Chapter 1

Introduction to S-Plus

1.1 Introduction

1.1.1 Data and cases

In this course, we will be fitting more complicated statistical models than those covered in 475.201/8. However, we still start by collecting data on the individual objects of our study, and measuring the characteristics of interest (eg height, weight, length, number of defects etc) for each object.

You will recall from last year that we usually call the objects *cases*. However, cases need not be human - they may be any type of object, animal, vegetable or mineral on which observations are made. The total number of cases in an investigation is called the *sample size*.

The characteristics we measure are called *variables*. The usual situation (which we confine ourselves to in these notes) is where the same variables are measured on each case. In this case the data we collect can be organised into a rectangular table or *data matrix*, with rows corresponding to cases and columns to variables.

The data in Table **??** illustrate this idea. Three measurements (BPD and AC measured in mm, and birth weight (BW) measured in grams) are made on each of 10 unborn infants using ultrasound techniques. Thus there are ten "cases" (infants), and each row of the data matrix consists of measurements on the same case.

As another example, a small part of the data from a dental survey is shown in Table **??**. The variables measured are Age, the patients' age in years, Sex, the patients' sex (M/F) and Attitude, their attitude to dentists, as measured by the question "Do you think dentists deserve the salaries they are paid?"

Types of data

One obvious difference between these two examples is that while all the variables in the first data set are numeric, the second data set contains some non-numeric variables. Numerical variables can either be *continuous*, consisting of readings made along some continuous scale such as weight, length or time, or *discrete*, consisting of counts. The variables BPD, AC, BW are all measurements of length or weight and so are continuous.

In the dental example, the variable Age is continuous or possibly discrete, but the other two are *dichotomies* i.e. variables having just two categories, in this instance atti-

Table 1.1: Ultrasound measurements on ten infants

Case	BPD	AC	Birth weight
1	77	248	1350
2	82	253	1500
3	90	265	2170
4	82	265	1780
5	86	260	1820
6	84	252	1740
7	80	230	1300
8	87	250	1730
9	79	242	1580
10	81	263	1620

Table 1.2: Data from the dental survey.

Case	Age	Sex	Attitude
1	37	M	Y
2	23	M	Y
3	45	F	Y
4	61	M	N
5	33	F	Y
6	29	F	N
7	57	M	N
8	41	F	Y

tude (Yes/No) and sex (Male/Female). Using a variable such as Attitude involves careful definition of the categories, and some rule for the classification of doubtful cases, since each case must be assigned unambiguously to exactly one category. Variables such as Attitude and Sex that are classifications are called *categorical* variables or *factors*. Sometimes the categories are ordered, such as socioeconomic status, or age groups, in which case the factor is called *ordinal*. If there is no implied ordering in the categories (e.g. categories such as sex, race, or religious preference) then the factor is called *nominal*.

The sort of analysis appropriate for a given set of data (e.g. the selection of the appropriate form of model) depends on the number and type of the variables involved.

Response and explanatory variables

Usually, there will be a particular variable that is to be studied, such as attitude or birth weight, and we want to investigate the effect that certain other variables (such as sex or socio-economic status in the case of attitudes) have on this particular characteristic. This particular characteristic is called the *response variable*, (often abbreviated to the *response*) or sometimes the *dependent variable*. The variables whose relationship to the response is to be studied are called the *explanatory variables*, or sometimes the *independent variables*, the *predictor variables*, or the *carriers*.

Models

To study the relationship between the response and the explanatory variables, we fit *statistical models* which describe how the distribution of the response is affected by the explanatory variables. The type of model depends on the type of variables involved. If both response and explanatory variables are continuous, we use a *regression model*. These are discussed in Chapter Three.

Chapter Four discusses how to incorporate factors into regression models. Note that there is no full treatment of analysis of variance in this course - this is done in 475.340.

Finally, a final chapter, to be supplied later, will look at models that are suitable for categorical responses, and deals with logistic regression (where the response is a binary variable) and log-linear models for contingency tables. A brief discussion of generalised linear models will also be included.

1.1.2 Use of computers

Usually the computer software necessary to fit statistical models is collected together, together with facilities for the entry and manipulation of data, into a set of programs called a *statistical package*. Such a package should come complete with a consistent *interface*, a set of conventions that allows the user to communicate his or her intentions to the computer, either by typing in instructions at a keyboard, or using a pointing device such as a “mouse”, or perhaps a combination of both.

