M3T1_2FixedScript.R

christiancobollogomez

2021-11-09

```
# install.packages("readr")
# install.packages("ggplot2")

library("readr")
library("ggplot2")

IrisDataset <- read.csv("iris.csv")

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   Length:150
##  1st Qu.:0.300   Class :character
##  Median :1.300   Mode  :character
##  Mean   :1.199
##  3rd Qu.:1.800
```

```
##  Max.   :2.500
```

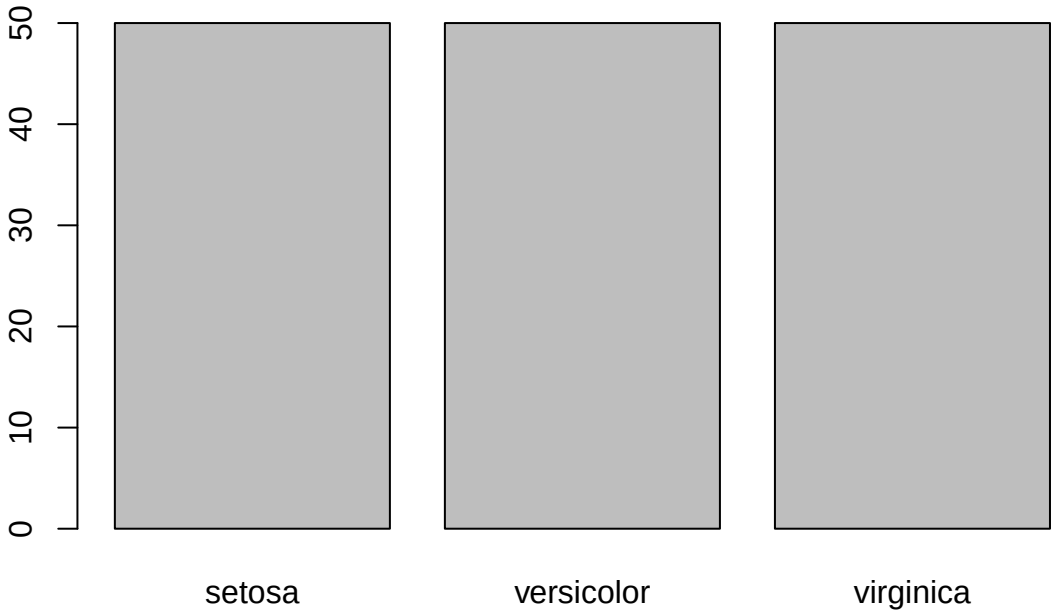
```
str(IrisDataset)
```

```
## '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     : chr  "setosa" "setosa" "setosa" "setosa" ...
```

