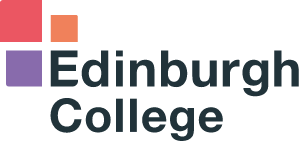
Data Science: ePortfolio 

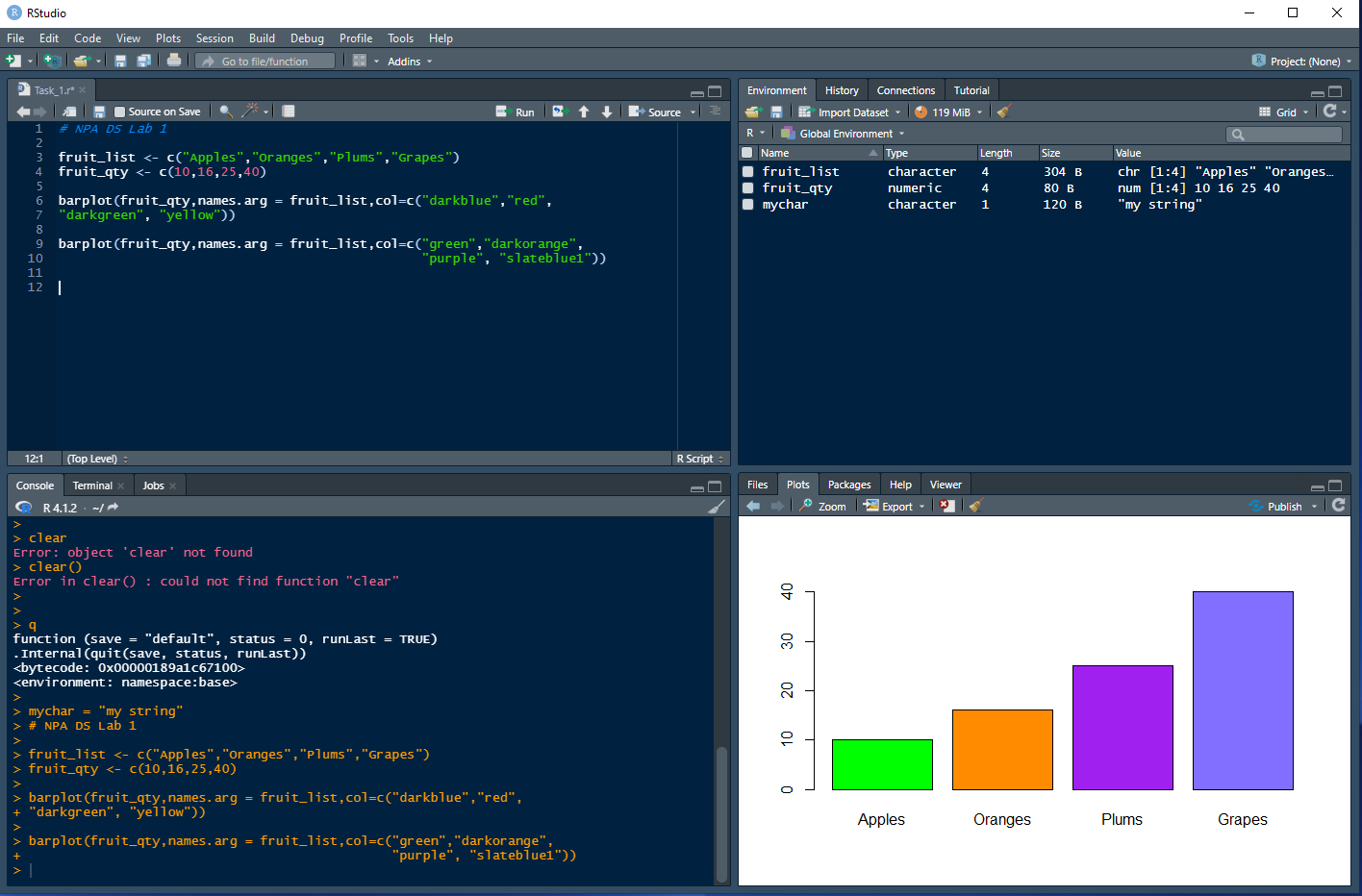
|  |  |
| --- | --- |
| First Name (s) | Kevin |
| Last Name (s) | Gemson |
| EC Email Address | EC2111447@edinburghcollege.ac.uk |

