Contents

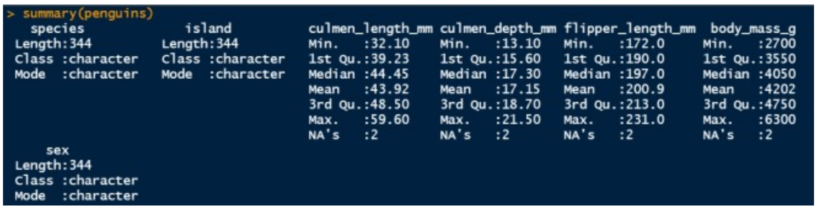
0. Load dataset and library on workspace.

```
library(palmerpenguins) # for data
library(dplyr) # for data-handling
library(corrplot) # for correlation plot
library(GGally) # for parallel coordinate plot
library(e1071) # for svm

data(penguins) # load pre-processed penguins
```

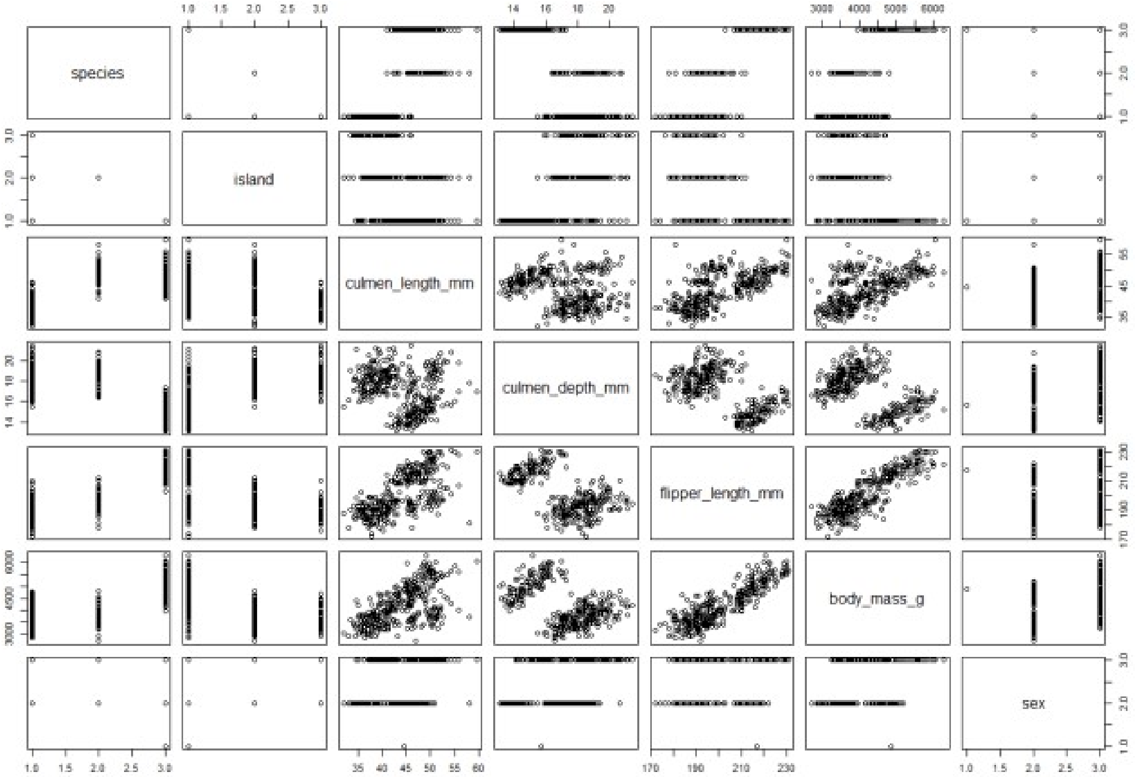
`palmerpenguins` have 2 data `penguins`, `penguins_raw` , and as you can see from their name, `penguins` is pre-processed data.

1. See the `summary` and `plot` of Dataset

```
summary(penguins)
plot(penguins)
```



```
> summary(penguins)
   species              island            culmen_length_mm culmen_depth_mm flipper_length_mm  body_mass_g
 Length:344         Length:344         Min.   :32.10    Min.   :13.10   Min.   :172.0     Min.   :2700
 Class :character   Class :character   1st Qu.:39.23    1st Qu.:15.60   1st Qu.:190.0     1st Qu.:3550
 Mode  :character   Mode  :character   Median :44.45    Median :17.30   Median :197.0     Median :4050
                                       Mean   :43.92    Mean   :17.15   Mean   :200.9     Mean   :4202
                                       3rd Qu.:48.50    3rd Qu.:18.70   3rd Qu.:213.0     3rd Qu.:4750
                                       Max.   :59.60    Max.   :21.50   Max.   :231.0     Max.   :6300
                                       NA's   :2        NA's   :2       NA's   :2         NA's   :2
     sex
 Length:344
 Class :character
 Mode  :character
```



