Tutorial 1 - Saying hello to R (Solutions).

Christopher A Oldnall

Welcome to the Regression and Simulation methods module. This is the first script for the first tutorial. This is a prime opportunity to learn a lot more about data science and how to programme in R so please make full use of this.

Throughout we will use a package called 'tidyverse'. This is a 'superpackage' which contains lots of other packages with lots of functions. If you haven't installed this already then you need to go ahead and run the line below. You only need to install a package once.

```
# install.packages("tidyverse")
```

You will notice in your console that the section above appeared in a 'grey' area. This is because this is an R Markdown document. This is represented by the file extension .Rmd. Meanwhile there are also R Scripts (represented by .R extensions). Typically when writing code for publishable purposes or for software we have a series of R Scripts, however R markdown files are becomming more flavoursome when examining unqiue datasets since a neater overview can be given, it also allows for a neat PDF (or other document type) to be an output showing text, code and code output.

Having installed tidyverse earlier, we must still load in the package for our system to use it. We do that with the 'library' command - this is a built in R function. Let's do that now below, noting we do not need to use the quotes to load it in:

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

You may have additionally noticed that to create a chunk of code in an R Markdown document we must use two sets of 3 's with a set of curly braces following the first set of 3 with 'r' written in it. This is telling the interpreter that this is a block of R code.

Exercise 1

Now that you know the basics of how to install and load packages, have a go at writing a code block below to install and load the package called "MASS". This will be the data set we will work with for the rest of this tutorial.

Your Answer:

```
# install.packages("MASS")
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

"MASS" is a package that has been curated for learning R. It contains several data sources. Typically we will not always install and load in data packages like that, but in future weeks we will touch upon other types of R data. For this tutorial we will be working with the cats data set. Run the below to select this from "MASS."

```
data("cats")
```

Of course, as any great statistician or data scientist will know, the first thing to do when loading in is to try and get some summary of the data. Run the below and answer the exercise.

```
str(cats)
```

```
## 'data.frame':    144 obs. of  3 variables:
##  $ Sex: Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Bwt: num  2 2 2 2.1 2.1 2.1 2.1 2.1 2.1 2.1 ...
##  $ Hwt: num  7 7.4 9.5 7.2 7.3 7.6 8.1 8.2 8.3 8.5 ...
```

```
head(cats)
```

```
##   Sex Bwt Hwt
## 1   F 2.0 7.0
## 2   F 2.0 7.4
## 3   F 2.0 9.5
## 4   F 2.1 7.2
## 5   F 2.1 7.3
## 6   F 2.1 7.6
```

```
tail(cats)
```

```
##     Sex Bwt  Hwt
## 139   M 3.6 15.0
## 140   M 3.7 11.0
## 141   M 3.8 14.8
## 142   M 3.8 16.8
## 143   M 3.9 14.4
## 144   M 3.9 20.5
```

```
summary(cats)
```

```
##  Sex         Bwt             Hwt
##  F:47   Min.   :2.000   Min.   : 6.30
##  M:97   1st Qu.:2.300   1st Qu.: 8.95
##         Median :2.700   Median :10.10
##         Mean   :2.724   Mean   :10.63
##         3rd Qu.:3.025   3rd Qu.:12.12
##         Max.   :3.900   Max.   :20.50
```

Exercise 2

Provide an overview of the data set cats, in particular discuss how many observartions and variables there are, as well as giving their data types.

Your Answer: There are 144 observations of 3 variables in the data set cats. The first is Sex which is saved as a factor (either Male or Female, denoted by M/F respectively). The other two variables are body weight and heart weight. Both are saved as floating point numbers.

Now we know how to get some built in summary features, we may want to level up and become more statistical. That is what about looking at the mean and standard deviation of our data? This is incredibly important. Run the below and see what comes out.

