Special_Task_Lab2

Andreas Stasinakis

December 7, 2018

Special task 3

3.1

Analysis

We implement LDA from scratch in order to plot the decision boundary. We start with the equation

$$x^T(\beta_1 - \beta_2) = \gamma_1 - \gamma_2$$

We do some calculations in order to find the equation which we use to fit the line and create the boundary. So the final equation, **equation of decision boundary** is :

$$x_2 = \frac{\gamma_2 - \gamma_1 - x_1(\beta_{11} - \beta_{21})}{\beta_{12} - \beta_{22}}$$

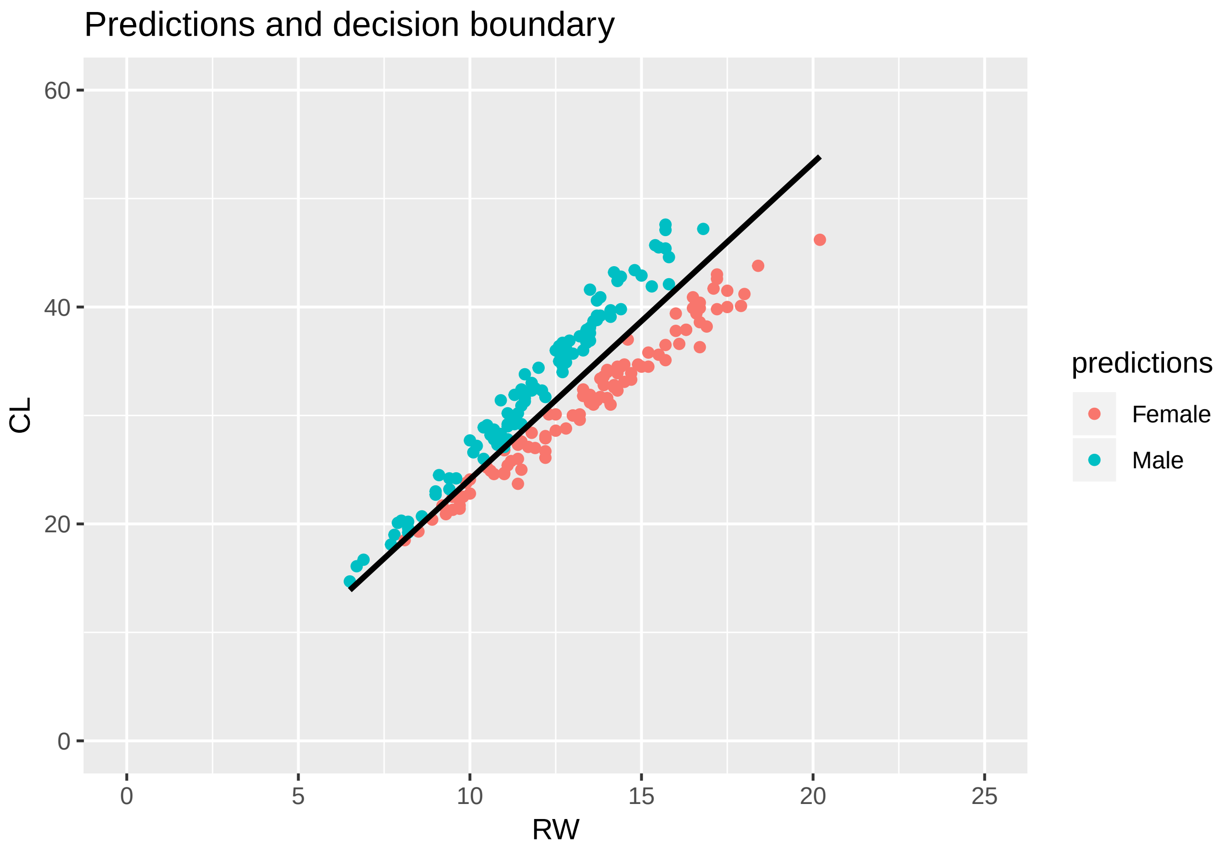
where $\gamma_c = -1/2\mu_c^T\Sigma^{-1}\mu_c + log\pi_c$, $\beta_c = \Sigma^{-1}\mu_c$ and $c =$ [Male,Female].