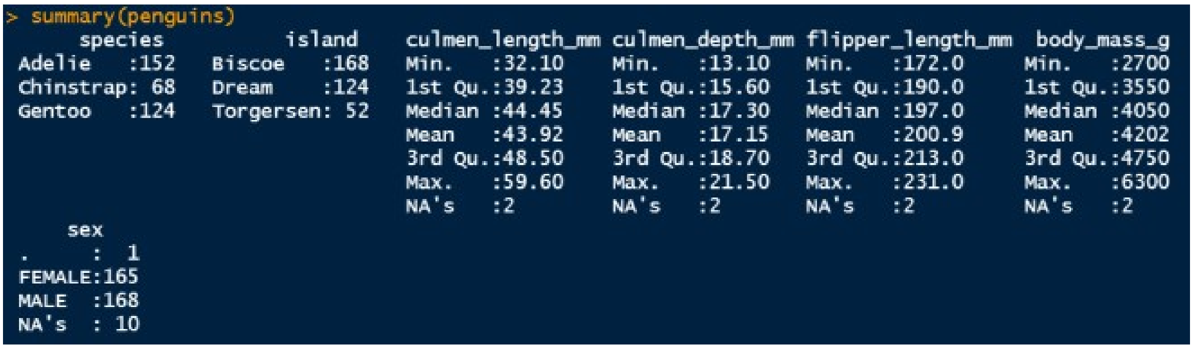
It seems `species` , `island` and `sex` is categorical features.
and remaining for numerical features.

2. Set the format of feature

```
penguins$species <- as.factor(penguins$species)
penguins$island <- as.factor(penguins$island)
```

```
penguins$sex <- as.factor(penguins$sex)

summary(penguins)
plot(penguins)
```

and see `summary` and `plot` again. note that result of `plot` is same.



```
> summary(penguins)
      species          island    culmen_length_mm culmen_depth_mm flipper_length_mm  body_mass_g
 Adelie   :152   Biscoe   :168   Min.   :32.10    Min.   :13.10   Min.   :172.0     Min.   :2700
 Chinstrap: 68   Dream    :124   1st Qu.:39.23    1st Qu.:15.60   1st Qu.:190.0     1st Qu.:3550
 Gentoo   :124   Torgersen: 52   Median :44.45    Median :17.30   Median :197.0     Median :4050
                                 Mean   :43.92    Mean   :17.15   Mean   :200.9     Mean   :4202
                                 3rd Qu.:48.50    3rd Qu.:18.70   3rd Qu.:213.0     3rd Qu.:4750
                                 Max.   :59.60    Max.   :21.50   Max.   :231.0     Max.   :6300
                                 NA's   :2        NA's   :2       NA's   :2         NA's   :2
     sex
 .     :  1
 FEMALE:165
 MALE  :168
 NA's  : 10
```

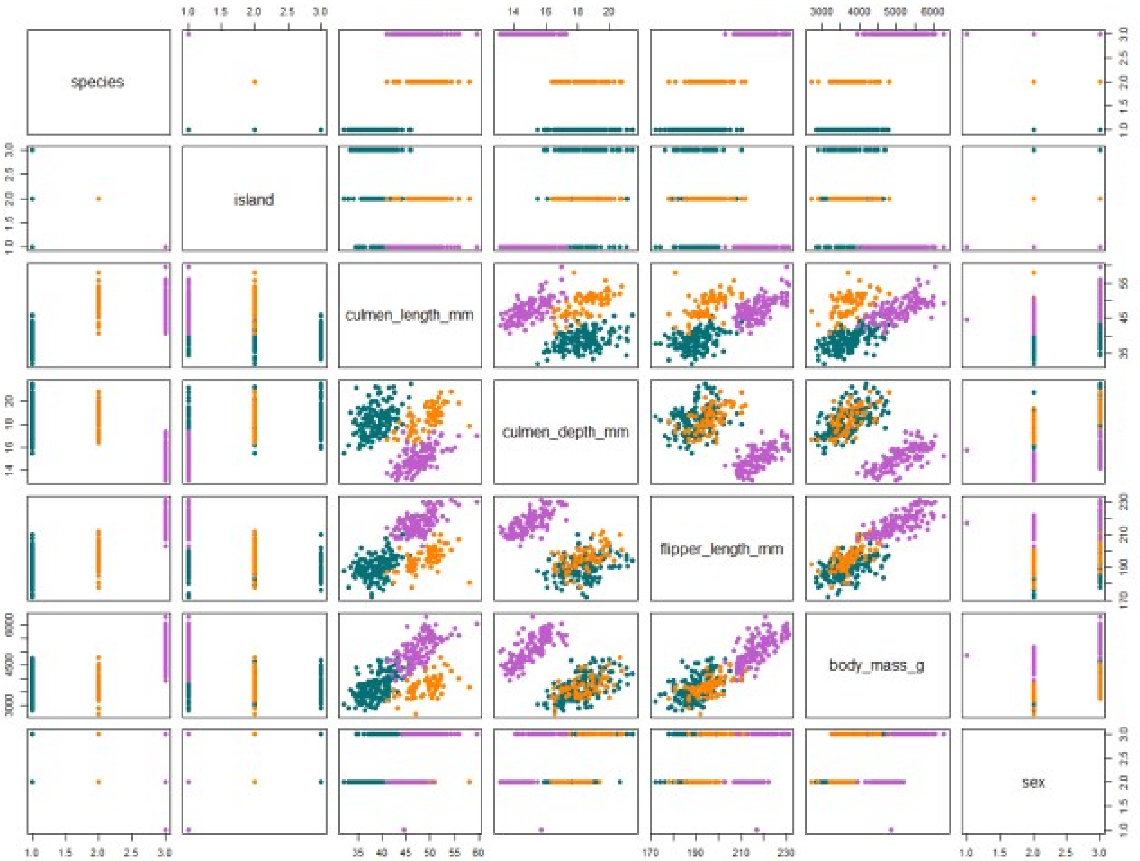
There's unwanted `NA` and `.` values in some features.