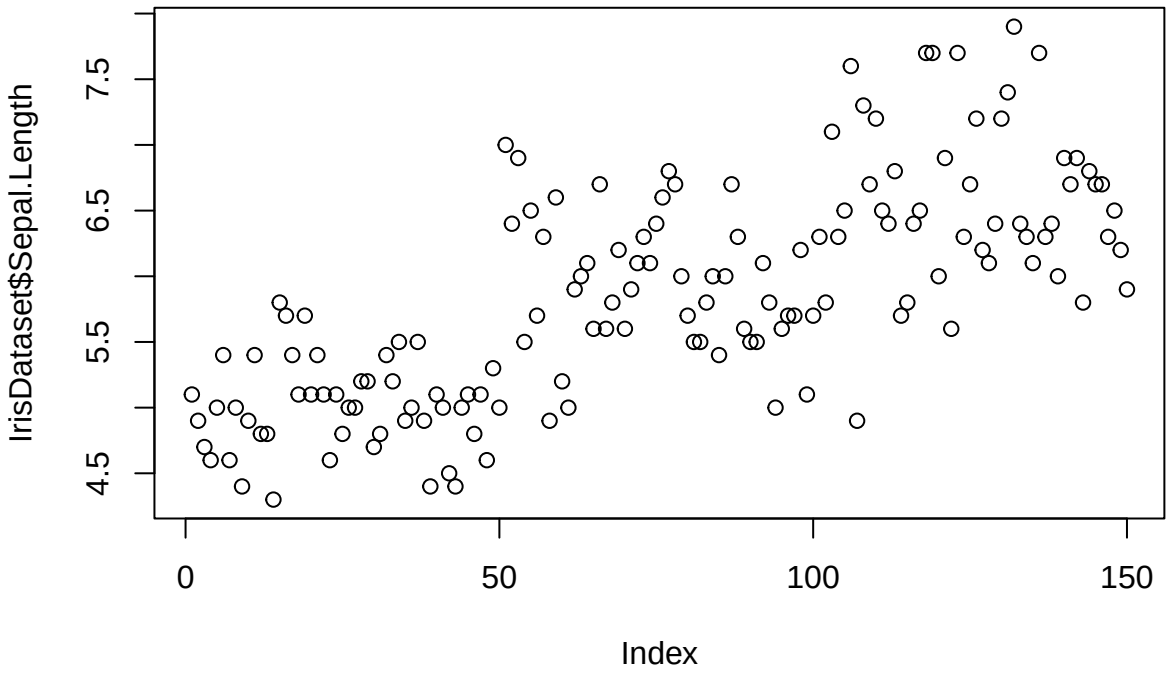
```
names(IrisDataset)
```

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

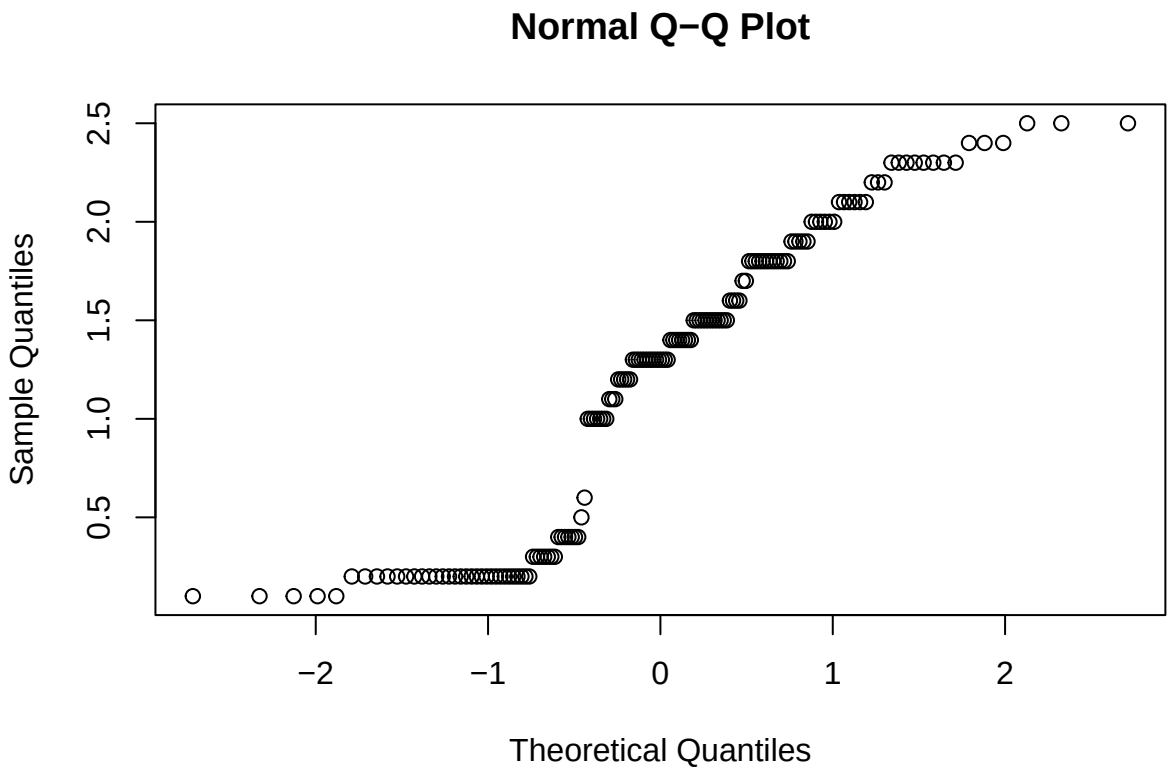
```
barplot(table(IrisDataset$Species)) # We cannot plot or make a histogram about
```



