[-2\pi, 2\pi]$.*

We use the `plot` *function in* MATLAB. *Create an M-file called* `MySineWave.m` *and copy the following commands in it. By entering* `MySineWave` *in the command window you should be able to run the script and see the figure in the figure window.*

SineWave.m

```
x = linspace(-2*pi,2*pi,100);        % x has 100 points equally spaced in [-2*pi, 2*pi]
y = sin(x);               % y is the term-wise sin of x, ie sin of every number in x is in y, resp.
plot(x,y,'.');                % plot x versus y as dots should appear in the Figure window
xlabel('x');               % label x-axis with the single quote enclosed string x
ylabel('sin(x)','FontSize',16);          % label y-axis with the single quote enclosed string
title('Sine Wave in [-2 pi, 2 pi]','FontSize',16);       % give a title; click Figure window to see changes
set(gca,'XTick',-8:1:8,'FontSize',16) % change the range and size of X-axis ticks
% you can go to the Figure window's File menu to print/save the plot
```

The plot was saved as an encapsulated postscript file from the File menu of the Figure window and is displayed below.

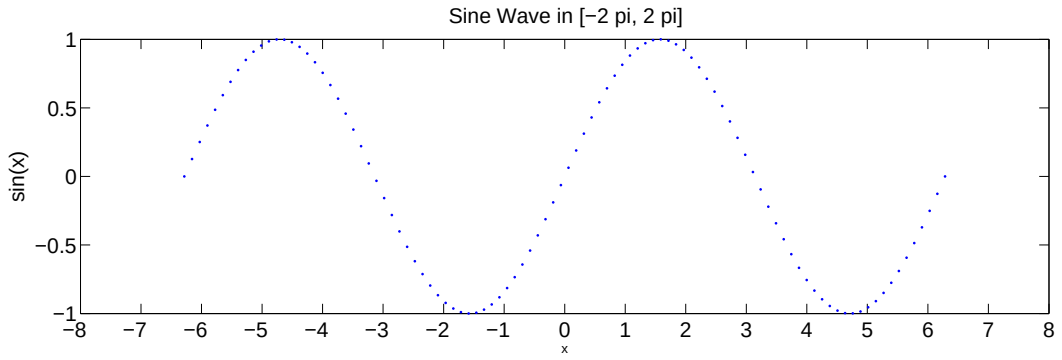


Figure 1.6: A plot of the sine wave over $[-2\pi, 2\pi]$

Let us first recall some elementary ideas from real analysis.

Definition 3 (Convergent sequence of real numbers) *A sequence of real numbers* $\langle x_i \rangle_{i=1}^{\infty} := x_1, x_2, \ldots$ *is said to converge to a limit* $a \in \mathbb{R}$ *and denoted by:*

$$\lim_{i \to \infty} x_i = a \ ,$$

if for every natural number $m \in \mathbb{N}$, *a natural number* $N_m \in \mathbb{N}$ *exists such that for every* $j \geq N_m$, $|x_j - a| \leq \frac{1}{m}$.

Example 1 *Let* $\langle x_i \rangle_{i=1}^{\infty} = 17, 17, 17, \ldots$. *Then* $\lim_{i \to \infty} x_i = 17$. *This is because for every* $m \in \mathbb{N}$, *we can take* $N_m = 1$ *and satisfy the definition of the limit, i.e.:*

$$\textit{for every } j \geq N_m = 1, \ |x_j - 17| = |17 - 17| = 0 \leq \frac{1}{m} \ .$$

Example 2 *Let* $\langle x_i \rangle_{i=1}^{\infty} = \frac{1}{1}, \frac{1}{2}, \frac{1}{3}, \ldots$, *i.e.* $x_i = \frac{1}{i}$, *then* $\lim_{i \to \infty} x_i = 0$. *This is because for every* $m \in \mathbb{N}$, *we can take* $N_m = m$ *and satisfy the definition of the limit, i.e.:*

$$\textit{for every } j \geq N_m = m, \ |x_j - 0| = \left| \frac{1}{j} - 0 \right| = \frac{1}{j} \leq \frac{1}{m} \ .$$

However, several other sequences also approach the limit 0. Some such sequences that approach the limit 0 from the right are:

$$\langle x_{1:\infty}\rangle = \frac{1}{1}, \frac{1}{4}, \frac{1}{9}, \dots \qquad \text{and} \qquad \langle x_{1:\infty}\rangle = \frac{1}{1}, \frac{1}{8}, \frac{1}{27}, \cdots \ ,$$

and some that approach the limit 0 from the left are:

$$\langle x_{1:\infty}\rangle = -\frac{1}{1}, -\frac{1}{2}, -\frac{1}{3}, \dots \qquad \text{and} \qquad \langle x_{1:\infty}\rangle = -\frac{1}{1}, -\frac{1}{4}, -\frac{1}{9}, \cdots \ ,$$

and finally some that approach 0 from either side are:

$$\langle x_{1:\infty}\rangle = -\frac{1}{1}, +\frac{1}{2}, -\frac{1}{3}, \dots \qquad \text{and} \qquad \langle x_{1:\infty}\rangle = -\frac{1}{1}, +\frac{1}{4}, -\frac{1}{9} \cdots \ .$$

When we do not particularly care about the specifics of a sequence of real numbers $\langle x_{1:\infty}\rangle$, in terms of the exact values it takes for each i, but we are only interested that it converges to a limit a we write:

$$x \to a$$

and say that x approaches a. If we are only interested in those sequences that converge to the limit a from the right or left, we write:

$$x \to a^+ \qquad \text{or} \qquad x \to a^-$$

and say x approaches a from the right or left, respectively.

Definition 4 (Limits of Functions) *We say a function $f(x) : \mathbb{R} \mapsto \mathbb{R}$ has a* **limit** *$L \in \mathbb{R}$ as x approaches a and write:*

$$\lim_{x \to a} f(x) = L \ ,$$

provided $f(x)$ is arbitrarily close to L for all values of x that are sufficiently close to, but not equal to, a. We say that f has a **right limit** *L_R or* **left limit** *L_L as x approaches a from the left or right, and write:*

$$\lim_{x \to a^+} f(x) = L_R \qquad \text{or} \qquad \lim_{x \to a^-} f(x) = L_L \ ,$$

provided $f(x)$ is arbitrarily close to L_R or L_L for all values of x that are sufficiently close to, but not equal to, a from the right of a or the left of a, respectively. When the limit is not an element of $\mathbb{R}$ or when the left and right limits are distinct, we say that the limit does not exist.

Example 3 *Consider the function $f(x) = \frac{1}{x^2}$. Then*

$$\lim_{x \to 1} f(x) = \lim_{x \to 1} \frac{1}{x^2} = 1$$

exists since the limit $1 \in \mathbb{R}$, and the right and left limits are the same:

$$\lim_{x \to 1^+} f(x) = \lim_{x \to 1^+} \frac{1}{x^2} = 1 \qquad \text{and} \qquad \lim_{x \to 1^-} f(x) = \lim_{x \to 1^-} \frac{1}{x^2} = 1 \ .$$

However, the following limit does not exist:

$$\lim_{x \to 0} f(x) = \lim_{x \to 0} \frac{1}{x^2} = \infty$$

since $\infty \notin \mathbb{R}$.

Example 4 *The limit of* $f(x) = (1+x)^{\frac{1}{x}}$ *as* x *approaches* 0 *exists and it is the Euler's constant* e *:*

$$\lim_{x\to 0} f(x) = \lim_{x\to 0}(1+x)^{\frac{1}{x}} = e \approxeq 2.71828 \ .$$

Notice that the above limit exists despite the fact that $f(0) = (1+0)^{\frac{1}{0}}$ *itself is undefined and does not exist.*

Example 5 *For* $f(x) = \frac{x^3-1}{x-1}$*, this limit exists:*

$$\lim_{x\to 1} f(x) = \lim_{x\to 1} \frac{x^3-1}{x-1} = \lim_{x\to 1} \frac{(x-1)(x^2+x+1)}{(x-1)} = \lim_{x\to 1} x^2+x+1 = 3$$

despite the fact that $f(1) = \frac{1^3-1}{1-1} = \frac{0}{0}$ *itself is undefined and does not exist.*

Next we look at some examples of limits at infinity.

Example 6 *The limit of* $f(n) = \left(1-\frac{\lambda}{n}\right)^n$ *as* n *approaches* ∞ *exists and it is* $e^{-\lambda}$ *:*

$$\lim_{n\to\infty} f(n) = \lim_{n\to\infty} \left(1-\frac{\lambda}{n}\right)^n = e^{-\lambda} \ .$$

Example 7 *The limit of* $f(n) = \left(1-\frac{\lambda}{n}\right)^{-\alpha}$*, for some* $\alpha > 0$*, as* n *approaches* ∞ *exists and it is* 1 *:*

$$\lim_{n\to\infty} f(n) = \lim_{n\to\infty} \left(1-\frac{\lambda}{n}\right)^{-\alpha} = 1 \ .$$

Definition 5 (Continuity of a function) *We say a real-valued function* $f(x) : D \mapsto \mathbb{R}$ *with the domain* $D \subset \mathbb{R}$ *is* **right continuous** *or* **left continuous** *at a point* $a \in D$*, provided:*

$$\lim_{x\to a^+} f(x) = f(a) \qquad or \qquad \lim_{x\to a^-} f(x) = f(a) \ ,$$

respectively. We say f *is* **continuous** *at* $a \in D$*, provided:*

$$\lim_{x\to a^+} f(x) = f(a) = \lim_{x\to a^-} f(x) \ .$$

Finally, f *is said to be continuous if* f *is continuous at every* $a \in D$*.*

Example 8 *Let us reconsider the function* $f(x) = (1+x)^{\frac{1}{x}} : \mathbb{R} \mapsto \mathbb{R}$*. Clearly,* $f(x)$ *is continuous at* 1*, since:*

$$\lim_{x\to 1} f(x) = \lim_{x\to 1}(1+x)^{\frac{1}{x}} = 2 = f(1) = (1+1)^{\frac{1}{1}} \ ,$$

but it is not continuous at 0*, since:*

$$\lim_{x\to 0} f(x) = \lim_{x\to 0}(1+x)^{\frac{1}{x}} = e \approxeq 2.71828 \neq f(0) = (1+0)^{\frac{1}{0}} \ .$$

Thus, $f(x)$ *is not a continuous function over* $\mathbb{R}$*.*

1.8 Elementary Number Theory

We introduce basic notions that we need from elementary number theory here. These notions include integer functions and modular arithmetic as they will be needed later on.

For any real number x:

$\lfloor x \rfloor := \max\{y : y \in \mathbb{Z} \text{ and } y \leq x\}$, i.e., the greatest integer less than or equal to x (the **floor** of x),

$\lceil x \rceil := \min\{y : y \in \mathbb{Z} \text{ and } y \geq x\}$, i.e., the least integer greater than or equal to x (the **ceiling** of x).

Example 9

$$\lfloor 1 \rfloor = 1, \quad \lceil 1 \rceil = 1, \quad \lfloor 17.8 \rfloor = 17, \quad \lfloor -17.8 \rfloor = -18, \quad \lceil \sqrt{2} \rceil = 2, \quad \lfloor \pi \rfloor = 3, \quad \lceil \frac{1}{10^{100}} \rceil = 1.$$

Labwork 5 *We can use* MATLAB *functions* `floor` *and* `ceil` *to compute* $\lfloor x \rfloor$ *and* $\lceil x \rceil$*, respectively. Also, the argument* x *to these functions can be an array.*

```
>> sqrt(2) % the square root of 2 is
ans =    1.4142
>> ceil(sqrt(2)) % ceiling of square root of 2
ans =     2
>> floor(-17.8) % floor of -17.8
ans =   -18
>> ceil([1 sqrt(2) pi -17.8 1/(10^100)]) %the ceiling of each element of an array
ans =     1     2     4   -17     1
>> floor([1 sqrt(2) pi -17.8 1/(10^100)]) % the floor of each element of an array
ans =     1     1     3   -18     0
```

Classwork 4 *Convince yourself of the following formulae. Use examples, plots and/or formal arguments.*

$$\begin{aligned} \lceil x \rceil = \lfloor x \rfloor &\iff x \in \mathbb{Z} \\ \lceil x \rceil = \lfloor x \rfloor + 1 &\iff x \notin \mathbb{Z} \\ \lfloor -x \rfloor &= -\lceil x \rceil \\ x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil &< x + 1 \end{aligned}$$

Let us define modular arithmetic next. Suppose x and y are any real numbers, i.e. $x, y \in \mathbb{R}$, we define the binary operation called "$x \mod y$" as:

$$x \mod y := \begin{cases} x - y\lfloor x/y \rfloor & \text{if } y \neq 0 \\ x & \text{if } y = 0 \end{cases}$$

Chapter 2

Probability Model

The mathematical model for probability or the probability model is an axiomatic system that may be motivated by the intuitive idea of 'long-term relative frequency'. If the axioms and definitions are intuitively motivated, the probability model simply follows from the application of logic to these axioms and definitions. No attempt to define probability in the real world is made. However, the application of probability models to real-world problems through statistical experiments has a fruitful track record. In fact, you are here for exactly this reason.

2.1 Probability

Idea 1 (The long-term relative frequency (LTRF) idea) *Suppose we are interested in the fairness of a coin, i.e. if landing Heads has the same "probability" as landing Tails. We can toss it n times and call $N(\mathtt{H}, n)$ the fraction of times we observed Heads out of n tosses. Suppose that after conducting the tossing experiment* 1000 *times, we rarely observed Heads, e.g.* 9 *out of the* 1000 *tosses, then $N(\mathtt{H}, 1000) = 9/1000 = 0.009$. Suppose we continued the number of tosses to a million and found that this number approached closer to* 0.1*, or, more generally, $N(\mathtt{H}, n) \to 0.1$ as $n \to \infty$. We might, at least intuitively, think that the coin is unfair and has a lower "probability" of* 0.1 *of landing Heads. We might think that it is fair had we observed $N(\mathtt{H}, n) \to 0.5$ as $n \to \infty$. Other crucial assumptions that we have made here are:*

1. **Something Happens***: Each time we toss a coin, we are certain to observe Heads* **or** *Tails, denoted by $\mathtt{H} \cup \mathtt{T}$. The probability that "something happens" is* 1*. More formally:*

$$N(\mathtt{H} \cup \mathtt{T}, n) = \frac{n}{n} = 1.$$

   This is an intuitively reasonable assumption that simply says that one of the possible outcomes is certain to occur, provided the coin is not so thick that it can land on or even roll along its circumference.

2. **Addition Rule***: Heads and Tails are mutually exclusive events in any given toss of a coin, i.e. they cannot occur simultaneously. The intersection of mutually exclusive events is the empty set and is denoted by $\mathtt{H} \cap \mathtt{T} = \emptyset$. The event $\mathtt{H} \cup \mathtt{T}$, namely that the event that "coin lands Heads* **or** *coin lands Tails" satisfies:*

$$N(\mathtt{H} \cup \mathtt{T}, n) = N(\mathtt{H}, n) + N(\mathtt{T}, n).$$

3. *The coin-tossing experiment is repeatedly performed in an* **independent** *manner, i.e. the outcome of any individual coin-toss does not affect that of another. This is an intuitively reasonable assumption since the coin has no memory and the coin is tossed identically each time.*

We will use the LTRF idea more generally to motivate a mathematical model of probability called probability model. Suppose A is an event associated with some experiment $\mathcal{E}$, so that A either does or does not occur when the experiment is performed. We want the probability that event A occurs in a specific performance of $\mathcal{E}$, denoted by $\mathbf{P}(A)$, to intuitively mean the following: if one were to perform a super-experiment $\mathcal{E}^\infty$ by independently repeating the experiment $\mathcal{E}$ and recording $N(A, n)$, the fraction of times A occurs in the first n performances of $\mathcal{E}$ within the super-experiment $\mathcal{E}^\infty$. Then the LTRF idea suggests:

$$N(A,n) := \frac{\text{Number of times } A \text{ occurs}}{n = \text{Number of performances of } \mathcal{E}} \to \mathbf{P}(A), \ as \quad n \to \infty \tag{2.1}$$

First, we need some definitions. We will set the scene for them with the following example.

Experiment 1 (The Bernoulli Experiment $\mathcal{E}_\theta^n$; Toss a coin n times) *Suppose our experiment entails tossing a coin n times and recording* `H` *for Heads and* `T` *for Tails. When $n = 3$, one possible outcome of this experiment is* `HHT`*, ie. a Head followed by another Head and then a Tail. Seven other outcomes are possible. Below, we refer to this experiment by the symbol $\mathcal{E}_\theta^3$. More generally, we refer to the experiment of tossing a coin n times as $\mathcal{E}_\theta^n$ and sometimes refer to $\mathcal{E}_\theta^1$ by $\mathcal{E}_\theta$ for simplicity. The reason for the θ subscrip will become apparent as we develop the theory.*

Definition 6 (Sample space, sample point or outcome and event) *The* **sample space** *is the set of all possible outcomes of an experiment. It is denoted by Ω (the Greek upper-case letter Omega). A particular element of Ω is called a* **sample point** *or* **outcome** *generally denoted by ω (the Greek lower-case omega), and a sequence of possibly distinct outcomes is generally denoted by $\omega_1, \omega_2, \ldots$. An* **event** *is a (measurable) subset of the sample space.*

The sample space for "toss a coin three times" experiment $\mathcal{E}_\theta^3$ is:

$$\Omega = \{\mathtt{H}, \mathtt{T}\}^3 = \{\mathtt{HHH}, \mathtt{HHT}, \mathtt{HTH}, \mathtt{HTT}, \mathtt{THH}, \mathtt{THT}, \mathtt{TTH}, \mathtt{TTT}\} \ ,$$

with a particular sample point or outcome $\omega = \mathtt{HTH}$, and another distinct outcome $\omega' = \mathtt{HHH}$. An event, say A, that 'at least two Heads occur' is the following subset of Ω:

$$A = \{\mathtt{HHH}, \mathtt{HHT}, \mathtt{HTH}, \mathtt{THH}\} \ .$$

Another event, say B, that 'no Heads occur' is:

$$B = \mathtt{TTT}$$

Note that the event B is also an outcome or sample point. Another interesting event is the empty set $\emptyset \subset \Omega$. The event that 'nothing in the sample space occurs' is $\emptyset$.

Classwork 5 *Can you think of a graphical way to enumerate the outcomes of the $\mathcal{E}_\theta^3$? Draw a diagram of this under the caption of Figure 2.1, using the caption as a hint (in other words, draw your own Figure 2.1).*

Figure 2.1: A binary tree whose leaves are all possible outcomes.

Algorithm 1 List Ω for "Toss a Coin Three Times" experiment $\mathcal{E}_\theta^3$

input: nothing
output: print/list all outcomes of $\mathcal{E}_\theta^3$
initialize: **SampleSpace1Toss** $= \{\mathtt{H}, \mathtt{T}\}$ $\{$**SampleSpace1Toss**[1] = H and **SampleSpace1Toss**[2] = T$\}$
for $i = 1$ to 2 **do**
  for $j = 1$ to 2 **do**
    for $k = 1$ to 2 **do**
      Print **SampleSpace1Toss**$[i]$ **SampleSpace1Toss**$[j]$ **SampleSpace1Toss**$[k]$
      Print " , " $\{$print a comma character to delimit outcomes$\}$
    end for
  end for
end for

In Labwork 6 we implement Algorithm 1 to print all the outcomes. The algorithm uses **for loops** to reach the leaves (outcomes of $\mathcal{E}_\theta^3$) of the binary tree.

Labwork 6 *Let's write a* MATLAB *code in a script file named* `OutcomesOf3Tosses.m` *that implements Algorithm 1 to print all the outcomes of* $\mathcal{E}_\theta^3$. *You need to go to the File menu and create a new file named* `OutcomesOf3Tosses.m` *Run it in the command window.*

```
>> type OutcomesOf3Tosses.m

SampleSpace1Toss='HT';        % declare a string vector or character array
% SampleSpace1Toss is the name of the char array
SampleSpace1Toss(1);                % access the first element 'H' this way
SampleSpace1Toss(2);                % access the second element 'T' this way
% Now let's write the routine for listing the sample space of 'toss 3 times'
w=' ';                % declare w to be the character ' '
for i = 1:1:2                % for loop for variable i = start:increment:end
  for j = 1:1:2                % for loop for variable j
    for k = 1:1:2        % for loop for variable j
     % next we concatenate using strcat -- strcat('A','B','C','D') concatenates the 4 char arrays
     x = strcat(SampleSpace1Toss(i),SampleSpace1Toss(j),SampleSpace1Toss(k),' , '); % ' ,' delimited outcome
     w = strcat(w,x); % recursively store the outcomes in a new array w
    end
  end
end
w               % print w at the end of the three for loops
>> lab1work4
>> SampleSpace1Toss(1)
ans = H
>> SampleSpace1Toss(2)
ans = T
```

```
>> w
w = HHH ,HHT ,HTH ,HTT ,THH ,THT ,TTH ,TTT ,
```

Now, we are finally ready to define probability.

Definition 7 (Probability) *Let $\mathcal{E}$ be an experiment with sample space Ω. Let $\mathcal{F}$ denote a suitable collection of events in Ω that satisfy the following conditions:*

1. *It (the collection) contains the sample space:* $\boxed{\Omega \in \mathcal{F}}$.
2. *It is closed under complementation:* $\boxed{A \in \mathcal{F} \quad \Longrightarrow \quad A^c \in \mathcal{F}}$.
3. *It is closed under countable unions:* $\boxed{A_1, A_2, \ldots \in \mathcal{F} \quad \Longrightarrow \quad \bigcup_i A_i := A_1 \cup A_2 \cup \cdots \in \mathcal{F}}$.

Formally, this collection of events is called a **sigma field** *or a* **sigma algebra**. *Our experiment $\mathcal{E}$ has a sample space Ω and a collection of events $\mathcal{F}$ that satisfy the three condition.*

Given a double, e.g. $(\Omega, \mathcal{F})$, **probability** *is just a function $\mathbf{P}$ which assigns each event $A \in \mathcal{F}$ a number $\mathbf{P}(A)$ in the real interval $[0, 1]$, i.e.* $\boxed{\mathbf{P} : \mathcal{F} \mapsto [0, 1]}$, *such that:*

1. *The 'Something Happens' axiom holds, i.e.* $\boxed{\mathbf{P}(\Omega) = 1}$.
2. *The 'Addition Rule' axiom holds, i.e. for events A and B:*

$$\boxed{A \cap B = \emptyset \quad \Longrightarrow \quad \mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B)}\ .$$

2.1.1 Consequences of our Definition of Probability

It is important to realize that we accept the 'addition rule' as an axiom in our mathematical definition of probability (or our probability model) and we do **not** prove this rule. However, the facts which are stated (with proofs) below, are logical consequences of our definition of probability:

1. For any event A, $\boxed{\mathbf{P}(A^c) = 1 - \mathbf{P}(A)}$.

   Proof: One line proof.

$$\overbrace{\mathbf{P}(A) + \mathbf{P}(A^c)}^{LHS} \underbrace{=}_{+\text{ rule } \because A \cap A^c = \emptyset} \mathbf{P}(A \cup A^c) \underbrace{=}_{A \cup A^c = \Omega} \mathbf{P}(\Omega) \underbrace{=}_{\because \mathbf{P}(\Omega) = 1} \overbrace{1}^{RHS} \underbrace{\Longrightarrow}_{LHS - \mathbf{P}(A)\ \&\ RHS - \mathbf{P}(A)} \mathbf{P}(A^c) = 1 - \mathbf{P}(A)$$

   - If $A = \Omega$ then $A^c = \Omega^c = \emptyset$ and $\boxed{\mathbf{P}(\emptyset) = 1 - \mathbf{P}(\Omega) = 1 - 1 = 0}$.

2. For any two events A and B, we have the **inclusion-exclusion principle**:

$$\boxed{\mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B) - \mathbf{P}(A \cap B)}.$$

Proof: Since:

$$\begin{aligned} A &= (A \setminus B) \cup (A \cap B) \qquad \text{and} \qquad (A \setminus B) \cap (A \cap B) = \emptyset, \\ A \cup B &= (A \setminus B) \cup B \qquad \text{and} \qquad (A \setminus B) \cap B = \emptyset \end{aligned}$$

the addition rule implies that:

$$\begin{aligned} \mathbf{P}(A) &= \mathbf{P}(A \setminus B) + \mathbf{P}(A \cap B) \\ \mathbf{P}(A \cup B) &= \mathbf{P}(A \setminus B) + \mathbf{P}(B) \end{aligned}$$

Substituting the first equality above into the second, we get:

$$\mathbf{P}(A \cup B) = \mathbf{P}(A \setminus B) + \mathbf{P}(B) = \mathbf{P}(A) - \mathbf{P}(A \cap B) + \mathbf{P}(B)$$

3. For a sequence of mutually disjoint events $A_1, A_2, A_3, \ldots, A_n$:

$$\boxed{A_i \cap A_j = \emptyset \quad \text{for any } i,j \quad \Longrightarrow \quad \mathbf{P}(A_1 \cup A_2 \cup \cdots \cup A_n) = \mathbf{P}(A_1) + \mathbf{P}(A_2) + \cdots + \mathbf{P}(A_n)}.$$

Proof: If A_1, A_2, A_3 are mutually disjoint events, then $A_1 \cup A_2$ is disjoint from A_3. Thus, two applications of the addition rule for disjoint events yields:

$$\mathbf{P}(A_1 \cup A_2 \cup A_3) = \mathbf{P}((A_1 \cup A_2) \cup A_3) \underbrace{=}_{+ \text{ rule}} \mathbf{P}(A_1 \cup A_2) + \mathbf{P}(A_3) \underbrace{=}_{+ \text{ rule}} \mathbf{P}(A_1) + \mathbf{P}(A_2) + \mathbf{P}(A_3)$$

The n-event case follows by mathematical induction.

We have formally defined the **probability model** specified by the **probability triple** $(\Omega, \mathcal{F}, \mathbf{P})$ that can be used to model an **experiment** $\mathcal{E}$.

Next, let us take a detour into how one might interpret it in the real world. The following is an adaptation from Williams D, *Weighing the Odds: A Course in Probability and Statistics*, Cambridge University Press, 2001, which henceforth is abbreviated as WD2001.

Probability Model	**Real-world Interpretation**
Sample space Ω	Set of all outcomes of an experiment
Sample point ω	Possible outcome of an experiment
(No counterpart)	Actual outcome $\omega^\star$ of an experiment
Event A, a (suitable) subset of Ω	The real-world event corresponding to A occurs if and only if $\omega^\star \in A$
$\mathbf{P}(A)$, a number between 0 and 1	Probability that A will occur for an experiment yet to be performed

Events in Probability Model	**Real-world Interpretation**
Sample space Ω	The certain even 'something happens'
The $\emptyset$ of Ω	The impossible event 'nothing happens'
The intersection $A \cap B$	'Both A and B occur'
$A_1 \cap A_2 \cap \cdots \cap A_n$	'All of the events $A_1, A_2, \ldots, A_n$ occur simultaneously'
The union $A \cup B$	'At least one of A and B occurs'
$A_1 \cup A_2 \cup \cdots \cup A_n$	'At least one of the events $A_1, A_2, \ldots, A_n$ occurs'
A^c, the complement of A	'A does not occur'
$A \setminus B$	'A occurs, but B does not occur'
$A \subset B$	'If A occurs, then B must occur'

Example 10 *Consider the 'Toss a coin once' experiment $\mathcal{E}_\theta^1$ (or simply $\mathcal{E}_\theta$). What is its sample space Ω and a reasonable collection of events $\mathcal{F}$ that underpin this experiment ?*

$$\Omega = \{\mathtt{H}, \mathtt{T}\}, \qquad \mathcal{F} = \{\mathtt{H}, \mathtt{T}, \Omega, \emptyset\} \ ,$$

A function that will satisfy the definition of probability for this collection of events $\mathcal{F}$ and assign $\mathbf{P}(\mathtt{H}) = \theta$ is summarized below. First assume that the above $\mathcal{F}$ is a sigma-algebra. Draw a picture for $\mathbf{P}$ with arrows that map elements in the domain $\mathcal{F}$ given above to elements in its range.

Event $A \in \mathcal{F}$	$\mathbf{P} : \mathcal{F} \mapsto [0,1]$	$\mathbf{P}(A) \in [0,1]$
$\Omega = \{\mathtt{H}, \mathtt{T}\}\bullet$	$\longrightarrow$	1
$\mathtt{T}\bullet$	$\longrightarrow$	$1-\theta$
$\mathtt{H}\bullet$	$\longrightarrow$	θ
$\emptyset\bullet$	$\longrightarrow$	0

Classwork 6 *Note that $\mathcal{F}' = \{\Omega, \emptyset\}$ is also a sigma algebra of the sample space $\Omega = \{\mathtt{H}, \mathtt{T}\}$. Can you think of a probability for the collection $\mathcal{F}'$?*

Event $A \in \mathcal{F}'$	$P : \mathcal{F}' \mapsto [0,1]$	$\mathbf{P}(A) \in [0,1]$
$\Omega = \{\mathtt{H}, \mathtt{T}\}\bullet$	$\longrightarrow$	
$\emptyset\bullet$	$\longrightarrow$	

Thus, $\mathcal{F}$ and $\mathcal{F}'$ are two distinct sigma algebras over our $\Omega = \{\mathtt{H}, \mathtt{T}\}$. Moreover, $\mathcal{F}' \subset \mathcal{F}$ and is called a sub sigma algebra. Try to show that $\{\Omega, \emptyset\}$ is the smallest possible sigma algebra over all possible sigma algebras over any given sample space Ω (think of intersecting an arbitrary family of sigma algebras)?

2.2 Conditional Probability

Next, we define conditional probability and the notion of independence of events. We use the LTRF idea to motivate the definition.

Idea 2 (LTRF intuition for conditional probability) *Let A and B be any two events associated with our experiment $\mathcal{E}$ with $\mathbf{P}(A) \neq 0$. The 'conditional probability that B occurs given that A occurs' denoted by $\mathbf{P}(B|A)$ is again intuitively underpinned by the super-experiment $\mathcal{E}^\infty$ which is the 'independent' repitition of our original experiment $\mathcal{E}$ 'infinitely' often. The LTRF idea is that $\mathbf{P}(B|A)$ is the long-term proportion of those experiments on which A occurs that B also occurs.*

Recall that $N(A,n)$ as defined in (2.1) is the fraction of times A occurs out of n independent repetitions of our experiment $\mathcal{E}$ (ie. the experiment $\mathcal{E}^n$). If $A \cap B$ is the event that 'A and B occur simultaneously', then we intuitively want

$$\mathbf{P}(B|A) \quad \text{“}\rightarrow\text{”} \quad \frac{N(A\cap B, n)}{N(A,n)} = \frac{N(A\cap B, n)/n}{N(A,n)/n} = \frac{\mathbf{P}(A\cap B)}{\mathbf{P}(A)}$$

as our $\mathcal{E}^n \rightarrow \mathcal{E}^\infty$. So, we **define** *conditional probability as we want.*

Definition 8 (Conditional Probability) *Suppose we are given an experiment $\mathcal{E}$ with a triple $(\Omega, \mathcal{F}, P)$. Let A and B be events, ie. $A, B \in \mathcal{F}$, such that $\mathbf{P}(A) \neq 0$. Then, we define the* **conditional probability** *of B given A by,*

$$\mathbf{P}(B|A) := \frac{\mathbf{P}(A\cap B)}{\mathbf{P}(A)} \ . \tag{2.2}$$

Note that for a **fixed** *event $A \in \mathcal{F}$ with $\mathbf{P}(A) > 0$ and* **any** *event $B \in \mathcal{F}$, the conditional probability $\mathbf{P}(B|A)$ is a probability as in Definition 7, ie. a function:*

$$\mathbf{P}(B|A) : \mathcal{F} \to [0, 1]$$

that assigns to each $B \in \mathcal{F}$ a number in the interval $[0, 1]$, such that,

1. $\mathbf{P}(\Omega|A) = 1$ *Meaning 'Something Happens given the event A happens'*
2. *The 'Addition Rule' axiom holds, ie. for events $B_1, B_2 \in \mathcal{F}$,*

$$B_1 \cap B_2 = \emptyset \quad \textit{implies} \quad \mathbf{P}(B_1 \cup B_2|A) = \mathbf{P}(B_1|A) + \mathbf{P}(B_2|A) \ .$$

Example 11 (Wasserman03, p. 11) *A medical test for a disease D has outcomes $+$ and $-$. the probabilities are:*

	Have Disease (D)	*Don't have disease (D^c)*
Test positive ($+$)	*0.009*	*0.099*
Test negative ($-$)	*0.001*	*0.891*

Using the definition of conditional probability, we can compute the conditional probability that you test positive given that you have the disease:

$$\mathbf{P}(+|D) = \frac{\mathbf{P}(+ \cap D)}{\mathbf{P}(D)} = \frac{0.009}{0.009 + 0.001} = 0.9 \ ,$$

and the conditional probability that you test negative given that you don't have the disease:

$$\mathbf{P}(-|D^c) = \frac{\mathbf{P}(- \cap D^c)}{\mathbf{P}(D^c)} = \frac{0.891}{0.099 + 0.891} \approxeq 0.9 \ .$$

Thus, the test is quite accurate since sick people test positive 90% of the time and healthy people test negative 90% of the time.

Classwork 7 *Now, suppose you go for a test and and test positive. What is the probability that you have the disease ?*

$$\mathbf{P}(D|+) = \frac{\mathbf{P}(D \cap +)}{\mathbf{P}(+)} = \frac{0.009}{0.009 + 0.099} \approxeq 0.08$$

Most people who are not used to the definition of conditional probability would intuitively associate a number much bigger than 0.08 for the answer. Interpret conditional probability in terms of the meaning of the numbers that appear in the numerator and denominator of the above calculations.

Next we look at one of the most elegant applications of the definition of conditional probability along with the addition rule for a partition of Ω.

Proposition 1 (Bayes' Theorem, 1763) *Suppose the events $A_1, A_2, \ldots, A_k \in \mathcal{F}$, with $\mathbf{P}(A_h) > 0$ for each $h \in \{1, 2, \ldots, k\}$, partition the sample space Ω, ie. they are mutually exclusive (disjoint) and exhaustive events with positive probability:*

$$A_i \cap A_j = \emptyset, \ \textit{for any distinct } i, j \in \{1, 2, \ldots, k\}, \qquad \bigcup_{h=1}^{k} A_h = \Omega, \qquad \mathbf{P}(A_h) > 0$$

Thus, precisely one of the A_h's will occur on any performance of our experiment $\mathcal{E}$.

Let $B \in \mathcal{F}$ be some event with $\mathbf{P}(B) > 0$, then

$$\mathbf{P}(A_h|B) = \frac{\mathbf{P}(B|A_h)\mathbf{P}(A_h)}{\sum_{h=1}^{k} \mathbf{P}(B|A_h)\mathbf{P}(A_h)} \tag{2.3}$$

Proof: *We apply elementary set theory, the definition of conditional probability $k+2$ times and the addition rule once:*

$$\begin{aligned}
\mathbf{P}(A_h|B) &= \frac{\mathbf{P}(A_h \cap B)}{\mathbf{P}(B)} = \frac{\mathbf{P}(B \cap A_h)}{\mathbf{P}(B)} = \frac{\mathbf{P}(B|A_h)\mathbf{P}(A_h)}{\mathbf{P}(B)} \\
&= \frac{\mathbf{P}(B|A_h)\mathbf{P}(A_h)}{P\left(\bigcup_{h=1}^{k}(B \cap A_h)\right)} = \frac{\mathbf{P}(B|A_h)\mathbf{P}(A_h)}{\sum_{h=1}^{k} P\left(B \cap A_h\right)} \\
&= \frac{\mathbf{P}(B|A_h)\mathbf{P}(A_h)}{\sum_{h=1}^{k} \mathbf{P}(B|A_h)\mathbf{P}(A_h)}
\end{aligned}$$

The operations done to the denominator in the proof above:

$$\mathbf{P}(B) = \sum_{h=1}^{k} \mathbf{P}(B|A_h)\mathbf{P}(A_h) \tag{2.4}$$

is also called 'the law of total probability'. We call $\mathbf{P}(A_h)$ the **prior probability of** *A_h and $\mathbf{P}(A_h|B)$ the* **posterior probability of** *A_h.*

Example 12 (Wasserman2003 p.12) *Suppose Larry divides his email into three categories: $A_1 =$ "spam", $A_2 =$ "low priority", and $A_3 =$ "high priority". From previous experience, he finds that $\mathbf{P}(A_1) = 0.7$, $\mathbf{P}(A_2) = 0.2$ and $\mathbf{P}(A_3) = 0.1$. Note that $\mathbf{P}(A_1 \cup A_2 \cup A_3) = \mathbf{P}(\Omega) = 0.7 + 0.2 + 0.1 = 1$. Let B be the event that the email contains the word "free." From previous experience, $\mathbf{P}(B|A_1) = 0.9$, $\mathbf{P}(B|A_2) = 0.01$ and $\mathbf{P}(B|A_3) = 0.01$. Note that $\mathbf{P}(B|A_1) + \mathbf{P}(B|A_2) + \mathbf{P}(B|A_3) = 0.9 + 0.01 + 0.01 \neq 1$. Now, suppose Larry receives an email with the word "free." What is the probability that it is "spam," "low priority," and "high priority" ?*

$$\begin{aligned}
\mathbf{P}(A_1|B) &= \frac{\mathbf{P}(B|A_1)\mathbf{P}(A_1)}{\mathbf{P}(B|A_1)\mathbf{P}(A_1)+\mathbf{P}(B|A_2)\mathbf{P}(A_2)+\mathbf{P}(B|A_3)\mathbf{P}(A_3)} = \frac{0.9\times0.7}{(0.9\times0.7)+(0.01\times0.2)+(0.01\times0.1)} = \frac{0.63}{0.633} \approxeq 0.995 \\
\mathbf{P}(A_2|B) &= \frac{\mathbf{P}(B|A_2)\mathbf{P}(A_2)}{\mathbf{P}(B|A_1)\mathbf{P}(A_1)+\mathbf{P}(B|A_2)\mathbf{P}(A_2)+\mathbf{P}(B|A_3)\mathbf{P}(A_3)} = \frac{0.01\times0.2}{(0.9\times0.7)+(0.01\times0.2)+(0.01\times0.1)} = \frac{0.002}{0.633} \approxeq 0.003 \\
\mathbf{P}(A_3|B) &= \frac{\mathbf{P}(B|A_3)\mathbf{P}(A_3)}{\mathbf{P}(B|A_1)\mathbf{P}(A_1)+\mathbf{P}(B|A_2)\mathbf{P}(A_2)+\mathbf{P}(B|A_3)\mathbf{P}(A_3)} = \frac{0.01\times0.1}{(0.9\times0.7)+(0.01\times0.2)+(0.01\times0.1)} = \frac{0.001}{0.633} \approxeq 0.002
\end{aligned}$$

Note that $\mathbf{P}(A_1|B) + \mathbf{P}(A_2|B) + \mathbf{P}(A_3|B) = 0.995 + 0.003 + 0.002 = 1$.

2.2.1 Independence and Dependence

Definition 9 (Independence of two events) *Any two events A and B are said to be* **independent** *if and only if*

$$\mathbf{P}(A \cap B) = \mathbf{P}(A)\mathbf{P}(B) \ . \tag{2.5}$$

Let us make sense of this definition in terms of our previous definitions. When $\mathbf{P}(A) = 0$ or $\mathbf{P}(B) = 0$, both sides of the above equality are 0. If $\mathbf{P}(A) \neq 0$, then rearranging the above equation we get:

$$\frac{\mathbf{P}(A \cap B)}{\mathbf{P}(A)} = \mathbf{P}(B) \ .$$

But, the LHS is $\mathbf{P}(B|A)$ by definition 2.2, and thus for independent events A and B, we get:

$$\mathbf{P}(B|A) = \mathbf{P}(B) \ .$$

This says that information about the occurrence of A does not affect the occurrence of B. If $\mathbf{P}(B) \neq 0$, then an analogous argument:

$$\mathbf{P}(A \cap B) = \mathbf{P}(A)\mathbf{P}(B) \iff \mathbf{P}(B \cap A) = \mathbf{P}(A)\mathbf{P}(B) \iff \frac{\mathbf{P}(B \cap A)}{\mathbf{P}(B)} = \mathbf{P}(A) \iff \mathbf{P}(A|B) = \mathbf{P}(A) \ ,$$

says that information about the occurrence of B does not affect the occurrence of A. Therefore, the probability of their joint occurence $\mathbf{P}(A \cap B)$ is simply the product of their individual probabilities $\mathbf{P}(A)\mathbf{P}(B)$.

Definition 10 (Independence of a sequence of events) *We say that a finite or infinite sequence of events $A_1, A_2, \ldots$ are independent if whenever $i_1, i_2, \ldots, i_k$ are distinct elements from the set of indices $\mathbb{N}$, such that $A_{i_1}, A_{i_2}, \ldots, A_{i_k}$ are defined (elements of $\mathcal{F}$), then*

$$\mathbf{P}(A_{i_1} \cap A_{i_2} \ldots \cap A_{i_k}) = \mathbf{P}(A_{i_1})\mathbf{P}(A_{i_2}) \cdots \mathbf{P}(A_{i_k})$$

Example 13 *Let us consider the 'toss a fair coin three times' experiment $\mathcal{E}^3_{\theta=0.5}$ where we toss the coin with $\mathbf{P}(\mathtt{H}) = \theta = 0.5$ and $\mathbf{P}(\mathtt{T}) = 1 - \theta = 0.5$ three times in an* **independent and identical** *manner. Then the probability of each outcome $\omega \in \{\mathtt{H}, \mathtt{T}\}^3$ is the same, i.e. $\mathbf{P}(\omega) = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{8}$.*

Table 2.1: The 8 ω's in the sample space Ω of the experiment $\mathcal{E}^3_\theta$ are given in the first row above.

ω:	HHH	HHT	HTH	HTT	THH	THT	TTH	TTT
$\mathbf{P}(\omega)$:	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$

Chapter 3

Random Variables

It can be be inconvenient to work with a set of outcomes Ω upon which arithmetic is not possible. We are often measuring our outcomes with subsets of real numbers. Some examples include:

Experiment	Possible measured outcomes
Counting the number of typos up to now	$\mathbb{Z}_+ := \{0, 1, 2, \ldots\} \subset \mathbb{R}$
Length in centi-meters of some shells on New Brighton beach	$(0, +\infty) \subset \mathbb{R}$
Waiting time in minutes for the next Orbiter bus to arrive	$\mathbb{R}_+ := [0, \infty) \subset \mathbb{R}$
Vertical displacement from current position of a pollen on water	$\mathbb{R}$

3.1 Basic Definitions

To take advantage of our measurements over the real numbers, in terms of its metric structure and arithmetic, we need to formally define this measurement process using the notion of a random variable.

Definition 11 (Random Variable) *Let $(\Omega, \mathcal{F}, P)$ be some probability triple. Then, a* **Random Variable (RV)***, say X, is a function from the sample space Ω to the set of real numbers $\mathbb{R}$*

$$X : \Omega \to \mathbb{R}$$

such that for every $x \in \mathbb{R}$, the inverse image of the half-open real interval $(-\infty, x]$ is an element of the collection of events $\mathcal{F}$, i.e.:

$$\text{for every } x \in \mathbb{R}, \qquad X^{[-1]}(\ (-\infty, x]\ ) := \{\omega : X(\omega) \leq x\} \in \mathcal{F}\ .$$

This definition can be summarised by the statement that a RV is an $\mathcal{F}$-measurable map. We assign probability to the RV X as follows:

$$\mathbf{P}(X \leq x) = \mathbf{P}(\ X^{[-1]}(\ (-\infty, x]\ )\ ) := \mathbf{P}(\ \{\omega : X(\omega) \leq x\}\ )\ . \tag{3.1}$$

Definition 12 (Distribution Function) *The* **Distribution Function (DF)** *or* **Cumulative Distribution Function (CDF)** *of any RV X, over a probability triple $(\Omega, \mathcal{F}, P)$, denoted by F is:*

$$F(x) := \mathbf{P}(X \leq x) = \mathbf{P}(\ \{\omega : X(\omega) \leq x\}\ ), \qquad \text{for any} \quad x \in \mathbb{R}\ . \tag{3.2}$$

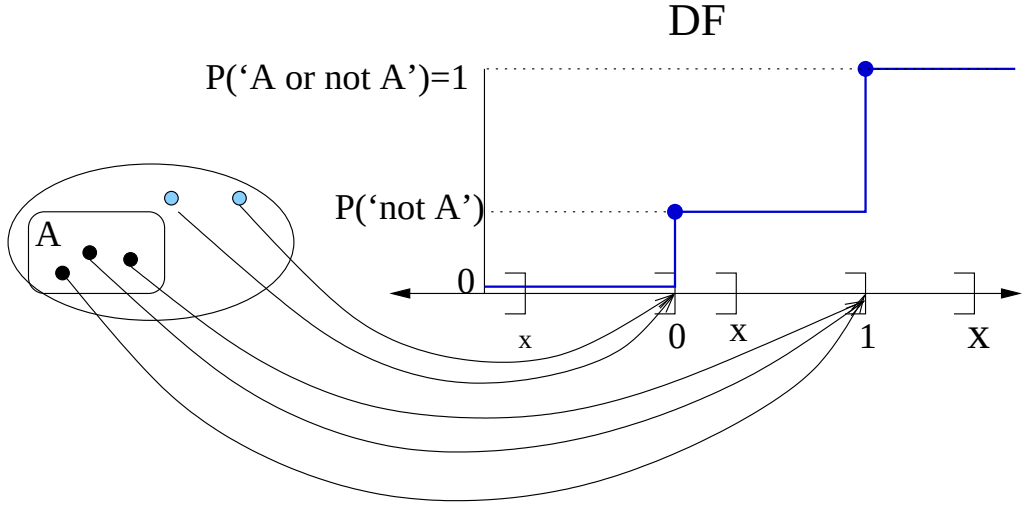
Thus, $F(x)$ or simply F is a non-decreasing, right continuous, $[0, 1]$-valued function over $\mathbb{R}$. When a RV X has DF F we write $X \sim F$.

A special RV that often plays the role of 'building-block' in Probability and Statistics is the indicator function of an event A that tells us whether the event A has occurred or not. Recall that an event belongs to the collection of possible events $\mathcal{F}$ for our experiment.

Definition 13 (Indicator Function) *The* **Indicator Function** *of an event A denoted $\mathbb{1}_A$ is defined as follows:*

$$\mathbb{1}_A(\omega) := \begin{cases} 1 & \text{if} \quad \omega \in A \\ 0 & \text{if} \quad \omega \notin A \end{cases} \tag{3.3}$$

Figure 3.1: The Indicator function of event $A \in \mathcal{F}$ is a RV $\mathbb{1}_A$ with DF F



Classwork 8 *Let us convince ourselves that $\mathbb{1}_A$ is really a RV. For $\mathbb{1}_A$ to be a RV, we need to verify that for any real number $x \in \mathbb{R}$, the inverse image $\mathbb{1}_A^{[-1]}(\ (-\infty, x]\ )$ is an event, ie :*

$$\mathbb{1}_A^{[-1]}(\ (-\infty, x]\ ) := \{\omega : \mathbb{1}_A(\omega) \leq x\} \in \mathcal{F}\ .$$

All we can assume about the collection of events $\mathcal{F}$ is that it contains the event A and that it is a sigma algebra. A careful look at the Figure 3.1 yields:

$$\mathbb{1}_A^{[-1]}(\ (-\infty, x]\ ) := \{\omega : \mathbb{1}_A(\omega) \leq x\} = \begin{cases} \emptyset & \text{if} \quad x < 0 \\ A^c & \text{if} \quad 0 \leq x < 1 \\ A \cup A^c = \Omega & \text{if} \quad 1 \leq x \end{cases}$$

Thus, $\mathbb{1}_A^{[-1]}(\ (-\infty, x]\ )$ is one of the following three sets that belong to $\mathcal{F}$; (1) $\emptyset$, (2) A^c and (3) Ω depending on the value taken by x relative to the interval $[0, 1]$. We have proved that $\mathbb{1}_A$ is indeed a RV.

Some useful properties of the Indicator Function are:

$$\mathbb{1}_{A^c} = 1 - \mathbb{1}_A, \qquad \mathbb{1}_{A \cap B} = \mathbb{1}_A \mathbb{1}_B, \qquad \mathbb{1}_{A \cup B} = \mathbb{1}_A + \mathbb{1}_B - \mathbb{1}_A \mathbb{1}_B$$

3.2 An Elementary Discrete Random Variable

When a RV takes at most countably many values from a discrete set $\mathbb{D} \subset \mathbb{R}$, we call it a **discrete** RV. Often, $\mathbb{D}$ is the set of integers $\mathbb{Z}$.

Definition 14 (probability mass function (PMF)) *Let X be a discrete RV over a probability triple $(\Omega, \mathcal{F}, P)$. We define the* **probability mass function** *(PMF) f of X to be the function $f : \mathbb{D} \to [0, 1]$ defined as follows:*

$$f(x) := \mathbf{P}(X = x) = \mathbf{P}(\ \{\omega : X(\omega) = x\}\ ), \qquad \text{where } x \in \mathbb{D}.$$

The DF F and PMF f for a discrete RV X satisfy the following:

1. For any $x \in \mathbb{R}$,
$$\mathbf{P}(X \leq x) = F(x) = \sum_{\mathbb{D} \ni y \leq x} f(y) := \sum_{y \ \in \ \mathbb{D} \cap (-\infty, x]} f(y)\ .$$

2. For any $a, b \in \mathbb{D}$ with $a < b$,
$$\mathbf{P}(a < X \leq b) = F(b) - F(a) = \sum_{y \ \in \ \mathbb{D} \cap (a, b]} f(y)\ .$$
In particular, when $\mathbb{D} = \mathbb{Z}$ and $a = b - 1$,
$$\mathbf{P}(b - 1 < X \leq b) = F(b) - F(b - 1) = f(b) = \mathbf{P}(\ \{\omega : X(\omega) = b\}\ )\ .$$

3. And of course
$$\sum_{x \in \mathbb{D}} f(x) = 1$$

The Indicator Function $\mathbb{1}_A$ of the event that 'A occurs' for the θ-specific experiment $\mathcal{E}$ over some probability triple $(\Omega, \mathcal{F}, \mathbf{P}_\theta)$, with $A \in \mathcal{F}$, is the Bernoulli(θ) RV. The parameter θ denotes the probability that 'A occurs' (see Figure 3.2 when A is the event that '`H` occurs'). This is our first example of a discrete RV.

Model 1 (Bernoulli(θ)**)** *Given a parameter $\theta \in [0, 1]$, the probability mass function (PMF) for the* Bernoulli(θ) *RV X is:*

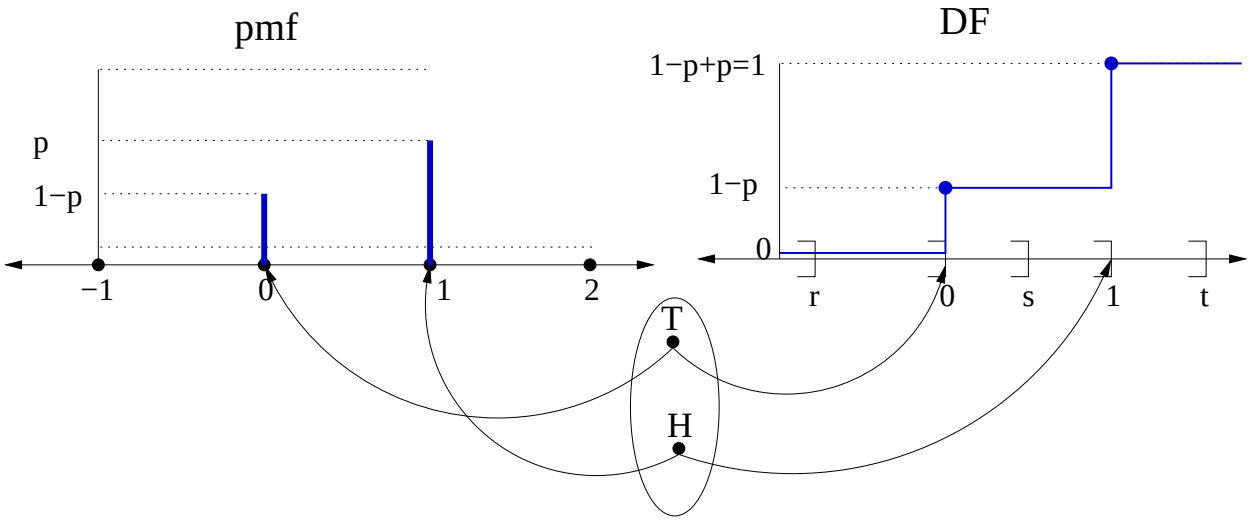
$$f(x; \theta) = \theta^x (1 - \theta)^{1-x} \mathbb{1}_{\{0,1\}}(x) = \begin{cases} \theta & \text{if } x = 1, \\ 1 - \theta & \text{if } x = 0, \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

and its DF is:

$$F(x; \theta) = \begin{cases} 1 & \text{if } 1 \leq x, \\ 1 - \theta & \text{if } 0 \leq x < 1, \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

We emphasise the dependence of the probabilities on the parameter θ by specifying it following the semicolon in the argument for f and F and by subscripting the probabilities, i.e. $\mathbf{P}_\theta(X = 1) = \theta$ and $\mathbf{P}_\theta(X = 0) = 1 - \theta$.

Figure 3.2: The Indicator Function $\mathbb{1}_{\mathtt{H}}$ of the event 'Heads occurs', for the experiment 'Toss 1 times,' $\mathcal{E}_\theta^1$, as the RV X from the sample space $\Omega = \{\mathtt{H}, \mathtt{T}\}$ to $\mathbb{R}$ and its DF F. The probability that 'Heads occurs' and that 'Tails occurs' are $f(1;\theta) = \mathbf{P}_\theta(X = 1) = \mathbf{P}_\theta(\mathtt{H}) = \theta$ and $f(0;\theta) = \mathbf{P}_\theta(X = 0) = \mathbf{P}_\theta(\mathtt{T}) = 1 - \theta$, respectively.



3.3 An Elementary Continuous Random Variable

When a RV takes values in the continuum we call it a **continuous** RV. An example of such a RV is the vertical position (in micro meters) since the original release of a pollen grain on water. Another example of a continuous RV is the volume of water (in cubic meters) that fell on the southern Alps last year.

Definition 15 (probability density function (PDF)) *A RV X is said to be 'continuous' if there exists a piecewise-continuous function f, called the probability density function (PDF) of X, such that for any $a, b \in \mathbb{R}$ with $a < b$,*

$$\mathbf{P}(a < X \leq b) = F(b) - F(a) = \int_a^b f(x)\, dx \ .$$

The following hold for a continuous RV X with PDF f:

1. For any $x \in \mathbb{R}$, $\mathbf{P}(X = x) = 0$.
2. Consequentially, for any $a, b \in \mathbb{R}$ with $a \leq b$,
$$\mathbf{P}(a < X < b) = \mathbf{P}(a < X \leq b) = \mathbf{P}(a \leq X \leq b) = \mathbf{P}(a \leq X < b) \ .$$
3. By the fundamental theorem of calculus, except possibly at finitely many points (where the continuous pieces come together in the piecewise-continuous f):
$$f(x) = \frac{d}{dx} F(x)$$
4. And of course f must satisfy:
$$\int_{-\infty}^{\infty} f(x)\, dx = \mathbf{P}(-\infty < X < \infty) = 1 \ .$$

An elementary and fundamental example of a continuous RV is the Uniform$(0,1)$ RV of Model 2. It forms the foundation for random variate generation and simulation. In fact, it is appropriate to call this the fundamental model since every other experiment can be obtained from this one.

Model 2 (The Fundamental Model) *The probability density function (PDF) of the fundamental model or the* Uniform$(0,1)$ *RV is*

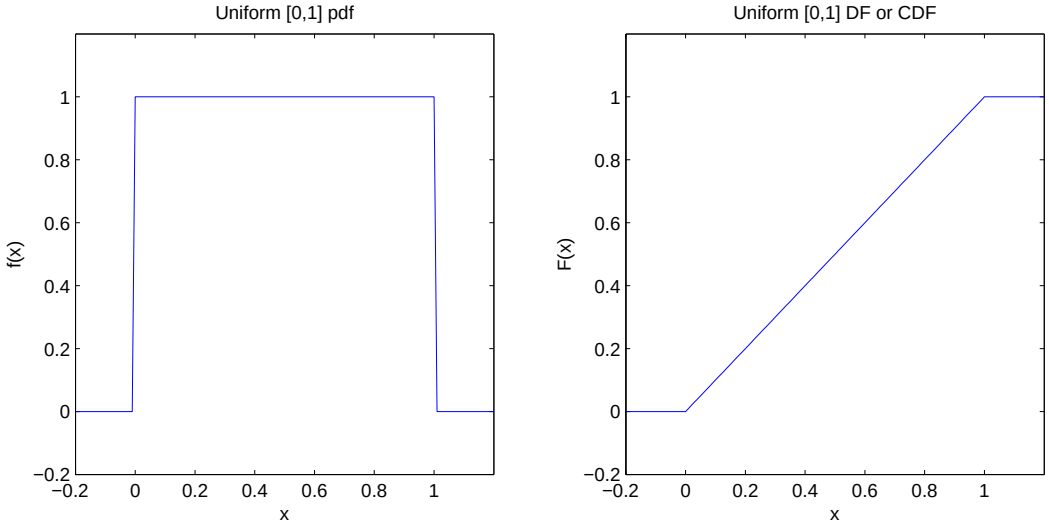
$$f(x) = \mathbb{1}_{[0,1]}(x) = \begin{cases} 1 & \text{if } 0 \leq x \leq 1, \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

and its distribution function (DF) or cumulative distribution function (CDF) is:

$$F(x) := \int_{-\infty}^{x} f(y)\, dy = \begin{cases} 0 & \text{if } x < 0, \\ x & \text{if } 0 \leq x \leq 1, \\ 1 & \text{if } x > 1 \end{cases} \tag{3.7}$$

Note that the DF is the identity map in $[0,1]$. *The PDF and DF are depicted in Figure 3.3.*

Figure 3.3: A plot of the PDF and DF or CDF of the Uniform$(0,1)$ continuous RV X.



Labwork 7 *Let us encode the PDF of* Uniform$(0,1)$ *as an M-file in* MATLAB. *Notice that the PDF function assigns* 1 *to every value of* $x \in [0,1]$ *and* 0 *to every value of* $x \notin [0,1]$. *So, the problem mostly boils down to finding the entries inside and outside the range. We can use* MATLAB*'s built-in* `find` *function for this purpose. We give an example to illustrate the syntax of* `find`.

```
>> Xs=[0.2511    1.6160    0.4733   -5.3517    0.8308    0.5853    2.5497] % an array Xs with real values
Xs =    0.2511    1.6160    0.4733   -5.3517    0.8308    0.5853    2.5497
```

We can obtain the indices of `Xs` *whose values are* ≥ 0, *i.e.* $\{i : \texttt{Xs}(i) \geq 0\}$ *and the indices of* `Xs` *whose values are* ≤ 1, *i.e.* $\{i : \texttt{Xs}(i) \leq 1\}$ *as follows:*

```
>> find(Xs >= 0)
ans =     1     2     3     5     6     7
>> find(Xs <= 1)
ans =     1     3     4     5     6
```

The intersection of the two sets of indices, i.e. $\{i : \texttt{Xs}(i) \geq 0 \text{ and } \texttt{Xs}(i) \leq 1\} = \{i : 0 \leq \texttt{Xs}(i) \leq 1\}$ *can be obtained by* `&`*, the Boolean and, as follows:*

```
>> find(Xs >= 0 & Xs <= 1)
ans =     1     3     5     6
```

Finally, we know which indices of the `Xs` *array should have the PDF value of* 1*. The remaining indices of* `Xs` *should therefore have the PDF value of* 0*. Let us declare an array called* `Pdf` *for the PDF values corresponding to the* `Xs`*. We can initialise this array with zeros using the* `zeros` *function and make it of the same size as* `Xs` *as follows:*

```
>> size(Xs)
ans =     1     7
>> Pdf = zeros(1,7)
Pdf =     0     0     0     0     0     0     0
```

Now, we can set the indices $1, 3, 5, 6$ *(returned by* `find(Xs >= 0 & Xs <= 1)`*) of* `Pdf` *array to 1.*

```
>> Pdf([1     3     5     6])=1
Pdf =     1     0     1     0     1     1     0
```

We can modularise this process for an arbitrary input array `x` *via a function in the following M-file.*

Unif01Pdf.m

```
function Pdf = Unif01Pdf (x)
% Unif01Pdf(x) returns the PDF of Uniform(0,1) RV X
% the input x can be an array
Pdf=zeros(size(x)); % Pdf is an array of zeros and of the same size as x
% use the built-in find function to find the Indices of x whose values are in the range [0,1]
Indices = find(x>=0 & x<=1);
Pdf(Indices) = 1; % Set these indices in array Pdf to 1=PDF of X over [0,1]
```

Let us call the function we wrote called `Unif01Pdf` *next.*

```
>> help Unif01Pdf
  Unif01Pdf(x) returns the PDF of Uniform(0,1) RV X
  the input x can be an array
>> Xs
Xs =    0.2511    1.6160    0.4733   -5.3517    0.8308    0.5853    2.5497
>> Unif01Pdf(Xs)
ans =     1     0     1     0     1     1     0
```

Labwork 8 *Understand each step in the function* `Unif01Cdf`*:*

Unif01Cdf.m

```
function Cdf = Unif01Cdf (x)
% Unif01Cdf(x) returns the CDF of Uniform(0,1) RV X
% the input x can be an array
Cdf=zeros(size(x));% Cdf is an array of zeros and of the same size as x
% use the built-in find function to find the indices of x whose values are >= 1
Indices = find(x>=1);
Cdf(Indices) = 1;% Set these indices in array Cdf to 1
Indices = find(x>=0 & x<=1); % find indices of x with values in [0,1]
Cdf(Indices)=x(Indices); % set the Cdf of x in [0,1] equal to x
```

When we type in `help Unif01Cdf`, `Xs` *and* `Unif01Cdf(Xs)` *we can confirm that the* `Unif01Cdf` *function is correctly reporting the CDF values of the input array* `Xs`.

```
>> help Unif01Cdf
  Unif01Cdf(x) returns the CDF of Uniform(0,1) RV X
  the input x can be an array
>> Xs
Xs =    0.2511    1.6160    0.4733   -5.3517    0.8308    0.5853    2.5497
>> Unif01Cdf(Xs)
ans =    0.2511    1.0000    0.4733         0    0.8308    0.5853    1.0000
```

Labwork 9 *Generate the plot of the PDF and the CDF for the* Uniform$(0, 1)$ *RV* X *by following the commands below. Go through every step and understand each command when you reproduce the plot.*

plotunif.m

```
% Plot the PDF and CDF for Uniform[0,1] RV
x = -1:0.01:2; % vector from -1 to 2; w/ increment .05
% get the [0,1] uniform pdf values of x in vector pdf
pdf = unif01pdf(x);
% get the [0,1] uniform DF or cdf of x in vector cdf
cdf = unif01cdf(x);
% do the plots
% subplot for pdf: subplot(1,2,1) means there is 1 row with
% 2 columns of subplots and here is the first of them
subplot(1,2,1), plot(x,pdf)
title('Uniform [0,1] pdf')       % title as a string
xlabel('x'), ylabel('f(x)')      % x and y axes labels
axis([-0.2 1.2 -0.2 1.2])   % range specs for x and y axes
axis square                      % axes scaled as square
% subplot for cdf: subplot(1,2,1) means there is 1 row with
% 2 columns of subplots and here is the first of them
subplot(1,2,2), plot(x,cdf)
title('Uniform [0,1] DF or CDF')
xlabel('x'), ylabel('F(x)')
axis([-0.2 1.2 -0.2 1.2])
axis square
```

The plot was saved as an encapsulated postscript file from the File menu of the Figure window and is displayed in Figure 3.3.

3.4 Expectations

It is convenient to summarise a RV by a single number. This single number can be made to represent some average or expected feature of the RV via an integral with respect to the density of the RV.