In this course, we will perform our analyses using the statistical package S-Plus, a widely available commercial product which is very similar to the language R used in 475.201/8. We will show you how to use S-Plus to fit a variety of statistical models to data. The next section of these notes contains some introductory material on using S-Plus which you should read over before your first computer session.

The package S-Plus is based on the statistical language S, which has been created by researches at Bell Laboratories in the USA. The program R used in 475.201/8 is a locally written implementation of this language. The primary resource for information about S-Plus is the set of manuals available in the Advanced Laboratory, together with the textbooks listed below. In addition, there is a good "on-line" help facility in S-Plus. See Section A.6 in Appendix A for more details on this.

There will be quite a lot of class discussion devoted to the use of the package. There are several books available on S-Plus, S and R. The book "An Introduction to S and S-Plus" by Phil Spector (Duxberry Press, 1994) is a good introduction. See also "Modern Applied Statistics With S-Plus" by Venables and Ripley.The books "The New S Language" by Chambers et al (Wadsworth, 1988) and "Statistical Modelling in S" by Chambers and Hastie (Wadsworth, 1992) are more advanced references to the basic language S. The use of R is covered in the 475.201/8 text, "Data Analysis".

1.2 A brief introduction to S-Plus

In this section we describe the basic facts you need to know in order to use S-Plus on the X-terminals in the Advanced Laboratory. We also give some hints on report writing which you may find useful when preparing your assignments. More technical details on S-Plus and the Unix operating system may be found in the appendices, which you should skim before your first computer session.

To analyse data using S-Plus, we need to follow the flowchart in Figure **??**: i.e.

- Log on to the computer;
- Get the data into the computer;
- Invoke S-Plus to perform the analysis interactively;
- Print out the results;
- Log out.

We then need to use the computers to create a typed report describing the analysis we have done. We now describe these steps in detail.

1.2.1 Logging on

The first step is to "log on" to the X-terminals. When you sit down, you will see a screen like the one in Figure **??**. To log on, you type your *login name* (which will be assigned to you at the beginning of term) and press the return key. You then type in your password (this will also be assigned to you) which will not appear on the screen for security reasons, and press the return key again. Note that the computer distinguishes between upper and lower case, so if your login name is `alee001`, typing `ALEE001` will not work. After you have successfully logged in, you will see an *xterm window* on the screen. The xterm window is where you can type commands to the UNIX operating system, which handles the details of printing, saving and retrieving files, logging on and off and so on.

The typed commands appear after the Unix system prompt `advlab>`. You type your UNIX commands after the prompt and then press the return key to execute them. For more on UNIX commands see Appendix B.

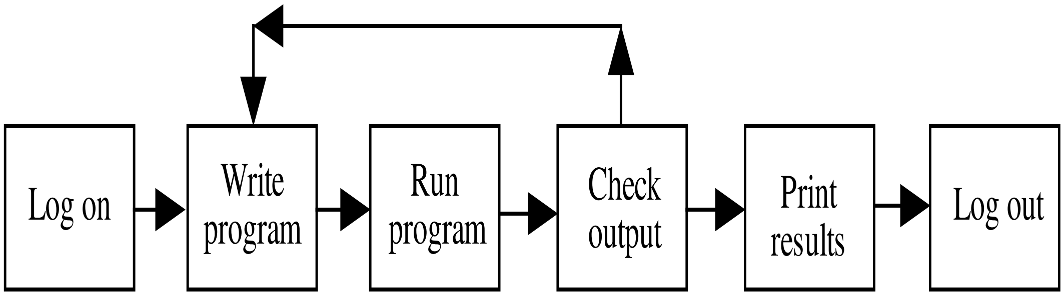

Figure 1.1: Analysing data using S-Plus.

In the examples that follow, the prompt is shown, then the command. Remember not to type the prompt, just what follows it!

There is also a "pop-up menu" which you will see if you move the cursor with the mouse to the grey part of the screen and press the right hand button.

1.2.2 Entering data

We can enter data directly into S-Plus, more or less as you did in with R. See Section A.3 in Appendix A for details on this. Alternatively, before we start up S-Plus, we can prepare an external file of data using a text editor. We then start S-Plus and then read in our data from this external file.

