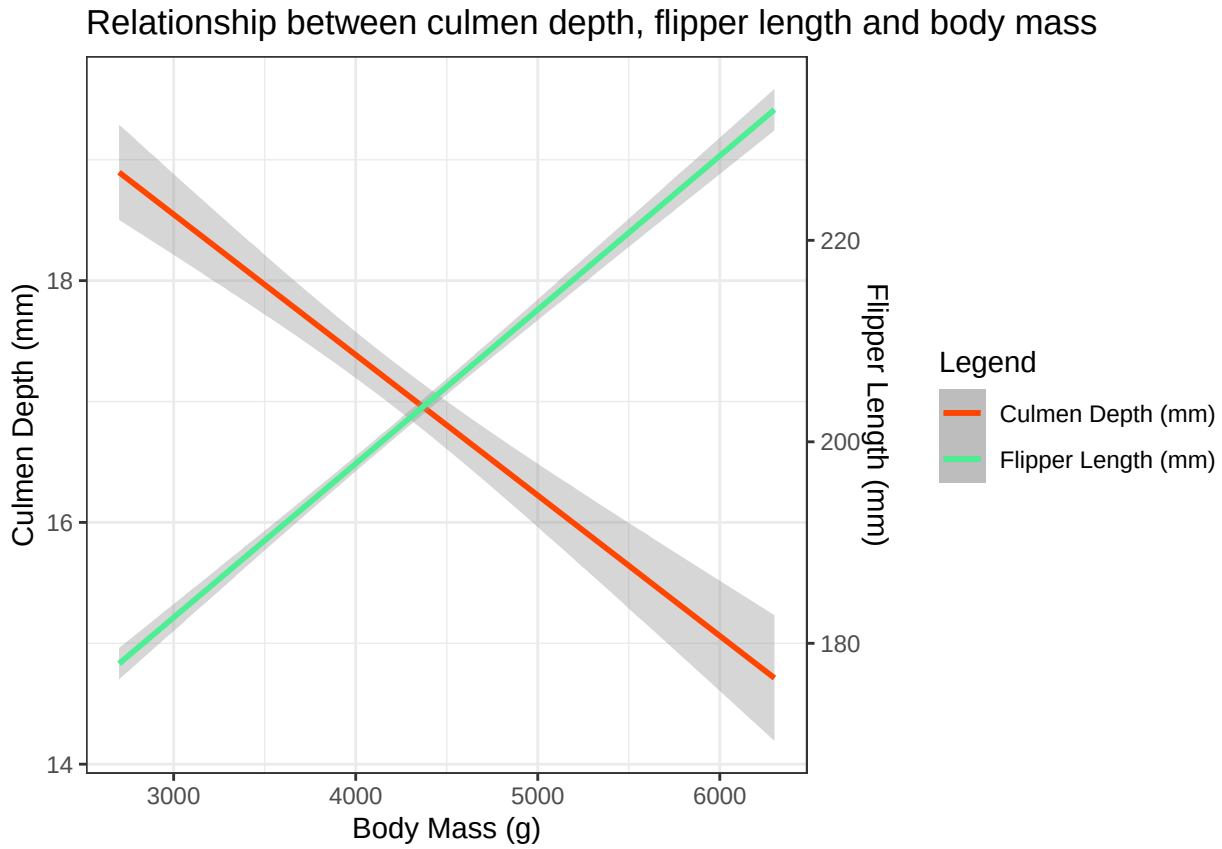
Penguin_Analysis

Candidate Number: 1053484

QUESTION 01: Data Visualisation for Science Communication




```
## pdf
##   2
```

This figure is a bad representation of the data; it plots two variables, culmen depth and flipper length, against body mass, showing the relationship between them. It makes it look like there is a negative correlation between culmen depth and body mass and that flipper length is inversely proportional to culmen depth. However, several aspects of this figure have misled the results.

One of which is that the Y axes don't start at 0; without reading the scale of each y-axis, it could look like the range of data is spread equally for both culmen depth and flipper length. This has been achieved by transforming the flipper length scale to bring an overlap to the two variables. (Wikipedia contributors, 2023)

Another important thing to note is that, unlike the linear model in the figure suggests, there isn't a visually clear linear relationship between culmen depth and body mass, at least when looking at all of the species together. If the data were divided between species, it would be evident that there is a positive correlation

between culmen depth and body mass within the species level, but the Gentoo species have skewed the Palmer penguin data due to their proportionally larger body mass and smaller culmen depth in comparison to the other species. The lack of data points on the graph also worsens this. If a scatter plot were used to show the data instead, there would be an evident split in the data points.