# Guidance - How to complete this document

* Check the course Moodle page to see what evidence is required.
* All screenshots should be cropped and labelled to show the key information.
* The lab title and date of completion should be added above each section.
* Text should be in 11pt or 12pt, any code should be in a monospaced font.
* Each week complete the summary with an overview of the tasks completed.

# Lab 1: Data In Society - Date Completed: 05/02/2022

1. A screenshot of RStudio Running on your computer



|  |
| --- |
| 1. Your plot from step 1.11 of the lab 2. Your plot from step 1.12 of the lab |

## Weekly summary, complete the following:

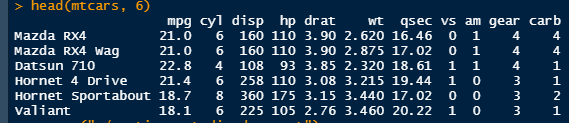
This week I learned about ... principles of Data Citizenship, and how data is collected and used (and can be misused) in Society. Also, learned the definition of Data Science and its relationship to machine learning and artificial intelligence.

In the lab I … learned how to install R and R Studio, and run a number of tasks to understand how the basic syntax works. Also followed a number of YouTube tutorials including the extra learning module, to try to get a better understanding of what R can do.

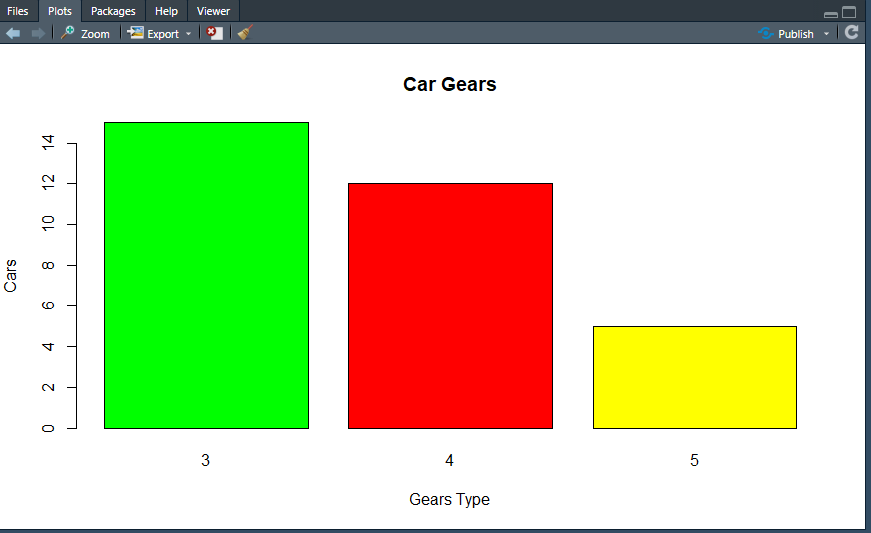
I enjoyed.. getting to see the possibilities of what R can do, being able to produce graphs with some fairly simple commands

I found it challenging when… working through some of the different components of the syntax (vectors matrices, datasets, etc), especially as the syntax is different to other languages I am more familiar with.