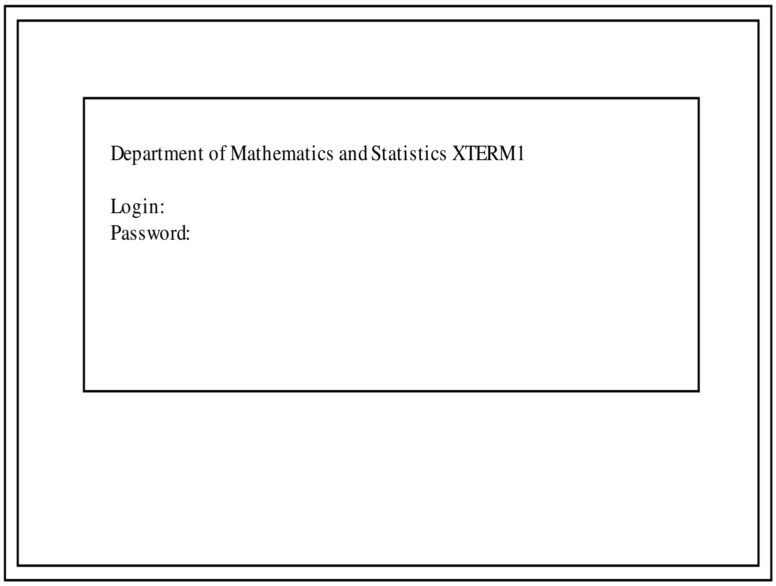

Figure 1.2: The login screen.

Using the text editor

Now we describe how to use the text editor `xedit` to prepare a file of data. Start `xedit` by typing

```
advlab> xedit &
```

in an xterm window window, or by selecting `xedit` from the pop-up menu. This produces the `xedit` window shown in Figure ??.

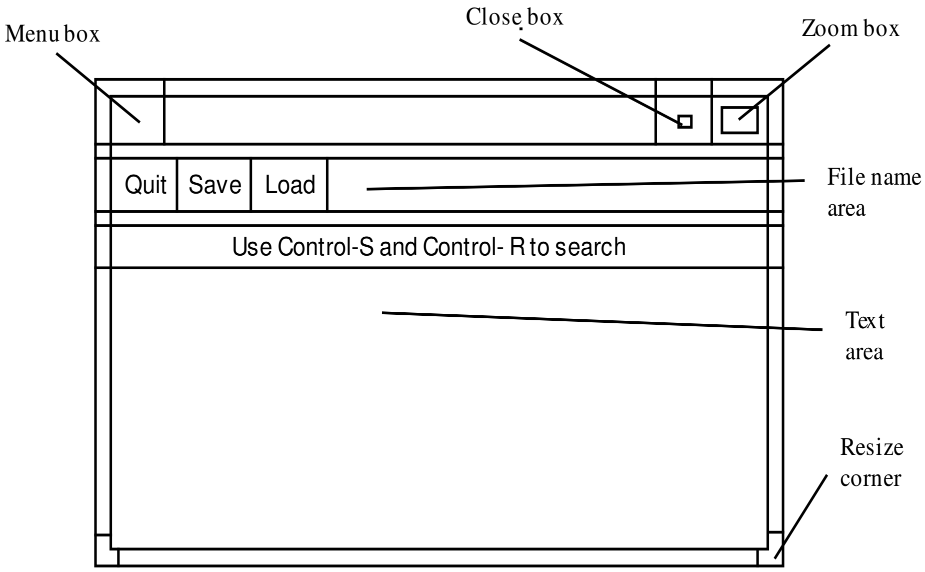

Figure 1.3: The `xedit` window.

Move the cursor with the mouse to the "filename area" and type in a name for your data file. The name should be of the form `xxxx.dat` where `xxxx` represents a string of letters or digits. Choose names that are meaningful, for example `ultrasound.dat`. Type this name in the file area. To actually enter the data, move the cursor to the "program area" and press the left hand button. You can then type in the data lines. End each line with a carriage return. Note how the insertion point ("$\wedge$", the point where the next character will go) moves along as you type. As an exercise, type in the ultrasound data given in Table 1.1. If you make mistakes, or want to make changes, the following should help:

- To *erase* a character just before the insertion point, press backspace.
- To *insert* text, use the mouse to move the cursor to the desired place and click the left hand button. Type in the new text.
- To *delete* text, select the text to be deleted by holding down the left hand button and dragging over the text. The text will "highlight". Release the left button and type Ctrl-w (hold down the Control key and at the same time press "w").
- To *move* text, delete it as above, move the cursor to the place where the deleted text is to be inserted, click the left button, click the middle button.

