Lab2

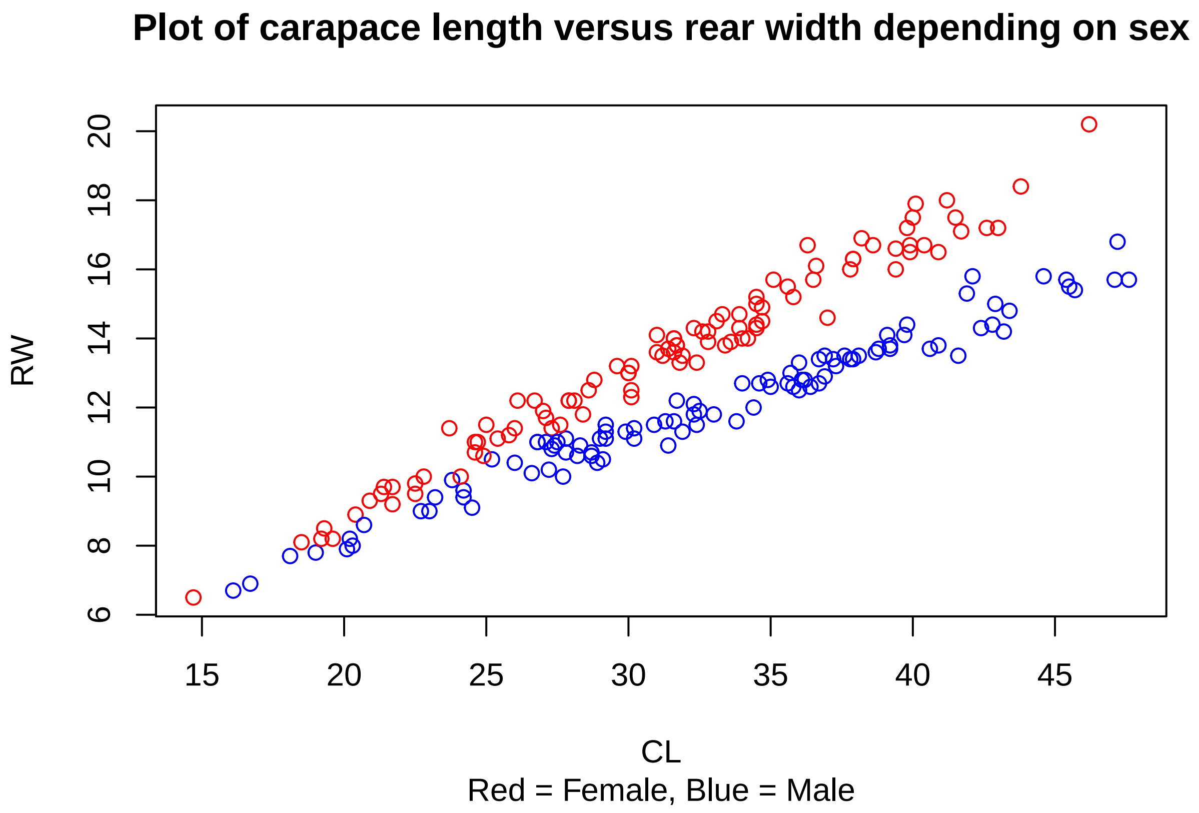
Christian von Koch

2019-12-07

Assignment 1

Task 1

```
Dataframe=read.csv("australian-crabs.csv")
n = length(Dataframe[,1])
CL = Dataframe$CL
RW = Dataframe$RW
plot(CL, RW, main="Plot of carapace length versus rear width depending on sex",
     sub="Red = Female, Blue = Male",
     col=c("red", "blue")[Dataframe$sex], xlab="CL", ylab="RW")
```



As we can see in the graph it looks like it would be suitable to classify this data using linear discriminative analysis since the pattern of both the red and blue is linear.