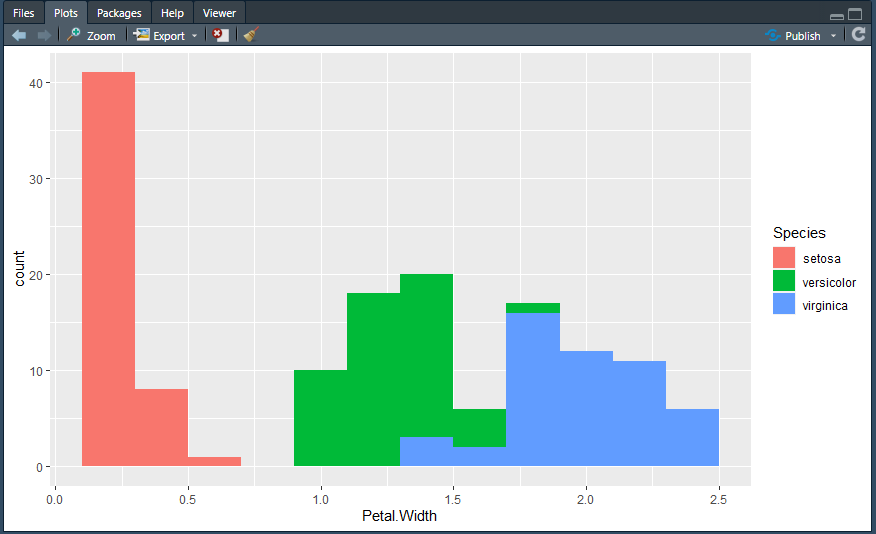
# Lab 2: Working With Data - Date Completed: 10/02/2022

1. A screenshot of running 1.4, how many mpg's was the Mazda RX-4? **- 21.0 mpg**
2. Add an image of, and the associated R code to create the bar plot from step 1.9.

barplot(mtcarsgear, main = "Car Gears", ylab = 'Cars', xlab = "Gears Type", col=c('green','red','yellow'))

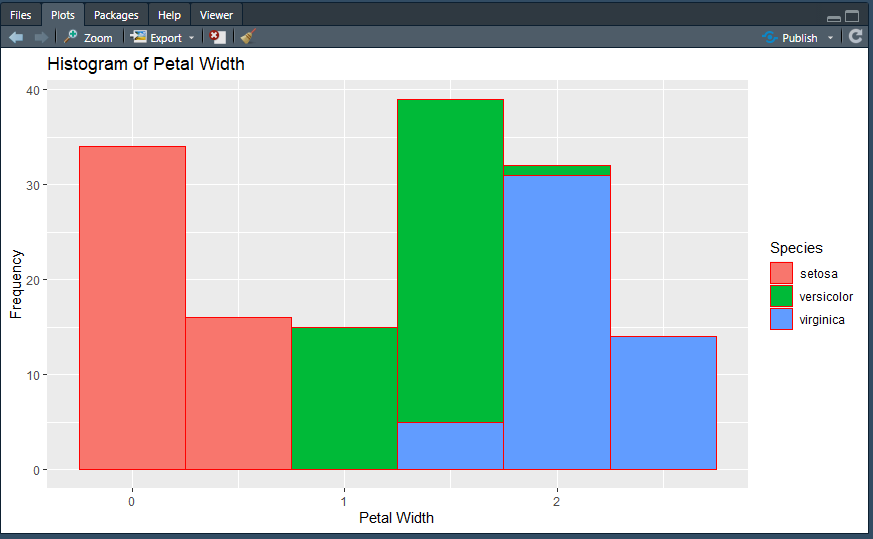


1. Add an image of the histogram from step 2.8.



1. Add an image of the chart from step 2.11, and add the associated code to create this. Add an explanation of what each part of the code does.

histogram + geom\_histogram(binwidth=0.5,color="red",aes(fill=Species)) + xlab("Petal Width") + ylab("Frequency") + ggtitle("Histogram of Petal Width")



Explanation of code -

histogram – variable defined earlier in code which is assigned a ‘ggplot’ graph based on iris’s PetalWidth values

geom\_histogram(binwidth=0.5,color="red",aes(fill=Species)) - displays the counts with bars. Width of bins is 0.5, outline of each bin is to be red. ‘aes’ = aesthetics; using the ‘Species’ data from ‘iris’ as the fill value groups the data together, based on the values in that field and overlays each set of values.

