STAT 260 - Introduction to R Assignment 1

1 Introduction to R

A video demonstrating how to download R and some basic introductory commands is available on Brightspace.

- To download R: `http://www.r-project.org/`
- When you first open R, there will be a single window open: the **R Console**. You can enter commands in this window, and the output will appear here as well.
- It may be wise to open a **Script Window**. In the PC version of R, you do this by selecting **File**, then **New Script**. In the Mac version, select **File**, then **New Document**. The script window is useful because you can write many lines of code here and save your work as you go along.
- Commands in the script window do not run automatically. Put your cursor on the command line that you want to run, hit **Ctrl+R** on a PC (**Command+return** on a Mac) to run. You can also highlight several commands and run them the same way.
- To display the help file for any command, simply enter a question mark before the command name. For example, to see help for the command **hist**, enter `?hist`.
- If you are not sure of the name of a particular command, R allows for fuzzy matching. Enter two question marks followed by the search term. For example, enter `??histogram` to see a list of possible commands related to histograms.
- **RStudio** is an integrated development environment which can make working with R projects easier. A free version is available for download on both Mac and PC. RStudio is optional; the RConsole alone is enough if you don't want to use it. To download RStudio on your own computer, first go download R as mentioned in the previous paragraph, then visit `https://www.rstudio.com/` and click the "Download" button at the top of the page.

2 Entering Data

- To start, we will work with numerical data stored as vectors. We will see tables/matrices and non-numerical data in later courses.
- For a small number of observations, it is easy to enter the data directly. Suppose we have a set of five steel bolts, and we measure their weights and lengths:

Bolt	1	2	3	4	5
Weight (g)	12.3	12.5	12.7	11.5	15.3
Length (cm)	1.9	2.2	2.1	1.8	2.5

- It is good practice to use a descriptive name for our vectors, rather than a simple letter. Names must not contain spaces, but may contain periods. For example, we can use **bolt.weight** as a the name for our weight vector.

- To create our bolt weight vector, we use the following command:

  ```
  bolt.weight <- c(12.3, 12.5, 12.7, 11.5, 15.3)
  ```

- We can read the symbol `<-` as meaning “is defined as”. The *c* before the data is the command R uses to create a vector.

- For a large number of observations, we can use the **scan()** command to scan in the data. Suppose we want to create a vector called **bolt.length** that consists of the bolt length measurements. We could enter the following:

  ```
  bolt.length <- scan()
  1.9
  2.2
  2.1
  1.8
  2.5
  ```

  Hitting **Enter** on a blank line stops the process of reading in numbers. At that point, R will tell you how many numbers it has read in. If you need to terminate the process early, hit **ESC**.

- The **scan()** command is especially useful if you are copying and pasting a column of data from a spreadsheet. The command will also accept data copied and pasted from a row spreadsheet; there is no need for commas to separate numbers with this command, only the spaces between the numbers.

- If we wish to display our vectors now in the computer’s memory, we just enter the vector name. When we enter `bolt.weight`, we get the following output:

  ```
  [1] 12.3 12.5 12.7 11.5 15.3
  ```

- The `[1]` at the beginning of the line is a counter. If the data had several lines worth of observations, there would be a counter at the beginning of each line to tell the reader which observation is at the start of that line.

3 Numerical Summaries

- To find the mean of our bolt weight data, we enter the command: `mean(bolt.weight)`
- The commands `var(bolt.weight)` and `sd(bolt.weight)` give us the sample variance and sample standard deviation (respectively) for our weight observations.
- The command `cor(bolt.weight,bolt.length)` returns the sample correlation coefficient for the bivariate bolt data.
- The command `cov(bolt.weight,bolt.length)` returns the sample correlation covariate for the bivariate bolt data.