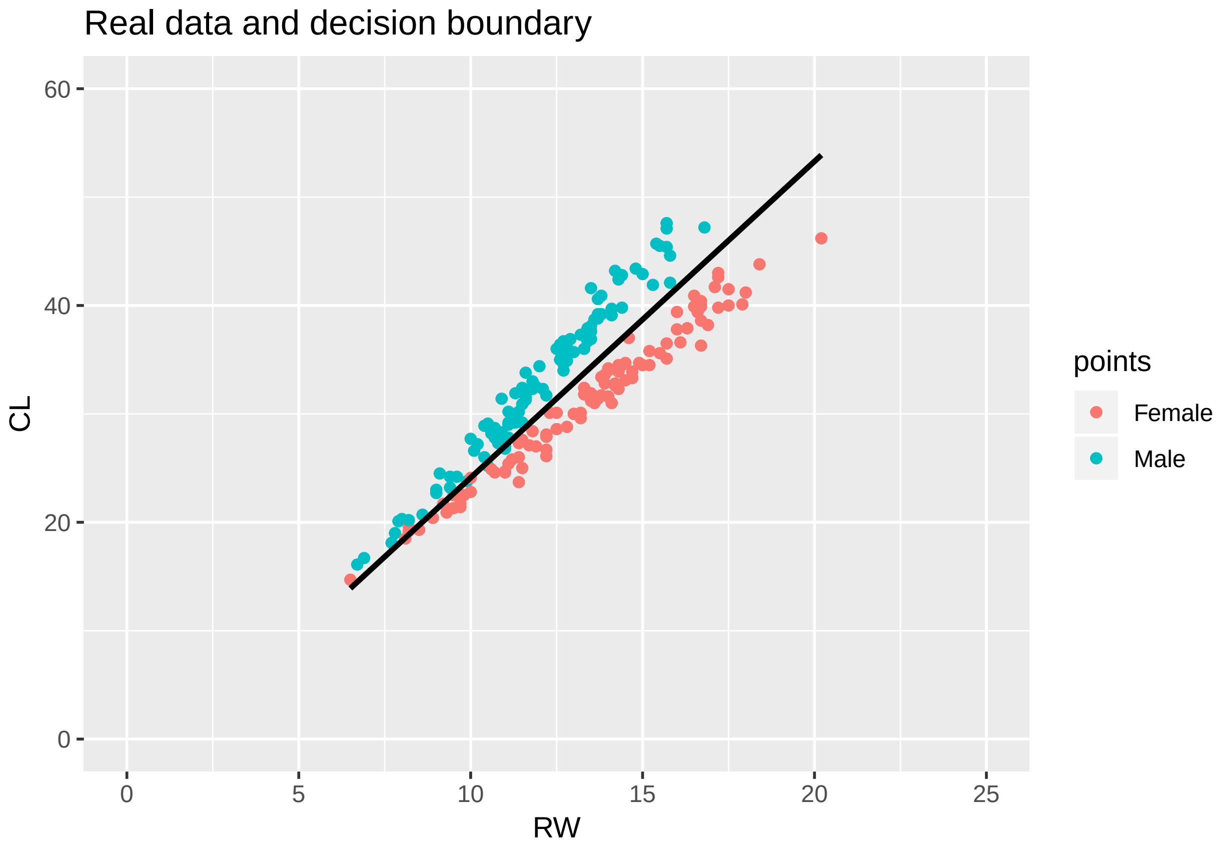
After that we use the **linear discriminant functions** below to classify the data.

$$\delta_c(x) = x^T\Sigma^{-1}\mu_c - 1/2\mu_c^T\Sigma^{-1}\mu_c + log\pi_c$$

We calculate the $\delta_c(x)$ for the female and the male and we classify each observation as the label with the higher value. Finally the predictions can be found above.