Reference: Wikipedia contributors. (2023, September 5). Misleading graph. Wikipedia. https://en.wikipedia.org/wiki/Misleading_graph

QUESTION 2: Data Pipeline

Introduction

Using the Palmer Penguins dataset I want to look at the relationship between body mass and flipper length for one species.

I will first explore the relationship between the two variables for all species, to understand whether there is a linear relationship between these two variables for any of the species in the data.

We need to install and load all the packages necessary to do this analysis.

```
#Install Packages
if(!require("ggplot2", character.only = TRUE)){
   install.packages("ggplot2")
}
if(!require("palmerpenguins", character.only = TRUE)){
   install.packages("palmerpenguins")
}
if(!require("janitor", character.only = TRUE)){
   install.packages("janitor")
}
if(!require("dplyr", character.only = TRUE)){
   install.packages("dplyr")
}
if(!require("gridExtra", character.only = TRUE)){
   install.packages("gridExtra")
}
if(!require("ragg", character.only = TRUE)){
   install.packages("ragg")
}
if(!require("stringr", character.only = TRUE)){
   install.packages("stringr")
}


#Load Packages
library(ggplot2)
library(palmerpenguins)
library(janitor)
library(dplyr)
library(gridExtra)
library(ragg)
library(stringr)
```

In order to explore the data I need to load it and clean it to ensure we only have the data we need to analyze.

```r
##Loading the data

# First we need to load the raw data from the "palmerpenguins" package

head(penguins_raw)
```

```
## # A tibble: 6 x 17
##   studyName `Sample Number` Species          Region Island Stage `Individual ID`
##   <chr>               <dbl> <chr>            <chr>  <chr>  <chr> <chr>
## 1 PAL0708                 1 Adelie Penguin ~ Anvers Torge~ Adul~ N1A1
## 2 PAL0708                 2 Adelie Penguin ~ Anvers Torge~ Adul~ N1A2
## 3 PAL0708                 3 Adelie Penguin ~ Anvers Torge~ Adul~ N2A1
## 4 PAL0708                 4 Adelie Penguin ~ Anvers Torge~ Adul~ N2A2
## 5 PAL0708                 5 Adelie Penguin ~ Anvers Torge~ Adul~ N3A1
## 6 PAL0708                 6 Adelie Penguin ~ Anvers Torge~ Adul~ N3A2
## # i 10 more variables: `Clutch Completion` <chr>, `Date Egg` <date>,
## #   `Culmen Length (mm)` <dbl>, `Culmen Depth (mm)` <dbl>,
## #   `Flipper Length (mm)` <dbl>, `Body Mass (g)` <dbl>, Sex <chr>,
## #   `Delta 15 N (o/oo)` <dbl>, `Delta 13 C (o/oo)` <dbl>, Comments <chr>
```

```r
#Save copy of raw data

write.csv(penguins_raw, "Data/penguins_raw.csv")

##Cleaning the raw data

#Load cleaning functions

source("Functions/Cleaning.R")

##Using functions in Functions\Cleaning.R

#We can clean the data and save it as a new dataset to ensure we don't change the raw data.

penguins_clean <- penguins_raw %>%
    clean_column_names() %>%
    shorten_species() %>%
    remove_empty_columns_rows()

#View clean data
head(penguins_clean)
```

```
## # A tibble: 6 x 17
##   study_name sample_number species region island    stage          individual_id
##   <chr>              <dbl> <chr>   <chr>  <chr>     <chr>          <chr>
## 1 PAL0708                1 Adelie  Anvers Torgersen Adult, 1 Egg ~ N1A1
## 2 PAL0708                2 Adelie  Anvers Torgersen Adult, 1 Egg ~ N1A2
## 3 PAL0708                3 Adelie  Anvers Torgersen Adult, 1 Egg ~ N2A1
## 4 PAL0708                4 Adelie  Anvers Torgersen Adult, 1 Egg ~ N2A2
## 5 PAL0708                5 Adelie  Anvers Torgersen Adult, 1 Egg ~ N3A1
## 6 PAL0708                6 Adelie  Anvers Torgersen Adult, 1 Egg ~ N3A2
## # i 10 more variables: clutch_completion <chr>, date_egg <date>,
## #   culmen_length_mm <dbl>, culmen_depth_mm <dbl>, flipper_length_mm <dbl>,
## #   body_mass_g <dbl>, sex <chr>, delta_15_n_o_oo <dbl>, delta_13_c_o_oo <dbl>,
## #   comments <chr>
```

```
#Save clean data into working directory
write.csv(penguins_clean,
          "Data/penguins_clean.csv")
```

