Introduction to Statistical Thinking
(With R, Without Calculus)

Benjamin Yakir, The Hebrew University

June, 2011

In memory of my father, Moshe Yakir, and the family he lost.

Preface

The target audience for this book is college students who are required to learn statistics, students with little background in mathematics and often no motivation to learn more. It is assumed that the students do have basic skills in using computers and have access to one. Moreover, it is assumed that the students are willing to actively follow the discussion in the text, to practice, and more importantly, to think.

Teaching statistics is a challenge. Teaching it to students who are required to learn the subject as part of their curriculum, is an art mastered by few. In the past I have tried to master this art and failed. In desperation, I wrote this book.

This book uses the basic structure of generic introduction to statistics course. However, in some ways I have chosen to diverge from the traditional approach. One divergence is the introduction of `R` as part of the learning process. Many have used statistical packages or spreadsheets as tools for teaching statistics. Others have used `R` in advanced courses. I am not aware of attempts to use `R` in introductory level courses. Indeed, mastering `R` requires much investment of time and energy that may be distracting and counterproductive for learning more fundamental issues. Yet, I believe that if one restricts the application of `R` to a limited number of commands, the benefits that `R` provides outweigh the difficulties that `R` engenders.

Another departure from the standard approach is the treatment of probability as part of the course. In this book I do not attempt to teach probability as a subject matter, but only specific elements of it which I feel are essential for understanding statistics. Hence, Kolmogorov's Axioms are out as well as attempts to prove basic theorems and a Balls and Urns type of discussion. On the other hand, emphasis is given to the notion of a *random variable* and, in that context, the *sample space*.

The first part of the book deals with descriptive statistics and provides probability concepts that are required for the interpretation of statistical inference. Statistical inference is the subject of the second part of the book.

The first chapter is a short introduction to statistics and probability. Students are required to have access to `R` right from the start. Instructions regarding the installation of `R` on a PC are provided.

The second chapter deals with data structures and variation. Chapter 3 provides numerical and graphical tools for presenting and summarizing the distribution of data.

The fundamentals of probability are treated in Chapters 4 to 7. The concept of a random variable is presented in Chapter 4 and examples of special types of random variables are discussed in Chapter 5. Chapter 6 deals with the Normal

random variable. Chapter 7 introduces sampling distribution and presents the Central Limit Theorem and the Law of Large Numbers. Chapter 8 summarizes the material of the first seven chapters and discusses it in the statistical context.

Chapter 9 starts the second part of the book and the discussion of statistical inference. It provides an overview of the topics that are presented in the subsequent chapter. The material of the first half is revisited.

Chapters 10 to 12 introduce the basic tools of statistical inference, namely point estimation, estimation with a confidence interval, and the testing of statistical hypothesis. All these concepts are demonstrated in the context of a single measurements.

Chapters 13 to 15 discuss inference that involve the comparison of two measurements. The context where these comparisons are carried out is that of regression that relates the distribution of a response to an explanatory variable. In Chapter 13 the response is numeric and the explanatory variable is a factor with two levels. In Chapter 14 both the response and the explanatory variable are numeric and in Chapter 15 the response in a factor with two levels.

Chapter 16 ends the book with the analysis of two case studies. These analyses require the application of the tools that are presented throughout the book.

This book was originally written for a pair of courses in the University of the People. As such, each part was restricted to 8 chapters. Due to lack of space, some important material, especially the concepts of correlation and statistical independence were omitted. In future versions of the book I hope to fill this gap.

Large portions of this book, mainly in the first chapters and some of the quizzes, are based on material from the online book "Collaborative Statistics" by Barbara Illowsky and Susan Dean (Connexions, March 2, 2010. `http://cnx.org/content/col10522/1.37/`). Most of the material was edited by this author, who is the only person responsible for any errors that where introduced in the process of editing.

Case studies that are presented in the second part of the book are taken from Rice Virtual Lab in Statistics can be found in their Case Studies section. The responsibility for mistakes in the analysis of the data, if such mistakes are found, are my own.

I would like to thank my mother Ruth who, apart from giving birth, feeding and educating me, has also helped to improve the pedagogical structure of this text. I would like to thank also Gary Engstrom for correcting many of the mistakes in English that I made.

Jerusalem, June 2011 *Benjamin Yakir*

Contents

Part I

Introduction to Statistics

Chapter 1

Introduction

1.1 Student Learning Objectives

This chapter introduces the basic concepts of statistics. Special attention is given to concepts that are used in the first part of this book, the part that deals with graphical and numeric statistical ways to describe data (descriptive statistics) as well as mathematical theory of probability that enables statisticians to draw conclusions from data.

The course applies the widely used freeware programming environment for statistical analysis, known as `R`. In this chapter we will discuss the installation of the program and present very basic features of that system.

By the end of this chapter, the student should be able to:

- Recognize key terms in statistics and probability.
- Install the `R` program on an accessible computer.
- Learn and apply a few basic operations of the computational system `R`.

1.2 Why Learn Statistics?

You are probably asking yourself the question, "When and where will I use statistics?". If you read any newspaper or watch television, or use the Internet, you will see statistical information. There are statistics about crime, sports, education, politics, and real estate. Typically, when you read a newspaper article or watch a news program on television, you are given sample information. With this information, you may make a decision about the correctness of a statement, claim, or "fact". Statistical methods can help you make the "best educated guess".

Since you will undoubtedly be given statistical information at some point in your life, you need to know some techniques to analyze the information thoughtfully. Think about buying a house or managing a budget. Think about your chosen profession. The fields of economics, business, psychology, education, biology, law, computer science, police science, and early childhood development require at least one course in statistics.

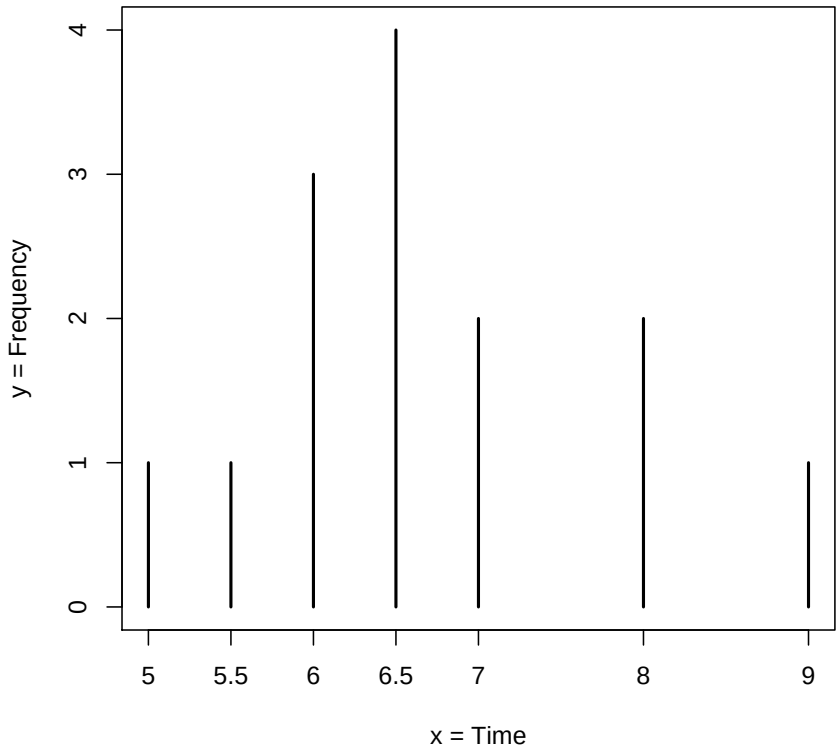

Figure 1.1: Frequency of Average Time (in Hours) Spent Sleeping per Night

Included in this chapter are the basic ideas and words of probability and statistics. In the process of learning the first part of the book, and more so in the second part of the book, you will understand that statistics and probability work together.