3. *Remove not necessary datas (in this tutorial,* `NA`)

```
penguins <- penguins %>% filter(sex == 'MALE' | sex == 'FEMALE')
summary(penguins)
```

And here, I additionally defined color values for each penguins to see better `plot` result

```
# Green, Orange, Purple
pCol <- c('#057076', '#ff8301', '#bf5ccb')
names(pCol) <- c('Gentoo', 'Adelie', 'Chinstrap')
plot(penguins, col = pCol[penguins$species], pch = 19)
```



Now, plot results are much better to give insights.

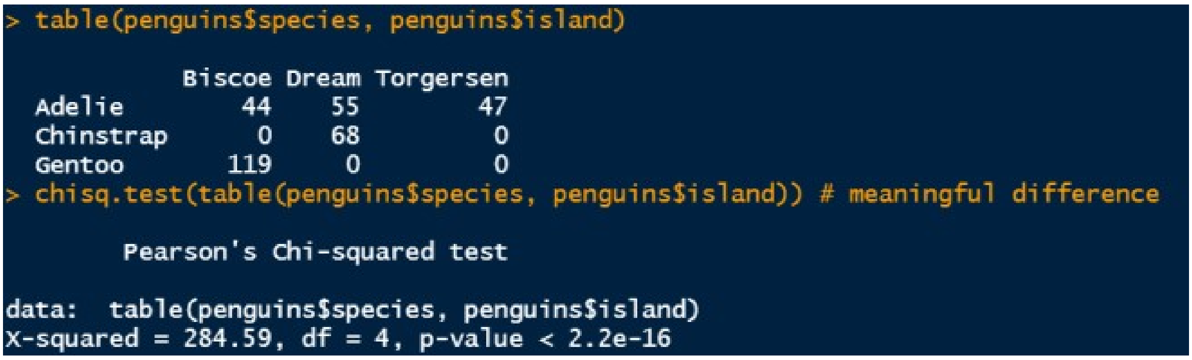
Note that, other pre-process step may requires for different datasets.

4. See relation of categorical features

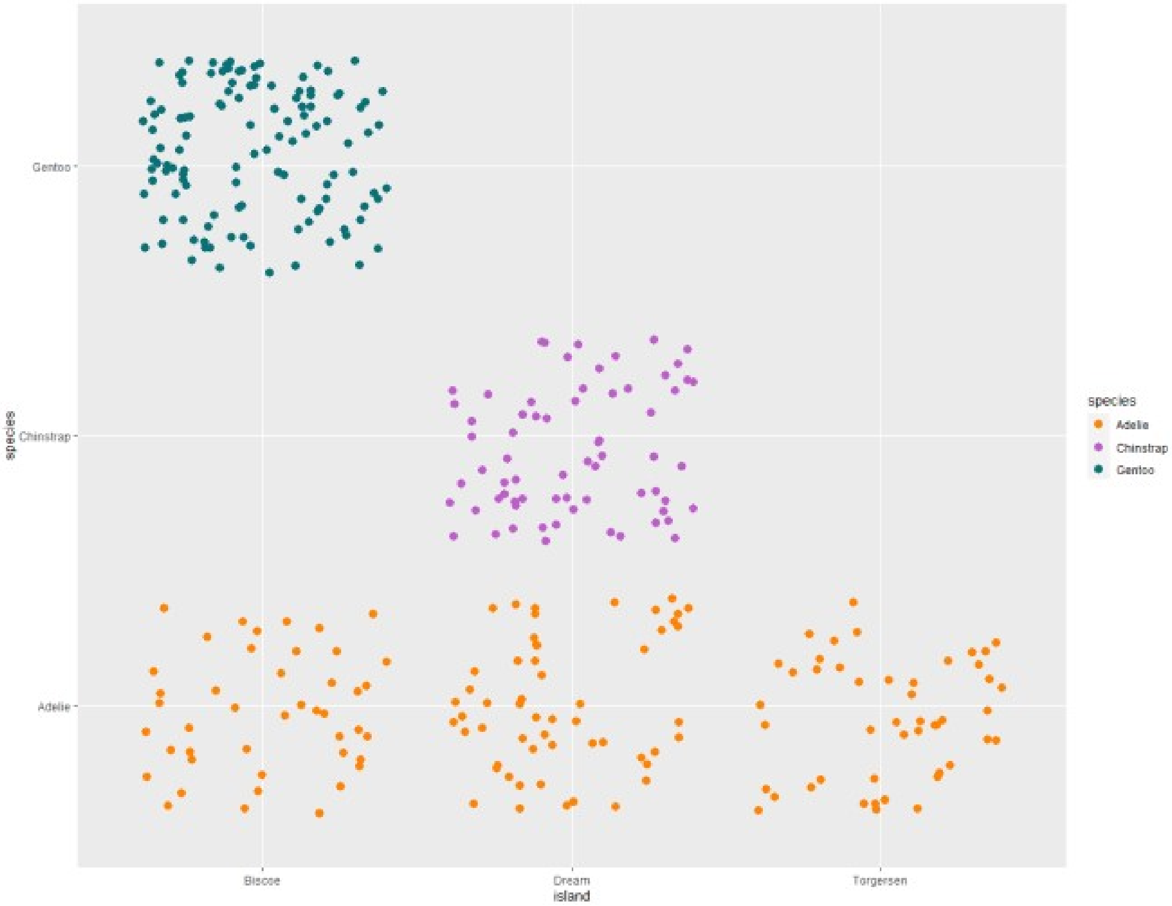
My first purpose of analysis this penguin is `species`
So, I will try to see relation between `species` and other categorical values