Now I have the clean data we can use this to explore the dataset and create a subset with which we can work with.

```
## Create a subset which only includes columns for species, flipper length and body mass.

#Create a subset using function from Cleaning.R

subset_flipper_body_species<- subset_columns(
  penguins_clean, c("species", "flipper_length_mm",
                    "body_mass_g"))

#View subset

head(subset_flipper_body_species)
```

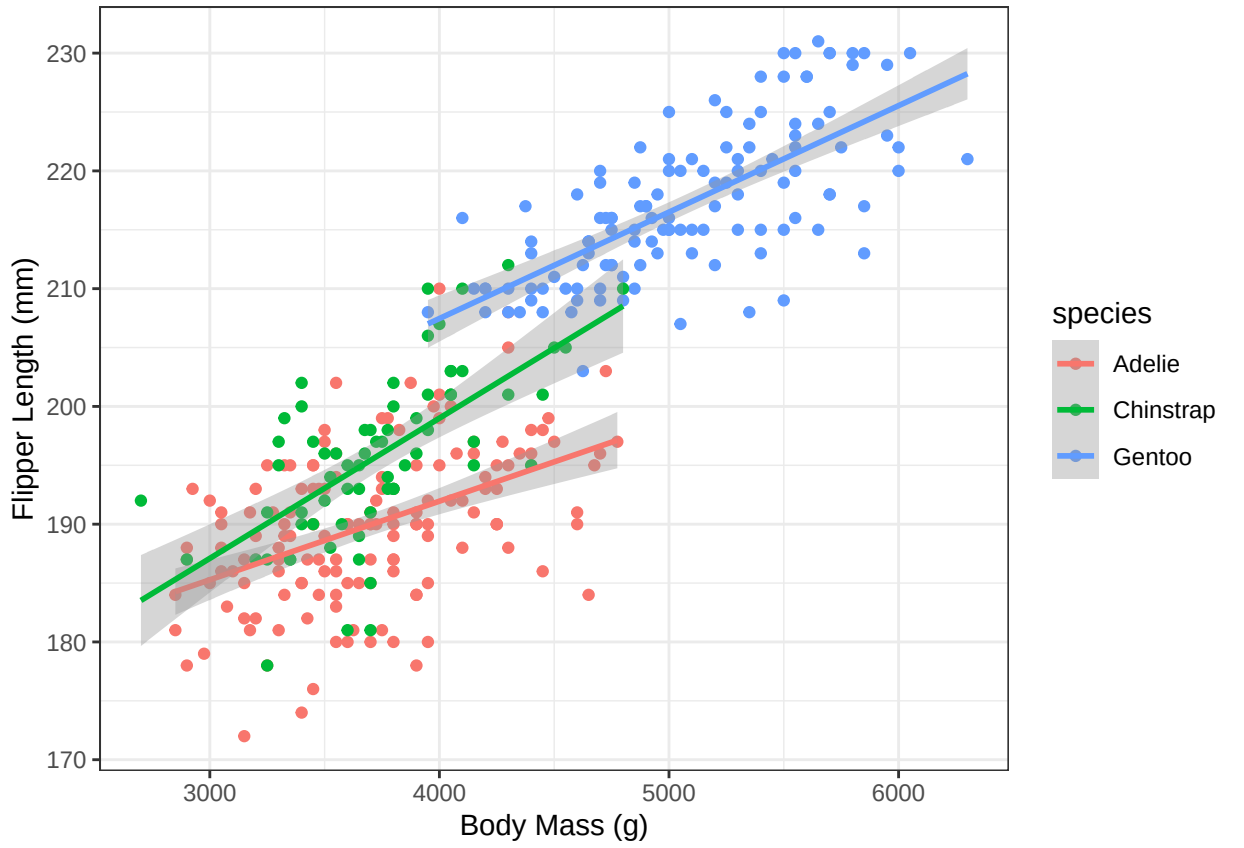
```
## # A tibble: 6 x 3
##   species flipper_length_mm body_mass_g
##   <chr>               <dbl>       <dbl>
## 1 Adelie                181        3750
## 2 Adelie                186        3800
## 3 Adelie                195        3250
## 4 Adelie                 NA          NA
## 5 Adelie                193        3450
## 6 Adelie                190        3650
```

