Penguins Assignment Question 4

2022-12-06

#Loading the packages

```
library(ggplot2)
library(palmerpenguins)
suppressPackageStartupMessages(library(janitor))
suppressPackageStartupMessages(library(dplyr))
library(tidyr)
library(ragg)
```

```
#Setting the working directory
setwd("~/Biology/Year 3/Year 3 R/Computing Assessments/PenguinProject")
```

##Question 4: Run a statistical test on the Palmer Penguins dataset and produce a figure to explain it.

```
penguins_raw
```

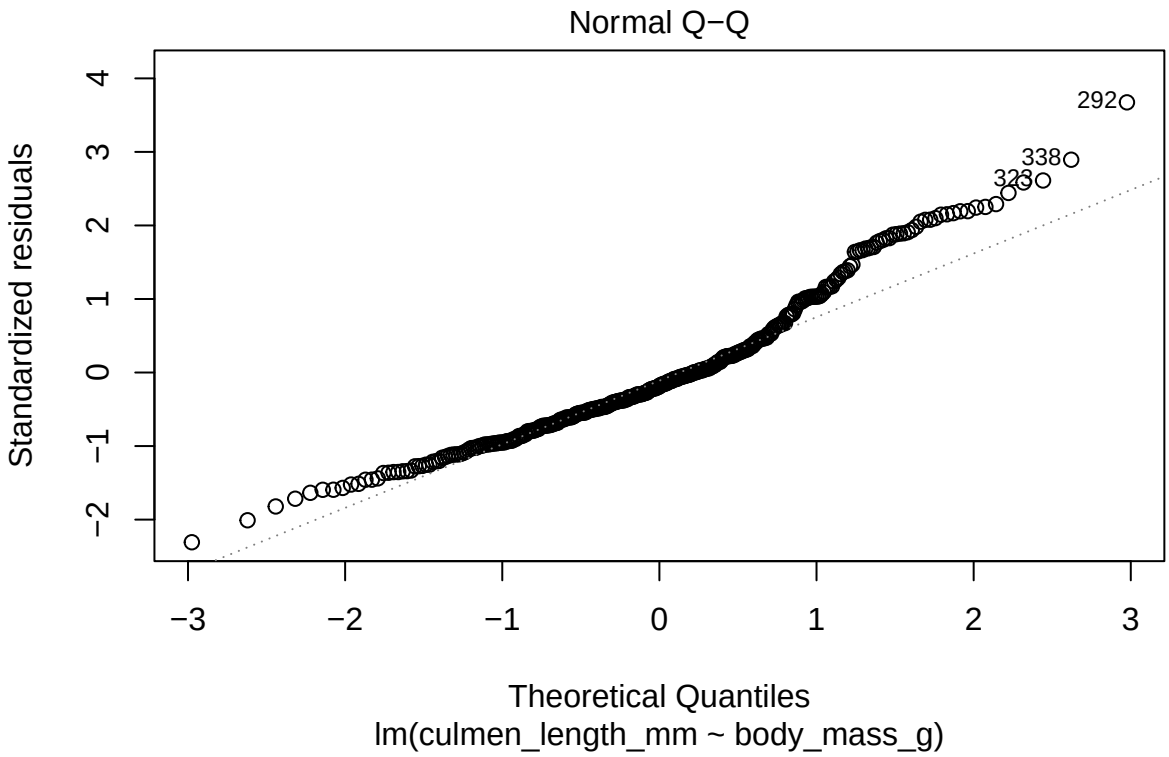
```
## # A tibble: 344 x 17
##    studyName Sample Num~1 Species Region Island Stage Indiv~2 Clutc~3 `Date Egg`
##    <chr>            <dbl> <chr>   <chr>  <chr>  <chr> <chr>   <chr>   <date>
##  1 PAL0708              1 Adelie~ Anvers Torge~ Adul~ N1A1    Yes     2007-11-11
##  2 PAL0708              2 Adelie~ Anvers Torge~ Adul~ N1A2    Yes     2007-11-11
##  3 PAL0708              3 Adelie~ Anvers Torge~ Adul~ N2A1    Yes     2007-11-16
##  4 PAL0708              4 Adelie~ Anvers Torge~ Adul~ N2A2    Yes     2007-11-16
##  5 PAL0708              5 Adelie~ Anvers Torge~ Adul~ N3A1    Yes     2007-11-16
##  6 PAL0708              6 Adelie~ Anvers Torge~ Adul~ N3A2    Yes     2007-11-16
##  7 PAL0708              7 Adelie~ Anvers Torge~ Adul~ N4A1    No      2007-11-15
##  8 PAL0708              8 Adelie~ Anvers Torge~ Adul~ N4A2    No      2007-11-15
##  9 PAL0708              9 Adelie~ Anvers Torge~ Adul~ N5A1    Yes     2007-11-09
## 10 PAL0708             10 Adelie~ Anvers Torge~ Adul~ N5A2    Yes     2007-11-09
## # ... with 334 more rows, 8 more variables: `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>, and abbreviated variable names
## #   1: `Sample Number`, 2: `Individual ID`, 3: `Clutch Completion`
```

```
#want to save this penguins_raw dataset
write.csv(penguins_raw, paste0("data_raw/penguins_raw.csv"))
#Saved penguins raw to a data raw folder

#Now we are defining the cleaning function and what it will do to penguins_raw
cleaning <- function(penguins_raw) {
  penguins_raw %>%
    select(-starts_with("delta")) %>%
```