Definition 16 (Expectation of a RV) *The* **expectation**, *or* **expected value**, *or* **mean**, *or* **first moment**, *of a random variable* X, *with distribution function* F *and density* f, *is defined to be*

$$\mathbf{E}(X) := \int x \, dF(x) = \begin{cases} \sum_x x f(x) & \text{if } X \text{ is discrete} \\ \int x f(x) \, dx & \text{if } X \text{ is continuous} , \end{cases} \tag{3.8}$$

provided the sum or integral is well-defined. We say the expectation exists if

$$\int |x| \, dF(x) < \infty \ . \tag{3.9}$$

Sometimes, we denote $\mathbf{E}(X)$ *by* $\mathbf{E}X$ *for brevity. Thus, the expectation is a single-number summary of the RV* X *and may be thought of as the average. We subscript* E *to specify the parameter* $\theta \in \mathbb{\Theta}$ *with respect to which the integration is undertaken.*

$$\mathbf{E}_\theta X := \int x \, dF(x; \theta)$$

Definition 17 (Variance of a RV) *Let* X *be a RV with mean or expectation* $\mathbf{E}(X)$*. Variance of* X *denoted by* $\mathbf{V}(X)$ *or* VX *is*

$$\mathbf{V}(X) := \mathbf{E}\left((X - \mathbf{E}(X))^2\right) = \int (x - \mathbf{E}(X))^2 \, dF(x) \ ,$$

provided this expectation exists. The **standard deviation** *denoted by* $\mathrm{sd}(X) := \sqrt{\mathbf{V}(X)}$*. Thus variance is a measure of "spread" of a distribution.*

Definition 18 (k-th moment of a RV) *We call*

$$\mathbf{E}(X^k) = \int x^k \, dF(x)$$

as the k*-th moment of the RV* X *and say that the* k*-th moment exists when* $\mathbf{E}(|X|^k) < \infty$*. We call the following expectation as the* k*-th central moment:*

$$\mathbf{E}\left((X - \mathbf{E}(X))^k\right) \ .$$

Properties of Expectations

1. If the k-th moment exists and if $j < k$ then the j-th moment exists.
2. If $X_1, X_2, \ldots, X_n$ are RVs and $a_1, a_2, \ldots, a_n$ are constants, then

$$\mathbf{E}\left(\sum_{i=1}^n a_i X_i\right) = \sum_{i=1}^n a_i \mathbf{E}(X_i) \ . \tag{3.10}$$

3. Let $X_1, X_2, \ldots, X_n$ be independent RVs, then

$$\mathbf{E}\left(\prod_{i=1}^n X_i\right) = \prod_{i=1}^n \mathbf{E}(X_i) \ . \tag{3.11}$$

4. $\mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2$. [prove by completing the square and applying (3.10)]
5. If a and b are constants then:

$$\mathbf{V}(aX + b) = a^2 \mathbf{V}(X) \ . \tag{3.12}$$

6. If $X_1, X_2, \ldots, X_n$ are independent and $a_1, a_2, \ldots, a_n$ are constants, then:

$$\mathbf{V}\left(\sum_{i=1}^n a_i X_i\right) = \sum_{i=1}^n a_i^2 \mathbf{V}(X_i) \ . \tag{3.13}$$

Mean and variance of Bernoulli(θ) **RV:** Let $X \sim \text{Bernoulli}(\theta)$. Then,

$$\mathbf{E}(X) = \sum_{x=0}^{1} x f(x) = (0 \times (1-\theta)) + (1 \times \theta) = 0 + \theta = \theta \ ,$$

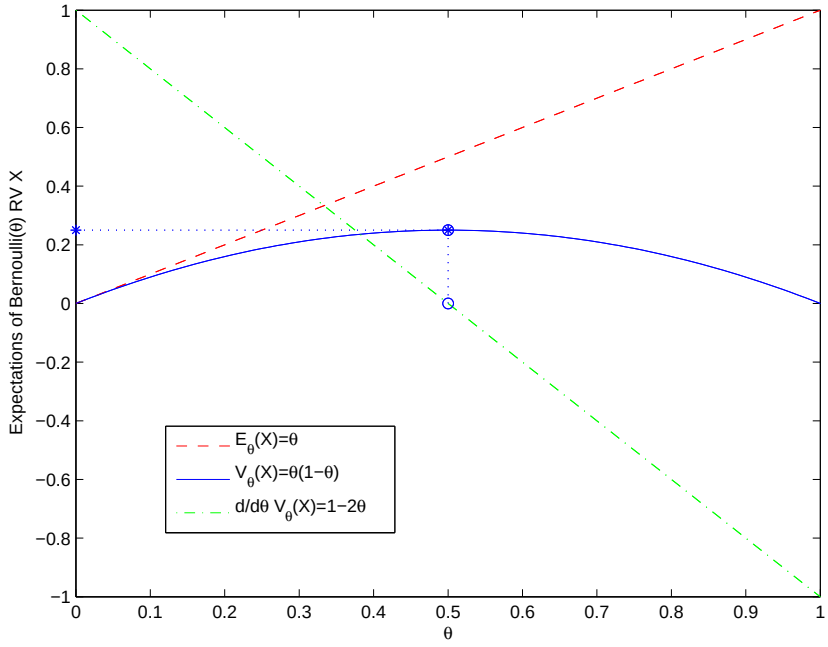
$$\mathbf{E}(X^2) = \sum_{x=0}^{1} x^2 f(x) = (0^2 \times (1-\theta)) + (1^2 \times \theta) = 0 + \theta = \theta \ ,$$

$$\mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2 = \theta - \theta^2 = \theta(1-\theta) \ .$$

Parameter specifically,

$$\mathbf{E}_\theta(X) = \theta \qquad \text{and} \qquad \mathbf{V}_\theta(X) = \theta(1-\theta) \ .$$

Figure 3.4: Mean ($\mathbf{E}_\theta(X)$), variance ($\mathbf{V}_\theta(X)$) and the rate of change of variance ($\frac{d}{d\theta}\mathbf{V}_\theta(X)$) of a Bernoulli($\theta$) RV X as a function of the parameter θ.



Maximum of the variance $\mathbf{V}_\theta(X)$ is found by setting the derivative to zero, solving for θ and showing the second derivative is locally negative, i.e. $\mathbf{V}_\theta(X)$ is concave down:

$$\mathbf{V}'_\theta(X) := \frac{d}{d\theta}\mathbf{V}_\theta(X) = 1 - 2\theta = 0 \iff \theta = \frac{1}{2} \ , \qquad \mathbf{V}''_\theta(X) := \frac{d}{d\theta}\left(\frac{d}{d\theta}\mathbf{V}_\theta(X)\right) = -2 < 0 \ ,$$

$$\max_{\theta \in [0,1]} \mathbf{V}_\theta(X) = \frac{1}{2}\left(1 - \frac{1}{2}\right) = \frac{1}{4} \ , \text{since } \mathbf{V}_\theta(X) \text{ is maximized at } \theta = \frac{1}{2}$$

The plot depicting these expectations as well as the rate of change of the variance are depicted in Figure 3.4. Note from this Figure that $\mathbf{V}_\theta(X)$ attains its maximum value of $1/4$ at $\theta = 0.5$ where $\frac{d}{d\theta}\mathbf{V}_\theta(X) = 0$. Furthermore, we know that we don't have a minimum at $\theta = 0.5$ since the second derivative $\mathbf{V}''_\theta(X) = -2$ is negative for any $\theta \in [0,1]$. This confirms that $\mathbf{V}_\theta(X)$ is concave down and therefore we have a maximum of $\mathbf{V}_\theta(X)$ at $\theta = 0.5$. We will revisit this example when we employ a numerical approach called Newton-Raphson method to solve for the maximum of a differentiable function by setting its derivative equal to zero.

Mean and variance of Uniform$(0,1)$ **RV:** Let $X \sim \text{Uniform}(0,1)$. Then,

$$\mathbf{E}(X) = \int_{x=0}^{1} x f(x)\,dx = \int_{x=0}^{1} x\ 1\,dx = \frac{1}{2}\left(x^2\right]_{x=0}^{x=1} = \frac{1}{2}(1-0) = \frac{1}{2}\ ,$$

$$\mathbf{E}(X^2) = \int_{x=0}^{1} x^2 f(x)\,dx = \int_{x=0}^{1} x^2\ 1\,dx = \frac{1}{3}\left(x^3\right]_{x=0}^{x=1} = \frac{1}{3}(1-0) = \frac{1}{3}\ ,$$

$$\mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2 = \frac{1}{3} - \left(\frac{1}{2}\right)^2 = \frac{1}{3} - \frac{1}{4} = \frac{1}{12}\ .$$

Proposition 2 (Winnings on Average) *Let $Y = r(X)$. Then*

$$\mathbf{E}(Y) = \mathbf{E}(r(X)) = \int r(x)\,dF(x)\ .$$

Think of playing a game where we draw $x \sim X$ and then I pay you $y = r(x)$. Then your average income is $r(x)$ times the chance that $X = x$, summed (or integrated) over all values of x.

Example 14 (Probability is an Expectation) *Let A be an event and let $r(X) = \mathbb{1}_A(x)$. Recall $\mathbb{1}_A(x)$ is 1 if $x \in A$ and $\mathbb{1}_A(x) = 0$ if $x \notin A$. Then*

$$\mathbf{E}(\mathbb{1}_A(X)) = \int \mathbb{1}_A(x)\,dF(x) = \int_A f(x)\,dx = \mathbf{P}(X \in A) = \mathbf{P}(A) \tag{3.14}$$

Thus, probability is a special case of expectation. Recall our LTRF motivation for the definition of probability and make the connection.

3.5 Stochastic Processes

Definition 19 (Independence of RVs) *A finite or infinite sequence of RVs $\{X_1, X_2, \ldots\}$ is said to be independent or independently distributed if*

$$\mathbf{P}(X_{i_1} \le x_{i_1}, X_{i_2} \le x_{i_2}, \ldots, X_{i_k} \le x_{i_k}) = \mathbf{P}(X_{i_1} \le x_{i_1})\mathbf{P}(X_{i_2} \le x_{i_2}) \cdots, \mathbf{P}(X_{i_k} \le x_{i_k})$$

for any distinct subset $\{i_1, i_2, \ldots, i_l\}$ of indices of the sequence of RVs and any sequence of real numbers $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$.

By the above definition, the sequence of **discrete** *RVs $X_1, X_2, \ldots$ taking values in an at most countable set $\mathbb{D}$ are said to be independently distributed if for any distinct subset of indices $\{i_1, i_2, \ldots, i_k\}$ such that the corresponding RVs $X_{i_1}, X_{i_2}, \ldots, X_{i_k}$ exists as a distinct subset of our original sequence of RVs $X_1, X_2, \ldots$ and for any elements $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ in $\mathbb{D}$, the following equality is satisfied:*

$$\mathbf{P}(X_{i_1} = x_{i_1}, X_{i_2} = x_{i_2}, \ldots, X_{i_k} = x_{i_k}) = \mathbf{P}(X_{i_1} = x_{i_1})\mathbf{P}(X_{i_2} = x_{i_2}) \cdots \mathbf{P}(X_{i_k} = x_{i_k})$$

For an independent sequence of RVs $\{X_1, X_2, \ldots\}$, we have

$$\begin{aligned}
&\mathbf{P}(X_{i+1} \le x_{i+1} | X_i \le x_i, X_{i-1} \le x_{i-1}, \ldots, X_1 \le x_1) \\
= \ & \frac{\mathbf{P}(X_{i+1} \le x_{i+1}, X_i \le x_i, X_{i-1} \le x_{i-1}, \ldots, X_1 \le x_1)}{\mathbf{P}(X_i \le x_i, X_{i-1} \le x_{i-1}, \ldots, X_1 \le x_1)} \\
= \ & \frac{\mathbf{P}(X_{i+1} \le x_{i+1})\mathbf{P}(X_i \le x_i)\mathbf{P}(X_{i-1} \le x_{i-1}) \cdots \mathbf{P}(X_1 \le x_1)}{\mathbf{P}(X_i \le x_i)\mathbf{P}(X_{i-1} \le x_{i-1}) \cdots \mathbf{P}(X_1 \le x_1)} \\
= \ & \mathbf{P}(X_{i+1} \le x_{i+1})
\end{aligned}$$

The above equality that

$$\mathbf{P}(X_{i+1} \leq x_{i+1} | X_i \leq x_i, X_{i-1} \leq x_{i-1}, \ldots, X_1 \leq x_1) = \mathbf{P}(X_{i+1} \leq x_{i+1})$$

simply says that the conditional distribution of the RV X_{i+1} given all previous RVs $X_i, X_{i-1}, \ldots, X_1$ is simply determined by the distribution of X_{i+1}.

When a sequence of RVs are not independent they are said to be **dependent**.

Definition 20 (Stochastic Process) *A collection of RVs*

$$\{X_\alpha\}_{\alpha \in N} := \{\ X_\alpha : \alpha \in \mathbb{A}\ \}$$

is called a **stochastic process**. *Thus, for every $\alpha \in \mathbb{A}$, the index set of the stochastic process, X_α is a RV. If the index set $\mathbb{A} \subset \mathbb{Z}$ then we have a* **discrete time stochastic process**, *typically denoted by*

$$\{X_i\}_{i \in \mathbb{Z}} := \{\ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots\},\ \textit{or}$$

$$\{X_i\}_{i \in \mathbb{N}} := \{X_1, X_2, \ldots\},\ \textit{or}$$

$$\{X_i\}_{i \in [n]} := \{X_1, X_2, \ldots, X_n\},\ \textit{where, } [n] := \{1, 2, \ldots, n\}\ .$$

If $\mathbb{A} \subset \mathbb{R}$ then we have a **continuous time stochastic process**, *typically denoted by $\{X_t\}_{t \in \mathbb{R}}$, etc.*

Definition 21 (Independent and Identically Distributed (IID)) *The finite or infinite sequence of RVs or the stochastic process $\{X_1, X_2, \ldots\}$ is said to be independent and identically distributed or IID if :*

- *$\{X_1, X_2, \ldots\}$ is independently distributed according to Definition 19, and*
- *$F(X_1) = F(X_2) = \cdots$, ie. all the X_i's have the same DF $F(X_1)$.*

This is perhaps the most elementary class of stochastic processes and we succinctly denote it by

$$\{X_i\}_{i \in [n]} := \{X_1, X_2, \ldots, X_n\} \overset{\text{IID}}{\sim} F, \quad \textit{or} \quad \{X_i\}_{i \in \mathbb{N}} := \{X_1, X_2, \ldots\} \overset{\text{IID}}{\sim} F\ .$$

We sometimes replace the DF F above by the name of the RV.

Definition 22 (Independently Distributed) *The sequence of RVs or the stochastic process $\{X_i\}_{i \in \mathbb{N}} := \{X_1, X_2, \ldots\}$ is said to be independently distributed if :*

- *$\{X_1, X_2, \ldots\}$ is independently distributed according to Definition 19.*

This is a class of stochastic processes that is more general than the IID class.

Let us consider a few discrete RVs for the simple coin tossing experiment $\mathcal{E}_\theta^3$ that build on the Bernoulli(p) RV X_i for the i-th toss in an **independent and identically distibuted (IID.)** manner.

Table 3.1: The 8 ω's in the sample space Ω of the experiment $\mathcal{E}_\theta^3$ are given in the first row above. The RV Y is the number of 'Heads' in the 3 tosses and the RV Z is the number of 'Tails' in the 3 tosses. Finally, the RVs Y' and Z' are the indicator functions of the event that 'all three tosses were Heads' and the event that 'all three tosses were Tails', respectively.

ω:	HHH	HHT	HTH	HTT	THH	THT	TTH	TTT	RV Definitions / Model
$\mathbf{P}(\omega)$:	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$X_i \overset{\text{IID}}{\sim} \text{Bernoulli}(\frac{1}{2})$
$Y(\omega)$:	3	2	2	1	2	1	1	0	$Y := X_1 + X_2 + X_3$
$Z(\omega)$:	0	1	1	2	1	2	2	3	$Z := (1 - X_1) + (1 - X_2) + (1 - X_3)$
$Y'(\omega)$:	1	0	0	0	0	0	0	0	$Y' := X_1 X_2 X_3$
$Z'(\omega)$:	0	0	0	0	0	0	0	1	$Y' := (1 - X_1)(1 - X_2)(1 - X_3)$

Classwork 9 *Describe the probability of the RV Y and Y' of Table 3.1 in terms of its PMF. Repeat the process for the RV Z in your spare time.*

$$\mathbf{P}(Y = y) = \begin{cases} \end{cases} \qquad \mathbf{P}(Y' = y') = \begin{cases} \end{cases}$$

Classwork 10 *1. What is conditional probability $\mathbf{P}(Y|Y' = 0)$?*

$\mathbf{P}(Y = y\|Y' = 0)$	$= \frac{\mathbf{P}(Y=y,Y'=0)}{\mathbf{P}(Y'=0)}$	$= \frac{\mathbf{P}(\{\omega:Y(\omega)=y \cap Y'(\omega)=0\})}{\mathbf{P}(\{\omega:Y'(\omega)=0\})}$	$=?$
$\mathbf{P}(Y = 0\|Y' = 0)$	$\frac{\mathbf{P}(Y=0,Y'=0)}{\mathbf{P}(Y'=0)}$	$\frac{\frac{1}{8}}{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}$	$\frac{1}{7}$
$\mathbf{P}(Y = 1\|Y' = 0)$	$\frac{\mathbf{P}(Y=1,Y'=0)}{\mathbf{P}(Y'=0)}$	$\frac{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}$	$\frac{3}{7}$
$\mathbf{P}(Y = 2\|Y' = 0)$	$\frac{\mathbf{P}(Y=2,Y'=0)}{\mathbf{P}(Y'=0)}$	$\frac{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}$	$\frac{3}{7}$
$\mathbf{P}(Y = 3\|Y' = 0)$	$\frac{\mathbf{P}(Y=3,Y'=0)}{\mathbf{P}(Y'=0)}$	$\frac{\mathbf{P}(\emptyset)}{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}$	0
$\mathbf{P}(Y \in \{0,1,2,3\}\|Y' = 0)$	$\frac{\sum_{y=0}^{3}\mathbf{P}(Y=y,Y'=0)}{\mathbf{P}(Y'=0)}$	$\frac{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}{\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}$	1

2. What is $\mathbf{P}(Y|Y' = 1)$?

$$\mathbf{P}(Y = y|Y' = 1) = \begin{cases} 1 & \text{if } y = 3 \\ 0 & \text{otherwise} \end{cases}$$

3.5.1 Markov Processes

When a stochastic process $\{X_\alpha\}_{\alpha\in\mathbb{A}}$ is not independent it is said to be dependent. Next we introduce ourselves to one of the simplest stochastic process with a 'first-order' dependence called Markov dependence.

Definition 23 (Markov chain) *The stochastic process $\{X_i\}_{i\in\mathbb{Z}}$ is called a discrete time markov chain if*

$$\mathbf{P}(X_{i+1} \leq x_{i+1}|X_i \leq x_i, X_{i-1} \leq x_{i-1}, \ldots) = \mathbf{P}(X_{i+1} \leq x_{i+1}|X_i)$$

Thus, the probability of the future RV X_{i+1}, when given the entire past information about the RVs $X_i, X_{i-1}, \ldots$, is only determined by the present RV X_i, ie. probability of the future RV X_{i+1} given the entire history is only dependent on the present RV X_i.

Chapter 4

Uniform Random Number Generators

4.1 Introduction

Our probability model and the elementary continuous Uniform$(0, 1)$ RV are built from the abstract concept of a random variable over a probability triple. A direct implementation of these ideas on a computing machine is not possible. In practice, random variables are typically simulated by **deterministic** methods or algorithms. Such algorithms generate sequences of numbers whose behavior is virtually indistinguishable from that of truly random sequences. In computational statistics, simulating realisations from a given RV is usually done in two distinct steps. First, sequences of numbers that imitate independent and identically distributed (IID) Uniform$(0, 1)$ RVs are generated. Second, appropriate transformations are made to these imitations of IID Uniform$(0, 1)$ random variates in order to imitate IID random variates from other random variables or other random structures. These two steps are essentially independent and are studied by two non-overlapping groups of researchers. The formal term **pseudo-random number generator** (PRNG) or simply **random number generator** (RNG) usually refers to some deterministic algorithm used in the first step to produce pseudo-random numbers (PRNs) that imitate IID Uniform$(0, 1)$ random variates.

In the following chapters, we focus on transforming IID Uniform$(0, 1)$ variates to other non-uniform variates. In this chapter, we focus on the art of imitating IID Uniform$(0, 1)$ variates. Before we delve into the theory of RNGs, let us learn to implement the MATLAB function that generates PRNs.

4.2 Uniform Random Numbers in MATLAB

In MATLAB the function `rand` produces a deterministic PRN sequence. First, read `help rand`. We can generate PRNs as follows:

```
>> rand(1,10) % generate a 1 X 10 array of PRNs
ans =
    0.8147    0.9058    0.1270    0.9134    0.6324    0.0975    0.2785    0.5469    0.9575    0.9649
>> rand(1,10) % generate another 1 X 10 array of PRNs
ans =
    0.1576    0.9706    0.9572    0.4854    0.8003    0.1419    0.4218    0.9157    0.7922    0.9595
>> rand('twister',5489) % reset the PRNG to default state Mersenne Twister with seed=5489
>> rand(1,10) % reproduce the first array
ans =
    0.8147    0.9058    0.1270    0.9134    0.6324    0.0975    0.2785    0.5469    0.9575    0.9649
>> rand(1,10) % reproduce the second array
```

```
ans =
    0.1576    0.9706    0.9572    0.4854    0.8003    0.1419    0.4218    0.9157    0.7922    0.9595
```

In general, you can use any seed value to initiate your PRNG. You may use the `clock` command to set the seed:

```
>> rand('twister',SeedFromClock) % initialize the PRNG
>> rand(1,10)
ans =
    0.3696    0.3974    0.6428    0.6651    0.6961    0.7311    0.8982    0.6656    0.6991    0.8606
>> rand(2,10)
ans =
    0.3432    0.9511    0.3477    0.1007    0.8880    0.0853    0.6067    0.6976    0.4756    0.1523
    0.5827    0.5685    0.0125    0.1555    0.5551    0.8994    0.2502    0.5955    0.5960    0.5700
>> rand('twister',SeedFromClock) % initialize the PRNG to same SeedFromClock
>> rand(1,10)
ans =
    0.3696    0.3974    0.6428    0.6651    0.6961    0.7311    0.8982    0.6656    0.6991    0.8606
```

Chapter 5

Statistics

5.1 Data and Statistics

Definition 24 (Data) *The function X measures the outcome ω of an experiment with sample space Ω [Often, the sample space is also denoted by S]. Formally, X is a random variable [or a random vector $X = (X_1, X_2, \ldots, X_n)$, i.e. a vector of random variables] taking values in the* **data space** $\mathbb{X}$*:*

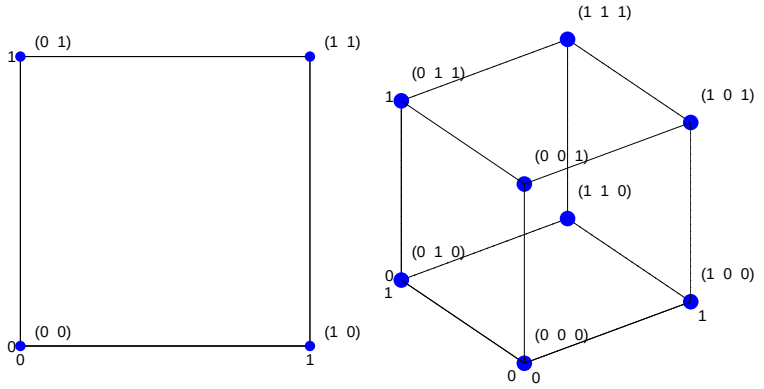
$$X(\omega) : \Omega \mapsto \mathbb{X} \ .$$

The realisation of the RV X when an experiment is performed is the observation or data $x \in \mathbb{X}$. That is, when the experiment is performed once and it yields a specific $\omega \in \Omega$, the data $X(\omega) = x \in \mathbb{X}$ is the corresponding realisation of the RV X.

Figure 5.1: Sample Space, Random Variable, Realisation, Data, and Data Space.

Example 15 *For some given parameter $\theta \in \Theta := [0, 1]$, consider n IID* Bernoulli(θ) *trials, i.e. $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim}$ Bernoulli(θ). Then the random vector $X = (X_1, X_2, \ldots, X_n)$, which takes values in the data space $\mathbb{X} = \{0, 1\}^n := \{(x_1, x_2, \ldots, x_n) : x_i \in \{0, 1\},\ i = 1, 2, \ldots, n\}$, made up of vertices of the n-dimensional hyper-cube, measures the outcomes of this experiment. A particular realisation of X, upon performance of this experiment, is the observation, data or data vector*

Figure 5.2: Data Spaces $\mathbb{X} = \{0,1\}^2$ and $\mathbb{X} = \{0,1\}^3$ for two and three Bernoulli trials, respectively.



$(x_1, x_2, \ldots, x_n)$. *For instance, if we observed* $n-1$ *tails and* 1 *heads, in that order, then our data vector* $(x_1, x_2, \ldots, x_{n-1}, x_n) = (0, 0, \ldots, 0, 1)$.

Definition 25 (Statistic) *A* **statistic** T *is any function of the data:*

$$T(x) : \mathbb{X} \mapsto \mathbb{T} \ .$$

Thus, a statistic T *is also an RV that takes values in the space* $\mathbb{T}$. *When* $x \in \mathbb{X}$ *is the realisation of an experiment, we let* $T(x) = t$ *denote the corresponding realisation of the statistic* T. *Sometimes we use* $T_n(X)$ *and* $\mathbb{T}_n$ *to emphasise that* X *is an* n*-dimensional random vector, i.e.* $\mathbb{X} \subset \mathbb{R}^n$

Classwork 11 *Is the RV* X*, for which the realisation is the observed data* $X(\omega) = x$*, a statistic? In other words, is the data a statistic? [Hint: consider the identity map* $T(x) = x : \mathbb{X} \mapsto \mathbb{T} = \mathbb{X}$.*]*

Next, we define two important statistics called the **sample mean** and **sample variance**. Since they are obtained from the sample data, they are called **sample moments**, as opposed to the **population moments**. The corresponding population moments are $\mathbf{E}(X_1)$ and $\mathbf{V}(X_1)$, respectively.

Definition 26 (Sample Mean) *From a given a sequence of RVs* $X_1, X_2, \ldots, X_n$*, we may obtain another RV called the* n*-samples mean or simply the sample mean:*

$$T_n(\ (X_1, X_2, \ldots, X_n)\ ) = \overline{X}_n(\ (X_1, X_2, \ldots, X_n)\ ) := \frac{1}{n}\sum_{i=1}^{n} X_i \ . \tag{5.1}$$

For brevity, we write

$$\overline{X}_n(\ (X_1, X_2, \ldots, X_n)\ ) \quad as \quad \overline{X}_n \ ,$$

and its realisation

$$\overline{X}_n(\ (x_1, x_2, \ldots, x_n)\ ) \quad as \quad \overline{x}_n \ .$$

Note that the expectation and variance of $\overline{X}_n$ are:

$$\begin{aligned}\mathbf{E}(\overline{X}_n) &= \mathbf{E}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) && \text{[by definition (5.1)]}\\ &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{E}\left(X_i\right) && \text{[by property (3.10)]}\end{aligned}$$

Furthermore, if every X_i in the original sequence of RVs $X_1, X_2, \ldots$ is **identically** distributed with the same expectation, by convention $\mathbf{E}(X_1)$, then:

$$\mathbf{E}(\overline{X}_n) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{E}\left(X_i\right) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{E}\left(X_1\right) = \frac{1}{n}\; n\; \mathbf{E}\left(X_1\right) = \mathbf{E}\left(X_1\right)\ . \tag{5.2}$$

Similarly, we can show that:

$$\begin{aligned}\mathbf{V}(\overline{X}_n) &= \mathbf{V}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) && \text{[by definition (5.1)]}\\ &= \left(\frac{1}{n}\right)^2\mathbf{V}\left(\sum_{i=1}^{n} X_i\right) && \text{[by property (3.12)]}\end{aligned}$$

Furthermore, if the original sequence of RVs $X_1, X_2, \ldots$ is **independently** distributed then:

$$\mathbf{V}(\overline{X}_n) = \left(\frac{1}{n}\right)^2\mathbf{V}\left(\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\;\sum_{i=1}^{n}\mathbf{V}\left(X_i\right) \qquad \text{[by property (3.13)]}$$

Finally, if the original sequence of RVs $X_1, X_2, \ldots$ is **independently and identically** distributed with the same variance ($\mathbf{V}(X_1)$ by convention) then:

$$\mathbf{V}(\overline{X}_n) = \frac{1}{n^2}\;\sum_{i=1}^{n}\mathbf{V}\left(X_i\right) = \frac{1}{n^2}\;\sum_{i=1}^{n}\mathbf{V}\left(X_1\right) = \frac{1}{n^2}\;n\;\mathbf{V}\left(X_1\right) = \frac{1}{n}\;\mathbf{V}\left(X_1\right)\ . \tag{5.3}$$

Labwork 10 *After initializing the fundamental sampler, we draw five samples and then obtain the sample mean using the* MATLAB *function* `mean`. *In the following, we will reuse the samples stored in the array* `XsFromUni01Twstr101`.

```
>> rand('twister',101); % initialise the fundamental Uniform(0,1) sampler
>> XsFromUni01Twstr101=rand(1,5); % simulate n=5 IID samples from Uniform(0,1) RV
>> SampleMean=mean(XsFromUni01Twstr101);% find sample mean
>> disp(XsFromUni01Twstr101); % The data-points x_1,x_2,x_3,x_4,x_5 are:
    0.5164    0.5707    0.0285    0.1715    0.6853
>> disp(SampleMean); % The Sample mean is :
    0.3945
```

We can thus use `mean` *to obtain the sample mean* $\overline{x}_n$ *of* n *sample points* $x_1, x_2, \ldots, x_n$.

We may also obtain the sample mean using the `sum` *function and a division by sample size:*

```
>> sum(XsFromUni01Twstr101) % take the sum of the elements of the XsFromUni01Twstr101 array
ans =    1.9723
>> sum(XsFromUni01Twstr101) / 5 % divide the sum by the sample size 5
ans =    0.3945
```

We can also obtain the sample mean via matrix product or multiplication as follows:

```
>> size(XsFromUni01Twstr101) % size(SomeArray) gives the size or dimensions of the arrar SomeArray
ans =     1     5
>> ones(5,1) % here ones(5,1) is an array of 1's with size or dimension 5 X 1
ans =
     1
     1
     1
     1
     1
>> XsFromUni01Twstr101 * ones(5,1) % multiplying an 1 X 5 matrix with a 5 X 1 matrix of Ones
ans =    1.9723
>> XsFromUni01Twstr101 * ( ones(5,1) * 1/5) % multiplying an 1 X 5 matrix with a 5 X 1 matrix of 1/5 's
ans =    0.3945
```

Definition 27 (Sample Variance & Standard Deviation) *From a given a sequence of random variables $X_1, X_2, \ldots, X_n$, we may obtain another statistic called the n-samples variance or simply the sample variance :*

$$T_n(\ (X_1, X_2, \ldots, X_n)\ ) = S_n^2(\ (X_1, X_2, \ldots, X_n)\ ) := \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X}_n)^2\ . \tag{5.4}$$

For brevity, we write $S_n^2(\ (X_1, X_2, \ldots, X_n)\ )$ as S_n^2 and its realisation $S_n^2(\ (x_1, x_2, \ldots, x_n)\ )$ as s_n^2.

Sample standard deviation is simply the square root of sample variance:

$$S_n(\ (X_1, X_2, \ldots, X_n)\ ) = \sqrt{S_n^2(\ (X_1, X_2, \ldots, X_n)\ )} \tag{5.5}$$

For brevity, we write $S_n(\ (X_1, X_2, \ldots, X_n)\ )$ as S_n and its realisation $S_n(\ (x_1, x_2, \ldots, x_n)\ )$ as s_n.

Once again, if $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} X_1$, the expectation of the sample variance is:

$$\mathbf{E}(S_n^2) = \mathbf{V}(X_1)\ .$$

Labwork 11 *We can compute the sample variance and sample standard deviation for the five samples stored in the array* `XsFromUni01Twstr101` *from Labwork 10 using* MATLAB*'s functions* `var` *and* `std`*, respectively.*

```
>> disp(XsFromUni01Twstr101); % The data-points x_1,x_2,x_3,x_4,x_5 are :
    0.5164    0.5707    0.0285    0.1715    0.6853
>> SampleVar=var(XsFromUni01Twstr101);% find sample variance
>> SampleStd=std(XsFromUni01Twstr101);% find sample standard deviation
>> disp(SampleVar) % The sample variance is:
    0.0785
>> disp(SampleStd) % The sample standard deviation is:
    0.2802
```

It is important to bear in mind that the statistics such as sample mean and sample variance are random variables and have an underlying distribution.

Definition 28 (Order Statistics) *Suppose* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F$*, where* F *is the DF from the set of all DFs over the real line. Then, the* n*-sample* **order statistics** $X_{([n])}$ *is:*

$$X_{([n])}(\ (X_1, X_2, \ldots, X_n)\ ) := \big(X_{(1)}, X_{(2)}, \ldots X_{(n)}\big)\ ,\ \textit{such that, } X_{(1)} \leq X_{(2)} \leq \ldots \leq X_{(n)}\ . \quad (5.6)$$

For brevity, we write $X_{([n])}(\ (X_1, X_2, \ldots, X_n)\ )$ *as* $X_{([n])}$ *and its realisation* $X_{([n])}(\ (x_1, x_2, \ldots, x_n)\ )$ *as* $x_{([n])} = (x_{(1)}, x_{(2)}, \ldots x_{(n)})$*.*

Without going into the details of how to sort the data in ascending order to obtain the order statistics (an elementary topic of an Introductory Computer Science course), we simply use MATLAB's function `sort` to obtain the order statistics, as illustrated in the following example.

Labwork 12 *The order statistics for the five samples stored in* `XsFromUni01Twstr101` *from Labwork 10 can be computed using* `sort` *as follows:*

```
>> disp(XsFromUni01Twstr101); % display the sample points
    0.5164    0.5707    0.0285    0.1715    0.6853
>> SortedXsFromUni01Twstr101=sort(XsFromUni01Twstr101); % sort data
>> disp(SortedXsFromUni01Twstr101); % display the order statistics
    0.0285    0.1715    0.5164    0.5707    0.6853
```

Therefore, we can use `sort` *to obtain our order statistics* $x_{(1)}, x_{(2)}, \ldots, x_{(n)}$ *from* n *sample points* $x_1, x_2, \ldots, x_n$*.*

Next, we will introduce a family of common statistics, called the q^{th} quantile, by first defining the function:

Definition 29 (Inverse DF or Inverse CDF or Quantile Function) *Let* X *be an RV with DF* F*. The* **inverse DF** *or* **inverse CDF** *or* **quantile function** *is:*

$$F^{[-1]}(q) := \inf\{x : F(x) > q\}, \quad \textit{for some } q \in [0, 1]\ . \quad (5.7)$$

If F *is strictly increasing and continuous then* $F^{[-1]}(q)$ *is the unique* $x \in \mathbb{R}$ *such that* $F(x) = q$*.*

A **functional** is merely a function of another function. Thus, $T(F) : \{\text{All DFs }\} \mapsto \mathbb{T}$, being a map or function from the space of DFs to its range $\mathbb{T}$, is a functional. Some specific examples of functionals we have already seen include:

1. The **mean** of RV $X \sim F$ is a function of the DF F:
$$T(F) = \mathbf{E}(X) = \int x\, dF(x)\ .$$
2. The **variance** of RV $X \sim F$ is a function of the DF F:
$$T(F) = \mathbf{E}(X - \mathbf{E}(X))^2 = \int (x - \mathbf{E}(X))^2\, dF(x)\ .$$
3. The **value of DF at a given** $x \in \mathbb{R}$ of RV $X \sim F$ is also a function of DF F:
$$T(F) = F(x)\ .$$

Other functionals of F that depend on the quantile function $F^{[-1]}$ are:

1. The q^{th} **quantile** of RV $X \sim F$:
$$T(F) = F^{[-1]}(q) \quad \text{where } q \in [0,1] \ .$$
2. The **first quartile** or the 0.25^{th} **quantile** of the RV $X \sim F$:
$$T(F) = F^{[-1]}(0.25) \ .$$
3. The **median** or the **second quartile** or the 0.50^{th} **quantile** of the RV $X \sim F$:
$$T(F) = F^{[-1]}(0.50) \ .$$
4. The **third quartile** or the 0.75^{th} **quantile** of the RV $X \sim F$:
$$T(F) = F^{[-1]}(0.75) \ .$$

Definition 30 (Empirical Distribution Function (EDF or ECDF)) *Suppose we have n IID RVs, $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F$, where F is a DF from the set of all DFs over the real line. Then, the n-sample empirical distribution function (EDF or ECDF) is the discrete distribution function $\widehat{F}_n$ that puts a probability mass of $1/n$ at each sample or data point x_i:*

$$\widehat{F}_n(x) = \frac{\sum_{i=1}^n \mathbb{1}(X_i \leq x)}{n} \ , \qquad \textit{where} \qquad \mathbb{1}(X_i \leq x) := \begin{cases} 1 & \textit{if } x_i \leq x \\ 0 & \textit{if } x_i > x \end{cases} \tag{5.8}$$

Labwork 13 *Let us plot the ECDF for the five samples drawn from the $Uniform(0,1)$ RV in Labwork 10 using the* MATLAB *function* `ECDF` *(given in Labwork 49). Let us super-impose the samples and the true DF as depicted in Figure 5.3 with the following script:*

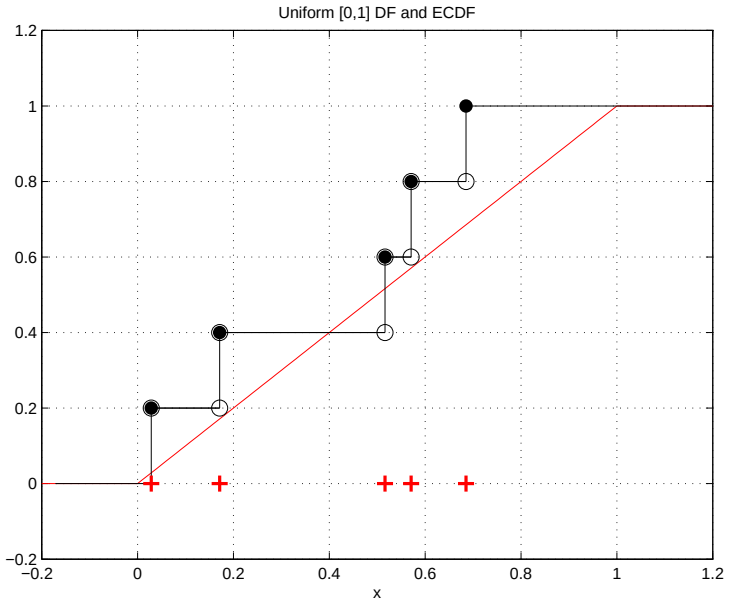
plotunifecdf.m
```
xs = -1:0.01:2; % vector xs from -1 to 2 with increment .05 for x values
% get the [0,1] uniform DF or cdf of xs in vector cdf
cdf=zeros(size(xs));% initialise cdf as zero
indices = find(xs>=1); cdf(indices) = 1; % set cdf as 1 when xs >= 1
indices = find(xs>=0 & xs<=1); cdf(indices)=xs(indices); % cdf=xs when 0 <= xs <= 1
plot(xs,cdf,'r') % plot the DF
hold on; title('Uniform [0,1] DF and ECDF'); xlabel('x'); axis([-0.2 1.2 -0.2 1.2])
x=[0.5164, 0.5707, 0.0285, 0.1715, 0.6853]; % five samples
plot(x,zeros(1,5),'r+','LineWidth',2,'MarkerSize',10)% plot the data as red + marks
hold on; grid on; % turn on grid
ECDF(x,1,.2,.6);% ECDF (type help ECDF) plot is extended to left and right by .2 and .4, respectively.
```

Definition 31 (q^{th} Sample Quantile) *For some $q \in [0,1]$ and n IID RVs $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F$, we can obtain the ECDF $\widehat{F}_n$ using* (10.1)*. The q^{th}* **sample quantile** *is defined as the statistic (statistical functional):*

$$T(\widehat{F}_n) = \widehat{F}_n^{[-1]}(q) := \inf\{x : \widehat{F}_n^{[-1]}(x) \geq q\} \ . \tag{5.9}$$

*By replacing q in this definition of the q^{th} sample quantile by 0.25, 0.5 or 0.75, we obtain the first, second (***sample median***) or third* **sample quartile***, respectively.*

Figure 5.3: Plot of the DF of Uniform$(0, 1)$, five IID samples from it, and the ECDF based on the five samples. Note that the ECDF $\widehat{F}_5$ for data points $x = (x_1, x_2, x_3, x_4, x_5) = (0.5164, 0.5707, 0.0285, 0.1715, 0.6853)$ jumps by $1/5 = 0.20$ at each of the five samples.



Algorithm 2 q^{th} Sample Quantile of Order Statistics

input:

1. q in the q^{th} sample quantile, i.e. the argument q of $\widehat{F}_n^{[-1]}(q)$,
2. order statistic $(x_{(1)}, x_{(2)}, \ldots, x_{(n)})$, i.e. the sorted $(x_1, x_2, \ldots, x_n)$, where $n > 0$.

output: $\widehat{F}_n^{[-1]}(q)$, the q^{th} sample quantile
$i \leftarrow \lfloor (n-1)q \rfloor$
$\delta \leftarrow (n-1)q - i$
if $i = n - 1$ **then**
    $\widehat{F}_n^{[-1]}(q) \leftarrow x_{(i+1)}$
else
    $\widehat{F}_n^{[-1]}(q) \leftarrow (1-\delta)x_{(i+1)} + \delta x_{(i+2)}$
end if
return: $\widehat{F}_n^{[-1]}(q)$

The following algorithm can be used to obtain the q^{th} sample quantile of n IID samples $(x_1, x_2, \ldots, x_n)$ on the basis of their order statistics $(x_{(1)}, x_{(2)}, \ldots, x_{(n)})$.

The q^{th} sample quantile, $\widehat{F}_n^{[-1]}(q)$, is found by interpolation from the order statistics $(x_{(1)}, x_{(2)}, \ldots, x_{(n)})$ of the n data points $(x_1, x_2, \ldots, x_n)$, using the formula:

$$\widehat{F}_n^{[-1]}(q) = (1-\delta)x_{(i+1)} + \delta x_{(i+2)}, \quad \text{where,} \quad i = \lfloor (n-1)q \rfloor \quad \text{and} \quad \delta = (n-1)q - \lfloor (n-1)q \rfloor \ .$$

Thus, the **sample minimum** of the data points $(x_1, x_2, \ldots, x_n)$ is given by $\widehat{F}_n^{[-1]}(0)$, the **sample maximum** is given by $\widehat{F}_n^{[-1]}(1)$ and the **sample median** is given by $\widehat{F}_n^{[-1]}(0.5)$, etc.

Labwork 14 *Use the implementation of Algorithm 2 in Labwork 50 as the* MATLAB *function* `qthSampleQuantile` *to find the q^{th} sample quantile of two simulated data arrays:*

1. `SortedXsFromUni01Twstr101`*, the order statistics that was constructed in Labwork 12 and*
2. *Another sorted array of* 7 *samples called* `SortedXs`

```
>> disp(SortedXsFromUni01Twstr101)
    0.0285    0.1715    0.5164    0.5707    0.6853
>> rand('twister',420);
>> SortedXs=sort(rand(1,7));
>> disp(SortedXs)
    0.1089    0.2670    0.3156    0.3525    0.4530    0.6297    0.8682
>> for q=[0, 0.25, 0.5, 0.75, 1.0]
       disp([q, qthSampleQuantile(q,SortedXsFromUni01Twstr101) ...
                qthSampleQuantile(q,SortedXs)])
   end
         0    0.0285    0.1089
    0.2500    0.1715    0.2913
    0.5000    0.5164    0.3525
    0.7500    0.5707    0.5414
    1.0000    0.6853    0.8682
```

5.2 Exploring Data and Statistics

5.2.1 Univariate Data

A **histogram** is a graphical representation of the frequency with which elements of a data array:

$$x = (x_1, x_2, \ldots, x_n) \ ,$$

of real numbers fall within each of the m intervals or **bins** of some **interval partition**:

$$b := (b_1, b_2, \ldots, b_m) := ([\underline{b}_1, \overline{b}_1], [\underline{b}_2, \overline{b}_2], \ldots, [\underline{b}_m, \overline{b}_m])$$

of the **data range** of x given by the closed interval:

$$\mathcal{R}(x) := [\min\{x_1, x_2, \ldots, x_n\}, \max\{x_1, x_2, \ldots, x_n\}] \ .$$

Elements of this partition b are called bins, their mid-points are called **bin centres**:

$$c := (c_1, c_2, \ldots, c_m) := ((\underline{b}_1 + \overline{b}_1)/2, (\underline{b}_2 + \overline{b}_2)/2, \ldots, (\underline{b}_m + \overline{b}_m)/2)$$

and their overlapping boundaries, i.e. $\overline{b}_i = \underline{b}_{i+1}$ for $1 \leq i < m$, are called **bin edges**:

$$d := (d_1, d_2, \ldots, d_{m+1}) := (\underline{b}_1, \underline{b}_2, \ldots, \underline{b}_{m-1}, \underline{b}_m, \overline{b}_m) \ .$$

For a given partition of the data range $\mathcal{R}(x)$ or some superset of $\mathcal{R}(x)$, three types of histograms are possible: frequency histogram, relative frequency histogram and density histogram. Typically, the partition b is assumed to be composed of m overlapping intervals of the same width $w = \overline{b}_i - \underline{b}_i$ for all $i = 1, 2, \ldots, m$. Thus, a histogram can be obtained by a set of bins along with their corresponding **heights**:

$$h = (h_1, h_2, \ldots, h_m) \ , \text{ where } h_k := g(\#\{x_i : x_i \in b_k\})$$

Thus, h_k, the height of the k-th bin, is some function g of the number of data points that fall in the bin b_k Formally, a histogram is a sequence of ordered pairs:

$$((b_1, h_1), (b_2, h_2), \ldots, (b_m, h_m)) \ .$$

Given a partition b, a **frequency histogram** is the histogram:

$$((b_1, h_1), (b_2, h_2), \ldots, (b_m, h_m)) \ , \text{ where } h_k := \#\{x_i : x_i \in b_k\} \ ,$$

a **relative frequency histogram** is the histogram:

$$((b_1, h_1), (b_2, h_2), \ldots, (b_m, h_m)) \ , \text{ where } h_k := n^{-1}\#\{x_i : x_i \in b_k\} \ ,$$

and a **density histogram** is the histogram:

$$((b_1, h_1), (b_2, h_2), \ldots, (b_m, h_m)) \ , \text{ where } h_k := (w_k n)^{-1}\#\{x_i : x_i \in b_k\} \ , w_k := \overline{b}_k - \underline{b}_k \ .$$

Labwork 15 *Let us use samples from the* `rand('twister',5489)` *as our data set x and plot various histograms. Let us use* `hist` *function (read* `help hist`*) to make a default histogram with ten bins.*

```
>> rand('twister',5489);
>> hist(x)
>> rand('twister',5489);
>> x=rand(1,100); % generate 100 PRNs
>> [Fs, Cs] = hist(x) % Cs is the bin centers and Fs is the frequencies of data set x
Fs =
     9    11    10     8     7    10     8    11    10    16
Cs =
    0.0598    0.1557    0.2516    0.3474    0.4433    0.5392    0.6351    0.7309    0.8268    0.9227
>> % produce a histogram plot the last argument 1 is the width value for immediately adjacent bars -- help bar
>> bar(Cs,Fs,1) % create a frequency histogram
>> bar(Cs,Fs/100,1) % create a relative frequency histogram
>> bar(Cs,Fs/(0.1*100),1) % create a density histogram (area of bars sum to 1)
>> sum(Fs/(0.1*100) .* ones(1,10)*0.1)
>> ans = 1
```

Try making a density histogram with 1000 samples from `rand` *with 15 bins. You can specify the number of bins by adding an extra argument to* `hist`*, for e.g.* `[Fs, Cs] = hist(x,15)` *will produce 15 bins of equal width over the data range $\mathcal{R}(x)$.*

Labwork 16 *We can also visualise the 100 data points in the array* `x` *using stem plot as well as the ECDF plots:*

```
>> rand('twister',5489);
>> x=rand(1,100); % produce 100 samples with rand
>> stem(x) % make a stem plot of the 100 data points in x
>> ECDF(x,2,.2,.6);% ECDF (type help ECDF) plot is extended to left and right by .2 and .4, respectively.
```

We can also visually summarise univariate data using the **box plot** or **box-whisker plot** available in the Stats Toolbox of MATLAB. These family of plots display a set of sample quantiles, typically they are include, the median, the first and third quartiles and the minimum and maximum values of our data array x.

5.2.2 Bivariate Data

By bivariate data array x we mean a $2 \times n$ matrix of real numbers or equivalently n ordered pairs of points $(x_{1,i}, x_{2,i})$ as $i = 1, 2, \ldots, n$. The most elementary visualisation of these n ordered pairs is in orthogonal Cartesian co-ordinates. Such plots are termed 2D **scatter plots** in statistics.

Labwork 17 *Generate a* 2×5 *array representing samples of* 5 *ordered pairs sampled uniformly at random over the unit square* $[0, 1] \times [0, 1]$*:*

```
>> rand('twister',5489);
>> x=rand(2,5)% create a sequence of 5 ordered pairs uniformly from unit square [0,1]X[0,1]
x =
    0.8147    0.1270    0.6324    0.2785    0.9575
    0.9058    0.9134    0.0975    0.5469    0.9649
>> plot(x(1,:),x(2,:),'x') % a 2D scatter plot with marker 'x'
```

2D histograms

Surface Plot

5.2.3 Trivariate Data

Similarly, we can make **3D scatter plots** as follows:

Labwork 18 *Repeat the visualisation below with a larger array, say* `x=rand(3,1000)`*, and use the rotate 3D feature in the Figure window to visually explore the samples in the unit cube. Do they seem to be uniformly distributed inside the unit cube?*

```
>> rand('twister',5489);
>> x=rand(3,5)% create a sequence of 5 ordered triples uniformly from unit cube [0,1]X[0,1]X[0,1]
x =
    0.8147    0.9134    0.2785    0.9649    0.9572
    0.9058    0.6324    0.5469    0.1576    0.4854
    0.1270    0.0975    0.9575    0.9706    0.8003
>> plot3(x(1,:),x(2,:),x(3,:),'x') % a 2D scatter plot with marker 'x'
```

Iso-surface Plots

5.2.4 Multivariate Data

Scatter Matrix

Parallel Co-ordinate Plot

Chapter 6

Common Random Variables

The Uniform$(0,1)$ RV of Model 2 forms the foundation for random variate generation and simulation. This is appropriately called the fundamental model or experiment, since every other experiment can be obtained from this one.

Next, we simulate or generate samples from other RVs by making the following two assumptions:

1. independent samples from the Uniform$(0,1)$ RV can be generated, and
2. real arithmetic can be performed exactly in a computer.

Both these assumptions are, in fact, not true and require a more careful treatment of the subject. We may return to these careful treatments later on.

6.1 Inversion Sampler for Continuous Random Variables

Proposition 3 *Let $F(x) := \int_{-\infty}^{x} f(y)\,dy : \mathbb{R} \to [0,1]$ be a continuous DF with density f, and let its inverse $F^{[-1]} : [0,1] \to \mathbb{R}$ be:*

$$F^{[-1]}(u) := \inf\{x : F(x) = u\} \ .$$

Then, $F^{[-1]}(U)$ has the distribution function F, provided U is a Uniform$(0,1)$ *RV. Recall* $\inf(A)$ *or infimum of a set A of real numbers is the greatest lower bound of every element of A.*

Proof: The "one-line proof" of the proposition is due to the following equalities:

$$\mathbf{P}(F^{[-1]}(U) \le x) = \mathbf{P}(\inf\{y : F(y) = U)\} \le x) = \mathbf{P}(U \le F(x)) = F(x), \quad \textit{for all } x \in \mathbb{R}.$$

This yields the inversion sampler or the inverse (C)DF sampler, where we (i) *generate* $u \sim$ Uniform$(0,1)$ and (ii) *return* $x = F^{[-1]}(u)$, as formalised by the following algorithm.

This algorithm emphasises the fundamental sampler's availability in an *input* step, and its set-up needs in an *initialise* step. In the following sections, we will not mention these universal steps; they will be taken for granted. The direct applicability of Algorithm 3 is limited to univariate densities for which the inverse of the cumulative distribution function is explicitly known. The next section will consider some examples.

Algorithm 3 Inversion Sampler or Inverse (C)DF Sampler

1: *input:* (1) $F^{[-1]}(x)$, inverse of the DF of the target RV X, (2) the fundamental sampler
2: *initialise:* set the seed, if any, for the fundamental sampler
3: *output:* a sample from X distributed according to F
4: *draw* $u \sim \text{Uniform}(0,1)$
5: *return:* $x = F^{[-1]}(u)$

6.2 Some Simulations of Continuous Random Variables

Model 3 (Uniform(θ_1, θ_2)) *Given two real parameters $\theta_1, \theta_2 \in \mathbb{R}$, such that $\theta_1 < \theta_2$, the PDF of the $Uniform(\theta_1, \theta_2)$ RV X is:*

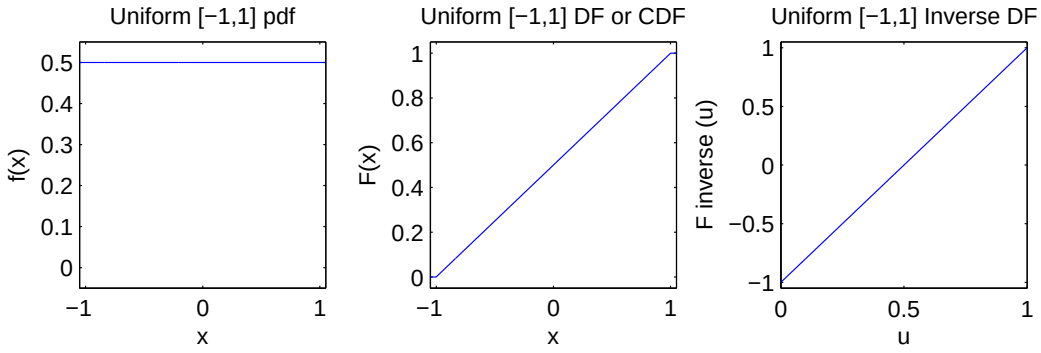
$$f(x; \theta_1, \theta_2) = \begin{cases} \frac{1}{\theta_2 - \theta_1} & \text{if } \theta_1 \leq x \leq \theta_2, \\ 0 & \text{otherwise} \end{cases} \tag{6.1}$$

and its DF given by $F(x; \theta_1, \theta_2) = \int_{-\infty}^{x} f(y; \theta_1, \theta_2)\, dy$ is:

$$F(x; \theta_1, \theta_2) = \begin{cases} 0 & \text{if } x < \theta_1 \\ \frac{x - \theta_1}{\theta_2 - \theta_1} & if\ \theta_1 \leq x \leq \theta_2, \\ 1 & \text{if } x > \theta_2 \end{cases} \tag{6.2}$$

Recall that we emphasise the dependence of the probabilities on the two parameters θ_1 and θ_2 by specifying them following the semicolon in the argument for f and F.

Figure 6.1: A plot of the PDF, DF or CDF and inverse DF of the Uniform$(-1, 1)$ RV X.



Simulation 1 (Uniform(θ_1, θ_2)) *To simulate from* Uniform(θ_1, θ_2) *RV X using the Inversion Sampler, we first need to find $F^{[-1]}(u)$ by solving for x in terms of $u = F(x; \theta_1, \theta_2)$:*

$$u = \frac{x - \theta_1}{\theta_2 - \theta_1} \iff x = (\theta_2 - \theta_1)u + \theta_1 \iff F^{[-1]}(u; \theta_1, \theta_2) = \theta_1 + (\theta_2 - \theta_1)u$$

Here is a simple implementation of the Inversion Sampler for the Uniform(θ_1, θ_2) *RV in* MATLAB*:*

```
>> rand('twister',786); % initialise the fundamental sampler for Uniform(0,1)
>> theta1=-1; theta2=1; % declare values for parameters theta1 and theta2
```

```
>> u=rand; % rand is the Fundamental Sampler and u is a sample from it
>> x=theta1+(theta2 - theta1)*u; % sample from Uniform(-1,1]) RV
>> disp(x); % display the sample from Uniform[-1,,1] RV
    0.5134
```

It is just as easy to draw n IID samples from Uniform(θ_1, θ_2) *RV X by transforming n IID samples from the* Uniform$(0,1)$ *RV as follows:*

```
>> rand('twister',786543); % initialise the fundamental sampler
>> theta1=-83; theta2=1004; % declare values for parameters a and b
>> u=rand(1,5); % now u is an array of 5 samples from Uniform(0,1)
>> x=theta1+(theta2 - theta1)*u; % x is an array of 5 samples from Uniform(-83,1004]) RV
>> disp(x); % display the 5 samples just drawn from Uniform(-83,1004) RV
  465.3065  111.4994   14.3535  724.8881  254.0168
```

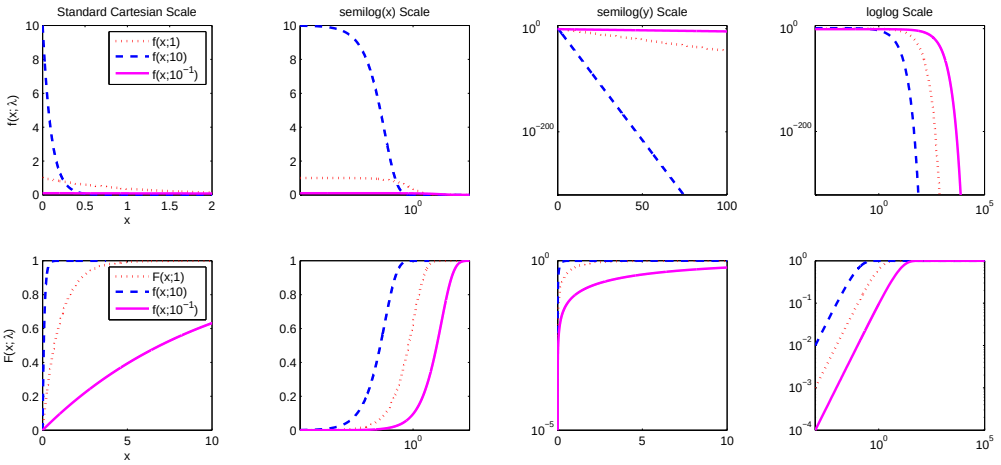
Model 4 (Exponential(λ)) *For a given $\lambda > 0$, an* Exponential(λ) *RV has the following PDF f and DF F:*

$$f(x;\lambda) = \lambda e^{-\lambda x} \qquad F(x;\lambda) = 1 - e^{-\lambda x} \ . \tag{6.3}$$

This distribution is fundamental because of its property of **memorylessness** *and plays a fundamental role in continuous time processes as we will see later.*

We encode the PDF and DF of the Exponential(λ) RV as MATLAB functions `ExponentialPdf` and `ExponentialCdf` and use them to produce Figure 6.2 in Labwork 48.

Figure 6.2: Density and distribution functions of *Exponential*(λ) RVs, for $\lambda = 1, 10, 10^{-1}$, in four different axes scales.



Mean and Variance of Exponential(λ)**:** Show that the mean of an Exponential(λ) RV X is:

$$\mathbf{E}_\lambda(X) = \int_0^\infty x f(x;\lambda)\, dx = \int_0^\infty x\lambda e^{-\lambda x}\, dx = \frac{1}{\lambda} \ ,$$

and the variance is:

$$\mathbf{V}_\lambda(X) = \left(\frac{1}{\lambda}\right)^2 \ .$$

Let us consider the problem of simulating from an Exponential(λ) RV with realisations in $\mathbb{R}_+ := [0,\infty) := \{x : x \geq 0, x \in \mathbb{R}\}$ to model the waiting time for a bus at a bus stop.

Simulation 2 (Exponential(λ)) *For a given $\lambda > 0$, an* Exponential(λ) *RV has the following PDF f, DF F and inverse DF $F^{[-1]}$:*

$$f(x;\lambda) = \lambda e^{-\lambda x} \qquad F(x;\lambda) = 1 - e^{-\lambda x} \qquad F^{[-1]}(u;\lambda) = \frac{-1}{\lambda}\log_e(1-u) \tag{6.4}$$

We write the natural logarithm $\log_e$ as log *for notational simplicity. An implementation of the Inversion Sampler for* Exponential(λ) *as a function in the M-file:*

ExpInvCDF.m

```
function x = ExpInvCDF(u,lambda);
% Return the Inverse CDF of Exponential(lambda) RV X
% Call Syntax: x = ExpInvCDF(u,lambda);
%              ExpInvCDF(u,lambda);
% Input      : lambda = rate parameter,
%              u = array of numbers in [0,1]
% Output     : x
x=-(1/lambda) * log(1-u);
```

We can simply call the function to draw a sample from, say the Exponential($\lambda = 1.0$) *RV by:*

```
lambda=1.0;                        % some value for lambda
u=rand;                          % rand is the Fundamental Sampler
ExpInvCDF(u,lambda)        % sample from Exponential(1) RV via function in ExpInvCDF.m
```

Because of the following:

$$U \sim \text{Uniform}(0,1) \quad \Longrightarrow \quad -U \sim \text{Uniform}(-1,0) \quad \Longrightarrow \quad 1-U \sim \text{Uniform}(0,1)\ ,$$

we could save a subtraction operation in the above algorithm by replacing `-(1/lambda) * log(1-u)` *by* `-(1/lambda) * log(u)`*. This is implemented as the following function.*

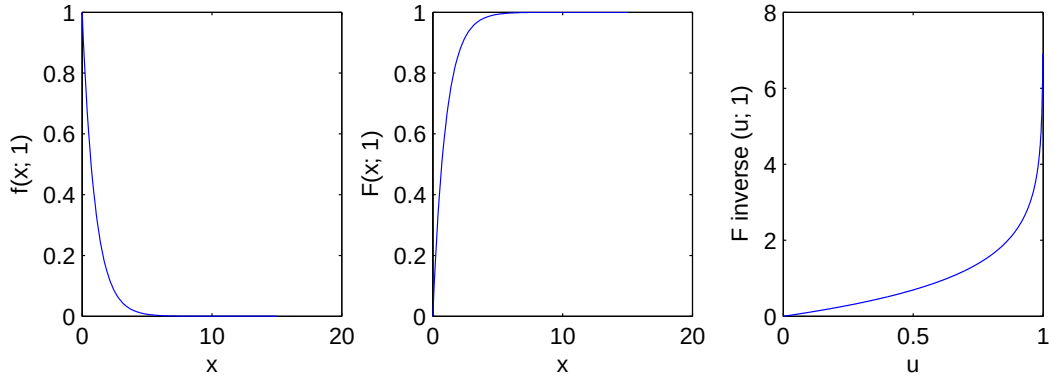
ExpInvSam.m

```
function x = ExpInvSam(u,lambda);
% Return the Inverse CDF based Sample from Exponential(lambda) RV X
% Call Syntax: x = ExpInvSam(u,lambda);
%              or ExpInvSam(u,lambda);
% Input      : lambda = rate parameter,
%              u = array of numbers in [0,1] from Uniform[0,1] RV
% Output     : x
x=-(1/lambda) * log(u);
```

```
>> rand('twister',46678); % initialise the fundamental sampler
>> Lambda=1.0;  % declare Lambda=1.0
>> x=ExpInvSam(rand(1,5),Lambda); % pass an array of 5 Uniform(0,1) samles from rand
>> disp(x); % display the Exponential(1.0) distributed samples
    0.5945    2.5956    0.9441    1.9015    1.3973
```

It is straightforward to do replicate experiments. Consider the experiment of drawing five independent samples from the Exponential($\lambda = 1.0$) RV. Suppose we want to repeat or replicate this experiment seven times and find the sum of the five outcomes in each of these replicates. Then we may do the following:

Figure 6.3: The PDF f, DF F, and inverse DF $F^{[-1]}$ of the the Exponential($\lambda = 1.0$) RV.




```
>> rand('twister',1973); % initialise the fundamental sampler
>> % store 7 replications of 5 IID draws from Exponential(1.0) RV in array a
>> lambda=1.0; a= -1/lambda * log(rand(5,7)); disp(a);
    0.7267    0.3226    1.2649    0.4786    0.3774    0.0394    1.8210
    1.2698    0.4401    1.6745    1.4571    0.1786    0.4738    3.3690
    0.4204    0.1219    2.2182    3.6692    0.9654    0.0093    1.7126
    2.1427    0.1281    0.8500    1.4065    0.1160    0.1324    0.2635
    0.6620    1.1729    0.6301    0.6375    0.3793    0.6525    0.8330
>> %sum up the outcomes of the sequence of 5 draws in each replicate
>> s=sum(a); disp(s);
    5.2216    2.1856    6.6378    7.6490    2.0168    1.3073    7.9990
```

Labwork 19 *Consider the problem of modelling the arrival of buses at a bus stop. Suppose that the time between arrivals is an* Exponential($\lambda = 0.1$) *RV X with a mean inter-arrival time of* $1/\lambda = 10$ *minutes. Suppose you go to your bus stop and zero a stop-watch. Simulate the times of arrival for the next seven buses as indicated by your stop-watch. Seed the fundamental sampler by your Student ID (eg. if your ID is* `11424620` *then type* `rand('twister', 11424620);` *just before the simulation). Hand in the code with the arrival times of the next seven buses at your ID-seeded bus stop.*

The support of the Exponential(λ) RV is $\mathbb{R}_+ := [0, \infty)$. Let us consider a RV built by mirroring the Exponential(λ) RV about the origin with the entire real line as its support.

Model 5 (Laplace(λ) **or** Double Exponential(λ) **RV)** *If a RV X is equally likely to be either positive or negative with an exponential density, then the* Laplace(λ) *or* Double Exponential(λ) *RV, with the rate parameter $\lambda > 0, \lambda \in \mathbb{R}$, may be used to model it. The density function for the* Laplace(λ) *RV given by $f(x; \lambda)$ is*

$$f(x; \lambda) = \frac{\lambda}{2} e^{-\lambda|x|} = \begin{cases} \frac{\lambda}{2} e^{\lambda x} & \text{if } x < 0 \\ \frac{\lambda}{2} e^{-\lambda x} & \text{if } x \geq 0 \end{cases} .$$

Let us define the sign of a real number x by

$$\text{sign}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \, . \end{cases}$$

Then, the DF of the Laplace(λ) *RV* X *is*

$$F(x;\lambda) = \int_{-\infty}^{x} f(y;\lambda)\,dy = \frac{1}{2}\left(1 + \text{sign}(x)\left(1 - e^{-\lambda|x|}\right)\right) \ , \tag{6.5}$$

and its inverse DF is

$$F^{[-1]}(u;\lambda) = -\frac{1}{\lambda}\ \text{sign}\left(u - \frac{1}{2}\right)\log\left(1 - 2\left|u - \frac{1}{2}\right|\right) \ , \ u \in [0,1] \tag{6.6}$$

Mean and Variance of Laplace(λ) **RV** X**:** Show that the mean of a Laplace(λ) RV X is

$$\mathbf{E}(X) = \int_0^\infty x f(x;\lambda)\,dx = \int_0^\infty x\frac{\lambda}{2}e^{-\lambda|x|}\,dx = 0 \ ,$$

and the variance is

$$\mathbf{V}(X) = \left(\frac{1}{\lambda}\right)^2 + \left(\frac{1}{\lambda}\right)^2 = 2\left(\frac{1}{\lambda}\right)^2 \ .$$

Note that the mean is 0 due to the symmetry of the density about 0 and the variance is twice that of the Exponential(λ) RV.