Task 3.2


Predictions and decision boundary
60
40
20
0
CL
0
5
10
15
20
25
RW
predictions
● Female
● Male


Real data and decision boundary
60
40
20
0
CL
0
5
10
15
20
25
RW
points
● Female
● Male

Analysis

If we compare the two plots, we can see that the line fit really good the data. Except for a few misclassified points (only 7 out of 200), the rest classified correctly through the line. Also the mislassification rate for the fit is only 0.035. As a result, we can conclude that the line fit well the data.

Appendix

```
knitr::opts_chunk$set(echo = FALSE,
                      warning = F, error = F, message = F)
library(readxl)
library(MASS)
library(ggplot2)

all_data = read.csv("../dataset/australian-crabs.csv")

data = all_data[,c("RW","CL","sex")]

male_df = data[which(data$sex == "Male"),]
female_df = data[which(data$sex == "Female"),]
pi_male = nrow(male_df)/ nrow(data)
pi_female = nrow(female_df)/ nrow(data)

# calculate the means

mean_matrix = matrix(c(mean(male_df$RW),mean(female_df$RW),mean(male_df$CL),mean(female_df$CL)), nrow =
rownames(mean_matrix) = c("Male","Female")
colnames(mean_matrix) = c("RW","CL")

#covariance matrix for male
covar_male = cov(male_df[c(1,2)])

#Covariance matrix for female
covar_female = cov(female_df[c(1,2)])

# pooled covariance matrix
cov = (covar_male + covar_female)/2

# coefficients
coef_male = solve(cov)%*%mean_matrix[1,]
coef_female = solve(cov)%*%mean_matrix[2,]
coef = rbind(t(coef_male),t(coef_female))
rownames(coef) = c("male","female")

#discrimination function
predictions = c()
z = c()

# loop for all observations
for (i in 1:nrow(data)) {

  #take each different observation
  x = as.matrix(data[i,c(1,2)])


  g_male = log(pi_male) -1/2*t(as.matrix(mean_matrix[1,]))%*%solve(cov)%*%as.matrix(mean_matrix[1,])

  g_female = log(pi_female) -1/2*t(as.matrix(mean_matrix[2,]))%*%solve(cov)%*%as.matrix(mean_matrix[2,]
```

```r
  #  equation of the decision boundary

  equation = (g_female - g_male - x[1]*(coef[1,1] - coef[2,1]))/ (coef[1,2] - coef[2,2])

  #  map Cl to RW coordinates
  z = c(z,equation)

  #Linear discriminant functions

  delta_male = g_male + x %*% solve(cov) %*% as.matrix(mean_matrix[1,])

  delta_female = g_female + x %*% solve(cov) %*% as.matrix(mean_matrix[2,])

  #Classify  the observations

  if(delta_female >= delta_male){

    predictions = c(predictions,"Female")
  } else if(delta_male >delta_female) {

    predictions = c(predictions,"Male")
  }
}
#data frame with the predictions and the real data
a = data.frame("Predictions" = predictions, "Real values" = data$sex)

#data frame for the plot

df1 =  data.frame(data$RW, data$CL , points = data$sex,z)
df2 =  data.frame(data$RW, data$CL , predictions ,z)

#Plot for the predictions
ggplot(df2) + geom_point(mapping = aes_string(x = df2$data.RW,y= df2$data.CL, colour = "predictions"))+
  scale_y_continuous(limits = c(0,60)) + labs(title = "Predictions and decision boundary" ,x = "RW" , y

#Plot for the real data
ggplot(df1) + geom_point(mapping = aes_string(x = df1$data.RW,y= df1$data.CL, colour = "points"))+ geom
  scale_y_continuous(limits = c(0,60)) + labs(title = "Real data and decision boundary" ,x = "RW" , y =
```