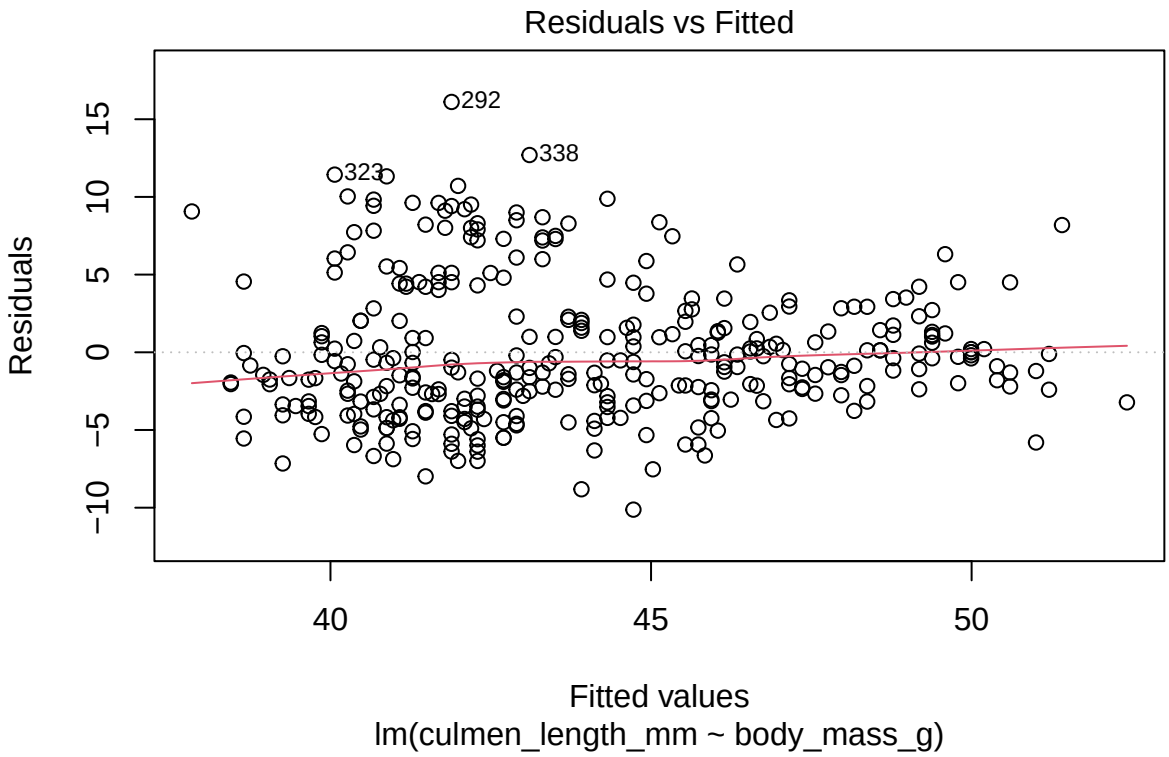
```
    select(-Comments)%>%
    clean_names()}

#This is within the cleaning.r function that is saved separately. It removes NA
#values and selects the variables I am including in my data analysis
remove_empty_mass_and_length <- function(penguins_clean){
    penguins_clean %>%
    filter(!is.na(culmen_length_mm)) %>%
    filter(!is.na(body_mass_g)) %>%
    select(body_mass_g, culmen_length_mm, species)
}

source("functions/cleaning.r")
#This specifies where the cleaning function is saved

#We are then applying this function to the penguins_raw dataset
penguins_clean <- cleaning(penguins_raw)
penguins_now_clean <- remove_empty_mass_and_length(penguins_clean)

#Saving penguins clean dataset
write.csv(penguins_clean, paste0("data_clean/penguins_clean.csv"))

#Saving penguins now clean dataset
write.csv(penguins_now_clean, paste0("data_clean/penguins_now_clean.csv"))
```