Simulation 3 (*Laplace*(λ)**)** *Here is an implementation of an inversion sampler to draw IID samples from a* Laplace(λ) *RV* X *by transforming IID samples from the* Uniform$(0,1)$ *RV* U*:*

LaplaceInvCDF.m

```
function x = LaplaceInvCDF(u,lambda);
% Call Syntax: x = LaplaceInvCDF(u,lambda);
%              or LaplaceInvCDF(u,lambda);
%
% Input      : lambda = rate parameter > 0,
%              u = an 1 X n array of IID samples from Uniform[0,1] RV
% Output     : an 1Xn array x of IID samples from Laplace(lambda) RV
%                               or Inverse CDF of Laplace(lambda) RV
x=-(1/lambda)*sign(u-0.5) .* log(1-2*abs(u-0.5));
```

We can simply call the function to draw a sample from, say the Laplace($\lambda = 1.0$) *RV by*

```
>> lambda=1.0;               % some value for lambda
>> rand('twister',6567);       % initialize the fundamental sampler
>> u=rand(1,5);               % draw 5 IID samples from Uniform(0,1) RV
>> disp(u);                % display the samples in u
    0.6487    0.9003    0.3481    0.6524    0.8152

>> x=LaplaceInvCDF(u,lambda); % draw 5 samples from Laplace(1) RV using inverse CDF
>> disp(x);                   % display the samples
    0.3530    1.6127   -0.3621    0.3637    0.9953
```

Next, let us become familiar with an RV for which the expectation does not exist. This will help us appreciate the phrase "none of which is dominant" in the informal statement of the CLT later.

Model 6 (Cauchy**)** *The density of the* Cauchy *RV* X *is:*

$$f(x) = \frac{1}{\pi(1+x^2)}, \qquad -\infty < x < \infty \ ,$$

and its DF is:

$$F(x) = \frac{1}{\pi}\tan^{-1}(x) + \frac{1}{2} \ . \tag{6.7}$$

Randomly spinning a LASER emitting improvisation of "Darth Maul's double edged lightsaber" that is centered at $(1,0)$ *in the plane* $\mathbb{R}^2$ *and recording its intersection with the* y*-axis, in terms of the* y *coordinates, gives rise to the Standard Cauchy RV.*

Mean of Cauchy **RV:** The expectation of the Cauchy RV X, obtained via integration by parts (set $u = x$ and $v = tan^{-1}(x)$) does not exist , since:

$$\int |x| \; dF(x) = \frac{2}{\pi} \int_0^\infty \frac{x}{1+x^2} \, dx = \left(x \tan^{-1}(x) \right]_0^\infty - \int_0^\infty \tan^{-1}(x) \, dx = \infty \; . \tag{6.8}$$

Variance and higher moments cannot be defined when the expectation itself is undefined.

Simulation 4 (Cauchy**)** *We can draw n IID samples from the* Cauchy *RV X by transforming n IID samples from* Uniform$(0, 1)$ *RV U using the inverse DF as follows:*

```
>> rand('twister',2435567);        % initialise the fundamental sampler
>> u=rand(1,5);                        % draw 5 IID samples from Uniform(0,1) RV
>> disp(u);                         % display the samples in u
    0.7176    0.6655    0.9405    0.9198    0.2598
>> x=tan(pi * u);     % draw 5 samples from Standard cauchy RV using inverse CDF
>> disp(x);  % display the samples in x
   -1.2272   -1.7470   -0.1892   -0.2575    1.0634
```

Recall that the mean of the Cauchy RV X does not exist since $\int |x| \; dF(x) = \infty$ (6.8). We will investigate this in Labwork 20.

Labwork 20 *Let us see what happens when we plot the running sample mean for an increasing sequence of IID samples from the Standard Cauchy RV X by implementing the following script file:*

PlotStandardCauchyRunningMean.m

```
% script to plot the oscillating running mean of Std Cauchy samples
% relative to those for the Uniform(0,10) samples
rand('twister',25567);        % initialize the fundamental sampler
for i=1:5
N = 10^5;                          % maximum sample size
u=rand(1,N);             % draw N IID samples from Uniform(0,1)
x=tan(pi * u);     % draw N IID samples from Standard cauchy RV using inverse CDF
n=1:N;              % make a vector n of current sample size [1 2 3 ... N-1 N]
CSx=cumsum(x); % CSx is the cumulative sum of the array x (type 'help cumsum')
% Runnign Means <- vector division of cumulative sum of samples by n
RunningMeanStdCauchy = CSx ./ n; % Running Mean for Standard Cauchy samples
RunningMeanUnif010 = cumsum(u*10.0) ./ n; % Running Mean for Uniform(0,10) samples
semilogx(n, RunningMeanStdCauchy) %
hold on;
semilogx(n, RunningMeanUnif010, 'm')
end
xlabel('n = sample size');
ylabel('Running mean from n samples')
```

The resulting plot is shown in Figure 6.4. Notice that the running means or the sample mean of n samples as a function of n, for each of the five replicate simulations, never settles down to a particular value. This is because of the "thick tails" of the density function for this RV which produces extreme observations. Compare them with the running means, based on n IID samples from the Uniform$(0, 10)$ *RV, for each of five replicate simulations (magenta lines). The latter sample means have settled down stably to the mean value of* 5 *after about* 700 *samples.*

For a continuous RV X with a closed-form expression for the inverse DF $F^{[-1]}$, we can employ Algorithm 3 to draw samples from X. Table 6.1 summarises some random variables that are amenable to Algorithm 3.

Next, we familiarise ourselves with the Gaussian or Normal RV.

Figure 6.4: Unending fluctuations of the running means based on n IID samples from the Standard Cauchy RV X in each of five replicate simulations (blue lines). The running means, based on n IID samples from the Uniform$(0, 10)$ RV, for each of five replicate simulations (magenta lines).

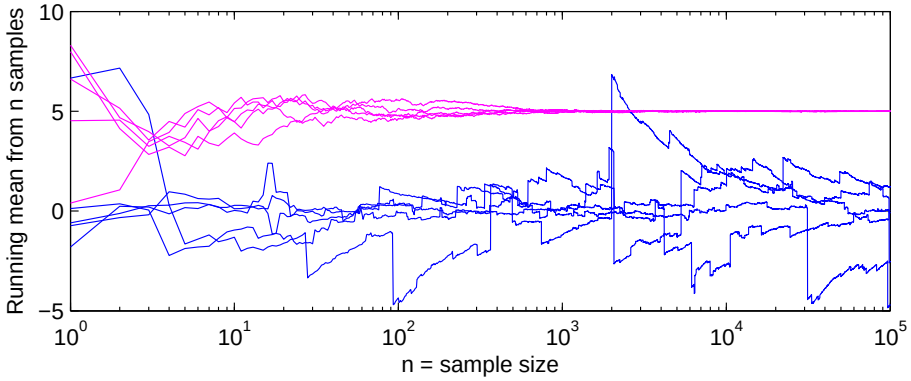


Table 6.1: Some continuous RVs that can be simulated from using Algorithm 3.

Random Variable X	$F(x)$	$X = F^{[-1]}(U), \quad U \sim \text{Uniform}(0,1)$	Simplified form
Uniform(a, b)	(6.2)	$a + (b-a)U$	–
Exponential(λ)	(6.3)	$\frac{-1}{\lambda}\log(1-U)$	$\frac{-1}{\lambda}\log(U)$
Laplace(λ)	(6.6)	$-\frac{1}{\lambda}\ \text{sign}\left(U-\frac{1}{2}\right)\log\left(1-2\left\lvert U-\frac{1}{2}\right\rvert\right)$	–
Cauchy	(6.7)	$\tan\left(\pi\left(U-\frac{1}{2}\right)\right)$	$\tan(\pi U)$

Model 7 (Normal(μ, σ^2)) *X has a* Normal(μ, σ^2) *or* Gaussian(μ, σ^2) *distribution with the location parameter $\mu \in \mathbb{R}$ and the scale or variance parameter $\sigma^2 > 0$, if:*

$$f(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right), \qquad x \in \mathbb{R} \tag{6.9}$$

Normal$(0, 1)$ *distributed RV, which plays a fundamental role in asymptotic statistics, is conventionally denoted by Z. Z is said to have the* **Standard Normal** *distribution with PDF $f(z; 0, 1)$ and DF $F(z; 0, 1)$ conventionally denoted by $\phi(z)$ and $\Phi(z)$, respectively.*

There is no closed form expression for $\Phi(z)$ or $F(x; \mu, \sigma)$. The latter is simply defined as:

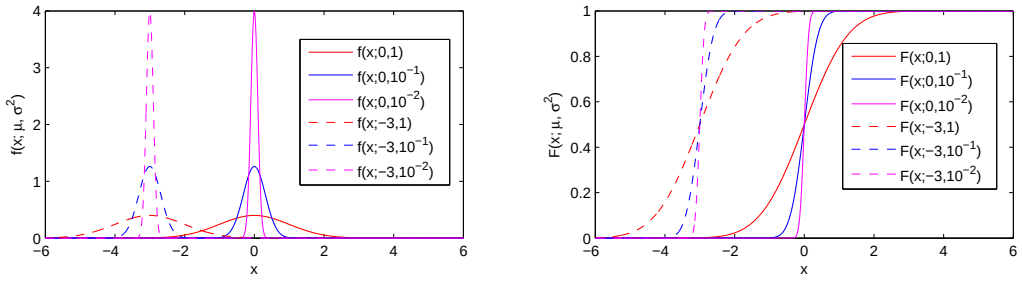
$$F(x; \mu, \sigma^2) = \int_{-\infty}^{x} f(y; \mu, \sigma)\, dy$$

*We can express $F(x; \mu, \sigma^2)$ in terms of the error function (*erf*) as follows:*

$$F(x; \mu, \sigma^2) = \frac{1}{2}\ \text{erf}\left(\frac{x-\mu}{\sqrt{2\sigma^2}}\right) + \frac{1}{2} \tag{6.10}$$

We implement the PDF (6.9) and DF (6.10) for a Normal(μ, σ^2) RV X as MATLAB functions `NormalPdf` and `NormalCdf`, respectively, in Labwork 47, and then produce their plots for various Normal(μ, σ^2) RVs, shown in Figure 6.5. Observe the concentration of probability mass, in terms of the PDF and DF plots, about the location parameter μ as the variance parameter σ^2 decreases.

Figure 6.5: Density and distribution function of several Normal(μ, σ^2) RVs.



Mean and Variance of Normal(μ, σ^2)**:** The mean of a Normal(μ, σ^2) RV X is:

$$\mathbf{E}(X) = \int_{-\infty}^{\infty} x f(x; \mu, \sigma^2)\, dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right) dx = \mu \ ,$$

and the variance is:

$$\mathbf{V}(X) = \int_{-\infty}^{\infty} (x-\mu)^2 f(x; \mu, \sigma^2)\, dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} (x-\mu)^2 \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right) dx = \sigma^2 \ .$$

Labwork 21 *Write a function to evaluate the* $\mathbf{P}(X \in [a, b])$ *for the* Normal$(0, 1)$ *RV* X *for user-specified values of* a *and* b*. [Hint: one option is by making two calls to* `NormalCdf` *and doing one arithmetic operation.]*

Simulations 1 and 2, 3 and 4 produce samples from a continuous RV X with a closed-form expression for the inverse DF $F^{[-1]}$ via Algorithm 3 (Table 6.1). But only a few RVs have an explicit $F^{[-1]}$. For example, Normal$(0, 1)$ RV does not have an explicit $F^{[-1]}$. Algorithm 4 is a more general but inexact method that relies on an approximate numerical solution of x, for a given u, that satisfies the equation $F(x) = u$.

Algorithm 4 Inversion Sampler by Numerical Solution of $F(X) = U$ via Newton-Raphson Method

input: $F(x)$, the DF of the target RV X
input: $f(x)$, the density of X
input: A reasonable `Stopping Rule`,
  e.g. a specified tolerance $\epsilon > 0$ and a maximum number of iterations `MAX`
input: a careful mechanism to specify x_0
output: a sample from X distributed according to F
draw: $u \sim$ Uniform$(0, 1)$
initialise: $i \leftarrow 0$, $x_i \leftarrow x_0$, $x_{i+1} \leftarrow x_0 - \frac{F(x_0)-u}{f(x_0)}$
while `Stopping Rule` is not satisfied,
  e.g. $|F(x_i) - F(x_{i-1})| > \epsilon$ AND $i <$ `MAX` **do**
    $x_i \leftarrow x_{i+1}$
    $x_{i+1} \leftarrow \left(x_i - \frac{F(x_i)-u}{f(x_i)}\right)$
    $i \leftarrow i + 1$
end while
return: $x \leftarrow x_i$

Simulation 5 (Normal(μ, σ^2)**)** *We may employ Algorithm 4 to sample from the Normal*(μ, σ^2) *RV* X *using the following function.*

Sample1NormalByNewRap.m

```
function x = Sample1NormalByNewRap(u,Mu,SigmaSq)
% Returns a sample from Normal(Mu, SigmaSq)
% Newton-Raphson numerical solution of F(x)=u
% Input: u = one random Uniform(0,1) sample
%        Mu = Mean of Normal(Mu, SigmaSq)
%        SigmaSq = Variance of Normal(Mu, SigmaSq)
% Usage: x = Sample1NormalByNewRap(u,Mu,SigmaSq)
% To transform an array Us of uniform samples to array Xs of Normal samples via arrayfun
%        Xs = arrayfun(@(u)(Sample1NormalByNewRap(u,-100.23,0.01)),Us);
Epsilon=1e-5; % Tolerance in stopping rule
MaxIter=10000; % Maximum allowed iterations in stopping rule
x=0; % initialize the output x as 0
% initialize i, xi, and xii
i=0;        % Mu is an ideal initial condition since F(x; Mu, SigmaSq)
xi = Mu;    % is convex when x < Mu and concave when x > Mu and the
% Newton-Raphson method started at Mu converges
xii = xi - (NormalCdf(xi,Mu,SigmaSq)-u)/NormalPdf(xi,Mu,SigmaSq);
% Newton-Raphson Iterations
while (abs(NormalCdf(xii,Mu,SigmaSq)-NormalCdf(xi,Mu,SigmaSq))...
        > Epsilon & i < MaxIter),
    xi = xii;
    xii = xii - (NormalCdf(xii,Mu,SigmaSq)-u)/NormalPdf(xii,Mu,SigmaSq);
    i=i+1;
end
x=xii; % record the simulated x from the j-th element of u
```

We draw five samples from the Normal$(0,1)$ *RV* Z *and store them in* z *as follows. The vector* z *can be obtained by a Newton-Raphson-based numerical transformation of the vector* u *of* 5 *IID samples from the* Uniform$(0,1)$ *RV. We simply need to apply the function* `Sample1NormalByNewRap` *to each element of an arra of* $Unifrm(0,1)$ *samples.* MATLAB*'s* `arrayfun` *command can be used to apply* `@(u)(Sample1NormalByNewRap(u,0,1))` *(i.e.,* `Sample1NormalByNewRap` *as a function of* u*) to every element of our array of* Uniform$(0,1)$ *samples, say* `Us`*. Note that* $F(z)$ *is the same as the drawn* u *from* U *at least up to four significant digits.*

```
>> rand('twister',563987);
>> Us=rand(1,5); % store 5 samples from Uniform(0,1) RV in array Us
>> disp(Us); % display Us
    0.8872    0.2569    0.5275    0.8650    0.8517
>> z=Sample1NormalByNewRap(u(1),0,1); %transform Us(1) to a Normal(0,1) sample z
>> disp(z); % display z
    1.2119
>> z = arrayfun(@(u)(Sample1NormalByNewRap(u,0,1)),Us); %transform array Us via arrayfun
>> % dislay array z obtained from applying Sample1NormalByNewRap to each element of Us
>> disp(z);
    1.2119   -0.6530    0.0691    1.1031    1.0439
>> % check that numerical inversion of F worked, i.e., is F(z)=u ?
>> disp(NormalCdf(z,0,1));
    0.8872    0.2569    0.5275    0.8650    0.8517
```

Next we draw five samples from the Normal$(-100.23, 0.01)$ *RV* X*, store it in an array* x *and observe that the numerical method is reasonably accurate by the equality of* u *and* $F(x)$*.*

```
>> rand('twister',563987);
>> disp(Us); % display Us
    0.8872    0.2569    0.5275    0.8650    0.8517
>> % transform array Us via arrayfun
>> x = arrayfun(@(u)(Sample1NormalByNewRap(u,-100.23,0.01)),Us);
>> disp(x);
```

```
 -100.1088 -100.2953 -100.2231 -100.1197 -100.1256
>> disp(NormalCdf(x,-100.23,0.01));
    0.8872    0.2569    0.5275    0.8650    0.8517
```

One has to be extremely careful with this approximate simulation algorithm implemented in floating-point arithmetic. More robust samplers for the Normal(μ, σ^2) *RV exist. However, Algorithm 4 is often the only choice when simulating from an arbitrary RV with an unknown closed-form expression for its* $F^{[-1]}$.

Next, we use our simulation capability to gain an informal and intuitive understanding of one of the most elementary theorems in probability and statistics, namely, the Central Limit Theorem (CLT). We will see a formal treatment of CLT later.

Informally, the CLT can be stated as follows:
"The sample mean of a large number of IID samples, none of which is dominant, tends to the Normal distribution as the number of samples increases."

Labwork 22 *Let us investigate the histograms from* 10000 *simulations of the sample mean of* $n = 10, 100, 1000$ *IID* Exponential$(\lambda = 0.1)$ *RVs as follows:*

```
>> rand('twister',1973); % initialise the fundamental sampler
>> % a demonstration of Central Limit Theorem (CLT) -- Details of CLT are in the sequel
>> % the sample mean should be a Normal(1/lambda,lambda/n) RV
>> lambda=0.1; Reps=10000; n=10; hist(sum(-1/lambda * log(rand(n,Reps)))/n)
>> lambda=0.1; Reps=10000; n=100; hist(sum(-1/lambda * log(rand(n,Reps)))/n,20)
>> lambda=0.1; Reps=10000; n=1000; hist(sum(-1/lambda * log(rand(n,Reps)))/n,20)
```

Do you see a pattern in the histograms ?

See the histograms generated from the following code that produces sample means from the Cauchy *RV:*

```
>> Reps=10000; n=1000; hist(sum(tan(pi * rand(n,Reps)))/n,20)
>> Reps=10000; n=1000; hist(sum(tan(pi * rand(n,Reps)))/n,20)
>> Reps=10000; n=1000; hist(sum(tan(pi * rand(n,Reps)))/n,20)
```

Classwork 12 *Explain in words why the mean of n IID samples from the* Cauchy *RV is* **not** *obeying the Central Limit Theorem. Also relate it to Figure 6.4 of Labwork 20.*

6.3 Inversion Sampler for Discrete Random Variables

Next, consider the problem of **sampling from a random variable X with a discontinuous or discrete DF** using the inversion sampler. We need to define the inverse more carefully here.

Proposition 4 *Let the support of the RV X be over some real interval $[a, b]$ and let its inverse DF be defined as follows:*

$$F^{[-1]}(u) := \inf\{x \in [a, b] : F(x) \geq u,\ 0 \leq u \leq 1\} \ .$$

If $U \sim$ Uniform$(0, 1)$ *then* $F^{[-1]}(U)$ *has the DF F, i.e.* $F^{[-1]}(U) \sim F \sim X$.

Proof: The proof is a consequence of the following equalities:

$$\mathbf{P}(F^{[-1]}(U) \leq x) = \mathbf{P}(U \leq F(x)) = F(x) := \mathbf{P}(X \leq x)$$

6.4 Some Simulations of Discrete Random Variables

Simulation 6 (Bernoulli(θ)**)** *Consider the problem of simulating from a* Bernoulli(θ) *RV based on an input from a* Uniform$(0,1)$ *RV. Recall that* $\lfloor x \rfloor$ *(called the 'floor of* x*') is the largest integer that is smaller than or equal to* x*, e.g.* $\lfloor 3.8 \rfloor = 3$*. Using the floor function, we can simulate a* Bernoulli(θ) *RV* X *as follows:*

```
>>  theta = 0.3;               % set theta = Prob(X=1)
  % return x  -- floor(y) is the largest integer less than or equal to y
>> x = floor(rand + theta)        % rand is the Fundamental Sampler
>> disp(x) % display the outcome of the simulation
     0
>> n=10;  % set the number of IID Bernoulli(theta=0.3) trials you want to simulate
>> x = floor(rand(1,10)+theta); % vectorize the operation
>> disp(x) % display the outcomes of the simulation
     0     0     1     0     0     0     0     0     1     1
```

Again, it is straightforward to do replicate experiments, e.g. to demonstrate the Central Limit Theorem for a sequence of n IID Bernoulli(θ) trials.

```
>> % a demonstration of Central Limit Theorem --
>> % the sample mean of a sequence of n IID Bernoulli(theta) RVs is Gaussian(theta,theta(1-theta)/n)
>> theta=0.5; Reps=10000; n=10; hist(sum(floor(rand(n,Reps)+theta))/n)
>> theta=0.5; Reps=10000; n=100; hist(sum(floor(rand(n,Reps)+theta))/n,20)
>> theta=0.5; Reps=10000; n=1000; hist(sum(floor(rand(n,Reps)+theta))/n,30)
```

Consider the class of discrete RVs with distributions that place all probability mass on a single real number. This is the probability model for the deterministic real variable.

Model 8 (Point Mass(θ)**)** *Given a specific point* $\theta \in \mathbb{R}$*, we say an RV* X *has point mass at* θ *or is* Point Mass(θ) *distributed if the DF is:*

$$F(x;\theta) = \begin{cases} 0 & \text{if } x < \theta \\ 1 & \text{if } x \geq \theta \end{cases} \tag{6.11}$$

and the PMF is:

$$f(x;\theta) = \begin{cases} 0 & \text{if } x \neq \theta \\ 1 & \text{if } x = \theta \end{cases} \tag{6.12}$$

Thus, Point Mass(θ) RV X is deterministic in the sense that every realisation of X is exactly equal to $\theta \in \mathbb{R}$. We will see that this distribution plays a central limiting role in asymptotic statistics.

Mean and variance of Point Mass(θ) **RV:** Let $X \sim$ Point Mass(θ). Then:

$$\mathbf{E}(X) = \sum_x x f(x) = \theta \times 1 = \theta \ , \qquad \mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2 = \theta^2 - \theta^2 = 0 \ .$$

Simulation 7 (Point Mass(θ)**)** *Let us simulate a sample from the* Point Mass(θ) *RV* X*. Since this RV produces the same realisation* θ *we can implement it via the following M-file:*

Sim1PointMass.m

```
function  x = Sim1PointMass(u,theta)
% Returns one sample from the Point Mass(theta) RV X
% Call Syntax:  x = SimPointMass(u,theta);
% Input      : u = one uniform random number eg. rand()
%              theta = a real number (scalar)
% Output     : x = sample from X
x=theta;
```

Here is call to the function.

```
>> Sim1PointMass(rand(),2)
ans =     2
>> % % we can use arrayfun to apply Sim1Pointmass to any array of Uniform(0,1) samples
>> arrayfun(@(u)(Sim1PointMass(u,17)),rand(2,10))
ans =
    17    17    17    17    17    17    17    17    17    17
    17    17    17    17    17    17    17    17    17    17
```

Note that it is not necessary to have input IID samples from Uniform$(0,1)$ *RV via* `rand` *in order to draw samples from the* Point Mass(θ) *RV. For instance, an input matrix of zeros can do the job:*

```
>> arrayfun(@(u)(Sim1PointMass(u,17)),zeros(2,8))
ans =
    17    17    17    17    17    17    17    17
    17    17    17    17    17    17    17    17
```

Next let us consider a natural generalization of the Bernoulli(θ) RV with more than two outcomes.

Model 9 (de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$**)** *Given a specific point* $(\theta_1, \theta_2, \ldots, \theta_k)$ *in the* k*-Simplex:*

$$\triangle_k := \{\, (\theta_1, \theta_2, \ldots, \theta_k) : \theta_1 \geq 0, \theta_2 \geq 0, \ldots, \theta_k \geq 0, \sum_{i=1}^{k} \theta_i = 1 \,\} \ ,$$

we say that an RV X *is* de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ *distributed if its PMF is:*

$$f(x; \theta_1, \theta_2, \ldots, \theta_k) = \begin{cases} 0 & \text{if } x \notin [k] := \{1, 2, \ldots, k\}, \\ \theta_x & \text{if } x \in [k]. \end{cases}$$

The DF for de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ *RV* X *is:*

$$F(x; \theta_1, \theta_2, \ldots, \theta_k) = \begin{cases} 0 & \text{if } -\infty < x < 1 \\ \theta_1 & \text{if } 1 \leq x < 2 \\ \theta_1 + \theta_2 & \text{if } 2 \leq x < 3 \\ \vdots & \\ \theta_1 + \theta_2 + \cdots + \theta_{k-1} & \text{if } k-1 \leq x < k \\ \theta_1 + \theta_2 + \cdots + \theta_{k-1} + \theta_k = 1 & \text{if } k \leq x < \infty \end{cases} \tag{6.13}$$

The de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ *RV can be thought of as a probability model for "the outcome of rolling a polygonal cylindrical die with* k *rectangular faces that are marked with* $1, 2, \ldots, k$*". The parameters* $\theta_1, \theta_2, \ldots, \theta_k$ *specify how the die is loaded and may be idealised as specifying the cylinder's centre of mass with respect to the respective faces. Thus, when* $\theta_1 = \theta_2 = \cdots = \theta_k = 1/k$*, we have a probability model for the outcomes of a fair die.*

Mean and variance of de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ **RV:** The not too useful expressions for the first two moments of $X \sim$ de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ are,

$$\mathbf{E}(X) = \sum_{x=1}^{k} x\theta(x) = \theta_1 + 2\theta_2 + \cdots + k\theta_k \ , \text{ and}$$

$$\mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2 = \left(\theta_1 + 2^2\theta_2 + \cdots + k^2\theta_k\right) - \left(\theta_1 + 2\theta_2 + \cdots + k\theta_k\right)^2 \ .$$

However, if $X \sim$ de Moivre$(1/k, 1/k, \ldots, 1/k)$, then the mean and variance for the fair k-faced die based on Faulhaber's formula for $\sum_{i=1}^{k} i^m$, with $m \in \{1, 2\}$, are,

$$\mathbf{E}(X) = \frac{1}{k}\left(1 + 2 + \cdots + k\right) = \frac{1}{k}\frac{k(k+1)}{2} = \frac{k+1}{2} \ ,$$

$$\mathbf{E}(X^2) = \frac{1}{k}\left(1^2 + 2^2 + \cdots + k^2\right) = \frac{1}{k}\frac{k(k+1)(2k+1)}{6} = \frac{2k^2+3k+1}{6} \ ,$$

$$\begin{aligned}\mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2 &= \frac{2k^2+3k+1}{6} - \left(\frac{k+1}{2}\right)^2 = \frac{2k^2+3k+1}{6} - \left(\frac{k^2+2k+1}{4}\right) \\ &= \frac{8k^2+12k+4-6k^2-12k-6}{24} = \frac{2k^2-2}{24} = \frac{k^2-1}{12} \ .\end{aligned}$$

Next we simulate from de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ RV X via its inverse DF

$$F^{[-1]} : [0,1] \to [k] := \{1, 2, \ldots, k\} \ ,$$

given by:

$$F^{[-1]}(u; \theta_1, \theta_2, \ldots, \theta_k) = \begin{cases} 1 & \text{if } 0 \le u < \theta_1 \\ 2 & \text{if } \theta_1 \le u < \theta_1 + \theta_2 \\ 3 & \text{if } \theta_1 + \theta_2 \le u < \theta_1 + \theta_2 + \theta_3 \\ \vdots & \\ k & \text{if } \theta_1 + \theta_2 + \cdots + \theta_{k-1} \le u < 1 \end{cases}$$

When $k = 2$ in the de Moivre(θ_1, θ_2) model, we have an RV that is similar to the Bernoulli$(p = \theta_1)$ RV. The DF F and its inverse $F^{[-1]}$ for a specific $\theta_1 = 0.3$ are depicted in Figure 6.6.

First we simulate from an equi-probable special case of the de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ RV, with $\theta_1 = \theta_2 = \cdots = \theta_k = 1/k$.

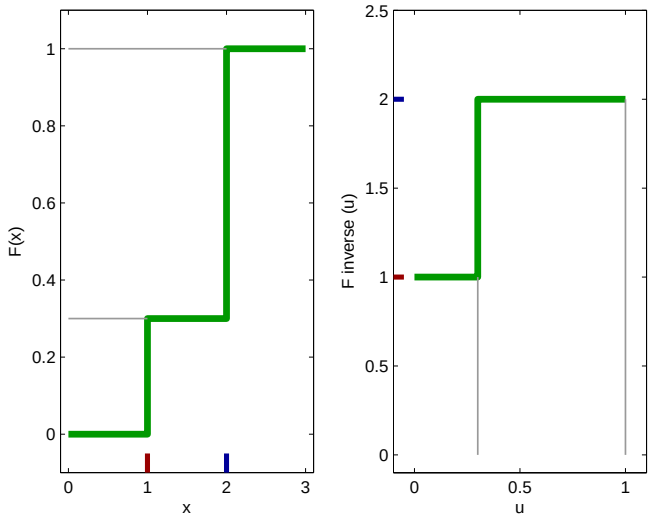
Simulation 8 (de Moivre$(1/k, 1/k, \ldots, 1/k)$**)** *The equi-probable de Moivre$(1/k, 1/k, \ldots, 1/k)$ RV X with a discrete uniform distribution over $[k] = \{1, 2, \ldots k\}$ can be efficiently sampled using the ceiling function. Recall that $\lceil y \rceil$ is the smallest integer larger than or equal to y, eg. $\lceil 13.1 \rceil = 14$. Algorithm 5 produces samples from the de Moivre$(1/k, 1/k, \ldots, 1/k)$ RV X.*

The M-file implementing Algorithm 5 is:

SimdeMoivreEqui.m

```
function x = SimdeMoivreEqui(u,k);
% return samples from de Moivre(1/k,1/k,...,1/k) RV X
% Call Syntax:  x = SimdeMoivreEqui(u,k);
% Input      : u = array of uniform random numbers eg. rand
%              k = number of equi-probabble outcomes of X
% Output     : x = samples from X
x = ceil(k * u) ; % ceil(y) is the smallest integer larger than y
%x = floor(k * u); if outcomes are in {0,1,...,k-1}
```

Figure 6.6: The DF $F(x; 0.3, 0.7)$ of the de Moivre$(0.3, 0.7)$ RV and its inverse $F^{[-1]}(u; 0.3, 0.7)$.



Algorithm 5 Inversion Sampler for de Moivre$(1/k, 1/k, \ldots, 1/k)$ RV

1: *input:*

1. k in de Moivre$(1/k, 1/k, \ldots, 1/k)$ RV X
2. $u \sim$ Uniform$(0, 1)$

2: *output:* a sample from X

3: *return:* $x \leftarrow \lceil ku \rceil$

Let us use the function `SimdeMoivreEqui` *to draw five samples from a fair seven-faced cylindrical dice.*

```
>> k=7; % number of faces of the fair dice
>> n=5; % number of trials
>> rand('twister',78657); % initialise the fundamental sampler
>> u=rand(1,n); % draw n samples from Uniform(0,1)
>> % inverse transform samples from Uniform(0,1) to samples
>> % from de Moivre(1/7,1/7,1/7,1/7,1/7,1/7,1/7)
>> outcomes=SimdeMoivreEqui(u,k); % save the outcomes in an array
>> disp(outcomes);
     6     5     5     5     2
```

Now, let us consider the more general problem of implementing a sampler for an arbitrary but specified de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ RV. That is, the values of θ_i need not be equal to $1/k$.

Simulation 9 (de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$) *We can generate samples from a* de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ *RV* X *when* $(\theta_1, \theta_2, \ldots, \theta_k)$ *are specifiable as an input vector via the following algorithm.*

The M-file implementing Algorithm 6 is:

SimdeMoivreOnce.m

```
function  x = SimdeMoivreOnce(u,thetas)
% Returns a sample from the de Moivre(thetas=(theta_1,...,theta_k)) RV X
% Call Syntax:  x = SimdeMoivreOnce(u,thetas);
```

Algorithm 6 Inversion Sampler for de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ RV X

1: *input:*

   1. parameter vector $(\theta_1, \theta_2, \ldots, \theta_k)$ of de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ RV X.
   2. $u \sim \text{Uniform}(0, 1)$

2: *output:* a sample from X
3: *initialise:* $F \leftarrow \theta_1$, $i \leftarrow 1$
4: **while** $u > F$ **do**
5: $\quad i \leftarrow i + 1$
6: $\quad F \leftarrow F + \theta_i$
7: **end while**
8: *return:* $x \leftarrow i$

```
%               deMoivreEqui(u,thetas);
% Input      : u = a uniform random number eg. rand
%              thetas = an array of probabilities thetas=[theta_1 ... theta_k]
% Output     : x = sample from X
x=1; % initial index is 1
cum_theta=thetas(x);
while u > cum_theta;
    x=x+1;
    cum_theta = cum_theta + thetas(x);
end
```

Let us use the function `deMoivreEqui` *to draw five samples from a fair seven-faced dice.*

```
>> k=7; % number of faces of the fair dice
>> n=5; % number of trials
>> rand('twister',78657); % initialise the fundamental sampler
>> Us=rand(1,n); % draw n samples from Uniform(0,1)
>> disp(Us);
    0.8330    0.6819    0.6468    0.6674    0.2577
>> % inverse transform samples from Uniform(0,1) to samples
>> % from de Moivre(1/7,1/7,1/7,1/7,1/7,1/7,1/7)
>> f=[1/7 1/7 1/7 1/7 1/7 1/7 1/7];
>> disp(f);
    0.1429    0.1429    0.1429    0.1429    0.1429    0.1429    0.1429
>> % use funarray to apply function-handled SimdeMoivreOnce to
>> % each element of array Us and save it in array outcomes2
>> outcomes2=arrayfun(@(u)(SimdeMoivreOnce(u,f)),Us);
>> disp(outcomes2);
     6     5     5     5     2
>> disp(SimdeMoivreEqui(u,k)); % same result using the previous algorithm
     6     5     5     5     2
```

Clearly, Algorithm 6 may be used to sample from any de Moivre$(\theta_1, \ldots, \theta_k)$ *RV* X. *We demostrate this by producing five samples from a randomly generated PMF* `f2`.

```
>> rand('twister',1777); % initialise the fundamental sampler
>> f2=rand(1,10); % create an arbitrary array
>> f2=f2/sum(f2); % normalize to make a probability mass function
>> disp(f2); % display the weights of our 10-faced die
    0.0073    0.0188    0.1515    0.1311    0.1760    0.1121    ...
    0.1718    0.1213    0.0377    0.0723
```

```
>> disp(sum(f2)); % the weights sum to 1
    1.0000
>> disp(arrayfun(@(u)(SimdeMoivreOnce(u,f2)),rand(5,5))) % the samples from f2 are
     4     3     4     7     3
     6     7     4     5     3
     5     8     7    10     6
     2     3     5     7     7
     6     5     9     5     7
```

Note that the principal work here is the sequential search, in which the mean number of comparisons until success is:

$$1\theta_1 + 2\theta_2 + 3\theta_3 + \ldots + k\theta_k = \sum_{i=1}^{k} i\theta_i$$

For the de Moivre$(1/k, 1/k, \ldots, 1/k)$ RV, the right-hand side of the above expression is:

$$\sum_{i=1}^{k} i\frac{1}{k} = \frac{1}{k}\sum_{i=1}^{k} i = \frac{1}{k}\frac{k(k+1)}{2} = \frac{k+1}{2} \ ,$$

indicating that the average-case efficiency is linear in k. This linear dependence on k is denoted by $O(k)$. In other words, as the number of faces k increases, one has to work linearly harder to get samples from de Moivre$(1/k, 1/k, \ldots, 1/k)$ RV using Algorithm 6. Using the simpler Algorithm 5, which exploits the fact that all values of θ_i are equal, we generated samples in constant time, which is denoted by $O(1)$.

Let us consider a RV that arises from an IID stochastic process of Bernoulli(θ) RVs $\{X_i\}_{i\in\mathbb{N}}$, ie.

$$\{X_i\}_{i\in\mathbb{N}} := \{X_1, X_2, \ldots\} \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta) \ .$$

When we consider the number of IID Bernoulli(θ) trials before the first 'Head' occurs we get the following discrete RV.

Model 10 (Geometric(θ) **RV)** *Given a parameter $\theta \in (0,1)$, the PMF of the* Geometric(θ) *RV X is*

$$f(x;\theta) = \begin{cases} \theta(1-\theta)^x & \text{if } x \in \mathbb{Z}_+ := \{0,1,2,\ldots\} \\ 0 & \text{otherwise} \end{cases} \tag{6.14}$$

It is straightforward to verify that $f(x;\theta)$ is indeed a PDF :

$$\sum_{x=0}^{\infty} f(x;\theta) = \sum_{x=0}^{\infty} \theta(1-\theta)^x = \theta\left(\frac{1}{1-(1-\theta)}\right) = \theta\left(\frac{1}{\theta}\right) = 1$$

The above equality is a consequence of the geometric series identity (6.15) *with $a = \theta$ and $\vartheta := 1-\theta$:*

$$\sum_{x=0}^{\infty} a\vartheta^x = a\left(\frac{1}{1-\vartheta}\right), \text{ provided, } 0 < \vartheta < 1 \ . \tag{6.15}$$

Proof:

$$a + a\vartheta + a\vartheta^2 + \cdots + a\vartheta^n = \sum_{0\le x\le n} a\vartheta^x = a + \sum_{1\le x\le n} a\vartheta^x = a + \vartheta \sum_{1\le x\le n} a\vartheta^{x-1} = a + \vartheta \sum_{0\le x\le n-1} a\vartheta^x = a + \vartheta \sum_{0\le x\le n} a\vartheta^x - a\vartheta^{n+1}$$

Therefore,

$$\begin{aligned}
\sum_{0\le x\le n} a\vartheta^x &= a + \vartheta \sum_{0\le x\le n} a\vartheta^x - a\vartheta^{n+1} \\
\left(\sum_{0\le x\le n} a\vartheta^x\right) - \left(\vartheta \sum_{0\le x\le n} a\vartheta^x\right) &= a - a\vartheta^{n+1} \\
\left(\sum_{0\le x\le n} a\vartheta^x\right)(1-\vartheta) &= a(1-\vartheta^{n+1}) \\
\sum_{0\le x\le n} a\vartheta^x &= a\left(\frac{1-\vartheta^{n+1}}{1-\vartheta}\right) \\
\sum_{x=0}^{\infty} a\vartheta^x := \lim_{n\to\infty} \sum_{0\le x\le n} a\vartheta^x &= a\left(\frac{1}{1-\vartheta}\right), \text{ provided, } 0<\vartheta<1
\end{aligned}$$

The outcome of a Geometric(θ) *RV can be thought of as "the number of tosses needed before the appearance of the first 'Head' when tossing a coin with probability of 'Heads' equal to θ in a independent and identical manner."*

Mean and variance of Geometric(θ) **RV:** Let $X \sim$ Geometric(θ) RV. Then,

$$\mathbf{E}(X) = \sum_{x=0}^{\infty} x\theta(1-\theta)^x = \theta \sum_{x=0}^{\infty} x(1-\theta)^x$$

In order to simplify the RHS above, let us employ differentiation with respect to θ:

$$\frac{-1}{\theta^2} = \frac{d}{d\theta}\left(\frac{1}{\theta}\right) = \frac{d}{d\theta}\sum_{x=0}^{\infty}(1-\theta)^x = \sum_{x=0}^{\infty} -x(1-\theta)^{x-1}$$

Multiplying the LHS and RHS above by $-(1-\theta)$ and substituting in $\mathbf{E}(X) = \theta\sum_{x=0}^{\infty} x(1-\theta)^x$, we get a much simpler expression for $\mathbf{E}(X)$:

$$\frac{1-\theta}{\theta^2} = \sum_{x=0}^{\infty} x(1-\theta)^x \implies \mathbf{E}(X) = \theta\left(\frac{1-\theta}{\theta^2}\right) = \frac{1-\theta}{\theta}\ .$$

Similarly, it can be shown that

$$\mathbf{V}(X) = \frac{1-\theta}{\theta^2}\ .$$

Simulation 10 (Geometric(θ)**)** *We can simulate a sample x from a* Geometric(θ) *RV X using the following simple algorithm:*

$$x \leftarrow \lfloor \log(u)/\log(1-\theta) \rfloor, \qquad \textit{where, } u \sim \text{Uniform}(0,1)\ .$$

To verify that the above procedure is valid, note that:

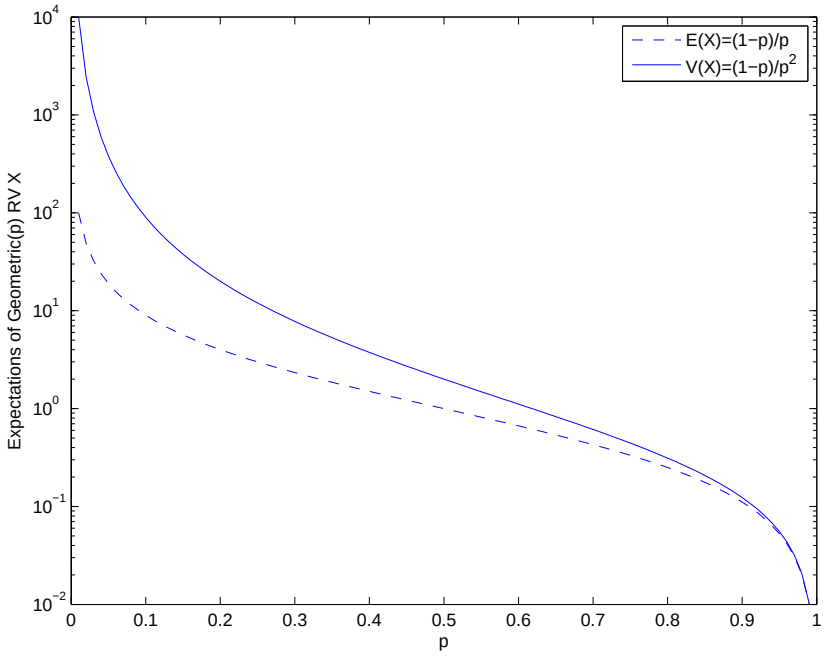
$$\begin{aligned}
\lfloor \log(U)/\log(1-\theta) \rfloor = x &\iff x \le \log(U)/\log(1-\theta) < x+1 \\
&\iff x \le \log_{1-\theta}(U) < x+1 \\
&\iff (1-\theta)^x \ge U > (1-\theta)^{x+1}
\end{aligned}$$

The inequalities are reversed since the base being exponentiated is $1-\theta \le 1$. The uniform event $(1-\theta)^x \ge U > (1-\theta)^{x+1}$ happens with the desired probability:

$$(1-\theta)^x - (1-\theta)^{x+1} = (1-\theta)^x(1-(1-\theta)) = \theta(1-\theta)^x =: f(x;\theta), \quad X \sim \text{Geometric}(\theta)\ .$$

We implement the sampler to generate samples from Geometric(θ) *RV with $\theta = 0.5$, for instance:*

Figure 6.7: Mean and variance of a Geometric(θ) RV X as a function of the parameter θ.




```
>> theta=0.5; u=rand(); % choose some theta and uniform(0,1) variate
>> % Simulate from a Geomertic(theta) RV
>> floor(log (u) / log (1 - theta))
ans =     0
>> floor(log ( rand(1,10) ) / log (1 - 0.5)) % theta=0.5, 10 samples
ans =     0     0     1     0     2     1     0     0     0     0
```

Labwork 23 *It is a good idea to make a relative frequency histogram of a simulation algorithm and compare that to the PDF of the discrete RV we are simulating from. We use the following script to create Figure 6.8:*

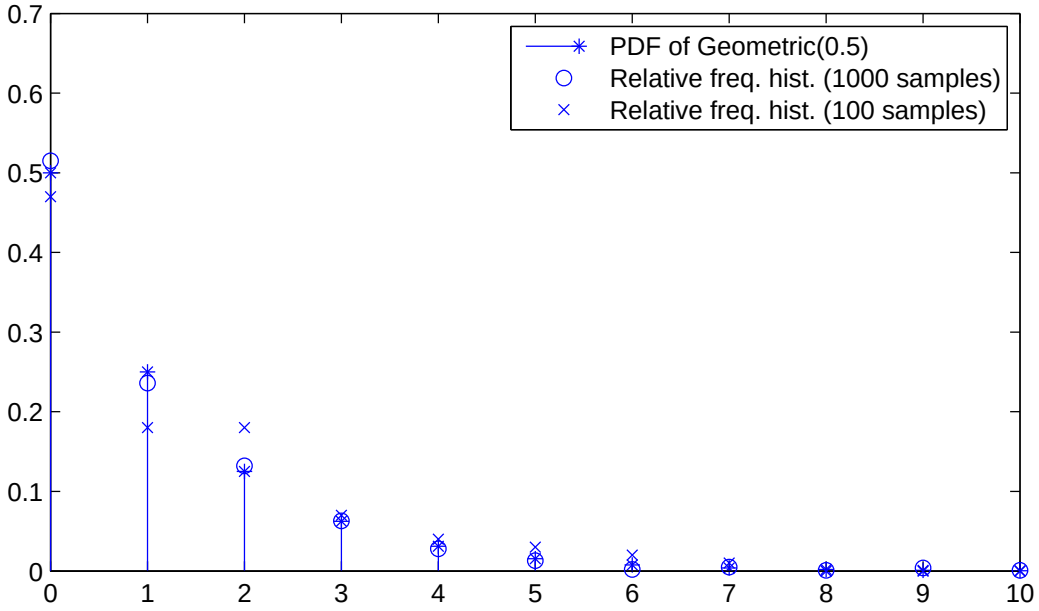
PlotPdfSimGeometric.m

```
theta=0.5;
SampleSize=1000;
% simulate 1000 samples from Geometric(theta) RV
Samples=floor(log(rand(1,SampleSize))/ log (1-theta));
Xs = 0:10; % get some values for x
RelFreqs=hist(Samples,Xs)/SampleSize; % relative frequencies of Samples
stem(Xs,theta*((1-theta) .^ Xs),'*')% PDF of Geometric(theta) over Xs
hold on;
plot(Xs,RelFreqs,'o')% relative frequence histogram
RelFreqs100=hist(Samples(1:100),Xs)/100; % Relative Frequencies of first 100 samples
plot(Xs,RelFreqs100,'x')
legend('PDF of Geometric(0.5)', 'Relative freq. hist. (1000 samples)', ...
    'Relative freq. hist. (100 samples)')
```

The RV Y in Table 3.1 may be generalized to an experiment $\mathcal{E}_\theta^n$ with n coin tosses. Let X_i be the Indicator function of the event 'Heads on the i-th toss' as before. Then Y defined by,

$$Y := \sum_{i=1}^{n} X_i := X_1 + X_2 + \cdots + X_n \ ,$$

Figure 6.8: PDF of $X \sim$ Geometric($\theta = 0.5$) and the relative frequency histogram based on 100 and 1000 samples from X.



is the number of 'Heads' in n tosses. Akin to the second row of Table 3.1, for the 'Toss n times' experiment $\mathcal{E}_\theta^n$ the RV Y as defined above will take values in $\{0, 1, 2, \ldots, n\}$ and is therefore a discrete RV. This is called the Binomial RV as defined next. But, first we remind ourselves of some elementary definitions involving arrangements of objects from a collection (recall Section 1.6).

Model 11 (Binomial(n, θ) **RV)** *Let the RV* $X = \sum_{i=1}^n X_i$ *be the sum of* n *independent and identically distributed* Bernoulli(θ) *RVs, i.e.:*

$$X = \sum_{i=1}^{n} X_i, \qquad X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta) \ .$$

Given two parameters n *and* θ*, the PMF of the* Binomial(n, θ) *RV* X *is:*

$$f(x; n, \theta) = \begin{cases} \binom{n}{x} \theta^x (1-\theta)^{n-x} & \text{if } x \in \{0, 1, 2, 3, \ldots, n\} \ , \\ 0 & \text{otherwise} \end{cases} \tag{6.16}$$

where, $\binom{n}{x}$ *is:*

$$\binom{n}{x} = \frac{n(n-1)(n-2)\ldots(n-x+1)}{x(x-1)(x-2)\cdots(2)(1)} = \frac{n!}{x!(n-x)!} \ .$$

$\binom{n}{x}$ *is read as "n choose x."*

Proof: Observe that for the Binomial(n, θ) RV X, $\mathbf{P}(X = x) = f(x; n, \theta)$ is the probability that x of the n Bernoulli(θ) trials result in an outcome of 1's. Next note that if all n X_i's are 0's, then $X = 0$, and if all n X_i's are 1's, then $X = n$. In general, if some of the n X_i's are 1's and the others are 0, then X can only take values in $\{0, 1, 2, \ldots, n\}$ and therefore $f(x; n, \theta) = 0$ if $x \notin \{0, 1, 2, \ldots, n\}$.

Now, let us compute $f(x;n,\theta)$ when $x \in \{0,1,2,\ldots,n\}$. Consider the set of indices $\{1,2,3,\ldots,n\}$ for the n IID Bernoulli(θ) RVs $\{X_1,X_2,\ldots,X_n\}$. Now choose x indices from $\{1,2,\ldots,n\}$ to mark those trials in a particular realization of $\{x_1,x_2,\ldots,x_n\}$ with the Bernoulli outcome of 1. The probability of each such event is $\theta^x(1-\theta)^{n-x}$ due to the IID assumption. For each realization $\{x_1,x_2,\ldots,x_n\} \in \{0,1\}^n := \{\text{all binary } (0-1) \text{ strings of length } n\}$, specified by a choice of x trial indices with Bernoulli outcome 1, the binomial RV $X = \sum_{i=1}^n X_i$ takes the value x. Since there are exactly $\binom{n}{x}$ many ways in which we can choose x trial indices (with outcome 1) from the set of n trial indices $\{1,2,\ldots,n\}$, we get the desired product for $f(x;n,\theta) = \binom{n}{x}\theta^x(1-\theta)^{n-x}$ when $x \in \{0,1,\ldots,n\}$.

Mean and variance of Binomial(n,θ) **RV:** Let $X \sim \text{Binomial}(n,\theta)$. Based on the definition of expectation:

$$\mathbf{E}(X) = \int x\, dF(x;n,\theta) = \sum_x x f(x;n,\theta) = \sum_{x=0}^{n} x \binom{n}{x} \theta^x (1-\theta)^{n-x} \ .$$

However, this is a nontrivial sum to evaluate. Instead, we may use (3.10) and (3.13) by noting that $X = \sum_{i=1}^n X_i$, where the $\{X_1,X_2,\ldots,X_n\} \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta)$, $\mathbf{E}(X_i) = \theta$ and $\mathbf{V}(X_i) = \theta(1-\theta)$:

$$\mathbf{E}(X) = \mathbf{E}(X_1 + X_2, \cdots, X_n) = \mathbf{E}\left(\sum_{i=1}^n X_i\right) = \sum_{i=1}^n \mathbf{E}(X_i) = n\theta \ ,$$

$$\mathbf{V}(X) = \mathbf{V}\left(\sum_{i=1}^n X_i\right) = \sum_{i=1}^n \mathbf{V}(X_i) = \sum_{i=1}^n \theta(1-\theta) = n\theta(1-\theta) \ .$$

Labwork 24 *We may implement the* MATLAB *function* `BinomialCoefficient` *to compute:*

$$\binom{n}{x} = \frac{n!}{x!(n-x)!} = \frac{n(n-1)(n-2)\ldots(n-x+1)}{x(x-1)(x-2)\cdots(2)(1)} = \frac{\prod_{i=(n-x+1)}^{n} i}{\prod_{i=2}^{x}} \ ,$$

with the following M-file:

BinomialCoefficient.m

```
function BC = BinomialCoefficient(n,x)
% returns the binomial coefficient of n choose x
% i.e. the combination of n objects taken x at a time
% x and n are scalar integers and 0 <= x <= n
NminusX = n-x;
NumeratorPostCancel = prod(n:-1:(max([NminusX,x])+1)) ;
DenominatorPostCancel = prod(2:min([NminusX, x]));
BC = NumeratorPostCancel/DenominatorPostCancel;
```

and call `BinomialCoefficient` *in the function* `BinomialPdf` *to compute the PDF* $f(x;n,\theta)$ *of the* Binomial(n,θ) *RV* X *as follows:*

BinomialPdf.m

```
function fx = BinomialPdf(x,n,theta)
%  Binomial probability mass function. Needs BinomialCoefficient(n,x)
%   f = BinomialPdf(x,n,theta)
%   f is the prob mass function for the Binomial(x;n,theta) RV
%   and x can be array of samples.
%   Values of x are integers in [0,n] and theta is a number in [0,1]
fx = zeros(size(x));
fx = arrayfun(@(xi)(BinomialCoefficient(n,xi)),x);
fx = fx .* (theta .^ x) .* (1-theta) .^ (n-x);
```

For example, we can compute the desired PDF for an array of samples `x` *from* Binomial$(8, 0.5)$ *RV* X*, as follows:*

```
>> x=0:1:8
x =     0     1     2     3     4     5     6     7     8
>> BinomialPdf(x,8,0.5)
ans =    0.0039    0.0312    0.1094    0.2188    0.2734    0.2188    0.1094    0.0312    0.0039
```

Simulation 11 (Binomial(n, θ) **as** $\sum_{i=1}^{n}$ Bernoulli(θ)**)** *Since the* Binomial(n, θ) *RV* X *is the sum of* n *IID* Bernoulli(θ) *RVs we can also simulate from* X *by first simulating* n *IID* Bernoulli(θ) *RVs and then adding them up as follows:*

```
>> rand('twister',17678);
>> theta=0.5; % give some desired theta value, say 0.5
>> n=5; % give the parameter n for Binomial(n,theta) RV X, say n=5
>> xis=floor(rand(1,n)+theta) % produce n IID samples from Bernoulli(theta=0.5) RVs X1,X2,...Xn
xis =     1     1     0     0     0
>> x=sum(xis) % sum up the xis to get a sample from Binomial(n=5,theta=0.5) RV X
x =     2
```

It is straightforward to produce more than one sample from X *by exploiting the column-wise summing property of* MATLAB*'s* `sum` *function when applied to a two-dimensional array:*

```
>> rand('twister',17);
>> theta=0.25; % give some desired theta value, say 0.25 this time
>> n=3; % give the parameter n for Binomial(n,theta) RV X, say n=3 this time
>> xis10 = floor(rand(n,10)+theta) % produce an n by 10 array of IID samples from Bernoulli(theta=0.25) RVs
xis10 =
     0     0     0     0     1     0     0     0     0     0
     0     1     0     1     1     0     0     0     0     0
     0     0     0     0     0     0     0     1     0     0
>> x=sum(xis10) % sum up the array column-wise to get 10 samples from Binomial(n=3,theta=0.25) RV X
x =     0     1     0     1     2     0     0     1     0     0
```

In Simulation 11, the number of IID Bernoulli(θ) RVs needed to simulate one sample from the Binomial(n, θ) RV is exactly n. Thus, as n increases, the amount of time needed to simulate from Binomial(n, θ) is $O(n)$, i.e. linear in n. We can simulate more efficiently by exploiting a simple relationship between the Geometric(θ) RV and the Binomial(n, θ) RV.

The Binomial(n, θ) RV X is related to the IID Geometric(θ) RV $Y_1, Y_2, \ldots$: X is the number of successful Bernoulli(θ) outcomes (outcome is 1) that occur in a total of n Bernoulli(θ) trials, with the number of trials between consecutive successes distributed according to IID Geometric(θ) RV.

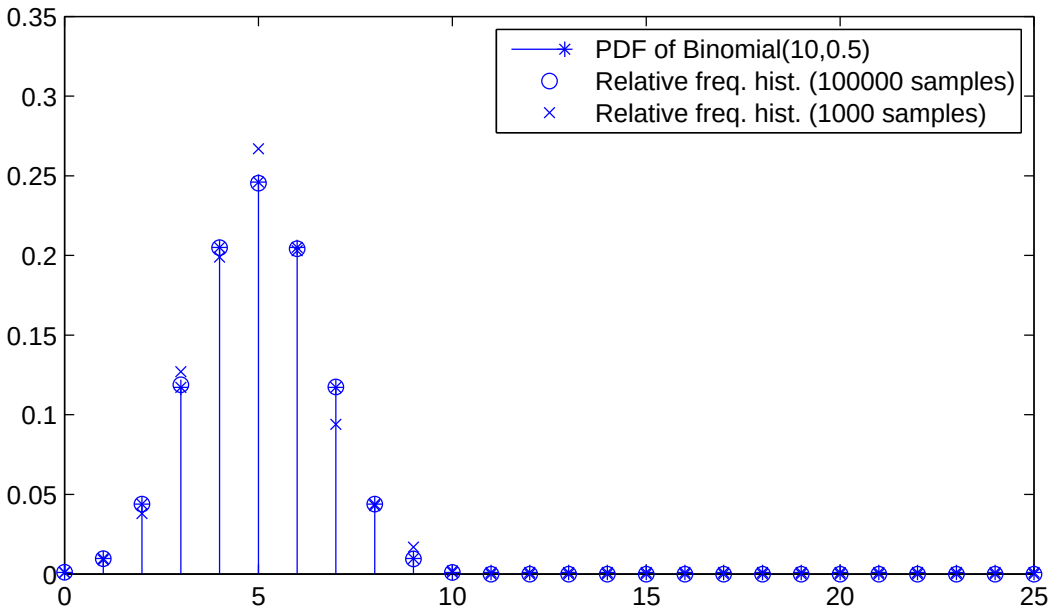
Simulation 12 (Binomial(θ) **from IID** Geometric(θ) **RVs)** *By this principle, we can simulate from the* Binomial(θ) X *by* Step 1*: generating IID* Geometric(θ) *RVs* $Y_1, Y_2, \ldots$*,* Step 2*: stopping as soon as* $\sum_{i=1}^{k}(Y_i + 1) > n$ *and* Step 3*: setting* $x \leftarrow k - 1$*.*

We implement the above algorithm via the following M-file:

Sim1BinomByGeoms.m

```
function x = Sim1BinomByGeoms(n,theta)
% Simulate one sample from Binomial(n,theta) via Geometric(theta) RVs
YSum=0; k=0; % initialise
while (YSum <= n),
    TrialsToSuccess=floor(log(rand)/log (1-theta)) + 1; % sample from Geometric(theta)+1
    YSum = YSum + TrialsToSuccess; % total number of trials
    k=k+1; % number of Bernoulli successes
end
x=k-1; % return x
```

Figure 6.9: PDF of $X \sim \text{Binomial}(n = 10, \theta = 0.5)$ and the relative frequency histogram based on $100,000$ samples from X.



Here is a call to simulate 12 *samples from* Binomial$(n = 10, \theta = 0.5)$ *RV:*

```
>> theta=0.5; % declare theta
>> n=10; % say n=10
>> SampleSize=12;% say you want to simulate 12 samples
>> rand('twister',10001) % seed the fundamental sampler
>> Samples=arrayfun(@(T)Sim1BinomByGeoms(n,T),theta*ones(1,SampleSize))
Samples =     7     5     8     8     4     1     4     8     2     4     6     5
```

Figure 6.9 depicts a comparison of the PDF of Binomial$(n = 10, \theta = 0.5)$ *RV and a relative frequency histogram based on* $100,000$ *simulations from it.*

In several situations it becomes cumbersome to model the events using the Binomial(n, θ) RV, especially when when the parameter $\theta \propto 1/n$ and the events become rare. However, for some real parameter $\lambda > 0$, the Binomial$(n, \lambda/n)$ RV with probability of the number of successes in n trials, with per-trial success probability λ/n, approaches the Poisson distribution with expectation λ, as n approaches ∞ (actually, it converges in distribution as defined later). The Poisson(λ) RV is much simpler to work with than the combinatorially laden Binomial$(n, \theta = \lambda/n)$ RV. We sketch the details of this next.

Let $X \sim \text{Binomial}(n, \theta = \lambda/n)$, then for any $x \in \{0, 1, 2, 3, \ldots, n\}$,

$$\begin{aligned}
\mathbf{P}(X = x) &= \binom{n}{x} \left(\frac{\lambda}{n}\right)^x \left(1 - \frac{\lambda}{n}\right)^{n-x} \\
&= \frac{n(n-1)(n-2)\cdots(n-x+1)}{x(x-1)(x-2)\cdots(2)(1)} \left(\frac{\lambda^x}{n^x}\right) \left(1 - \frac{\lambda}{n}\right)^n \left(1 - \frac{\lambda}{n}\right)^{-x} \\
&= \overbrace{\left(\frac{n}{n}\right)\left(\frac{n-1}{n}\right)\left(\frac{n-2}{n}\right)\cdots\left(\frac{n-x+1}{n}\right)}\overbrace{\left(\frac{\lambda^x}{x!}\right)} \underbrace{\left(1 - \frac{\lambda}{n}\right)^n}\underbrace{\left(1 - \frac{\lambda}{n}\right)^{-x}}
\end{aligned} \tag{6.17}$$

As $n \to \infty$, the expression below the first overbrace $\to 1$, while that below the second overbrace, being independent of n remains the same. By the elementary examples of limits 6 and 7, as $n \to \infty$, the expression over the first underbrace approaches $e^{-\lambda}$ while that over the second underbrace approaches 1. Finally, we get the desired limit:

$$\lim_{n\to\infty} \mathbf{P}(X = x) = \frac{e^{-\lambda}\lambda^x}{x!} \ .$$

Model 12 (Poisson(λ) **RV)** *Given a real parameter $\lambda > 0$, the discrete RV X is said to be* Poisson(λ) *distributed if X has PDF:*

$$f(x;\lambda) = \begin{cases} \frac{e^{-\lambda}\lambda^x}{x!} & \text{if } x \in \mathbb{Z}_+ := \{0, 1, 2, \ldots\} \ , \\ 0 & \text{otherwise} \ . \end{cases} \tag{6.18}$$

Note that the PDF integrates to 1*:*

$$\sum_{x=0}^{\infty} f(x;\lambda) = \sum_{x=0}^{\infty} \frac{e^{-\lambda}\lambda^x}{x!} = e^{-\lambda} \sum_{x=0}^{\infty} \frac{\lambda^x}{x!} = e^{-\lambda}e^{\lambda} = 1 \ ,$$

where we exploit the Taylor series of e^{λ} to obtain the second-last equality above.

Mean and variance of Poisson(λ) **RV:** Let $X \sim$ Poisson(λ). Then:

$$\mathbf{E}(X) = \sum_{x=0}^{\infty} x f(x;\lambda) = \sum_{x=0}^{\infty} x \frac{e^{-\lambda}\lambda^x}{x!} = e^{-\lambda} \sum_{x=0}^{\infty} x \frac{\lambda^x}{x!} = e^{-\lambda} \sum_{x-1=0}^{\infty} \frac{\lambda\lambda^{x-1}}{(x-1)!} = e^{-\lambda}\lambda e^{\lambda} = \lambda \ .$$

Similarly, it can be shown that

$$\mathbf{V}(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2 = \lambda + \lambda^2 - \lambda^2 = \lambda \ .$$

Note that Poisson(λ) distribution is one whose mean and variance are the same, namely λ.

The Poisson(λ) RV X is also related to the IID Exponential(λ) RV $Y_1, Y_2, \ldots$: X is the number of occurrences, per unit time, of an instantaneous event whose inter-occurrence time is the IID Exponential(λ) RV. For example, the number of buses arriving at our bus-stop in the next minute, with exponentially distributed inter-arrival times, has a Poisson distribution.

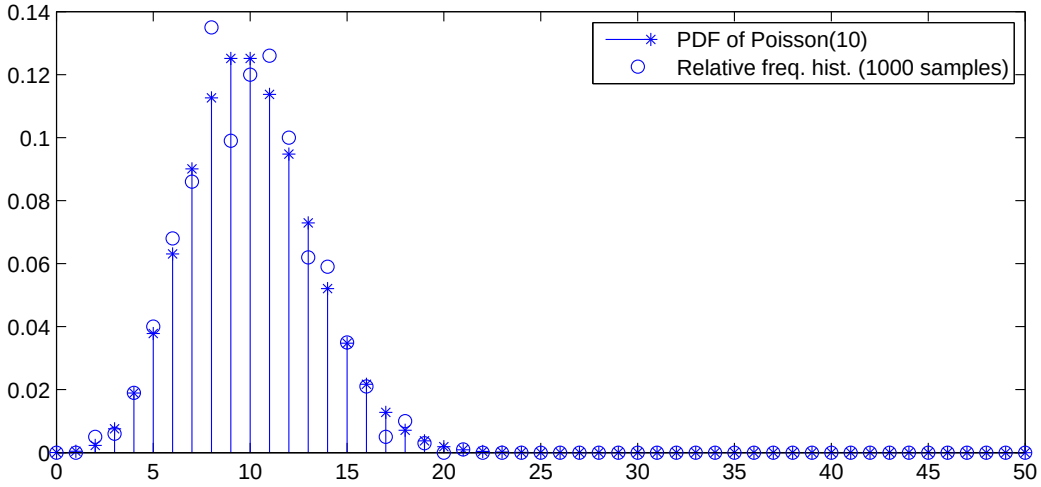
Simulation 13 (Poisson(λ) **from IID** Exponential(λ) **RVs)** *By this principle, we can simulate from the* Poisson(λ) *X by* Step 1*: generating IID* Exponential(λ) *RVs $Y_1, Y_2, \ldots,$* Step 2*: stopping as soon as $\sum_{i=1}^{k} Y_i \geq 1$ and* Step 3*: setting $x \leftarrow k - 1$.*

We implement the above algorithm via the following M-file:

Sim1Poisson.m

```
function x = Sim1Poisson(lambda)
% Simulate one sample from Poisson(lambda) via Exponentials
YSum=0; k=0; % initialise
while (YSum < 1),
    YSum = YSum + -(1/lambda) * log(rand);
    k=k+1;
end
x=k-1; % return x
```

Figure 6.10: PDF of $X \sim \text{Poisson}(\lambda = 10)$ and the relative frequency histogram based on 1000 samples from X.



