Chapter 1

Introducing Statistics

1.1 Introduction

So far in this module, we have focused entirely on *probability theory* – the mathematical theory of chance. For the rest of this module, we will study Statistics, which will involve using many concepts from probability theory. But what is Statistics? The Oxford English Dictionary gives the definition

> The systematic collection and arrangement of numerical facts or data of any kind; (also) the branch of science or mathematics concerned with the analysis and interpretation of numerical data and appropriate ways of gathering such data.

It still may not be clear to you from this definition what the subject involves, so to be a little more precise, the following four tasks cover fairly well the sorts of things statisticians typically do.

1. **Summarising and visualising data**
   Suppose we have collected data on some quantity of interest, for example: exam marks of students on a particular module; number of reported burglaries in each town/city in the UK in 2016; lifetimes of patients on a new cancer drug. Imagine the data as a long list of numbers stored in a spreadsheet. How can we present such data in a way that makes it easily understandable? Can we describe, concisely, the main features of the data?

2. **Inference**
   Statistical inference is the process of drawing conclusions about the characteristics of a population given a sample of data from that population. For example, suppose two different methods of teaching children to read are compared in an experiment. Fifty children are assigned to each method, and are tested on their reading comprehension both before and after instruction with the their assigned method. How do we make conclusions about how well each method would work for *all* children, and not just the children in the study?

3. **Forecasting/prediction** Prediction is sometimes an extension of the inference problem. In the reading example above, given the data from 100 children in the experiment, we may

wish to predict how well each method would work on children in the future. In other cases, we observe data sequentially in time (e.g. stock prices, daily maximum temperatures, passenger numbers) and wish to forecast what will happen next. Forecasts will rarely be *precisely* wrong, and statistical methods can play a crucial role in quantifying uncertainty in such forecasts.

4. **Experimental design** Given an understanding of how to analyse and model random variation in data, we can think about what sort of data would be most informative, given the budget/practical constraints at hand: we can design experiments to 'minimise the disruption' of random variation and 'bias', and give the most amount of information.

1.2 Statistics in other guises

There are other disciplines which have much in common with Statistics (though usually with more interesting sounding names!) These relationships have two-way benefits: new methods developed in other disciplines can help with statistical problems, and statistical theory can help in other disciplines (and there are even more jobs requiring statistical expertise than you might realise!) Two important related disciplines are as follows.

Machine learning

Machine learning is related to artificial intelligence, and is described as the science of getting computers to act without explicit programming: to learn from 'experience' rather than being told explicitly what to do in any particular situation. A few examples (of many different) machine learning problems include identifying spam email, recognising handwriting or speech, identifying fraudulent credit card transactions, and recommending products to suitable customers based on previous purchases. These are all tasks that one might want a computer to do without any human intervention.

Many machine-learning methods are *data-driven* or statistical in nature. For example, to build a system for speech recognition, first a data set is constructed of recorded speech (converted into digital signals), together with the words that were actually spoken. A statistical model is built to describe the relationship between the received digital signals and the words spoken. This statistical model can then be used predict the most likely word spoken, given a new digital signal only.

Data science

Data science is a rather more loosely defined field, but definitions typically encompass both the analysis of data (with statistical and machine learning methods), and the collection, storage and processing of (often) large and complex data sets, sometimes under the heading of 'big data'. The term 'data science' has become popular in industry, and you should be alert to job opportunities under this heading. A good combination of statistical and computing skills will help your CV.

1.3 R

In this module (and in other Statistics modules) we will be using the software environment R and you will see R code and output throughout these lecture notes. In addition to showing you how to implement methods in this module, the aim is also

- to prepare you for later modules that use R;
- to develop your general (statistical) computing skills.

Bearing this in mind, we will class R code in this module under one of three levels:

1. **Basic**. This refers to R commands that you need to **memorize**: 'basic' R code may appear in your exam. You will need to know what the commands do and how to interpret the output. There won't be much R code under this heading, and you'll often be able to guess what is going on in any case. Here's an example of 'basic' R code:

```
x <- c(10, 12, 20)
sum(x)

## [1] 42

pnorm(1.96)

## [1] 0.9750021
```

2. **Intermediate**. Most of the R commands you'll see will be under this heading. You should try to understand what all such code does, and you may need such commands to do homeworks and online tests, or later on in further modules. You will not be tested on 'intermediate' R commands in the exam, so these don't need to be memorised.
3. **Advanced**. Occasionally, I may demonstrate something using more 'advanced' R commands. You're encouraged to try out such code, but if you're struggling with R, I suggest you skip anything I describe as 'advanced'.

1.3.1 Some hints on learning R

A good way to learn R is to **experiment for yourself**. Two suggestions are as follows.

Make a small change to a command, and see what happens

Suppose you see a command, and you're not sure what all terms inside the command mean:

```
seq(from = 0, to = 10, length = 11)
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10
```

For example, if you're not sure what `length = 11` does, try changing it something else:

```
seq(from = 0, to = 10, length = 2)
```

```
## [1]  0 10
```

```
seq(from = 0, to = 10, length = 3)
```

```
## [1]  0  5 10
```

It's now clearer that `length` controls the number of elements in the sequence.

Breaking up 'chained' commands

R commands are often chained together, and sometimes it helps to try running little sections of a command. For example, in the following,

```
x <- c(2, 3, 3)
sum(unique(x))
```

```
## [1] 5
```

although you might guess what the second command has done, you could try `unique(x)` on its own:

```
unique(x)
```

```
## [1] 2 3
```

and this makes it easier to see that the unique elements of `x` (which are 2 and 3) have been summed.