Task 2

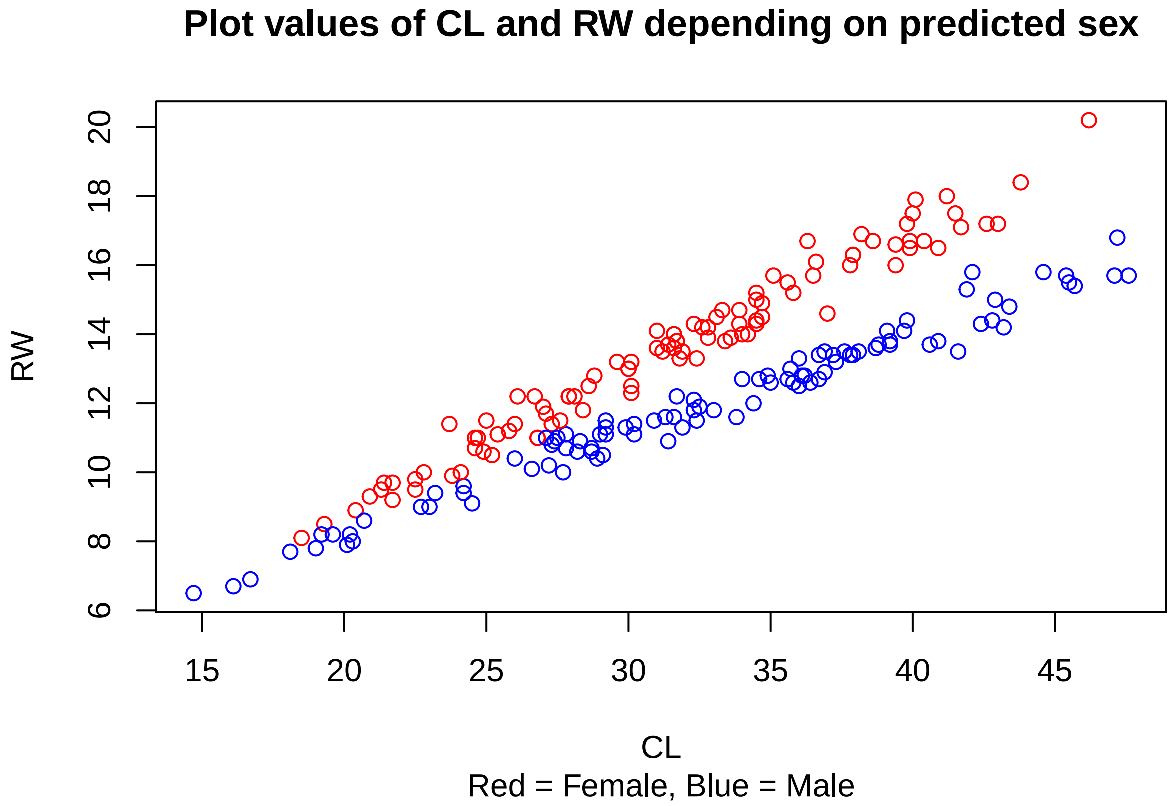
```
#Create function for misclassification rate
missclass=function(conf_matrix, fit_matrix){
  n=length(fit_matrix[,1])
  return(1-sum(diag(conf_matrix))/n)
}

#LDA analysis with target Sex, and features CL and RW and proportional prior
library("MASS")
model = lda(sex ~ CL+RW, data=Dataframe)
predicted = predict(model, data=Dataframe)
confusion_matrix = table(Dataframe$sex, predicted$class)
misclass = missclass(confusion_matrix, Dataframe)
print(confusion_matrix)
```

```
##
##          Female Male
##   Female     97    3
##   Male        4   96
```

```
print(misclass)
```

```
## [1] 0.035
```

```
plot(CL, RW, main="Plot values of CL and RW depending on predicted sex",
     sub="Red = Female, Blue = Male",
     col=c("red", "blue")[predicted$class], xlab="CL", ylab="RW")
```



When comparing the graph from step 1 and the graph of the predicted values it is notable that the classifications do not differ that much. With a misclassification rate of only 0.035 and 200 datapoints it can be concluded that 7 observations were classified inaccurately. When comparing the graphs it is difficult to find the points which have changed color (since they have been classified incorrectly) but one example is the point farthest to the left which was classified as *male* but should have been classified as *female*. The model classifies the data very accurately.