1.3 Statistics

The science of statistics deals with the collection, analysis, interpretation, and presentation of data. We see and use data in our everyday lives. To be able to use data correctly is essential to many professions and is in your own best self-interest.

For example, assume the average time (in hours, to the nearest half-hour) a group of people sleep per night has been recorded. Consider the following data:

$$5,\ 5.5,\ 6,\ 6,\ 6,\ 6.5,\ 6.5,\ 6.5,\ 6.5,\ 7,\ 7,\ 8,\ 8,\ 9\ .$$

In Figure 1.1 this data is presented in a graphical form (called a bar plot). A bar plot consists of a number axis (the x-axis) and bars (vertical lines) positioned

above the number axis. The length of each bar corresponds to the number of data points that obtain the given numerical value. In the given plot the frequency of average time (in hours) spent sleeping per night is presented with hours of sleep on the horizontal x-axis and frequency on vertical y-axis.

Think of the following questions:

- Would the bar plot constructed from data collected from a different group of people look the same as or different from the example? Why?
- If one would have carried the same example in a different group with the same size and age as the one used for the example, do you think the results would be the same? Why or why not?
- Where does the data appear to cluster? How could you interpret the clustering?

The questions above ask you to analyze and interpret your data. With this example, you have begun your study of statistics.

In this course, you will learn how to organize and summarize data. Organizing and summarizing data is called descriptive statistics. Two ways to summarize data are by graphing and by numbers (for example, finding an average). In the second part of the book you will also learn how to use formal methods for drawing conclusions from "good" data. The formal methods are called inferential statistics. Statistical inference uses probabilistic concepts to determine if conclusions drawn are reliable or not.

Effective interpretation of data is based on good procedures for producing data and thoughtful examination of the data. In the process of learning how to interpret data you will probably encounter what may seem to be too many mathematical formulae that describe these procedures. However, you should always remember that the goal of statistics is not to perform numerous calculations using the formulae, but to gain an understanding of your data. The calculations can be done using a calculator or a computer. The understanding must come from you. If you can thoroughly grasp the basics of statistics, you can be more confident in the decisions you make in life.

1.4 Probability

Probability is the mathematical theory used to study uncertainty. It provides tools for the formalization and quantification of the notion of uncertainty. In particular, it deals with the chance of an event occurring. For example, if the different potential outcomes of an experiment are equally likely to occur then the probability of each outcome is taken to be the reciprocal of the number of potential outcomes. As an illustration, consider tossing a fair coin. There are two possible outcomes – a head or a tail – and the probability of each outcome is 1/2.

If you toss a fair coin 4 times, the outcomes may not necessarily be 2 heads and 2 tails. However, if you toss the same coin 4,000 times, the outcomes will be close to 2,000 heads and 2,000 tails. It is very unlikely to obtain more than 2,060 tails and it is similarly unlikely to obtain less than 1,940 tails. This is consistent with the expected theoretical probability of heads in any one toss. Even though the outcomes of a few repetitions are uncertain, there is a regular

pattern of outcomes when the number of repetitions is large. Statistics exploits this pattern regularity in order to make extrapolations from the observed sample to the entire population.

The theory of probability began with the study of games of chance such as poker. Today, probability is used to predict the likelihood of an earthquake, of rain, or whether you will get an "A" in this course. Doctors use probability to determine the chance of a vaccination causing the disease the vaccination is supposed to prevent. A stockbroker uses probability to determine the rate of return on a client's investments. You might use probability to decide to buy a lottery ticket or not.

Although probability is instrumental for the development of the theory of statistics, in this introductory course we will not develop the mathematical theory of probability. Instead, we will concentrate on the philosophical aspects of the theory and use computerized simulations in order to demonstrate probabilistic computations that are applied in statistical inference.

1.5 Key Terms

In statistics, we generally want to study a population. You can think of a population as an entire collection of persons, things, or objects under study. To study the larger population, we select a sample. The idea of sampling is to select a portion (or subset) of the larger population and study that portion (the sample) to gain information about the population. Data are the result of sampling from a population.

Because it takes a lot of time and money to examine an entire population, sampling is a very practical technique. If you wished to compute the overall grade point average at your school, it would make sense to select a sample of students who attend the school. The data collected from the sample would be the students' grade point averages. In presidential elections, opinion poll samples of 1,000 to 2,000 people are taken. The opinion poll is supposed to represent the views of the people in the entire country. Manufacturers of canned carbonated drinks take samples to determine if the manufactured 16 ounce containers does indeed contain 16 ounces of the drink.

From the sample data, we can calculate a *statistic*. A statistic is a number that is a property of the sample. For example, if we consider one math class to be a sample of the population of all math classes, then the average number of points earned by students in that one math class at the end of the term is an example of a statistic. The statistic can be used as an estimate of a population *parameter*. A parameter is a number that is a property of the population. Since we considered all math classes to be the population, then the average number of points earned per student over all the math classes is an example of a parameter.

One of the main concerns in the field of statistics is how accurately a statistic estimates a parameter. The accuracy really depends on how well the sample represents the population. The sample must contain the characteristics of the population in order to be a representative sample.

Two words that come up often in statistics are *average* and *proportion*. If you were to take three exams in your math classes and obtained scores of 86, 75, and 92, you calculate your average score by adding the three exam scores and dividing by three (your average score would be 84.3 to one decimal place). If, in

your math class, there are 40 students and 22 are men and 18 are women, then the proportion of men students is 22/40 and the proportion of women students is 18/40. Average and proportion are discussed in more detail in later chapters.

1.6 The R Programming Environment

The R Programming Environment is a widely used open source system for statistical analysis and statistical programming. It includes thousands of functions for the implementation of both standard and exotic statistical methods and it is probably the most popular system in the academic world for the development of new statistical tools. We will use R in order to apply the statistical methods that will be discussed in the book to some example data sets and in order to demonstrate, via simulations, concepts associated with probability and its application in statistics.

The demonstrations in the book involve very basic R programming skills and the applications are implemented using, in most cases, simple and natural code. A detailed explanation will accompany the code that is used.

Learning R, like the learning of any other programming language, can be achieved only through practice. Hence, we strongly recommend that you not only read the code presented in the book but also run it yourself, in parallel to the reading of the provided explanations. Moreover, you are encouraged to play with the code: introduce changes in the code and in the data and see how the output changes as a result. One should not be afraid to experiment. At worst, the computer may crash or freeze. In both cases, restarting the computer will solve the problem ...

You may download R from the R project home page `http://www.r-project.org` and install it on the computer that you are using[1].

1.6.1 Some Basic R Commands

R is an object-oriented programming system. During the session you may create and manipulate objects by the use of functions that are part of the basic installation. You may also use the R programming language. Most of the functions that are part of the system are themselves written in the R language and one may easily write new functions or modify existing functions to suit specific needs.

Let us start by opening the `R Console` window by double-clicking on the R icon. Type in the `R Console` window, immediately after the "`>`" prompt, the expression "`1+2`" and then hit the Return key. (Do not include the double quotation in the expression that you type!):

```
> 1+2
[1] 3
>
```

The prompt "`>`" indicates that the system is ready to receive commands. Writing an expression, such as "`1+2`", and hitting the Return key sends the expression

[1] Detailed explanation of how to install the system on an XP Windows Operating System may be found here: `http://pluto.huji.ac.il/~msby/StatThink/install_R_WinXP.html`.

to be executed. The execution of the expression may produce an object, in this case an object that is composed of a single number, the number "3".