Here is a call to simulate 10 samples from $\text{Poisson}(\lambda = 10.0)$ *and* $\text{Poisson}(\lambda = 0.1)$ *RVs:*

```
>> arrayfun(@(lambda)Sim1Poisson(lambda),10.0*ones(1,10)) % lambda=10.0
ans =    14     7    10    13    11     3     6     5     8     5
>> arrayfun(@(lambda)Sim1Poisson(lambda),0.1*ones(1,10)) % lambda=0.1
ans =     2     0     0     0     0     0     0     0     0     0
```

Figure 6.10 depicts a comparison of the PDF of $\text{Poisson}(\lambda = 10)$ *RV and a relative frequency histogram based on 1000 simulations from it.*

Simulating from a $\text{Poisson}(\lambda)$ RV is also a special case of simulating from the following more general RV.

Model 13 ($GD(\theta_0, \theta_1, \ldots)$) *We say X is a* General Discrete$(\theta_0, \theta_1, \ldots)$ *or* GD$(\theta_0, \theta_1, \ldots)$ *RV over the countable discrete state space $\mathbb{Z}_+ := \{0, 1, 2, \ldots\}$ with parameters $(\theta_0, \theta_1, \ldots)$ if the PMF of X is defined as follows:*

$$f(X = x; \theta_0, \theta_1, \ldots) = \begin{cases} 0, & if \quad x \notin \{0, 1, 2, \ldots\} \\ \theta_0, & if \quad x = 0 \\ \theta_1, & if \quad x = 1 \\ \vdots \end{cases}$$

Algorithm 7 allows us to simulate from any member of the class of non-negative discrete RVs as specified by the probabilities $(\theta_0, \theta_1, \ldots)$. When an RV X takes values in another countable set $\mathbb{X} \neq \mathbb{Z}_+$, then we can still use the above algorithm provided we have a one-to-one and onto mapping D from $\mathbb{Z}_+$ to $\mathbb{X}$ that allows us to think of $\{0, 1, 2, \ldots\}$ as indices of an array D.

Simulation 14 (Binomial(n, θ)) *To simulate from a* Binomial(n, θ) *RV X, we can use Algorithm 7 with:*

$$\theta_0 = (1 - \theta)^n, \qquad C(x + 1) = \frac{\theta(n - x)}{(1 - \theta)(x + 1)}, \qquad \textit{Mean Efficiency: } O(1 + n\theta) \ .$$

Algorithm 7 Inversion Sampler for $GD(\theta_0, \theta_1, \ldots)$ RV X

1: *input:*

1. θ_0 and $\{C(i) = \theta_i/\theta_{i-1}\}$ for any $i \in \{1, 2, 3, \ldots\}$.
2. $u \sim \text{Uniform}(0, 1)$

2: *output:* a sample from X
3: *initialise:* $p \leftarrow \theta_0$, $q \leftarrow \theta_0$, $i \leftarrow 0$
4: **while** $u > q$ **do**
5: $\quad i \leftarrow i + 1$, $p \leftarrow p\, C(i)$, $q \leftarrow q + p$
6: **end while**
7: *return:* $x = i$

Similarly, with the appropriate θ_0 and $C(x + 1)$, we can also simulate from the Geometric(θ) and Poisson(λ) RVs.

Labwork 25 *1. Implement Algorithm 7 via a function named* `MyGenDiscInvSampler` *in* `MATLAB`. *Hand in the* `M-file` *named* `MyGenDiscInvSampler.m` *giving detailed comments explaining your understanding of each step of the code. [Hint: $C(i)$ should be implemented as a function (use function handles via* `@`*) that can be passed as a parameter to the function* `MyGenDiscInvSampler`*].*

2. Show that your code works for drawing samples from a Binomial(n, p) *RV by doing the following:*

(a) Seed the fundamental sampler by your Student ID (if your ID is `11424620` *then type* `rand('twister', 11424620);`*)*

(b) Draw 100 samples from the Binomial$(n = 20, p = 0.5)$ *RV and report the results in an 2×2 table with column headings* `x` *and No. of observations. [Hint: the inputs θ_0 and $C(i)$ for the* Binomial(n, p) *RV is given above].*

3. Show that your code works for drawing samples from a Geometric(p) *RV by doing the following:*

(a) Seed the fundamental sampler by your Student ID.

(b) Set the variable `Mytheta=rand`*.*

(c) Draw 100 samples from the Geometric(`Mytheta`) *RV and report the sample mean. [Note: the inputs θ_0 and $C(i)$ for the* Geometric(θ) *RV should be derived and the workings shown].*

To make concrete sense of the Binomial(n, θ) and other more sophisticated concepts in the sequel, let us take a historical detour into some origins of statistical thinking in 19th century England.

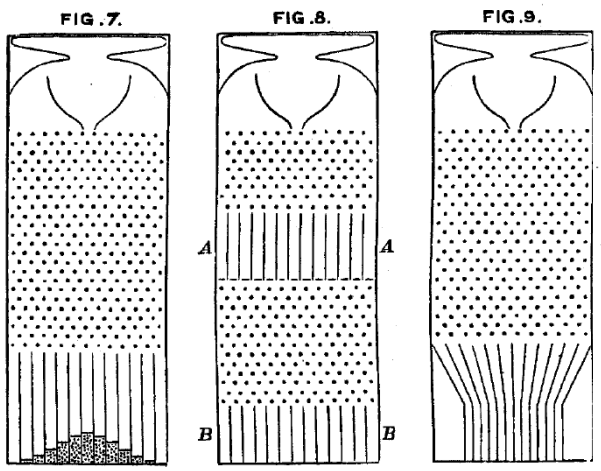
6.5 Sir Francis Galton's Quincunx

This section is introduced to provide some forms for a kinesthetic (hands-on) and visual understanding of some elementary statistical distributions and laws. The following words are from Sir Francis

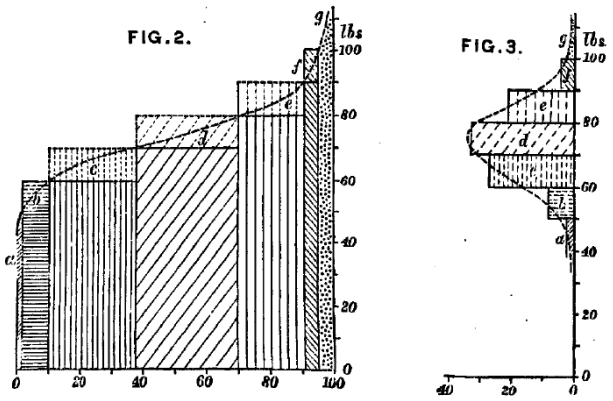
Galton, F.R.S., *Natural Inheritance*, pp. 62-65, Macmillan, 1889. In here you will already find the kernels behind the construction of Binomial(θ) RV as sum of IID Bernoulli(θ) RVs, Weak Law of Large Numbers, Central Limit Theorem, and more. We will mathematically present these concepts in the sequel as a way of giving precise meanings to Galton's observations with his Quincunx. "The Charms of Statistics.–*It is difficult to understand why statisticians commonly limit their inquiries to Averages, and do not revel in more comprehensive views. Their souls seem as dull to the charm of variety as that of the native of one of our flat English counties, whose retrospect of Switzerland was that, it its mountains could be thrown into its lakes, two nuances would be got rid of at once. An Average is but a solitary fact, whereas if a single other fact be added to it, an entire Normal Scheme, which nearly corresponds to the observed one, starts potentially into existence.*

Some people hate the very name of statistics, but I find them full of beauty and interest. Whenever they are not brutalised, but delicately handled by the higher methods, and are warily interpreted, their power of dealing with complicated phenomenon is extraordinary. They are the only tools by which an opening can be cut through the formidable thicket of difficulties that bars the path of those who pursue the Science of man.

Figure 6.11: Figures from Sir Francis Galton, F.R.S., *Natural Inheritance*, , Macmillan, 1889.



(a) FIG. 7, FIG. 8, and FIG. 9 (p. 63)



(b) FIG. 2 and FIG. 3 (p. 38)

Mechanical Illustration of the Cause of the Curve of Frequency.–*The Curve of Frequency, and that of Distribution, are convertible : therefore if the genesis of either of them can be made clear, that of the other also becomes intelligible. I shall now illustrate the origin of the Curve of Frequency, by means of an apparatus shown in Fig. 7, that mimics in a very pretty way the conditions on which Deviation depends. It is a frame glazed in front, leaving a depth of about a quarter of an inch behind the glass. Strips are placed in the upper part to act as a funnel. Below the outlet of the funnel stand a succession of rows of pins stuck squarely into the backboard, and below these again are a series of vertical compartments. A charge of small shot is inclosed. When the frame is held topsy-turvy, all the shot runs to the upper end; then, when it is turned back into its working position, the desired action commences. Lateral strips, shown in the diagram, have the effect of directing all the shot that had collected at the upper end of the frame to run into the wide moutn of the funnel. The shot passes through the funnel and issuing from its narrow end, scampers deviously down through the pins in a curious and interesting way; each of them darting a step to the right or left, as the case may be, every time it strikes a pin. The pins are disposed in a quincunx fashion, so that every descending shot strikes against a pin in each successive row. The cascade issuing from the funnel broadens as it descends, and, at length, every shot finds itself caught in a compartment immediately*

after freeing itself from the last row of pins. The outline of the columns of shot that accumulate in the successive compartments approximates to the Curve of Frequency (Fig. 3, p. 38), and is closely of the same shape however often the experiment is repeated. The outline of the columns would become more nearly identical with the Normal Curve of Frequency, if the rows of pins were much more numerous, the shot smaller, and the compartments narrower; also if a larger quantity of shot was used.

The principle on which the action of the apparatus depends is, that a number of small and independent accidents befall each shot in its career. In rare cases, a long run of luck continues to favour the course of a particular shot towards either outside place, but in the large majority of instances the number of accidents that cause Deviation to the right, balance in a greater or less degree those that cause Deviation to the left. Therefore most of the shot finds its way into the compartments that are situated near to a perpendicular line drawn from the outlet of the funnel, and the Frequency with which shots stray to different distances to the right or left of that line diminishes in a much faster ratio than those distances increase. This illustrates and explains the reason why mediocrity is so common."

6.6 Random Vectors

Let us try to relate some discrete probability models to the Quincunx. First, we need to introduce simple random vectors ($\mathrm{R}\vec{\mathrm{V}}$), i.e. ordered pairs, ordered triples, or more generally ordered m-tuples of random variables $(X_1, X_2, \ldots, X_m)$. We focus on elementary definitions needed to define bivariate $\mathrm{R}\vec{\mathrm{V}}$ obtained from a pair of RVs. Here is a simple example of a discrete bivariate $\mathrm{R}\vec{\mathrm{V}}$ that illustrates the notions of joint and marginal probabilities.

Example 16 *Let X_1 and X_2 be a pair of IID* Bernoulli(1/2) *RVs each taking values in the set $\{0, 1\}$ with the following joint probabilities:*

	$X_2 = 0$	$X_2 = 1$	
$X_1 = 0$	1/4	1/4	1/2
$X_1 = 1$	1/4	1/4	1/2
	1/2	1/2	1

From the above Table we can read for instance that the joint probability $\mathbf{P}((X_1, X_2) = (0, 0)) = 1/4$ and that the marginal probability $\mathbf{P}(X_1 = 0) = 1/2$.

Definition 32 (Joint PDF, PMF, CDF) *A function $f(x_1, x_2)$ is called a* **joint PDF (or PMF)** *for the ordered pair of random variables (X_1, X_2) if:*

1. *$f(x_1, x_2) \geq 0$ for all $(x_1, x_2) \in \mathbb{R}^2$*
2. $$1 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dF(x_1, x_2) = \begin{cases} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x_1, x_2)dx_1dx_2 & \text{if } (X_1, X_2) \text{ are continuous} \\ \sum_{x_1}\sum_{x_2} f(x_1, x_2) & \text{if } (X_1, X_2) \text{ are discrete} \end{cases}$$
3. *for any event $A \subset \mathbb{R}^2$,*
$$\mathbf{P}(A) = \int\int_A dF(x_1, x_2) = \begin{cases} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathbb{1}_A((x_2, x_2))f(x_1, x_2)dx_1dx_2 & \text{if } (X_1, X_2) \text{ are continuous} \\ \sum_{x_1}\sum_{x_2} \mathbb{1}_A((x_2, x_2))f(x_1, x_2) & \text{if } (X_1, X_2) \text{ are discrete} \end{cases}$$

The **joint CDF or joint DF** *for discrete or continuous $R\vec{V}$ (X_1, X_2) is:*

$$F(x_1, x_2) := \mathbf{P}(X_1 \leq x_1, X_2 \leq x_2) \ .$$

Definition 33 (Marginal PDF or PMF) *If the $R\vec{V}$ (X_1, X_2) has $f(x_1, x_2)$ as its joint density, i.e. joint PDF or joint PMF, then the* **marginal PDF or PMF** *of X_1 is defined by:*

$$f(x_1) = \mathbf{P}(X_1 = x_1) = \begin{cases} \int_{-\infty}^{\infty} f(x_1, x_2) dx_2 & \text{if } (X_1, X_2) \text{ are continuous} \\ \sum_{x_2} f(x_1, x_2) & \text{if } (X_1, X_2) \text{ are discrete} \end{cases}$$

and the **marginal PDF or PMF** *of X_2 is defined by:*

$$f(x_2) = \mathbf{P}(X_2 = x_2) = \begin{cases} \int_{-\infty}^{\infty} f(x_1, x_2) dx_1 & \text{if } (X_1, X_2) \text{ are continuous} \\ \sum_{x_1} f(x_1, x_2) & \text{if } (X_1, X_2) \text{ are discrete} \end{cases}$$

Example 17 (Bivariate Uniform) *Let (X_1, X_2) be uniformly distributed on the square $[0,1]^2 := [0,1] \times [0,1]$. Then,*

$$f(x_1, x_2) = \mathbb{1}_{[0,1]^2}(x_1, x_2) \ .$$

Let the rectangular event $A = \{X_1 < 1/3, Y < 1/2\} \subset [0,1]^2$. By integrating the joint PDF over A, which amounts here to finding the area of A, we compute $\mathbf{P}(A) = (1/3)(1/2) = 1/6$. Note that the marginal PDF of X_1 or X_2 is the PDF of the Uniform$(0,1)$ *RV.*

Definition 34 (Conditional PDF or PMF) *Let (X_1, X_2) be a discrete bivariate $R\vec{V}$. The conditional PMF of $X_1|X_2 = x_2$, where $f(X_2 = x_2) := \mathbf{P}(X_2 = x_2) > 0$ is:*

$$f(x_1|x_2) := \mathbf{P}(X_1 = x_1|X_2 = x_2) = \frac{\mathbf{P}(X_1 = x_1, X_2 = x_2)}{\mathbf{P}(X_2 = x_2)} = \frac{f(x_1, x_2)}{f(x_2)} \ .$$

Similarly, if $f(X_1 = x_1) > 0$, then the conditional PMF of $X_2|X_1 = x_1$ is:

$$f(x_2|x_1) := \mathbf{P}(X_2 = x_2|X_1 = x_1) = \frac{\mathbf{P}(X_1 = x_1, X_2 = x_2)}{\mathbf{P}(X_1 = x_1)} = \frac{f(x_1, x_2)}{f(x_1)} \ .$$

If (X_1, X_2) are continuous RVs such that $f(x_2) > 0$, then the conditional PDF of $X_1|X_2 = x_2$ is:

$$f(x_1|x_2) = \frac{f(x_1, x_2)}{f(x_2)}, \qquad \mathbf{P}(X_1 \in A|X_2 = x_2) = \int_A f(x_1|x_2) dx_1 \ .$$

Similarly, if $f(x_1) > 0$, then the conditional PDF of $X_2|X_1 = x_1$ is:

$$f(x_2|x_1) = \frac{f(x_1, x_2)}{f(x_1)}, \qquad \mathbf{P}(X_2 \in A|X_1 = x_1) = \int_A f(x_2|x_1) dx_2 \ .$$

We need a new notion for the variance of two RVs.

Definition 35 (Covariance) *Suppose X_1 and X_2 are random variables, such that $\mathbf{E}(X_1^2) < \infty$ and $\mathbf{E}(X_2)^2 < \infty$. Then, $\mathbf{E}(|X_1 X_2|) < \infty$ and $\mathbf{E}(|(X_1 - \mathbf{E}(X_1))(X_2 - \mathbf{E}(X_2))|) < \infty$. We therefore define the covariance $\mathbf{Cov}(X_1, X_2)$ of X_1 and X_2 as:*

$$\mathbf{Cov}(X_1, X_2) := \mathbf{E}\left((X_1 - \mathbf{E}(X_1))(X_2 - \mathbf{E}(X_2))\right) = \mathbf{E}(X_1 X_2) - \mathbf{E}(X_1)\mathbf{E}(X_2)$$

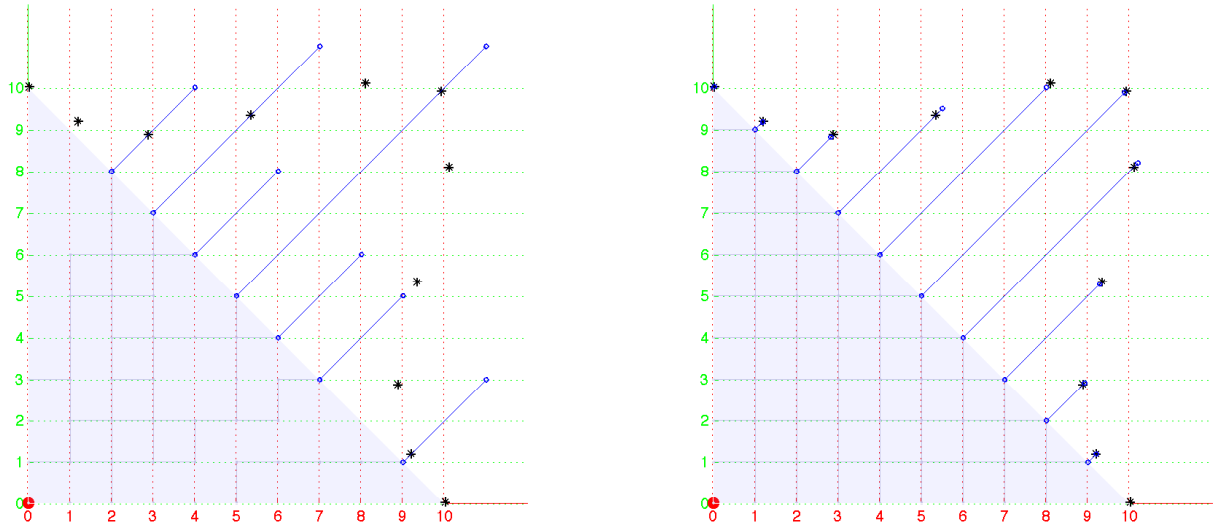
Let us consider the natural two-dimensional analogue of the Bernoulli(θ) RV in the real plane $\mathbb{R}^2 := (-\infty, \infty)^2 := (-\infty, \infty) \times (-\infty, \infty)$. A natural possibility is to use the **ortho-normal basis vectors** in $\mathbb{R}^2$:

$$\boxed{e_1 := (1, 0), \qquad e_2 := (0, 1)} \, .$$

Recall that vector addition and subtraction are done component-wise, i.e. $(x_1, x_2) \pm (y_1, y_2) = (x_1 \pm y_1, x_2 \pm y_2)$.

Classwork 13 (Geometry of Vector Addition) *Recall elementary vector addition in the plane. What is* $(1, 0) + (1, 0)$, $(1, 0) + (0, 1)$, $(0, 1) + (0, 1)$ *? What is the relationship between* $(1, 0)$, $(0, 1)$ *and* $(1, 1)$ *geometrically ? How does the diagonal of the parallelogram relate the its two sides in the geometry of addition in the plane? What is* $(1, 0) + (0, 1) + (1, 0)$ *?*

Figure 6.12: Quincunx on the Cartesian plane. Simulations of Binomial($n = 10, \theta = 0.5$) RV as the x-coordinate of the ordered pair resulting from the culmination of sample trajectories formed by the accumulating sum of $n = 10$ IID Bernoulli($\theta = 0.5$) random vectors over $\{(1, 0), (0, 1)\}$ with probabilities $\{\theta, 1 - \theta\}$, respectively. The blue lines and black asterisks perpendicular to and above the diagonal line, i.e. the line connecting $(0, 10)$ and $(10, 0)$, are the density histogram of the samples and the PDF of our Binomial($n = 10, \theta = 0.5$) RV, respectively.



(a) Ten samples

(b) Thousand samples

Model 14 (Bernoulli(θ) $\mathbf{R\vec{V}}$**)** *Given a parameter* $\theta \in [0, 1]$*, we say that* $X := (X_1, X_2)$ *is a* Bernoulli(θ) *random vector (*$R\vec{V}$*) if it has only two possible outcomes in the set* $\{e_1, e_2\} \subset \mathbb{R}^2$*, i.e.* $x := (x_1, x_2) \in \{(1, 0), (0, 1)\}$*. The PMF of the* $R\vec{V}$ $X := (X_1, X_2)$ *with realisation* $x := (x_1, x_2)$ *is:*

$$f(x; \theta) := \mathbf{P}(X = x) = \theta \mathbb{1}_{\{e_1\}}(x) + (1 - \theta)\mathbb{1}_{\{e_2\}}(x) = \begin{cases} \theta & \text{if} \quad x = e_1 := (1, 0) \\ 1 - \theta & \text{if} \quad x = e_2 := (0, 1) \\ 0 & \text{otherwise} \end{cases}$$

Classwork 14 *What is the Expectation of* Bernoulli(θ) $R\vec{V}$*?*

$$\mathbf{E}_\theta(X) = \mathbf{E}_\theta((X_1, X_2)) = \sum_{(x_1,x_2)\in\{e_1,e_2\}} (x_1, x_2) f((x_1, x_2); \theta) = (1, 0)\theta + (0, 1)(1 - \theta) = (\theta, 1 - \theta) \ .$$

How about the variance ? [Hint: Use the definitions of $\mathbf{E}(X)$ *and* $\mathbf{V}(X)$ *for the* $R\vec{V}$ X*.* $\mathbf{E}(X^2)$ *is not a single number and you may need new words such as covariance to deal with terms like* $\mathbf{E}(X_1X_2)$*.]*

We can write the Binomial(n, θ) RV Y as a Binomial(n, θ) R$\vec{\text{V}}$ $X := (Y, n - Y)$. In fact, this is the underlying model and the **bi** in the Binomial(n, θ) does refer to two in Latin. In the coin-tossing context this can be thought of keeping track of the number of Heads and Tails out of an IID sequence of n tosses of a coin with probability θ of observing Heads. In the Quincunx context, this amounts to keeping track of the number of right and left turns made by the ball as it drops through n levels of pegs where the probability of a right turn at each peg is independently and identically θ. In other words, the Binomial(n, θ) R$\vec{\text{V}}$ $(Y, n - Y)$ is the sum of n IID Bernoulli(θ) R$\vec{\text{V}}$s $X_1 := (X_{1,1}, X_{1,2}), X_2 := (X_{2,1}, X_{2,2}), \ldots, X_n := (X_{n,1}, X_{n,2})$:

$$(Y, n - Y) = X_1 + X_2 + \cdots + X_n = (X_{1,1}, X_{1,2}) + (X_{2,1}, X_{2,2}) + \cdots + (X_{n,1}, X_{n,2})$$

Go the Biomathematics Research Centre on the 6th floor of Erskine to play with the Quincunx built by Ryan Lawrence in 2007 (See the project by Ashman and Lawrence at `http://www.math.canterbury.ac.nz/~r.sainudiin/courses/STAT218/projects/Stat218StudentProjects2007.pdf` for details). It is important to gain a physical intimacy with the Quincunx to appreciate the following model of it. We can make a statistical model of Galton's observations earlier regarding the dynamics of lead shots through the Quincunx as the sum of n IID Bernoulli(0.5) R$\vec{\text{V}}$s, where n is number of pegs that each ball bounces on before making a left or right turn with equal probability. See animations of the balls dropping through the Quincunx with $n = 10$ for 10 and 100 samples or balls at `http://www.math.canterbury.ac.nz/~r.sainudiin/courses/Demos/Quincunx` done by Bry Ashman in 2007/8. When we drop 1000 balls into our simulated Quincunx the density histogram is much closer to the PDF of Binomial($n = 10, \theta = 0.5$) RV than when we only drop 10 balls. See Figure 6.12 for a description of the simulations.

Classwork 15 *How does the number of paths that lead to a bucket* (x_1, x_2) *with* $x_1 + x_2 = n$ *relate to the binomial coefficient* $\binom{n}{x_1}$*?*

We are now ready to extend the Binomial(n, θ) RV or R$\vec{\text{V}}$ to its multivariate version called the Multinomial($n, \theta_1, \theta_2, \ldots, \theta_k$) R$\vec{\text{V}}$. We develop this R$\vec{\text{V}}$ as the sum of n IID de Moivre($\theta_1, \theta_2, \ldots, \theta_k$) R$\vec{\text{V}}$ that is defined next.

Model 15 (de Moivre($\theta_1, \theta_2, \ldots, \theta_k$) $\mathbf{R\vec{V}}$**)** *The PMF of the* de Moivre($\theta_1, \theta_2, \ldots, \theta_k$) $R\vec{V}$ $X := (X_1, X_2, \ldots, X_k)$ *taking value* $x := (x_1, x_2, \ldots, x_k) \in \{e_1, e_2, \ldots, e_k\}$*, where the* e_i*'s are orthonormal basis vectors in* $\mathbb{R}^k$ *is:*

$$f(x; \theta_1, \theta_2, \ldots, \theta_k) := \mathbf{P}(X = x) = \sum_{i=1}^{k} \theta_i \mathbb{1}_{\{e_i\}}(x) = \begin{cases} \theta_1 & \text{if} \quad x = e_1 := (1, 0, \ldots, 0) \in \mathbb{R}^k \\ \theta_1 & \text{if} \quad x = e_1 := (0, 1, \ldots, 0) \in \mathbb{R}^k \\ \vdots & \\ \theta_k & \text{if} \quad x = e_k := (0, 0, \ldots, 1) \in \mathbb{R}^k \\ 0 & \text{otherwise} \end{cases}$$

Of course, $\sum_{i=1}^{k} \theta_i = 1$.

When we add n IID de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ R$\vec{\text{V}}$ together, we get the Multinomial$(n, \theta_1, \theta_2, \ldots, \theta_k)$ R$\vec{\text{V}}$ as defined below.

Model 16 (Multinomial$(n, \theta_1, \theta_2, \ldots, \theta_k)$ **R$\vec{\text{V}}$)** *We say that a R$\vec{\text{V}}$* $Y := (Y_1, Y_2, \ldots, Y_k)$ *obtained from the sum of n IID* de Moivre$(\theta_1, \theta_2, \ldots, \theta_k)$ *R$\vec{\text{V}}$s with realisations*

$$y := (y_1, y_2, \ldots, y_k) \in \mathbb{Y} := \{(y_1, y_2, \ldots, y_k) \in \mathbb{Z}_+^k : \sum_{i=1}^{k} y_i = n\}$$

has the PMF given by:

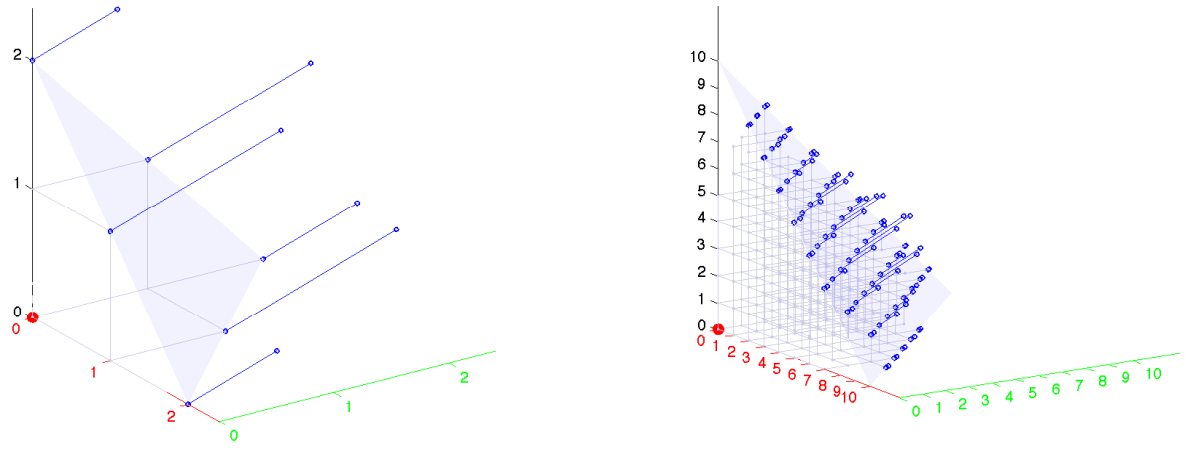
$$f(y; n, \theta) := f(y; n, \theta_1, \theta_2, \ldots, \theta_k) := \mathbf{P}(Y = y; n, \theta_1, \theta_2, \ldots, \theta_k) = \binom{n}{y_1, y_2, \ldots, y_k} \prod_{i=1}^{k} \theta_i^{y_i} \ ,$$

where, the multinomial coefficient:

$$\binom{n}{y_1, y_2, \ldots, y_k} := \frac{n!}{y_1! y_2! \cdots y_k!} \ .$$

Note that the marginal PMF of Y_j is Binomial(n, θ_j) *for any $j = 1, 2, \ldots, k$.*

Figure 6.13: Septcunx on the Cartesian co-ordinates. Simulations of Multinomial$(n = 2, \theta_1 = 1/3, \theta_2 = 1/3, \theta_3 = 1/3)$ R$\vec{\text{V}}$ as the sum of n IID de Moivre$(\theta_1 = 1/3, \theta_2 = 1/3, \theta_3 = 1/3)$ R$\vec{\text{V}}$s over $\{(1,0,0), (0,1,0), (0,0,1)\}$ with probabilities $\{\theta_1, \theta_2, \theta_3\}$, respectively. The blue lines perpendicular to the sample space of the Multinomial$(3, \theta_1, \theta_2, \theta_3)$ R$\vec{\text{V}}$, i.e. the plane in $\mathbb{R}^3$ connecting $(n, 0, 0)$, $(0, n, 0)$ and $(0, 0, n)$, are the density histogram of the samples.



(a) Thousand Samples with $n = 2$

(b) Thousand Samples with $n = 10$

We can visualise the Multinomial$(n, \theta_1, \theta_2, \theta_3)$ process as a sum of n IID de Moivre$(\theta_1, \theta_2, \theta_3)$ R$\vec{\text{V}}$s via a three dimensional extension of the Quincunx called the "Septcunx" and relate the number of paths that lead to a given trivariate sum (y_1, y_2, y_3) with $\sum_{i-1}^{3} y_i = n$ as the multinomial

coefficient $\frac{n!}{y_1!y_2!y_3!}$. In the Septcunx, balls choose from one of three paths along e_1, e_2 and e_3 with probabilities θ_1, θ_2 and θ_3, respectively, in an IID manner at each of the n levels, before they collect at buckets placed at the integral points in the 3-simplex, $\mathbb{Y} = \{(y_1, y_2, y_3) \in \mathbb{Z}_+^3 : \sum_{i=1}^3 y_i = n\}$. Once again, we can visualise that the sum of n IID de Moivre$(\theta_1, \theta_2, \theta_3)$ R$\vec{\text{V}}$s constitute the Multinomial$(n, \theta_1, \theta_2, \theta_3)$ R$\vec{\text{V}}$ as depicted in Figure 6.13.

Labwork 26 (PDF of Multinomial(n, θ) **R$\vec{\text{V}}$)** *We can implement the following* MATLAB *function* `MultinomialPdf` *to compute the PDF of the* Multinomial(n, θ) *R$\vec{V}$ where* $\theta := (\theta_1, \theta_2, \ldots, \theta_k)$ *is a point in the* k*-simplex* $\triangle_k$ *as follows:*

MultinomialPdf.m

```
function MP = MultinomialPdf(x,n,theta)
% returns the multinomial Pdf of x(1),x(2),...,x(k) given theta(1),...,theta(k).
% x and theta are vectors and sum to the scalars n and 1, respectively and 0 <= x(i) <= n
% Double precision numbers in MATLAB only have about 15 digits, the answer is accurate for n <= 21
NonZeroXs = find(x>0);
MP=exp(log(factorial(n))+sum((log(theta(NonZeroXs)) .* x(NonZeroXs)) - log(factorial(x(NonZeroXs)))));
```

We can call this function to evaluate the PDF at a specific sample $x = (x_1, x_2, \ldots, x_k)$ *as follows:*

```
>> MultinomialPdf([2 0 0],2,[1/3 1/3 1/3])
ans =    0.1111
>> MultinomialPdf([0 2 0],2,[1/3 1/3 1/3])
ans =    0.1111
>> MultinomialPdf([0 0 2],2,[1/3 1/3 1/3])
ans =    0.1111
>> MultinomialPdf([1 1 0],2,[1/3 1/3 1/3])
ans =    0.2222
>> MultinomialPdf([1 0 1],2,[1/3 1/3 1/3])
ans =    0.2222
>> MultinomialPdf([0 1 1],2,[1/3 1/3 1/3])
ans =    0.2222
```

Simulation 15 *Using the identity matrix* I *in* $\mathbb{R}^3$ *that can be created in* MATLAB *using the* `eye(3)` *command, and the* de Moivre$(1/3, 1/3, 1/3)$ *RV sampler, simulate vector-valued samples from* de Moivre$(1/3, 1/3, 1/3)$ *R$\vec{V}$. Finally add up* $n = 10$ *samples from* de Moivre$(1/3, 1/3, 1/3)$ *R$\vec{V}$ to produce samples from* Multinomial$(10, 1/3, 1/3, 1/3)$ *R$\vec{V}$.*

6.7 von Neumann Rejection Sampler (RS)

Rejection sampling [John von Neumann, 1947, in *Stanislaw Ulam 1909-1984*, a special issue of Los Alamos Science, Los Alamos National Lab., 1987, p. 135-136] is a Monte Carlo method to draw independent samples from a target RV X with probability density $f(x)$, where $x \in \mathbb{X} \subset \mathbb{R}^k$. Typically, the target density f is only known up to a constant and therefore the (normalised) density f itself may be unknown and it is difficult to generate samples directly from X.

Suppose we have another density or mass function g for which the following are true:

(a) we can generate random variables from g;

(b) the support of g contains the support of f, i.e. $\mathbb{Y} \supset \mathbb{X}$;

(c) a constant $a > 1$ exists, such that:

$$f(x) \leq ag(x). \tag{6.19}$$

for any $x \in \mathbb{X}$, the support of X. Then x can be generated from Algorithm 8.

Algorithm 8 Rejection Sampler (RS) of von Neumann

1: *input:*

(1) a target density $f(x)$,

(2) a proposal density $g(x)$ satisfying (a), (b) and (c) above.

2: *output:* a sample x from RV X with density f
3: **repeat**
4: Generate $y \sim g$ and $u \sim \text{Uniform}(0,1)$
5: **until** $u \leq \frac{f(y)}{ag(y)}$
6: *return:* $x \leftarrow y$

Proposition 5 (Fundamental Theorem of Simulation) *The von Neumann rejection sampler of Algorithm 8 produces a sample x from the random variable X with density $f(x)$.*

Proof: *We shall prove the result for the continuous case. For any real number t:*

$$F(t) = \mathbf{P}(X \leq t) = \mathbf{P}\left(Y \leq t \,|\, U \leq \frac{f(Y)}{ag(Y)}\right) = \frac{\mathbf{P}\left(Y \leq t, U \leq \frac{f(Y)}{ag(Y)}\right)}{\mathbf{P}\left(U \leq \frac{f(Y)}{ag(Y)}\right)}$$
$$= \frac{\int_{-\infty}^{t}\left(\int_0^{f(y)/ag(y)} 1du\right) g(y)dy}{\int_{-\infty}^{\infty}\left(\int_0^{f(y)/ag(y)} 1du\right) g(y)dy} = \frac{\int_{-\infty}^{t}\left(\frac{f(y)}{ag(y)}\right) g(y)dy}{\int_{-\infty}^{\infty}\left(\frac{f(y)}{ag(y)}\right) g(y)dy}$$
$$= \int_{-\infty}^{t} f(y)dy$$

Simulation 16 (Rejection Sampling Normal$(0,1)$ **with** Laplace(1) **proposals)** *Suppose we wish to generate from $X \sim$ Normal$(0,1)$. Consider using the rejection sampler with proposals from $Y \sim$ Laplace(1) (using inversion sampler of Simulation 3). The support of both RVs is $(-\infty, \infty)$. Next:*

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \text{ and } g(x) = \frac{1}{2}\exp(-|x|)$$

and therefore:

$$\frac{f(x)}{g(x)} = \sqrt{\frac{2}{\pi}} \exp\left(|x| - \frac{x^2}{2}\right) \leq \sqrt{\frac{2}{\pi}} \exp\left(\frac{1}{2}\right) = a \approx 1.3155 \ .$$

Hence, we can use the rejection method with:

$$\frac{f(y)}{ag(y)} = \frac{f(y)}{g(y)}\frac{1}{a} = \sqrt{\frac{2}{\pi}} \exp\left(|y| - \frac{y^2}{2}\right) \frac{1}{\sqrt{\frac{2}{\pi}}\exp\left(\frac{1}{2}\right)} = \exp\left(|y| - \frac{y^2}{2} - \frac{1}{2}\right)$$

Let us implement a rejection sampler as a function in the M-file `RejectionNormalLaplace.m` *by reusing the function in* `LaplaceInvCDF.m`.

RejectionNormalLaplace.m

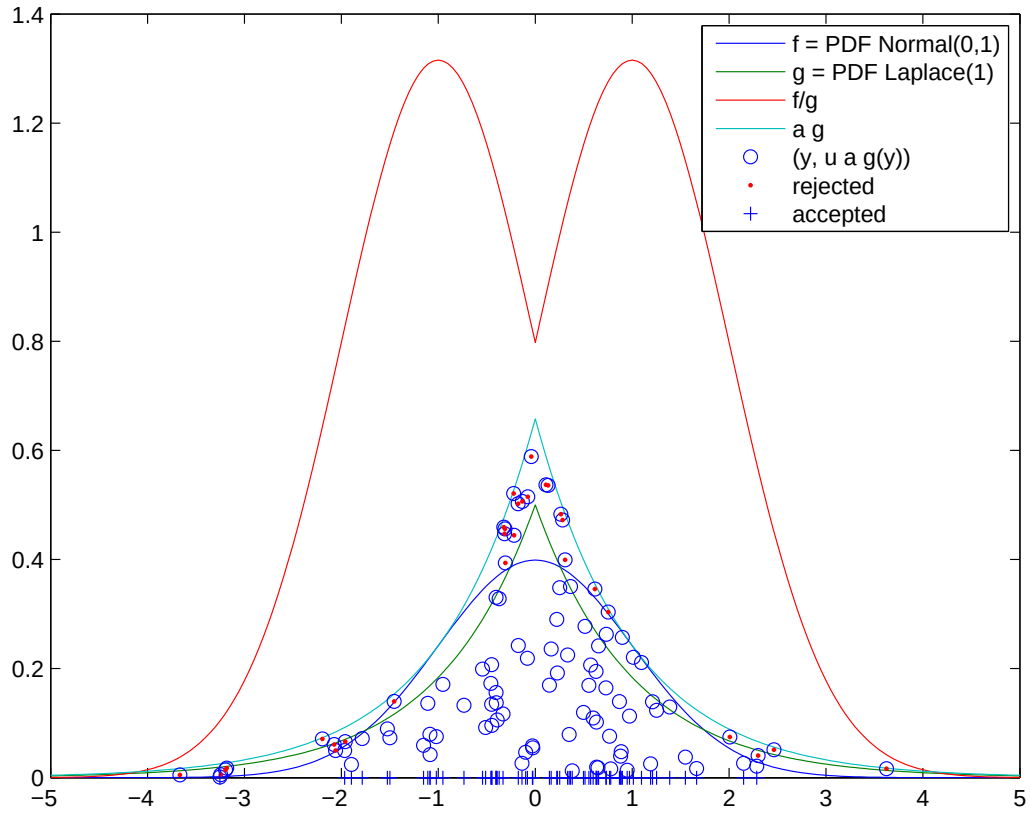
```
function x = RejectionNormalLaplace()
Accept = 0; % a binary variable to indicate whether a proposed point is accepted
while ~Accept % ~ is the logical NOT operation
    y = LaplaceInvCDF(rand(),1); % sample Laplace(1) RV
```

```
    Bound = exp( abs(y) -  (y*y+1)/2 );
    u = rand();
    if u <= Bound
        x = y;
        Accept = 1;
    end % if
end % while
```

We may obtain a large number of samples an plot them as a histogram using the following commands:

```
>> % use funarray to convert 1000 zeros into samples from the Normal(0,1)
>> y=arrayfun(@(x)(RejectionNormalLaplace()),zeros(1,1000));
>> hist(y,20) % histogram with 20 bins
```

Figure 6.14: Rejection Sampling from $X \sim \text{Normal}(0,1)$ with PDF f based on 100 proposals from $Y \sim \text{Laplace}(1)$ with PDF g.



Classwork 16 *The condition $f(x) \leq ag(x)$ is equivalent to $f(x)/g(x) \leq a$, which says that $f(x)/g(x)$ must be bounded; therefore, g must have higher tails than f. The rejection method cannot be used to generate from a Cauchy distribution using a normal distribution, because the latter has lower tails than the former.*

The next result tells us how many iterations of the algorithm are needed, on average, to get a sample value from a RV with PDF f.

Proposition 6 (Acceptance Probability of RS) *The expected number of iterations of the rejection algorithm to get a sample x is the constant a.*

Proof: *For the continuous case:*

$$\mathbf{P}(\text{`accept } y\text{'}) = \mathbf{P}\left(u \leq \frac{f(y)}{ag(y)}\right) = \int_{-\infty}^{\infty}\left(\int_0^{f(y)/ag(y)} du\right) g(y)dy = \int_{-\infty}^{\infty} \frac{f(y)}{ag(y)} g(y)dy = \frac{1}{a}.$$

And the number of proposals before acceptance is the Geometric$(1/a)$ *RV with expectation* $\frac{1}{1/a} = a$.

The closer $ag(x)$ is to $f(x)$, especially in the tails, the closer a will be to 1, and hence the more efficient the rejection method will be.

The rejection method can still be used only if the un-normalised form of f or g (or both) is known. In other words, if we use:

$$f(x) = \frac{\tilde{f}(x)}{\int \tilde{f}(x)dx} \text{ and } g(x) = \frac{\tilde{g}(x)}{\int \tilde{g}(x)dx}$$

we know only $\tilde{f}(x)$ and/or $\tilde{g}(x)$ in closed-form. Suppose the following are satisfied:

(a) we can generate random variables from g;

(b) the support of g contains the support of f, i.e. $\mathbb{Y} \supset \mathbb{X}$;

(c) a constant $\tilde{a} > 0$ exists, such that:

$$\tilde{f}(x) \leq \tilde{a}\tilde{g}(x), \tag{6.20}$$

for any $x \in \mathbb{X}$, the support of X. Then x can be generated from Algorithm 9.

Algorithm 9 Rejection Sampler (RS) of von Neumann – target shape

1: *input:*

(1) shape of a target density $\tilde{f}(x) = \left(\int \tilde{f}(x)dx\right) f(x)$,

(2) a proposal density $g(x)$ satisfying (a), (b) and (c) above.

2: *output:* a sample x from RV X with density f

3: **repeat**

4: Generate $y \sim g$ and $u \sim$ Uniform$(0, 1)$

5: **until** $u \leq \frac{\tilde{f}(y)}{\tilde{a}\tilde{g}(y)}$

6: *return:* $x \leftarrow y$

Now, the expected number of iterations to get an x is no longer $\tilde{a}$ but rather the integral ratio:

$$\left(\frac{\int_{\mathbb{X}} \tilde{f}(x)dx}{\int_{\mathbb{Y}} \tilde{a}\tilde{g}(y)dy}\right)^{-1}.$$

The **Ziggurat Method** [G. Marsaglia and W. W. Tsang, SIAM Journal of Scientific and Statistical Programming, volume 5, 1984] is a rejection sampler that can efficiently draw samples from the $Z \sim$ Normal$(0, 1)$ RV. The MATLAB function `randn` uses this method to produce samples from Z. See `http://www.mathworks.com/company/newsletters/news_notes/clevescorner/spring01_cleve.html` or `http://en.wikipedia.org/wiki/Ziggurat_algorithm` for more details.

Labwork 27 (Gaussian Sampling with `randn`) *We can use* MATLAB *function* `randn` *that implements the Ziggurat method to draw samples from an RV* $Z \sim \text{Normal}(0,1)$ *as follows:*

```
>> randn('state',67678); % initialise the seed at 67678 and method as Ziggurat -- TYPE help randn
>> randn % produce 1 sample from Normal(0,1) RV
ans =    1.5587
>> randn(2,8) % produce an 2 X 8 array of samples from Normal(0,1) RV
ans =
    1.2558    0.7834    0.6612    0.3247    0.1407    1.0562    0.8034    1.2970
   -0.5317    0.0417   -0.3454    0.6182   -1.4162    0.4796   -1.5015    0.3718
```

If we want to produce samples from $X \sim \text{Normal}(\mu, \sigma^2)$ *with some user-specified* μ *and* σ*, then we can use the following relationship between* X *and* $Z \sim \text{Normal}(0,1)$*:*

$$X \leftarrow \mu + \sigma Z, \qquad Z \sim \text{Normal}(0,1) \ .$$

Suppose we want samples from $X \sim \text{Normal}(\mu = \pi, \sigma^2 = 2)$*, then we can do the following:*

```
>> randn('state',679); % initialise the seed at 679 and method as Ziggurat -- TYPE help randn
>> mu=pi % set the desired mean parameter mu
mu =    3.1416
>> sigma=sqrt(2) % set the desired standard deviation parameter sigma
sigma =    1.4142
>> mu + sigma * randn(2,8) % produces a 2 X 8 array of samples from Normal(3.1416,1.4.42)
ans =
    1.3955    1.7107    3.9572    3.2618    6.1652    2.6971    2.4940    4.5928
    0.8442    4.7617    3.5397    5.0282    1.6139    5.0977    2.0477    2.3286
```

6.8 Other Continuous Random Variables

Here, we see other common continuous RVs that can be simulated from transforming RVs we have already encountered.

Model 17 ($\text{Gamma}(\lambda, k)$ **RV)** *Given a scale parameter* $\lambda > 0$ *and a shape parameter* $k > 0$*, the RV* X *is said to be* $\text{Gamma}(\lambda, k)$ *distributed if its PDF is:*

$$f(x; \lambda, k) = \frac{1}{\Gamma(k)} \lambda \exp(-\lambda x)(\lambda x)^{k-1}, \qquad x \geq 0 \ ,$$

where, the gamma function which interpolates the factorial function is:

$$\Gamma(k) := \int_0^\infty \exp(-y) y^{k-1} dy \ .$$

When $k \in \mathbb{N}$*, then* $\Gamma(k) = (k-1)!$*. The DF of* X *is:*

$$F(x; \lambda, k) = \mathbb{1}_{\mathbb{R}_{>0}}(x) \frac{1}{\Gamma(k)} \int_0^{\lambda x} y^{k-1} \exp(-y) dy = \begin{cases} 0 & \text{if } x \leq 0 \\ \frac{1}{\Gamma(k)} \int_0^{\lambda x} y^{k-1} \exp(-y) dy & \text{if } x > 0 \end{cases}$$

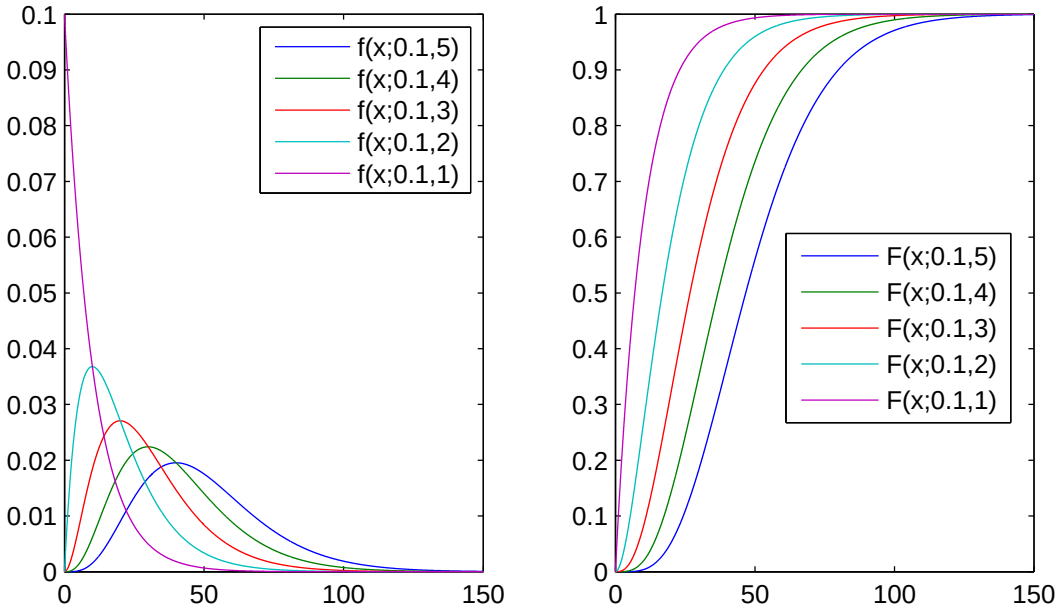
$\frac{1}{\Gamma(k)} \int_0^{\lambda x} y^{k-1} \exp(-y) dy$ *is called the incomplete Gamma function.*

The Gamma function and the incomplete Gamma function are available as MATLAB functions `gamma` and `gammainc`, respectively. Thus, `gamma(k)` returns $\Gamma(k)$ and `gammainc(lambda*x,k)` returns $F(\texttt{x};\texttt{lambda},\texttt{k})$. Using these functions, it is straightforward to evaluate the PDF and CDF of $X \sim \text{Gamma}(\lambda, k)$. We use the following script to get a sense for the impact upon the PDF and CDF of the shape parameter k as it ranges in $\{1, 2, 3, 4, 5\}$ for a given scale parameter $\lambda = 0.1$.

PlotPdfCdfGamma.m

```
lambda=0.1; % choose soem scale parameter
Xs=0:0.01:150; % choose some x values
% Plot PDFs for k=5,4,3,2,1
k=5; fXsk5=(1/gamma(k))*(lambda*exp(-lambda*Xs).*(lambda*Xs).^(k-1));% PDF for k=5
k=4; fXsk4=(1/gamma(k))*(lambda*exp(-lambda*Xs).*(lambda*Xs).^(k-1));% PDF for k=4
k=3; fXsk3=(1/gamma(k))*(lambda*exp(-lambda*Xs).*(lambda*Xs).^(k-1));% PDF for k=3
k=2; fXsk2=(1/gamma(k))*(lambda*exp(-lambda*Xs).*(lambda*Xs).^(k-1));% PDF for k=2
k=1; fXsk1=(1/gamma(k))*(lambda*exp(-lambda*Xs).*(lambda*Xs).^(k-1));% PDF for k=1
clf; % clear any previous figures
subplot(1,2,1); % make first PDF plot
plot(Xs,fXsk5, Xs, fXsk4, Xs, fXsk3, Xs, fXsk2, Xs, fXsk1)
legend('f(x;0.1,5)','f(x;0.1,4)','f(x;0.1,3)','f(x;0.1,2)','f(x;0.1,1)')
subplot(1,2,2) % make second CDF plots using MATLAB's gammainc (incomplete gamma function)
plot(Xs,gammainc(lambda*Xs,5), Xs,gammainc(lambda*Xs,4), Xs,gammainc(lambda*Xs,3),...
    Xs,gammainc(lambda*Xs,2), Xs,gammainc(lambda*Xs,1))
legend('F(x;0.1,5)','F(x;0.1,4)','F(x;0.1,3)','F(x;0.1,2)','F(x;0.1,1)')
```

Figure 6.15: PDF and CDF of $X \sim \text{Gamma}(\lambda = 0.1, k)$ with $k \in \{1, 2, 3, 4, 5\}$.



Note that if $X \sim \text{Gamma}(\lambda, 1)$ then $X \sim \text{Exponential}(\lambda)$, since:

$$f(x; \lambda, 1) = \frac{1}{\Gamma(1)} \lambda \exp(-\lambda x)(\lambda x)^{1-1} = \frac{1}{(1-1)!} \lambda \exp(-\lambda x) = \lambda \exp(-\lambda x) \ .$$

More generally, if $X \sim \text{Gamma}(\lambda, k)$ and $k \in \mathbb{N}$, then $X \sim \sum_{i=1}^{k} Y_i$, where $Y_i \overset{IID}{\sim} \text{Exponential}(\lambda)$ RVS, i.e. the sum of k IID Exponential(λ) RVs forms the model for the Gamma(λ, k) RV. If you model the inter-arrival time of buses at a bus-stop by IID Exponential(λ) RV, then you can think of the arrival time of the k^{th} bus as a Gamma(λ, k) RV.

Simulation 17 (Gamma(λ, k) **for integer** k**)** *Using this relationship we can simulate from* $X \sim$ Gamma(λ, k)*, for an integer-valued* k*, by simply summing* k *IID samples from* Exponential(λ) *RV as follows:*

```
>> lambda=0.1; %declare some lambda parameter
>> k=5; % declare some k parameter (has to be integer)
>> rand('twister',7267); % initialise the fundamental sampler
>> % sum k IID Exponential(lambda) samples for one desired sample from Gamma(lambda,k)
>> x= sum(-1/lambda*log(rand(k,1)))
x =   28.1401
>> % sum the 10 columns of k X 10 IID Exponential(lambda) samples for 10 desired samples from Gamma(lambda,k)
>> x= sum(-1/lambda*log(rand(k,10)))
x =
   83.8150   61.2674   80.3683  103.5748   48.4454   20.2269   93.8310   56.1909   77.0656   29.0851
```

Model 18 (Lognormal(λ, ζ)**)** *X has a* Lognormal(λ, ζ) *distribution if* $\log(X)$ *has a* Normal(λ, ζ^2) *distribution. The location parameter* $\lambda = \mathbf{E}(\log(X)) > 0$ *and the scale parameter* $\zeta > 0$*. The PDF is:*

$$f(x; \lambda, \zeta) = \frac{1}{\sqrt{2\pi}\zeta x} \exp\left(-\frac{1}{2\zeta^2}(\log(x) - \lambda)^2\right), \qquad x > 0 \tag{6.21}$$

No closed form expression for $F(x; \lambda, \zeta)$ *exists and it is simply defined as:*

$$F(x; \lambda, \zeta) = \int_0^x f(y; \lambda, \zeta)\, dy$$

We can express $F(x; \lambda, \zeta)$ *in terms of* Φ *(and, in turn, via the associated error function* erf*) as follows:*

$$F(x; \lambda, \zeta) = \Phi\left(\frac{\log(x) - \lambda}{\zeta}\right) = \frac{1}{2}\ \text{erf}\left(\frac{\log(x) - \lambda}{\sqrt{2}\zeta}\right) + \frac{1}{2} \tag{6.22}$$

Labwork 28 *Transform a sequence of samples obtained from the fundamental sampler to those from the* Lognormal(λ_C, ζ_C) *RV* C *by using only Algorithm 4 or* MATLAB*'s* `randn` *as an intermediate step. [Hint: If* Y *is a* Normal(λ, ζ^2) *RV, then* $Z = e^Y$ *is said to be a* Lognormal(λ, ζ) *RV.]*

1. *Seed the fundamental sampler by your Student ID,*
2. *generate* 1000 *samples from an RV* $C \sim$ Lognormal$(\lambda = 10.36, \zeta = 0.26)$ *by exponentiating the samples from the* Normal$(10.36, 0.26^2)$ *RV and*
3. *and report:*
   (a) *how many of the samples are larger than* 35000*,*
   (b) *the sample mean, and*
   (c) *the sample standard deviation.*

Beta RV

Chi-Square

F distribution

t-distribution

Weibul

Heavy-tail family

6.9 Other Random Vectors

Multivariate Normal

Uniform Distribution on Sphere

Dirichlet Distribution

6.10 Summary of Random Variables

Model	PDF	Mean	Variance
Point Mass(θ)	$\mathbb{1}_{\{\theta\}}(x)$	θ	0
Bernoulli(θ)	$\theta^x(1-\theta)^{1-x}\mathbb{1}_{\{0,1\}}(x)$	θ	$\theta(1-\theta)$
Binomial(n,θ)	$\binom{n}{\theta}\theta^x(1-\theta)^{n-x}\mathbb{1}_{\{0,1,\ldots,n\}}(x)$	$n\theta$	$n\theta(1-\theta)$
Geometric(θ)	$\theta(1-\theta)^x\mathbb{1}_{\mathbb{Z}_+}(x)$	$\frac{1}{\theta}-1$	$\frac{1-\theta}{\theta^2}$
Poisson(λ)	$\frac{\lambda^x e^{-\lambda}}{x!}\mathbb{1}_{\mathbb{Z}_+}(x)$	λ	λ
Uniform(θ_1,θ_2)	$\mathbb{1}_{[\theta_1,\theta_2]}(x)/(\theta_2-\theta_1)$	$\frac{\theta_1+\theta_2}{2}$	$\frac{(\theta_2-\theta_1)^2}{12}$
Normal(μ,σ^2)	$\frac{1}{\sigma\sqrt{2\pi}}e^{(x-\mu)^2/(2\sigma^2)}$	μ	σ^2
Exponential(λ)	$\lambda e^{-\lambda x}$	λ^{-1}	λ^{-2}
Gamma(α,β)	$\frac{x^{\alpha-1}e^{-x/\beta}}{\Gamma(\alpha)\beta^\alpha}$	$\alpha\beta$	$\alpha\beta^2$
Beta(α,β)	$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1}$	$\frac{\alpha}{\alpha+\beta}$	$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
χ^2_p	$\frac{1}{\Gamma(p/2)2^{p/2}}x^{(p-2)-1}e^{-x/2}$	p	$2p$

Table 6.2: Random Variables with PDF, Mean and Variance

6.11 Statistical Experiments

We formalize the notion of a staistical experiment. Let us first motivate the need for a statistical experiment. Recall that statistical inference or learning is the process of using observations or data to infer the distribution that generated it. A generic question is:

Given realizations from $X_1, X_2, \ldots, X_n \sim$ some unknown DF F, how do we infer F?

However, to make this question tractable or even sensible it is best to restrict ourselves to a particular class or family of DFs that may be assumed to contain the unknown DF F.

Definition 36 *A statistical experiment $\mathcal{E}$ is a set of probability distributions (DFs, PDFs or PMFs) $\mathbb{P} := \{P_\theta : \theta \in \Theta\}$ associated with a RV X and indexed by the set Θ. We refer to Θ as the parameter space or the index set and $d : \Theta \to \mathbb{P}$ that associates to each $\theta \in \Theta$ a probability $P_\theta \in \mathbb{P}$ as the index map.*

Next, let's formally consider some experiments we have already encountered.

Experiment 2 (The Fundamental Experiment) *The 'uniformly pick a number in the interval $[0,1]$' experiment is the following singleton family of DFs :*

$$\mathbb{P} = \{\, F(x) = x\mathbb{1}_{[0,1]}(x) \,\}$$

where, the only distribution $F(x)$ in the family $\mathbb{P}$ is a re-expression of (3.7) *using the indicator function $\mathbb{1}_{[0,1]}(x)$. The parameter space of the fundamental experiment is a singleton whose DF is its own inverse, ie. $F(x) = F^{[-1]}(x)$.*

Experiment 3 (Bernoulli) *The 'toss 1 times' experiment is the following family of densities (PMFs) :*

$$\mathbb{P} = \{ f(x;p) : p \in [0,1] \}$$

where, $f(x;p)$ is given in (3.4)*. The one dimensional parameter space or index set for this experiment is $\mathbb{\Theta} = [0,1] \subset \mathbb{R}$.*

Experiment 4 (Point Mass) *The 'deterministically choose a specific real number' experiment is the following family of DFs :*

$$\mathbb{P} = \{ F(x;a) : a \in \mathbb{R} \}$$

where, $F(x;a)$ is given in (6.11)*. The one dimensional parameter space or index set for this experiment is $\mathbb{\Theta} = \mathbb{R}$, the entire real line.*

Note that we can use the PDF's or the DF's to specify the family $\mathbb{P}$ of an experiment. When an experiment can be parametrized by finitely many parameters it is said to a **parametric** experiment. Experiment 3 involving discrete RVs as well as Experiment 4 are **parametric** since they both have only one parameter (the parameter space is one dimensional for Experiments 3 and 4). The Fundamental Experiment 2 involving the continuous RV of Model 2 is also parametric since its parameter space, being a point, is zero-dimensional. The next example is also parametric and involves $(k-1)$-dimensional families of discrete RVs.

Experiment 5 (de Moivre[k]) *The 'pick a number from the set $[k] := \{1, 2, \ldots, k\}$ somehow' experiment is the following family of densities (PMFs) :*

$$\mathbb{P} = \{ f(x;\theta_1, \theta_2, \ldots, \theta_k) : (\theta_1, \theta_2, \ldots, \theta_k) \in \triangle_k \}$$

where, $f(x;\theta_1, \theta_2, \ldots, \theta_k)$ is any PMF such that

$$f(x;\theta_1, \theta_2, \ldots, \theta_k) = \theta_x, \qquad x \in \{1, 2, \ldots, k\} \ .$$

The $k-1$ dimensional parameter space $\mathbb{\Theta}$ is the k-Simplex $\triangle_k$. This as an 'exhaustive' experiment since all possible densities over the finite set $[k] := \{1, 2, \ldots, k\}$ are being considered that can be thought of as "the outcome of rolling a convex polyhedral die with k faces and an arbirtary center of mass specified by the θ_i's."

An experiment with infinite dimensional parameter space $\mathbb{\Theta}$ is said to be **nonparametric** . Next we consider two nonparametric experiments.

Experiment 6 (All DFs) *The 'pick a number from the Real line in an arbitrary way' experiment is the following family of distribution functions (DFs) :*

$$\mathbb{P} = \{ F(x;F) : F \text{ is a } DF \} = \mathbb{\Theta}$$

where, the DF $F(x;F)$ is indexed or parameterized by itself. Thus, the parameter space

$$\mathbb{\Theta} = \mathbb{P} = \{\text{all } DFs\}$$

is the infinite dimensional space of **All DFs** *".*

Figure 6.16: Geometry of the $\mathbb{\Theta}$'s for de Moivre[k] Experiments with $k \in \{1, 2, 3, 4\}$.

Next we consider a **nonparametric** experiment involving continuous RVs.

Experiment 7 (Sobolev Densities) *The 'pick a number from the Real line in some reasonable way' experiment is the following family of densities (pdfs) :*

$$\mathbb{P} = \left\{ f(x; f) : \int (f''(x))^2 < \infty \right\} = \mathbb{\Theta}$$

where, the density $f(x; f)$ is indexed by itself. Thus, the parameter space $\mathbb{\Theta} = \mathbb{P}$ is the infinite dimensional **Sobolev space** *of "not too wiggly functions".*

Some of the concrete problems involving experiments include:

- **Simulation:** Often it is necessary to simulate a RV with some specific distribution to gain insight into its features or simulate whole systems such as the air-traffic queues at 'London Heathrow' to make better management decisions.
- **Estimation:**
  1. **Parametric Estimation:** Using samples from some unknown DF F parameterized by some unknown θ, we can estimate θ from a statistic T_n called the estimator of θ using one of several methods (maximum likelihood, moment estimation, or parametric bootstrap).
  2. **Nonparametric Estimation of the DF:** Based on n IID observations from an unknown DF F, we can estimate it under the general assumption that $F \in \{\text{all DFs}\}$.
  3. **Confidence Sets:** We can obtain a $1 - \alpha$ confidence set for the point estimates, of the unknown parameter $\theta \in \mathbb{\Theta}$ or the unknown DF $F \in \{\text{all DFs}\}$
- **Hypothesis Testing:** Based on observations from some DF F that is hypothesized to belong to a subset $\mathbb{\Theta}_0$ of $\mathbb{\Theta}$ called the space of null hypotheses, we will learn to test (attempt to reject) the falsifiable null hypothesis that $F \in \mathbb{\Theta}_0 \subset \mathbb{\Theta}$.
- ...

Chapter 7

Limits of Random Variables

7.1 Convergence of Random Variables

This important topic is concerned with the limiting behavior of sequences of RVs

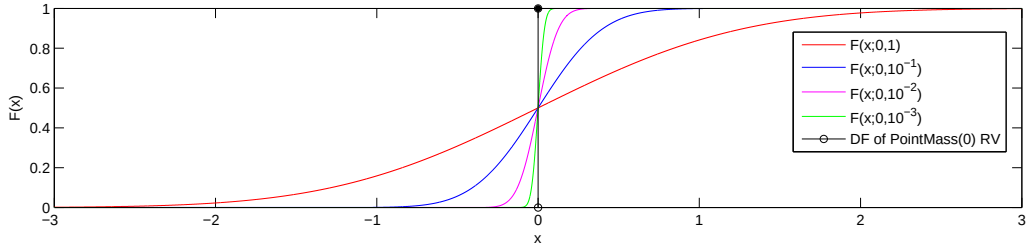
$$\{X_i\}_{i=1}^n := X_1, X_2, X_3, \ldots X_{n-1}, X_n \qquad \text{as } n \to \infty \ .$$

From a statistical viewpoint $n \to \infty$ is associated with the amount of data or information $\to \infty$. Refresh yourself with notions of convergence, limits and continuity in the real line (§ 1.7) before proceeding further.

Classwork 17 *Suppose you are given an independent sequence of RVs* $\{X_i\}_{i=1}^n$, *where* $X_i \sim \text{Normal}(0, 1/i)$. *How would you talk about the convergence of* $X_n \sim \text{Normal}(0, 1/n)$ *as* n *approaches* ∞ *? Take a look at Figure 7.1 for insight. The probability mass of* X_n *increasingly concentrates about* 0 *as* n *approaches* ∞ *and the variance* $1/n$ *approaches* 0, *as depicted in Figure 7.1. Based on this observation, can we expect* $\lim_{n\to\infty} X_n = X$, *where the limiting RV* $X \sim$ Point Mass(0) *?*

The answer is **no**. *This is because* $\mathbf{P}(X_n = X) = 0$ *for any* n, *since* $X \sim$ Point Mass(0) *is a discrete RV with exactly one outcome* 0 *and* $X_n \sim \text{Normal}(0, 1/n)$ *is a continuous RV for every* n, *however large. In other words, a continuous RV, such as* X_n, *has* 0 *probability of realizing any single real number in its support, such as* 0.