Task 3

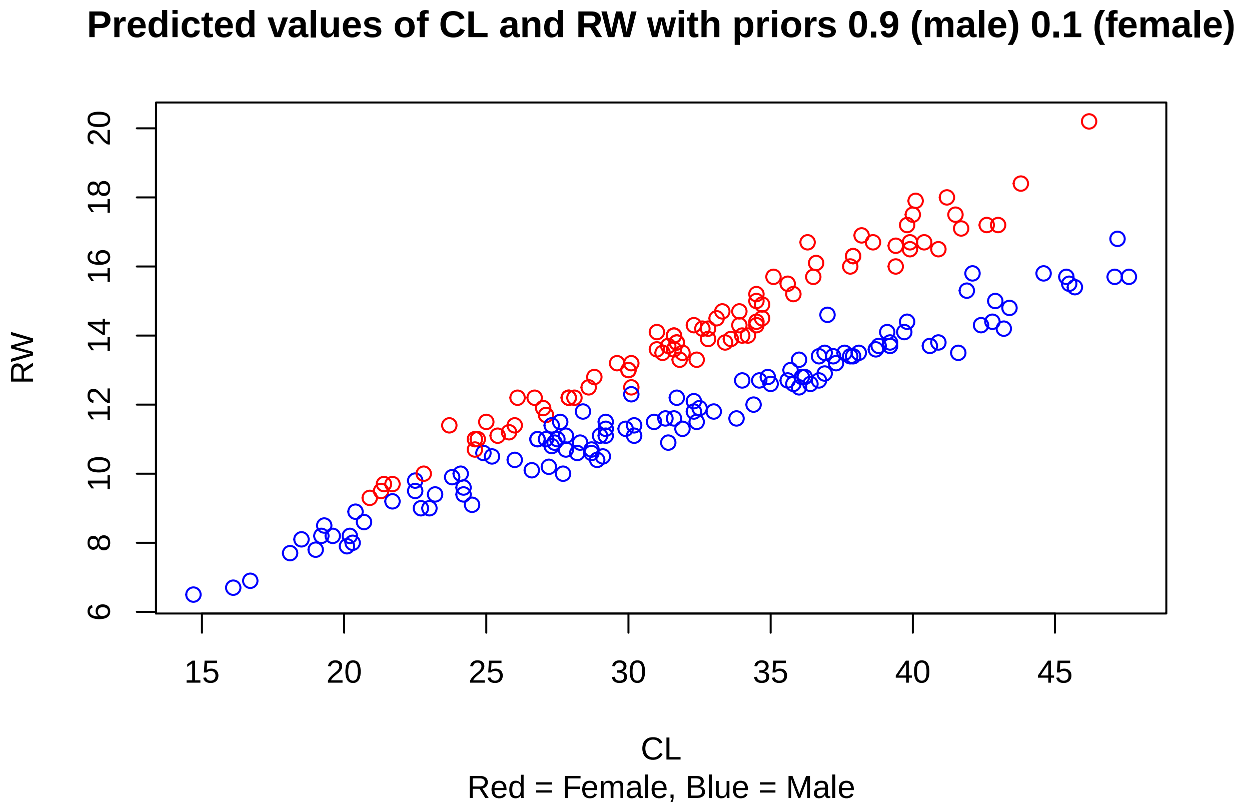
```
#Repeat step 2 but use priors p(Male)=0.9 and p(Female)=0.1
model2 = lda(sex ~ CL+RW, data=Dataframe, prior=c(1,9)/10)
predicted2 = predict(model2, data=Dataframe)
confusion_matrix2 = table(Dataframe$sex, predicted2$class)
misclass2 = missclass(confusion_matrix2, Dataframe)
print(confusion_matrix2)
```

```
##
##          Female Male
##   Female     84   16
##   Male        0  100
```

```
print(misclass2)
```

```
## [1] 0.08
```

```
plot(CL, RW, main="Predicted values of CL and RW with priors 0.9 (male) 0.1 (female)"
     , sub="Red = Female, Blue = Male", col=c("red", "blue")[predicted2$class], xlab="CL",
     ylab="RW")
```