Whenever required, the R system takes an action. If no other specifications are given regarding the required action then the system will apply the pre-programmed action. This action is called the *default* action. In the case of hitting the Return key after the expression that we wrote the default is to display the produced object on the screen.

Next, let us demonstrate R in a more meaningful way by using it in order to produce the bar-plot of Figure 1.1. First we have to input the data. We will produce a sequence of numbers that form the data[2]. For that we will use the function "c" that combines its arguments and produces a sequence with the arguments as the components of the sequence. Write the expression:

```
> c(5,5.5,6,6,6,6.5,6.5,6.5,6.5,7,7,8,8,9)
```

at the prompt and hit return. The result should look like this:

```
> c(5,5.5,6,6,6,6.5,6.5,6.5,6.5,7,7,8,8,9)
 [1] 5.0 5.5 6.0 6.0 6.0 6.5 6.5 6.5 6.5 7.0 7.0 8.0 8.0 9.0
>
```

The function "c" is an example of an R function. A function has a name, "c" in this case, that is followed by brackets that include the input to the function. We call the components of the input the *arguments* of the function. Arguments are separated by commas. A function produces an output, which is typically an R object. In the current example an object of the form of a sequence was created and, according to the default application of the system, was sent to the screen and not saved.

If we want to create an object for further manipulation then we should save it and give it a name. For example, it we want to save the vector of data under the name "X" we may write the following expression at the prompt (and then hit return):

```
> X <- c(5,5.5,6,6,6,6.5,6.5,6.5,6.5,7,7,8,8,9)
>
```

The arrow that appears after the "X" is produced by typing the less than key "<" followed by the minus key "-". This arrow is the assignment operator.

Observe that you may save typing by calling and editing lines of code that were processes in an earlier part of the session. One may browse through the lines using the up and down arrows on the right-hand side of the keyboard and use the right and left arrows to move along the line presented at the prompt. For example, the last expression may be produced by finding first the line that used the function "c" with the up and down arrow and then moving to the beginning of the line with the left arrow. At the beginning of the line all one has to do is type "X <- " and hit the Return key.

Notice that no output was sent to the screen. Instead, the output from the "c" function was assigned to an object that has the name "X". A new object by the given name was formed and it is now available for further analysis. In order to verify this you may write "X" at the prompt and hit return:

[2] In R, a sequence of numbers is called a *vector*. However, we will use the term *sequence* to refer to vectors.

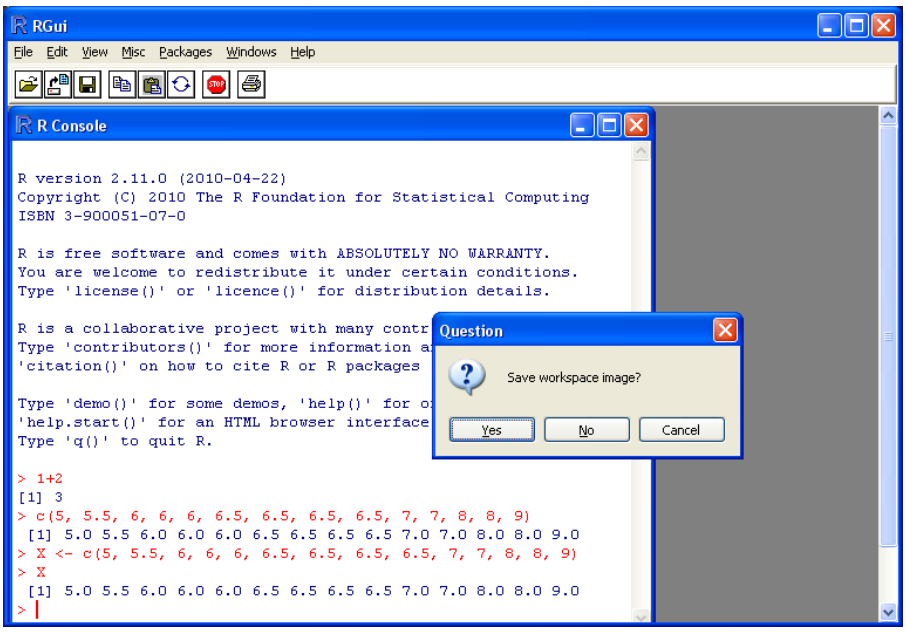

Figure 1.2: Save Workspace Dialog

```
> X
 [1] 5.0 5.5 6.0 6.0 6.0 6.5 6.5 6.5 6.5 7.0 7.0 8.0 8.0 9.0
```

The content of the object "X" is sent to the screen, which is the default output. Notice that we have not changed the given object, which is still in the memory.

The object "X" is in the memory, but it is not saved on the hard disk. With the end of the session the objects created in the session are erased unless specifically saved. The saving of all the objects that were created during the session can be done when the session is finished. Hence, when you close the R Console window a dialog box will open (See the screenshot in Figure 1.2). Via this dialog box you can choose to save the objects that were created in the session by selecting "Yes", not to save by selecting the option "No", or you may decide to abort the process of shutting down the session by selecting "Cancel". If you save the objects then they will be uploaded to the memory the next time that the R Console is opened.

We used a capital letter to name the object. We could have used a small letter just as well or practically any combination of letters. However, you should note that R distinguishes between capital and small letter. Hence, typing "x" in the console window and hitting return will produce an error message:

```
> x
Error: object "x" not found
```

An object named "x" does not exist in the R system and we have not created such object. The object "X", on the other hand, does exist.

Names of functions that are part of the system are fixed but you are free to choose a name to objects that you create. For example, if one wants to create

an object by the name "`my.vector`" that contains the numbers 3, 7, 3, 3, and -5 then one may write the expression "`my.vector <- c(3,7,3,3,-5)`" at the prompt and hit the Return key.

If we want to produce a table that contains a count of the frequency of the different values in our data we can apply the function "`table`" to the object "`X`" (which is the object that contains our data):

```
> table(X)
X
  5 5.5   6 6.5   7   8   9
  1   1   3   4   2   2   1
```

Notice that the output of the function "`table`" is a table of the different levels of the input vector and the frequency of each level. This output is yet another type of an object.

The bar-plot of Figure 1.1 can be produced by the application of the function "`plot`" to the object that is produced as an output of the function "`table`":

```
> plot(table(X))
```

Observe that a graphical window was opened with the target plot. The plot that appears in the graphical window should coincide with the plot in Figure 1.3. This plot is practically identical to the plot in Figure 1.1. The only difference is in the names given to the access. These names were changed in Figure 1.1 for clarity.

Clearly, if one wants to produce a bar-plot to other numerical data all one has to do is replace in the expression "`plot(table(X))`" the object "`X`" by an object that contains the other data. For example, to plot the data in "`my.vector`" you may use "`plot(table(my.vector))`".

1.7 Solved Exercises

Question 1.1. A potential candidate for a political position in some state is interested to know what are her chances to win the primaries of her party and be selected as parties candidate for the position. In order to examine the opinions of her party voters she hires the services of a polling agency. The polling is conducted among 500 registered voters of the party. One of the questions that the pollsters refers to the willingness of the voters to vote for a female candidate for the job. Forty two percent of the people asked said that they prefer to have a women running for the job. Thirty eight percent said that the candidate's gender is irrelevant. The rest prefers a male candidate. Which of the following is (i) a population (ii) a sample (iii) a parameter and (iv) a statistic:

1. The 500 registered voters.
2. The percentage, among all registered voters of the given party, of those that prefer a male candidate.
3. The number 42% that corresponds to the percentage of those that prefer a female candidate.
4. The voters in the state that are registered to the given party.