Figure 7.1: Distribution functions of several Normal(μ, σ^2) RVs for $\sigma^2 = 1, \frac{1}{10}, \frac{1}{100}, \frac{1}{1000}$.



Thus, we need more sophisticated notions of convergence for sequences of RVs. Two such notions are formalized next as they are minimal prerequisites for a clear understanding of three basic propositions in Statistics :

1. Weak Law of Large Numbers,
2. Central Limit Theorem,
3. Gilvenko-Cantelli Theorem.

Definition 37 (Convergence in Distribution) *Let $X_1, X_2, \ldots,$ be a sequence of RVs and let X be another RV. Let F_n denote the DF of X_n and F denote the DF of X. The we say that X_n converges to X in distribution, and write:*

$$X_n \rightsquigarrow X$$

if for any real number t at which F is continuous,

$$\lim_{n\to\infty} F_n(t) = F(t) \qquad \textit{[in the sense of Definition 4].}$$

The above limit, by (3.2) *in our Definition 12 of a DF, can be equivalently expressed as follows:*

$$\begin{aligned} &\lim_{n\to\infty} \mathbf{P}(\ \{\omega : X_n(\omega) \le t\}\ ) = \mathbf{P}(\ \{\omega : X(\omega) \le t\}\ ), \\ \textit{i.e.}\quad &\mathbf{P}(\ \{\omega : X_n(\omega) \le t\}\ ) \to \mathbf{P}(\ \{\omega : X(\omega) \le t\}\ ), \quad \textit{as} \quad n \to \infty\ . \end{aligned}$$

Definition 38 (Convergence in Probability) *Let $X_1, X_2, \ldots,$ be a sequence of RVs and let X be another RV. Let F_n denote the DF of X_n and F denote the DF of X. The we say that X_n converges to X in probability, and write:*

$$X_n \xrightarrow{P} X$$

if for every real number $\epsilon > 0$,

$$\lim_{n\to\infty} \mathbf{P}(|X_n - X| > \epsilon) = 0 \qquad \textit{[in the sense of Definition 4].}$$

Once again, the above limit, by (3.1) *in our Definition 11 of a RV, can be equivalently expressed as follows:*

$$\lim_{n\to\infty} \mathbf{P}(\ \{\omega : |X_n(\omega) - X(\omega)| > \epsilon\}\ ) = 0, \qquad \textit{ie,} \qquad \mathbf{P}(\ \{\omega : |X_n(\omega) - X(\omega)| > \epsilon\}\ ) \to 0, \quad \textit{as} \quad n \to \infty\ .$$

Let us revisit the problem of convergence in Classwork 17 armed with our new notions of convergence.

Example 18 *Suppose you are given an independent sequence of RVs $\{X_i\}_{i=1}^n$, where $X_i \sim \text{Normal}(0, 1/i)$ with DF F_n and let $X \sim$ Point Mass(0) with DF F. We can formalize our observation in Classwork 17 that X_n is concentrating about 0 as $n \to \infty$ by the statement:*

$$X_n \textit{ is converging in distribution to } X\textit{, ie,} \qquad X_n \rightsquigarrow X\ .$$

Proof: *To check that the above statement is true we need to verify that the definition of convergence in distribution is satisfied for our sequence of RVs $X_1, X_2, \ldots$ and the limiting RV X. Thus, we need to verify that for any continuity point t of the* Point Mass(0) *DF F, $\lim_{n\to\infty} F_n(t) = F(t)$. First note that*

$$X_n \sim \text{Normal}(0, 1/n) \implies Z := \sqrt{n}X_n \sim \text{Normal}(0, 1)\ ,$$

and thus

$$F_n(t) = \mathbf{P}(X_n < t) = \mathbf{P}(\sqrt{n}X_n < \sqrt{n}t) = \mathbf{P}(Z < \sqrt{n}t) \ .$$

The only discontinuous point of F is 0 where F jump from 0 to 1.

When $t < 0$, $F(t)$, being the constant 0 function over the interval $(-\infty, 0)$, is continuous at t. Since $\sqrt{n}t \to -\infty$, as $n \to \infty$,

$$\lim_{n\to\infty} F_n(t) = \lim_{n\to\infty} \mathbf{P}(Z < \sqrt{n}t) = 0 = F(t) \ .$$

And, when $t > 0$, $F(t)$, being the constant 1 function over the interval $(0, \infty)$, is again continuous at t. Since $\sqrt{n}t \to \infty$, as $n \to \infty$,

$$\lim_{n\to\infty} F_n(t) = \lim_{n\to\infty} \mathbf{P}(Z < \sqrt{n}t) = 1 = F(t) \ .$$

Thus, we have proved that $X_n \rightsquigarrow X$ by verifying that for any t at which the Point Mass(0) *DF F is continuous, we also have the desired equality: $\lim_{n\to\infty} F_n(t) = F(t)$.*

However, note that

$$F_n(0) = \frac{1}{2} \neq F(0) = 1 \ ,$$

and so convergence fails at 0, i.e. $\lim_{n\to\infty} F_n(t) \neq F(t)$ at $t = 0$. But, $t = 0$ is not a continuity point of F and the definition of convergence in distribution only requires the convergence to hold at continuity points of F.

For the same sequence of RVs in Classwork 17 and Example 18 we are tempted to ask whether $X_n \sim \text{Normal}(0, 1/n)$ converges in probability to $X \sim$ Point Mass(0), i.e. whether $X_n \xrightarrow{P} X$. We need some elementary inequalities in Probability to help us answer this question. We visit these inequalities next.

Proposition 7 (Markov's Inequality) *Let $(\Omega, \mathcal{F}, P)$ be a probability triple and let $X = X(\omega)$ be a non-negative RV. Then,*

$$\mathbf{P}(X \geq \epsilon) \leq \frac{\mathbf{E}(X)}{\epsilon}, \qquad \textit{for any} \quad \epsilon > 0 \ . \tag{7.1}$$

Proof:

$$\begin{aligned} X &= X\mathbb{1}_{\{y:y\geq\epsilon\}}(x) + X\mathbb{1}_{\{y:y<\epsilon\}}(x) \\ &\geq X\mathbb{1}_{\{y:y\geq\epsilon\}}(x) \\ &\geq \epsilon\mathbb{1}_{\{y:y\geq\epsilon\}}(x) \end{aligned} \tag{7.2}$$

Finally, taking expectations on both sides of the above inequality and then using the fact that the expectation of an indicator function of an event is simply the probability of that event (3.14)*, we get the desired result:*

$$\mathbf{E}(X) \geq \epsilon\mathbf{E}(\mathbb{1}_{\{y:y\geq\epsilon\}}(x)) = \epsilon\mathbf{P}(X \geq \epsilon) \ .$$

Let us look at some immediate consequences of Markov's inequality.

Proposition 8 (Chebychev's Inequality) *For* **any** *RV X and any $\epsilon > 0$,*

$$\mathbf{P}(|X| > \epsilon) \leq \frac{\mathbf{E}(|X|)}{\epsilon} \tag{7.3}$$

$$\mathbf{P}(|X| > \epsilon) = \mathbf{P}(X^2 \geq \epsilon^2) \leq \frac{\mathbf{E}(X^2)}{\epsilon^2} \tag{7.4}$$

$$\mathbf{P}(|X - \mathbf{E}(X)| \geq \epsilon) = \mathbf{P}((X - \mathbf{E}(X))^2 \geq \epsilon^2) \leq \frac{\mathbf{E}(X - \mathbf{E}(X))^2}{\epsilon^2} = \frac{\mathbf{V}(X)}{\epsilon^2} \tag{7.5}$$

Proof: *All three forms of Chebychev's inequality are mere corollaries (careful reapplications) of Markov's inequality.*

Armed with Markov's inequality we next enquire the convergence in probability for the sequence of RVs in Classwork 17 and Example 18.

Example 19 *Does the the sequence of RVs* $\{X_n\}_{n=1}^{\infty}$*, where* $X_n \sim \text{Normal}(0, 1/n)$*, converge in probability to* $X \sim \text{Point Mass}(0)$*, i.e. does* $X_n \xrightarrow{P} X$ *?*

To find out if $X_n \xrightarrow{P} X$*, we need to show that for any* $\epsilon > 0$*,* $\lim_{n\to\infty} \mathbf{P}(|X_n - X| > \epsilon) = 0$*.*

Let ϵ *be any real number greater than* 0*, then*

$$\begin{aligned} \mathbf{P}(|X_n| > \epsilon) &= \mathbf{P}(|X_n|^2 > \epsilon^2) \\ &= \frac{\mathbf{E}(X_n^2)}{\epsilon^2} \qquad \textit{[by Markov's Inequality (7.1)]} \\ &= \frac{\frac{1}{n}}{\epsilon^2} \to 0, \quad \textit{as} \quad n \to \infty \qquad \textit{[in the sense of Definition 4].} \end{aligned}$$

Hence, we have shown that for any $\epsilon > 0$*,* $\lim_{n\to\infty} \mathbf{P}(|X_n - X| > \epsilon) = 0$ *and therefore by Definition 38,* $X_n \xrightarrow{P} X$ *or* $X_n \xrightarrow{P} 0$*.*

Convention: *When* X *has a* Point Mass(θ) *distribution and* $X_n \xrightarrow{P} X$*, we simply write* $X_n \xrightarrow{P} \theta$*.*

Now that we have been introduced to two notions of convergence for sequences of RVs we can begin to appreciate the statements of the basic limit theorems of Statistics.

7.2 Some Basic Limit Laws of Statistics

Proposition 9 (Weak Law of Large Numbers (WLLN)) *If we are given a sequence if independent and identically distributed RVs,* $X_1, X_2, \ldots \overset{\text{IID}}{\sim} X_1$ *and if* $\mathbf{E}(X_1)$ *exists, as per* (3.9)*, then the sample mean* $\overline{X}_n$ *converges in probability to the expectation of any one of the IID RVs, say* $\mathbf{E}(X_1)$ *by convention. More formally, we wtite:*

$$\textit{If} \quad X_1, X_2, \ldots \overset{\text{IID}}{\sim} X_1 \textit{ and if } \mathbf{E}(X_1) \textit{ exists, then } \overline{X}_n \xrightarrow{P} \mathbf{E}(X_1) \ .$$

Proof: *For simplicity, we will prove a slightly weaker result by assuming finite variance of* X_1*. Suppose* $\mathbf{V}(X_1) < \infty$*, then:*

$$\begin{aligned} \mathbf{P}(|\overline{X}_n - \mathbf{E}(\overline{X}_n)| \geq \epsilon) &= \frac{\mathbf{V}(\overline{X}_n)}{\epsilon^2} \qquad \textit{[by applying Chebychev's inequality (7.5) to the RV } \overline{X}_n\textit{]} \\ &= \frac{\frac{1}{n}\mathbf{V}(X_1)}{\epsilon^2} \qquad \textit{[by the IID assumption of } X_1, X_2, \ldots \textit{ we can apply (5.3)]} \end{aligned}$$

Therefore, for any given $\epsilon > 0$*,*

$$\begin{aligned} \mathbf{P}(|\overline{X}_n - \mathbf{E}(X_1)| \geq \epsilon) &= \mathbf{P}(|\overline{X}_n - \mathbf{E}(\overline{X}_n)| \geq \epsilon) \qquad \textit{[by the IID assumption of } X_1, X_2, \ldots, \mathbf{E}(\overline{X}_n) = \mathbf{E}(X_1)\textit{, as per (5.2)]} \\ &= \frac{\frac{1}{n}\mathbf{V}(X_1)}{\epsilon^2} \to 0, \quad \textit{as} \quad n \to \infty \ , \end{aligned}$$

or equivalently, $\lim_{n\to\infty} \mathbf{P}(|\overline{X}_n - \mathbf{E}(X_1)| \geq \epsilon) = 0$*. And the last statement is the definition of the claim made by the weak law of large numbers (WLLN), namely that* $\overline{X}_n \xrightarrow{P} \mathbf{E}(X_1)$ *.*

Heuristic Interpretation of WLLN: *The distribution of the sample mean RV* $\overline{X}_n$ *obtained from an independent and identically distributed sequence of RVs* $X_1, X_2, \ldots$ *[i.e. all the RVs* X_i*'s are independent of one another and have the same distribution function, and thereby the same expectation, variance and higher moments], concentrates around the expectation of any one of the RVs in the sequence, say that of the first one* $\mathbf{E}(X_1)$ *[without loss of generality], as* n *approaches infinity.*

Example 20 (Bernoulli WLLN and Galton's Quincunx) *We can appreciate the WLLN for* $\overline{X}_n = n^{-1}S_n = \sum_{i=1}^n X_i$, *where* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim}$ *Bernoulli*(p) *using the paths of balls dropped into a device built by Galton called the Quincunx.*

Proposition 10 (Central Limit Theorem (CLT)) *Let* $X_1, X_2, \ldots \overset{\text{IID}}{\sim} X_1$ *and suppose* $\mathbf{E}(X_1)$ *and* $\mathbf{V}(X_1)$ *exists, then*

$$\overline{X}_n = \frac{1}{n}\sum_{i=1}^n X_i \quad \rightsquigarrow \quad X \sim \text{Normal}\left(\mathbf{E}(X_1), \frac{\mathbf{V}(X_1)}{n}\right) , \tag{7.6}$$

$$\overline{X}_n - \mathbf{E}(X_1) \quad \rightsquigarrow \quad X - \mathbf{E}(X_1) \sim \text{Normal}\left(0, \frac{\mathbf{V}(X_1)}{n}\right) , \tag{7.7}$$

$$\sqrt{n}\left(\overline{X}_n - \mathbf{E}(X_1)\right) \quad \rightsquigarrow \quad \sqrt{n}\left(X - \mathbf{E}(X_1)\right) \sim \text{Normal}\left(0, \mathbf{V}(X_1)\right) , \tag{7.8}$$

$$Z_n := \frac{\overline{X}_n - \mathbf{E}(\overline{X}_n)}{\sqrt{\mathbf{V}(\overline{X}_n)}} = \frac{\sqrt{n}\left(\overline{X}_n - \mathbf{E}(X_1)\right)}{\sqrt{\mathbf{V}(X_1)}} \quad \rightsquigarrow \quad Z \sim \text{Normal}\left(0, 1\right) , \tag{7.9}$$

$$\lim_{n\to\infty} P\left(\frac{\overline{X}_n - \mathbf{E}(\overline{X}_n)}{\sqrt{\mathbf{V}(\overline{X}_n)}} \leq z\right) = \lim_{n\to\infty} \mathbf{P}(Z_n \leq z) \quad = \quad \Phi(z) := \int_{-\infty}^z \left(\frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x^2}{2}\right)\right) dx . \tag{7.10}$$

Thus, for sufficiently large n *(say* $n > 30$*) we can make the following approximation:*

$$P\left(\frac{\overline{X}_n - \mathbf{E}(\overline{X}_n)}{\sqrt{\mathbf{V}(\overline{X}_n)}} \leq z\right) \approxeq \mathbf{P}(Z \leq z) = \Phi(z) := \int_{-\infty}^z \left(\frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x^2}{2}\right)\right) dx . \tag{7.11}$$

Proof: *See any intermediate to advanced undergraduate text in Probability. Start from the index looking for "Central Limit Theorem" to find the page number for the proof . . . .*

Heuristic Interpretation of CLT: *Probability statements about the sample mean RV* $\overline{X}_n$ *can be approximated using a Normal distribution.*

Here is a simulation showing CLT in action.

```
>> % a demonstration of Central Limit Theorem --
>> % the sample mean of a sequence of n IID Exponential(lambda) RVs
>> % itself a Gaussian(1/lambda,lambda/n) RV
>> lambda=0.1; Reps=10000; n=10; hist(sum(-1/lambda * log(rand(n,Reps)))/n)
>> lambda=0.1; Reps=10000; n=100; hist(sum(-1/lambda * log(rand(n,Reps)))/n,20)
>> lambda=0.1; Reps=10000; n=1000; hist(sum(-1/lambda * log(rand(n,Reps)))/n,20)
```

Let us look at an example that makes use of the CLT next.

Example 21 (Errors in computer code (Wasserman03, p. 78)) *Suppose the collection of RVs* $X_1, X_2, \ldots, X_n$ *model the number of errors in* n *computer programs named* $1, 2, \ldots, n$, *respectively. Suppose that the RV* X_i *modeling the number of errors in the* i*-th program is the Poisson*$(\lambda = 5)$ *for any* $i = 1, 2, \ldots, n$. *Further suppose that they are independently distributed. Succinctly, we suppose that*

$$X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} \text{Poisson}(\lambda = 5) .$$

Suppose we have $n = 125$ *programs and want to make a probability statement about* $\overline{X}_n$ *which is the average error per program out of these* 125 *programs. Since* $\mathbf{E}(X_i) = \lambda = 5$ *and* $\mathbf{V}(X_i) = \lambda = 5$,

we may want to know how often our sample mean $\overline{X}_{125}$ differs from the expectation of 5 errors per program. Using the CLT we can approximate $\mathbf{P}(\overline{X}_n < 5.5)$, for instance, as follows:

$$\begin{aligned}
\mathbf{P}(\overline{X}_n < 5.5) &= P\left(\frac{\sqrt{n}(\overline{X}_n - \mathbf{E}(X_1))}{\sqrt{\mathbf{V}(X_1)}} < \frac{\sqrt{n}(5.5 - \mathbf{E}(X_1))}{\sqrt{\mathbf{V}(X_1)}}\right) \\
&\approxeq P\left(Z < \frac{\sqrt{n}(5.5-\lambda)}{\sqrt{\lambda}}\right) \qquad \textit{[by (7.11), and } \mathbf{E}(X_1) = \mathbf{V}(X_1) = \lambda\textit{]} \\
&= P\left(Z < \frac{\sqrt{125}(5.5-5)}{\sqrt{5}}\right) \qquad \textit{[Since, } \lambda = 5 \textit{ and } n = 125 \textit{ in this Example]} \\
&= \mathbf{P}(Z \leq 2.5) = \Phi(2.5) = \int_{-\infty}^{2.5} \left(\frac{1}{\sqrt{2\pi}} \exp\left(\frac{-x^2}{2}\right)\right) dx \approxeq 0.993790334674224 \ .
\end{aligned}$$

The last number above needed the following:

Labwork 29 *The numerical approximation of $\Phi(2.5)$ was obtained via the following call to our* erf*-based* `NormalCdf` *function from 47.*

```
>> format long
>> disp(NormalCdf(2.5,0,1))
   0.993790334674224
```

The CLT says that if $X_1, X_2, \ldots \overset{\text{IID}}{\sim} X_1$, then $Z_n := \sqrt{n}(\overline{X}_n - \mathbf{E}(X_1))/\sqrt{\mathbf{V}(X_1)}$ is approximately distributed as Normal$(0,1)$. In Example 21, we knew $\sqrt{\mathbf{V}(X_1)}$. However, in general, we may not know $\sqrt{\mathbf{V}(X_1)}$. The next proposition says that we may estimate $\sqrt{\mathbf{V}(X_1)}$ using the sample standard deviation S_n of $X_1, X_2, \ldots, X_n$, according to (5.5), and still make probability statements about the sample mean $\overline{X}_n$ using a Normal distribution.

Proposition 11 (CLT based on Sample Variance) *Let $X_1, X_2, \ldots \overset{\text{IID}}{\sim} X_1$ and suppose $\mathbf{E}(X_1)$ and $\mathbf{V}(X_1)$ exists, then*

$$\frac{\sqrt{n}\left(\overline{X}_n - \mathbf{E}(X_1)\right)}{S_n} \rightsquigarrow \text{Normal}(0,1) \ . \tag{7.12}$$

We will use (7.12) for statistical estimation in the sequel.

Chapter 8

Fundamentals of Estimation

8.1 Introduction

Now that we have been introduced to two notions of convergence for RV sequences, we can begin to appreciate the basic limit theorems used in statistical inference. The problem of estimation is of fundamental importance in statistical inference and learning. We will formalise the general estimation problem here. There are two basic types of estimation. In point estimation we are interested in estimating a particular point of interest that is supposed to belong to a set of points. In (confidence) set estimation, we are interested in estimating a set with a particular form that has a specified probability of "trapping" the particular point of interest from a set of points. Here, a point should be interpreted as an element of a collection of elements from some space.

8.2 Point Estimation

Point estimation is any statistical methodology that provides one with a "**single best guess**" of some specific quantity of interest. Traditionally, we denote this **quantity of interest as** θ^* and **its point estimate as** $\widehat{\theta}$ **or** $\widehat{\theta}_n$. The subscript n in the point estimate $\widehat{\theta}_n$ emphasises that our estimate is based on n observations or data points from a given statistical experiment to estimate θ^*. This quantity of interest, which is usually unknown, can be:

- an **integral** $\vartheta^* := \int_A h(x)\,dx \in \boldsymbol{\Theta}$. If ϑ^* is finite, then $\boldsymbol{\Theta} = \mathbb{R}$, or
- a **parameter** θ^* which is an element of the **parameter space** $\boldsymbol{\Theta}$, denoted $\theta^* \in \boldsymbol{\Theta}$,
- a **distribution function (DF)** $F^* \in \mathbb{F}$:= the set of all DFs
- a **density function (pdf)** $f \in \{$"not too wiggly Sobolev functions"$\}$, or
- a **regression function** $g^* \in \mathbb{G}$, where $\mathbb{G}$ is a class of regression functions in a regression experiment with model: $Y = g^*(X) + \epsilon$, such that $\mathbf{E}(\epsilon) = 0$, from pairs of observations $\{(X_i, Y_i)\}_{i=1}^n$, or
- a **classifier** $g^* \in \mathbb{G}$, i.e. a regression experiment with discrete $Y = g^*(X) + \epsilon$, or
- a **prediction** in a regression experiment, i.e. when you want to estimate Y_i given X_i.

Recall that a statistic is an RV $T(X)$ that maps every data point x in the data space $\mathbb{X}$ with $T(x) = t$ in its range $\mathbb{T}$, i.e. $T(x) : \mathbb{X} \mapsto \mathbb{T}$ (Definition 25). Next, we look at a specific class of statistics whose range is the parameter space $\boldsymbol{\Theta}$.

Definition 39 *A* **point estimator** $\widehat{\Theta}$ *of some* **fixed and possibly unknown** $\theta^* \in \Theta$ *is a statistic that associates each data point* $x \in \mathbb{X}$ *with an estimate* $\widehat{\Theta}(x) = \widehat{\theta} \in \Theta$,

$$\boxed{\widehat{\Theta} := \widehat{\Theta}(x) = \widehat{\theta} : \mathbb{X} \mapsto \Theta} .$$

If our data point $x := (x_1, x_2, \ldots, x_n)$ *is an* n*-vector or a point in the* n*-dimensional real space, i.e.* $x := (x_1, x_2, \ldots, x_n) \in \mathbb{X}_n \subset \mathbb{R}^n$*, then we emphasise the dimension* n *in our point estimator* $\widehat{\Theta}_n$ *of* $\theta^* \in \Theta$.

$$\boxed{\widehat{\Theta}_n := \widehat{\Theta}_n(x := (x_1, x_2, \ldots, x_n)) = \widehat{\theta}_n : \mathbb{X}_n \mapsto \Theta, \quad \mathbb{X}_n \subset \mathbb{R}^n} .$$

The typical situation for us involves point estimation of $\theta^* \in \Theta$ *on the basis of one realisation* $x \in \mathbb{X}_n \subset \mathbb{R}^n$ *of an independent and identically distributed (IID) random vector* $X = (X_1, X_2, \ldots, X_n)$*, such that* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} X_1$ *and the DF of* X_1 *is* $F(x_1; \theta^*)$*, i.e. the distribution of the IID RVs,* $X_1, X_2, \ldots, X_n$*, is parameterised by* $\theta^* \in \Theta$.

Example 22 (Coin Tossing Experiment ($X_1, \ldots, X_n \overset{IID}{\sim} \text{Bernoulli}(\theta^*)$**))** *I tossed a coin that has an unknown probability* θ^* *of landing Heads independently and identically* 10 *times in a row. Four of my outcomes were Heads and the remaining six were Tails, with the actual sequence of Bernoulli outcomes (Heads* $\mapsto 1$ *and Tails* $\mapsto 0$*) being* $(1, 0, 0, 0, 1, 1, 0, 0, 1, 0)$*. I would like to estimate the probability* $\theta^* \in \Theta = [0, 1]$ *of observing Heads using the natural estimator* $\widehat{\Theta}_n((X_1, X_2, \ldots, X_n))$ *of* θ^**:*

$$\widehat{\Theta}_n((X_1, X_2, \ldots, X_n)) := \widehat{\Theta}_n = \frac{1}{n} \sum_{i=1}^{n} X_i =: \overline{X}_n$$

For the coin tossing experiment I just performed ($n = 10$ *times), the point estimate of the unknown* θ^* *is:*

$$\begin{aligned} \widehat{\theta}_{10} = \widehat{\Theta}_{10}((x_1, x_2, \ldots, x_{10})) &= \widehat{\Theta}_{10}((1, 0, 0, 0, 1, 1, 0, 0, 1, 0)) \\ &= \frac{1 + 0 + 0 + 0 + 1 + 1 + 0 + 0 + 1 + 0}{10} = \frac{4}{10} = 0.40 . \end{aligned}$$

Labwork 30 (Bernoulli(38/75) **Computer Experiment)** *Simulate one thousand IID samples from a* Bernoulli($\theta^* = 38/75$) *RV and store this data in an array called* `Samples`*. Use your student ID to initialise the fundamental sampler. Now, pretend that you don't know the true* θ^* *and estimate* θ^* *using our estimator* $\widehat{\Theta}_n = \overline{X}_n$ *from the data array* `Samples` *for each sample size* $n = 1, 2, \ldots, 1000$*. Plot the one thousand estimates* $\widehat{\theta}_1, \widehat{\theta}_2, \ldots, \widehat{\theta}_{1000}$ *as a function of the corresponding sample size. Report your observations regarding the behaviour of our estimator as the sample size increases.*

8.3 Some Properties of Point Estimators

Given that an estimator is merely a function from the data space to the parameter space, we need choose only the best estimators available. Recall that a point estimator $\widehat{\Theta}_n$, being a statistic or an RV of the data has a probability distribution over its range Θ. This distribution over Θ is called the **sampling distribution** of $\widehat{\Theta}_n$. Note that the sampling distribution not only depends on the statistic $\widehat{\Theta}_n := \widehat{\Theta}_n(X_1, X_2, \ldots, X_n)$ but also on θ^* which in turn determines the distribution of the IID data vector $(X_1, X_2, \ldots, X_n)$. The following definitions are useful for selecting better estimators from some lot of them.

Definition 40 (Bias of a Point Estimator) *The* bias_n *of an estimator* $\widehat{\Theta}_n$ *of* $\theta^* \in \mathbb{\Theta}$ *is:*

$$\boxed{\mathsf{bias}_n = \mathsf{bias}_n(\widehat{\Theta}_n) := \mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^* = \int_{\mathbb{X}_n} \widehat{\Theta}_n(x)\, dF(x;\theta^*) - \theta^*} \ . \tag{8.1}$$

We say that the estimator $\widehat{\Theta}_n$ *is* **unbiased** *if* $\mathsf{bias}_n(\widehat{\Theta}_n) = 0$ *or if* $\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) = \theta^*$ *for every* n*. If* $\lim_{n\to\infty} \mathsf{bias}_n(\widehat{\Theta}_n) = 0$*, we say that the estimator is* **asymptotically unbiased**.

Since the expectation of the sampling distribution of the point estimator $\widehat{\Theta}_n$ depends on the unknown θ^*, we emphasise the θ^*-dependence by $\mathbf{E}_{\theta^*}(\widehat{\Theta}_n)$.

Example 23 (Bias of our Estimator of θ^*) *Consider the sample mean estimator* $\widehat{\Theta}_n := \overline{X}_n$ *of* θ^**, from* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta^*)$*. That is, we take the sample mean of the* n *IID* Bernoulli(θ^*) *trials to be our point estimator of* $\theta^* \in [0,1]$*. Then,* **this estimator is unbiased** *since:*

$$\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) = \mathbf{E}_{\theta^*}\left(n^{-1}\sum_{i=1}^n X_i\right) = n^{-1}\mathbf{E}_{\theta^*}\left(\sum_{i=1}^n X_i\right) = n^{-1}\sum_{i=1}^n \mathbf{E}_{\theta^*}(X_i) = n^{-1}n\theta^* = \theta^* \ .$$

Definition 41 (Standard Error of a Point Estimator) *The standard deviation of the point estimator* $\widehat{\Theta}_n$ *of* $\theta^* \in \mathbb{\Theta}$ *is called the* **standard error***:*

$$\boxed{\mathsf{se}_n = \mathsf{se}_n(\widehat{\Theta}_n) = \sqrt{\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)} = \sqrt{\int_{\mathbb{X}_n} \left(\widehat{\Theta}_n(x) - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)^2 dF(x;\theta^*)}} \ . \tag{8.2}$$

Since the variance of the sampling distribution of the point estimator $\widehat{\Theta}_n$ depends on the fixed and possibly unknown θ^*, as emphasised by $\mathbf{V}_{\theta^*}$ in (8.2), the se_n is also a possibly unknown quantity and may itself be estimated from the data. The estimated standard error, denoted by $\widehat{\mathsf{se}}_n$, is calculated by replacing $\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)$ in (8.2) with its appropriate estimate.

Example 24 (Standard Error of our Estimator of θ^*) *Consider the sample mean estimator* $\widehat{\Theta}_n := \overline{X}_n$ *of* θ^**, from* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta^*)$*. Observe that the statistic:*

$$T_n((X_1, X_2, \ldots, X_n)) := n\,\widehat{\Theta}_n((X_1, X_2, \ldots, X_n)) = \sum_{i=1}^n X_i$$

is the Binomial(n, θ^*) *RV. The standard error* se_n *of this estimator is:*

$$\mathsf{se}_n = \sqrt{\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)} = \sqrt{\mathbf{V}_{\theta^*}\left(\sum_{i=1}^n \frac{X_i}{n}\right)} = \sqrt{\left(\sum_{i=1}^n \frac{1}{n^2}\mathbf{V}_{\theta^*}(X_i)\right)} = \sqrt{\frac{n}{n^2}\mathbf{V}_{\theta^*}(X_i)} = \sqrt{\theta^*(1-\theta^*)/n} \ .$$

Another reasonable property of an estimator is that it converge to the "true" parameter θ^* – here "true" means the supposedly fixed and possibly unknown θ^*, as we gather more and more and more IID data from a θ^*-specified DF $F(x;\theta^*)$. This property is stated precisely next.

Definition 42 (Asymptotic Consistency of a Point Estimator) *A point estimator* $\widehat{\Theta}_n$ *of* $\theta^* \in \mathbb{\Theta}$ *is said to be* **asymptotically consistent** *if:*

$$\boxed{\widehat{\Theta}_n \xrightarrow{P} \theta^*} \qquad \textit{i.e., for any real } \epsilon > 0, \quad \boxed{\lim_{n\to\infty} \mathbf{P}(|\widehat{\Theta}_n - \theta^*| > \epsilon) = 0} \ .$$

Definition 43 (Mean Squared Error (MSE) of a Point Estimator) *Often, the quality of a point estimator $\widehat{\Theta}_n$ of $\theta^* \in \Theta$ is assessed by the* **mean squared error** *or* MSE_n *defined by:*

$$\boxed{\mathsf{MSE}_n = \mathsf{MSE}_n(\widehat{\Theta}_n) := \mathbf{E}_{\theta^*}\left((\widehat{\Theta}_n - \theta^*)^2\right) = \int_{\mathbb{X}} (\widehat{\Theta}_n(x) - \theta^*)^2 \, dF(x;\theta^*)} \ . \tag{8.3}$$

The following proposition shows a simple relationship between the mean square error, bias and variance of an estimator $\widehat{\Theta}_n$ of θ^*.

Proposition 12 (The $\sqrt{\mathsf{MSE}_n} : \mathsf{se}_n : \mathsf{bias}_n$–Sided Right Triangle of an Estimator) *Let $\widehat{\Theta}_n$ be an estimator of $\theta^* \in \Theta$. Then:*

$$\boxed{\mathsf{MSE}_n(\widehat{\Theta}_n) = (\mathsf{se}_n(\widehat{\Theta}_n))^2 + (\mathsf{bias}_n(\widehat{\Theta}_n))^2} \ . \tag{8.4}$$

Proof:

$$\begin{aligned}
& LHS \\
= \ & \mathsf{MSE}_n(\widehat{\Theta}_n) \\
:= \ & \mathbf{E}_{\theta^*}\left((\widehat{\Theta}_n - \theta^*)^2\right), \qquad \textit{by definition of } \mathsf{MSE}_n \ (8.3) \\
= \ & \mathbf{E}_{\theta^*}\left(\left(\underbrace{\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)}_{A} + \underbrace{\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)}_{B}\right)^2\right), \qquad \textit{by subtracting and adding the constant } \mathbf{E}_{\theta^*}(\widehat{\Theta}_n) \\
= \ & \mathbf{E}_{\theta^*}\left(\underbrace{\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)^2}_{A^2} + \underbrace{2\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)}_{2AB} + \underbrace{\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)^2}_{B^2}\right), \qquad \because \ (A+B)^2 = A^2 + 2AB + B^2 \\
= \ & \mathbf{E}_{\theta^*}\left(\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)^2\right) + \mathbf{E}_{\theta^*}\left(2\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)\right) + \mathbf{E}_{\theta^*}\left(\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)^2\right), \\
= \ & \mathbf{E}_{\theta^*}\left(\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)^2\right) + \underbrace{2\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)}_{C}\underbrace{\mathbf{E}_{\theta^*}\left(\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)\right)}_{D} + \mathbf{E}_{\theta^*}\left(\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)^2\right), \quad \because \ C \textit{ is constant} \\
= \ & \mathbf{E}_{\theta^*}\left(\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)^2\right) + 0 + \mathbf{E}_{\theta^*}\left(\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)^2\right), \qquad \because \ D := \mathbf{E}_{\theta^*}\left(\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)\right) = \mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n) = 0 \\
= \ & \mathbf{V}_{\theta^*}(\widehat{\Theta}_n) + \mathbf{E}_{\theta^*}\left(\left(\mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*\right)^2\right), \qquad \because \ \mathbf{V}_{\theta^*}(\widehat{\Theta}_n) := \mathbf{E}_{\theta^*}\left(\left(\widehat{\Theta}_n - \mathbf{E}_{\theta^*}(\widehat{\Theta}_n)\right)^2\right), \textit{ by definition of variance} \\
= \ & \left(\sqrt{\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)}\right)^2 + \mathbf{E}_{\theta^*}\left(\left(\mathsf{bias}_n(\widehat{\Theta}_n)\right)^2\right), \qquad \because \ \mathsf{bias}_n(\widehat{\Theta}_n) = \mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^*, \textit{ by definition of } \mathsf{bias}_n \textit{ of an estimator } \widehat{\Theta}_n \\
= \ & \left(\mathsf{se}_n(\widehat{\Theta}_n)\right)^2 + \mathbf{E}_{\theta^*}\left(\left(\mathsf{bias}_n(\widehat{\Theta}_n)\right)^2\right), \qquad \because \ \mathsf{se}_n(\widehat{\Theta}_n) := \sqrt{\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)}, \textit{ by definition } (8.2) \\
= \ & \left(\mathsf{se}_n(\widehat{\Theta}_n)\right)^2 + \left(\mathsf{bias}_n(\widehat{\Theta}_n)\right)^2, \qquad \because \ \mathsf{bias}_n(\widehat{\Theta}_n) = \mathbf{E}_{\theta^*}(\widehat{\Theta}_n) - \theta^* \textit{ and } \left(\mathsf{bias}_n(\widehat{\Theta}_n)\right)^2 \textit{ are constants.} \\
= \ & RHS
\end{aligned}$$

Proposition 13 *Let $\widehat{\Theta}_n$ be an estimator of $\theta^* \in \Theta$. Then, if $\mathsf{bias}_n(\widehat{\Theta}_n) \to 0$ and $\mathsf{se}_n(\widehat{\Theta}_n) \to 0$ as $n \to \infty$, the estimator $\widehat{\Theta}_n$ is asymptotically consistent:*

$$\widehat{\Theta}_n \xrightarrow{P} \theta^* \ .$$

Proof: *If $\mathsf{bias}_n(\widehat{\Theta}_n) \to 0$ and $\mathsf{se}_n(\widehat{\Theta}_n) \to 0$, then by (8.4), $\mathsf{MSE}_n(\widehat{\Theta}_n) \to 0$, i.e. that $\mathbf{E}_{\theta^*}\left((\widehat{\Theta}_n - \theta^*)^2\right) \to 0$. This type of convergence of the RV $\widehat{\Theta}_n$ to the Point Mass(θ^*) RV as $n \to \infty$ is called convergence in* **quadratic mean** *or* **convergence in $\mathbf{L_2}$** *and denoted by $\widehat{\Theta}_n \xrightarrow{qm} \theta^*$. Convergence in quadratic mean is a stronger notion of convergence than convergence in probability, in the sense that*

$$\mathbf{E}_{\theta^*}\left((\widehat{\Theta}_n - \theta^*)^2\right) \to 0 \quad \textit{or} \quad \widehat{\Theta}_n \xrightarrow{qm} \theta^* \implies \widehat{\Theta}_n \xrightarrow{P} \theta^* \ .$$

Thus, if we prove the above implication we are done with the proof of our proposition. To show that convergence in quadratic mean implies convergence in probability for general sequence of RVs X_n converging to an RV X, we first assume that $X_n \xrightarrow{qm} X$. Now, fix any $\epsilon > 0$. Then by Markov's inequality (7.1),

$$\mathbf{P}(|X_n - X| > \epsilon) = \mathbf{P}(|X_n - X|^2 > \epsilon^2) \leq \frac{\mathbf{E}(|X_n - X|^2)}{\epsilon^2} \to 0 \ ,$$

and we have shown that the definition of convergence in probability holds provided convergence in quadratic mean holds.

We want our estimator to be unbiased with small standard errors as the sample size n gets large. The **point estimator** $\widehat{\Theta}_n$ will then produce a **point estimate** $\widehat{\theta}_n$:

$$\widehat{\Theta}_n((x_1, x_2, \ldots, x_n)) = \widehat{\theta}_n \in \mathbf{\Theta} \ ,$$

on the basis of the **observed data** $(x_1, x_2, \ldots, x_n)$, that is close to the **true parameter** $\theta^* \in \mathbf{\Theta}$.

Example 25 (Asymptotic consistency of our Estimator of θ^*) *Consider the sample mean estimator $\widehat{\Theta}_n := \overline{X}_n$ of θ^*, from $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta^*)$. Since $\mathsf{bias}_n(\widehat{\Theta}_n) = 0$ for any n and $\mathsf{se}_n = \sqrt{\theta^*(1-\theta^*)/n} \to 0$, as $n \to \infty$, by Proposition 13, $\widehat{\Theta}_n \xrightarrow{P} \theta^*$. That is $\widehat{\Theta}_n$ is an* **asymptotically consistent estimator** *of θ^*.*

8.4 Confidence Set Estimation

As we saw in Section 8.2, the point estimate $\widehat{\theta}_n$ is a "single best guess" of the fixed and possibly unknown parameter $\theta^* \in \mathbf{\Theta}$. However, if we wanted to make a statement about our confidence in an estimation procedure, then one possibility is to produce subsets from the parameter space $\mathbf{\Theta}$ called **confidence sets** that "engulf" θ^* with a probability of at least $1 - \alpha$.

Formally, an $\mathbf{1-\alpha}$ **confidence interval** for the parameter $\theta^* \in \mathbf{\Theta} \subset \mathbb{R}$, based on n observations or data points $X_1, X_2, \ldots, X_n$, is an interval C_n that is a function of the data:

$$C_n := [\underline{C}_n, \overline{C}_n] = [\underline{C}_n(X_1, X_2, \ldots, X_n), \overline{C}_n(X_1, X_2, \ldots, X_n)] \ ,$$

such that:

$$\mathbf{P}_{\theta^*}\left(\theta^* \in C_n := [\underline{C}_n, \overline{C}_n]\right) \geq 1 - \alpha \ .$$

Note that the confidence interval $C_n := [\underline{C}_n, \overline{C}_n]$ is a two-dimensional RV or a random vector in $\mathbb{R}^2$ that depends on the two statistics $\underline{C}_n(X_1, X_2, \ldots, X_n)$ and $\overline{C}_n(X_1, X_2, \ldots, X_n)$, as well as θ^*, which in turn determines the distribution of the data $(X_1, X_2, \ldots, X_n)$. In words, C_n engulfs the true parameter $\theta^* \in \mathbf{\Theta}$ with a probability of at least $1 - \alpha$. We call $1 - \alpha$ as the **coverage** of the confidence interval C_n.

Formally, a $1-\alpha$ **confidence set** C_n for a vector-valued $\theta^* \in \mathbf{\Theta} \subset \mathbb{R}^k$ is any subset of $\mathbf{\Theta}$ such that $\mathbf{P}_{\theta^*}(\theta^* \in C_n) \geq 1 - \alpha$. The typical forms taken by C_n are k-dimensional boxes or hyper-cuboids, hyper-ellipsoids and subsets defined by inequalities involving level sets of some estimator of θ^*.

Typically, we take $\alpha = 0.05$ because we are interested in the $1 - \alpha = 0.95$ or 95% confidence interval/set $C_n \subset \mathbf{\Theta}$ of $\theta^* \in \mathbf{\Theta}$ from an estimator $\widehat{\Theta}_n$ of θ^*.

The following property of an estimator makes it easy to obtain confidence intervals.

Definition 44 (Asymptotic Normality of Estimators) *An estimator $\widehat{\Theta}_n$ of a fixed and possibly unknown parameter $\theta^* \in \mathbf{\Theta}$ is* **asymptotically normal** *if:*

$$\frac{\widehat{\Theta}_n - \theta^*}{\mathsf{se}_n} \rightsquigarrow \text{Normal}(0, 1) \ . \tag{8.5}$$

That is, $\widehat{\Theta}_n \rightsquigarrow \text{Normal}(\theta^*, \mathsf{se}_n^2)$. By a further estimation of $\mathsf{se}_n := \sqrt{\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)}$ by $\widehat{\mathsf{se}}_n$, we can see that $\widehat{\Theta}_n \rightsquigarrow \text{Normal}(\theta^*, \widehat{\mathsf{se}}_n^2)$ on the basis of (7.12).

Proposition 14 (Normal-based Asymptotic Confidence Interval) *Suppose an estimator $\widehat{\Theta}_n$ of parameter $\theta^* \in \mathbf{\Theta} \subset \mathbb{R}$ is asymptotically normal:*

$$\widehat{\Theta}_n \rightsquigarrow \text{Normal}(\theta^*, \widehat{\mathsf{se}}_n^2) \ .$$

Let the RV $Z \sim \text{Normal}(0,1)$ have DF Φ and inverse DF Φ^{-1}. Let:

$$z_{\alpha/2} = \Phi^{-1}(1-(\alpha/2)), \quad \textit{that is,} \quad \mathbf{P}(Z > z_{\alpha/2}) = \alpha/2 \ \textit{ and } \ \mathbf{P}(-z_{\alpha/2} < Z < z_{\alpha/2}) = 1-\alpha \ .$$

Then:

$$\mathbf{P}_{\theta^*}(\theta^* \in C_n) = \mathbf{P}\left(\theta^* \in [\widehat{\Theta}_n - z_{\alpha/2}\widehat{\mathsf{se}}_n, \widehat{\Theta}_n + z_{\alpha/2}\widehat{\mathsf{se}}_n]\right) \to 1-\alpha \ .$$

Therefore:

$$C_n := [\underline{C}_n, \overline{C}_n] = [\widehat{\Theta}_n - z_{\alpha/2}\widehat{\mathsf{se}}_n, \widehat{\Theta}_n + z_{\alpha/2}\widehat{\mathsf{se}}_n]$$

is the $1-\alpha$ Normal-based asymptotic confidence interval that relies on the asymptotic normality of the estimator $\widehat{\Theta}_n$ of $\theta^* \in \mathbf{\Theta} \subset \mathbb{R}$.

Proof: *Define the centralised and scaled estimator as $Z_n := (\widehat{\Theta}_n - \theta^*)/\widehat{\mathsf{se}}_n$. By assumption, $Z_n \rightsquigarrow Z \sim \text{Normal}(0,1)$. Therefore,*

$$\begin{aligned}
\mathbf{P}_{\theta^*}(\theta^* \in C_n) &= \mathbf{P}_{\theta^*}\left(\theta^* \in [\widehat{\Theta}_n - z_{\alpha/2}\widehat{\mathsf{se}}_n, \widehat{\Theta}_n + z_{\alpha/2}\widehat{\mathsf{se}}_n]\right) \\
&= \mathbf{P}_{\theta^*}\left(\widehat{\Theta}_n - z_{\alpha/2}\widehat{\mathsf{se}}_n \le \theta^* \le \widehat{\Theta}_n + z_{\alpha/2}\widehat{\mathsf{se}}_n\right) \\
&= \mathbf{P}_{\theta^*}\left(-z_{\alpha/2}\widehat{\mathsf{se}}_n \le \widehat{\Theta}_n - \theta^* \le z_{\alpha/2}\widehat{\mathsf{se}}_n\right) \\
&= \mathbf{P}_{\theta^*}\left(-z_{\alpha/2} \le \frac{\widehat{\Theta}_n - \theta^*}{\widehat{\mathsf{se}}_n} \le z_{\alpha/2}\right) \\
&\to \mathbf{P}_{\theta^*}\left(-z_{\alpha/2} \le Z \le z_{\alpha/2}\right) \\
&= 1-\alpha
\end{aligned}$$

Figure 8.1: Density and Confidence Interval of the Asymptotically Normal Point Estimator

For 95% confidence intervals, $\alpha = 0.05$ and $z_{\alpha/2} = z_{0.025} = 1.96 \approxeq 2$. This leads to the **approximate 95% confidence interval** *of $\widehat{\theta}_n \pm 2\widehat{\mathsf{se}}_n$, where $\widehat{\theta}_n = \widehat{\Theta}_n(x_1, x_2, \ldots, x_n)$ and $x_1, x_2, \ldots, x_n$ are the data or observations of the RVs $X_1, X_2, \ldots, X_n$.*

Example 26 (Confidence interval for θ^* from n Bernoulli(θ^*) trials) *Let $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} \text{Bernoulli}(\theta^*)$ for some fixed but unknown parameter $\theta^* \in \mathbf{\Theta} = [0,1]$. Consider the following point estimator of θ^*:*

$$\widehat{\Theta}_n((X_1, X_2, \ldots, X_n)) = n^{-1}\sum_{i=1}^{n} X_i \ .$$

That is, we take the sample mean of the n IID Bernoulli(θ^*) *trials to be our point estimator of* $\theta^* \in [0,1]$. *Then, we already saw that* **this estimator is unbiased**

We already saw that the standard error se_n *of this estimator is:*

$$\mathsf{se}_n = \sqrt{\theta^*(1-\theta^*)/n} \ .$$

Since θ^* *is unknown, we obtain the estimated standard error* $\widehat{\mathsf{se}}_n$ *from the point estimate* $\widehat{\theta}_n$ *of* θ^* *on the basis of* n *observed data points* $x = (x_1, x_2, \ldots, x_n)$ *of the experiment:*

$$\widehat{\mathsf{se}}_n = \sqrt{\widehat{\theta}_n(1-\widehat{\theta}_n)/n}, \quad \textit{where,} \quad \widehat{\theta}_n = \widehat{\Theta}_n((x_1, x_2, \ldots, x_n)) = n^{-1}\sum_{i=1}^{n} x_i \ .$$

By the central limit theorem, $\widehat{\Theta}_n \rightsquigarrow \text{Normal}(\theta^*, \widehat{\mathsf{se}}_n)$, *i.e.* $\widehat{\Theta}_n$ *is asymptotically normal. Therefore, an asymptotically (for large sample size* n*) approximate* $1-\alpha$ *normal-based confidence interval is:*

$$\widehat{\theta}_n \pm z_{\alpha/2}\widehat{\mathsf{se}}_n = \widehat{\theta}_n \pm z_{\alpha/2}\sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}} := \left[\widehat{\theta}_n - z_{\alpha/2}\sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}} \ , \ \widehat{\theta}_n + z_{\alpha/2}\sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}}\right]$$

We also saw that $\widehat{\Theta}_n$ *is an* **asymptotically consistent estimator** *of* θ^* *due to Proposition 13.*

The confidence Interval for the coin tossing experiment in Example 28 with the observed sequence of Bernoulli outcomes (Heads $\mapsto 1$ *and Tails* $\mapsto 0$*) being* $(1,0,0,0,1,1,0,0,1,0)$. *We estimated the probability* θ^* *of observing Heads with the* **unbiased, asymptotically consistent estimator** $\widehat{\Theta}_n((X_1, X_2, \ldots, X_n)) = n^{-1}\sum_{i=1}^{n} X_i$ *of* θ^*. *The point estimate of* θ^* *was:*

$$\begin{aligned} \widehat{\theta}_{10} = \widehat{\Theta}_{10}((x_1, x_2, \ldots, x_{10})) &= \widehat{\Theta}_{10}((1,0,0,0,1,1,0,0,1,0)) \\ &= \frac{1+0+0+0+1+1+0+0+1+0}{10} = \frac{4}{10} = 0.40 \ . \end{aligned}$$

The normal-based confidence interval for θ^* *may not be a valid approximation here with just* $n = 10$ *samples. Nevertheless, we will compute a* 95% *normal-based confidence interval:*

$$C_{10} = 0.40 \pm 1.96\sqrt{\frac{0.40(1-0.40)}{10}} = 0.40 \pm 0.3036 = [0.0964, 0.7036]$$

with a width of 0.6072. *When I increased the sample size* n *of the experiment from* 10 *to* 100 *by tossing the same coin another* 90 *times, I discovered that a total of* 57 *trials landed as Heads. Thus my point estimate and confidence interval for* θ^* *are:*

$$\widehat{\theta}_{100} = \frac{57}{100} = 0.57 \qquad \textit{and} \qquad C_{100} = 0.57 \pm 1.96\sqrt{\frac{0.57(1-0.57)}{100}} = 0.57 \pm 0.0495 = [0.5205, 0.6195]$$

with a much smaller width of 0.0990. *Thus our confidence interval shrank considerably from a width of* 0.6072 *after an additional* 90 *Bernoulli trials. Thus, we can make the width of the confidence interval as small as we want by making the number of observations or sample size* n *as large as we can.*

8.5 Likelihood

We take a look at one of the most fundamental concepts in Statistics.

Definition 45 (Likelihood Function) *Suppose $X_1, X_2, \ldots, X_n$ have joint density $f(x_1, x_2, \ldots, x_n; \theta)$ specified by parameter $\theta \in \Theta$. Let the observed data be $x_1, x_2, \ldots, x_n$.*

The **likeihood** *function given by $L_n(\theta)$ is proportional to $f(x_1, x_2, \ldots, x_n; \theta)$, the joint probability of the data, with the exception that we see it as a function of the parameter:*

$$L_n(\theta) := L_n(x_1, x_2, \ldots, x_n; \theta) = f(x_1, x_2, \ldots, x_n; \theta) \ . \tag{8.6}$$

The likelihood function has a simple product structure when the observations are independently and identically distributed:

$$X_1, X_2, \ldots, X_n \overset{IID}{\sim} f(x; \theta) \implies \boxed{L_n(\theta) := L_n(x_1, x_2, \ldots, x_n; \theta) = f(x_1, x_2, \ldots, x_n; \theta) := \prod_{i=1}^{n} f(x_i; \theta)} \ . \tag{8.7}$$

The **log-likelihood** *function is defined by:*

$$\boxed{\ell_n(\theta) := \log(L_n(\theta))} \tag{8.8}$$

Example 27 (Likelihood of the IID Bernoulli(θ^*) **experiment)** *Consider our IID Bernoulli experiment:*

$$X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{Bernoulli}(\theta^*), \text{ with PDF } f(x; \theta) = \theta^x (1-\theta)^{1-x} \mathbb{1}_{\{0,1\}}(x) \ .$$

Let us understand the likelihood function for one observation first. There are two possibilities for the first observation.

If we only have one observation and it happens to be $x_1 = 1$, then our likelihood function is:

$$L_1(\theta) = L_1(x_1; \theta) = f(x_1; \theta) = \theta^1 (1-\theta)^{1-1} \mathbb{1}_{\{0,1\}}(1) = \theta (1-\theta)^0 1 = \theta$$

If we only have one observation and it happens to be $x_1 = 0$, then our likelihood function is:

$$L_1(\theta) = L_1(x_1; \theta) = f(x_1; \theta) = \theta^0 (1-\theta)^{1-0} \mathbb{1}_{\{0,1\}}(0) = 1 (1-\theta)^1 1 = 1 - \theta$$

If we have n observations $(x_1, x_2, \ldots, x_n)$, i.e. a vertex point of the unit hyper-cube $\{0,1\}^n$, then our likelihood function is obtained by multiplying the densities:

$$\begin{aligned} L_n(\theta) &:= L_n(x_1, x_2, \ldots, x_n; \theta) = f(x_1, x_2, \ldots, x_n; \theta) \\ &= f(x_1; \theta) f(x_2; \theta) \cdots f(x_n; \theta) := \prod_{i=1}^{n} f(x_i; \theta) \\ &= \theta^{\sum_{i=1}^n x_i} (1-\theta)^{n - \sum_{i=1}^n x_i} := \theta^{t_n} (1-\theta)^{n - t_n} \end{aligned}$$

In the last step, we have formally defined the following statistic of the data:

$$T_n(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} X_i : \mathbb{X}_n \to \mathbb{T}_n$$

with the corresponding realisation $t_n := T_n(x_1, x_2, \ldots, x_n) = \sum_{i=1}^n x_i \in \mathbb{T}_n$.

Figure 8.2: Data Spaces $\mathbb{X}_1 = \{0,1\}$, $\mathbb{X}_2 = \{0,1\}^2$ and $\mathbb{X}_3 = \{0,1\}^3$ for one, two and three IID Bernoulli trials, respectively and the corresponding likelihood functions.

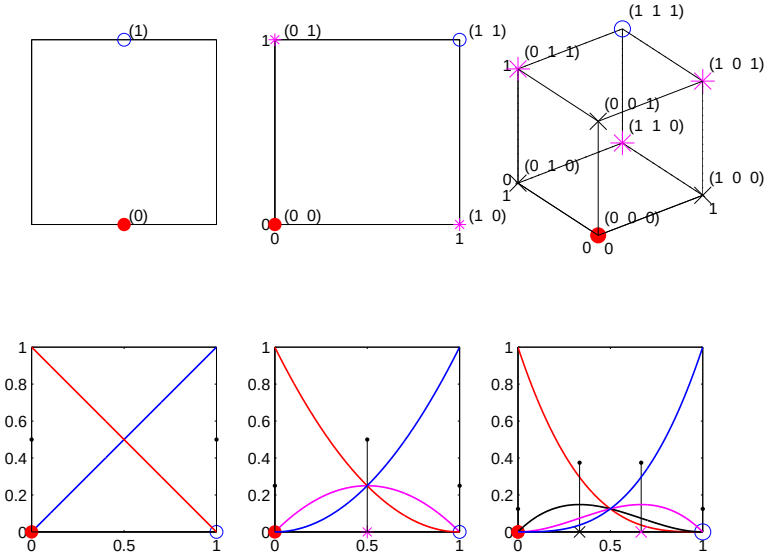
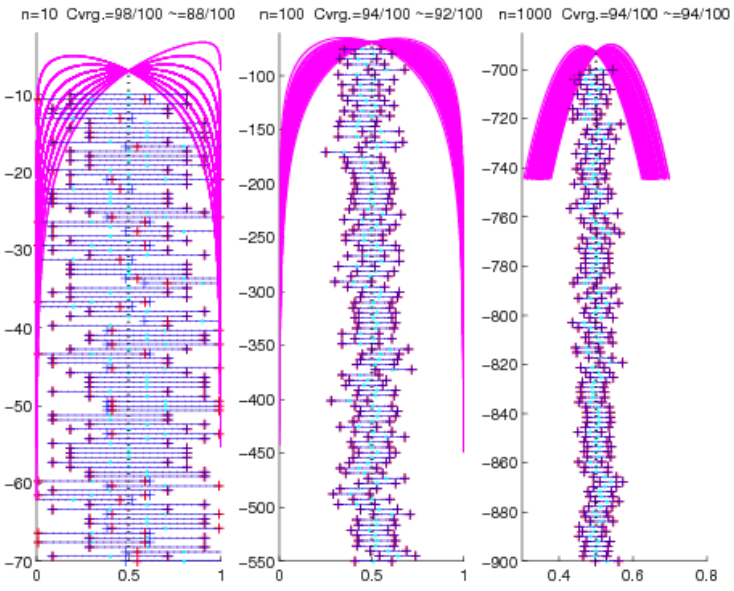


Figure 8.3: 100 realisations of $C_{10}, C_{100}, C_{1000}$ based on samples of size $n = 10$, 100 and 1000 drawn from the Bernoulli($\theta^* = 0.5$) RV as per Labwork 54. The MLE $\widehat{\theta}_n$ (cyan dot) and the log-likelihood function (magenta curve) for each of the 100 replications of the experiment for each sample size n are depicted. The approximate normal-based 95% confidence intervals with blue boundaries are based on the exact $\mathsf{se}_n = \sqrt{\theta^*(1-\theta^*)/n} = \sqrt{1/4}$, while those with red boundaries are based on the estimated $\widehat{\mathsf{se}}_n = \sqrt{\widehat{\theta}_n(1-\widehat{\theta}_n)/n}$. The fraction of times the true parameter $\theta^* = 0.5$ was engulfed by the exact and approximate confidence interval (empirical coverage) over the 100 replications of the experiment for each of the three sample sizes are given by the numbers after `Cvrg.=` and `~=`, above each sub-plot, respectively.



Chapter 9

Maximum Likelihood Estimator

Next we look at a specific point estimator called the maximum likelihood estimator (MLE) of a possibly unknown but fixed parameter θ^* in a parametric experiment, i.e. $\theta^* \in \Theta \subset \mathbb{R}^k$ with $k < \infty$. Other point estimators in such a setting include the moment estimator (MME).

Recall that the likelihood function (See Definition 45) for an IID experiment with observations $x_1, x_2, \ldots, x_n$ is simply the product of the densities:

$$L_n(\theta) = \prod_{i=1}^{n} f(x_i; \theta) : \Theta \mapsto (0, \infty) \ ,$$

and its logarithm or log-likelihood function is:

$$\ell_n(\theta) = \log(L_n(\theta)) = \sum_{i=1}^{n} \log(f(x_i)) : \Theta \mapsto (-\infty, \infty) \ .$$

9.1 Introduction to Maximum Likelihood Estimation

Definition 46 (Maximum Likelihood Estimator (MLE)) *Let $X_1, \ldots, X_n \sim f(x_1, \ldots, x_n; \theta^*)$. The maximum likelihood estimator (MLE) $\widehat{\Theta}_n$ of the fixed and possibly unknown parameter $\theta^* \in \Theta$ is the value of θ that maximises the likelihood function:*

$$\boxed{\widehat{\Theta}_n := \widehat{\Theta}_n(X_1, X_2, \ldots, X_n) := \arg\max_{\theta \in \Theta} L_n(\theta) \ ,}$$

Equivalently, MLE is the value of θ that maximises the log-likelihood function:

$$\boxed{\widehat{\Theta}_n := \arg\max_{\theta \in \Theta} \ell_n(\theta) \ ,}$$

since the maximum of the likelihood coincides with that of the log-likelihood. It is analytically and numerically convenient to work with the log-likelihood instead of the likelihood. Optimisation algorithms can be used to find the MLE numerically. Such algorithms by convention tend to find the minimum and the value that minimises a function. So, the MLE is also the the value of θ that minimises the negative likelihood or negative log-likelihood functions:

$$\boxed{\widehat{\Theta}_n := \arg\min_{\theta \in \Theta} -L_n(\theta), \qquad \widehat{\Theta}_n := \arg\min_{\theta \in \Theta} -\ell_n(\theta) \ .}$$

Once again, the realisation of the MLE, namely $\widehat{\theta}_n = \widehat{\Theta_n}(x_1, \ldots, x_n)$ based on the observation is the maximum likelihood estimate (MLe) of the θ^*.

Example 28 (Coin Tossing Experiment ($X_1, \ldots, X_n \overset{IID}{\sim} \text{Bernoulli}(\theta^*)$)) *I tossed a coin that has an unknown probability θ^* of landing Heads independently and identically 10 times in a row. Four of my outcomes were Heads and the remaining six were Tails, with the actual sequence of Bernoulli outcomes (Heads $\mapsto$ 1 and Tails $\mapsto$ 0) being $(1,0,0,0,1,1,0,0,1,0)$. I would like to estimate the probability $\theta^* \in \Theta = [0,1]$ of observing Heads using the maximum likelihood estimator or MLE $\widehat{\Theta}_n((X_1, X_2, \ldots, X_n))$ of θ. We derive the MLE next.*

First, the likelihood function is:

$$L_n(\theta) \;:=\; L_n(x_1, x_2, \ldots, x_n; \theta) = \prod_{i=1}^{n} f(x_i|\theta) = \theta^{\sum_{i=1}^n x_i}(1-\theta)^{n-\sum_{i=1}^n x_i} := \theta^{t_n}(1-\theta)^{n-t_n}$$

In the last step, we have formally defined the following statistic of the data:

$$T_n(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} X_i : \mathbb{X}_n \to \mathbb{T}_n$$

with the corresponding realisation $t_n := T_n(x_1, x_2, \ldots, x_n) = \sum_{i=1}^n x_i \in \mathbb{T}_n$. Let us now take the natural logarithm of both sides:

$$\log(L_n(\theta)) := \log(L(x_1, x_2, \ldots, x_n; \theta)) = \log\left(\theta^{t_n}(1-\theta)^{n-t_n}\right) = t_n \log(\theta) + (n - t_n)\log(1-\theta)$$

Next, we take the derivative with respect to the parameter θ:

$$\begin{aligned} \frac{\partial}{\partial\theta}\log(L_n(\theta)) &= \frac{\partial}{\partial\theta} t_n \log(\theta) + \frac{\partial}{\partial\theta}(n - t_n)\log(1-\theta) \\ &= \frac{t_n}{\theta} - \frac{n - t_n}{1-\theta} \end{aligned}$$

Now, set $\frac{\partial}{\partial\theta}\log(L_n(\theta)) = 0$ and solve for θ to obtain the maximum likelihood estimate $\widehat{\theta}_n$:

$$\frac{\partial}{\partial\theta}\log(L(\theta)) = 0 \iff \frac{t_n}{\theta} = \frac{n-t_n}{1-\theta} \iff \frac{1-\theta}{\theta} = \frac{n-t_n}{t_n} \iff \frac{1}{\theta} - 1 = \frac{n}{t_n} - 1 \iff \widehat{\theta}_n = \frac{t_n}{n}$$

Therefore the MLE is:

$$\widehat{\Theta}_n(X_1, X_2, \ldots, X_n) = \frac{1}{n} T_n(X_1, X_2, \ldots, X_n) = \frac{1}{n}\sum_{i=1}^{n} X_i = \overline{X}_n$$

For the coin tossing experiment I just performed ($n = 10$ times), the point estimate of θ is:

$$\begin{aligned} \widehat{\theta}_{10} = \widehat{\Theta}_{10}((x_1, x_2, \ldots, x_{10})) &= \widehat{\Theta}_{10}((1,0,0,0,1,1,0,0,1,0)) \\ &= \frac{1+0+0+0+1+1+0+0+1+0}{10} = \frac{4}{10} = 0.40\ . \end{aligned}$$

9.2 Practical Excursion in One-dimensional Optimisation

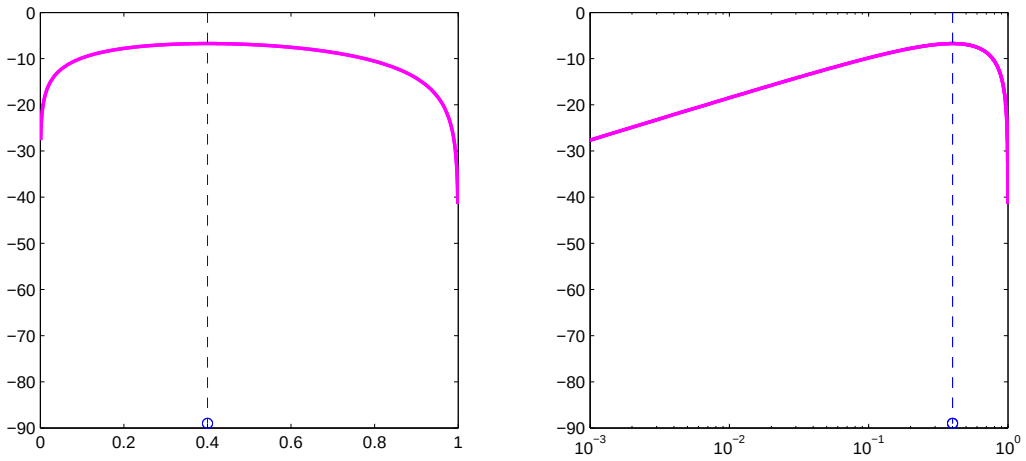
Numerically maximising a log-likelihood function of one parameter is a useful technique. This can be used for models with no analytically known MLE. A fairly large field of maths, called optimisation, exists for this sole purpose. Conventionally, in optimisation, one is interested in minimisation. Therefore, the basic algorithms are cast in the "find the minimiser and the minimum" of a target function $f : \mathbb{R} \mapsto \mathbb{R}$. Since we are interested in maximising our target, which is the likelihood

or log-likelihood function, say $\log(L(x_1, \ldots, x_n; \theta)) : \Theta \mapsto \mathbb{R}$, we will simply apply the standard optimisation algorithms directly to $-\log(L(x_1, \ldots, x_n; \theta)) : \Theta \mapsto \mathbb{R}$.