+ xlab("Petal Width") - sets X-axis label

+ ylab("Frequency") - sets Y-axis label

+ ggtitle("Histogram of Petal Width") - sets title

## Weekly summary, complete the following:

This week I learned about ... the applications of Data Science and the Data Science lifecycle, and how this compares with regular software development projects. I also learned about the four Vs of Data, and about different types of bias, and how these can affect outcomes in a Data Science project.

In the lab I … found out how to import datasets and convert these into bar charts with some basic design elements. Then, I was able to import an additional library and use it to convert the data into a histogram with some additional formatting

I enjoyed ... seeing the types of graphs and charts that can be produced with some fairly simple commands, and seeing how these can be enhanced with the addition of extra parameters.

I found it challenging when … trying to work out some of the syntax (for example, the ‘sample’ statement from the ‘iris’ dataset)

# Lab 3: Management And Security - Date Completed: 24/02/2022

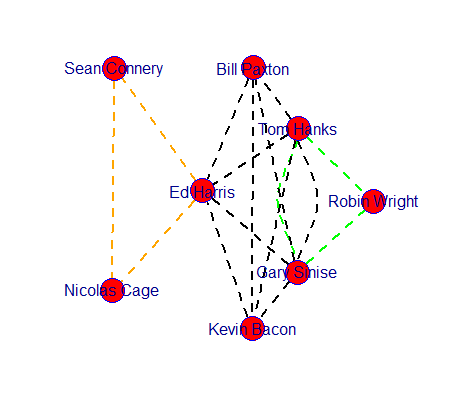
1. A screenshot of running 1.4, the head command running in part 1.4



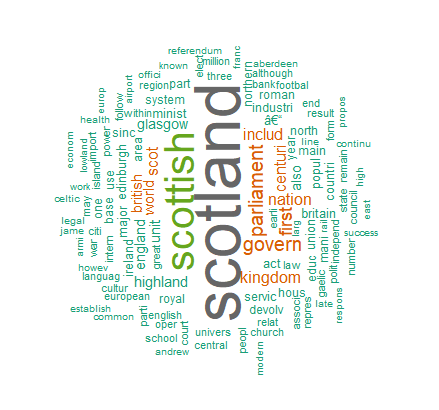
1. The finalised plot from part 1.8, with the code used to create it

plot(actorNetwork, vertex.color="Red", vertex.frame.color = "Blue",

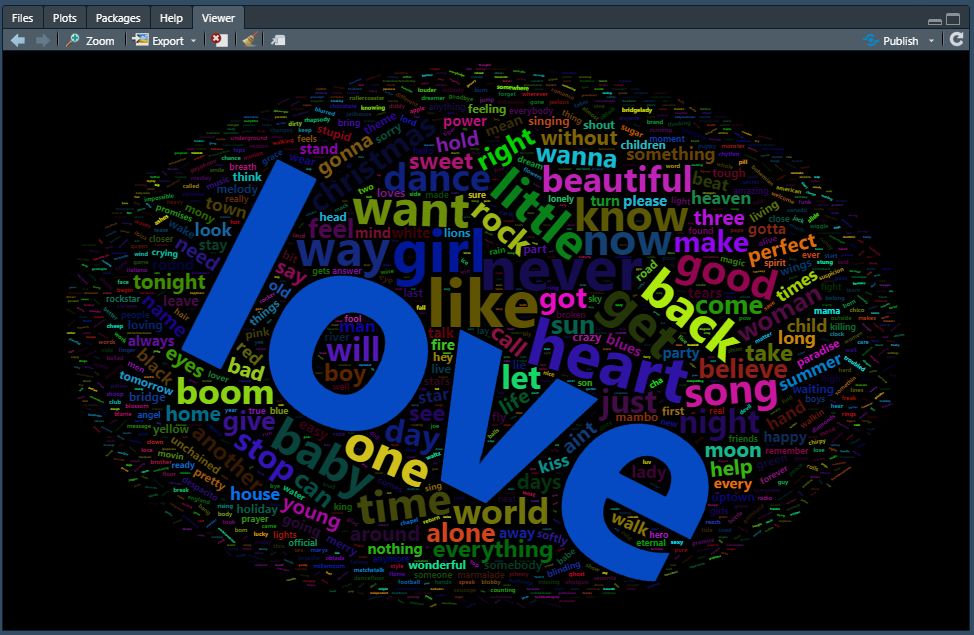
vertex.size = 19, vertex.label.family = "Arial", edge.width=2, edge.lty=2)

1. One of the word clouds created in step 2.5

Example 3 - Improved cloud with colour palette:



1. Find a news or magazine article, convert the article to a text file and use the skills from this lab to create a wordcloud of this.