4 Visual Summaries

- The **hist** command is used to create histograms:
  - To create a histogram of the bolt data, enter the command `hist(bolt.weight)`. We can use various options to make our histogram even better looking. The options **main**, **xlab**, and ylab allow us to enter a customized title for our histogram and *x*-axis.
  - With the command below, the title of the histogram will read “Bolt Study”, and below the *x*-axis it will ready “weight of bolts (in g)”.

    ```
    hist(bolt.weight, main=‘‘Bolt Study’’, xlab=‘‘weight of bolts (in g)’’,
    ylab=‘‘Frequency’’)
    ```
  - Optional: If we want to change R’s default axis scaling, we can manually set the length of *x* and *y*-axis by using the **xlim** and **ylim** options, respectively. For example, if we want our histogram to show an *x*-axis starting at 6.0 and going to 20.5, we would add `xlim=c(6.0,20.5)` into our histogram command.
  - Optional: We can add colour to the bars in our histogram by using the **col** option. The easiest way to specify a colour is to use one of R’s 657 default colour names. You can view a list of these colours in R by typing `colors()`, or you can view an online guide: `http://www.stat.columbia.edu/~tzheng/files/Rcolor.pdf`

    For example, if we wanted our bars to be coloured “seagreen”, we would add the option `col = ‘‘seagreen’’` to our histogram command.
- The **boxplot** command is used to create boxplots:
  - Enter `boxplot(bolt.weight)` to create a boxplot of the bolt data.
  - If we wish to compare two (or more) sets of data with boxplots, we can create side-by-side boxplots. For example, along with our previous bolt data, we might also have a set of data for washers:

Washer	1	2	3	4	5	6	7
Weight (g)	2.4	3.1	3.2	1.9	2.3	2.1	2.6

if we enter `boxplot(bolt.weight, washer.weight)`, R will put boxplots for both data sets on the same axes.

▷ To label the boxplots on the x-axis as bolts and washers, enter the command:

```
boxplot(bolt.weight, washer.weight, names=c(‘‘bolts’’,‘‘washers’’))
```

▷ We can also add a title to our plot, and labels to the x-axis and y-axis using the **main**, **xlab**, and **ylab** options, as we did in histograms.

▷ If we wish to know what values R is using for the boxplots, we can use the summary command. When we enter the command `summary(bolt.weight)` we get the following output:

```
Min.   1st Qu.  Median  Mean   3rd Qu.  Max.
12.10  12.30    12.50   12.44  12.60    12.70
```

▷ The bottom of the box is at the first quartile (12.30) and the top of the box is at the third quartile (12.60). The line in the middle of the box is at the median (12.50). The smallest observation is 12.10, and the largest observation is 12.70.

▷ Optional: We can add colours to the boxes with the **col** option, similarly to with histograms above. We can make each box in our plot in a different colour by entering our colours as a vector.

For example, if we want the left box to be “blueviolet” and the right box to be “darkcyan”, we would use the option `col = c(‘‘blueviolet’’, ‘‘darkcyan’’)` in the boxplot command.

- The **plot** command is used to create scatterplots:

▷ Enter `plot(bolt.weight,bolt.length)` to create a scatterplot of the bolt data; bolt.weight is plotted on the x-axis, while bolt.length is plotted on the y-axis. The options **main**, **xlab**, and **ylab** allow you to specify the main title, x-axis label, and y-axis label:

```
plot(bolt.weight,bolt.length, main = ‘‘Bolt lengths versus weights",
xlab = ‘‘Bolt weights (g)", ylab = ‘‘Bolt length (cm)")
```

- For the Windows version of R, if you right-click on your graphic that you have created, you can select Copy as metafile. Then, in a Word or Open Office document, you can paste it in using Ctrl+V. On a Mac, select your graph, enter Command+C and then paste it into a Word document using Command+V.

- **WARNING:** There is a known bug that sometimes occurs when converting boxplot images into PDFs, where strange diagonal lines appear over the image. If this happens to you, screenshot your boxplot in R, and then paste the (neatly cropped) screenshot into your document instead of the metafile.