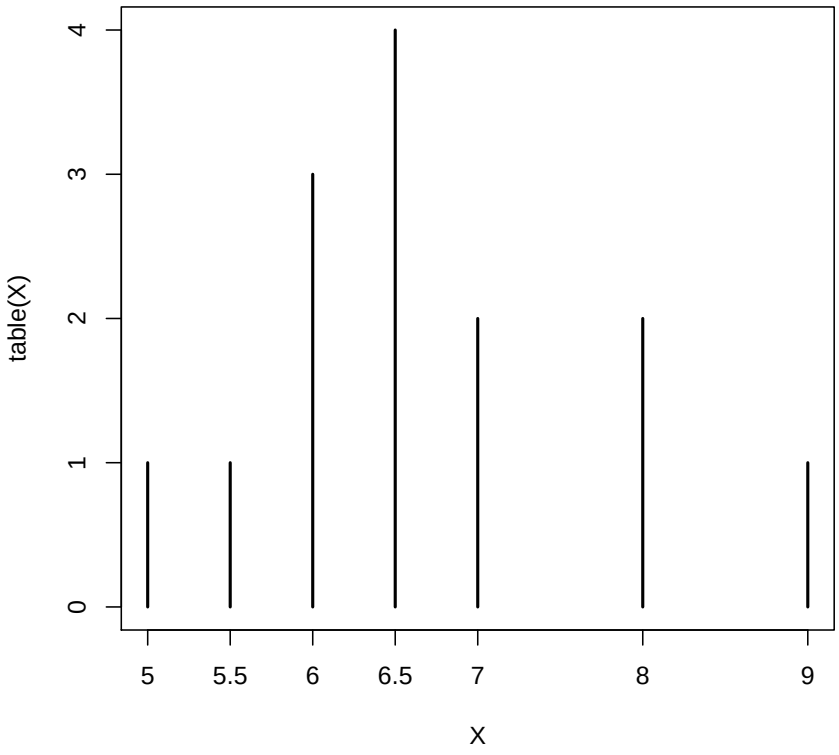

Figure 1.3: The Plot Produced by the Expression "`plot(table(X))`"

Solution (to Question 1.1.1): According to the information in the question the polling was conducted among 500 registered voters. The 500 registered voters corresponds to the sample.

Solution (to Question 1.1.2): The percentage, among all registered voters of the given party, of those that prefer a male candidate is a parameter. This quantity is a characteristic of the population.

Solution (to Question 1.1.3): It is given that 42% of the sample prefer a female candidate. This quantity is a numerical characteristic of the data, of the sample. Hence, it is a statistic.

Solution (to Question 1.1.4): The voters in the state that are registered to the given party is the target population.

Question 1.2. The number of customers that wait in front of a coffee shop at the opening was reported during 25 days. The results were:

$$4, 2, 1, 1, 0, 2, 1, 2, 4, 2, 5, 3, 1, 5, 1, 5, 1, 2, 1, 1, 3, 4, 2, 4, 3\ .$$

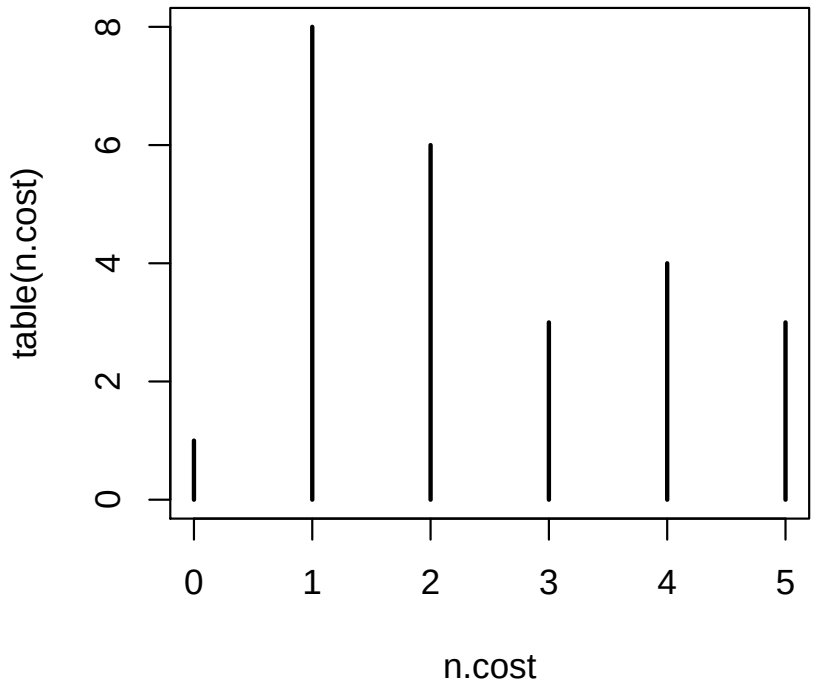

Figure 1.4: The Plot Produced by the Expression "`plot(table(n.cost))`"

1. Identify the number of days in which 5 costumers where waiting.
2. The number of waiting costumers that occurred the largest number of times.
3. The number of waiting costumers that occurred the least number of times.

Solution (to Question 1.2): One may read the data into `R` and create a table using the code:

```
> n.cost <- c(4,2,1,1,0,2,1,2,4,2,5,3,1,5,1,5,1,2,1,1,3,4,2,4,3)
> table(n.cost)
n.cost
0 1 2 3 4 5
1 8 6 3 4 3
```

For convenience, one may also create the bar plot of the data using the code:

```
> plot(table(n.cost))
```

The bar plot is presented in Figure 1.4.

Solution (to Question 1.2.1): The number of days in which 5 costumers where waiting is 3, since the frequency of the value "5" in the data is 3. That can be seen from the table by noticing the number below value "5" is 3. It can also be seen from the bar plot by observing that the hight of the bar above the value "5" is equal to 3.

Solution (to Question 1.2.2): The number of waiting costumers that occurred the largest number of times is 1. The value "1" occurred 8 times, more than any other value. Notice that the bar above this value is the highest.

Solution (to Question 1.2.3): The value "0", which occurred only once, occurred the least number of times.

1.8 Summary

Glossary

Data: A set of observations taken on a sample from a population.

Statistic: A numerical characteristic of the data. A statistic estimates the corresponding population parameter. For example, the average number of contribution to the course's forum for this term is an estimate for the average number of contributions in all future terms (parameter).

Statistics The science that deals with processing, presentation and inference from data.

Probability: A mathematical field that models and investigates the notion of randomness.

Discuss in the forum

A sample is a subgroup of the population that is supposed to represent the entire population. In your opinion, is it appropriate to attempt to represent the entire population only by a sample?

When you formulate your answer to this question it may be useful to come up with an example of a question from you own field of interest one may want to investigate. In the context of this example you may identify a target population which you think is suited for the investigation of the given question. The appropriateness of using a sample can be discussed in the context of the example question and the population you have identified.

Chapter 2

Sampling and Data Structures

2.1 Student Learning Objectives

In this chapter we deal with issues associated with the data that is obtained from a sample. The variability associated with this data is emphasized and critical thinking about validity of the data encouraged. A method for the introduction of data from an external source into `R` is proposed and the data types used by `R` for storage are described. By the end of this chapter, the student should be able to:

- Recognize potential difficulties with sampled data.
- Read an external data file into `R`.
- Create and interpret frequency tables.

2.2 The Sampled Data

The aim in statistics is to learn the characteristics of a population on the basis of a sample selected from the population. An essential part of this analysis involves consideration of variation in the data.

2.2.1 Variation in Data

Variation is given a central role in statistics. To some extent the assessment of variation and the quantification of its contribution to uncertainties in making inference is the statistician's main concern.