Now the data has been cleaned, I am going to test to see if there is a significant correlation between body mass and culmen length. I will use Pearson's correlation coefficient because the two variables are numerical and continuous

```
penguins_model <- lm(culmen_length_mm ~ body_mass_g, data=penguins_now_clean)
#Making a linear regression model for body mass and culmen length

#Can do Pearson's correlation coefficient if the assumption of bivariate normality is not breached.
plot(penguins_model, which =2)
```




```
plot(penguins_model, which=1)
```



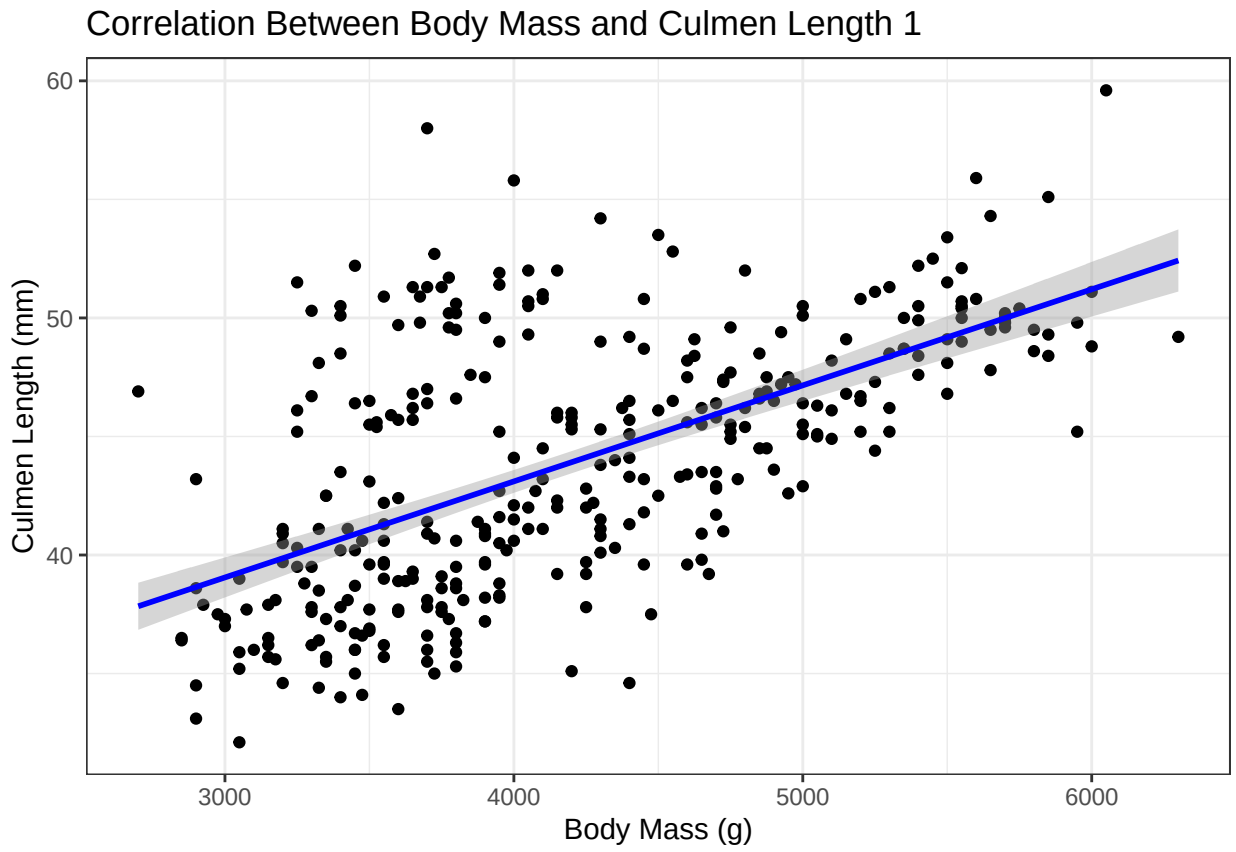

```
#The assumptions of normality and homogeneity of variance are well-met for this model

#Performing the Pearson's correlation coefficient
cor.test(penguins_clean$body_mass_g, penguins_clean$culmen_length_mm, method=c("pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  penguins_clean$body_mass_g and penguins_clean$culmen_length_mm
## t = 13.654, df = 340, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5220040 0.6595358
## sample estimates:
##       cor
## 0.5951098
```

```
#That has given me a p value of 2.2e-16. Suggests that there is a significant correlation
#between the two variables. Will now visualise this with a scatter plot


#Defining my functions for plotting a graph for all the penguins, and broken down by penguin species
plot_penguin_graph1 <- function(penguins_now_clean){
  penguins_now_clean %>%
    ggplot(
```