4-1. `species`, `island`

```
table(penguins$species, penguins$island)
chisq.test(table(penguins$species, penguins$island)) # meaningful difference

ggplot(penguins, aes(x = island, y = species, color = species)) +
  geom_jitter(size = 3) +
  scale_color_manual(values = pCol)
```



```
> table(penguins$species, penguins$island)

            Biscoe Dream Torgersen
  Adelie        44    55        47
  Chinstrap      0    68         0
  Gentoo       119     0         0
> chisq.test(table(penguins$species, penguins$island)) # meaningful difference

	Pearson's Chi-squared test

data:  table(penguins$species, penguins$island)
X-squared = 284.59, df = 4, p-value < 2.2e-16
```



Wow, there's strong relationship between `species` and `island`

– `Adelie` lives in every island
– `Gentoo` lives in only `Biscoe`

– `Chinstrap` lives in only `Dream`

4-2 & 4.3.
However, `species` and `sex` or `sex` and `island` did not show any meaningful relation.
You can try following codes.

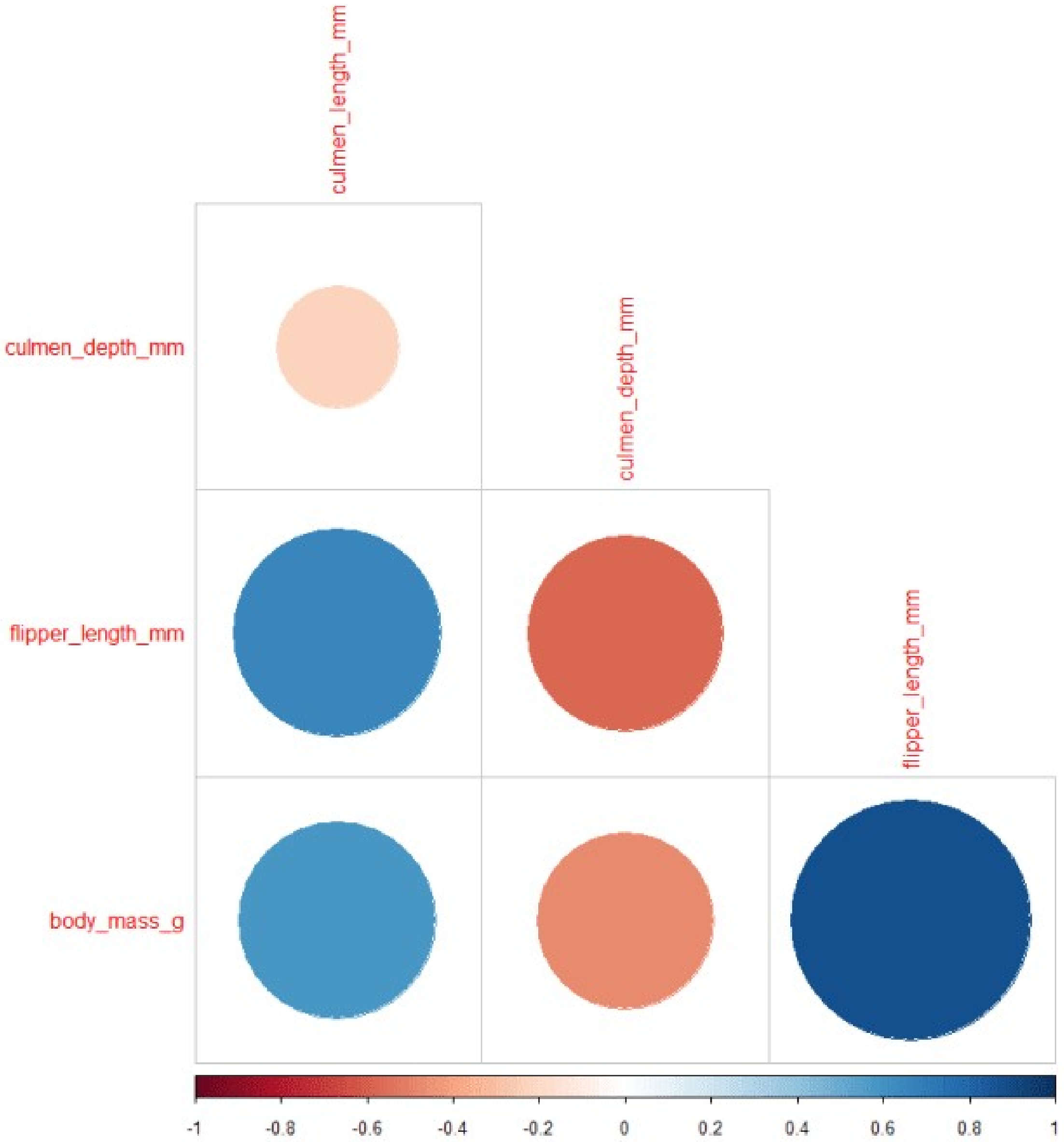
```
# species vs sex
table(penguins$sex, penguins$species)
chisq.test(table(penguins$sex, penguins$species)[-1,]) # not meaningful
difference 0.916

# sex vs island
table(penguins$sex, penguins$island) # 0.9716
chisq.test(table(penguins$sex, penguins$island)[-1,]) # not meaningful
difference 0.9716
```