We can see that this subset contains all the data we need, we can now go on to plot the subset.

```
#Load plotting functions
source("Functions/Plotting.R")

#Plot scatter plot using function
plot1<-plot_body_flipper(subset = subset_flipper_body_species)

plot1
```




```
#Save the figure using function
save_plot_png(plot1,
              "Plots/exploratory_figure.png", size= 15, res=600, scaling=1)
```

```
## pdf
##   2
```

Looking at the data as a whole I can see that there is a linear relationship within species between body mass and flipper length. I will look specifically at the Chinstrap species for my hypothesis

Hypothesis

I hypothesize that there is a correlation between body mass and flipper length for the Chinstrap species.

- The null hypothesis: There is no correlation between body mass and flipper length
- Alternative hypothesis: There is a correlation between body mass and flipper length

To test this hypothesis I will create a subset which only includes body mass and flipper length data for the Chinstrap species

```
#Filter by Chinstrap species using function in Cleaning.R
chinstrap_subset<-filter_by_species(
  subset_flipper_body_species, "Chinstrap")

#View subset
head(chinstrap_subset)
```

```
## # A tibble: 6 x 3
##   species   flipper_length_mm body_mass_g
```

```
##   <chr>                <dbl>       <dbl>
## 1 Chinstrap              192        3500
## 2 Chinstrap              196        3900
## 3 Chinstrap              193        3650
## 4 Chinstrap              188        3525
## 5 Chinstrap              197        3725
## 6 Chinstrap              198        3950
```

Statistical Methods

To test this hypothesis I will test the correlation with Pearsons Correlation Coefficient test. I first need to check that the data follows the assumptions of the test:

- Normality: I can test the distribution of both variables with the Shapiro-Wilk test

```
#Starting with the body mass variable
#Define the variables
chinstrap_body_mass<-chinstrap_subset$body_mass_g
chinstrap_flipper_length<-chinstrap_subset$flipper_length_mm
#Shapiro wilks test both
body_mass_test<-shapiro.test(chinstrap_body_mass)
flipper_length_test<-shapiro.test(chinstrap_flipper_length)
#Print results
print(body_mass_test)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  chinstrap_body_mass
## W = 0.98449, p-value = 0.5605
```

```
print(flipper_length_test)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  chinstrap_flipper_length
## W = 0.98891, p-value = 0.8106
```

I can see that the p-value for both is higher than 0.05 therefore we cannot reject the null hypothesis that the data follows a normal distribution, therefore we can assume the data follows a normal distibution.

- Linearity: Looking at the exploratory figure we can see the linear relationship between flipper length and body mass for the Chinstrap species.

Knowing that my subset follows the assumptions of the Pearson Correlation Coefficient test we can apply the test.

```
#Run Pearson's Correlation Coefficient test

correlation_test<-cor.test(chinstrap_body_mass,chinstrap_flipper_length)

#View results

print(correlation_test)
```

```
##
##  Pearson's product-moment correlation
```

```
##
## data:  chinstrap_body_mass and chinstrap_flipper_length
## t = 6.7947, df = 66, p-value = 3.748e-09
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4759352 0.7632368
## sample estimates:
##       cor
## 0.6415594
```

The results show that the correlation coefficient is estimated at 0.64 with a p-value of 3.74e-09

Results

I will plot our Chinstrap subset with a linear regression and our Correlation coefficient results