Variation is present in any set of data. For example, 16-ounce cans of beverage may contain more or less than 16 ounces of liquid. In one study, eight 16 ounce cans were measured and produced the following amount (in ounces) of beverage:

$$15.8,\ 16.1,\ 15.2,\ 14.8,\ 15.8,\ 15.9,\ 16.0,\ 15.5\ .$$

Measurements of the amount of beverage in a 16-ounce may vary because the conditions of measurement varied or because the exact amount, 16 ounces of

liquid, was not put into the cans. Manufacturers regularly run tests to determine if the amount of beverage in a 16-ounce can falls within the desired range.

Be aware that if an investigator collects data, the data may vary somewhat from the data someone else is taking for the same purpose. This is completely natural. However, if two investigators or more, are taking data from the same source and get very different results, it is time for them to reevaluate their data-collection methods and data recording accuracy.

2.2.2 Variation in Samples

Two or more samples from the same population, all having the same characteristics as the population, may nonetheless be different from each other. Suppose Doreen and Jung both decide to study the average amount of time students sleep each night and use all students at their college as the population. Doreen may decide to sample randomly a given number of students from the entire body of collage students. Jung, on the other hand, may decide to sample randomly a given number of classes and survey all students in the selected classes. Doreen's method is called *random sampling* whereas Jung's method is called *cluster sampling*. Doreen's sample will be different from Jung's sample even though both samples have the characteristics of the population. Even if Doreen and Jung used the same sampling method, in all likelihood their samples would be different. Neither would be wrong, however.

If Doreen and Jung took larger samples (i.e. the number of data values is increased), their sample results (say, the average amount of time a student sleeps) would be closer to the actual population average. But still, their samples would be, most probably, different from each other.

The size of a sample (often called the number of observations) is important. The examples you have seen in this book so far have been small. Samples of only a few hundred observations, or even smaller, are sufficient for many purposes. In polling, samples that are from 1200 to 1500 observations are considered large enough and good enough if the survey is random and is well done. The theory of statistical inference, that is the subject matter of the second part of this book, provides justification for these claims.

2.2.3 Frequency

The primary way of summarizing the variability of data is via the frequency distribution. Consider an example. Twenty students were asked how many hours they worked per day. Their responses, in hours, are listed below:

$$5,\ 6,\ 3,\ 3,\ 2,\ 4,\ 7,\ 5,\ 2,\ 3,\ 5,\ 6,\ 5,\ 4,\ 4,\ 3,\ 5,\ 2,\ 5,\ 3\ .$$

Let us create an `R` object by the name "`work.hours`" that contains these data:

```
> work.hours <- c(5,6,3,3,2,4,7,5,2,3,5,6,5,4,4,3,5,2,5,3)
```

Next, let us create a table that summarizes the different values of working hours and the frequency in which these values appear in the data:

```
> table(work.hours)
work.hours
2 3 4 5 6 7
3 5 3 6 2 1
```

Recall that the function "`table`" takes as input a sequence of data and produces as output the frequencies of the different values.

We may have a clearer understanding of the meaning of the output of the function "`table`" if we presented outcome as a frequency listing the different data values in ascending order and their frequencies. For that end we may apply the function "`data.frame`" to the output of the "`table`" function and obtain:

```
> data.frame(table(work.hours))
  work.hours Freq
2          2    3
3          3    5
4          4    3
5          5    6
6          6    2
7          7    1
```

A frequency is the number of times a given datum occurs in a data set. According to the table above, there are three students who work 2 hours, five students who work 3 hours, etc. The total of the frequency column, 20, represents the total number of students included in the sample.

The function "`data.frame`" transforms its input into a data frame, which is the standard way of storing statistical data. We will introduce data frames in more detail in Section 2.3 below.

A relative frequency is the fraction of times a value occurs. To find the relative frequencies, divide each frequency by the total number of students in the sample – 20 in this case. Relative frequencies can be written as fractions, percents, or decimals.

As an illustration let us compute the relative frequencies in our data:

```
> freq <- table(work.hours)
> freq
work.hours
2 3 4 5 6 7
3 5 3 6 2 1
> sum(freq)
[1] 20
> freq/sum(freq)
work.hours
   2    3    4    5    6    7
0.15 0.25 0.15 0.30 0.10 0.05
```

We stored the frequencies in an object called "`freq`". The content of the object are the frequencies 3, 5, 3, 6, 2 and 1. The function "`sum`" sums the components of its input. The sum of the frequencies is the sample size , the total number of students that responded to the survey, which is 20. Hence, when we apply the function "`sum`" to the object "`freq`" we get 20 as an output.

The outcome of dividing an object by a number is a division of each element in the object by the given number. Therefore, when we divide "`freq`" by "`sum(freq)`" (the number 20) we get a sequence of relative frequencies. The first entry to this sequence is $3/20 = 0.15$, the second entry is $5/20 = 0.25$, and the last entry is $1/20 = 0.05$. The sum of the relative frequencies should always be equal to 1:

```
> sum(freq/sum(freq))
[1] 1
```

The cumulative relative frequency is the accumulation of previous relative frequencies. To find the cumulative relative frequencies, add all the previous relative frequencies to the relative frequency of the current value. Alternatively, we may apply the function "`cumsum`" to the sequence of relative frequencies:

```
> cumsum(freq/sum(freq))
   2    3    4    5    6    7
0.15 0.40 0.55 0.85 0.95 1.00
```

Observe that the cumulative relative frequency of the smallest value 2 is the frequency of that value (0.15). The cumulative relative frequency of the second value 3 is the sum of the relative frequency of the smaller value (0.15) and the relative frequency of the current value (0.25), which produces a total of $0.15 + 0.25 = 0.40$. Likewise, for the third value 4 we get a cumulative relative frequency of $0.15 + 0.25 + 0.15 = 0.55$. The last entry of the cumulative relative frequency column is one, indicating that one hundred percent of the data has been accumulated.

The computation of the cumulative relative frequency was carried out with the aid of the function "`cumsum`". This function takes as an input argument a numerical sequence and produces as output a numerical sequence of the same length with the cumulative sums of the components of the input sequence.

2.2.4 Critical Evaluation

Inappropriate methods of sampling and data collection may produce samples that do not represent the target population. A naïve application of statistical analysis to such data may produce misleading conclusions.

Consequently, it is important to evaluate critically the statistical analyses we encounter before accepting the conclusions that are obtained as a result of these analyses. Common problems that occurs in data that one should be aware of include:

Problems with Samples: A sample should be representative of the population. A sample that is not representative of the population is biased. Biased samples may produce results that are inaccurate and not valid.

Data Quality: Avoidable errors may be introduced to the data via inaccurate handling of forms, mistakes in the input of data, etc. Data should be cleaned from such errors as much as possible.

Self-Selected Samples: Responses only by people who choose to respond, such as call-in surveys, that are often biased.

Sample Size Issues: Samples that are too small may be unreliable. Larger samples, when possible, are better. In some situations, small samples are unavoidable and can still be used to draw conclusions. Examples: Crash testing cars, medical testing for rare conditions.

Undue Influence: Collecting data or asking questions in a way that influences the response.

Causality: A relationship between two variables does not mean that one causes the other to occur. They may both be related (correlated) because of their relationship to a third variable.

Self-Funded or Self-Interest Studies: A study performed by a person or organization in order to support their claim. Is the study impartial? Read the study carefully to evaluate the work. Do not automatically assume that the study is good but do not automatically assume the study is bad either. Evaluate it on its merits and the work done.

Misleading Use of Data: Improperly displayed graphs and incomplete data.