- To *copy* text, select it as above, move the cursor to the place where the copy is to be inserted, click the left button, click the right button.
- To *search* for text or *replace* text, type Ctrl-S or Ctrl-R. Fill in the boxes as directed.

When you have typed in your data, "save" the file (i.e. save a copy of the program on the Sun's disk) by clicking on the "save" box with the left button. A copy of `ultrasound.dat` is now permanently stored on the disk, and will remain there until you erase it.

Close the `xedit` window by clicking on the close box of the window. The window collapses into an icon. You can open the window again by clicking on the icon with the middle button. Closing the window simply reduces the space it occupies on the screen. To dispose of the window completely, click the box marked Quit in Figure ??.

Having exited from the editor, we can look at the file we have just made by using the UNIX command `more`. This prints out the file on the computer screen:

```
advlab> more ultrasound.dat
BPD   AC    BW
77    248   1350
82    253   1500
90    265   2170
82    265   1780
86    260   1820
84    252   1740
80    230   1300
87    250   1730
79    242   1580
81    263   1620
```

1.2.3 Using S-Plus

To use S-Plus, first use the pop-up menu to make a new xterm window, as described in Section B.4.2 of Appendix B. Then position the cursor in the new window and type

```
advlab> Splus
```

The computer responds with

```
S-PLUS : Copyright (c) 1988, 1995 MathSoft, Inc.
S : Copyright AT&T.
Version 3.3 Release 1 for Sun SPARC, SunOS 5.3 : 1995
Working data will be in .Data
>
```

The prompt > appears whenever S-Plus expects you to type something. It is followed by an insertion point ($\wedge$). Just as in R last year, the insertion point tells you where characters typed on the keyboard will appear on the screen. In the examples that follow, text after the prompt > represents text that you type. Lines not beginning with the prompt represent responses by S-Plus . Remember not to type the prompt! Note also that S-Plus has opened a "graphics window" and a "help window". You may need to resize them by putting the cursor in the resize box and dragging the mouse.

Suppose we want to read in the ultrasound data, plot BW versus BPD, and then fit a straight line to the data. The first step is to get our data into the computer. There are many ways to do this, but we will assume that we are reading our data from an external file. (Other ways of entering data are discussed below in the section Getting data into S-Plus in Appendix A.)

Remember that prior to starting up S-Plus, we used the data editor `xedit` to create a text file `ultrasound.dat` containing the data. The first line of the file contains names for the variables. To read the data into S-Plus, and create what is known as a *data frame*, which is the S-Plus equivalent of a data matrix, type

```
> ultra.df<- read.table("ultrasound.dat", header=T)
```

To see what the object `ultra.df` looks like, just type its name:

```
> ultra.df
   BPD  AC   BW
 1  77 248 1350
 2  82 253 1500
 3  90 265 2170
 4  82 265 1780
 5  86 260 1820
 6  84 252 1740
 7  80 230 1300
 8  87 250 1730
 9  79 242 1580
10  81 263 1620
```

The function `read.table` works rather like the function `read.file` used in R. The main difference is that the object produced is a "data frame " rather than a series of vectors. (Data frames are discussed in detail in Section A.2.4 of Appendix A.) The first argument to `read.table` is the name of the external file and the second tells S-Plus to look for the variable names in the first line of the file.

One problem with data frames is that S-Plus doesn't know about the variables in them, so we can't use the variables (in this case BPD, AC and BW) in S-Plus expressions. We get round this by using the `attach` function: typing

```
> attach(ultra.df)
```

makes the variables accessible, so we can use them in S-Plus expressions.