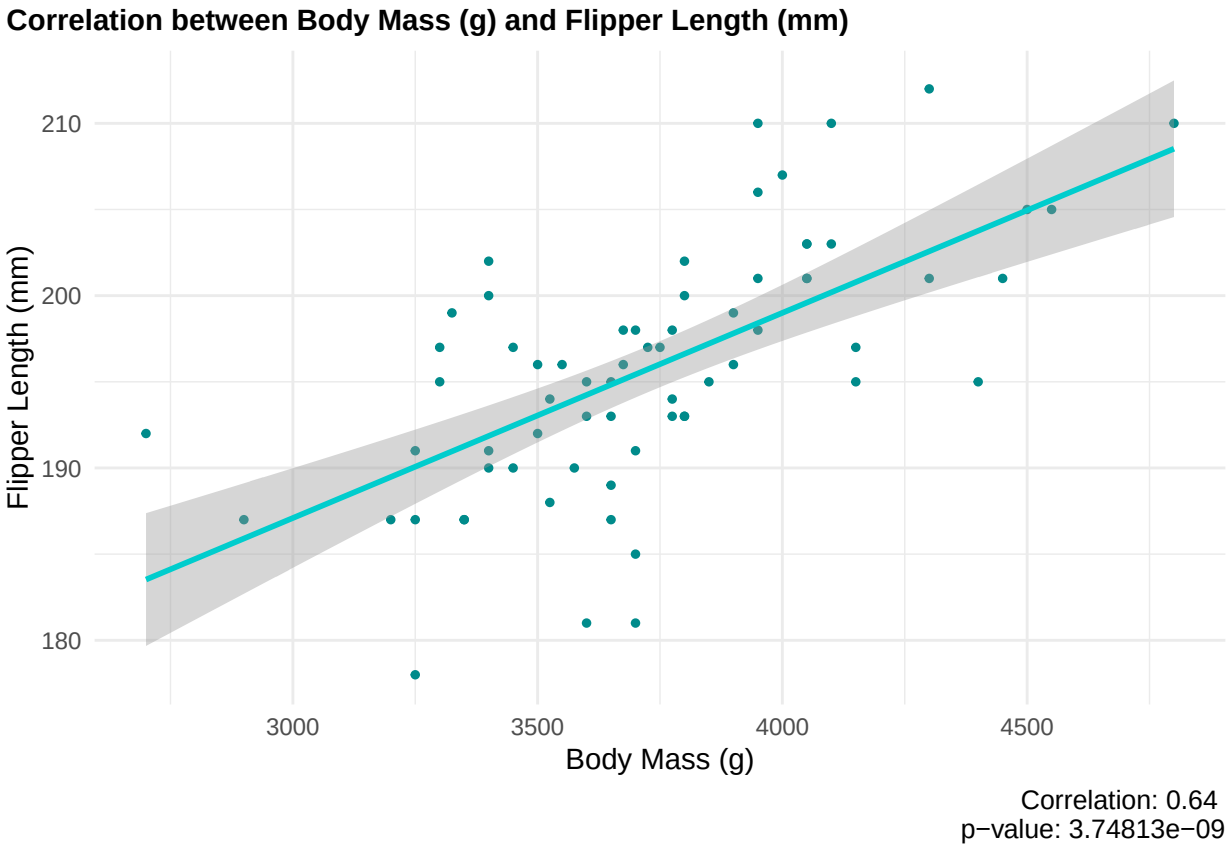
```
source("Functions/Plotting.R")

#Create results figure

chinstrap_correlation_figure<-
  scatter_with_correlation(chinstrap_subset,
                           x="body_mass_g",
                           y="flipper_length_mm")

#View figure

print(chinstrap_correlation_figure)
```




```
#Save figure
save_plot_png(chinstrap_correlation_figure,
              "Plots/stats_figure.png", 15, 300,
              scaling =1)
```

```
## pdf
##   2
```

Discussion

With the p-value of 3.748e-09 the null hypothesis can be rejected and we can confirm that there is a correlation between flipper length and body size for the Chinstrap species.

A correlation coefficient estimate of 0.64 shows that there is a strong positive correlation between flipper length and body size within this species.

Conclusion

In this data analysis of the Chinstrap species a strong positive correlation was found between body mass and flipper length. The data followed the assumptions of a Pearson correlation coefficient test and therefore we can accept the alternative hypothesis that there is a correlation between these two variables. I think this analysis will be useful for further analyses using this dataset which look at the correlations between two continuous variables.

QUESTION 3: Open Science

a) GitHub

*Upload your RProject you created for **Question 2** and any files and subfolders used to GitHub. Do not include any identifiers such as your name. Make sure your GitHub repo is public.*

GitHub link:

You will be marked on your repo organisation and readability.

b) Share your repo with a partner, download, and try to run their data pipeline.