Confounding: Confounding in this context means confusing. When the effects of multiple factors on a response cannot be separated. Confounding makes it difficult or impossible to draw valid conclusions about the effect of each factor.

2.3 Reading Data into R

In the examples so far the size of the data set was very small and we were able to input the data directly into R with the use of the function "c". In more practical settings the data sets to be analyzed are much larger and it is very inefficient to enter them manually. In this section we learn how to upload data from a file in the Comma Separated Values (CSV) format.

The file "ex1.csv" contains data on the sex and height of 100 individuals. This file is given in the CSV format. The file can be found on the internet at http://pluto.huji.ac.il/~msby/StatThink/Datasets/ex1.csv. We will discuss the process of reading data from a file into R and use this file as an illustration.

2.3.1 Saving the File and Setting the Working Directory

Before the file is read into R you may find it convenient to obtain a copy of the file and store it in some directory on the computer and read the file from that directory. We recommend that you create a special directory in which you keep all the material associated with this course. In the explanations provided below we assume that the directory to which the file is stored in called "IntroStat". (See Figure 2.1)

Files in the CSV format are ordinary text files. They can be created manually or as a result of converting data stored in a different format into this particular format. A convenient way to produce, browse and edit CSV files is by the use of a standard electronic spreadsheet programs such as Excel or Calc. The Excel spreadsheet is part of the Microsoft's Office suite. The Calc spreadsheet is part of OpenOffice suite that is freely distributed by the OpenOffice Organization.

Opening a CSV file by a spreadsheet program displays a spreadsheet with the content of the file. Values in the cells of the spreadsheet may be modified directly. (However, when saving, one should pay attention to save the file in the CVS format.) Similarly, new CSV files may be created by the entering of the data in an empty spreadsheet. The first row should include the name of the variable, preferably as a single character string with no empty spaces. The

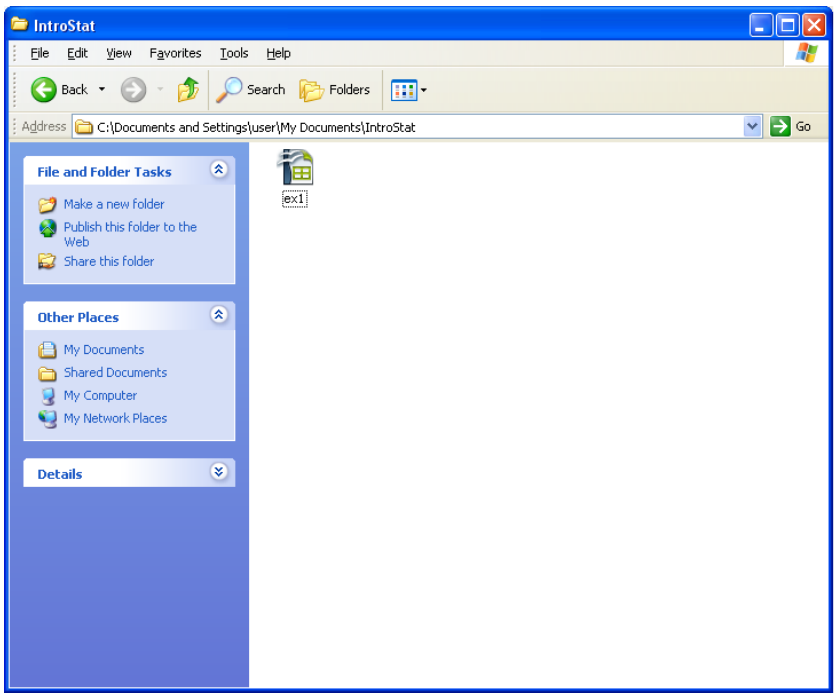

Figure 2.1: The File "`read.csv`"

following rows may contain the data values associated with this variable. When saving, the spreadsheet should be saved in the CSV format by the use of the "Save by name" dialog and choosing there the option of CSV in the "`Save by Type`" selection.

After saving a file with the data in a directory, `R` should be notified where the file is located in order to be able to read it. A simple way of doing so is by setting the directory with the file as `R`'s *working directory*. The working directory is the first place `R` is searching for files. Files produced by `R` are saved in that directory. In Windows, during an active `R` session, one may set the working directory to be some target directory with the "`File/Change Dir...`" dialog. This dialog is opened by selecting the option "`File`" on the left hand side of the ruler on the top of the `R Console` window. Selecting the option of "`Change Dir...`" in the ruler that opens will start the dialog. (See Figure 2.2.) Browsing via this dialog window to the directory of choice, selecting it, and approving the selection by clicking the "`OK`" bottom in the dialog window will set the directory of choice as the working directory of `R`.

Rather than changing the working directory every time that `R` is opened one may set a selected directory to be `R`'s working directory on opening. Again, we demonstrate how to do this on the XP Windows operating system.

The `R` icon was added to the Desktop when the `R` system was installed. The `R Console` is opened by double-clicking on this icon. One may change the properties of the icon so that it sets a directory of choice as `R`'s working directory.

In order to do so click on the icon with the mouse's **right** bottom. A menu

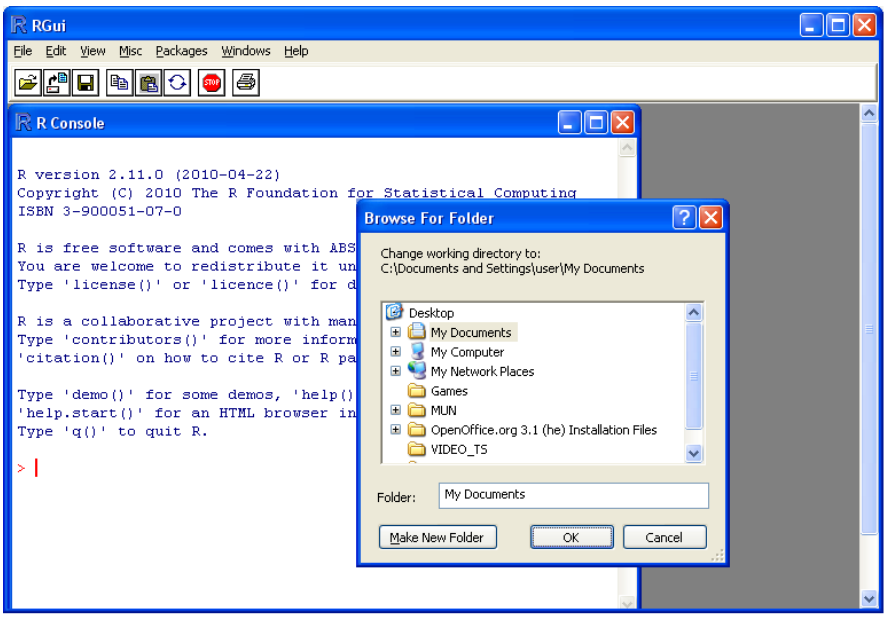

Figure 2.2: Changing The Working Directory

opens in which you should select the option "`Properties`". As a result, a dialog window opens. (See Figure 2.3.) Look at the line that starts with the words "`Start in`" and continues with a name of a directory that is the current working directory. The name of this directory is enclosed in double quotes and is given with it's full path, i.e. its address on the computer. This name and path should be changed to the name and path of the directory that you want to fix as the new working directory.

Consider again Figure 2.1. Imagine that one wants to fix the directory that contains the file "`ex1.csv`" as the permanent working directory. Notice that the full address of the directory appears at the "`Address`" bar on the top of the window. One may copy the address and paste it instead of the name of the current working directory that is specified in the "`Properties`" dialog of the `R` icon. One should make sure that the address to the new directory is, again, placed between double-quotes. (See in Figure 2.4 the dialog window after the changing the address of the working directory. Compare this to Figure 2.3 of the window before the change.) After approving the change by clicking the "`OK`" bottom the new working directory is set. Henceforth, each time that the `R Console` is opened by double-clicking the icon it will have the designated directory as its working directory.