```
HeartSummary <- cats %>%
    summarise(cats_heart_mean = mean(Hwt), cats_heart_sd = sd(Hwt))

HeartSummary
```

```
##   cats_heart_mean cats_heart_sd
## 1        10.63056      2.434636
```

Did you notice some weird syntax above? Let's address that. In R we use '<-' to assign a value to a variable. Above we have also started to use the tidyverse. '%>%' is known as a pipe and allows us to more intuatively access the data we have by 'piping into it' and then applying some functions.

Exercise 3

Analogously to above, provide a summary for the body weight variable in the cats data set.

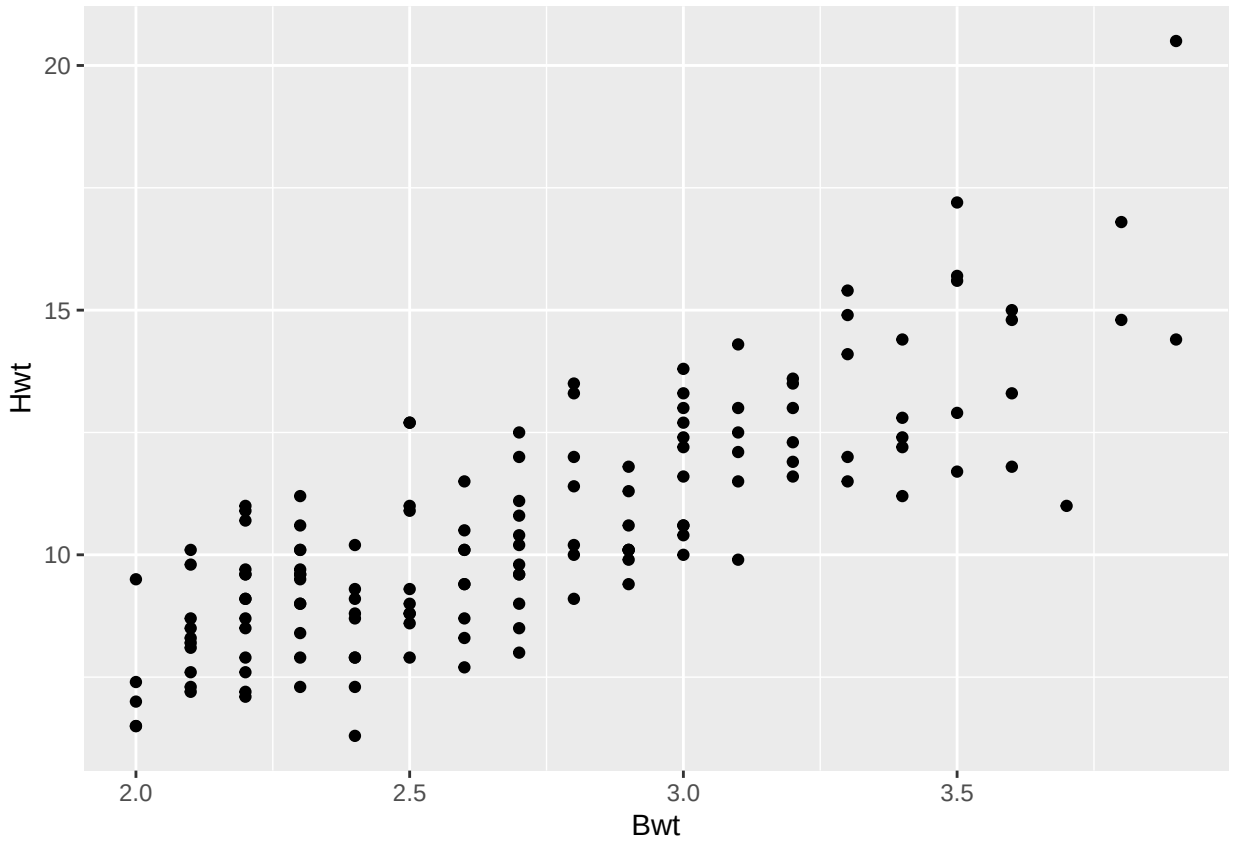
Your Answer:

```
WeightSummary <- cats %>%
    summarise(cats_body_mean = mean(Bwt), cats_body_sd = sd(Bwt))

WeightSummary
```

```
##   cats_body_mean cats_body_sd
## 1       2.723611    0.4853066
```

Now that we know how to summarise individual variables, we may want to start to look at relationships between variables. The go-to method to do this is a scatterplot. Run the code below and see what it does.