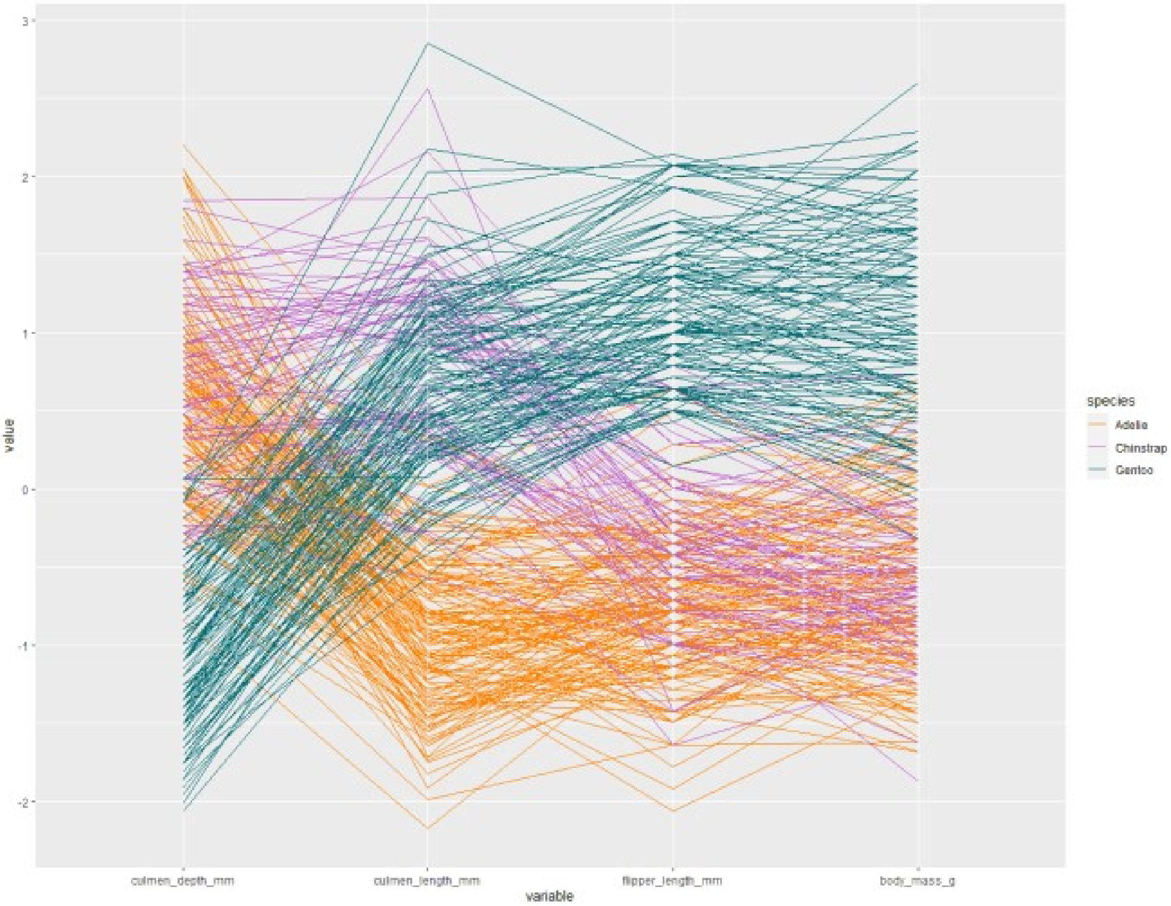
5. See with numerical features

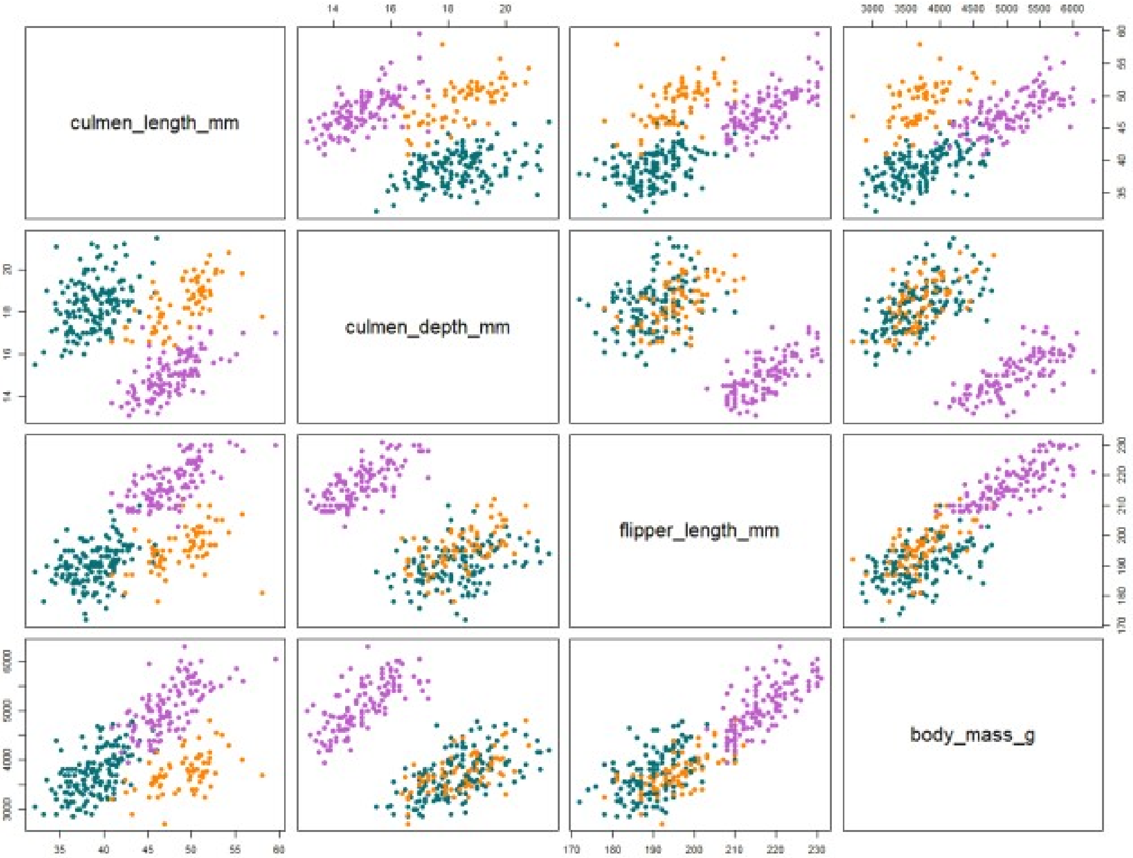
I will select numerical features.
and see correlation plot and parallel coordinate plots.

```
# Select numericals
penNumeric <- penguins %>% select(-species, -island, -sex)

# Cor-relation between numerics

corrplot(cor(penNumeric), type = 'lower', diag = FALSE)

# parallel coordinate plots

ggparcoord(penguins, columns = 3:6, groupColumn = 1, order = c(4,3,5,6)) +
  scale_color_manual(values = pCol)

plot(penNumeric, col = pCol[penguins$species], pch = 19)
```