1. Add an explanation of why you chose the text that you have chosen, what are the most common words, was this expected? - the text I chose is a page listing all the UK number 1 singles since the music charts were first compiled in 1952. I thought it would be interesting to see which words occur the most in the titles of these singles. The results show that, as expected, the word ‘love’ is by far and away the most commonly used word (97 times), but following that, there is a huge gap to the next most common word (‘like’, used 23 times), then there is very small differences in the number of word following that (‘heart’ used 22 times, ‘never’ used 21 times, ‘girl’ used 20 times). I had to try changing some of the settings, such as with stemming and removing stopwords to get a decent result, but this may be skewing the results (for example, ‘the’ and ‘I’ are very common words that are removed from the final output). I would maybe have to try changing the settings to get satisfactory results, but think the wordcloud above shows some useful information.

## Weekly summary, complete the following:

This week I learned about ... a number of aspects of Data Management and Security, including methods of data management/data governance and data security and understanding data privacy. Also, the rights and responsibilities of data subjects and data owner and understanding what legislation is in place to protect data.

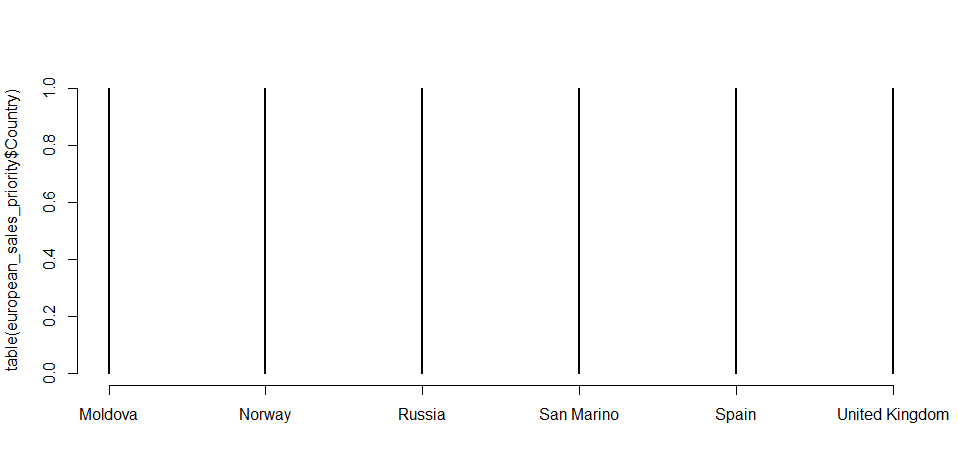
In the lab I … tried to use various R packages and libraries to produce visualizations of textual information, i.e. igraph functions and wordcloud packages

I enjoyed... seeing what can be done with the various packages available to R – the wordcloud, for example, seemed to produce some impressive output with a small number of commands

I found it challenging when … it took some time to try to understand what was happening in the igraph section, as some of the parameters being passed in were not obvious. Also, I had a number of technical issues with installing some of the libraries and ended up having to reinstall some of the software. There were a few other issues such as getting RStudio to find font packages, which required a bit of effort to try to understand and fix. This is probably all good practice in identifying faults, reading error messages and trying to find fixes, however.