The algorithm implemented in `fminbnd` is based on the golden section search and an inverse parabolic interpolation, and attempts to find the minimum of a function of one variable within a given fixed interval. Briefly, the golden section search proceeds by successively **bracketing** the minimum of the target function within an acceptably small interval inside the given starting interval [see Section 8.2 of Forsythe, G. E., M. A. Malcolm, and C. B. Moler, 1977, *Computer Methods for Mathematical Computations*, Prentice-Hall]. MATLAB's `fminbnd` also relies on Brent's inverse parabolic interpolation [see Chapter 5 of Brent, Richard. P., 1973, *Algorithms for Minimization without Derivatives*, Prentice-Hall, Englewood Cliffs, New Jersey]. Briefly, additional smoothness conditions are assumed for the target function to aid in a faster bracketing strategy through polynomial interpolations of past function evaluations. MATLAB's `fminbnd` has several limitations, including:

- The likelihood function must be continuous.
- Only local MLE solutions, i.e. those inside the starting interval, are given.
- One needs to know or carefully guess the starting interval that contains the MLE.
- MATLAB's `fminbnd` exhibits slow convergence when the solution is on a boundary of the starting interval.

Figure 9.1: Plot of $\log(L(1,0,0,0,1,1,0,0,1,0;\theta))$ as a function of the parameter θ over the parameter space $\Theta = [0,1]$ and the MLE $\widehat{\theta}_{10}$ of 0.4 for the coin-tossing experiment.



Labwork 31 (Coin-tossing experiment) *The following script was used to study the coin-tossing experiment in* MATLAB. *The plot of the log-likelihood function and the numerical optimisation of MLE are carried out using* MATLAB*'s built-in function* `fminbnd` *(See Figure 9.1).*

BernoulliMLE.m

```
% To simulate n coin tosses, set theta=probability of heads and n
% Then draw n IID samples from Bernoulli(theta) RV
% theta=0.5; n=20; x=floor(rand(1,n) + theta);
% enter data from a real coin tossing experiment
x=[1 0 0 0 1 1 0 0 1 0]; n=length(x);
```

```
t = sum(x); % statistic t is the sum of the x_i values
% display the outcomes and their sum
display(x)
display(t)

% Analyticaly MLE is t/n
MLE=t/n
% l is the log-likelihood of data x as a function of parameter theta
l=@(theta)log(theta ^ t * (1-theta)^(n-t));
ThetaS=[0:0.001:1]; % sample some values for theta

% plot the log-likelihood function and MLE in two scales
subplot(1,2,1);
plot(ThetaS,arrayfun(l,ThetaS),'m','LineWidth',2);
hold on; stem([MLE],[-89],'b--'); % plot MLE as a stem plot
subplot(1,2,2);
semilogx(ThetaS,arrayfun(l,ThetaS),'m','LineWidth',2);
hold on; stem([MLE],[-89],'b--'); % plot MLE as a stem plot

% Now we will find the MLE by finding the minimiser or argmin of -l
% negative log-likelihood function of parameter theta
negl=@(theta)-(log(theta ^ t * (1-theta)^(n-t)));
% read help fminbnd
% you need to supply the function to be minimised and its search interval
% NumericalMLE = fminbnd(negl,0,1)
% to see the iteration in the numerical minimisation
 NumericalMLE = fminbnd(negl,0,1,optimset('Display','iter'))
```

```
>> BernoulliMLE
x =     1     0     0     0     1     1     0     0     1     0
t =     4
MLE =    0.4000
Func-count     x          f(x)         Procedure
    1       0.381966      6.73697        initial
    2       0.618034      7.69939        golden
    3       0.236068       7.3902        golden
    4       0.408979      6.73179        parabolic
    5       0.399339      6.73013        parabolic
    6       0.400045      6.73012        parabolic
    7       0.400001      6.73012        parabolic
    8       0.399968      6.73012        parabolic
Optimisation terminated:
 the current x satisfies the termination criteria using OPTIONS.TolX of 1.000000e-04
NumericalMLE =   0.4000
```

Example 29 (MLE of an IID Exponential(λ^*) **experiment)** *Let us derive the MLE* $\widehat{\Lambda}_n$ *of the fixed and possibly unknown* λ^* *for the IID experiment:*

$$X_1, \ldots, X_n \overset{IID}{\sim} \text{Exponential}(\lambda^*), \qquad \lambda^* \in \mathbf{\Lambda} = (0, \infty) \ .$$

Note that $\mathbf{\Lambda}$ *is the parameter space.*

We first obtain the log-likelihood function of λ *for the data* $x_1, x_2, \ldots, x_n \overset{IID}{\sim} \text{Exponential}(\lambda)$.

$$\begin{aligned}
\ell(\lambda) := \log(L(x_1, x_2, \ldots, x_n; \lambda)) &= \log\left(\prod_{i=1}^{n} f(x_i; \lambda)\right) = \log\left(\prod_{i=1}^{n} \lambda e^{-\lambda x_i}\right) \\
&= \log\left(\lambda e^{-\lambda x_1} \cdot \lambda e^{-\lambda x_2} \cdots \lambda e^{-\lambda x_n}\right) = \log\left(\lambda^n e^{-\lambda \sum_{i=1}^{n} x_i}\right) \\
&= \log\left(\lambda^n\right) + \log\left(e^{-\lambda \sum_{i=1}^{n} x_i}\right) = \log\left(\lambda^n\right) - \lambda \sum_{i=1}^{n} x_i
\end{aligned}$$

Now, let us take the derivative with respect to λ,

$$\frac{\partial}{\partial\lambda}\left(\ell(\lambda)\right) := \frac{\partial}{\partial\lambda}\left(\log\left(\lambda^n\right) - \lambda\sum_{i=1}^{n} x_i\right) = \frac{\partial}{\partial\lambda}\left(\log\left(\lambda^n\right)\right) - \frac{\partial}{\partial\lambda}\left(\lambda\sum_{i=1}^{n} x_i\right)$$
$$= \frac{1}{\lambda^n}\frac{\partial}{\partial\lambda}\left(\lambda^n\right) - \sum_{i=1}^{n} x_i = \frac{1}{\lambda^n} n\lambda^{n-1} - \sum_{i=1}^{n} x_i = \frac{n}{\lambda} - \sum_{i=1}^{n} x_i \ .$$

Next, we set the derivative to 0, solve for λ, and set the solution equal to the ML estimate $\widehat{\lambda}_n$.

$$0 = \frac{\partial}{\partial\lambda}\left(\ell(\lambda)\right) \iff 0 = \frac{n}{\lambda} - \sum_{i=1}^{n} x_i \iff \sum_{i=1}^{n} x_i = \frac{n}{\lambda} \iff \lambda = \frac{n}{\sum_{i=1}^{n} x_i} \iff \boxed{\widehat{\lambda}_n = \frac{1}{\overline{x}_n}} \ .$$

Therefore, the ML estimate $\widehat{\lambda}_n$ of the unknown rate parameter $\lambda^* \in \mathbf{\Lambda}$ on the basis of n IID observations $x_1, x_2, \ldots, x_n \overset{IID}{\sim}$ Exponential(λ^*) is $1/\overline{x}_n$ and the ML estimator $\widehat{\Lambda}_n = 1/\overline{X}_n$. Let us apply this ML estimator of the rate parameter for the supposedly exponentially distributed waiting times at the on-campus Orbiter bus-stop.

Labwork 32 (Numerical MLE of λ from n IID Exponential(λ) RVs) *Joshua Fenemore and Yiran Wang collected data on waiting times between buses at an Orbiter bus-stop close to campus and modelled the waiting times as IID Exponential(λ^*) RVs (`http://www.math.canterbury.ac.nz/~r.sainudiin/courses/STAT218/projects/Stat218StudentProjects2007.pdf`). We can use their data `sampleTimes` to find the MLE of λ^* under the assumption that the waiting times $X_1, \ldots, X_{132}$ are IID Exponential(λ^*). We find the ML estimate $\widehat{\lambda}_{132} = 0.1102$ and thus the estimated mean waiting time is $1/\widehat{\lambda}_{132} = 9.0763$ minutes. The estimated mean waiting time for a bus to arrive is well within the 10 minutes promised by the Orbiter bus company. The following script was used to generate the Figure 9.2:*

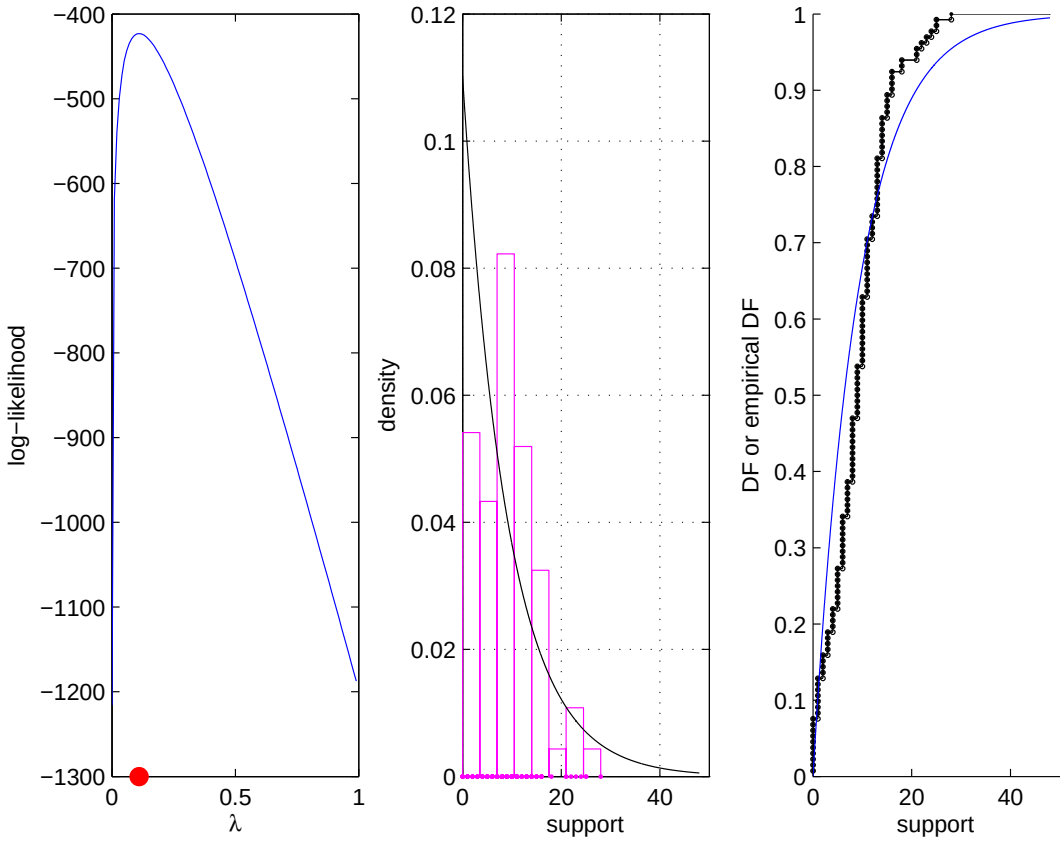
ExponentialMLEOrbiter.m

```
% Joshu Fenemore's Data from 2007 on Waiting Times at Orbiter Bust Stop
%The raw data -- the waiting times i minutes for each direction
antiTimes=[8 3 7 18 18 3 7 9 9 25 0 0 25 6 10 0 10 8 16 9 1 5 16 6 4 1 3 21 0 28 3 8 ...
6 6 11 8 10 15 0 8 7 11 10 9 12 13 8 10 11 8 7 11 5 9 11 14 13 5 8 9 12 10 13 6 11 13];
clockTimes=[0 0 11 1 9 5 14 16 2 10 21 1 14 2 10 24 6 1 14 14 0 14 4 11 15 0 10 2 13 2 22 ...
10 5 6 13 1 13 10 11 4 7 9 12 8 16 15 14 5 10 12 9 8 0 5 13 13 6 8 4 13 15 7 11 6 23 1];
sampleTimes=[antiTimes clockTimes];% pool all times into 1 array
% L = Log Likelihood of data x as a function of parameter lambda
L=@(lambda)sum(log(lambda*exp(-lambda * sampleTimes)));
LAMBDAS=[0.0001:0.01:1]; % sample some values for lambda
clf;
subplot(1,3,1);
plot(LAMBDAS,arrayfun(L,LAMBDAS)); % plot the Log Likelihood function
% Now we will find the Maximum Likelihood Estimator by finding the minimizer of -L
MLE = fminbnd(@(lambda)-sum(log(lambda*exp(-lambda * sampleTimes))),0.0001,1)
MeanEstimate=1/MLE
hold on; % plot the MLE
plot([MLE],[-1300],'r.','MarkerSize',25); ylabel('log-likelihood'); xlabel('\lambda');
subplot(1,3,2); % plot a histogram estimate
histogram(sampleTimes,1,[min(sampleTimes),max(sampleTimes)],'m',2);
hold on; TIMES=[0.00001:0.01:max(sampleTimes)+20]; % points on support
plot(TIMES,MLE*exp(-MLE * TIMES ),'k-') % plot PDF at MLE to compare with histogram
% compare the empirical DF to the best fitted DF
subplot(1,3,3)
ECDF(sampleTimes,5,0.0,20); hold on
plot(TIMES,ExponentialCdf(TIMES,MLE),'b-')
ylabel('DF or empirical DF'); xlabel('support');
```

The script output the following in addition to the plot:

```
>> ExponentialMLEOrbiter
MLE =    0.1102
MeanEstimate =    9.0763
```

Figure 9.2: Plot of $\log(L(\lambda))$ as a function of the parameter λ and the MLE $\widehat{\lambda}_{132}$ of 0.1102 for Fenemore-Wang Orbiter Waiting Times Experiment from STAT 218 S2 2007. The density or PDF and the DF at the MLE of 0.1102 are compared with a histogram and the empirical DF.



Notice how poorly the exponential PDF $f(x; \widehat{\lambda}_{132} = 0.1102)$ and the DF $F(x; \widehat{\lambda}_{132} = 0.1102)$ based on the MLE fits with the histogram and the empirical DF, respectively. This is an indication of the inadequacy of our parametric model. Partly this discrepancy is due to the resolution of the the measurements being confined to whole minutes. We can overcome this problem by fitting a minute-discretized PMF from the Exponential(λ) PDF. In the next Labwork, we simulate data from an Exponential($\lambda^* = 0.1$) RV to conduct point estimation in the theoretically ideal setting.

Labwork 33 (MLE of the rate parameter for waiting times at my bus stop) *Recall Labwork 19 where you modeled the arrival of buses at a bus stop using the IID* Exponential($\lambda^* = 0.1$) *distributed inter-arrival times with a mean of* $1/\lambda^* = 10$ *minutes. Once again, seed the fundamental sampler by your Student ID (e.g. if your ID is* `11424620` *then type* `rand('twister', 11424620);`*), just before simulating the inter-arrival times of the next seven buses. Hand in the following six items:*

1. *Waiting times $x_1, x_2, \ldots, x_7$ between arrivals of the next seven buses at your ID-seeded bus stop;*

2. *A plot of the empirical DF $\widehat{F}_n$ from your (simulated) data $x_1, x_2, \ldots, x_7$. [You may use the* MATLAB *function* `ECDF` *of Labwork 49)];*

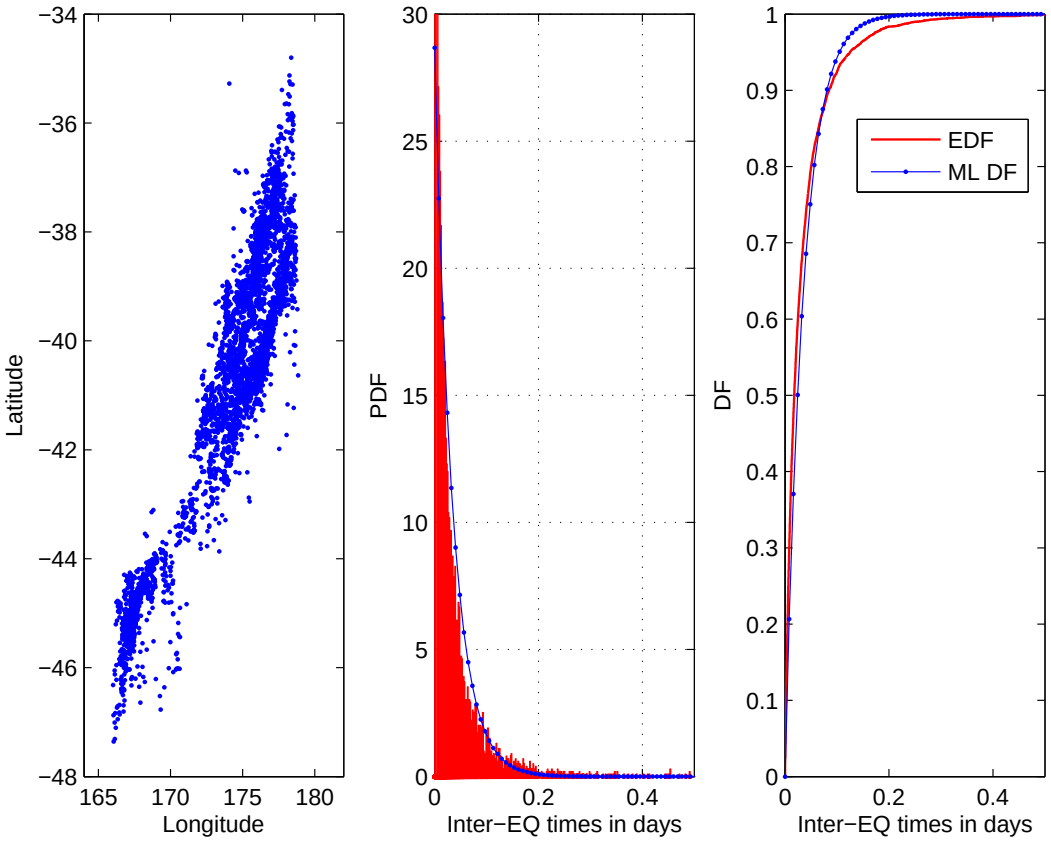
3. *The first, second and third sample quartiles as well as the 0.20^{th} sample quantile for your data $x_1, x_2, \ldots, x_7$. [You may use the* MATLAB *function* `qthSampleQuantile` *of Labwork 50];*

4. *Pretending that you did not know the true parameter ($\lambda^* = 0.1$) used in the simulation, produce the maximum likelihood estimate (ML estimate) $\widehat{\lambda}_7$ from your seven observations $x_1, x_2, \ldots, x_7$;*

5. *Plot the log-likelihood function for your data $x_1, x_2, \ldots, x_7$ as a function of the parameter λ; and*

6. *Show that you have verified that the numerical optimisation routine* `fminbnd` *returns the correct ML estimate $\widehat{\lambda}_7$.*

Figure 9.3: Comparing the Exponential($\widehat{\lambda}_{6128} = 28.6694$) PDF and DF with a histogram and empirical DF of the times (in units of days) between earth quakes in NZ. The epicentres of 6128 earth quakes are shown in left panel.



Labwork 34 (Time between Earth Quakes in NZ) *We model the time between* 6128 *earthquakes in NZ from 18-Jan-2008 02:23:44 to 18-Aug-2008 19:29:29 as:*

$$X_1, X_2, \ldots, X_{6128} \overset{IID}{\sim} \text{Exponential}(\lambda^*) \ .$$

Then, the ML estimate of $\lambda^* = 1/\overline{x}_{6128} = 1/0.0349 = 28.6694$ *as computed in the following script:*

NZSIEarthQuakesExponentialMLE.m

```
%% The columns in earthquakes.csv file have the following headings
%%CUSP_ID,LAT,LONG,NZMGE,NZMGN,ORI_YEAR,ORI_MONTH,ORI_DAY,ORI_HOUR,ORI_MINUTE,ORI_SECOND,MAG,DEPTH
EQ=dlmread('earthquakes.csv',','); % load the data in the matrix EQ
size(EQ) % report thr size of the matrix EQ

% Read help datenum -- converts time stamps into numbers in units of days
MaxD=max(datenum(EQ(:,6:11)));% maximum datenum
MinD=min(datenum(EQ(:,6:11)));% minimum datenum
disp('Earth Quakes in NZ between')
disp(strcat(datestr(MinD),' and ',datestr(MaxD)))% print MaxD and MinD as a date string

% get an array of sorted time stamps of EQ events starting at 0
Times=sort(datenum(EQ(:,6:11))-MinD);
TimeDiff=diff(Times); % inter-EQ times = times between successtive EQ events
clf % clear any current figures
%figure
%plot(TimeDiff) % plot the inter-EQ times
subplot(1,3,1)
plot(EQ(:,3),EQ(:,2),'.')
axis([164 182 -48 -34])
xlabel('Longitude'); ylabel('Latitude');

subplot(1,3,2) % construct a histogram estimate of inter-EQ times
histogram(TimeDiff',1,[min(TimeDiff),max(TimeDiff)],'r',2);
SampleMean=mean(TimeDiff) % find the sample mean
% the MLE of LambdaStar if inter-EQ times are IID Exponential(LambdaStar)
MLELambdaHat=1/SampleMean
hold on;
TIMES=linspace(0,max(TimeDiff),100);
plot(TIMES,MLELambdaHat*exp(-MLELambdaHat*TIMES),'b.-')
axis([0 0.5 0 30])
xlabel('Inter-EQ times in days'); ylabel('PDF');

subplot(1,3,3)
[x y]=ECDF(TimeDiff,0,0,0); % get the coordinates for empirical DF
stairs(x,y,'r','linewidth',1) % draw the empirical DF
hold on; plot(TIMES,ExponentialCdf(TIMES,MLELambdaHat),'b.-');% plot the DF at MLE
axis([0 0.5 0 1])
xlabel('Inter-EQ times in days'); ylabel('DF'); legend('EDF','ML DF')
```

We first load the data in the text file `earthquakes.csv` *into a matrix* `EQ`*. Using the* `datenum` *function in* MATLAB *we transform the time stamps into a number starting at zero. These transformed time stamps are in units of days. Then we find the times between consecutive events and estimate a histogram. We finally compute the ML estimate of* λ^* *and super-impose the PDF of the* $\text{Exponential}(\widehat{\lambda}_{6128} = 28.6694)$ *upon the histogram.*

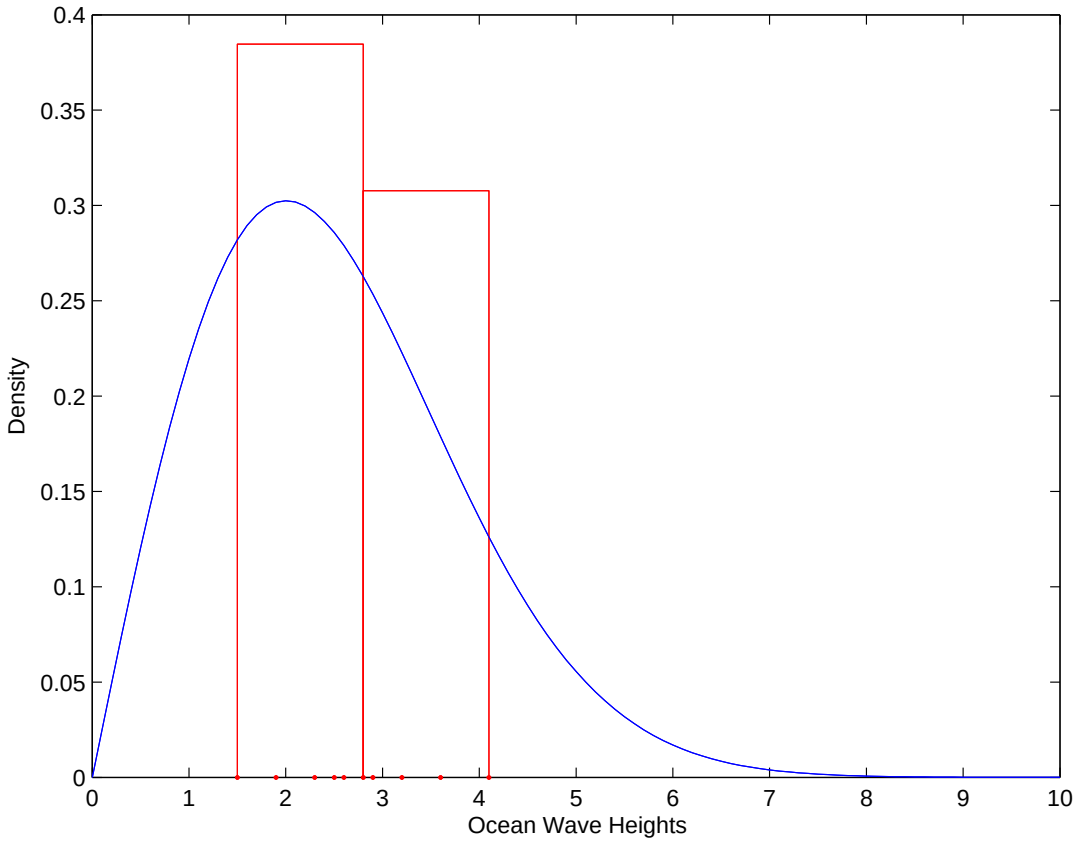
```
>> NZSIEarthQuakesExponentialMLE
ans =        6128          13

Earth Quakes in NZ between
18-Jan-2008 02:23:44 and18-Aug-2008 19:29:29

SampleMean =    0.0349
MLELambdaHat =   28.6694
```

Thus, the average time between earth quakes is $0.0349 * 24 * 60 = 50.2560$ *minutes.*

Figure 9.4: The ML fitted Rayleigh($\widehat{\alpha}_{10} = 2$) PDF and a histogram of the ocean wave heights.



Labwork 35 (6.7, p. 275 of Ang & Tang) *The distribution of ocean wave heights,* H*, may be modeled with the* Rayleigh(α) *RV with parameter* α *and probability density function,*

$$f(h;\alpha) = \frac{h}{\alpha^2}\exp\left(-\frac{1}{2}(h/\alpha)^2\right), \qquad h \in \mathbb{H} := [0,\infty) \ .$$

The parameter space for alpha is $\mathbb{A} = (0,\infty)$*. Suppose that the following measurements* $h_1, h_2, \ldots, h_{10}$ *of wave heights in meters were observed to be*

$$1.50,\ 2.80,\ 2.50,\ 3.20,\ 1.90,\ 4.10,\ 3.60,\ 2.60,\ 2.90,\ 2.30 \ ,$$

respectively. Under the assumption that the 10 *samples are IID realisations from a* Rayleigh(α^*) *RV with a fixed and unknown parameter* α^**, find the ML estimate* $\widehat{\alpha}_{10}$ *of* α^**.*

We first obtain the log-likelihood function of α for the data $h_1, h_2, \ldots, h_n \overset{IID}{\sim} \text{Rayleigh}(\alpha)$.

$$\begin{aligned}\ell(\alpha) := \log(L(h_1, h_2, \ldots, h_n; \alpha)) &= \log\left(\prod_{i=1}^{n} f(h_i;\alpha)\right) = \sum_{i=1}^{n} \log(f(h_i;\alpha)) \\ &= \sum_{i=1}^{n} \log\left(\frac{h_i}{\alpha^2} e^{-\frac{1}{2}(h_i/\alpha)^2}\right) = \sum_{i=1}^{n}\left(\log(h_i) - 2\log(\alpha) - \frac{1}{2}(h_i/\alpha)^2\right) \\ &= \sum_{i=1}^{n} (\log(h_i)) - 2n\log(\alpha) - \sum_{i=1}^{n}\left(\frac{1}{2}h_i^2\alpha^{-2}\right)\end{aligned}$$

Now, let us take the derivative with respect to α,

$$\begin{aligned}\frac{\partial}{\partial\alpha}(\ell(\alpha)) &:= \frac{\partial}{\partial\alpha}\left(\sum_{i=1}^{n} (\log(h_i)) - 2n\log(\alpha) - \sum_{i=1}^{n}\left(\frac{1}{2}h_i^2\alpha^{-2}\right)\right) \\ &= \frac{\partial}{\partial\alpha}\left(\sum_{i=1}^{n} (\log(h_i))\right) - \frac{\partial}{\partial\alpha}(2n\log(\alpha)) - \frac{\partial}{\partial\alpha}\left(\sum_{i=1}^{n}\left(\frac{1}{2}h_i^2\alpha^{-2}\right)\right) \\ &= 0 - 2n\frac{1}{\alpha} - \sum_{i=1}^{n}\left(\frac{1}{2}h_i^2(-2\alpha^{-3})\right) = -2n\alpha^{-1} + \alpha^{-3}\sum_{i=1}^{n}\left(h_i^2\right)\end{aligned}$$

Next, we set the derivative to 0, solve for α, and set the solution equal to the ML estimate $\widehat{\alpha}_n$.

$$\begin{aligned}0 = \frac{\partial}{\partial\alpha}(\ell(\alpha)) &\iff 0 = -2n\alpha^{-1} + \alpha^{-3}\sum_{i=1}^{n} h_i^2 \iff 2n\alpha^{-1} = \alpha^{-3}\sum_{i=1}^{n} h_i^2 \\ &\iff 2n\alpha^{-1}\alpha^3 = \sum_{i=1}^{n} h_i^2 \iff \alpha^2 = \frac{1}{2n}\sum_{i=1}^{n} h_i^2 \iff \widehat{\alpha}_n = \sqrt{\frac{1}{2n}\sum_{i=1}^{n} h_i^2}\end{aligned}$$

Therefore, the ML estimate of the unknown $\alpha^* \in \mathbb{A}$ on the basis of our 10 observations $h_1, h_2, \ldots, h_{10}$ of wave heights is

$$\begin{aligned}\widehat{\alpha}_{10} &= \sqrt{\frac{1}{2*10}\sum_{i=1}^{10} h_i^2} \\ &= \sqrt{\frac{1}{20}\left(1.50^2 + 2.80^2 + 2.50^2 + 3.20^2 + 1.90^2 + 4.10^2 + 3.60^2 + 2.60^2 + 2.90^2 + 2.30^2\right)} \approxeq 2\end{aligned}$$

We use the following script file to compute the MLE $\widehat{\alpha}_{10}$ and plot the PDF at $\widehat{\alpha}_{10}$ in Figure 9.4.

RayleighOceanHeightsMLE.m

```
OceanHeights=[1.50, 2.80, 2.50, 3.20, 1.90, 4.10, 3.60, 2.60, 2.90, 2.30];% data
histogram(OceanHeights,1,[min(OceanHeights),max(OceanHeights)],'r',2); % make a histogram
Heights=0:0.1:10; % get some heights for plotting
AlphaHat=sqrt(sum(OceanHeights .^ 2)/(2*length(OceanHeights))) % find the MLE
hold on; % superimpose the PDF at the MLE
plot(Heights,(Heights/AlphaHat^2) .* exp(-((Heights/AlphaHat).^2)/2))
xlabel('Ocean Wave Heights'); ylabel('Density');
```

```
>> RayleighOceanHeightsMLE
AlphaHat =    2.0052
```

9.3 Properties of the Maximum Likelihood Estimator

Next, we list some nice properties of the ML Estimator $\widehat{\Theta}_n$ for the fixed and possibly unknown $\theta^* \in \mathbf{\Theta}$.

1. The ML Estimator is asymptotically consistent, i.e. $\widehat{\Theta}_n \xrightarrow{P} \theta^*$.
2. The ML Estimator is asymptotically normal, i.e. $(\widehat{\Theta}_n - \theta^*)/\widehat{se}_n \rightsquigarrow \text{Normal}(0,1)$.
3. The estimated standard error of the ML Estimator, $\widehat{\mathsf{se}}_n$, can usually be computed analytically using the **Fisher Information**.
4. Because of the previous two properties, the $1-\alpha$ confidence interval can also be computed analytically as $\widehat{\Theta}_n \pm z_{\alpha/2}\widehat{\mathsf{se}}_n$.
5. The ML Estimator is **equivariant**, i.e. $\widehat{\psi}_n = g(\widehat{\theta}_n)$ is the ML Estimate of $\psi^* = g(\theta^*)$, for some smooth function $g(\theta) = \psi : \mathbf{\Theta} \mapsto \mathbf{\Psi}$.
6. We can also obtain the estimated standard error of the estimator $\widehat{\Psi}_n$ of $\psi^* \in \mathbf{\Psi}$ via the **Delta Method**.
7. The ML Estimator is **asymptotically optimal** or **efficient**. This means that the MLE has the smallest variance among the well-behaved class of estimators as the sample size gets larger.
8. ML Estimator is close to the Bayes estimator (obtained in the Bayesian inferential paradigm).

9.4 Fisher Information

Let $X_1, X_2, \ldots, X_n \overset{IID}{\sim} f(X_1;\theta)$. Here, $f(X_1;\theta)$ is the probability density function (pdf) or the probability mass function (pmf) of the RV X_1. Since all RVs are identically distributed, we simply focus on X_1 without loss of generality.

Definition 47 (Fisher Information) *The* **score function** *of an RV X for which the density is parameterised by θ is defined as:*

$$\mathcal{S}(X;\theta) := \frac{\partial log f(X;\theta)}{\partial \theta}, \qquad \textit{and} \quad \mathbf{E}_\theta(\mathcal{S}(X;\theta)) = 0 \ .$$

The **Fisher Information** *is*

$$I_n := \mathbf{V}_\theta\left(\sum_{i=1}^{n} \mathcal{S}(X_i;\theta)\right) = \sum_{i=1}^{n} \mathbf{V}_\theta\left(\mathcal{S}(X_i;\theta)\right) = nI_1(\theta), \tag{9.1}$$

where I_1 is the Fisher Information of just one of the RVs X_i,e.g. X:

$$\begin{aligned} I_1(\theta) &:= \mathbf{V}_\theta\left(\mathcal{S}(X;\theta)\right) = \mathbf{E}_\theta\left(\mathcal{S}^2(X,\theta)\right) \\ &= -\mathbf{E}_\theta\left(\frac{\partial^2 log f(X;\theta)}{\partial^2 \theta}\right) = \begin{cases} -\sum_{x\in\mathbb{X}}\left(\frac{\partial^2 \log f(x;\theta)}{\partial^2\theta}\right) f(x;\theta) & \textit{for discrete } X \\ -\int_{x\in\mathbb{X}}\left(\frac{\partial^2 \log f(x;\theta)}{\partial^2\theta}\right) f(x;\theta)dx & \textit{for continuous } X \end{cases} \end{aligned} \tag{9.2}$$

Next, we give a **general method** for obtaining:

1. The standard error $\mathsf{se}_n(\widehat{\Theta}_n)$ of **any** maximum likelihood estimator $\widehat{\Theta}_n$ of the possibly unknown and fixed parameter of interest $\theta^* \in \mathbf{\Theta}$, and
2. The $1-\alpha$ confidence interval for θ^*.

Proposition 15 (Asymptotic Normality of the ML Estimator & Confidence Intervals) *Let $\widehat{\Theta}_n$ be the maximum likelihood estimator of $\theta^* \in \mathbf{\Theta}$ with standard error $\mathsf{se}_n := \sqrt{\mathbf{V}_{\theta^*}(\widehat{\Theta}_n)}$. Under appropriate regularity conditions, the following propositions are true:*

1. *The standard error se_n can be approximated by the side of a square whose area is the inverse Fisher Information at θ^*, and the distribution of $\widehat{\Theta}_n$ approaches that of the $\text{Normal}(\theta^*, \mathsf{se}_n^2)$ distribution as the samples size n gets larger. In other terms:*

$$\mathsf{se}_n \approxeq \sqrt{1/I_n(\theta^*)} \qquad \text{and} \quad \frac{\widehat{\Theta}_n - \theta^*}{\mathsf{se}_n} \rightsquigarrow \text{Normal}(0,1)$$

2. *The approximation holds even if we substitute the ML Estimate $\widehat{\theta}_n$ for θ^* and use the estimated standard error $\widehat{\mathsf{se}}_n$ instead of se_n. Let $\widehat{\mathsf{se}}_n = \sqrt{1/I_n(\widehat{\theta}_n)}$. Then:*

$$\frac{\widehat{\Theta}_n - \theta^*}{\widehat{\mathsf{se}}_n} \rightsquigarrow \text{Normal}(0,1)$$

3. *Using the fact that $\widehat{\Theta}_n \rightsquigarrow \text{Normal}(\theta^*, \widehat{se}_n^2)$, we can construct the estimate of an approximate Normal-based $1-\alpha$ confidence interval as:*

$$C_n = [\underline{C}_n, \overline{C}_n] = [\widehat{\theta}_n - z_{\alpha/2}\widehat{se}_n, \widehat{\theta}_n + z_{\alpha/2}\widehat{\mathsf{se}}_n] = \widehat{\theta}_n \pm z_{\alpha/2}\widehat{\mathsf{se}}_n$$

Now, let us do an example.

Example 30 (MLE and Confidence Interval for the IID Poisson(λ) experiment) *Suppose the fixed parameter $\lambda^* \in \mathbf{\Lambda} = (0, \infty)$ is unknown. Let $X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{Poisson}(\lambda^*)$. We want to find the ML Estimate $\widehat{\lambda}_n$ of λ^* and produce a $1-\alpha$ confidence interval for λ^*.*

The MLE can be obtained as follows:

The likelihood function is:

$$L(\lambda) := L(x_1, x_2, \ldots, x_n; \lambda) = \prod_{i=1}^{n} f(x_i; \lambda) = \prod_{i=1}^{n} e^{-\lambda}\frac{\lambda_i^x}{x_i!}$$

Hence, the log-likelihood function is:

$$\begin{aligned}
\ell(\theta) := \log(L(\lambda)) &= \log\left(\prod_{i=1}^{n} e^{-\lambda}\frac{\lambda^{x_i}}{x_i!}\right) = \sum_{i=1}^{n}\log\left(e^{-\lambda}\frac{\lambda^{x_i}}{x_i!}\right) = \sum_{i=1}^{n}\Big(\log(e^{-\lambda}) + \log(\lambda^{x_i}) - \log(x_i!)\Big) \\
&= \sum_{i=1}^{n}(-\lambda + x_i\log(\lambda) - \log(x_i!)) = \sum_{i=1}^{n} -\lambda + \sum_{i=1}^{n} x_i\log(\lambda) - \sum_{i=1}^{n}\log(x_i!) \\
&= n(-\lambda) + \log(\lambda)\left(\sum_{i=1}^{n} x_i\right) - \sum_{i=1}^{n}\log(x_i!)
\end{aligned}$$

Next, take the derivative of $\ell(\lambda)$*:*

$$\frac{\partial}{\partial\lambda}\ell(\lambda) = \frac{\partial}{\partial\lambda}\left(n(-\lambda) + \log(\lambda)\left(\sum_{i=1}^{n} x_i\right) - \sum_{i=1}^{n}\log(x_i!)\right) = n(-1) + \frac{1}{\lambda}\left(\sum_{i=1}^{n} x_i\right) + 0$$

and set it equal to 0 *to solve for* λ*, as follows:*

$$0 = n(-1) + \frac{1}{\lambda}\left(\sum_{i=1}^{n} x_i\right) + 0 \iff n = \frac{1}{\lambda}\left(\sum_{i=1}^{n} x_i\right) \iff \lambda = \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right) = \overline{x}_n$$

Finally, the ML Estimator of λ^* *is* $\widehat{\Lambda}_n = \overline{X}_n$ *and the ML estimate is* $\widehat{\lambda}_n = \overline{x}_n$*.*

Now, we want an $1-\alpha$ *confidence interval for* λ^* *using the* $\widehat{\mathsf{se}}_n \approxeq \sqrt{1/I_n(\widehat{\lambda}_n)}$ *that is based on the Fisher Information* $I_n(\lambda) = nI_1(\lambda)$ *given in* (9.1)*. We need* I_1 *given in* (9.2)*. Since* $X_1, X_2, \ldots, X_n \sim \text{Poisson}(\lambda)$*, we have discrete RVs:*

$$I_1 = -\sum_{x\in\mathbb{X}}\left(\frac{\partial^2\log(f(x;\lambda))}{\partial^2\lambda}\right)f(x;\lambda) = -\sum_{x=0}^{\infty}\left(\frac{\partial^2\log(f(x;\lambda))}{\partial^2\lambda}\right)f(x;\lambda)$$

First find

$$\begin{aligned}\frac{\partial^2\log(f(x;\lambda))}{\partial^2\lambda} &= \frac{\partial}{\partial\lambda}\left(\frac{\partial}{\partial\lambda}\log\left(f(x;\lambda)\right)\right) = \frac{\partial}{\partial\lambda}\left(\frac{\partial}{\partial\lambda}\log\left(e^{-\lambda}\frac{\lambda^x}{x!}\right)\right)\\ &= \frac{\partial}{\partial\lambda}\left(\frac{\partial}{\partial\lambda}\left(-\lambda + x\log(\lambda) - \log(x!)\right)\right) = \frac{\partial}{\partial\lambda}\left(-1 + \frac{x}{\lambda} - 0\right) = -\frac{x}{\lambda^2}\end{aligned}$$

Now, substitute the above expression into the right-hand side of I_1 to obtain:

$$I_1 = -\sum_{x=0}^{\infty}\left(-\frac{x}{\lambda^2}\right)f(x;\lambda) = \frac{1}{\lambda^2}\sum_{x=0}^{\infty}(x)\,f(x;\lambda) = \frac{1}{\lambda^2}\sum_{x=0}^{\infty}(x)\,e^{-\lambda}\frac{\lambda^x}{x!} = \frac{1}{\lambda^2}\mathbf{E}_\lambda(X) = \frac{1}{\lambda^2}\lambda = \frac{1}{\lambda}$$

In the third-to-last step above, we recognise the sum as the expectation of the Poisson(λ) RV X, namely $\mathbf{E}_\lambda(X) = \lambda$. Therefore, the estimated standard error is:

$$\widehat{\mathsf{se}}_n \approxeq \sqrt{1/I_n(\widehat{\lambda}_n)} = \sqrt{1/(nI_1(\widehat{\lambda}_n))} = \sqrt{1/(n(1/\widehat{\lambda}_n))} = \sqrt{\widehat{\lambda}_n/n}$$

and the approximate $1-\alpha$ confidence interval is

$$\widehat{\lambda}_n \pm z_{\alpha/2}\widehat{\mathsf{se}}_n = \widehat{\lambda}_n \pm z_{\alpha/2}\sqrt{\widehat{\lambda}_n/n}$$

Thus, using the MLE and the estimated standard error via the Fisher Information, we can carry out point estimation and confidence interval construction in **most** parametric families of RVs encountered in typical engineering applications.

Example 31 (Fisher Information of the Bernoulli **Experiment)** *Suppose* $X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{Bernoulli}(\theta^*)$*. Also, suppose that* $\theta^* \in \Theta = [0,1]$ *is unknown. We have already shown in Example 28 that the ML estimator of* θ^* *is* $\widehat{\theta}_n = \overline{X}_n$*. Using the identity:*

$$\widehat{\mathsf{se}}_n = \frac{1}{\sqrt{I_n(\widehat{\theta}_n)}}$$

(1) we can compute $\widehat{\mathsf{se}}_n(\widehat{\Theta}_n)$*, the estimated standard error of the unknown parameter* θ^* *as follows:*

$$\widehat{\mathsf{se}}_n(\widehat{\Theta}_n) = \frac{1}{\sqrt{I_n(\widehat{\theta}_n)}} = \frac{1}{\sqrt{nI_1(\widehat{\theta}_n)}} \ .$$

So, we need to first compute $I_1(\theta)$*, the Fisher Information of one sample. Due to* (9.2) *and the fact that the* Bernoulli(θ^*) *distributed RV* X *is discrete with probability mass function* $f(x;\theta) = \theta^x(1-\theta)^{1-x}$*, for* $x \in \mathbb{X} := \{0,1\}$*, we have,*

$$I_1(\theta) = -\mathbf{E}_\theta\left(\frac{\partial^2 log f(X;\theta)}{\partial^2\theta}\right) = -\sum_{x\in\mathbb{X}=\{0,1\}}\left(\frac{\partial^2 \log\left(\theta^x(1-\theta)^{1-x}\right)}{\partial^2\theta}\right)\theta^x(1-\theta)^{1-x}$$

Next, let us compute,

$$\begin{aligned}\frac{\partial^2 \log\left(\theta^x(1-\theta)^{1-x}\right)}{\partial^2\theta} &:= \frac{\partial}{\partial\theta}\left(\frac{\partial}{\partial\theta}\left(\log\left(\theta^x(1-\theta)^{1-x}\right)\right)\right) = \frac{\partial}{\partial\theta}\left(\frac{\partial}{\partial\theta}\left(x\log(\theta)+(1-x)\log(1-\theta)\right)\right)\\ &= \frac{\partial}{\partial\theta}\left(x\theta^{-1}+(1-x)(1-\theta)^{-1}(-1)\right) = \frac{\partial}{\partial\theta}\left(x\theta^{-1}-(1-x)(1-\theta)^{-1}\right)\\ &= x(-1)\theta^{-1-1}-(1-x)(-1)(1-\theta)^{-1-1}(-1) = -x\theta^{-2}-(1-x)(1-\theta)^{-2}\end{aligned}$$

Now, we compute the expectation I_1*, i.e. the sum over the two possible values of* $x \in \{0,1\}$*,*

$$\begin{aligned}I_1(\theta) &= -\sum_{x\in\mathbb{X}=\{0,1\}}\left(\frac{\partial^2 \log\left(\theta^x(1-\theta)^{1-x}\right)}{\partial^2\theta}\right)\theta^x(1-\theta)^{1-x}\\ &= -\left(\left(-0\ \theta^{-2}-(1-0)(1-\theta)^{-2}\right)\theta^0(1-\theta)^{1-0}+\left(-1\ \theta^{-2}-(1-1)(1-\theta)^{-2}\right)\theta^1(1-\theta)^{1-1}\right)\\ &= -\left(\left(0-1(1-\theta)^{-2}\right)\ 1\ (1-\theta)^1+\left(-\theta^{-2}-0\right)\theta^1\ 1\right) = (1-\theta)^{-2}(1-\theta)^1+\theta^{-2}\theta^1\\ &= (1-\theta)^{-1}+\theta^{-1} = \frac{1}{1-\theta}+\frac{1}{\theta} = \frac{\theta}{\theta(1-\theta)}+\frac{1-\theta}{\theta(1-\theta)} = \frac{\theta+(1-\theta)}{\theta(1-\theta)} = \frac{1}{\theta(1-\theta)}\end{aligned}$$

Therefore, the desired estimated standard error of our estimator, can be obtained by substituting the ML estimate $\widehat{\theta}_n = \overline{x}_n := n^{-1}\sum_{i=1}^n x_i$ *of the unknown* θ^* *as follows:*

$$\widehat{\mathsf{se}}_n(\widehat{\theta}_n) = \frac{1}{\sqrt{I_n(\widehat{\theta}_n)}} = \frac{1}{\sqrt{nI_1(\widehat{\theta}_n)}} = \sqrt{\frac{1}{n\frac{1}{\widehat{\theta}_n(1-\widehat{\theta}_n)}}} = \sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}} = \sqrt{\frac{\overline{x}_n(1-\overline{x}_n)}{n}} \ .$$

(2) Using $\widehat{\mathsf{se}}_n(\widehat{\theta}_n)$ *we can construct an approximate* 95% *confidence interval* C_n *for* θ^**, due to the asymptotic normality of the ML estimator of* θ^**, as follows:*

$$C_n = \widehat{\theta}_n \pm 1.96\sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}} = \overline{x}_n \pm 1.96\sqrt{\frac{\overline{x}_n(1-\overline{x}_n)}{n}}$$

Recall that C_n *is the realisation of a random set based on your observed samples or data* $x_1, x_2, \ldots, x_n$*. Furthermore,* C_n*'s construction procedure ensures the engulfing of the unknown* θ^* *with probability approaching* 0.95 *as the sample size* n *gets large.*

Example 32 ([Fisher Information of the Exponential **Experiment)** *] Let us get our hands dirty with a continuous RV next. Let* $X_1, X_2, \ldots, X_n \overset{IID}{\sim}$ Exponential(λ^*)*. We saw that the ML*

estimator of $\lambda^* \in \mathbb{\Lambda} = (0, \infty)$ *is* $\widehat{\Lambda}_n = 1/\overline{X}_n$ *and its ML estimate is* $\widehat{\lambda}_n = 1/\overline{x}_n$*, where* $x_1, x_2, \ldots, x_n$ *are our observed data.*

(1) Let us obtain the Fisher Information I_n *for this experiment to find the standard error:*

$$\widehat{\mathsf{se}}_n(\widehat{\Lambda}_n) = \frac{1}{\sqrt{I_n(\widehat{\lambda}_n)}} = \frac{1}{\sqrt{nI_1(\widehat{\lambda}_n)}}$$

and construct an approximate 95% *confidence interval for* λ^* *using the asymptotic normality of its ML estimator* $\widehat{\Lambda}_n$*.*

So, we need to first compute $I_1(\theta)$*, the Fisher Information of one sample. Due to* (9.2) *and the fact that the* Exponential(λ^*) *distributed RV* X *is continuous with probability density function* $f(x;\lambda) = \lambda e^{-\lambda x}$*, for* $x \in \mathbb{X} := [0, \infty)$*, we have,*

$$I_1(\theta) = -\mathbf{E}_\theta\left(\frac{\partial^2 log f(X;\theta)}{\partial^2\theta}\right) = -\int_{x\in\mathbb{X}=[0,\infty)}\left(\frac{\partial^2 \log\left(\lambda e^{-\lambda x}\right)}{\partial^2\lambda}\right)\lambda e^{-\lambda x}\, dx$$

Let us compute the above integrand next.

$$\begin{aligned}\frac{\partial^2 \log\left(\lambda e^{-\lambda x}\right)}{\partial^2\lambda} &:= \frac{\partial}{\partial\lambda}\left(\frac{\partial}{\partial\lambda}\left(\log\left(\lambda e^{-\lambda x}\right)\right)\right) = \frac{\partial}{\partial\lambda}\left(\frac{\partial}{\partial\lambda}\left(\log(\lambda) + \log(e^{-\lambda x}\right)\right)\\ &= \frac{\partial}{\partial\lambda}\left(\frac{\partial}{\partial\lambda}\left(\log(\lambda) - \lambda x\right)\right) = \frac{\partial}{\partial\lambda}\left(\lambda^{-1} - x\right) = -\lambda^{-2} - 0 = -\frac{1}{\lambda^2}\end{aligned}$$

Now, let us evaluate the integral by recalling that the expectation of the constant 1 *is 1 for any RV* X *governed by some parameter, say* θ*. For instance when* X *is a continuous RV,* $\mathbf{E}_\theta(1) = \int_{x\in\mathbb{X}} 1\ f(x;\theta) = \int_{x\in\mathbb{X}}\ f(x;\theta) = 1$*. Therefore, the Fisher Information of one sample is*

$$\begin{aligned}I_1(\theta) = -\int_{x\in\mathbb{X}=[0,\infty)}\left(\frac{\partial^2 \log\left(\lambda e^{-\lambda x}\right)}{\partial^2\lambda}\right)\lambda e^{-\lambda x}\, dx &= -\int_0^\infty\left(-\frac{1}{\lambda^2}\right)\lambda e^{-\lambda x}\, dx\\ &= -\left(-\frac{1}{\lambda^2}\right)\int_0^\infty \lambda e^{-\lambda x}\, dx = \frac{1}{\lambda^2}\ 1 = \frac{1}{\lambda^2}\end{aligned}$$

Now, we can compute the desired estimated standard error, by substituting in the ML estimate $\widehat{\lambda}_n = 1/(\overline{x}_n) := 1/\left(\sum_{i=1}^n x_i\right)$ *of* λ^**, as follows:*

$$\widehat{\mathsf{se}}_n(\widehat{\Lambda}_n) = \frac{1}{\sqrt{I_n(\widehat{\lambda}_n)}} = \frac{1}{\sqrt{nI_1(\widehat{\lambda}_n)}} = \frac{1}{\sqrt{n\frac{1}{\widehat{\lambda}_n^2}}} = \frac{\widehat{\lambda}_n}{\sqrt{n}} = \frac{1}{\sqrt{n}\ \overline{x}_n}$$

Using $\widehat{\mathsf{se}}_n(\widehat{\lambda}_n)$ *we can construct an approximate* 95% *confidence interval* C_n *for* λ^**, due to the asymptotic normality of the ML estimator of* λ^**, as follows:*

$$C_n = \widehat{\lambda}_n \pm 1.96\frac{\widehat{\lambda}_n}{\sqrt{n}} = \frac{1}{\overline{x}_n} \pm 1.96\frac{1}{\sqrt{n}\ \overline{x}_n}\ .$$

Let us compute the ML estimate and the 95% *confidence interval for the rate parameter for the waiting times at the Orbiter bus-stop (see labwork 32). The sample mean* $\overline{x}_{132} = 9.0758$ *and the ML estimate is:*

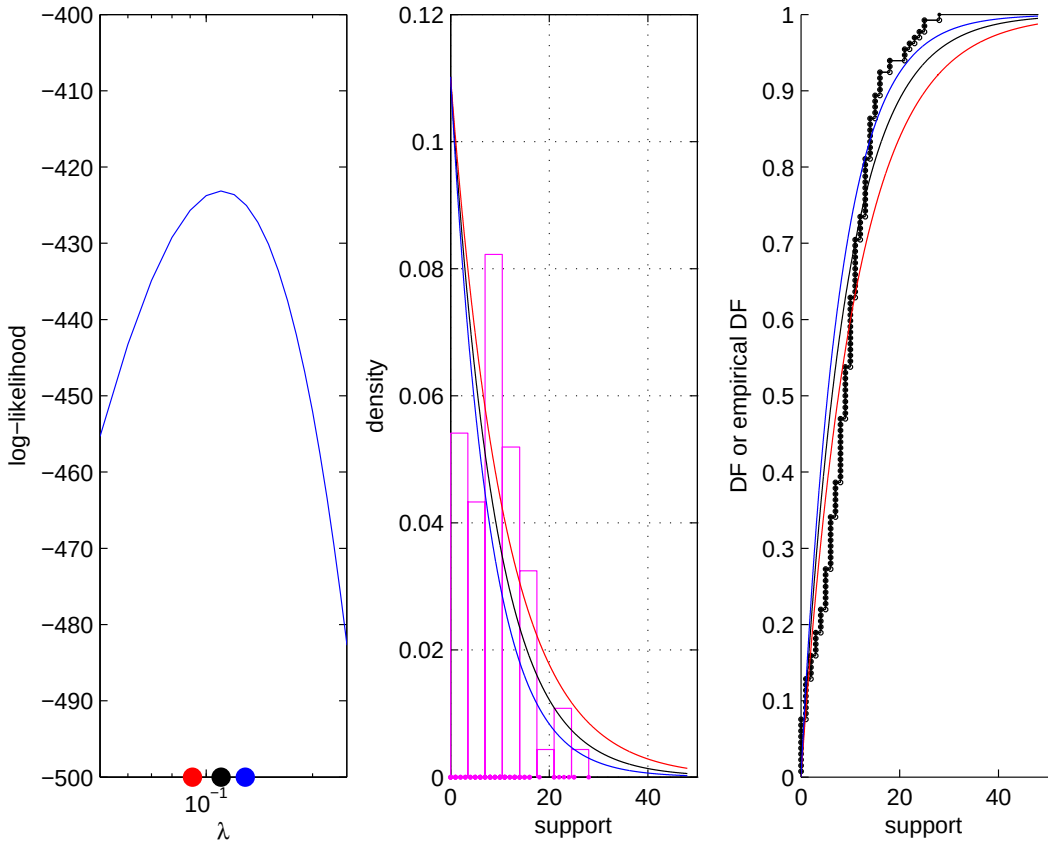
$$\widehat{\lambda}_{132} = 1/\overline{x}_{132} = 1/9.0758 = 0.1102\ ,$$

and the 95% confidence interval is:

$$C_n = \widehat{\lambda}_{132} \pm 1.96\frac{\widehat{\lambda}_{132}}{\sqrt{132}} = \frac{1}{\overline{x}_{132}} \pm 1.96\frac{1}{\sqrt{132}\,\overline{x}_{132}} = 0.1102 \pm 1.96 \cdot 0.0096 = [0.0914, 0.1290] \ .$$

Notice how poorly the exponential PDF $f(x; \widehat{\lambda}_{132} = 0.1102)$ and the DF $F(x; \widehat{\lambda}_{132} = 0.1102)$ based on the MLE fits with the histogram and the empirical DF, respectively, in Figure 9.5, despite taking the the confidence interval into account. This is a further indication of the inadequacy of our parametric model.

Figure 9.5: Plot of $\log(L(\lambda))$ as a function of the parameter λ, the MLE $\widehat{\lambda}_{132} = 0.1102$ and 95% confidence interval $C_n = [0.0914, 0.1290]$ for Fenemore-Wang Orbiter Waiting Times Experiment from STAT 218 S2 2007. The PDF and the DF at (1) the MLE 0.1102 (black), (2) lower 95% confidence bound 0.0914 (red) and (3) upper 95% confidence bound 0.1290 (blue) are compared with a histogram and the empirical DF.



Labwork 36 *The above analysis was undertaken with the following M-file:*

ExponentialMLECIOrbiter.m

```
OrbiterData; % load the Orbiter Data sampleTimes
% L = Log Likelihood of data x as a function of parameter lambda
L=@(lambda)sum(log(lambda*exp(-lambda * sampleTimes)));
LAMBDAS=[0.01:0.01:1]; % sample some values for lambda
```

```
clf;
subplot(1,3,1);
semilogx(LAMBDAS,arrayfun(L,LAMBDAS)); % plot the Log Likelihood function
axis([0.05 0.25 -500 -400])
SampleMean = mean(sampleTimes) % sample mean
MLE = 1/SampleMean % ML estimate is 1/mean
n=length(sampleTimes) % sample size
StdErr=1/(sqrt(n)*SampleMean) % Standard Error
MLE95CI=[MLE-(1.96*StdErr), MLE+(1.96*StdErr)] % 95 % CI
hold on; % plot the MLE
plot([MLE], [-500],'k.','MarkerSize',25);
plot([MLE95CI(1)],[-500],'r.','MarkerSize',25);
plot([MLE95CI(2)],[-500],'b.','MarkerSize',25);
 ylabel('log-likelihood'); xlabel('\lambda');
subplot(1,3,2); % plot a histogram estimate
histogram(sampleTimes,1,[min(sampleTimes),max(sampleTimes)],'m',2);
hold on; TIMES=[0.00001:0.01:max(sampleTimes)+20]; % points on support
% plot PDF at MLE and 95% CI to compare with histogram
plot(TIMES,MLE*exp(-MLE*TIMES),'k-')
plot(TIMES,MLE*exp(-MLE95CI(1)*TIMES),'r-'); plot(TIMES,MLE*exp(-MLE95CI(2)*TIMES),'b-')
% compare the empirical DF to the best fitted DF at MLE and 95% CI
subplot(1,3,3)
ECDF(sampleTimes,5,0.0,20); hold on; plot(TIMES,ExponentialCdf(TIMES,MLE),'k-');
plot(TIMES,ExponentialCdf(TIMES,MLE95CI(1)),'r-'); plot(TIMES,ExponentialCdf(TIMES,MLE95CI(2)),'b-')
ylabel('DF or empirical DF'); xlabel('support');
```

A call to the script generates Figure 9.5 and the following output of the sample mean, MLE, sample size, standard error and the 95% *confidence interval.*

```
>> ExponentialMLECIOrbiter
SampleMean =    9.0758
MLE =    0.1102
n =   132
StdErr =    0.0096
MLE95CI =    0.0914    0.1290
```

Labwork 37 *Recall labwork 19 where you modeled the arrival of buses using* Exponential($\lambda^* = 0.1$) *distributed inter-arrival time with a mean of* $1/\lambda^* = 10$ *minutes. Using the data of these seven inter-arrival times at your ID-seeded bus stop and pretending that you do not know the true* λ^**, report (1) the ML estimate of* λ^**, (2)* 95% *confidence interval for it and (3) whether the true value* $\lambda^* = 1/10$ *is engulfed by your confidence interval.*

9.5 Delta Method

A more general estimation problem of interest concerns some function of the parameter $\theta \in \mathbf{\Theta}$, say $g(\theta) = \psi : \mathbf{\Theta} \mapsto \mathbf{\Psi}$. So, $g(\theta) = \psi$ is a function from the parameter space $\mathbf{\Theta}$ to $\mathbf{\Psi}$. Thus, we are not only interested in estimating the fixed and possibly unknown $\theta^* \in \mathbf{\Theta}$ using the ML estimator $\widehat{\Theta}_n$ and its ML estimate $\widehat{\theta}_n$, but also in estimating $\psi^* = g(\theta^*) \in \mathbf{\Psi}$ via an estimator $\widehat{\Psi}_n$ and its estimate $\widehat{\psi}_n$. We exploit the equivariance property of the ML estimator $\widehat{\Theta}_n$ of θ^* and use the Delta method to find the following analytically:

1. The ML estimator of $\psi^* = g(\theta^*) \in \mathbf{\Psi}$ is

$$\boxed{\widehat{\Psi}_n = g(\widehat{\Theta}_n)}$$

and its point estimate is

$$\boxed{\widehat{\psi}_n = g(\widehat{\theta}_n)}$$

2. Suppose $g(\theta) = \psi : \mathbb{\Theta} \mapsto \mathbb{\Psi}$ is **any** smooth function of θ, i.e. g is differentiable, and $g'(\theta) := \frac{\partial}{\partial \theta} g(\theta) \neq 0$. Then, the distribution of the ML estimator $\widehat{\Psi}_n$ is asymptotically Normal$(\psi^*, \widehat{\mathsf{se}}_n(\widehat{\Psi}_n)^2)$, i.e.:

$$\boxed{\frac{\widehat{\Psi}_n - \psi^*}{\widehat{\mathsf{se}}_n(\widehat{\Psi}_n)} \rightsquigarrow \text{Normal}(0,1)}$$

where the standard error $\widehat{\mathsf{se}}_n(\widehat{\Psi}_n)$ of the ML estimator $\widehat{\Psi}_n$ of the unknown quantity $\psi^* \in \mathbb{\Psi}$ can be obtained from the standard error $\widehat{\mathsf{se}}_n(\widehat{\Theta}_n)$ of the ML estimator $\widehat{\Theta}_n$ of the parameter $\theta^* \in \mathbb{\Theta}$, as follows:

$$\boxed{\widehat{\mathsf{se}}_n(\widehat{\Psi}_n) = |g'(\widehat{\theta}_n)| \widehat{\mathsf{se}}_n(\widehat{\Theta}_n)}$$

3. Using Normal$(\psi^*, \widehat{\mathsf{se}}_n(\widehat{\Psi}_n)^2)$, we can construct the estimate of an approximate Normal-based $1-\alpha$ confidence interval for $\psi^* \in \mathbb{\Psi}$:

$$\boxed{C_n = [\underline{C}_n, \overline{C}_n] = \widehat{\psi}_n \pm z_{\alpha/2} \widehat{\mathsf{se}}_n(\widehat{\psi}_n)}$$

Let us do an example next.