and below are result of them.


culmen_length_mm
culmen_depth_mm
flipper_length_mm
culmen_depth_mm
flipper_length_mm
body_mass_g
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1





lucky, every numeric features (even only 4) have meaningful correlation and there is trend with their combination for `species` (See parallel coordinate plot)

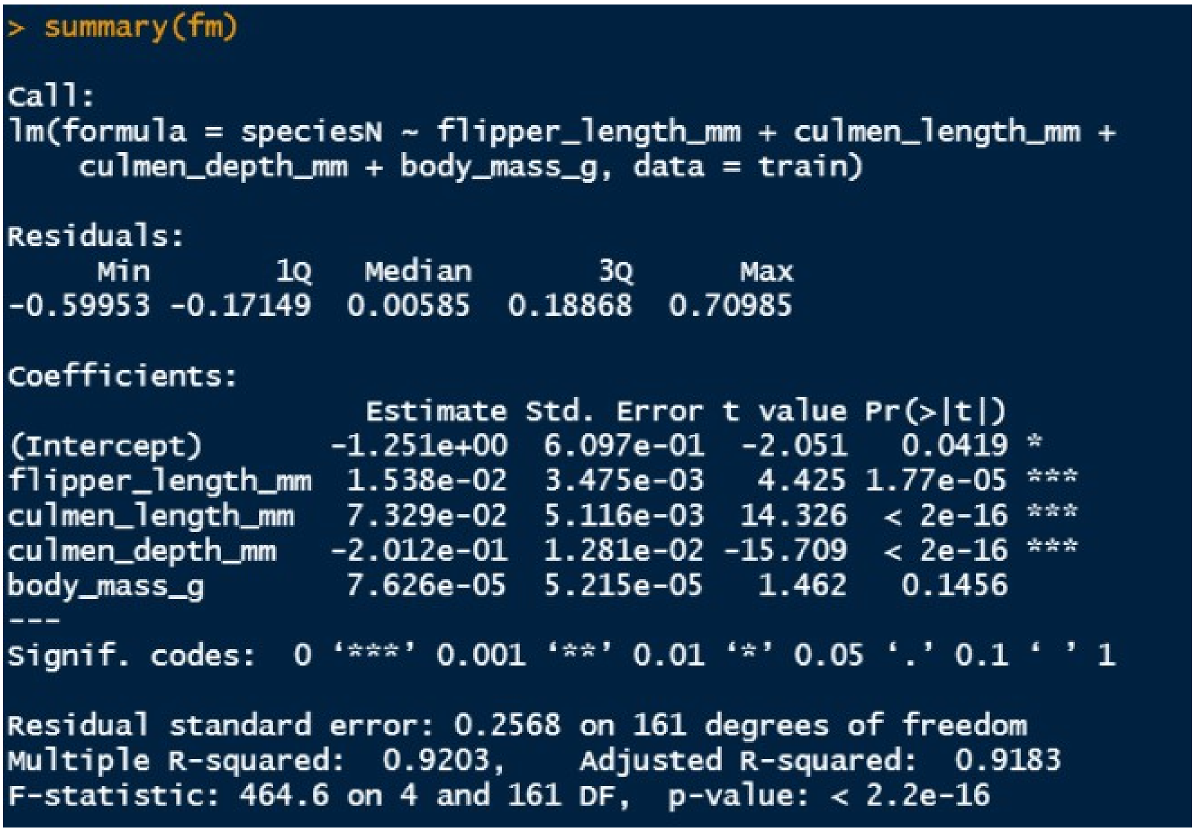
6. Give statistical work on dataset.