# Lab 4: Title – Open Data - Date Completed: 07/03/2022

1. The plot created in step 1.13...



… and the code used to create this -

attach(X100\_sales\_records)

european\_sales\_priority <- X100\_sales\_records[Region == "Europe" & `Order Priority` == "L" , ]

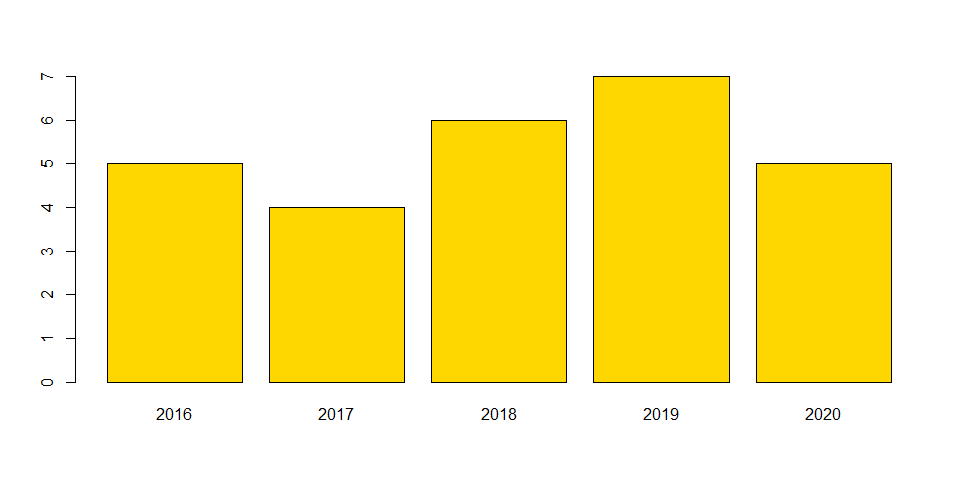
plot(table(european\_sales\_priority$Country))

1. The command used to create the table from step 1.14

barplot(table(X100\_sales\_records$Country,X100\_sales\_records$`Item Type`))

1. The plot from 3.6 and the code used to create it

barplot(my\_group\_data\_new,col="gold1",names.arg=colnames(df\_group[-1]))



1. An explanation of the $, then the [ ] and  [[ ]] and what we can use these for in R

R has a number of datatypes which contain multiple data elements, including vectors, lists, data frames and matrices. R provides several constructs which allow access to individual elements or subsets of these datatypes through indexing operations. These are the [, [[ and $ operators. For example -

x[i]

x[i, j]

x[[i]]

x[[i, j]]

x$a

x$"a"

These are all used to access elements of a data structure, but with some differences. Generally, the single [ will return a subset of an object of the same type. So using [] on a vector will return a vector, and using [] on a data frame will return a data frame. The single square bracket should be used for singular indexing, i.e. returning a single object from a vector or matrix, where all elements are of the same type. It can also be used for indexing by vectors, and will return a list of all the indexed elements.

The double square brackets are used to index one element from a recursive data structure (i.e. where elements may not be of the same type) such as a list or data frame. It will return the individual element, such as a character or an integer, rather than a list or other data structure. The double square bracket cannot be used with a vector index.

$ is a special case of [[ in which you access a single item by actual name. It is used for recursive objects such as lists or data frames. It allows only a literal character string or a symbol as the index, such as a column header.

1. An explanation of the indexing of a vector in R, how are vectors indexed?

Vectors are data structures containing elements of the same type, such as integers or characters. Vectors are indexed using single square brackets. The index value within the square brackets can be a single integer or a vector, such as a range of elements to be selected. It could also be a negative integer, and this will return all elements except those indicated by the negative value.

## Weekly summary, complete the following:

This week I learned about ... different data types such as qualitative and quantitative, structured and unstructured etc. Also learned about types and sources of open data and about data ethics, i.e. the rights and wrongs of holding and using data

In the lab I … imported data from outside sources, converted this into R objects and used these to access individual elements and produce visual representations of the data.

I enjoyed ... finding out more about how to import data from files and access individual elements of it to produce R data structures

I found it challenging when … trying to understand some of the syntax such as indexing data frames and accessing individual column elements, particularly when using more than one condition in the command.