Example 33 *Let* $X_1, X_2, \ldots, X_n \overset{IID}{\sim}$ Bernoulli(θ^*). *Let* $\psi = g(\theta) = \log(\theta/(1-\theta))$. *Suppose we are interested in producing a point estimate and confidence interval for* $\psi^* = g(\theta^*)$. *We can use the Delta method as follows:*

First, the estimated standard error of the ML estimator of θ^*, *as shown in Example 31, is*

$$\widehat{\mathsf{se}}_n(\widehat{\Theta}_n) = \sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}} \ .$$

The ML estimator of ψ^* *is:*

$$\widehat{\Psi}_n = \log(\widehat{\Theta}_n/(1-\widehat{\Theta}_n))$$

and the ML estimate of ψ^* *is:*

$$\widehat{\psi}_n = \log(\widehat{\theta}_n/(1-\widehat{\theta}_n)) \ .$$

Since, $g'(\theta) = 1/(\theta(1-\theta))$, *by the Delta method, the estimated standard error of the ML estimator of* ψ^* *is:*

$$\widehat{\mathsf{se}}_n(\widehat{\Psi}_n) = |g'(\widehat{\theta}_n)|(\widehat{\mathsf{se}}_n(\widehat{\Theta}_n)) = \frac{1}{\widehat{\theta}_n(1-\widehat{\theta}_n)} \sqrt{\frac{\widehat{\theta}_n(1-\widehat{\theta}_n)}{n}} = \frac{1}{\sqrt{n\widehat{\theta}_n(1-\widehat{\theta}_n)}} = \frac{1}{\sqrt{n\overline{x}_n(1-\overline{x}_n)}} \ .$$

An approximate 95% *confidence interval for* $\psi^* = \log(\theta^*/(1-\theta^*))$ *is:*

$$\widehat{\psi}_n \pm \frac{1.96}{\sqrt{n\widehat{\theta}_n(1-\widehat{\theta}_n)}} = \log(\widehat{\theta}_n/(1-\widehat{\theta}_n)) \pm \frac{1.96}{\sqrt{n\widehat{\theta}_n(1-\widehat{\theta}_n)}} = \log(\overline{x}_n/(1-\overline{x}_n)) \pm \frac{1.96}{\sqrt{n\overline{x}_n(1-\overline{x}_n)}} \ .$$

Example 34 (Delta Method for a Normal **Experiment)** *Let us try the Delta method on a continuous RV. Let* $X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{Normal}(\mu^*, \sigma^{*2})$. *Suppose that* μ^* *is known and* σ^* *is unknown. Let us derive the ML estimate* $\widehat{\psi}_n$ *of* $\psi^* = \log(\sigma^*)$ *and a* 95% *confidence interval for it in 6 steps.*

(1) First let us find the log-likelihood function $\ell(\sigma)$

$$\begin{aligned}
\ell(\sigma) &:= \log(L(\sigma)) := \log(L(x_1, x_2, \ldots, x_n; \sigma)) = \log\left(\prod_{i=1}^{n} f(x_i;\sigma)\right) = \sum_{i=1}^{n} \log\left(f(x_i;\sigma)\right) \\
&= \sum_{i=1}^{n} \log\left(\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}(x_i-\mu)^2\right)\right) \quad \because f(x_i;\sigma) \text{ in (6.9) is pdf of Normal}(\mu,\sigma^2) \text{ RV with known } \mu \\
&= \sum_{i=1}^{n} \left(\log\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \log\left(\exp\left(-\frac{1}{2\sigma^2}(x_i-\mu)^2\right)\right)\right) \\
&= \sum_{i=1}^{n} \log\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \sum_{i=1}^{n}\left(-\frac{1}{2\sigma^2}(x_i-\mu)^2\right) = n\log\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \left(-\frac{1}{2\sigma^2}\right)\sum_{i=1}^{n}(x_i-\mu)^2 \\
&= n\left(\log\left(\frac{1}{\sqrt{2\pi}}\right) + \log\left(\frac{1}{\sigma}\right)\right) - \left(\frac{1}{2\sigma^2}\right)\sum_{i=1}^{n}(x_i-\mu)^2 \\
&= n\log\left(\sqrt{2\pi}^{-1}\right) + n\log\left(\sigma^{-1}\right) - \left(\frac{1}{2\sigma^2}\right)\sum_{i=1}^{n}(x_i-\mu)^2 \\
&= -n\log\left(\sqrt{2\pi}\right) - n\log\left(\sigma\right) - \left(\frac{1}{2\sigma^2}\right)\sum_{i=1}^{n}(x_i-\mu)^2
\end{aligned}$$

(2) Let us find its derivative with respect to the unknown parameter σ *next.*

$$\begin{aligned}
\frac{\partial}{\partial\sigma}\ell(\sigma) &:= \frac{\partial}{\partial\sigma}\left(-n\log\left(\sqrt{2\pi}\right) - n\log\left(\sigma\right) - \left(\frac{1}{2\sigma^2}\right)\sum_{i=1}^{n}(x_i-\mu)^2\right) \\
&= \frac{\partial}{\partial\sigma}\left(-n\log\left(\sqrt{2\pi}\right)\right) - \frac{\partial}{\partial\sigma}\left(n\log\left(\sigma\right)\right) - \frac{\partial}{\partial\sigma}\left(\left(\frac{1}{2\sigma^2}\right)\sum_{i=1}^{n}(x_i-\mu)^2\right) \\
&= 0 - n\frac{\partial}{\partial\sigma}\left(\log(\sigma)\right) - \left(\frac{1}{2}\sum_{i=1}^{n}(x_i-\mu)^2\right)\frac{\partial}{\partial\sigma}\left(\sigma^{-2}\right) \\
&= -n\sigma^{-1} - \left(\frac{1}{2}\sum_{i=1}^{n}(x_i-\mu)^2\right)\left(-2\sigma^{-3}\right) = -n\sigma^{-1} + \sigma^{-3}\sum_{i=1}^{n}(x_i-\mu)^2
\end{aligned}$$

(3) Now, let us set the derivative equal to 0 *and solve for* σ.

$$\begin{aligned}
0 = -n\sigma^{-1} + \sigma^{-3}\sum_{i=1}^{n}(x_i-\mu)^2 &\iff n\sigma^{-1} = \sigma^{-3}\sum_{i=1}^{n}(x_i-\mu)^2 \iff n\sigma^{-1}\sigma^{+3} = \sum_{i=1}^{n}(x_i-\mu)^2 \\
&\iff n\sigma^{-1+3} = \sum_{i=1}^{n}(x_i-\mu)^2 \iff n\sigma^2 = \sum_{i=1}^{n}(x_i-\mu)^2 \\
&\iff \sigma^2 = \left(\sum_{i=1}^{n}(x_i-\mu)^2\right)/n \iff \sigma = \sqrt{\sum_{i=1}^{n}(x_i-\mu)^2/n}
\end{aligned}$$

Finally, we set the solution, i.e. the maximiser of the concave-down log-likelihood function of σ *with a known and fixed* μ^* *as our ML estimate* $\widehat{\sigma}_n = \sqrt{\sum_{i=1}^{n}(x_i-\mu^*)^2/n}$. *Analogously, the ML estimator*

of σ^* *is* $\widehat{\Sigma}_n = \sqrt{\sum_{i=1}^n (X_i - \mu^*)^2/n}$. *Don't confuse* Σ, *the upper-case sigma, with* $\sum_{i=1}^n \bigcirc_i$, *the summation over some* $\bigcirc_i$*'s. This is usually clear from the context.*

(4) Next, let us get the estimated standard error $\widehat{se}_n$ *for the estimator of* σ^* *via Fisher Information. The Log-likelihood function of* σ, *based on one sample from the* $\text{Normal}(\mu, \sigma^2)$ *RV with known* μ *is,*

$$\log f(x;\sigma) = \log\left(\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)\right) = -\log\left(\sqrt{2\pi}\right) - \log(\sigma) - \left(\frac{1}{2\sigma^2}\right)(x-\mu)^2$$

Therefore, in much the same way as in part (2) earlier,

$$\begin{aligned}\frac{\partial^2 \log f(x;\sigma)}{\partial^2\sigma} &:= \frac{\partial}{\partial\sigma}\left(\frac{\partial}{\partial\sigma}\left(-\log\left(\sqrt{2\pi}\right) - \log(\sigma) - \left(\frac{1}{2\sigma^2}\right)(x-\mu)^2\right)\right)\\ &= \frac{\partial}{\partial\sigma}\left(-\sigma^{-1} + \sigma^{-3}(x-\mu)^2\right) = \sigma^{-2} - 3\sigma^{-4}(x-\mu)^2\end{aligned}$$

Now, we compute the Fisher Information of one sample as an expectation of the continuous RV X *over* $\mathbb{X} = (-\infty, \infty)$ *with density* $f(x;\sigma)$,

$$\begin{aligned}I_1(\sigma) &= -\int_{x\in\mathbb{X}=(-\infty,\infty)}\left(\frac{\partial^2 \log f(x;\sigma)}{\partial^2\lambda}\right) f(x;\sigma)\ dx = -\int_{-\infty}^{\infty}\left(\sigma^{-2} - 3\sigma^{-4}(x-\mu)^2\right) f(x;\sigma)\ dx\\ &= \int_{-\infty}^{\infty} -\sigma^{-2} f(x;\sigma)\ dx + \int_{-\infty}^{\infty} 3\sigma^{-4}(x-\mu)^2 f(x;\sigma)\ dx\\ &= -\sigma^{-2}\int_{-\infty}^{\infty} f(x;\sigma)\ dx + 3\sigma^{-4}\int_{-\infty}^{\infty}(x-\mu)^2 f(x;\sigma)\ dx\\ &= -\sigma^{-2} + 3\sigma^{-4}\sigma^2 \qquad\qquad \because \sigma^2 = \mathbf{V}(X) = \mathbf{E}(X - \mathbf{E}(X))^2 = \int_{-\infty}^{\infty}(x-\mu)^2 f(x;\sigma)\ dx\\ &= -\sigma^{-2} + 3\sigma^{-4+2} = -\sigma^{-2} + 3\sigma^{-2} = 2\sigma^{-2}\end{aligned}$$

Therefore, the estimated standard error of the estimator of the unknown σ^* *is*

$$\widehat{\mathsf{se}}_n(\widehat{\Sigma}_n) = \frac{1}{\sqrt{I_n(\widehat{\sigma}_n)}} = \frac{1}{\sqrt{nI_1(\widehat{\sigma}_n)}} = \frac{1}{\sqrt{n2\sigma^{-2}}} = \frac{\sigma}{\sqrt{2n}}\ .$$

(5) Given that $\psi = g(\sigma) = \log(\sigma)$, *we derive the estimated standard error of* $\psi^* = \log(\sigma^*)$ *via the Delta method as follows:*

$$\widehat{\mathsf{se}}_n(\widehat{\Psi}_n) = |g'(\sigma)|\widehat{\mathsf{se}}_n(\widehat{\Sigma}_n) = \left|\frac{\partial}{\partial\sigma}\log(\sigma)\right|\frac{\sigma}{\sqrt{2n}} = \frac{1}{\sigma}\frac{\sigma}{\sqrt{2n}} = \frac{1}{\sqrt{2n}}\ .$$

(6) Finally, the 95% *confidence interval for* ψ^* *is* $\widehat{\psi}_n \pm 1.96\widehat{\mathsf{se}}_n(\widehat{\Psi}_n) = \log(\widehat{\sigma}_n) \pm 1.96\frac{1}{\sqrt{2n}}$.

Chapter 10

Non-parametric DF Estimation

So far, we have been interested in some estimation problems involved in parametric experiments. In parametric experiments, the parameter space $\mathbf{\Theta}$ can have many dimensions, but these are finite. For example, in the n IID Bernoulli(θ^*) and the n IID Exponential(λ^*) experiments:

$$X_1, \ldots, X_n \overset{IID}{\sim} \text{Bernoulli}(\theta^*), \qquad \theta^* \in \mathbf{\Theta} = [0,1] \subset \mathbb{R}^1 \ ,$$
$$X_1, \ldots, X_n \overset{IID}{\sim} \text{Exponential}(\lambda^*), \qquad \lambda^* \in \mathbf{\Lambda} = (0,\infty) \subset \mathbb{R}^1 \ ,$$

the parameter spaces $\mathbf{\Theta}$ and $\mathbf{\Lambda}$ are of dimension 1. Similarly, in the n IID Normal(μ, σ^2) and the n IID Lognormal(λ, ζ), experiments:

$$X_1, \ldots, X_n \overset{IID}{\sim} \text{Normal}(\mu, \sigma^2), \qquad (\mu, \sigma^2) \in \mathbf{\Theta} = (-\infty, +\infty) \times (0, +\infty) \subset \mathbb{R}^2$$
$$X_1, \ldots, X_n \overset{IID}{\sim} \text{Lognormal}(\lambda, \zeta), \qquad (\lambda, \zeta) \in \mathbf{\Theta} = (0, +\infty) \times (0, +\infty) \subset \mathbb{R}^2$$

the parameter space is of dimension 2.

An experiment with an infinite dimensional parameter space $\mathbf{\Theta}$ is said to be **non-parametric** . Next we consider a non-parametric experiment in which n IID samples are drawn according to some fixed and possibly unknown DF F^* from the space of **All Distribution Functions**:

$$\boxed{X_1, X_2, \ldots, X_n \overset{IID}{\sim} F^*, \qquad F^* \in \mathbf{\Theta} = \{\text{All DFs}\} := \{\, F(x;F) : F \ is \ a \ DF \,\}}$$

where the DF $F(x;F)$ is indexed or parameterised by itself. Thus, the parameter space $\mathbf{\Theta} = \{\text{All DFs}\}$ is the **infinite dimensional** space of **All DFs**. In this section, we look at estimation problems in non-parametric experiments with an infinite dimensional parameter space. That is, we want to estimate the DF F^* from which our IID data are drawn.

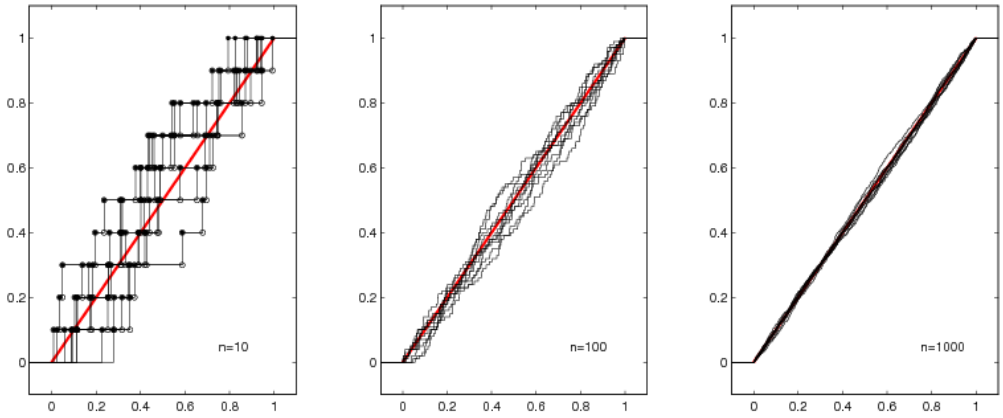
The next proposition is often referred to as the **fundamental theorem of statistics** and is at the heart of non-parametric inference, empirical processes, and computationally intensive bootstrap techniques. Recall Definition 30 of the n-sample empirical distribution function (EDF or ECDF) $\widehat{F}_n$ that assigns a probability mass of $1/n$ at each data point x_i:

$$\widehat{F}_n(x) = \frac{\sum_{i=1}^n \mathbb{1}(X_i \leq x)}{n} \ , \qquad where \qquad \mathbb{1}(X_i \leq x) := \begin{cases} 1 & \text{if } x_i \leq x \\ 0 & \text{if } x_i > x \end{cases}$$

Proposition 16 (Gilvenko-Cantelli Theorem) *Let* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F^*$. *Then:*

$$\sup_x |\widehat{F}_n(x) - F^*(x)| \xrightarrow{P} 0 \ .$$

Figure 10.1: Plots of ten distinct ECDFs $\widehat{F}_n$ based on 10 sets of n IID samples from Uniform$(0,1)$ RV X, as n increases from 10 to 100 to 1000. The DF $F(x) = x$ over $[0,1]$ is shown in red. The script of Labwork 55 was used to generate this plot.



Heuristic Interpretation of the Gilvenko-Cantelli Theorem: As the sample size n increases, the empirical distribution function $\widehat{F}_n$ converges to the true DF F^* in probability, as shown in Figure 10.1.

Proposition 17 (The Dvoretzky-Kiefer-Wolfowitz (DKW) Inequality) *Let* $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F^*$. *Then, for any* $\epsilon > 0$*:*

$$P\left(\sup_x |\widehat{F}_n(x) - F^*(x)| > \epsilon\right) \leq 2\exp\left(-2n\epsilon^2\right) \tag{10.1}$$

Recall that $\sup(A)$ *or supremum of a set* $A \subset \mathbb{R}$ *is the least upper bound of every element in* A.

10.1 Estimating DF

Let $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F^*$, where F^* is some particular DF in the space of all possible DFs, i.e. the experiment is non-parametric. Then, based on the data sequence $X_1, X_2, \ldots, X_n$ we want to estimate F^*.

For any fixed value of x, the expectation and variance of the empirical DF (10.1) are:

$$\mathbf{E}\left(\widehat{F}_n(x)\right) = F^*(x) \implies \mathsf{bias}_n\left(\widehat{F}_n(x)\right) = 0 \tag{10.2}$$

$$\mathbf{V}\left(\widehat{F}_n(x)\right) = \frac{F^*(x)(1-F^*(x))}{n} \implies \lim_{n\to\infty} \mathsf{se}_n\left(\widehat{F}_n(x)\right) = 0 \tag{10.3}$$

Therefore, by Proposition 13, the empirical DF evaluated at x, i.e. $\widehat{F}_n(x)$ is an asymptotically consistent estimator of the DF evaluated at x, i.e. $F^*(x)$. More formally, (10.2) and (10.3), by Proposition 13, imply that for any fixed value of x:

$$\widehat{F}_n(x) \xrightarrow{P} F^*(x) \ .$$

We are interested in a point estimate of the entire DF F^*, i.e. $F^*(x)$ over all x. A point estimator $T_n = T_n(X_1, X_2, \ldots, X_n)$ of a fixed and possibly unknown $F \in \{\text{All DFs}\}$ is the empirical DF $\widehat{F}_n$.

This estimator has an asymptotically desirable property:

$$\sup_x |\widehat{F}_n(x) - F^*(x)| \xrightarrow{P} 0$$

because of the Gilvenko-Cantelli theorem in Proposition 16. Thus, we can simply use $\widehat{F}_n$, based on the realized data $(x_1, x_2, \ldots, x_n)$, as a point estimate of F^*.

On the basis of the DKW inequality (10.1), we can obtain a $1-\alpha$ confidence set or **confidence band** $C_n(x) := [\underline{C}_n(x), \overline{C}_n(x)]$ about our point estimate of F^*:

$$\begin{aligned} \underline{C}_n(x) &= \max\{\widehat{F}_n(x) - \epsilon_n, 0\}, \\ \overline{C}_n(x) &= \min\{\widehat{F}_n(x) + \epsilon_n, 1\}, \\ \epsilon_n &= \sqrt{\frac{1}{2n}\log\left(\frac{2}{\alpha}\right)}\,. \end{aligned} \tag{10.4}$$

It follows from (10.1) that for any fixed and possibly unknown F^*:

$$P\left(\underline{C}_n(x) \le F^*(x) \le \overline{C}_n(x)\right) \ge 1-\alpha\,.$$

Let us look at a simple example next.

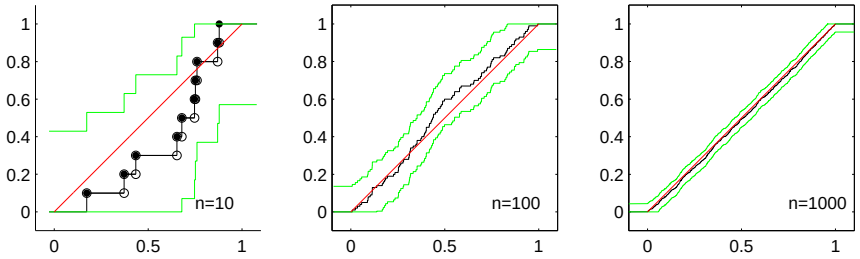
Labwork 38 (Estimating the DF of Uniform$(0,1)$ **RV)** *Consider the problem of estimating the DF of* Uniform$(0,1)$ *RV U on the basis of $n_=10$ samples. We use the function* `ECDF` *of Labwork 49 and* MATLAB*'s built-in function* `stairs` *to render the plots. Figure 10.2 was generated by* `PlotUniformECDFsConfBands.m` *given below.*

PlotUniformECDFsConfBands.m

```
% script PlotUniformECDFsConfBands.m to plot the ECDF from 10 and 100 samples
% from Uniform(0,1) RV
rand('twister',76534); % initialize the Uniform(0,1) Sampler
N = 3; % 10^N is the maximum number of samples from Uniform(0,1) RV
u = rand(1,10^N); % generate 1000 samples from Uniform(0,1) RV U

% plot the ECDF from the first 10 samples using the function ECDF
for i=1:N
    SampleSize=10^i;
    subplot(1,N,i)
    % Get the x and y coordinates of SampleSize-based ECDF in x1 and y1 and
    % plot the ECDF using the function ECDF
    if (i==1) [x1 y1] = ECDF(u(1:SampleSize),2,0.2,0.2);
    else
        [x1 y1] = ECDF(u(1:SampleSize),0,0.1,0.1);
        stairs(x1,y1,'k');
    end
    % Note PlotFlag is 1 and the plot range of x-axis is
    % incremented by 0.1 or 0.2 on either side due to last 2 parameters to ECDF
    % being 0.1 or 0.2
    Alpha=0.05; % set alpha to 5% for instance
    Epsn = sqrt((1/(2*SampleSize))*log(2/Alpha)); % epsilon_n for the confidence band
    hold on;
    stairs(x1,max(y1-Epsn,zeros(1,length(y1))),'g'); % lower band plot
    stairs(x1,min(y1+Epsn,ones(1,length(y1))),'g'); % upper band plot
    axis([-0.1 1.1 -0.1 1.1]);
    axis square;
    x=[0:0.001:1];
    plot(x,x,'r'); % plot the DF of Uniform(0,1) RV in red
    LabelString=['n=' num2str(SampleSize)];
    text(0.75,0.05,LabelString)
    hold off;
end
```

Figure 10.2: The empirical DFs $\widehat{F}_n^{(1)}$ from sample size $n = 10, 100, 1000$ (black), is the point estimate of the fixed and known DF $F(x) = x, x \in [0, 1]$ of Uniform$(0, 1)$ RV (red). The 95% confidence band for each $\widehat{F}_n$ are depicted by green lines.



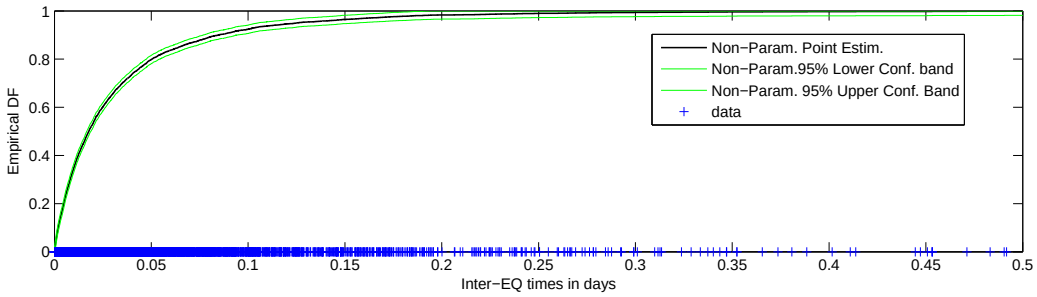
Next we look at a more interesting example involving real-world data.

Labwork 39 (Non-parametric Estimation of the DF of Times Between Earth Quakes) *Suppose that the* $6,128$ *observed times between Earth quakes in NZ between 18-Jan-2008 02:23:44 and 18-Aug-2008 19:29:29 are:*

$$X_1, \ldots, X_{6128} \overset{IID}{\sim} F^*, \qquad F^* \in \{all\ DFs\} \ .$$

Then the non-parametric point estimate of the unknown F^* *is* $\widehat{F}_{6128}$*, the ECDF of the inter earth quake times. We plot the non-parametric point estimate as well as the 95% confidence bands for* F^* *in Figure 10.3.*

Figure 10.3: The empirical DF $\widehat{F}_{6128}$ for the inter earth quake times and the 95% confidence bands for the non-parametric experiment.



NZSIEQTimesECDFsConfBands.m

```
%% The columns in earthquakes.csv file have the following headings
%%CUSP_ID,LAT,LONG,NZMGE,NZMGN,ORI_YEAR,ORI_MONTH,ORI_DAY,ORI_HOUR,ORI_MINUTE,ORI_SECOND,MAG,DEPTH
EQ=dlmread('earthquakes.csv',','); % load the data in the matrix EQ
size(EQ) % report thr size of the matrix EQ
% Read help datenum -- converts time stamps into numbers in units of days
MaxD=max(datenum(EQ(:,6:11)));% maximum datenum
MinD=min(datenum(EQ(:,6:11)));% minimum datenum
% get an array of sorted time stamps of EQ events starting at 0
Times=sort(datenum(EQ(:,6:11))-MinD);
TimeDiff=diff(Times); % inter-EQ times = times between successtive EQ events
n=length(TimeDiff); %sample size
clf % clear any current figures
%% Non-parametric Estimation X_1,X_2,...,X_132 ~ IID F
[x y] = ECDF(TimeDiff,0,0,0); % get the coordinates for empirical DF
```

```
stairs(x,y,'k','linewidth',1) % draw the empirical DF
hold on;
% get the 5% non-parametric confidence bands
Alpha=0.05; % set alpha to 5% for instance
Epsn = sqrt((1/(2*n))*log(2/Alpha)); % epsilon_n for the confidence band
stairs(x,max(y-Epsn,zeros(1,length(y))),'g'); % non-parametric 95% lower confidence band
stairs(x,min(y+Epsn,ones(1,length(y))),'g'); % non-parametric 95% upper confidence band
plot(TimeDiff,zeros(1,n),'+')
axis([0 0.5 0 1])
xlabel('Inter-EQ times in days'); ylabel('Empirical DF');
legend('Non-Param. Point Estim.','Non-Param.95% Lower Conf. band','Non-Param. 95% Upper Conf. Band','data')
```

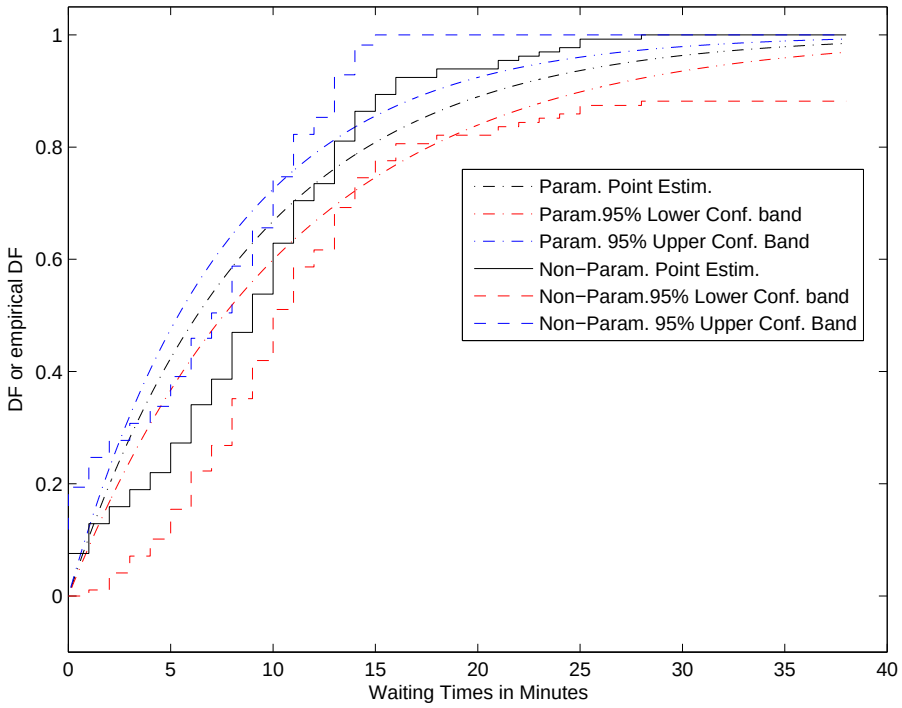
Recall the poor fit of the Exponential PDF at the MLE for the Orbiter waiting time data. We can attribute the poor fit to coarse resolution of the waiting time measurements in minutes and the rigid decaying form of the exponential PDFs. Let us revisit the Orbiter waiting time problem with our non-parametric estimator.

Labwork 40 (Non-parametric Estimation of Orbiter Waiting Times DF) *Suppose that the waiting times at the Orbiter bus stop are:*

$$X_1, \ldots, X_{132} \overset{IID}{\sim} F^*, \qquad F^* \in \{all\ DFs\} \ .$$

Then the non-parametric point estimate of F^* is $\widehat{F}_{132}$, the ECDF of the 132 Orbiter waiting times. We compute and plot the non-parametric point estimate as well as the 95% confidence bands for

Figure 10.4: The empirical DF $\widehat{F}_{132}$ for the Orbiter waiting times and the 95% confidence bands for the non-parametric experiment.



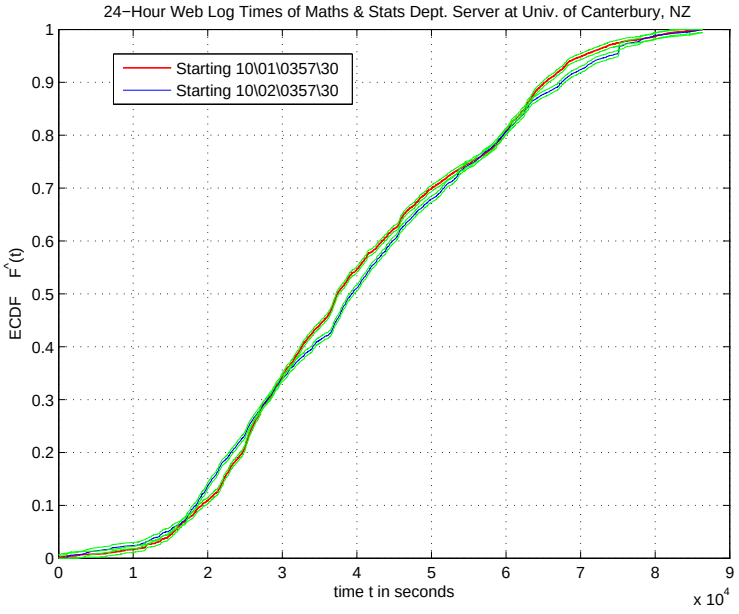
the unknown DF F^* beside the parametric estimate and 95% confidence bands from Labwork 36. Clearly, the non-parametric estimate is preferable to the parametric one for this example. Notice how the non-parametric confidence bands do not contain the parametric estimate of the DF.

OrbiterECDFsConfBands.m

```
OrbiterData; % load the Orbiter Data sampleTimes
clf; % clear any current figures
%% Parametric Estimation X_1,X_2,...,X_132 ~ IID Exponential(lambda)
SampleMean = mean(sampleTimes) % sample mean
MLE = 1/SampleMean % ML estimate is 1/mean
n=length(sampleTimes) % sample size
StdErr=1/(sqrt(n)*SampleMean) % Standard Error
MLE95CI=[MLE-(1.96*StdErr), MLE+(1.96*StdErr)] % 95 % CI
TIMES=[0.00001:0.01:max(sampleTimes)+10]; % points on support
plot(TIMES,ExponentialCdf(TIMES,MLE),'k-.'); hold on; % Parametric Point Estimate
plot(TIMES,ExponentialCdf(TIMES,MLE95CI(1)),'r-.');% Normal-based Parametric 95% lower C.I.
plot(TIMES,ExponentialCdf(TIMES,MLE95CI(2)),'b-.');% Normal-based Parametric 95% upper C.I.
ylabel('DF or empirical DF'); xlabel('Waiting Times in Minutes');
%% Non-parametric Estimation X_1,X_2,...,X_132 ~ IID F
[x1 y1] = ECDF(sampleTimes,0,0.0,10); stairs(x1,y1,'k');% plot the ECDF
% get the 5% non-parametric confidence bands
Alpha=0.05; % set alpha to 5% for instance
Epsn = sqrt((1/(2*n))*log(2/Alpha)); % epsilon_n for the confidence band
stairs(x1,max(y1-Epsn,zeros(1,length(y1))),'r--'); % non-parametric 95% lower confidence band
stairs(x1,min(y1+Epsn,ones(1,length(y1))),'b--'); % non-parametric 95% upper confidence band
axis([0 40 -0.1 1.05]);
legend('Param. Point Estim.','Param.95% Lower Conf. band','Param. 95% Upper Conf. Band',...
    'Non-Param. Point Estim.','Non-Param.95% Lower Conf. band','Non-Param. 95% Upper Conf. Band')
```

Example 35 *First take a look at Data 1 to understand how the web login times to our Maths & Stats Department's web server (or requests to our WWW server) were generated. Figure 10.5 shows the login times in units of seconds over a 24 hour period starting at 0357 hours and 30 seconds (just before 4:00AM) on October 1st, 2007 (red line) and on October 2nd, 2007 (magenta). If we assume*

Figure 10.5: The empirical DFs $\widehat{F}^{(1)}_{n_1}$ with $n_1 = 56485$, for the web log times starting October 1, and $\widehat{F}^{(2)}_{n_2}$ with $n_2 = 53966$, for the web log times starting October 2. Their 95% confidence bands are indicated by the green.



that some fixed and unknown DF $F^{(1)}$ specifies the distribution of login times for October 1st data and another DF $F^{(2)}$ for October 2nd data, then the non-parametric point estimates of $F^{(1)}$ and

$F^{(2)}$ are simply the empirical DFs $\widehat{F}^{(1)}_{n_1}$ with $n_1 = 56485$ and $\widehat{F}^{(2)}_{n_2}$ with $n_2 = 53966$, respectively, as depicted in Figure 10.5. See the script of `WebLogDataProc.m` *in Data 1 to appreciate how the ECDF plots in Figure 10.5 were made.*

10.2 Plug-in Estimators of Statistical Functionals

Recall from Chapter 5 that a **statistical functional** is simply any function of the DF F. For example, the median $T(F) = F^{[-1]}(1/2)$ is a statistical functional. Thus, $T(F) : \{\text{All DFs }\} \mapsto \mathbb{T}$, being a map or function from the space of DFs to its range $\mathbb{T}$, is a functional. The idea behind the plug-in estimator for a statistical functional is simple: just plug-in the point estimate $\widehat{F}_n$ instead of the unknown DF F^* to estimate the statistical functional of interest.

Definition 48 (Plug-in Estimator) *Suppose, $X_1, \ldots, X_n \overset{IID}{\sim} F^*$. The plug-in estimator of a statistical functional of interest, namely, $T(F^*)$, is defined by:*

$$\widehat{T}_n := \widehat{T}_n(X_1, \ldots, X_n) = T(\widehat{F}_n) \ .$$

Definition 49 (Linear Functional) *If $T(F) = \int r(x)dF(x)$ for some function $r(x) : \mathbb{X} \mapsto \mathbb{R}$, then T is called a* **linear functional***. Thus, T is linear in its arguments:*

$$T(aF + a'F') = aT(F) + a'T(F') \ .$$

Proposition 18 *The plug-in estimator for a linear functional $T = \int r(x)dF(x)$ is:*

$$\boxed{T(\widehat{F}_n) = \int r(x)d\widehat{F}_n(x) = \frac{1}{n}\sum_{i=1}^{n} r(X_i)} \ .$$

Some specific examples of statistical linear functionals we have already seen include:

1. The **mean** of RV $X \sim F$ is a function of the DF F:
$$T(F) = \mathbf{E}(X) = \int x \, dF(x) \ .$$
2. The **variance** of RV $X \sim F$ is a function of the DF F:
$$T(F) = \mathbf{E}(X - \mathbf{E}(X))^2 = \int (x - \mathbf{E}(X))^2 \, dF(x) \ .$$
3. The **value of DF at a given** $x \in \mathbb{R}$ of RV $X \sim F$ is also a function of DF F:
$$T(F) = F(x) \ .$$
4. The q^{th} **quantile** of RV $X \sim F$:
$$T(F) = F^{[-1]}(q) \ \text{ where } q \in [0, 1] \ .$$
5. The **first quartile** or the 0.25^{th} **quantile** of the RV $X \sim F$:
$$T(F) = F^{[-1]}(0.25) \ .$$

6. The **median** or the **second quartile** or the 0.50^{th} **quantile** of the RV $X \sim F$:

$$T(F) = F^{[-1]}(0.50) \ .$$

7. The **third quartile** or the 0.75^{th} **quantile** of the RV $X \sim F$:

$$T(F) = F^{[-1]}(0.75) \ .$$

Labwork 41 (Plug-in Estimate for Median of Web Login Data) *Compute the plug-in estimates for the median for each of the data arrays:*

`WebLogSeconds20071001035730` *and* `WebLogSeconds20071002035730`

that can be loaded into memory by following the commands in the first 13 *lines of the script file* `WebLogDataProc.m` *of Data 1.*

Labwork 42 (Plug-in Estimates of Times Between Earth Quakes) *Compute the plug-in estimates for the median and mean time in minutes between earth quakes in NZ using the data in* `earthquakes.csv`.

NZSIEQTimesPlugInEstimates.m

```
%% The columns in earthquakes.csv file have the following headings
%%CUSP_ID,LAT,LONG,NZMGE,NZMGN,ORI_YEAR,ORI_MONTH,ORI_DAY,ORI_HOUR,ORI_MINUTE,ORI_SECOND,MAG,DEPTH
EQ=dlmread('earthquakes.csv',','); % load the data in the matrix EQ
% Read help datenum -- converts time stamps into numbers in units of days
MaxD=max(datenum(EQ(:,6:11)));% maximum datenum
MinD=min(datenum(EQ(:,6:11)));% minimum datenum
% get an array of sorted time stamps of EQ events starting at 0
Times=sort(datenum(EQ(:,6:11))-MinD);
TimeDiff=diff(Times); % inter-EQ times = times between successtive EQ events
n=length(TimeDiff); %sample size
PlugInMedianEstimate=median(TimeDiff) % plug-in estimate of median
PlugInMedianEstimateMinutes=PlugInMedianEstimate*24*60 % median estimate in minutes
PlugInMeanEstimate=mean(TimeDiff) % plug-in estimate of mean
PlugInMeanEstimateMinutes=PlugInMeanEstimate*24*60 % mean estimate in minutes
```

```
>> NZSIEQTimesPlugInEstimates
PlugInMedianEstimate =    0.0177
PlugInMedianEstimateMinutes =   25.5092
PlugInMeanEstimate =    0.0349
PlugInMeanEstimateMinutes =   50.2278
```

Note that any statistical functional can be estimated using the plug-in estimator. However, to produce a $1 - \alpha$ confidence set for the plug-in point estimate, we need bootstrap methods. The subject of next chapter.

Chapter 11

Bootstrap

The **bootstrap** is a statistical method for estimating standard errors and confidence sets of statistics, such as estimators.

11.1 Non-parametric Bootstrap for Confidence Sets

Let $T_n := T_n((X_1, X_2, \ldots, X_n))$ be a statistic, i.e. any function of the data $X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F^*$. Suppose we want to know its variance $\mathbf{V}_{F^*}(T_n)$, which clearly depends on the fixed and possibly unknown DF F^*.

If our statistic T_n is one with an analytically unknown variance, then we can use the bootstrap to estimate it. The bootstrap idea has the following two basic steps:

Step 1: Estimate $\mathbf{V}_{F^*}(T_n)$ with $\mathbf{V}_{\widehat{F}_n}(T_n)$.

Step 2: Approximate $\mathbf{V}_{\widehat{F}_n}(T_n)$ using simulated data from the "Bootstrap World."

For example, if $T_n = \overline{X}_n$, in Step 1, $\mathbf{V}_{\widehat{F}_n}(T_n) = s_n^2/n$, where $s_n^2 = n^{-1}\sum_{i=1}^n (x_i - \overline{x}_n)$ is the sample variance and $\overline{x}_n$ is the sample mean. In this case, Step 1 is enough. However, when the statistic T_n is more complicated (e.g. $T_n = \widetilde{X}_n = F^{[-1]}(0.5)$), the sample median, then we may not be able to find a simple expression for $\mathbf{V}_{\widehat{F}_n}(T_n)$ and may need Step 2 of the bootstrap.

$$\begin{array}{rccc} \text{Real World Data come from} & F^* & \implies X_1, X_2, \ldots, X_n & \implies T_n((X_1, X_2, \ldots, X_n)) = t_n \\ \text{Bootstrap World Data come from} & \widehat{F}_n & \implies X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet & \implies T_n((X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet)) = t_n^\bullet \end{array}$$

Observe that drawing an observation from the ECDF $\widehat{F}_n$ is equivalent to drawing one point at random from the original data (think of the indices $[n] := \{1, 2, \ldots, n\}$ of the original data $X_1, X_2, \ldots, X_n$ being drawn according to the equi-probable de Moivre$(1/n, 1/n, \ldots, 1/n)$ RV on $[n]$). Thus, to simulate $X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet$ from $\widehat{F}_n$, it is enough to drawn n observations with replacement from $X_1, X_2, \ldots, X_n$.

In summary, the algorithm for Bootstrap Variance Estimation is:

Step 1: Draw $X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet \sim \widehat{F}_n$

Step 2: Compute $t_n^\bullet = T_n((X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet))$

Step 3: Repeat Step 1 and Step 2 B times, for some large B, say $B > 1000$, to get $t^\bullet_{n,1}, t^\bullet_{n,2}, \ldots, t^\bullet_{n,B}$

Step 4: Several ways of estimating the bootstrap confidence intervals are possible:

(a) The $1-\alpha$ Normal-based bootstrap confidence interval is:

$$C_n = [T_n - z_{\alpha/2}\widehat{se}_{boot}, T_n + z_{\alpha/2}\widehat{se}_{boot}] \ ,$$

where the bootstrap-based standard error estimate is:

$$\widehat{se}_{boot} = \sqrt{v_{boot}} = \sqrt{\frac{1}{B}\sum_{b=1}^{B}\left(t^\bullet_{n,b} - \frac{1}{B}\sum_{r=1}^{B} t^\bullet_{n,r}\right)^2}$$

(b) The $1-\alpha$ percentile-based bootstrap confidence interval is:

$$C_n = [\widehat{G^\bullet}_n^{-1}(\alpha/2), \widehat{G^\bullet}_n^{-1}(1-\alpha/2)],$$

where $\widehat{G^\bullet}_n$ is the empirical DF of the bootstrapped $t^\bullet_{n,1}, t^\bullet_{n,2}, \ldots, t^\bullet_{n,B}$ and $\widehat{G^\bullet}_n^{-1}(q)$ is the q^{th} sample quantile (5.9) of $t^\bullet_{n,1}, t^\bullet_{n,2}, \ldots, t^\bullet_{n,B}$.

Labwork 43 (Confidence Interval for Median Estimate of Inter Earth Quake Times) *Let us find the* 95% *Normal-based bootstrap confidence interval as well as the* 95% *percentile-based bootstrap confidence interval for our plug-in estimate of the median of inter earth quake times from Labwork 42 using the following script:*

NZSIEQTimesMedianBootstrap.m

```
%% The columns in earthquakes.csv file have the following headings
%%CUSP_ID,LAT,LONG,NZMGE,NZMGN,ORI_YEAR,ORI_MONTH,ORI_DAY,ORI_HOUR,ORI_MINUTE,ORI_SECOND,MAG,DEPTH
EQ=dlmread('earthquakes.csv',','); % load the data in the matrix EQ
% Read help datenum -- converts time stamps into numbers in units of days
MaxD=max(datenum(EQ(:,6:11)));% maximum datenum
MinD=min(datenum(EQ(:,6:11)));% minimum datenum
% get an array of sorted time stamps of EQ events starting at 0
Times=sort(datenum(EQ(:,6:11))-MinD);
TimeDiff=diff(Times); % inter-EQ times = times between successtive EQ events
n=length(TimeDiff) %sample size
Medianhat=median(TimeDiff)*24*60 % plug-in estimate of median in minutes
B= 1000 % Number of Bootstrap replications
% REPEAT B times: PROCEDURE of sampling n indices uniformly from 1,...,n with replacement
BootstrappedDataSet = TimeDiff([ceil(n*rand(n,B))]);
size(BootstrappedDataSet) % dimension of the BootstrappedDataSet
BootstrappedMedians=median(BootstrappedDataSet)*24*60; % get the statistic in Bootstrap world
% 95% Normal based Confidence Interval
SehatBoot = std(BootstrappedMedians); % std of BootstrappedMedians
% 95% C.I. for median from Normal approximation
ConfInt95BootNormal = [Medianhat-1.96*SehatBoot, Medianhat+1.96*SehatBoot]
% 95% Percentile based Confidence Interval
ConfInt95BootPercentile = ...
    [qthSampleQuantile(0.025,sort(BootstrappedMedians)),...
    qthSampleQuantile(0.975,sort(BootstrappedMedians))]
```

We get the following output when we call the script file.

```
>> NZSIEQTimesMedianBootstrap
n =        6127
Medianhat =   25.5092
B =        1000
ans =        6127        1000
ConfInt95BootNormal =   24.4383   26.5800
ConfInt95BootPercentile =   24.4057   26.4742
```

Labwork 44 (Confidence Interval for Median Estimate of Web Login Data) *Find the* 95% *Normal-based bootstrap confidence interval as well as the* 95% *percentile-based bootstrap confidence interval for our plug-in estimate of the median for each of the data arrays:*

`WebLogSeconds20071001035730` *and* `WebLogSeconds20071002035730` .

Once again, the arrays can be loaded into memory by following the commands in the first 13 *lines of the script file* `WebLogDataProc.m` *of Section 1. Produce four intervals (two for each data-set). Do the confidence intervals for the medians for the two days intersect?*

```
>> WebLogDataProc % load in the data
>> Medianhat = median(WebLogSeconds20071001035730) % plug-in estimate of median
Medianhat =       37416
>> % store the length of data array
>> K=length(WebLogSeconds20071001035730)
K =       56485
>> B= 1000 % Number of Bootstrap replications
B =        1000
>> BootstrappedDataSet = WebLogSeconds20071001035730([ceil(K*rand(K,B))]);
>> size(BootstrappedDataSet) % dimension of the BootstrappedDataSet
ans =       56485        1000
>> BootstrappedMedians=median(BootstrappedDataSet); % get the statistic in Bootstrap world
>> % 95% Normal based Confidence Interval
>> SehatBoot = std(BootstrappedMedians); % std of BootstrappedMedians
>> % 95% C.I. for median from Normal approximation
>> ConfInt95BootNormal = [Medianhat-1.96*SehatBoot, Medianhat+1.96*SehatBoot]
ConfInt95BootNormal =    37242    37590
>> % 95% Percentile based Confidence Interval
ConfInt95BootPercentile = ...
    [qthSampleQuantile(0.025,sort(BootstrappedMedians)),...
    qthSampleQuantile(0.975,sort(BootstrappedMedians))]
ConfInt95BootPercentile =   37239    37554
```

Labwork 45 (Confidence interval for correlation) *Here is a classical data set used by Bradley Efron (the inventor of bootstrap) to illustrate the method. The data are LSAT (Law School Admission Test in the U.S.A.) scores and GPA of fifteen individuals.*

Thus, we have bivariate data of the form (Y_i, Z_i)*, where* $Y_i = \text{LSAT}_i$ *and* $Z_i = \text{GPA}_i$*. For example, the first individual had an LSAT score of* $y_1 = 576$ *and a GPA of* $z_1 = 3.39$ *while the fifteenth individual had an LSAT score of* $y_{15} = 594$ *and a GPA of* $z_{15} = 3.96$*. We supose that the bivariate data* $(Y_i, Z_i) \overset{IID}{\sim} F^*$*, such that* $F^* \in \{\text{all bivariate DFs}\}$*. This is a bivariate non-parametric experiment. The bivariate data are plotted in Figure .*

The law school is interested in the correlation between the GPA and LSAT scores:

$$\theta^* = \frac{\int\int (y - \mathbf{E}(Y))(z - \mathbf{E}(Z)) dF(y,z)}{\sqrt{\int (y - \mathbf{E}(Y))^2 dF(y) \int (z - \mathbf{E}(Z))^2 dF(z)}}$$

The plug-in estimate of the population correlation θ^* *is the sample correlation:*

$$\widehat{\Theta}_n = \frac{\sum_{i=1}^n (Y_i - \overline{Y}_n)(Z_i - \overline{Z}_n)}{\sqrt{\sum_{i=1}^n (Y_i - \overline{Y}_n)^2 \sum_{i=1}^n (Z_i - \overline{Z}_n)^2}}$$

```
%% Data from Bradley Efron's LSAT,GPA correlation estimation
LSAT=[576 635 558 578 666 580 555 661 651 605 653 575 545 572 594]; % LSAT data
```

```
GPA=[3.39 3.30 2.81 3.03 3.44 3.07 3.00 3.43 3.36 3.13 3.12 2.74 2.76 2.88 3.96]; % GPA data
subplot(1,2,1); plot(LSAT,GPA,'o'); xlabel('LSAT'); ylabel('GPA')  % make a plot of the data
CC=corrcoef(LSAT,GPA); % use built-in function to compute sample correlation coefficient matrix
SampleCorrelation=CC(1,2) % plug-in estimate of the correlation coefficient
%% Bootstrap
B = 1000; % Number of Bootstrap replications
BootstrappedCCs=zeros(1,B); % initialise a vector of zeros
N = length(LSAT); % sample size
rand('twister',767671); % initialise the fundamental sampler
for b=1:B
    Indices=ceil(N*rand(N,1));% uniformly sample random indices from 1 to 15 with replacement
    BootstrappedLSAT = LSAT([Indices]); % bootstrapped LSAT data
    BootstrappedGPA = GPA([Indices]); % bootstrapped GPA data
    CCB=corrcoef(BootstrappedLSAT,BootstrappedGPA);
    BootstrappedCCs(b)=CCB(1,2); % sample correlation of bootstrapped data
end
%plot the histogram ofBootstrapped Sample Correlations with 15 bins
subplot(1,2,2);hist(BootstrappedCCs,15);xlabel('Bootstrapped Sample Correlations')

% 95% Normal based Confidence Interval
SehatBoot = std(BootstrappedCCs); % std of BootstrappedMedians
% 95% C.I. for median from Normal approximation
ConfInt95BootNormal = [SampleCorrelation-1.96*SehatBoot, SampleCorrelation+1.96*SehatBoot]
% 95% Percentile based Confidence Interval
ConfInt95BootPercentile = ...
    [qthSampleQuantile(0.025,sort(BootstrappedCCs)),...
    qthSampleQuantile(0.975,sort(BootstrappedCCs))]
```

We get the following output when we call the script file.

```
>> LSATGPACorrBootstap
SampleCorrelation =    0.5459
ConfInt95BootNormal =    0.1770    0.9148
ConfInt95BootPercentile =    0.2346    0.9296
```

11.2 Parametric Bootstrap for Confidence Sets

The **bootstrap** may also be employed for estimating standard errors and confidence sets of statistics, such as estimators, even in a parametric setting. This is much easier than the the variance calculation based on Fisher Information and/or the Delta method.

The only difference in the **parametric bootstrap** as opposed to the **non-parametric bootstrap** we saw earlier is that our statistic of interest $T_n := T_n((X_1, X_2, \ldots, X_n))$ is a function of the data:

$$X_1, X_2, \ldots, X_n \overset{\text{IID}}{\sim} F(x; \theta^*) \ .$$

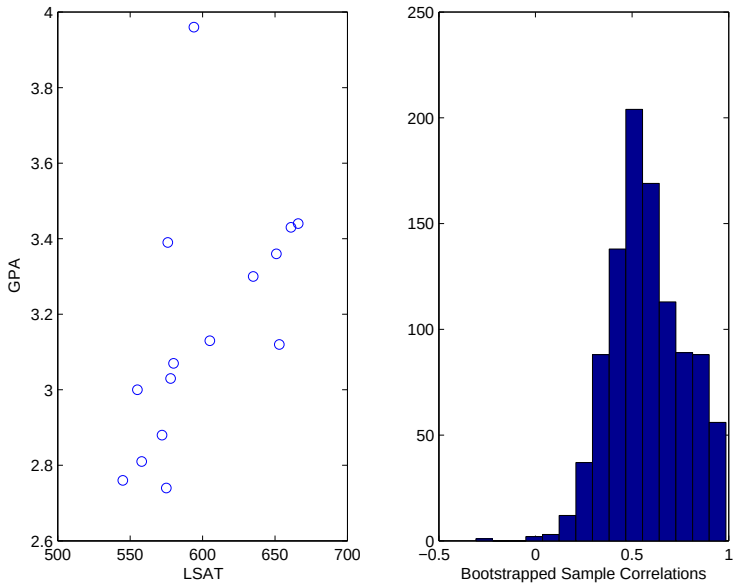
That is, our data come from a parametric distribution $F(x; \theta^*)$ and we want to know the variance of our statistic T_n, i.e. $\mathbf{V}_{\theta^*}(T_n)$.

The parametric bootstrap concept has the following two basic steps:

Step 1: Estimate $\mathbf{V}_{\theta^*}(T_n)$ with $\mathbf{V}_{\widehat{\theta}_n}(T_n)$, where $\widehat{\theta}_n$ is an estimate of θ^* based on maximum likelihood or the method of moments.

Step 2: Approximate $\mathbf{V}_{\widehat{\theta}_n}(T_n)$ using simulated data from the "Bootstrap World."

Figure 11.1: Data from Bradley Efrons LSAT and GPA scores for fifteen individuals (left). The confidence interval of the sample correlation, the plug-in estimate of the population correlation, is obtained from the sample correlation of one thousand bootstrapped data sets (right).



For example, if $T_n = \overline{X}_n$, the sample mean, then in Step 1, $\mathbf{V}_{\widehat{\theta}_n}(T_n) = n^{-1}\sum_{i=1}^{n}(x_i - \overline{x}_n)$ is the sample variance. Thus, in this case, Step 1 is enough. However, when the statistic T_n is more complicated, say $T_n = \widetilde{X}_n = F^{[-1]}(0.5)$, the sample median, then we may not be able to write down a simple expression for $\mathbf{V}_{\widehat{\theta}_n}(T_n)$ and may need Step 2 of the bootstrap.

$$\begin{aligned} \text{Real World Data come from} \quad & F(\theta^*) \implies X_1, X_2, \ldots, X_n \implies T_n((X_1, X_2, \ldots, X_n)) = t_n \\ \text{Bootstrap World Data come from} \quad & F(\widehat{\theta}_n) \implies X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet \implies T_n((X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet)) = t_n^\bullet \end{aligned}$$

To simulate $X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet$ from $F(\widehat{\theta}_n)$, we must have a simulation algorithm that allows us to draw IID samples from $F(\theta)$, for instance the inversion sampler. In summary, the algorithm for Bootstrap Variance Estimation is:

Step 1: Draw $X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet \sim F(\widehat{\theta}_n)$

Step 2: Compute $t_n^\bullet = T_n((X_1^\bullet, X_2^\bullet, \ldots, X_n^\bullet))$

Step 3: Repeat Step 1 and Step 2 B times, for some large B, say $B \geq 1000$, to get $t_{n,1}^\bullet, t_{n,2}^\bullet, \ldots, t_{n,B}^\bullet$

Step 4: We can estimate the bootstrap confidence intervals in several ways:

(a) The $1-\alpha$ normal-based bootstrap confidence interval is:

$$C_n = [T_n - z_{\alpha/2}\widehat{se}_{boot}, T_n + z_{\alpha/2}\widehat{se}_{boot}] \ ,$$

where the bootstrap-based standard error estimate is:

$$\widehat{se}_{boot} = \sqrt{v_{boot}} = \sqrt{\frac{1}{B}\sum_{b=1}^{B}\left(t_{n,b}^\bullet - \frac{1}{B}\sum_{r=1}^{B} t_{n,r}^\bullet\right)^2}$$

(b) The $1-\alpha$ percentile-based bootstrap confidence interval:

$$C_n = [\widehat{G^\bullet}_n^{-1}(\alpha/2), \widehat{G^\bullet}_n^{-1}(1-\alpha/2)],$$

where $\widehat{G^\bullet}_n$ is the empirical DF of the bootstrapped $t^\bullet_{n,1}, t^\bullet_{n,2}, \ldots, t^\bullet_{n,B}$ and $\widehat{G^\bullet}_n^{-1}(q)$ is the q^{th} sample quantile (5.9) of $t^\bullet_{n,1}, t^\bullet_{n,2}, \ldots, t^\bullet_{n,B}$.

Let us apply the bootstrap method to the previous problem of estimating the standard error of the coefficient of variation from $n = 100$ samples from Normal$(100, 10^2)$ RV. The confidence intervals from bootstrap-based methods are similar to those from the Delta method.

CoeffOfVarNormalBoot.m

```
n=100; Mustar=100; Sigmastar=10; % sample size, true mean and standard deviation
rand('twister',67345);
x=arrayfun(@(u)(Sample1NormalByNewRap(u,Mustar,Sigmastar^2)),rand(n,1)); % normal samples
Muhat=mean(x) Sigmahat=std(x) Psihat=Sigmahat/Muhat % MLE of Mustar, Sigmastar and Psistar
Sehat = sqrt((1/Muhat^4)+(Sigmahat^2/(2*Muhat^2)))/sqrt(n) % standard error estimate
% 95% Confidence interval by Delta Method
ConfInt95DeltaMethod=[Psihat-1.96*Sehat, Psihat+1.96*Sehat] % 1.96 since 1-alpha=0.95
B = 1000; % B is number of bootstrap replications
% Step 1: draw n IID samples in Bootstrap World from Normal(Muhat,Sigmahat^2)
xBoot = arrayfun(@(u)(Sample1NormalByNewRap(u,Muhat,Sigmahat^2)),rand(n,B));
% Step 2: % Compute Bootstrapped Statistic Psihat
PsihatBoot = std(xBoot) ./ mean(xBoot);
% 95% Normal based Confidence Interval
SehatBoot = std(PsihatBoot); % std of PsihatBoot
ConfInt95BootNormal = [Psihat-1.96*SehatBoot, Psihat+1.96*SehatBoot] % 1-alpha=0.95
% 95% Percentile based Confidence Interval
ConfInt95BootPercentile = ...
    [qthSampleQuantile(0.025,sort(PsihatBoot)),qthSampleQuantile(0.975,sort(PsihatBoot))]
```

```
>> CoeffOfVarNormal
Muhat =  100.3117
Sigmahat =   10.9800
Psihat =    0.1095
Sehat =    0.0077
ConfInt95DeltaMethod =    0.0943    0.1246
ConfInt95BootNormal =    0.0943    0.1246
ConfInt95BootPercentile =    0.0946    0.1249
```

Chapter 12

Hypothesis Testing

The subset of **all posable hypotheses** that remain **falsifiable** is the space of **scientific hypotheses**. Roughly, a falsifiable hypothesis is one for which a statistical experiment can be designed to produce data that an experimenter can use to falsify or reject it. In the statistical decision problem of hypothesis testing, we are interested in empirically falsifying a scientific hypothesis, i.e. we attempt to reject an hypothesis on the basis of empirical observations or data. Thus, hypothesis testing has its roots in the philosophy of science and is based on Karl Popper's falsifiability criterion for demarcating scientific hypotheses from the set of all posable hypotheses.

12.1 Introduction

Usually, the hypothesis we attempt to reject or falsify is called the **null hypothesis** or H_0 and its complement is called the **alternative hypothesis** or H_1. For example, consider the following two hypotheses:

H_0: The average waiting time at an Orbiter bus stop is less than or equal to 10 minutes.
H_1: The average waiting time at an Orbiter bus stop is more than 10 minutes.

If the sample mean $\overline{x}_n$ is much larger than 10 minutes then we may be inclined to reject the null hypothesis that the average waiting time is less than or equal to 10 minutes. We will learn to formally test hypotheses in the sequel.

Suppose we are interested in the following hypothesis test for the bus-stop problem:

H_0: The average waiting time at an Orbiter bus stop is equal to 10 minutes.
H_1: The average waiting time at an Orbiter bus stop is not 10 minutes.

Once again we can use the sample mean as the test statistic. Our procedure for rejecting this null hypothesis is different and is often called the Wald test.

More generally, suppose $X_1, X_2, \ldots, X_n \overset{IID}{\sim} F(x_1; \theta^*)$, with an unknown and fixed $\theta^* \in \mathbf{\Theta}$. Let us partition the parameter space $\mathbf{\Theta}$ into $\mathbf{\Theta}_0$, the null parameter space, and $\mathbf{\Theta}_1$, the alternative parameter space, ie,

$$\mathbf{\Theta}_0 \cup \mathbf{\Theta}_1 = \mathbf{\Theta}, \qquad \text{and} \qquad \mathbf{\Theta}_0 \cap \mathbf{\Theta}_1 = \emptyset \ .$$

Then, we can formalise testing the null hypothesis versus the alternative as follows:

$$H_0 : \theta^* \in \mathbf{\Theta}_0 \qquad \text{versus} \qquad H_1 : \theta^* \subset \mathbf{\Theta}_1 \ .$$

The basic idea involves finding an appropriate rejection region $\mathbb{X}_R$ within the data space $\mathbb{X}$ and rejecting H_0 if the observed data $x := (x_1, x_2, \ldots, x_n)$ falls inside the rejection region $\mathbb{X}_R$,

$$\text{If } x := (x_1, x_2, \ldots, x_n) \in \mathbb{X}_R \subset \mathbb{X}, \text{ then reject } H_0, \text{ else do not reject } H_0.$$

Typically, the rejection region $\mathbb{X}_R$ is of the form:

$$\mathbb{X}_R := \{x := (x_1, x_2, \ldots, x_n) : T(x) > c\}$$

where, T is the **test statistic** and c is the **critical value**. Thus, the problem of finding $\mathbb{X}_R$ boils down to that of finding T and c that are appropriate. Once the rejection region is defined, the possible outcomes of a hypothesis test are summarised in the following table.

Table 12.1: Outcomes of an hypothesis test.

	Do not Reject H_0	Reject H_0
H_0 is True	OK	Type I Error
H_1 is True	Type II Error	OK

Definition 50 (Power, Size and Level of a Test) *The* **power function** *of a test with rejection region $\mathbb{X}_R$ is*

$$\beta(\theta) := \mathbf{P}_\theta(x \in \mathbb{X}_R) \ . \tag{12.1}$$

So $\beta(\theta)$ is the power of the test at the parameter value θ, i.e. the probability that the observed data x, sampled from the distribution specified by θ, falls in $\mathbb{X}_R$ and thereby leads to a rejection of the null hypothesis.

The size *of a test with rejection region $\mathbb{X}_R$ is the supreme power under the null hypothesis, i.e. the supreme probability of rejecting the null hypothesis when the null hypothesis is true:*

$$\text{size} := \sup_{\theta \in \Theta_0} \beta(\theta) := \sup_{\theta \in \Theta_0} \mathbf{P}_\theta(x \in \mathbb{X}_R) \ . \tag{12.2}$$

The size *of a test is often denoted by α. A test is said to have* level α *if its* size *is less than or equal to α.*

Let us familiarize ourselves with some terminology in hypothesis testing next.

Table 12.2: Some terminology in hypothesis testing.

Θ	Test: H_0 versus H_1	Nomenclature
$\Theta \subset \mathbb{R}^m, m \geq 1$	$H_0 : \theta^* = \theta_0$ versus $H_1 : \theta^* \neq \theta_1$	Simple Hypothesis Test
$\Theta \subset \mathbb{R}^m, m \geq 1$	$H_0 : \theta^* \in \Theta_0$ versus $H_1 : \theta^* \in \Theta_1$	Composite Hypothesis Test
$\Theta \subset \mathbb{R}^1$	$H_0 : \theta^* = \theta_0$ versus $H_1 : \theta^* \neq \theta_0$	Two-sided Hypothesis Test
$\Theta \subset \mathbb{R}^1$	$H_0 : \theta^* \geq \theta_0$ versus $H_1 : \theta^* < \theta_0$	One-sided Hypothesis Test
$\Theta \subset \mathbb{R}^1$	$H_0 : \theta^* \leq \theta_0$ versus $H_1 : \theta^* > \theta_0$	One-sided Hypothesis Test

We introduce some widely used tests next.

12.2 The Wald Test

The Wald test is based on a direct relationship between the $1 - \alpha$ confidence interval and a size α test. It can be used for testing simple hypotheses involving a scalar parameter.

Definition 51 (The Wald Test) *Let $\widehat{\Theta}_n$ be an asymptotically normal estimator of the fixed and possibly unknown parameter $\theta^* \in \mathbb{\Theta} \subset \mathbb{R}$ in the parametric IID experiment:*

$$X_1, X_2, \ldots, X_n \overset{IID}{\sim} F(x_1; \theta^*) \ .$$

Consider testing:

$$H_0 : \theta^* = \theta_0 \qquad versus \qquad H_1 : \theta^* \neq \theta_0 \ .$$

Suppose that the null hypothesis is true and the estimator $\widehat{\Theta}_n$ of $\theta^* = \theta_0$ is asymptotically normal:

$$\theta^* = \theta_0, \qquad \frac{\widehat{\Theta}_n - \theta_0}{\widehat{\mathsf{se}}_n} \rightsquigarrow \text{Normal}(0,1) \ .$$

Then, the Wald test based on the test statistic W is:

$$\boxed{\text{Reject } H_0 \text{ when } |W| > z_{\alpha/2}, \text{ where } W := W((X_1, \ldots, X_n)) = \frac{\widehat{\Theta}_n((X_1, \ldots, X_n)) - \theta_0}{\widehat{\mathsf{se}}_n}.}$$

The rejection region for the Wald test is:

$$\boxed{\mathbb{X}_R = \{x := (x_1, \ldots, x_n) : |W(x_1, \ldots, x_n)| > z_{\alpha/2}\} \ .}$$

Proposition 19 *As the sample size n approaches infinity, the* size *of the Wald test approaches α :*

$$\boxed{\mathsf{size} = \mathbf{P}_{\theta_0}\left(|W| > z_{\alpha/2}\right) \to \alpha \ .}$$

Proof: Let $Z \sim \text{Normal}(0,1)$. The size of the Wald test, i.e. the supreme power under H_0 is:

$$\begin{aligned}
\mathsf{size} &:= \sup_{\theta \in \mathbb{\Theta}_0} \beta(\theta) := \sup_{\theta \in \{\theta_0\}} \mathbf{P}_\theta(x \in \mathbb{X}_R) = \mathbf{P}_{\theta_0}(x \in \mathbb{X}_R) \\
&= \mathbf{P}_{\theta_0}\left(|W| > z_{\alpha/2}\right) = \mathbf{P}_{\theta_0}\left(\frac{|\widehat{\theta}_n - \theta_0|}{\widehat{\mathsf{se}}_n} > z_{\alpha/2}\right) \\
&\to \mathbf{P}\left(|Z| > z_{\alpha/2}\right) \\
&= \alpha \ .
\end{aligned}$$

Next, let us look at the power of the Wald test when the null hypothesis is false.

Proposition 20 *Suppose $\theta^* \neq \theta_0$. The power $\beta(\theta^*)$, which is the probability of correctly rejecting the null hypothesis, is approximately equal to:*

$$\boxed{\Phi\left(\frac{\theta_0 - \theta^*}{\widehat{\mathsf{se}}_n} - z_{\alpha/2}\right) + \left(1 - \Phi\left(\frac{\theta_0 - \theta^*}{\widehat{\mathsf{se}}_n} + z_{\alpha/2}\right)\right) \ ,}$$

where, Φ is the DF of Normal$(0,1)$ *RV. Since $\widehat{\mathsf{se}}_n \to 0$ as $n \to 0$ the power increase with sample* size *n. Also, the power increases when $|\theta_0 - \theta^*|$ is large.*

Now, let us make the connection between the size α Wald test and the $1 - \alpha$ confidence interval explicit.

Proposition 21 *The* size α *Wald test rejects:*

$$H_0 : \theta^* = \theta_0 \text{ versus } H_1 : \theta^* \neq \theta_0 \text{ if and only if } \theta_0 \notin C_n := (\widehat{\theta}_n - \widehat{\mathsf{se}}_n z_{\alpha/2}, \widehat{\theta}_n + \widehat{\mathsf{se}}_n z_{\alpha/2}).$$

Therefore, testing the hypothesis is equivalent to verifying whether the null value θ_0 is in the confidence interval.

Example 36 *Let us use the Wald test to attempt to reject the null hypothesis that the mean waiting time at our Orbiter bus-stop is* 10 *minutes under an IID* Exponential(λ^*) *model. Let $\alpha = 0.05$ for this test. We can formulate this test as follows:*

$$H_0 : \lambda^* = \lambda_0 = \frac{1}{10} \quad \text{versus} \quad H_1 : \lambda^* \neq \frac{1}{10}, \quad \text{where,} \quad X_1 \ldots, X_{132} \overset{IID}{\sim} \text{Exponential}(\lambda^*) \ .$$

Based on Example 32 and Labwork 36 we obtained the 95% *confidence interval to be* $[0.0914, 0.1290]$. *Since our null value $\lambda_0 = 0.1$ belongs to this confidence interval, we fail to reject the null hypothesis from a* size $\alpha = 0.05$ *Wald test.*

We can use bootstrap-based confidence interval C_n in conjunction with Wald test as shown by the next example.