In this step, I usually do `linear modeling` or `svm` to **predict**

6.1 `linear modeling`

`species` is categorical value, so it needs to be change to **numeric value**

```
set.seed(1234)
idx <- sample(1:nrow(penguins), size = nrow(penguins)/2)

# as. numeric
speciesN <- as.numeric(penguins$species)
penguins$speciesN <- speciesN

train <- penguins[idx,]
test <- penguins[-idx,]


fm <- lm(speciesN ~ flipper_length_mm + culmen_length_mm + culmen_depth_mm +
body_mass_g, train)

summary(fm)
```

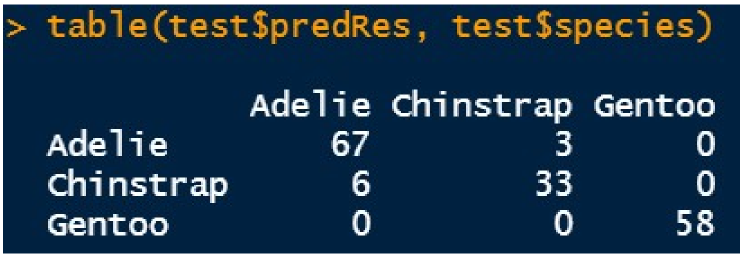


```
> summary(fm)

Call:
lm(formula = speciesN ~ flipper_length_mm + culmen_length_mm +
    culmen_depth_mm + body_mass_g, data = train)

Residuals:
     Min       1Q   Median       3Q      Max
-0.59953 -0.17149  0.00585  0.18868  0.70985

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       -1.251e+00  6.097e-01  -2.051   0.0419 *
flipper_length_mm  1.538e-02  3.475e-03   4.425 1.77e-05 ***
culmen_length_mm   7.329e-02  5.116e-03  14.326  < 2e-16 ***
culmen_depth_mm   -2.012e-01  1.281e-02 -15.709  < 2e-16 ***
body_mass_g        7.626e-05  5.215e-05   1.462   0.1456
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2568 on 161 degrees of freedom
Multiple R-squared:  0.9203,    Adjusted R-squared:  0.9183
F-statistic: 464.6 on 4 and 161 DF,  p-value: < 2.2e-16
```

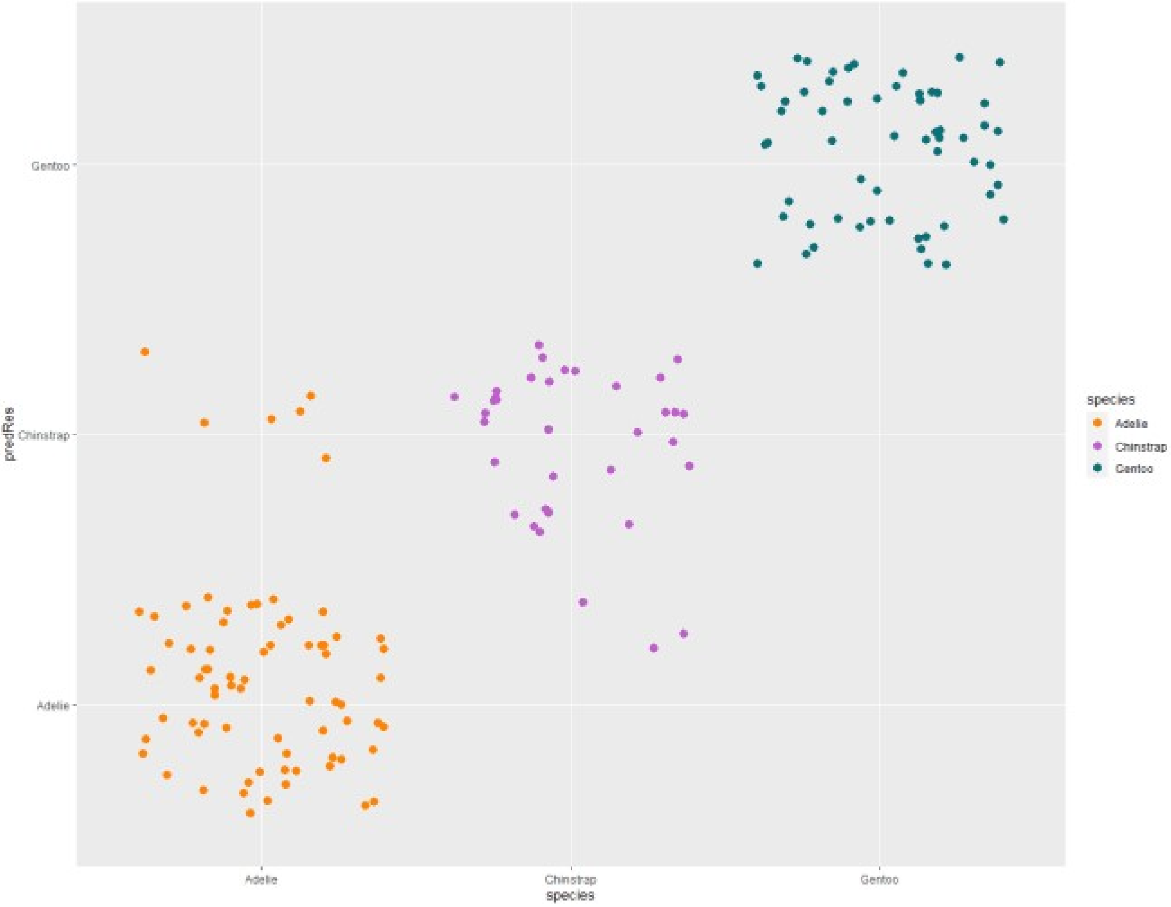
It shows that, `body_mass_g` is not meaningful feature as seen in `plot` above (it may explain `gentoo`, but not other penguins)