Having got our data into S-Plus, we can now draw our plot, using the function `plot` as we did in R and draw the plot in the usual way:

```
> plot(BPD,BW)
```

Suppose we want to fit a straight line through the plot. Models in S-Plus are written in a special way. We want to fit the model

$$BW = \alpha + \beta BPD + e$$

which in S-Plus notation is written

$$\texttt{BW} \sim \texttt{BPD}$$

The constant term and the errors do not need to be specified. To fit the model, we use the function `lm` (`lm` = "linear model" which is another name for regression.) We type

```
> ultra.reg<-lm(BW~BPD)
```

This stores the result of the regression in the "regression object" `ultra.reg`. We can see the results using the function `summary`:

```
> summary(ultra.reg)

Call: lm(formula = BW ~ BPD)
Residuals:
  Min  1Q Median    3Q Max
 -205 -90   12.5 101.3 165

Coefficients:
                 Value Std. Error    t value   Pr(>|t|)
(Intercept) -2895.0000   929.1386    -3.1158     0.0143
        BPD    55.0000    11.2099     4.9064     0.0012

Residual standard error: 133.4 on 8 degrees of freedom
Multiple R-Squared: 0.7506
F-statistic: 24.07 on 1 and 8 degrees of freedom, the p-value is 0.001184

Correlation of Coefficients:
    (Intercept)
BPD -0.999
```

Other regression quantities can be displayed:

```
> coefficients(ultra.reg)
 (Intercept) BPD
       -2895  55
> residuals(ultra.reg)
  1    2   3   4   5  6    7    8   9 10
 10 -115 115 165 -15 15 -205 -160 130 60
```

We can add the fitted line to our plot by typing

```
> abline(-2895,55)
```

or

```
> abline(coefficients(ultra.reg))
```

(i.e. by giving the intercept and slope of the line) and give the plot a title by typing

```
> title(``Figure 1: Birth weight versus Biparietal diameter'')
```

The resulting graph is shown in Figure **??**.

To get a paper copy of the graph, click on the box marked print graph in the graphics window, using the left-hand mouse button. The graph will then be queued for printing. Swipe your ID card and follow the instructions on the console by the printer. Your graph should then be printed. Put the cursor in the xterm window where S-Plus is running and press return to get the S-Plus prompt back.

To exit from the program S-Plus use the function `q`. Notice that the brackets `()` are necessary, even though there are no arguments.

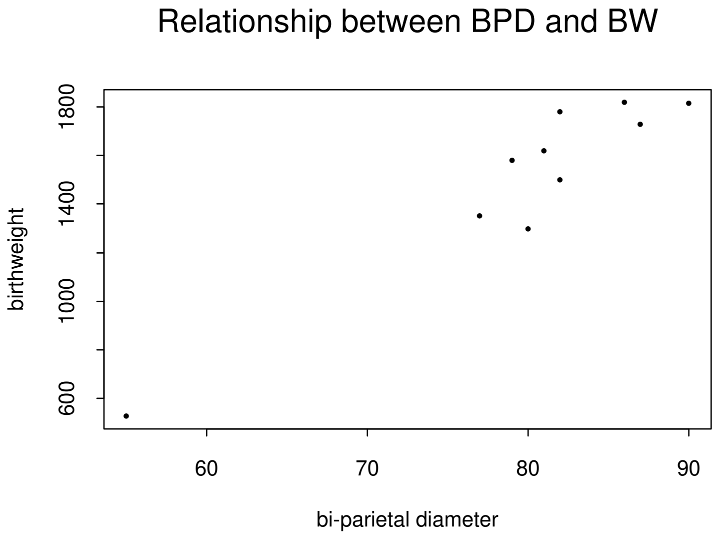

Figure 1.4: A typical S-plus plot.

```
> q()
```

The UNIX prompt reappears in the xterm window, and you can type further UNIX commands.

1.2.4 Printing files

We saw that printing what was displayed in the S-Plus graphics window was easy - just clicking the box. Printing files in UNIX is a little bit harder.

Suppose that you have made a disk file `assignment1.notes` containing S-Plus output and notes for the answer to your assignment. You can print it out on the laser printer in the Advanced Lab by using the `mpage` command. For example, to print out the file `assignment1.notes`, type

```
advlab> mpage -1 assignment1.notes
```

in an xterm window. Go to the printer and run your ID card through the "swipe card" reader. The file `assignment1.notes` will then be printed out on the laser printer.