```
plot1 <- cats %>%
  ggplot(aes(x = Bwt, y = Hwt)) +
  geom_point()

plot1
```



Exercise 4

Comment on the relationship between body weight and heart weight, using the plot.

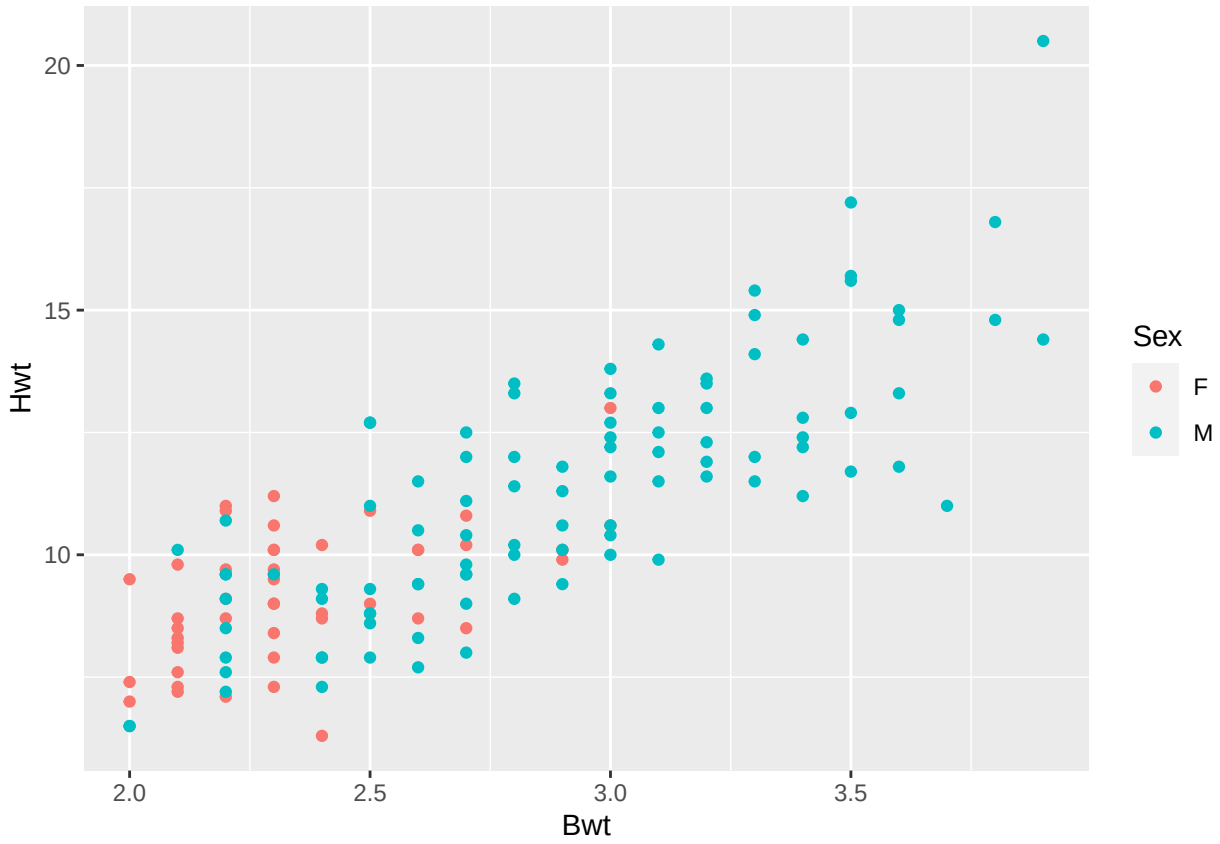
Your Answer: Here we see that there is a linear relationship between the two variables Bwt and Hwt. There is one outlier where we see the heart weight being above 20 and the body weight above 3.5.

Exercise 5

Amend the code given above, to colour the points on the plot by the sex of the cat. It may be useful to look at the documentation for the ggplot package. We do this by placing a question mark next to the package name and running the code, ie. "?ggplot".

Your Answer:

```
plot2 <- cats %>%
  ggplot(aes(x = Bwt, y = Hwt, colour=Sex)) +
  geom_point()

plot2
```

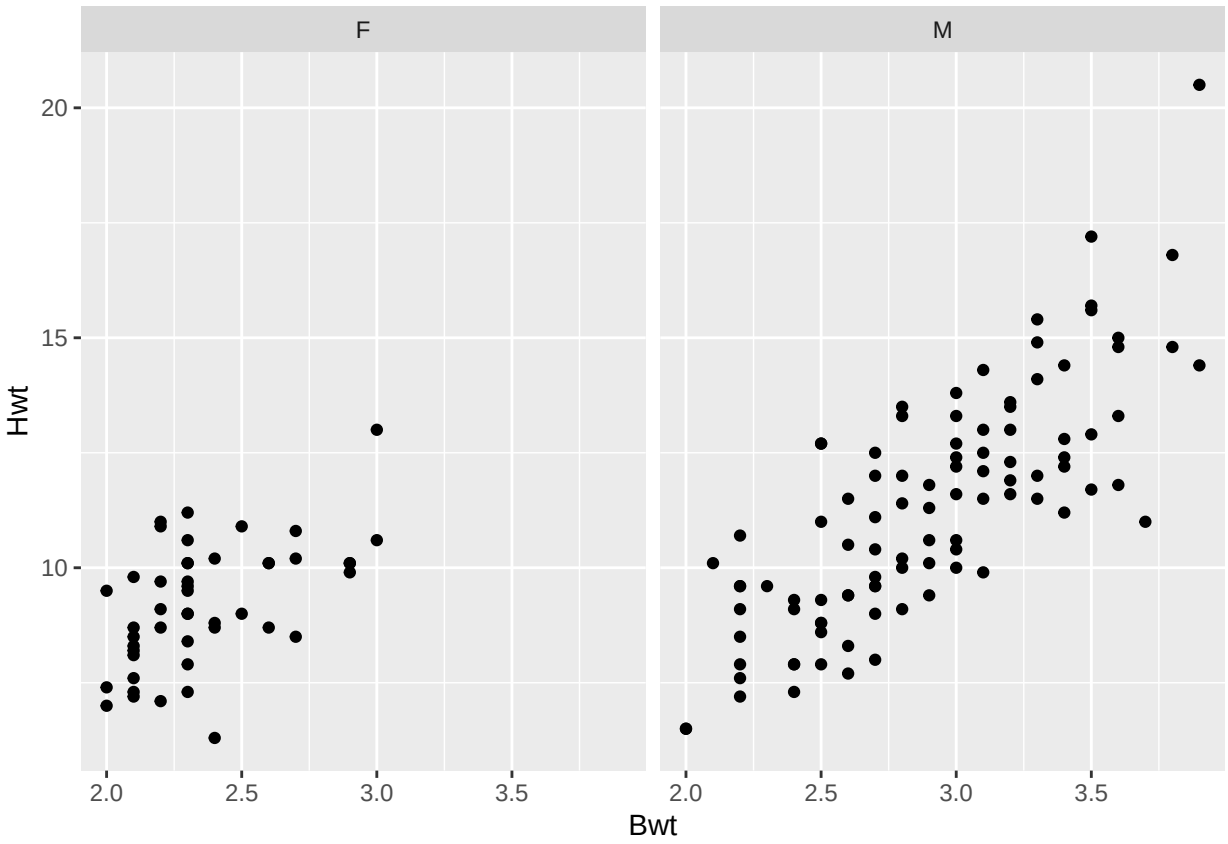


Exercise 6

Once again amend the code. This time make it so that there are two plots which show two different the points seperated by the sex of the cats. Do you notice anything different from when they were on one plot? Write a short explanation.

Your Answer:

```
plot3 <- cats %>%
  ggplot(aes(x = Bwt, y = Hwt)) +
  geom_point() +
  facet_wrap(~Sex)

plot3
```



We can see that in the female group now the relationship is not so linear and the two variables seem to be quite uncorrelated. The male group seems to still retain a linear relationship.