From this graph we can see that a few more data points were classified incorrectly. This is due to the higher prior set on classifying a data point as male, i.e. 0.9. It is noteable in the confusion matrix that no females were classified incorrectly. This is also due to the low prior which basically says that it is not that likely that a datapoint will be classified as a female. When the model in fact classify a data point as female it has to be sure of it (since the low prior) and this can be seen as stated above in the confusion matrix. On the other hand, more males are classified inaccurately since the higher prior. This also results in a higher misclassification rate of 0.08.

Task 4

```
#Repeat step 2 but now with logistic regression
model3 = glm(sex ~ CL+RW, data=Dataframe, family='binomial')
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

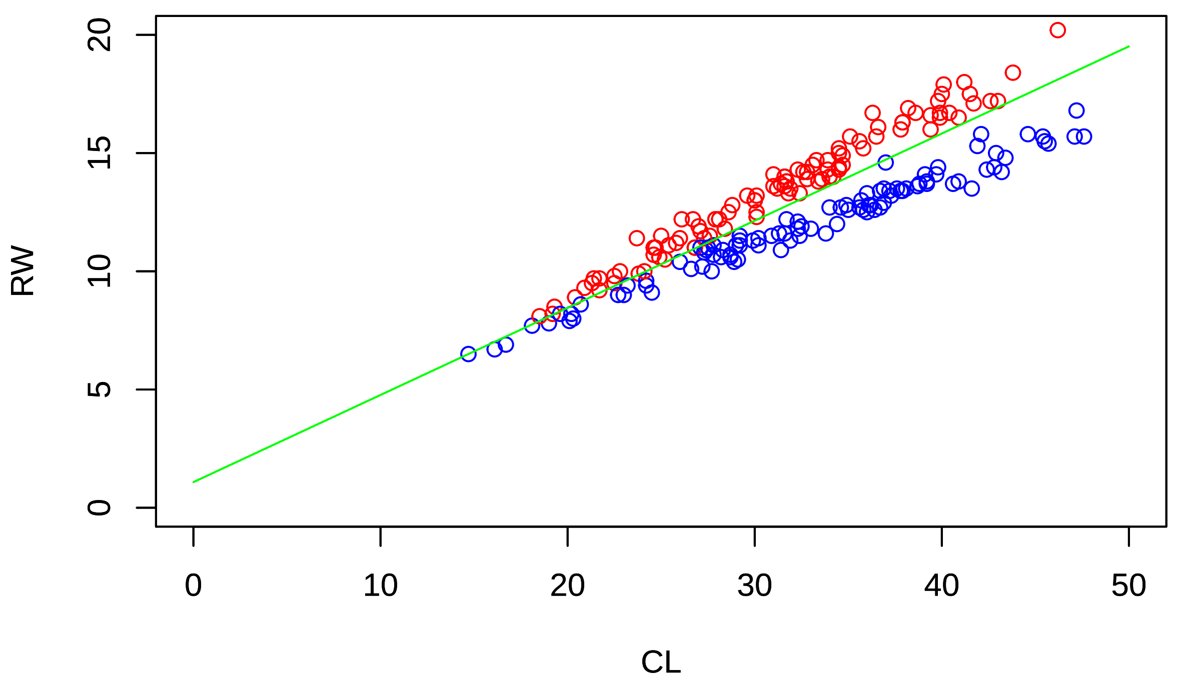
```
predicted3 = predict(model3, newdata=Dataframe, type='response')
sexvector = c()
for (i in predicted3) {
  if (i>0.5) {
    sexvector = c(sexvector, 'Male')
  } else {
    sexvector = c(sexvector, 'Female')
  }
}
sexvector_factor = as.factor(sexvector)
confusion_matrix3 = table(Dataframe$sex, sexvector_factor)
misclass3 = missclass(confusion_matrix3, Dataframe)
print(confusion_matrix3)
```