NOTE: printing on the laser printer is quite expensive. Confine printing to material that you want to include in your assignment answers. You will get an allowance of 100 pages free, but any more than this will have to be paid for at the Student Resource Centre. You can save paper by replacing -1 in the `mpage` command above by -2. This prints out at half size. (It is possible to print out at 1/4 or 1/8 size but your elderly lecturers wont be able to read it.)

1.2.5 Logging out

At the end of your session, you must "sign off" by logging out. Move the mouse onto the grey area outside all the open windows, click on the right hand mouse button, and, with the mouse button still depressed, move the cursor over the "Log Out" command, then release the button.

Failure to do this may result in another user gaining access to your files and using some of your quota of print pages. So, DON'T FORGET TO LOG OUT!!

1.2.6 Preparing reports

As we discussed in the Preface a key part of this course is the writing of reports for each assignment describing your analyses. To quote again from the preface:

> *The people with an interest in the answers are your customers, and your conclusions are your product. When handing in assignments, imagine that the marker is your client, who is hiring you for your professional expertise. Your consulting report (i.e. your assignment) should be neat, correctly spelled, well-organised, relevant and concise – in other words, professional.*

Reports must not be handwritten. They can be prepared on any word processor to which you have access. The rest of this section describes report preparation using the text editor `xedit` and the program Latex.

Your reports should contain (at least) three sections.

- Section 1: Summary: A brief statement of the problem and your conclusions. You should regard this as your "executive summary" it should contain no symbols or mathematics and take up no more than one page.
- Section 2: Results: A description of your analysis. The description should include summaries of the models that you have fitted. These summaries may include output from S-Plus commands (see below how to do this), S-Plus graphics and tables. Graphics and tables should be labelled "Figure 1", "Table 2", etc and either placed in the body, or attached to the end of the report. In the text of the report you should refer to "Figure 1", or "Table 2".
- Section 3: Conclusions: This section should give the conclusions from your analysis with reasons justifying them. You might like to bear in mind the following quote from Ronald D. Snee, who was for many years a statistical consultant with the giant U.S. chemical company DuPont.

  > *When I am writing a report, the single most important thing I want people to learn is always displayed in the form of a plot – preferably near the beginning. If I feel that I am obligated to include some information, but I don't want people to notice it, I display it as a table.*

We suggest that you use the typesetting package Latex to produce your reports. Using the editor `xedit` create a file whose name has the suffix `.tex`, e.g. `Report1.tex`. You can copy the example file `report.tex` from the directory `/users/stats/lee/330` with the command

```
advlab> cp /users/stats/lee/330/report.tex .
```

Try experimenting with this file, making small changes and observing the effect. You use Latex by inserting special commands in the text of your document (e.g to change fonts or type size, or to centre text. These commands are preceded with a backslash "\". See the introductory document *Essential Latex* for more details. You can read this by typing

```
advlab> xdvi ~lee/330/essential
```

in an xterm window. When you have prepared your report, typeset it using the Unix command

```
advlab> latex report
```

If everything goes well a file called `report.dvi` will be produced. You can view it using the previewer `xdvi`.

```
advlab> xdvi report
```

If it is not to your satisfaction, make changes, reedit the file `report.tex`, run it through Tex, and preview again. When you feel that your report is ready for printing, print it using the command.

```
advlab> dvips report
```

1.2.7 Incorporating output from S-Plus

We have seen how to get hard copies of S-Plus graphs. The easiest way to get hard copies of output from S-Plus commands is to open up `xedit`, then cut and paste the desired text from the S-Plus window into the editor.

To cut and paste: Highlight the text to be pasted by holding down the left-hand button and dragging the mouse over the desired text. Using the mouse, move the cursor to the text input area of the editor. Click the **left hand** button to mark the point where the text is to go. Pressing the **middle button** will copy the selected text into the desired place.

Alternatively you can use the S-Plus function `sink`. When you are in S-Plus, typing

```
> sink("filename")
```

causes all subsequent output from S-Plus to be written to the file `"filename"`. This file can then be edited to form the basis of your report. To redirect S-Plus output to the screen after you have used `sink`, call `sink` with no argument:

```
> sink()
```

1.2.8 Documentation

Full details of all S-Plus commands are given in the manuals in the Advanced Laboratory. You can also use the extensive "on-line help" facility of S-Plus. See Section A.6 of Appendix A for details.