In the rest of this book we assume that a designated directory is set as `R`'s working directory and that all external files that need to be read into `R`, such as "`ex1.csv`" for example, are saved in that working directory. Once a working directory has been set then the history of subsequent `R` sessions is stored in that directory. Hence, if you choose to save the image of the session when you end the session then objects created in the session will be uploaded the next time

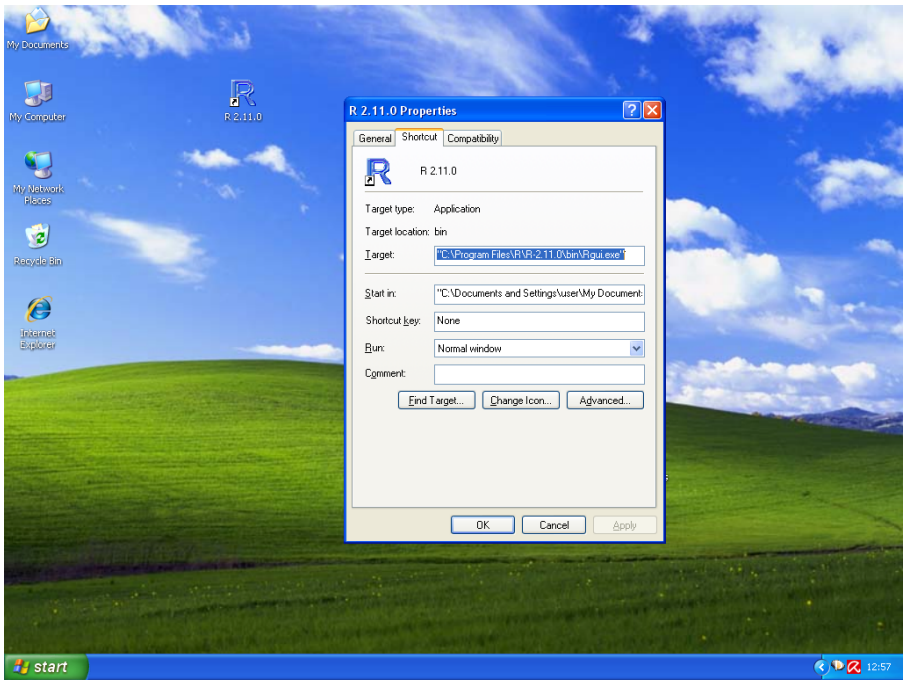

Figure 2.3: Setting the Working Directory (Before the Change)

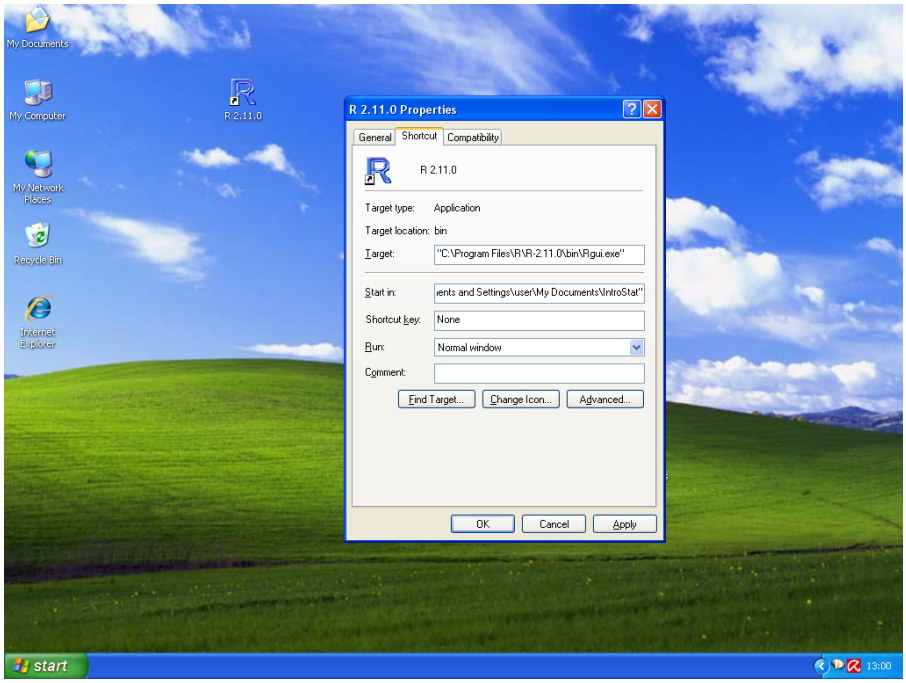

Figure 2.4: Setting the Working Directory (After the Change)

the R Console is opened.

2.3.2 Reading a CSV File into R

Now that a copy of the file "ex1.csv" is placed in the working directory we would like to read its content into R. Reading of files in the CSV format can be carried out with the R function "read.csv". To read the file of the example we run the following line of code in the R Console window:

```
> ex.1 <- read.csv("ex1.csv")
```

The function "read.csv" takes as an input argument the address of a CSV file and produces as output a *data frame* object with the content of the file. Notice that the address is placed between double-quotes. If the file is located in the working directory then giving the name of the file as an address is sufficient[1].

Consider the content of that R object "ex.1" that was created by the previous expression:

```
> ex.1
         id    sex height
1   5696379 FEMALE   182
2   3019088   MALE   168
3   2038883   MALE   172
4   1920587 FEMALE   154
5   6006813   MALE   174
6   4055945 FEMALE   176
.       .      .      .
.       .      .      .
.       .      .      .
98  9383288   MALE   195
99  1582961 FEMALE   129
100 9805356   MALE   172
>
```

(Noticed that we have erased the middle rows. In the R Console window you should obtain the full table. However, in order to see the upper part of the output you may need to scroll up the window.)

The object "ex.1", the output of the function "read.csv" is a *data frame*. Data frames are the standard tabular format of storing statistical data. The columns of the table are called *variables* and correspond to measurements. In this example the three variables are:

id: A 7 digits number that serves as a unique identifier of the subject.

sex: The sex of each subject. The values are either "MALE" or "FEMALE".

height: The height (in centimeter) of each subject. A numerical value.

[1] If the file is located in a different directory then the complete address, including the path to the file, should be provided. The file need not reside on the computer. One may provide, for example, a URL (an internet address) as the address. Thus, instead of saving the file of the example on the computer one may read its content into an R object by using the line of code "ex.1 <- read.csv("http://pluto.huji.ac.il/~msby/StatThink/Datasets/ex1.csv")" instead of the code that we provide and the working method that we recommend to follow.

When the values of the variable are numerical we say that it is a *quantitative variable* or a *numeric variable*. On the other hand, if the variable has qualitative or level values we say that it is a *factor*. In the given example, `sex` is a factor and `height` is a numeric variable.

The rows of the table are called *observations* and correspond to the subjects. In this data set there are 100 subjects, with subject number 1, for example, being a female of height 182 cm and identifying number 5696379. Subject number 98, on the other hand, is a male of height 195 cm and identifying number 9383288.

2.3.3 Data Types

The columns of `R` data frames represent variables, i.e. measurements recorded for each of the subjects in the sample. `R` associates with each variable a type that characterizes the content of the variable. The two major types are

- Factors, or Qualitative Data. The type is "`factor`".
- Quantitative Data. The type is "`numeric`".