To predict, I used this code. however, numeric predict generate **not complete value** (like 2.123 instead of 2) so I added rounding step.

```
predRes <- round(predict(fm, test))
predRes[which(predRes>3)] <- 3
predRes <- sort(names(pCol))[predRes]

test$predRes <- predRes
ggplot(test, aes(x = species, y = predRes, color = species))+
  geom_jitter(size = 3) +
  scale_color_manual(values = pCol)

table(test$predRes, test$species)
```



```
> table(test$predRes, test$species)

            Adelie Chinstrap Gentoo
  Adelie        67         3      0
  Chinstrap      6        33      0
  Gentoo         0         0     58
```



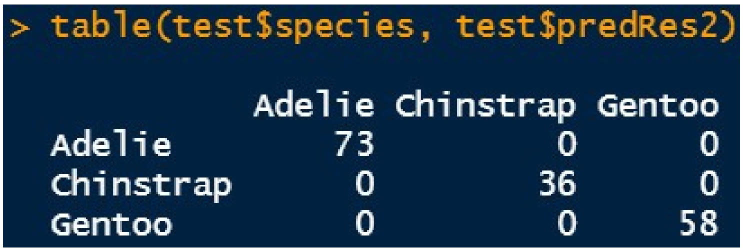
Accuracy of basic `linear modeling` is 94.6%

6-2 `svm`

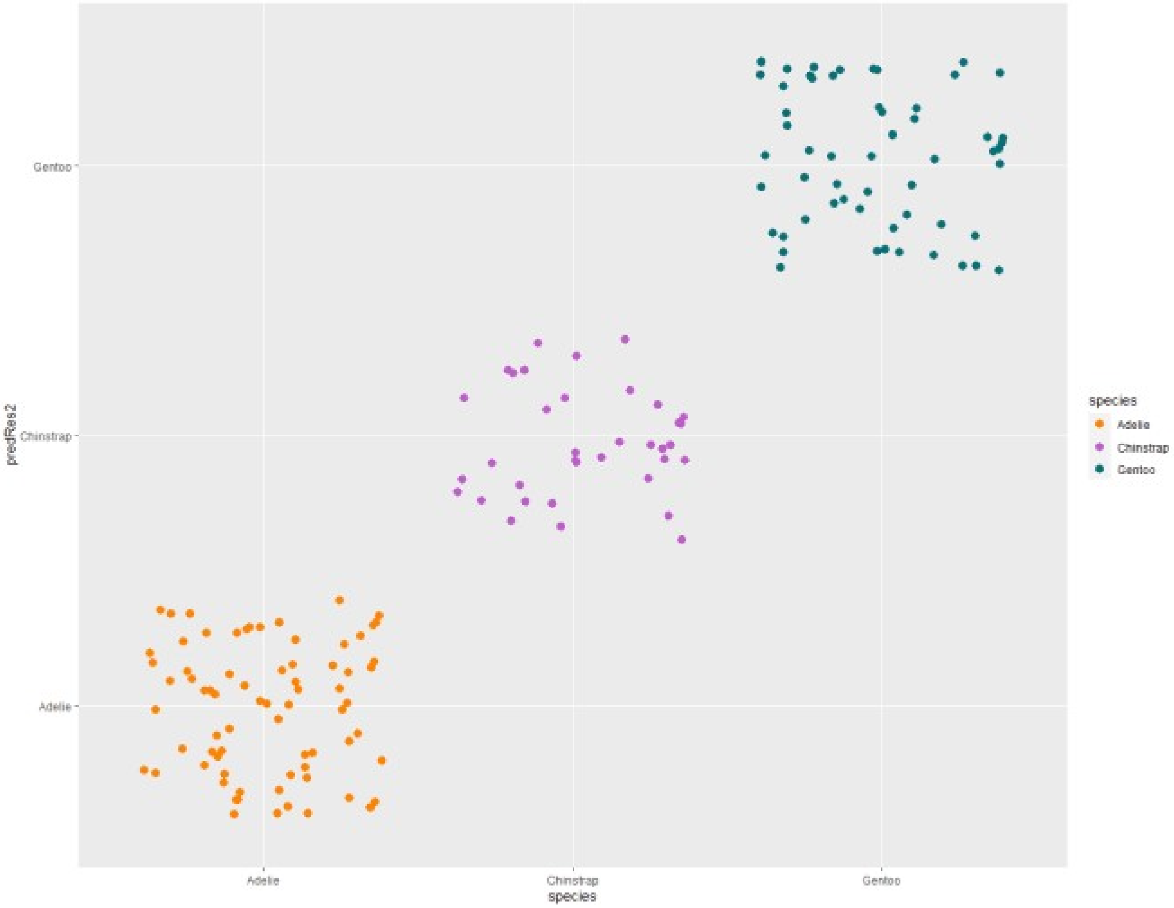
using `svm` is also easy step.

```
m <- svm(species ~., train)

predRes2 <- predict(m, test)
test$predRes2 <- predRes2

ggplot(test, aes(x = species, y = predRes2, color = species)) +
  geom_jitter(size = 3) +
  scale_color_manual(values = pCol)

table(test$species, test$predRes2)
```

and below are result of this code.



```
> table(test$species, test$predRes2)

            Adelie Chinstrap Gentoo
  Adelie        73         0      0
  Chinstrap      0        36      0
  Gentoo         0         0     58
```



Accuracy of `svm` is 100%. wow.