*Partner's GitHub link: https://github.com/Candidate1066540/ReproducibleScienceMT23_USE

c) Reflect on your experience running their code. (300-500 words)

Using the introductory section, my partner clearly outlined the Palmer penguin dataset and what it was compiled of, which helped me understand the rest of the analysis. I understood their data pipeline well and could clearly see how the raw data was loaded and saved. This raw data was then processed using several cleaning methods, including removing unnecessary columns that weren't required for the analysis. This produced a clean dataset, which was saved separately from the raw data. They then used the clean data to analyse two variables in the dataset, in this case, culmen depth and body mass. A clear hypothesis was constructed by which they wanted to test these variables in the dataset. They analysed the relationship between the two variables visually through the use of an exploratory figure as well as running a statistical analysis. They also visualised the statistical analysis results by plotting a linear regression. To ensure the completion of the data pipeline, the results figure should be saved in the repository so it can be easily accessed for reference.

The only issue I had running the code fell in a cleaning function named "remove_NA". This function wasn't called upon in the code, but when I tried to use the function, I found an error code. I needed to change "penguins_clean" to "penguins_data" to fix this.

```
remove_NA <- function(penguins_clean) {
  penguins_data %>%
    na.omit()
}
```

To make the code more reproducible, I would suggest piping multiple cleaning functions and assigning them to one function. This would mean that only one function would need to be called if the data needed to be cleaned again. By saving this function in a separate cleaning functions script, it could be used across multiple analyses just by sourcing those functions from the script and not by searching for the function elsewhere, which could be easily lost. I would also suggest doing the same for the plots. This would make the plots more reproducible; this way, the code for the plot wouldn't have to be copied multiple times when reproducing the analysis. It would also make the elements standard across each reproduced analysis, and the function would need to be edited to change them.

To make the code more understandable, I would suggest including the function "clean_names()" in the cleaning stage, making the column names uniform and easier to read and paste. I would also suggest breaking down aspects of the cleaning stage to explain what parts of the data are changing so that someone finds it easier to know when to use those functions.

If I needed to alter my partner's figure by changing specific aspects, I think it would be quite easy; the plot isn't in a function, so I can edit the main script to change elements of the plot, such as colour and axis labels. This is risky for the main script; if I wanted to reproduce the results, it would be easy to get a different graph than the original analysis.

d) Reflect on your own code based on your experience with your partner's code and their review of yours. (300-500 words)

My partner suggested stating the obvious regarding sourcing my folders, which hold functions and data. They said this would benefit those with less knowledge of R and programming generally. I agree that I didn't explain the purpose of loading these files, how they provided the functions for cleaning my raw data, and how they created my plots. Further illustrating this would ensure that someone could edit the functions to replicate my analysis using different variables or analysing other datasets.

Another suggestion made by my partner was that the correlation plot could have been better coloured to create more visual contrast for more accessible viewing. I agree that making the data points more visually prominent from the background would ensure clear results and reduce the risk of misinterpreting the figure.

Reflecting on my experience with my partner's analysis and what they did well, I believe that my hypothesis should have been formalised for my statistical test and that I could've further outlined my statistical results. This would have made my analysis more thorough. I also believe that I could've introduced the dataset better by showing the names of the columns to showcase all the variables that could be analysed and showing the total number of rows to showcase the sample size. Therefore, someone reading my analysis has the information to decide whether my sample size was sufficient for the analysis.

I also reflect that parts of my code, like plotting functions, may be challenging to use without the context of my analysis for reference. I would need to explain more precisely what variables the function will be used to plot, especially for the vaguely named function "scatter_with_correlation". I should have outlined the x and y variables within the function and named the function something more related to those variables (e.g. "plot_body_mass_with_flipper_length"), removing any chance of confusion about when the function would be useful when reproducing the analysis.

I learned that for code to be reproduced by others, it is better to overexplain each process step than not explain enough. It is important to ensure that someone can understand how results were gathered by tracing the data pipeline. Therefore, if they wanted to test whether they could reproduce the same results from the data, they would know exactly how to achieve them. I also learned that functions can be useful when performing the same analyses multiple times. They can be saved into folders, which can be referred to when doing different analyses. Using clearly labelled folders to organise data, functions, and results is also essential for reproducibility so that others can easily access all stages of your analysis.