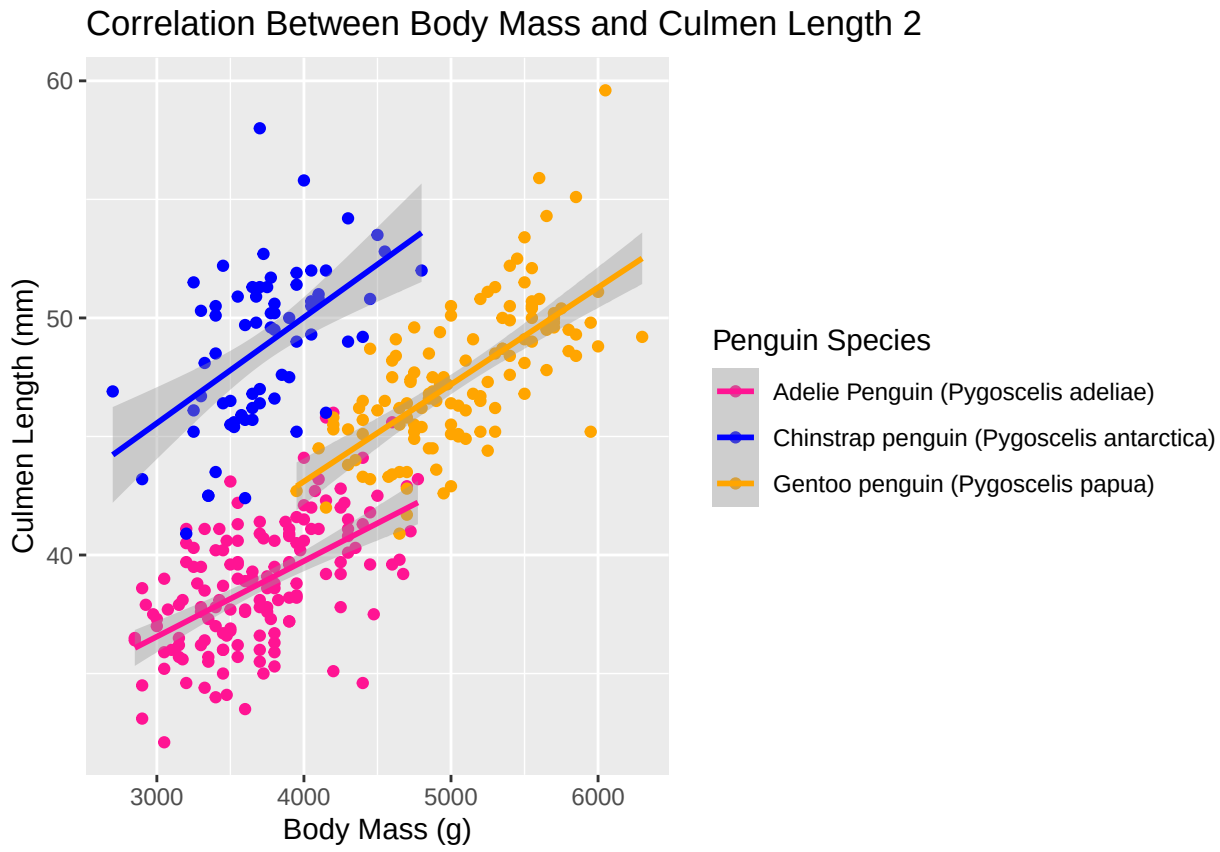
```
          aes(x=body_mass_g, y=culmen_length_mm))+geom_point()+
    geom_smooth(method="lm", colour="blue")+labs(x="Body Mass (g)", y="Culmen Length (mm)",
    title="Correlation Between Body Mass and Culmen Length 1")+
    theme_bw()
}

plot_penguin_graph2 <- function(penguins_now_clean){
  penguins_now_clean %>%
    ggplot(
          aes(x=body_mass_g, y=culmen_length_mm, colour=species))+ geom_point()+
    scale_colour_manual(values=c("deeppink", "blue", "orange"))+ geom_smooth(method="lm")+
    labs(x="Body Mass (g)", y="Culmen Length (mm)",
    title="Correlation Between Body Mass and Culmen Length 2", colour="Penguin Species")
}


source("functions/plotting.r")
#This is where my function is saved

#Calling the plotting function to make penguin_graph1 and penguin_graph2
penguin_graph1 <- plot_penguin_graph1(penguins_now_clean)
penguin_graph2 <- plot_penguin_graph2(penguins_now_clean)

#Looking at the graphs
penguin_graph1
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
penguin_graph2
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



The output of the Pearson's correlation coefficient gave a P value of 2.2e-16. This is less than 0.05, so we can reject the null hypothesis that there is no significant correlation between body mass and culmen length.

When this result is visualised with a scatterplot and regression lines, we see that there is a positive correlation between body mass and culmen length such that larger penguins have a longer culmen. The second scatterplot breaks down this positive correlation for each species, so we can see that the correlation holds true across all species of penguins.

```
#Defining a function for saving penguin graph 1 and penguin graph 2

save_graph1_png <- function(penguins_now_clean,
                                filename, size, res, scaling){
  agg_png("figures/penguin_graph1.png",
                    width   =  25,
                    height  =  20,
                    units   =  "cm",
                    res     =  600,
                    scaling =  1)
  penguin_graph1 <- plot_penguin_graph1(penguins_now_clean)
  print(penguin_graph1)
  dev.off()
}

save_graph2_png <- function(penguins_now_clean,
                                filename, size, res, scaling){
  agg_png("figures/penguin_graph2.png",
                    width   =  25,
```

```
                    height  =  20,
                    units   =  "cm",
                    res     =  600,
                    scaling =  1)
  penguin_graph2 <- plot_penguin_graph2(penguins_now_clean)
  print(penguin_graph2)
  dev.off()
}

source("functions/saving.r")
#this is where my saving function is stored

#Calling the saving function to save penguin graph 1 and penguin graph 2 as png files
penguin_graph1.png <- save_graph1_png(penguins_now_clean)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
penguin_graph2.png <- save_graph2_png(penguins_now_clean)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
#my graphs have now been saved in the "figures" folder
```