Factors are the result of categorizing or describing attributes of a population. Hair color, blood type, ethnic group, the car a person drives, and the street a person lives on are examples of qualitative data. Qualitative data are generally described by words or letters. For instance, hair color might be black, dark brown, light brown, blonde, gray, or red. Blood type might be AB+, O-, or B+. Qualitative data are not as widely used as quantitative data because many numerical techniques do not apply to the qualitative data. For example, it does not make sense to find an average hair color or blood type.

Quantitative data are always numbers and are usually the data of choice because there are many methods available for analyzing such data. Quantitative data are the result of counting or measuring attributes of a population. Amount of money, pulse rate, weight, number of people living in your town, and the number of students who take statistics are examples of quantitative data.

Quantitative data may be either discrete or continuous. All data that are the result of counting are called quantitative discrete data. These data take on only certain numerical values. If you count the number of phone calls you receive for each day of the week, you may get results such as 0, 1, 2, 3, etc. On the other hand, data that are the result of measuring on a continuous scale are quantitative continuous data, assuming that we can measure accurately. Measuring angles in radians may result in the numbers $\frac{\pi}{6}$, $\frac{\pi}{3}$, $\frac{\pi}{2}$, π, $\frac{3\pi}{4}$, etc. If you and your friends carry backpacks with books in them to school, the numbers of books in the backpacks are discrete data and the weights of the backpacks are continuous data.

Example 2.1 (Data Sample of Quantitative Discrete Data)**.** *The data are the number of books students carry in their backpacks. You sample five students. Two students carry 3 books, one student carries 4 books, one student carries 2 books, and one student carries 1 book. The numbers of books (3, 4, 2, and 1) are the quantitative discrete data.*

Example 2.2 (Data Sample of Quantitative Continuous Data)**.** *The data are the weights of the backpacks with the books in it. You sample the same five students. The weights (in pounds) of their backpacks are 6.2, 7, 6.8, 9.1, 4.3.*

Notice that backpacks carrying three books can have different weights. Weights are quantitative continuous data because weights are measured.

Example 2.3 (Data Sample of Qualitative Data)**.** *The data are the colors of backpacks. Again, you sample the same five students. One student has a red backpack, two students have black backpacks, one student has a green backpack, and one student has a gray backpack. The colors red, black, black, green, and gray are qualitative data.*

The distinction between continuous and discrete numeric data is not reflected usually in the statistical method that are used in order to analyze the data. Indeed, `R` does not distinguish between these two types of numeric data and store them both as "`numeric`". Consequently, we will also not worry about the specific categorization of numeric data and treat them as one. On the other hand, emphasis will be given to the difference between numeric and factors data.

One may collect data as numbers and report it categorically. For example, the quiz scores for each student are recorded throughout the term. At the end of the term, the quiz scores are reported as A, B, C, D, or F. On the other hand, one may code categories of qualitative data with numerical values and report the values. The resulting data should nonetheless be treated as a factor.

As default, `R` saves variables that contain non-numeric values as factors. Otherwise, the variables are saved as numeric. The variable type is important because different statistical methods are applied to different data types. Hence, one should make sure that the variables that are analyzed have the appropriate type. Especially that factors using numbers to denote the levels are labeled as factors. Otherwise `R` will treat them as quantitative data.

2.4 Solved Exercises

Question 2.1. Consider the following relative frequency table on hurricanes that have made direct hits on the U.S. between 1851 and 2004 (`http://www.nhc.noaa.gov/gifs/table5.gif`). Hurricanes are given a strength category rating based on the minimum wind speed generated by the storm. Some of the entries to the table are missing.

Category	# Direct Hits	Relative Freq.	Cum. Relative Freq.
1	109		
2	72	0.2637	0.6630
3		0.2601	
4	18		0.9890
5	3	0.0110	1.0000

Table 2.1: Frequency of Hurricane Direct Hits

1. What is the relative frequency of direct hits of category 1?
2. What is the relative frequency of direct hits of category 4 or more?

Solution (to Question 2.1.1): The relative frequency of direct hits of category 1 is 0.3993. Notice that the cumulative relative frequency of category

1 and 2 hits, the sum of the relative frequency of both categories, is 0.6630. The relative frequency of category 2 hits is 0.2637. Consequently, the relative frequency of direct hits of category 1 is 0.6630 - 0.2637 = 0.3993.

Solution (to Question 2.1.2): The relative frequency of direct hits of category 4 or more is 0.0769. Observe that the cumulative relative of the value "3" is 0.6630 + 0.2601 = 0.9231. This follows from the fact that the cumulative relative frequency of the value "2" is 0.6630 and the relative frequency of the value "3" is 0.2601. The total cumulative relative frequency is 1.0000. The relative frequency of direct hits of category 4 or more is the difference between the total cumulative relative frequency and cumulative relative frequency of 3 hits: 1.0000 - 0.9231 = 0.0769.

Question 2.2. The number of calves that were born to some cows during their productive years was recorded. The data was entered into an R object by the name "`calves`". Refer to the following `R` code:

```
> freq <- table(calves)
> cumsum(freq)
 1  2  3  4  5  6  7
 4  7 18 28 32 38 45
```

1. How many cows were involved in this study?

2. How many cows gave birth to a total of 4 calves?

3. What is the relative frequency of cows that gave birth to at least 4 calves?

Solution (to Question 2.2.1): The total number of cows that were involved in this study is 45. The object "`freq`" contain the table of frequency of the cows, divided according to the number of calves that they had. The cumulative frequency of all the cows that had 7 calves or less, which includes all cows in the study, is reported under the number "7" in the output of the expression "`cumsum(freq)`". This number is 45.

Solution (to Question 2.2.2): The number of cows that gave birth to a total of 4 calves is 10. Indeed, the cumulative frequency of cows that gave birth to 4 calves or less is 28. The cumulative frequency of cows that gave birth to 3 calves or less is 18. The frequency of cows that gave birth to exactly 4 calves is the difference between these two numbers: 28 - 18 = 10.

Solution (to Question 2.2.3): The relative frequency of cows that gave birth to at least 4 calves is 27/45 = 0.6. Notice that the cumulative frequency of cows that gave at most 3 calves is 18. The total number of cows is 45. Hence, the number of cows with 4 or more calves is the difference between these two numbers: 45 - 18 = 27. The relative frequency of such cows is the ratio between this number and the total number of cows: 27/45 = 0.6.

2.5 Summary

Glossary

Population: The collection, or set, of all individuals, objects, or measurements whose properties are being studied.

Sample: A portion of the population understudy. A sample is representative if it characterizes the population being studied.

Frequency: The number of times a value occurs in the data.

Relative Frequency: The ratio between the frequency and the size of data.

Cumulative Relative Frequency: The term applies to an ordered set of data values from smallest to largest. The cumulative relative frequency is the sum of the relative frequencies for all values that are less than or equal to the given value.

Data Frame: A tabular format for storing statistical data. Columns correspond to variables and rows correspond to observations.

Variable: A measurement that may be carried out over a collection of subjects. The outcome of the measurement may be numerical, which produces a quantitative variable; or it may be non-numeric, in which case a factor is produced.

Observation: The evaluation of a variable (or variables) for a given subject.

CSV Files: A digital format for storing data frames.

Factor: Qualitative data that is associated with categorization or the description of an attribute.

Quantitative: Data generated by numerical measurements.

Discuss in the forum

Factors are qualitative data that are associated with categorization or the description of an attribute. On the other hand, numeric data are generated by numerical measurements. A common practice is to code the levels of factors using numerical values. What do you think of this practice?

In the formulation of your answer to the question you may think of an example of factor variable from your own field of interest. You may describe a benefit or a disadvantage that results from the use of a numerical values to code the level of this factor.