```
##         sexvector_factor
##          Female Male
##   Female     97    3
##   Male        4   96
```

```
print(misclass3)
```

```
## [1] 0.035
```

```
plot(CL, RW, main="Predicted values of CL and RW but with logistic regression",
     col=c("red", "blue")[sexvector_factor], xlab="CL", ylab="RW", xlim=c(0,50), ylim=c(0,20))

boundaryline = function(length, coefficientvector, prior) {
  return(-coefficientvector[1]/coefficientvector[3]-
           (coefficientvector[2]/coefficientvector[3])*length+
           log(prior/(1-prior))/coefficientvector[3])
}
par(new=TRUE)
curve(boundaryline(x, model3$coefficients, 0.5), xlab="CL", ylab="RW", col="green",
      from=0, to=50, xlim=c(0,50), ylim=c(0,20),
      sub="Red = Female, Blue = Male, Green = Boundaryline")
```



Predicted values of CL and RW but with logistic regression

Red = Female, Blue = Male, Green = Boundaryline

When using logistic regression the results are similar as the first built model with LDA. This is simply a coincident and no real conclusion can be drawn regarding the exact same misclassification rate except from that the models seem to classify the data in the same way.The equation for the decision boundary is as follows:

{.tabset}

$R\hat{W}$=-(β_0+β_1)/β_2*CL

Assignment 2

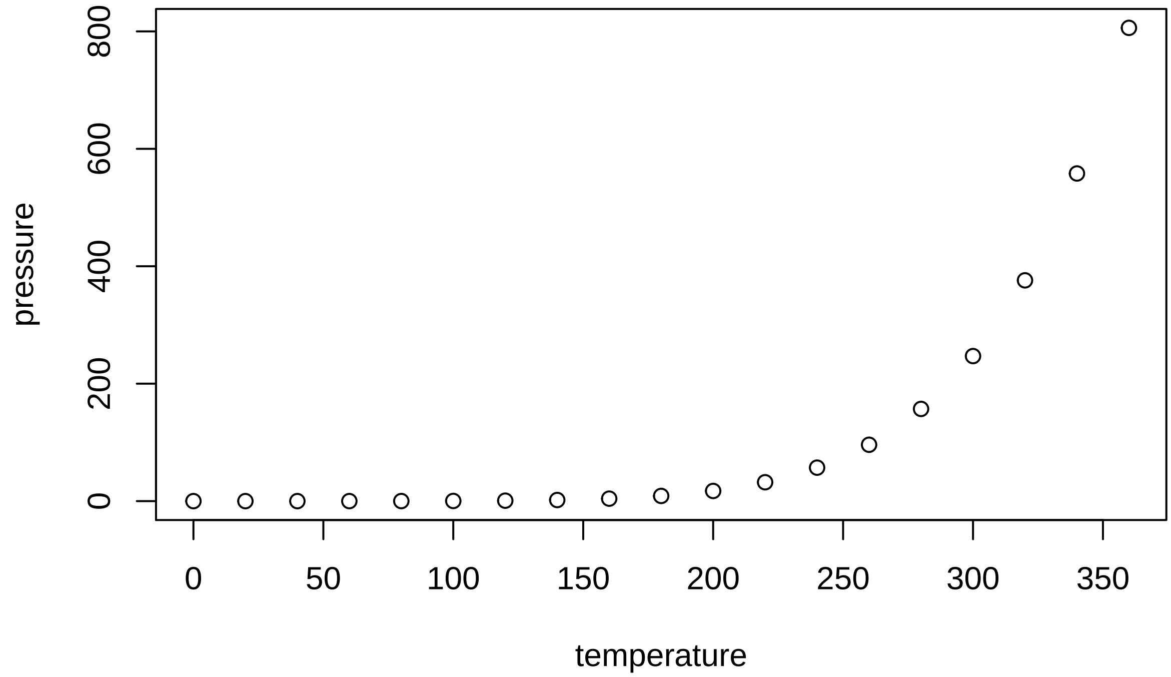
Task 1

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.