Iris Analysis

Alejandro Fernandez Benages

17/03/2021

**libraries**

#install.packages('tidyverse')  
# install.packages(readr)  
library("readr")  
#library(tidyverse)  
library(ggplot2)

**Load CSV**

setwd('/home/ale/Dropbox/UBIQUM/3. DA with R/1.GetStarted/Iris')  
IrisDataset <- read.csv('iris.csv')

**Some info about the Data**

attributes(IrisDataset)

## $names  
## [1] "X" "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"   
## [6] "Species"   
##   
## $class  
## [1] "data.frame"  
##   
## $row.names  
## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18  
## [19] 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36  
## [37] 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54  
## [55] 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72  
## [73] 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90  
## [91] 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108  
## [109] 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126  
## [127] 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144  
## [145] 145 146 147 148 149 150

summary(IrisDataset)

## X Sepal.Length Sepal.Width Petal.Length   
## Min. : 1.00 Min. :4.300 Min. :2.000 Min. :1.000   
## 1st Qu.: 38.25 1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600   
## Median : 75.50 Median :5.800 Median :3.000 Median :4.350   
## Mean : 75.50 Mean :5.843 Mean :3.057 Mean :3.758   
## 3rd Qu.:112.75 3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100   
## Max. :150.00 Max. :7.900 Max. :4.400 Max. :6.900   
## Petal.Width Species   
## Min. :0.100 setosa :50   
## 1st Qu.:0.300 versicolor:50   
## Median :1.300 virginica :50   
## Mean :1.199   
## 3rd Qu.:1.800   
## Max. :2.500

str(IrisDataset)

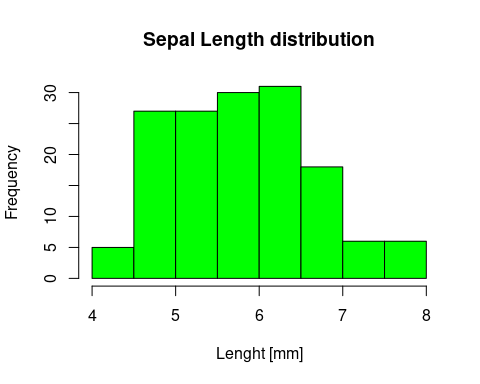
## 'data.frame': 150 obs. of 6 variables:  
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...  
## $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...  
## $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...  
## $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...  
## $ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

names(IrisDataset)

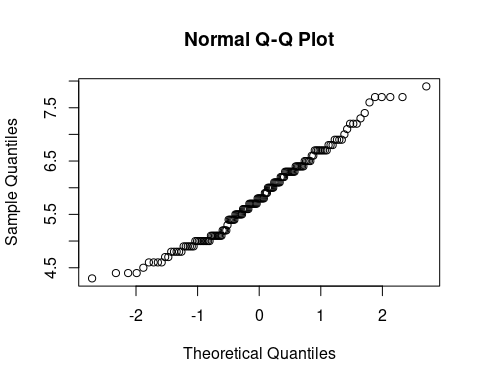
## [1] "X" "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"   
## [6] "Species"

**Some Plots**

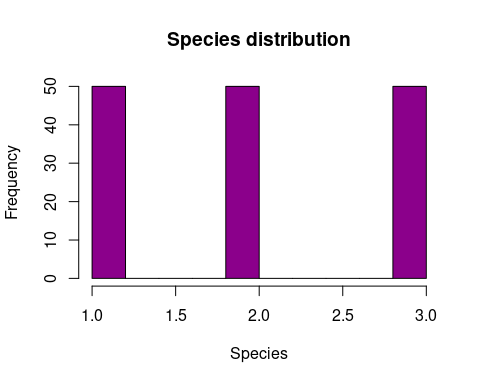
hist(IrisDataset$Sepal.Length,   
 main="Sepal Length distribution", xlab="Lenght [mm]",   
 col='green')



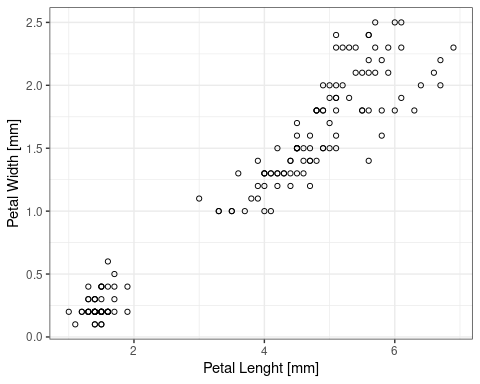
qqnorm(IrisDataset$Sepal.Length)



IrisDataset$Species<- as.numeric(IrisDataset$Species)   
hist(IrisDataset$Species,  
 main="Species distribution", xlab="Species",  
 col='darkmagenta')



ggplot(data = IrisDataset, aes(x = Petal.Length , y = Petal.Width)) +   
 geom\_point(shape = 1) +   
 xlab("Petal Lenght [mm]")+ylab("Petal Width [mm]") +  
 theme\_bw()



**Train/Test Split**

set.seed(123)

set dimension of Train/Test vectors

trainSize <- round(nrow(IrisDataset) \* 0.2)  
testSize <- nrow(IrisDataset) - trainSize

just to check, print the dimensions

trainSize

## [1] 30

testSize

## [1] 120

assign the values to empty vectors

training\_indices<-sample(seq\_len(nrow(IrisDataset)),size =trainSize)   
trainSet <- IrisDataset[training\_indices, ]  
testSet <- IrisDataset[-training\_indices, ]

**Linear Model**

The analysis goal is to predict a petal’s length using the petal’s width.  
Inp Var -> petal’s width Out Var -> petal’s length

LinearModel<- lm(Petal.Length ~ Petal.Width, trainSet )  
summary(LinearModel)

##   
## Call:  
## lm(formula = Petal.Length ~ Petal.Width, data = trainSet)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.06290 -0.31298 -0.00479 0.27231 0.93694   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.8632 0.1484 5.817 3e-06 \*\*\*  
## Petal.Width 2.4165 0.1034 23.381 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4316 on 28 degrees of freedom  
## Multiple R-squared: 0.9513, Adjusted R-squared: 0.9495   
## F-statistic: 546.7 on 1 and 28 DF, p-value: < 2.2e-16

prediction<-predict(LinearModel)  
   
prediction

## 14 50 118 43 150 148 90 91   
## 1.104873 1.346527 6.179589 1.346527 5.212977 5.696283 4.004711 3.763058   
## 143 92 137 99 72 26 7 78   
## 5.454630 4.246364 6.662895 3.521405 4.004711 1.346527 1.588180 4.971323   
## 81 147 103 117 76 32 106 109   
## 3.521405 5.454630 5.937936 5.212977 4.246364 1.829833 5.937936 5.212977   
## 136 9 41 74 23 27   
## 6.421242 1.346527 1.588180 3.763058 1.346527 1.829833