Example 37 *Recall the problem of estimating the confidence interval for the correlation coefficient between the LSAT scores $(Y_1, \ldots, Y_{15})$ and the GPA $(Z_1, \ldots, Z_{15})$ in Labwork 45. We assumed that the bivariate data $(Y_i, Z_i) \overset{IID}{\sim} F^*$, such that $F^* \in \{$all bivariate DFs$\}$. Suppose we are interested in testing the null hypothesis that the true correlation coefficient θ^* is* 0*:*

$$H_0 : \theta^* = \theta_0 = 0 \quad \text{versus} \quad H_1 : \theta^* \neq 0, \quad \text{where} \quad \theta^* = \frac{\int\int (y - \mathbf{E}(Y))(z - \mathbf{E}(Z)) dF(y, z)}{\sqrt{\int (y - \mathbf{E}(Y))^2 dF(y) \int (z - \mathbf{E}(Z))^2 dF(z)}} \ .$$

Since the percentile-based 95% *bootstrap confidence interval for the plug-in estimate of the correlation coefficient from Labwork 45 was* $[0.2346, 0.9296]$ *and this interval does not contain* 0*, we can reject the null hypothesis that the correlation coefficient is* 0 *using a* size $\alpha = 0.05$ *Wald test.*

12.3 A Composite Hypothesis Test

Often, we are interested in a testing a composite hypothesis, i.e. one in which the null hypothesis is not a singleton set. We revisit the Orbiter waiting time problem from this perspective next.

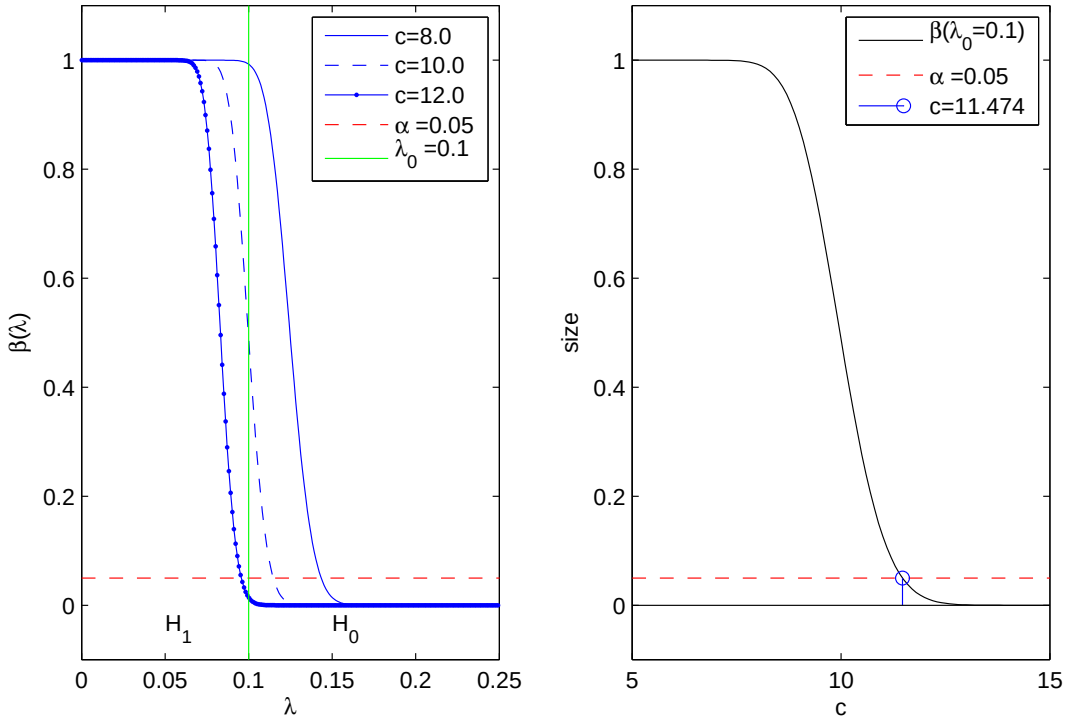
Example 38 (Testing the Mean Waiting Time at an Orbiter Bus-stop) *Let us test the following null hypothesis H_0.*

H_0: The average waiting time at an Orbiter bus stop is less than or equal to 10 *minutes.*
H_1: The average waiting time at an Orbiter bus stop is more than 10 *minutes.*

We have observations of $n = 132$ waiting times $x_1, x_2, \ldots, x_{132}$ at the Orbter bus-stop with $\overline{x}_{132} =$ 9.0758*. Let us assume a parametric model, say,*

$$X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{Exponential}(\lambda^*)$$

Figure 12.1: Plot of power function $\beta(\lambda)$ for different values of the critical value c and the size α as function of the critical values.



with an unknown and fixed $\lambda^* \in \mathbf{\Lambda} = (0, \infty)$. *Since the parameter* λ *of an* Exponential(λ) *RV is the reciprocal of the mean waiting time, we can formalise the above hypothesis testing problem of* H_0 *versus* H_1 *as follows:*

$$H_0 : \lambda^* \in \mathbf{\Lambda}_0 = [1/10, \infty) \qquad \textit{versus} \qquad H_1 : \lambda^* \in \mathbf{\Lambda}_1 = (0, 1/10)$$

Consider the test:

$$\textit{Reject } H_0 \textit{ if } T > c.$$

where the test statistic $T = \overline{X}_n$ *and the rejection region is:*

$$\mathbb{X}_R = \{(x_1, x_2, \ldots, x_n) : T(x_1, x_2, \ldots, x_n) > c\} \ .$$

Since the sum of n *IID* Exponential(λ) *RVs is* Gamma(λ, n) *distributed, the power function is:*

$$\begin{aligned}
\beta(\lambda) &= \mathbf{P}_\lambda\left(\overline{X}_n > c\right) = \mathbf{P}_\lambda\left(\sum_{i=1}^{n} X_i > nc\right) = 1 - \mathbf{P}_\lambda\left(\sum_{i=1}^{n} X_i \leq nc\right) \\
&= 1 - F\left(nc; \lambda, n\right) = 1 - \frac{1}{\Gamma(n)} \int_0^{\lambda nc} y^{n-1} \exp(-y) dy \\
&= 1 - \texttt{gammainc}(\lambda nc, n)
\end{aligned}$$

Clearly, $\beta(\lambda)$ *is a decreasing function of* λ *as shown in Figure 12.1. Hence the* size *of the test as a function of the critical region specified by the critical value* c *is:*

$$\text{size} = \sup_{\lambda \in \mathbf{\Lambda}_0} \beta(\lambda) = \sup_{\lambda \geq 1/10} \beta(\lambda) = \beta(1/10) = 1 - \texttt{gammainc}(132c/10, 132)$$

For a size $\alpha = 0.05$ *test we numerically solve for the critical value* c *that satisfies:*

$$0.05 = 1 - \texttt{gammainc}(132c/10, 132)$$

by trial and error as follows:

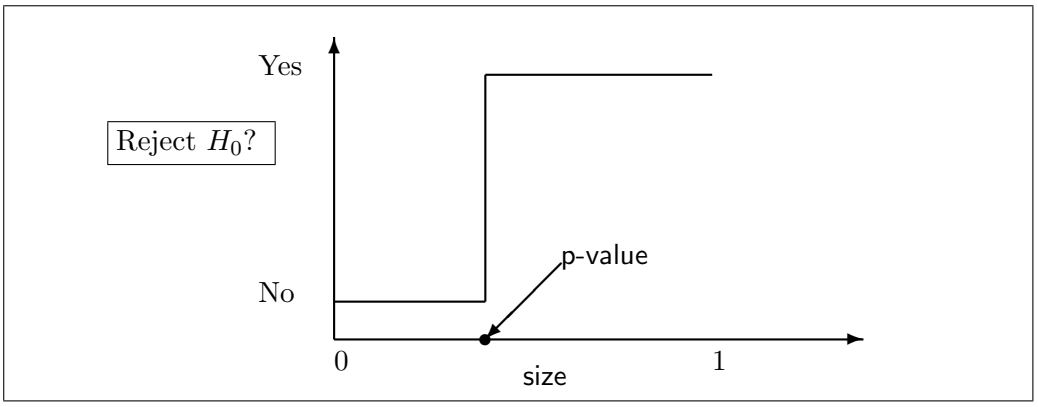
```
>> lambda0=1/10
lambda0 =    0.1000
>> S=@(C)(1-gammainc(lambda0*n*C,n)); % size as a function of c
>> Cs=[10 11 11.474 12 13] % some critical values c
Cs =   10.0000   11.0000   11.4740   12.0000   13.0000
>> Size=arrayfun(S,Cs) % corresponding size
Size =    0.4884    0.1268    0.0499    0.0143    0.0007
```

Thus, we reject H_0 *when* $\overline{X}_n > 11.4740$ *for a level* $\alpha = 0.05$ *test. Since our observed test statistic* $\overline{x}_{132} = 9.0758 < 11.4740$ *we fail to reject the null hypothesis that the mean waiting time is less than or equal to* 10 *minutes. Therefore, there is no evidence that the Orbiter bus company is violating its promise of an average waiting time of no more than* 10 *minutes.*

12.4 p-values

It is desirable to have a more informative decision than simply reporting "reject H_0" or "fail to reject H_0." For instance, we could ask whether the test rejects H_0 for each size $= \alpha$. Typically, if the test rejects at size α it will also reject at a larger size $\alpha' > \alpha$. Therefore, there is a smallest size α at which the test rejects H_0 and we call this α the p-value of the test.

Figure 12.2: The smallest α at which a size α test rejects the null hypothesis H_0 is the p-value.



Definition 52 (p-value) *Suppose that for every* $\alpha \in (0, 1)$ *we have a* size α *test with rejection region* $\mathbb{X}_{R,\alpha}$ *and test statistic* T*. Then,*

$$\text{p-value} := \inf\{\alpha : T(X) \in \mathbb{X}_{R,\alpha}\} \ .$$

That is, the p-value *is the smallest* α *at which a* size α *test rejects the null hypothesis.*

If the evidence against H_0 is strong then the p-value will be small. However, a large p-value is not strong evidence in favour of H_0. This is because a large p-value can occur for two reasons:

Table 12.3: Evidence scale against the null hypothesis in terms of the range of p-value.

p-value range	Evidence
$(0, 0.01]$	very strong evidence against H_0
$(0.01, 0.05]$	strong evidence against H_0
$(0.05, 0.1]$	weak evidence against H_0
$(0.1, 1)$	little or no evidence against H_0

1. H_0 is true.
2. H_0 is false but the test has low power.

Finally, it is important to realise that p-value is not the probability that the null hypothesis is true, i.e. p-value $\neq \mathbf{P}(H_0|x)$, where x is the data. The following tabulation of evidence scale is useful. The next proposition gives a convenient expression for the p-value for certain tests.

Proposition 22 *Suppose that the* size α *test based on the test statistic T and critical value c_α is of the form:*

$$\textit{Reject } H_0 \textit{ if and only if } T := T((X_1, \ldots, X_n)) > c_\alpha,$$

then

$$\text{p-value} = \sup_{\theta \in \mathbf{\Theta}_0} \mathbf{P}_\theta(T((X_1, \ldots, X_n)) \geq t := T((x_1, \ldots, x_n))) \ ,$$

where, $(x_1, \ldots, x_n)$ is the observed data and t is the observed value of the test statistic T. In words, the p-value *is the supreme probability under H_0 of observing a value of the test statistic the same as or more extreme than what was actually observed.*

Let us revisit the Orbiter waiting times example from the p-value perspective.

Example 39 (p-value for the parametric Orbiter experiment) *Let the waiting times at our bus-stop be $X_1, X_2, \ldots, X_{132} \overset{IID}{\sim}$ Exponential(λ^*). Consider the following testing problem:*

$$H_0 : \lambda^* = \lambda_0 = \frac{1}{10} \quad \textit{versus} \quad H_1 : \lambda^* \neq \lambda_0 \ .$$

We already saw that the Wald test statistic is:

$$W := W(X_1, \ldots, X_n) = \frac{\widehat{\Lambda}_n - \lambda_0}{\widehat{\mathsf{se}}_n(\widehat{\Lambda}_n)} = \frac{\frac{1}{\overline{X}_n} - \lambda_0}{\frac{1}{\sqrt{n}\overline{X}_n}} \ .$$

The observed test statistic is:

$$w = W(x_1, \ldots, x_{132}) = \frac{\frac{1}{\overline{X}_{132}} - \lambda_0}{\frac{1}{\sqrt{132}\overline{X}_{132}}} = \frac{\frac{1}{9.0758} - \frac{1}{10}}{\frac{1}{\sqrt{132} \times 9.0758}} = 1.0618 \ .$$

Since, $W \rightsquigarrow Z \sim$ Normal$(0, 1)$, the p-value *for this Wald test is:*

$$\begin{aligned} \text{p-value} &= \sup_{\lambda \in \mathbf{\Lambda}_0} \mathbf{P}_\lambda(|W| > |w|) = \sup_{\lambda \in \{\lambda_0\}} \mathbf{P}_\lambda(|W| > |w|) = \mathbf{P}_{\lambda_0}(|W| > |w|) \\ &\to \mathbf{P}(|Z| > |w|) = 2\Phi(-|w|) = 2\Phi(-|1.0618|) = 2 \times 0.1442 = 0.2884 \ . \end{aligned}$$

Therefore, there is little or no evidence against H_0 that the mean waiting time under an IID Exponential *model of inter-arrival times is exactly ten minutes.*

12.5 Permutation Test for the equality of any two DFs

Permutation test is a non-parametric exact method for testing whether two distributions are the same. It is non-parametric because we do not impose any restrictions on the class of DFs that the unknown DF should belong to. It is exact because we do not have any asymptotic approximations involving sample size approaching infinity. So this test works for any sample size.

Formally, we suppose that:

$$X_1, X_2, \ldots, X_m \overset{IID}{\sim} F^* \quad \text{and} \quad X_{m+1}, X_{m+2}, \ldots, X_{m+n} \overset{IID}{\sim} G^* \ ,$$

are two sets of independent samples. The possibly unknown DFs $F^*, G^* \in \{\text{all DFs}\}$. Now, consider the following hypothesis test:

$$H_0 : F^* = G^* \quad \text{versus} \quad H_1 : F^* \neq G^* \ .$$

Let our test statistic $T(X_1, \ldots, X_m, X_{m+1}, \ldots, X_{m+n})$ be some sensible one – T is large when F^* is too different from G^*, say:

$$T := T(X_1, \ldots, X_m, X_{m+1}, \ldots, X_{m+n}) = \text{abs}\left(\frac{1}{m}\sum_{i=1}^{m} X_i - \frac{1}{n}\sum_{i=m+1}^{n} X_i\right) \ .$$

Then the idea of a permutation test is as follows:

1. Let $N := m + n$ be the pooled sample size and consider all $N!$ permutations of the observed data $x_{\mathsf{obs}} := (x_1, x_2, \ldots, x_m, x_{m+1}, x_{m+2}, \ldots, x_{m+n})$.
2. For each permutation of the data compute the statistic T(permuted data x) and denote these $N!$ values of T by $t_1, t_2, \ldots, t_{N!}$.
3. Under $H_0 : X_1, \ldots, X_m, X_{m+1}, \ldots, X_{m+n} \overset{IID}{\sim} F^* = G^*$, each of the permutations of $x = (x_1, x_2, \ldots, x_m, x_{m+1}, x_{m+2}, \ldots, x_{m+n})$ has the same joint probability $\prod_{i=1}^{m+n} f(x_i)$, where $f(x_i) = dF(x_i) = dG(x_i)$. Therefore, the transformation of the data by our statistic T also has the same probability over the values of T, namely $\{t_1, t_2, \ldots, t_{N!}\}$. Let $\mathbf{P}_0$ be this permutation distribution that is dicrete and uniform over $\{t_1, t_2, \ldots, t_{N!}\}$.
4. Let $t_{\mathsf{obs}} := T(x_{\mathsf{obs}})$ be the observed value of the statistic.
5. Assuming we reject H_0 when T is large, the p-value is:

$$\text{p-value} = \mathbf{P}_0\left(T \geq t_{\mathsf{obs}}\right) = \frac{1}{N!}\left(\sum_{j=1}^{N!} \mathbb{1}(t_j \geq t_{\mathsf{obs}})\right), \qquad \mathbb{1}(t_j \geq t_{\mathsf{obs}}) = \begin{cases} 1 & \text{if} \quad t_j \geq t_{\mathsf{obs}} \\ 0 & \text{otherwise} \end{cases}$$

Let us look at a small example involving the diameters of coarse venus shells (*Dosinia anus*) that Guo Yaozong and Chen Shun found on the left and right sides of the New Brighton prier in Spring 2007. We are interested in testing the hypothesis that the distribution of shell diameters for this bivalve species is the same on both sides of the pier.

Example 40 (Guo-Chen Experiment with Venus Shell Diameters) *Let us look at the first two samples x_1 and x_2 from the left of pier and the first sample from the right side of pier, namely x_3. Since the permutation test is exact, we can use this small data set with merely three samples to conduct the following hypothesis test:*

$$H_0 : X_1, X_2, X_3 \overset{IID}{\sim} F^* = G^*, \qquad H_1 : X_1, X_2 \overset{IID}{\sim} F^*, X_3 \overset{IID}{\sim} G^*, \quad F^* \neq G^* \ .$$

Let us use the test statistic:

$$T(X_1, X_2, X_3) = \text{abs}\left(\frac{1}{2}\sum_{i=1}^{2} X_i - \frac{1}{1}\sum_{i=2+1}^{3} X_i\right) = \text{abs}\left(\frac{X_1 + x_2}{2} - \frac{X_3}{1}\right) \ .$$

The data giving the shell diameters in millimetres and t_{obs} *are:*

$$(x_1, x_2, x_3) = (52, 54, 58) \quad \textit{and} \quad t_{\text{obs}} = \text{abs}\left(\frac{52 + 54}{2} - \frac{58}{1}\right) = \text{abs}(53 - 58) = \text{abs}(-5) = 5 \ .$$

Let us tabulate the $(2+1)! = 3! = 3 \times 2 \times 1 = 6$ *permutations of the data* $(x_1, x_2, x_3) = (52, 54, 58)$*, the corresponding values of* T *and their probabilities under the null hypothesis, i.e., the permutation distribution* $\mathbf{P}_0(T)$*.*

Permutation	t	$\mathbf{P}_0(T = t)$
$(52, 54, 58)$	5	$\frac{1}{6}$
$(54, 52, 58)$	5	$\frac{1}{6}$
$(52, 58, 54)$	1	$\frac{1}{6}$
$(58, 52, 54)$	1	$\frac{1}{6}$
$(58, 54, 52)$	4	$\frac{1}{6}$
$(54, 58, 52)$	4	$\frac{1}{6}$

From the table, we get:

$$\text{p-value} = \mathbf{P}_0\left(T \geq t_{\text{obs}}\right) = \mathbf{P}_0\left(T \geq 5\right) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6} = \frac{1}{3} \approxeq 0.333 \ .$$

Therefore, there is little to no evidence against H_0*.*

When the pooled sample size $N = m + n$ gets large, $N!$ would be too numerous to tabulate exhaustively. In this situation, we can use a Monte Carlo approximation of the p-value by generating a large number of random permutations of the data according to the following Steps:

Step 1: Compute the observed statistic $t_{\text{obs}} := T(x_{\text{obs}})$ of data $x_{\text{obs}} := (x_1, \ldots, x_m, x_{m+1}, \ldots, x_{m+n})$.

Step 2: Randomly permute the data and compute the statistic again from the permuted data.

Step 3: Repeat Step 2 B times and let $t_1, \ldots, t_B$ denote the resulting values (B is large, say 1000).

Step 4: The (Monte Carlo) approximate p-value is:

$$\frac{1}{B}\sum_{j=1}^{B} \mathbb{1}(t_j \geq t_{\text{obs}}) \ .$$

Next we implement the above algorithm on the full data set of Guo and Chen obtained from coarse venus shells sampled from the two sides of the New Brighton pier.

Labwork 46 *Test the null hypothesis that the distribution of the diameters of coarse venus shells are the same on both sides of the New Brighton pier.*

Shells.m

```
% this data was collected by Guo Yaozong and Chen Shun as part of their STAT 218 project 2007
% coarse venus shell diameters in mm from left side of New Brighton Pier
left=[52 54 60 60 54 47 57 58 61 57 50 60 60 60 62 44 55 58 55 60 59 65 59 63 51 61 62 61 60 61 65 ...
    43 59 58 67 56 64 47 64 60 55 58 41 53 61 60 49 48 47 42 50 58 48 59 55 59 50 47 47 33 51 61 61 ...
    52 62 64 64 47 58 58 61 50 55 47 39 59 64 63 63 62 64 61 50 62 61 65 62 66 60 59 58 58 60 59 61 ...
    55 55 62 51 61 49 52 59 60 66 50 59 64 64 62 60 65 44 58 63];
% coarse venus shell diameters in mm from right side of New Brighton Pier
right=[58 54 60 55 56 44 60 52 57 58 61 66 56 59 49 48 69 66 49 72 49 50 59 59 59 66 62 ...
    44 49 40 59 55 61 51 62 52 63 39 63 52 62 49 48 65 68 45 63 58 55 56 55 57 34 64 66 ...
    54 65 61 56 57 59 58 62 58 40 43 62 59 64 64 65 65 59 64 63 65 62 61 47 59 63 44 43 ...
    59 67 64 60 62 64 65 59 55 38 57 61 52 61 61 60 34 62 64 58 39 63 47 55 54 48 60 55 ...
    60 65 41 61 59 65 50 54 60 48 51 68 52 51 61 57 49 51 62 63 59 62 54 59 46 64 49 61];
Tobs=abs(mean(left)-mean(right));% observed test statistic
nleft=length(left); % sample size of the left-side data
nright=length(right); % sample size of the right-side data
ntotal=nleft+nright; % sample size of the pooled data
total=[left right]; % observed data -- ordered: left-side data followed by right-side data
B=10000; % number of bootstrap replicates
TB=zeros(1,B); % initialise a vector of zeros for the bootstrapped test statistics
ApproxPValue=0; % initialise an accumulator for approximate p-value
for b=1:B % eneter the bootsrap replication loop
    % use MATLAB's randperm function to get a random permutation of indices{1,2,...,ntotal}
    PermutedIndices=randperm(ntotal);
    % use the first nleft of the PermutedIndices to get the bootstrapped left-side data
    Bleft=total(PermutedIndices(1:nleft));
    % use the last nright of the PermutedIndices to get the bootstrapped right-side data
    Bright=total(PermutedIndices(nleft+1:ntotal));
    TB(b) = abs(mean(Bleft)-mean(Bright)); % compute the test statistic for the bootstrapped data
    if(TB(b)>Tobs) % increment the ApproxPValue accumulator by 1/B if bootstrapped value > Tobs
        ApproxPValue=ApproxPValue+(1/B);
    end
end
ApproxPValue % report the Approximate p-value
```

When we execute the script to perform a permutation test and approximate the p-value*, we obtain:*

```
>> Shells
ApproxPValue =    0.8576
```

Therefore, there is little or no evidence against the null hypothesis.

12.6 Pearson's Chi-Square Test for Multinomial Trials

We derive the Chi-square distribution introduced by Karl Pearson in 1900 [*Philosophical Magazine*, Series 5, **50**, 157-175]. This historical work laid the foundations of modern statistics by showing why an experimenter cannot simply plot experimental data and just assert the correctness of his or her hypothesis. This derivation is adapted from Donald E. Knuth's treatment [*Art of Computer Programming, Vol. II, Seminumerical Algorithms*, 3rd Ed., 1997, pp. 55-56]. We show how de Moivre, Multinomial, Poisson and the Normal random vectors conspire to creare the Chi-square random variable.

Part 1: de Moivre *trials*
Consider n independent and identically distributed de Moivre$(\theta_1, \ldots, \theta_k)$ random vectors (R$\vec{\text{V}}$s):

$$X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{de Moivre}(\theta_1, \ldots, \theta_k) \ .$$

Recall from Model 15 that $X_1 \sim$ de Moivre$(\theta_1, \ldots, \theta_k)$ means $\mathbf{P}(X_1 = e_i) = \theta_i$ for $i \in \{1, \ldots, k\}$, where $e_1, \ldots, e_k$ are ortho-normal basis vectors in $\mathbb{R}^k$. Thus, for each $i \in \{1, 2, \ldots, n\}$, the corresponding X_i has k components, i.e. $X_i := (X_{i,1}, X_{i,2}, \ldots, X_{i,k})$.

Part 2: Multinomial *trial*
Suppose we are only interested in the experiment induced by their sum:

$$Y := (Y_1, \ldots, Y_k) := \sum_{i=1}^{n} X_i := \sum_{i=1}^{n} (X_{i,1}, X_{i,2}, \ldots, X_{i,k}) = \left(\sum_{i=1}^{n} X_{i,1}, \sum_{i=1}^{n} X_{i,2}, \ldots, \sum_{i=1}^{n} X_{i,k} \right) \ .$$

The $\vec{\text{RV}}$ Y, being the sum of n IID de Moivre$(\theta_1, \ldots, \theta_k)$ $\vec{\text{RV}}$s, is the Multinomial$(n, \theta_1, \ldots, \theta_k)$ $\vec{\text{RV}}$ of Model 16 and the probability that $Y := (Y_1, \ldots, Y_k) = y := (y_1, \ldots, y_k)$ is:

$$\frac{n!}{y_1! y_2! \cdots y_k!} \prod_{i=1}^{k} \theta_i^{y_i} \ .$$

The support of the $\vec{\text{RV}}$ Y, i.e. the set of possible realisations of $y := (y_1, \ldots, y_k)$ is:

$$\mathbb{Y} := \{(y_1, \ldots, y_k) \in \mathbb{Z}_+^k : \sum_{i=1}^{k} y_i = n\} \ .$$

Part 3: Conditional sum of Poisson *trials*
Here we consider an alternative formulation of the Multinomial$(n, \theta_1, \ldots, \theta_k)$ $\vec{\text{RV}}$ Y. Suppose,

$$Y_1 \sim \text{Poisson}(n\theta_1), Y_2 \sim \text{Poisson}(n\theta_2), \ldots, Y_k \sim \text{Poisson}(n\theta_k) \ ,$$

and that $Y_1, \ldots, Y_k$ are independent. Recall from Model 12 that $Y_i \sim \text{Poisson}(n\theta_i)$ means $\mathbf{P}(Y_i = y_i) = e^{-n\theta_i}(n\theta_i)^{y_i}/y_i!$ for $y_i \in \{0, 1, \ldots\}$. Then, the joint probability probability of the $\vec{\text{RV}}$ $(Y_1, \ldots, Y_k)$ is the product of the independent Poisson probabilities:

$$\begin{aligned}
\mathbf{P}\left((Y_1, \ldots, Y_k) = (y_1, \ldots, y_k)\right) := \mathbf{P}\left(Y_1 = y_1, \ldots, Y_k = y_k\right) &= \prod_{i=1}^{k} \mathbf{P}(Y_i = y_i) = \prod_{i=1}^{k} \frac{e^{-n\theta_i}(n\theta_i)^{y_i}}{y_i!} \\
&= \frac{\prod_{i=1}^{k} e^{-n\theta_i} n^{y_i} \theta_i^{y_i}}{\prod_{i=1}^{k} y_i!} = \left(e^{-n\sum_{i=1}^{k}\theta_i} n^{\sum_{i=1}^{k} y_i} \prod_{i=1}^{k} \theta_i^{y_i} \right) \frac{1}{\prod_{i=1}^{k} y_i!} \\
&= \frac{e^{-n} n^n \prod_{i=1}^{k} \theta_i^{y_i}}{\prod_{i=1}^{k} y_i!} \ .
\end{aligned}$$

Now, the probability that sum $Y_1 + \cdots + Y_k$ will equal n is obtained by summing over the probabilities of all $(y_1, \ldots, y_k) \in \mathbb{Y}$:

$$\begin{aligned}
\mathbf{P}\left(\sum_{i=1}^{k} Y_i = n \right) = \sum_{\substack{(y_1,\ldots,y_k) \\ \in \mathbb{Y}}} \mathbf{P}\left((Y_1, \ldots, Y_k) = (y_1, \ldots, y_k)\right) &= \sum_{\substack{(y_1,\ldots,y_k) \\ \in \mathbb{Y}}} \frac{e^{-n} n^n \prod_{i=1}^{k} \theta_i^{y_i}}{\prod_{i=1}^{k} y_i!} \\
= e^{-n} n^n \frac{1}{n!} \underbrace{\left(\sum_{\substack{(y_1,\ldots,y_k) \\ \in \mathbb{Y}}} \frac{n!}{\prod_{i=1}^{k} y_i!} \prod_{i=1}^{k} \theta_i^{y_i} \right)}_{=\mathbf{P}(\mathbb{Y})=1} &= \frac{e^{-n} n^n}{n!} \ .
\end{aligned}$$

Finally, the conditional probability that $(Y_1, \ldots, Y_k) = (y_1, \ldots, y_k)$ given $\sum_{i=1}^k Y_i = n$ is:

$$\mathbf{P}\left((Y_1, \ldots, Y_k) = (y_1, \ldots, y_k) | \sum_{i=1}^k Y_i = n\right) = \frac{\mathbf{P}\left((Y_1, \ldots, Y_k) = (y_1, \ldots, y_k), \sum_{i=1}^k Y_i = n\right)}{\mathbf{P}\left(\sum_{i=1}^k Y_i = n\right)}$$
$$= \frac{\mathbf{P}\left((Y_1, \ldots, Y_k) = (y_1, \ldots, y_k)\right)}{\mathbf{P}\left(\sum_{i=1}^k Y_i = n\right)} = \frac{e^{-n} n^n \prod_{i=1}^k \theta_i^{y_i}}{\prod_{i=1}^k y_i!} \frac{n!}{e^{-n} n^n} = \frac{n!}{y_1! y_2! \cdots y_k!} \prod_{i=1}^k \theta_i^{y_i} \ .$$

Therefore, we may also think of the random vector $Y := (Y_1, \ldots, Y_k) \sim \text{Multinomial}(n, \theta_1, \ldots, \theta_k)$ as k independent Poisson random variables, $Y_1 \sim \text{Poisson}(n\theta_1), \ldots, Y_k \sim \text{Poisson}(n\theta_k)$, that have been conditioned on their sum $\sum_{i=1}^n Y_i$ being n.

Part 4: The Normal *approximation of the centred and scaled* Poisson
Recall from Model 12 that the expectation and variance of a RV $Y_i \sim \text{Poisson}(n\theta_i)$ are $\mathbf{E}(Y_i) = \mathbf{V}(Y_i) = n\theta_i$. Let Z_i be $\mathbf{E}(Y_i)$-centred and $\sqrt{\mathbf{V}(Y_i)}$-scaled Y_i and

$$Z_i := \frac{Y_i - \mathbf{E}(Y_i)}{\sqrt{\mathbf{V}(Y_i)}} = \frac{Y_i - n\theta_i}{\sqrt{n\theta_i}} \ .$$

The condition that $Y_1 + \cdots + Y_k = n$ is equivalent to requiring that $\sqrt{\theta_1} Z_1 + \cdots + \sqrt{\theta_k} Z_k = 0$, since:

$$\sum_{i=1}^k Y_i = n \iff \sum_{i=1}^k Y_i - n = 0 \iff \sum_{i=1}^k Y_i - n \sum_{i=1}^k \theta_i = 0 \iff \sum_{i=1}^k Y_i - n\theta_i = 0$$
$$\iff \sum_{i=1}^k \frac{Y_i - n\theta_i}{\sqrt{n}} = 0 \iff \sum_{i=1}^k \sqrt{\theta_i} \frac{Y_i - n\theta_i}{\sqrt{n\theta_i}} = 0 \iff \sum_{i=1}^k \sqrt{\theta_i} Z_i = 0 \ .$$

Now consider the support of the R$\vec{\text{V}}$ $Z := (Z_1, \ldots, Z_k)$ conditioned on $\sum_{i=1}^k \sqrt{\theta_i} Z_i = 0$, i.e. the hyper-plane of $(k-1)$-dimensional vectors:

$$\mathbb{H} := \{(z_1, \ldots, z_k) : \sqrt{\theta_1} z_1 + \cdots + \sqrt{\theta_k} z_k = 0\}$$

Each $Z_i \rightsquigarrow \text{Normal}(0, 1)$ by the central limit theorem. Therefore, for large values of n, each Z_i is approximately distributed as the Normal$(0, 1)$ RV with PDF $f(z_i; 0, 1) = (2\pi)^{-1/2} \exp(-z_i^2/2)$. Since the Z_is are independent except for the condition that they lie in $\mathbb{H}$, the point in a differential volume $dz_2 \ldots dz_k$ of $\mathbb{H}$ occur with probability approximately proportional to:

$$\exp(-z_1^2/2) \times \cdots \times \exp(-z_k^2/2) = \exp(-(z_1^2 + \cdots + z_k^2)/2)$$

Part 5: Chi-square *distribution as the sum of squared* Normal*s*
We are interested in the sum of the area of squares with side-lengths $Z_1, \ldots, Z_k$. Let V be the desired sum of squares:

$$V := \sum_{i=1}^k Z_i^2 = \sum_{i=1}^k \frac{(Y_i - n\theta_i)^2}{n\theta_i}, \quad \text{such that} \quad Z_i \rightsquigarrow \text{Normal}(0, 1), \quad \sum_{i=1}^k \sqrt{\theta_i} Z_i = 0 \ .$$

The probability that $V \leq v$ as $n \to \infty$ is:

$$\frac{\int_{(z_1, \ldots, z_k) \in \mathbb{H} \text{ and } \sum_{i=1}^k z_i^2 \leq v} \exp(-(z_1^2 + \cdots + z_k^2)/2) \ dz_2 \ldots dz_k}{\int_{(z_1, \ldots, z_k) \in \mathbb{H}} \exp(-(z_1^2 + \cdots + z_k^2)/2) \ dz_2 \ldots dz_k}$$

Since the $(k-1)$-dimensional hyper-plane $\mathbb{H}$ passes through the origin of $\mathbb{R}^k$, the domain of integration in the numerator above is the interior of a $(k-1)$-dimensional hyper-sphere of radius $\sqrt{v}$ that is centred at the origin. Using a transformation of the above ratio of integrals into generalised polar co-ordinates with radius χ and angles $\alpha_1, \ldots, \alpha_{k-2}$, we get:

$$\frac{\int_{\chi^2 \leq v} \exp(-\chi^2/2)\chi^{k-2} g(\alpha_1, \ldots, \alpha_{k-2}) d\chi d\alpha_1 \cdots d\alpha_{k-2}}{\int \exp(-\chi^2/2)\chi^{k-2} g(\alpha_1, \ldots, \alpha_{k-2}) d\chi d\alpha_1 \cdots d\alpha_{k-2}} \ ,$$

for some function g of the angles [See Problem 15 in *Art of Computer Programming, Vol. II, Seminumerical Algorithms*, 3rd Ed., 1997, pp. 59]. The integration over the $(k-2)$ angles results in the same factor that cancels between the numerator and the denominator. This yields the formula for the probability that $V \leq v$ as $n \to \infty$:

$$\lim_{n \to \infty} \mathbf{P}(V \leq v) = \frac{\int_0^{\sqrt{v}} \exp(-\chi^2/2)\chi^{k-2} d\chi}{\int_0^{\infty} \exp(-\chi^2/2)\chi^{k-2} \ d\chi} \ .$$

By substituting $t = \chi^2/2$, we can express the integrals in terms of the incomplete Gamma function defined as $\gamma(a, x) := \int_0^a \exp(-t)t^{a-1} \ dt$ as follows:

$$\mathbf{P}(V \leq v) = \gamma\left(\frac{k-1}{2}, \frac{v}{2}\right) / \Gamma\left(\frac{k-1}{2}\right) \ .$$

This is the DF of the Chi-square distribution with $k-1$ degrees of freedom.

Model 19 (Chi-square(k) **RV)** *Given a parameter $k \in \mathbb{N}$ called degrees of freedom, we say that V is a* Chi-square(k) *RV if its PDF is:*

$$f(v; k) := \frac{v^{(k/2)-1} e^{-v/2}}{2^{k/2}\Gamma(k/2)} \mathbb{1}_{\{v \in \mathbb{R}: v > 0\}}(v)$$

Also, $\mathbf{E}(V) = k$ *and* $\mathbf{V}(V) = 2k$.

We can use the test statistic:

$$T := T(Y_1, \ldots, Y_k) = \frac{(Y_1 - n\theta_1^*)^2}{n\theta_1^*} + \cdots + \frac{(Y_k - n\theta_k^*)^2}{n\theta_k^*}$$

to test the null hypothesis H_0 that may be formalised in three equivalent ways:

$$\begin{aligned} & H_0 : X_1, X_2, \ldots, X_n \overset{IID}{\sim} \text{de Moivre}(\theta_1^*, \ldots, \theta_k^*) \ \vec{\text{RV}} \\ \iff & H_0 : Y := (Y_1, \ldots, Y_k) = \sum_{i=1}^{n} X_i \sim \text{Multinomial}(n, \theta_1^*, \ldots, \theta_k^*) \ \vec{\text{RV}} \\ \iff & H_0 : Y_1 \overset{IND}{\sim} \text{Poisson}(n\theta_1) \text{ RV}, \ldots, Y_k \overset{IND}{\sim} \text{Poisson}(n\theta_k) \text{ RV given that } \sum_{i=1}^{k} Y_i = n \end{aligned}$$

We have seen that under H_0, the test statistic $T \rightsquigarrow V \sim$ Chi-square$(k-1)$. Let t_{obs} be the observed value of the test statistic and let the upper alpha quantile be $\chi^2_{k-1,\alpha} := F^{[-1]}(1-\alpha)$, where F is the CDF of $V \sim$ Chi-square$(k-1)$. Hence the test:

Reject H_0 if $T > \chi^2_{k-1,\alpha}$ is an asymptotically size α test and the $\mathsf{p\text{-}value} = \mathbf{P}(V > t_{obs})$.

Chapter 13

Appendix

13.1 Code

Labwork 47 *Here are the functions to evaluate the PDF and DF of a* Normal(μ, σ^2) *RV* X *at a given* x*.*

NormalPdf.m

```
function fx = NormalPdf(x,Mu,SigmaSq)
% Returns the Pdf of Normal(Mu, SigmaSq), at x,
% where Mu=mean and SigmaSq = Variance
%
% Usage: fx = NormalPdf(x,Mu,SigmaSq)
if SigmaSq <= 0
   error('Variance must be > 0')
   return
end

Den = ((x-Mu).^2)/(2*SigmaSq);
Fac = sqrt(2*pi)*sqrt(SigmaSq);

fx = (1/Fac)*exp(-Den);
```

NormalCdf.m

```
function Fx = NormalCdf(x,Mu,SigmaSq)
% Returns the Cdf of Normal(Mu, SigmaSq), at x,
% where Mu=mean and SigmaSq = Variance using
% MATLAB's error function erf
%
% Usage: Fx = NormalCdf(x,Mu,SigmaSq)
if SigmaSq <= 0
   error('Variance must be > 0')
   return
end

Arg2Erf = (x-Mu)/sqrt(SigmaSq*2);
Fx = 0.5*erf(Arg2Erf)+0.5;
```

Plots of the PDF and DF of several Normally distributed RVs depicted in Figure 6.5 were generated using the following script file:

PlotPdfCdfNormal.m

```
% PlotPdfCdfNormal.m script file
% Plot of some pdf's and cdf's of the Normal(mu,SigmaSq) RV X
```

```
%
x=[-6:0.0001:6]; % points from the subset [-5,5] of the support of X
subplot(1,2,1) % first plot of a 1 by 2 array of plots
plot(x,NormalPdf(x,0,1),'r') % pdf of RV Z ~ Normal(0,1)
hold % to superimpose plots
plot(x,NormalPdf(x,0,1/10),'b') % pdf of RV X ~ Normal(0,1/10)
plot(x,NormalPdf(x,0,1/100),'m') % pdf of RV X ~ Normal(0,1/100)
plot(x,NormalPdf(x,-3,1),'r--') % pdf of RV Z ~ Normal(-3,1)
plot(x,NormalPdf(x,-3,1/10),'b--') % pdf of RV X ~ Normal(-3,1/10)
plot(x,NormalPdf(x,-3,1/100),'m--') % pdf of RV X ~ Normal(-3,1/100)
xlabel('x')
ylabel('f(x; \mu, \sigma^2)')
legend('f(x;0,1)','f(x;0,10^{-1})','f(x;0,10^{-2})','f(x;-3,1)','f(x;-3,10^{-1})','f(x;-3,10^{-2})')
subplot(1,2,2) % second plot of a 1 by 2 array of plots
plot(x,NormalCdf(x,0,1),'r') % DF of RV Z ~ Normal(0,1)
hold % to superimpose plots
plot(x,NormalCdf(x,0,1/10),'b') % DF of RV X ~ Normal(0,1/10)
plot(x,NormalCdf(x,0,1/100),'m') % DF of RV X ~ Normal(0,1/100)
plot(x,NormalCdf(x,-3,1),'r--') % DF of RV Z ~ Normal(-3,1)
plot(x,NormalCdf(x,-3,1/10),'b--') % DF of RV X ~ Normal(-3,1/10)
plot(x,NormalCdf(x,-3,1/100),'m--') % DF of RV X ~ Normal(-3,1/100)
xlabel('x')
ylabel('F(x; \mu, \sigma^2)')
legend('F(x;0,1)','F(x;0,10^{-1})','F(x;0,10^{-2})','F(x;-3,1)','F(x;-3,10^{-1})','F(x;-3,10^{-2})')
```

Labwork 48 *Here are the functions to evaluate the PDF and DF of an* Exponential(λ) *RV X ata given x (point or a vector).*

ExponentialPdf.m

```
function fx = ExponentialPdf(x,Lambda)
% Returns the Pdf of Exponential(Lambda) RV at x,
% where Lambda = rate parameter
%
% Usage: fx = ExponentialPdf(x,Lambda)
if Lambda <= 0
   error('Rate parameter Lambda must be > 0')
   return
end

fx = Lambda * exp(-Lambda * x);
```

ExponentialCdf.m

```
function Fx = ExponentialCdf(x,Lambda)
% Returns the Cdf of Exponential(Lambda) RV at x,
% where Lambda = rate parameter
%
% Usage: Fx = ExponentialCdf(x,Lambda)
if Lambda <= 0
   error('Rate parameter Lambda must be > 0')
   return
end
Fx = 1.0 - exp(-Lambda * x);
```

Plots of the PDF and DF of several Exponentially distributed RVs at four axes scales that are depicted in Figure 6.2 were generated using the following script file:

PlotPdfCdfExponential.m

```
% PlotPdfCdfExponential.m script file
% Plot of some pdf's and cdf's of the Exponential(Lambda) RV X
%
```

```
x=[0:0.0001:100]; % points from the subset [0,100] of the support of X
subplot(2,4,1) % first plot of a 1 by 2 array of plots
plot(x,ExponentialPdf(x,1),'r:','LineWidth',2) % pdf of RV X ~ Exponential(1)
hold on % to superimpose plots
plot(x,ExponentialPdf(x,10),'b--','LineWidth',2) % pdf of RV X ~ Exponential(10)
plot(x,ExponentialPdf(x,1/10),'m','LineWidth',2) % pdf of RV X ~ Exponential(1/10)
xlabel('x')
ylabel('f(x; \lambda)')
legend('f(x;1)','f(x;10)','f(x;10^{-1})')
axis square
axis([0,2,0,10])
title('Standard Cartesian Scale')
hold off

subplot(2,4,2)
semilogx(x,ExponentialPdf(x,1),'r:','LineWidth',2) % pdf of RV X ~ Exponential(1)
hold on % to superimpose plots
semilogx(x,ExponentialPdf(x,10),'b--','LineWidth',2) % pdf of RV X ~ Exponential(10)
semilogx(x,ExponentialPdf(x,1/10),'m','LineWidth',2) % pdf of RV X ~ Exponential(1/10)
%xlabel('x')
%ylabel('f(x; \lambda)')
%legend('f(x;1)','f(x;10)','f(x;10^{-1})')
axis square
axis([0,100,0,10])
title('semilog(x) Scale')
hold off

subplot(2,4,3)
semilogy(x,ExponentialPdf(x,1),'r:','LineWidth',2) % pdf of RV X ~ Exponential(1)
hold on % to superimpose plots
semilogy(x,ExponentialPdf(x,10),'b--','LineWidth',2) % pdf of RV X ~ Exponential(10)
semilogy(x,ExponentialPdf(x,1/10),'m','LineWidth',2) % pdf of RV X ~ Exponential(1/10)
%xlabel('x');
%ylabel('f(x; \lambda)');
%legend('f(x;1)','f(x;10)','f(x;10^{-1})')
axis square
axis([0,100,0,1000000])
title('semilog(y) Scale')
hold off

x=[ [0:0.001:1] [1.001:1:100000]]; % points from the subset [0,100] of the support of X
subplot(2,4,4)
loglog(x,ExponentialPdf(x,1),'r:','LineWidth',2) % pdf of RV X ~ Exponential(1)
hold on % to superimpose plots
loglog(x,ExponentialPdf(x,10),'b--','LineWidth',2) % pdf of RV X ~ Exponential(10)
loglog(x,ExponentialPdf(x,1/10),'m','LineWidth',2) % pdf of RV X ~ Exponential(1/10)
%xlabel('x')
%ylabel('f(x; \lambda)')
%legend('f(x;1)','f(x;10)','f(x;10^{-1})')
axis square
axis([0,100000,0,1000000])
title('loglog Scale')
hold off

x=[0:0.0001:100]; % points from the subset [0,100] of the support of X
subplot(2,4,5) % second plot of a 1 by 2 array of plots
plot(x,ExponentialCdf(x,1),'r:','LineWidth',2) % cdf of RV X ~ Exponential(1)
hold on % to superimpose plots
plot(x,ExponentialCdf(x,10),'b--','LineWidth',2) % cdf of RV X ~ Exponential(10)
plot(x,ExponentialCdf(x,1/10),'m','LineWidth',2) % cdf of RV X ~ Exponential(1/10)
xlabel('x')
ylabel('F(x; \lambda)')
legend('F(x;1)','f(x;10)','f(x;10^{-1})')
axis square
axis([0,10,0,1])
```

```
hold off

subplot(2,4,6) % second plot of a 1 by 2 array of plots
semilogx(x,ExponentialCdf(x,1),'r:','LineWidth',2) % cdf of RV X ~ Exponential(1)
hold on % to superimpose plots
semilogx(x,ExponentialCdf(x,10),'b--','LineWidth',2) % cdf of RV X ~ Exponential(10)
semilogx(x,ExponentialCdf(x,1/10),'m','LineWidth',2) % cdf of RV X ~ Exponential(1/10)
%xlabel('x')
%ylabel('F(x; \lambda)')
%legend('F(x;1)','F(x;10)','F(x;10^{-1})')
axis square
axis([0,100,0,1])
%title('semilog(x) Scale')
hold off

subplot(2,4,7)
semilogy(x,ExponentialCdf(x,1),'r:','LineWidth',2) % cdf of RV X ~ Exponential(1)
hold on % to superimpose plots
semilogy(x,ExponentialCdf(x,10),'b--','LineWidth',2) % cdf of RV X ~ Exponential(10)
semilogy(x,ExponentialCdf(x,1/10),'m','LineWidth',2) % cdf of RV X ~ Exponential(1/10)
%xlabel('x');
%ylabel('F(x; \lambda)');
%legend('F(x;1)','F(x;10)','F(x;10^{-1})')
axis square
axis([0,10,0,1])
%title('semilog(y) Scale')
hold off

x=[ [0:0.001:1] [1.001:1:100000]]; % points from the subset of the support of X
subplot(2,4,8)
loglog(x,ExponentialCdf(x,1),'r:','LineWidth',2) % cdf of RV X ~ Exponential(1)
hold on % to superimpose plots
loglog(x,ExponentialCdf(x,10),'b--','LineWidth',2) % cdf of RV X ~ Exponential(10)
loglog(x,ExponentialCdf(x,1/10),'m','LineWidth',2) % cdf of RV X ~ Exponential(1/10)
%xlabel('x')
%ylabel('F(x; \lambda)')
%legend('F(x;1)','F(x;10)','F(x;10^{-1})')
axis square
axis([0,100000,0,1])
%title('loglog Scale')
hold off
```

Labwork 49 *A* MATLAB *function to plot the empirical DF* (10.1) *of* n *user-specified samples efficiently for massive number of samples. Read the following M-file for the algorithm:*

ECDF.m

```
function [x1 y1] = ECDF(x, PlotFlag, LoxD, HixD)
% return the x1 and y1 values of empirical CDF
% based on samples in array x of RV X
% plot empirical CDF if PlotFlag is >= 1
%
% Call Syntax: [x1 y1] = ECDF(x, PlotFlag, LoxD,HixD);
% Input      : x = samples from a RV X (a vector),
%              PlotFlag is a number controlling plot (Y/N, marker-size)
%              LoxD is a number by which the x-axis plot range is extended to the left
%              HixD is a number by which the x-aixs plot range is extended to the right
% Output     : [x1 y1] & empirical CDF Plot IF PlotFlag >= 1
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

R=length(x);       % assume x is a vector and R = Number of samples in x
x1=zeros(1,R+2);
y1=zeros(1,R+2);        % initialize y to null vectors
for i=1:1:R        % loop to append to x and y axis values of plot
```

```
y1(i+1)=i/R;                    % append equi-increments of 1/R to y
end                 % end of for loop
x1(2:R+1)=sort(x);         % sorting the sample values
x1(1)=x1(2)-LoxD; x1(R+2)=x1(R+1)+HixD;   % padding x for emp CDF to start at min(x) and end at max(x)
y1(1)=0; y1(R+2)=1;          % padding y so emp CDF start at y=0 and end at y=1

% to make a ECDF plot for large number of points set the PlotFlag<1 and use
% MATLAB's plot function on the x and y values returned by ECDF -- stairs(x,y)
if PlotFlag >= 1       % Plot customized empirical CDF if PlotFlag >= 1
    %newplot;
    MSz=10/PlotFlag;   % set Markersize MSz for dots and circles in ECDF plot
                       % When PlotFlag is large MSz is small and the
                       % Markers effectively disappear in the ecdf plot
    R=length(x1);       % update R = Number of samples in x
        hold on            % hold plot for superimposing plots

    for i=1:1:R-1
        if(i>1 && i ~= R-1)
            plot([x1(i),x1(i+1)],[y1(i),y1(i)],'k o -','MarkerSize',MSz)
        end
        if (i< R-1)
          plot(x1(i+1),y1(i+1),'k .','MarkerSize',2.5*MSz)
        end
        plot([x1(i),x1(i+1)],[y1(i),y1(i)],'k -')
        plot([x1(i+1),x1(i+1)],  [y1(i),y1(i+1)],'k -')

    end

    hold off;
end
```

Ideally, this function needs to be rewritten using primitives such as MATLAB*'s* `line` *commands.*

Labwork 50 *Let us implement Algorithm 2 as the following* MATLAB *function:*

qthSampleQuantile.m

```
function qthSQ = qthSampleQuantile(q, SortedXs)
%
% return the q-th Sample Quantile from Sorted array of Xs
%
% Call Syntax: qthSQ = qthSampleQuantile(q, SortedXs);
%
% Input      : q = quantile of interest, NOTE: 0 <= q <= 1
%              SortedXs = sorted real data points in ascending order
% Output     : q-th Sample Quantile, ie, inverse ECDF evaluated at q

% store the length of the the sorted data array SortedXs in n
N = length(SortedXs);
Nminus1TimesQ = (N-1)*q; % store (N-1)*q in a variable
Index = floor(Nminus1TimesQ); % store its floor in a C-style Index variable
Delta = Nminus1TimesQ - Index;
if Index == N-1
    qthSQ = SortedXs(Index+1);
else
    qthSQ = (1.0-Delta)*SortedXs(Index+1) + Delta*SortedXs(Index+2);
end
```

Labwork 51 *Figure* **??** *was made with the following script file.*

LevyDensityPlot.m

```
x=linspace(-9,9,1500);y=x;
[X, Y]=meshgrid(x,y);
```

```
Z1 = (cos((0*X)+1) + 2*cos((1*X)+2) + 3*cos((2*X)+3) + 4*cos((3*X)+4) + 5*cos((4*X)+5));
Z2 = (cos((2*Y)+1) + 2*cos((3*Y)+2) + 3*cos((4*Y)+3) + 4*cos((5*Y)+4) + 5*cos((6*Y)+5));
Temp=50;
Z = exp(-(Z1 .* Z2 + (X + 1.42513) .^2 + (Y + 0.80032) .^ 2)/Temp);
mesh(X,Y,Z)
caxis([0, 10]);
rotate3d on
```

Labwork 52 *The negative of the Levy density* (**??**) *is encoded in the following M-file as a function to be passed to* MATLAB*'s* `fminsearch`*.*

NegLevyDensity.m

```
function NegLevyFunVal = NegLevyDensity(parameters);
X=parameters(1); Y=parameters(2); Temp=50.0;
Z1 = (cos((0*X)+1) + 2*cos((1*X)+2) + 3*cos((2*X)+3) + 4*cos((3*X)+4) + 5*cos((4*X)+5));
Z2 = (cos((2*Y)+1) + 2*cos((3*Y)+2) + 3*cos((4*Y)+3) + 4*cos((5*Y)+4) + 5*cos((6*Y)+5));
NegLevyFunVal = -exp(-(Z1 .* Z2 + (X + 1.42513) .^2 + (Y + 0.80032) .^ 2)/Temp);
```

Labwork 53 *Figure* **??** *was made with the following script file.*

LogNormalLogLklPlot.m

```
% Plots the log likelihood of LogNormal(lambd, zeta),
% for observed data vector x IIDLogNormal(lambd,zeta),
rand('twister',001);
x=exp(arrayfun(@(u)(Sample1NormalByNewRap(u,10.36,0.26^2)),rand(1,100)));
% log likelihood function
lambda=linspace(5,15.0,200);
zeta=linspace(0.1, 2,200);
[LAMBDA, ZETA]=meshgrid(lambda,zeta);
LAMBDA3=repmat(LAMBDA,[1 1 length(x)]);
ZETA3=repmat(ZETA,[1 1 length(x)]);

xx=zeros([1 1 length(x)]);xx(:)=x;
x3=repmat(xx,[length(lambda) length(zeta) 1]);
%l = -sum(log((1 ./ (sqrt(2*pi)*zeta) .* x) .* exp((-1/(2*zeta^2))*(log(x)-lambda).^2)));
LOGLKL = sum(log((1 ./ (sqrt(2*pi)*ZETA3) .* x3) .* exp((-1/(2*ZETA3.^2)).*(log(x3)-LAMBDA3).^2)),3);
LOGLKL(LOGLKL<0)=NaN;

caxis([0 0.1]*10^3);colorbar
axis([0 15 0 2 0 0.1*10^3])
clf; meshc(LAMBDA, ZETA, LOGLKL);
rotate3d on;
```

Labwork 54 *Figure 8.3 was made with the following script file.*

BernoulliMLEConsistency.m

```
clf;%clear any figures
rand('twister',736343); % initialize the Uniform(0,1) Sampler
N = 3; % 10^N is the maximum number of samples from RV
J = 100; % number of Replications for each n
u = rand(J,10^N); % generate 10X10^N samples from Uniform(0,1) RV U
p=0.5; % set p for the Bernoulli(p) trials
PS=[0:0.001:1]; % sample some values for p on [0,1] to plot likelihood
for i=1:N
    if(i==1) Pmin=0.; Pmax=1.0; Ymin=-70; Ymax=-10; Y=linspace(Ymin,Ymax,J); end
    if(i==2) Pmin=0.; Pmax=1.0; Ymin=-550; Ymax=-75; Y=linspace(Ymin,Ymax,J); end
    if(i==3) Pmin=0.3; Pmax=0.8; Ymin=-900; Ymax=-700; Y=linspace(Ymin,Ymax,J); end
    n=10^i;% n= sample size, ie, number of Bernoulli trials
    subplot(1,N,i)
    if(i==1) axis([Pmin Pmax Ymin -2]); end
```

```
    if(i==2) axis([Pmin Pmax Ymin -60]); end
    if(i==3) axis([Pmin Pmax Ymin -685]); end
    EmpCovSEhat=0; % track empirical coverage for SEhat
    EmpCovSE=0; % track empirical coverage for exact SE
    for j=1:J
     % transform the Uniform(0,1) samples to n Bernoulli(p) samples
     x=floor(u(j,1:n)+p);
     s = sum(x); % statistic s is the sum of x_i's
     % display the outcomes and their sum
     %display(x)
     %display(s)
     MLE=s/n; % Analyticaly MLE is s/n
     se = sqrt((1-p)*p/n); % standard error from known p
     sehat = sqrt((1-MLE)*MLE/n); % estimated standard error from MLE p
     Zalphaby2 = 1.96; % for 95% CI
     if(abs(MLE-p)<=2*sehat) EmpCovSEhat=EmpCovSEhat+1; end
     line([MLE-2*sehat MLE+2*sehat],[Y(j) Y(j)],'Marker','+','LineStyle',':','LineWidth',1,'Color',[1 .0 .0])
     if(abs(MLE-p)<=2*se) EmpCovSE=EmpCovSE+1; end
     line([MLE-2*se MLE+2*se],[Y(j) Y(j)],'Marker','+','LineStyle','-')
     % l is the Log Likelihood of data x as a function of parameter p
     l=@(p)sum(log(p ^ s * (1-p)^(n-s)));
     hold on;
     % plot the Log Likelihood function and MLE
     semilogx(PS,arrayfun(l,PS),'m','LineWidth',1);
     hold on; plot([MLE],[Y(j)],'.','Color','c'); % plot MLE
    end
    hold on;
    line([p p], [Ymin, l(p)],'LineStyle',':','Marker','none','Color','k','LineWidth',2)
    %axis([-0.1 1.1]);
    %axis square;
    LabelString=['n=' num2str(n) '  ' 'Cvrg.=' num2str(EmpCovSE) '/' num2str(J) ...
        ' ~=' num2str(EmpCovSEhat) '/' num2str(J)];
    %text(0.75,0.05,LabelString)
    title(LabelString)
    hold off;
end
```

Labwork 55 *The following script was used to generate the Figure 10.1.*

GilvenkoCantelliUnif01.m

```
% from Uniform(0,1) RV
rand('twister',76534); % initialize the Uniform(0,1) Sampler
N = 3; % 10^N is the maximum number of samples from Uniform(0,1) RV
u = rand(10,10^N); % generate 10 X 10^N samples from Uniform(0,1) RV U
x=[0:0.001:1];
% plot the ECDF from the first 10 samples using the function ECDF
for i=1:N
    SampleSize=10^i;
    subplot(1,N,i)
    plot(x,x,'r','LineWidth',2); % plot the DF of Uniform(0,1) RV in red
    % Get the x and y coordinates of SampleSize-based ECDF in x1 and y1 and
    % plot the ECDF using the function ECDF
    for j=1:10

     hold on;
     if (i==1) [x1 y1] = ECDF(u(j,1:SampleSize),2.5,0.2,0.2);
     else
        [x1 y1] = ECDF(u(j,1:SampleSize),0,0.1,0.1);
        stairs(x1,y1,'k');
     end
    end
%     % Note PlotFlag is 1 and the plot range of x-axis is
%     % incremented by 0.1 or 0.2 on either side due to last 2 parameters to ECDF
%     % being 0.1 or 0.2
```

```
%     Alpha=0.05; % set alpha to 5% for instance
%     Epsn = sqrt((1/(2*SampleSize))*log(2/Alpha)); % epsilon_n for the confidence band
     hold on;
%     stairs(x1,max(y1-Epsn,zeros(1,length(y1))),'g'); % lower band plot
%     stairs(x1,min(y1+Epsn,ones(1,length(y1))),'g'); % upper band plot
    axis([-0.1 1.1 -0.1 1.1]);
    %axis square;
    LabelString=['n=' num2str(SampleSize)];
    text(0.75,0.05,LabelString)
    hold off;
end
```

13.2 Data

Here we describe some of the data sets we analyze.

Data 1 (Our Maths & Stats Dept. Web Logs) *We assume access to a* `Unix` *terminal (*`Linux, Mas OS X, Sun Solaris`*, etc). We show how to get your hands dirty with web logs that track among others, every* `IP` *address and its time of login to our department web server over the world-wide-web. The raw text files of web logs may be manipulated but they are typically huge files and need some* `Unix` *command-line utilities.*

```
rsa64@mathopt03:~> cd October010203WebLogs/
rsa64@mathopt03:~/October010203WebLogs> ls -al
-rw-r--r--+  1 rsa64 math 7527169 2007-10-04 09:38 access-07_log.2
-rw-r--r--+  1 rsa64 math 7727745 2007-10-04 09:38 access-07_log.3
```

The files are quite large over 7.5 MB each. So we need to compress it. We use the `gzip` *and* `gunzip` *utility in any* `Unix` *environment to compress and decompress these large text files of web logs. After compression the file sizes are more reasonable.*

```
rsa64@mathopt03:~/October010203WebLogs> gzip access-07_log.3
rsa64@mathopt03:~/October010203WebLogs> gzip access-07_log.2
rsa64@mathopt03:~/October010203WebLogs> zcat access-07_log.2.gz | grep ' 200 '
| awk '{ print \$4}'| sed -e 's/\[\([0-9]\{2\}\)\/\([a-Z]\{3\}\)\/\([0-9]\{4\}\)
:\([0-9]\{2\}\):\([0-9]\{2\}\):\([0-9]\{2\}\)/\3 10 \1 \4 \5 \6/'
2007 10 02 03 57 48
2007 10 02 03 58 31
.
.
.
2007 10 03 03 56 21
2007 10 03 03 56 52
```

Finally, there are 56485 *and* 53966 *logins for the two 24-hour cycles, starting* `01/Oct` *and* `01/Oct`*, respectively. We can easily get these counts by further piping the previous output into the line counting utility* `wc` *with the -l option. All the* `Unix` *command-line tools mentioned earlier can be learned by typing* `man` *followed by the tool-name, for eg. type* `man sed` *to learn about the usage of* `sed` *ata* `Unix` *command shell. We further pipe the output of login times for the two 24-hour cycles starting* `01/Oct` *and* `02/Oct` *in format* `YYYY MM DD HH MM SS` *to* `| sed -e 's/2007 10 //' > WebLogTimes20071001035730.dat` *and* `... > WebLogTimes20071002035730.dat`*, respectively to strip away the redundant information on* `YYYY MM` *, namely* `2007 10` *, and only save*

the relevant information of `DD HH MM SS` *in files named* `WebLogTimes20071001035730.dat` *and* `WebLogTimes20071002035730.dat`*, respectively. These two files have the data of interest to us. Note that the size of these two uncompressed final data files in plain text are smaller than the compressed raw web log files we started out from.*

```
rsa64@mathopt03:~/October010203WebLogs> ls -al
-rw-r--r--+  1 rsa64 math 677820 2007-10-05 15:36 WebLogTimes20071001035730.dat
-rw-r--r--+  1 rsa64 math 647592 2007-10-05 15:36 WebLogTimes20071002035730.dat
-rw-r--r--+  1 rsa64 math 657913 2007-10-04 09:38 access-07_log.2.gz
-rw-r--r--+  1 rsa64 math 700320 2007-10-04 09:38 access-07_log.3.gz
```

Now that we have been familiarized with the data of login times to our web-server over 2 24-hour cycles, let us do some statistics. The log files and basic scripts are courtesy of the Department's computer systems administrators Paul Brouwers and Steve Gourdie. This data processing activity was shared in such detail to show you that statistics is only meaningful when the data and the process that generated it are clear to the experimenter. Let us process the data and visualize the empirical distribution functions using the following script:

WebLogDataProc.m

```
load WebLogTimes20071001035730.dat % read data from first file
% multiply day (October 1) by 24*60*60 seconds, hour by 60*60 seconds,
% minute by 60 seconds and seconds by 1, to rescale time in units of seconds
SecondsScale1 = [24*60*60; 60*60; 60; 1;];
StartTime1 = [1 3 57 30] * SecondsScale1; % find start time in seconds scale
%now convert time in Day/Hours/Minutes/Seconds format to seconds scale from
%the start time
WebLogSeconds20071001035730 = WebLogTimes20071001035730 * SecondsScale1 - StartTime1;

% repeat the data entry process above on the second file
load WebLogTimes20071002035730.dat %
SecondsScale1 = [24*60*60; 60*60; 60; 1;];
StartTime2 = [2 3 57 30] * SecondsScale1;
WebLogSeconds20071002035730 = WebLogTimes20071002035730 * SecondsScale1 - StartTime2;

% calling a more efficient ECDF function for empirical DF's
[x1 y1]=ECDF(WebLogSeconds20071001035730,0,0,0);
[x2 y2]=ECDF(WebLogSeconds20071002035730,0,0,0);
stairs(x1,y1,'r','linewidth',1) % draw the empirical DF for first dataset
hold on;
stairs(x2,y2,'b') % draw empirical cdf for second dataset

% set plot labels and legends and title
xlabel('time t in seconds')
ylabel('ECDF    F^\^(t)')
grid on
legend('Starting 10\01\0357\30', 'Starting 10\02\0357\30')
title('24-Hour Web Log Times of Maths & Stats Dept. Server at Univ. of Canterbury, NZ')

%To draw the confidence bands
Alpha=0.05; % set alpha
% compute epsilon_n for first dataset of size 56485
Epsn1 = sqrt((1/(2*56485))*log(2/Alpha));
stairs(x1,max(y1-Epsn1,zeros(1,length(y1))),'g') % lower 1-alpha confidence band
stairs(x1,min(y1+Epsn1,ones(1,length(y1))),'g') % upper 1-alpha confidence band

% compute epsilon_n for second dataset of size 53966
Epsn2 = sqrt((1/(2*53966))*log(2/Alpha));
stairs(x2,max(y2-Epsn2,zeros(1,length(y2))),'g') % lower 1-alpha confidence band
stairs(x2,min(y2+Epsn2,ones(1,length(y2))),'g') % upper 1-alpha confidence band
```

2007 Student Project Appendix

Please visit `http://www.math.canterbury.ac.nz/~r.sainudiin/courses/STAT218/projects/Stat218StudentProjects2007.pdf` to see the term projects completed by students of STAT 218 from 2007. Some of the simulation devices and non-perishable data from these projects are archived on the sixth floor of the Erskine Building.