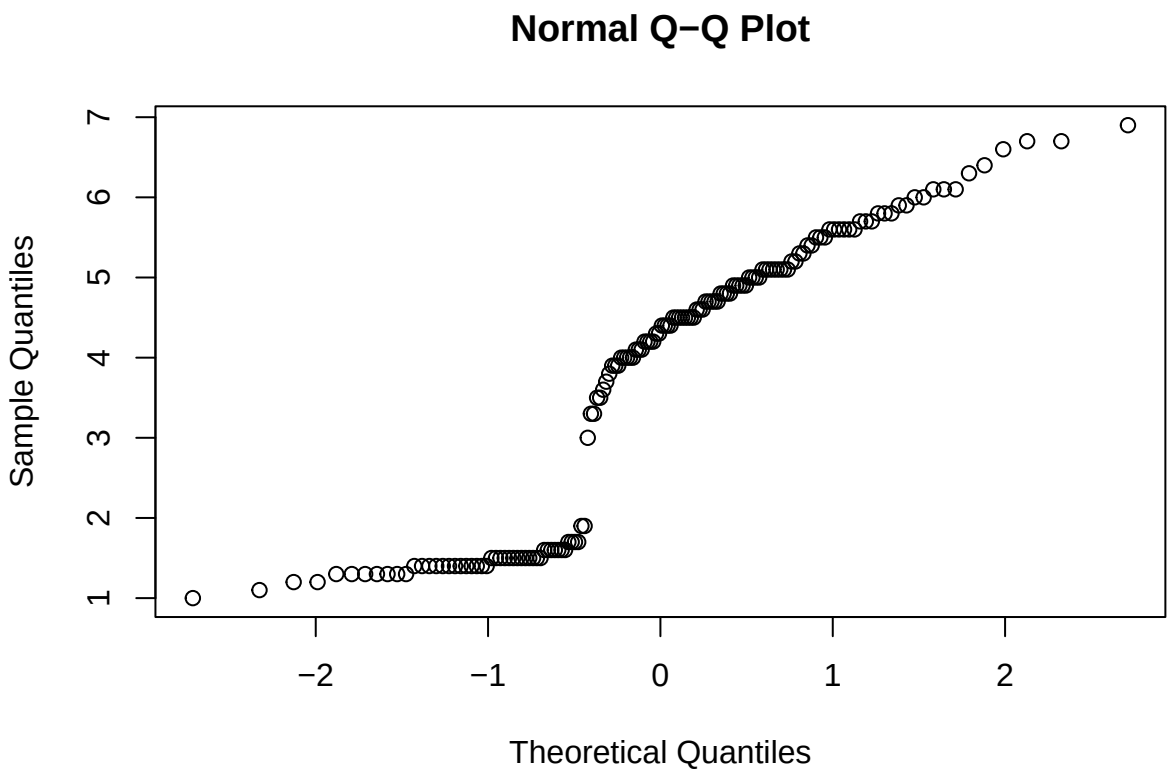

```
# the Species column. I find the the natural representation should be a count
# on the number of each species.

plot(IrisDataset$Sepal.Length)
```




```
# there is no special need to change the species into numeric values for this task.
# But we can do it with:

IrisDataset$Species<- as.numeric(as.factor(IrisDataset$Species))

# Represent the normality of the feature
qqnorm(IrisDataset$Petal.Width)
```



```r
# Represent the normality of the target
qqnorm(IrisDataset$Petal.Length)
```




```r
trainSize <- round(nrow(IrisDataset) * 0.2)

testSize <- nrow(IrisDataset) - trainSize

trainSize
```

```
## [1] 30
```

```r
testSize
```

```
## [1] 120
```

```r
set.seed(123)

training_indices<-sample(seq_len(nrow(IrisDataset)),size =trainSize)

trainSet <- IrisDataset[training_indices,]

testSet <- IrisDataset[-training_indices,]

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

summary(LinearModel) # R2 score = 0.95, p-value < 2.2e-16. Good enough prediction.
```

```
##
## 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,testSet)

prediction
```

```
##        1        2        3        4        5        6        8       10
## 1.346527 1.346527 1.346527 1.346527 1.346527 1.829833 1.346527 1.104873
##       11       12       13       15       16       17       18       19
## 1.346527 1.346527 1.104873 1.346527 1.829833 1.829833 1.588180 1.588180
##       20       21       22       24       25       28       29       30
## 1.588180 1.346527 1.829833 2.071486 1.346527 1.346527 1.346527 1.346527
##       31       33       34       35       36       37       38       39
## 1.346527 1.104873 1.346527 1.346527 1.346527 1.346527 1.104873 1.346527
##       40       42       44       45       46       47       48       49
## 1.346527 1.588180 2.313139 1.829833 1.588180 1.346527 1.346527 1.346527
##       51       52       53       54       55       56       57       58
## 4.246364 4.488017 4.488017 4.004711 4.488017 4.004711 4.729670 3.279752
##       59       60       61       62       63       64       65       66
## 4.004711 4.246364 3.279752 4.488017 3.279752 4.246364 4.004711 4.246364
##       67       68       69       70       71       73       75       77
## 4.488017 3.279752 4.488017 3.521405 5.212977 4.488017 4.004711 4.246364
##       79       80       82       83       84       85       86       87
## 4.488017 3.279752 3.279752 3.763058 4.729670 4.488017 4.729670 4.488017
##       88       89       93       94       95       96       97       98
## 4.004711 4.004711 3.763058 3.279752 4.004711 3.763058 4.004711 4.004711
##      100      101      102      104      105      107      108      110
## 4.004711 6.904548 5.454630 5.212977 6.179589 4.971323 5.212977 6.904548
##      111      112      113      114      115      116      119      120
## 5.696283 5.454630 5.937936 5.696283 6.662895 6.421242 6.421242 4.488017
##      121      122      123      124      125      126      127      128
## 6.421242 5.696283 5.696283 5.212977 5.937936 5.212977 5.212977 5.212977
##      129      130      131      132      133      134      135      138
## 5.937936 4.729670 5.454630 5.696283 6.179589 4.488017 4.246364 5.212977
##      139      140      141      142      144      145      146      149
## 5.212977 5.937936 6.662895 6.421242 6.421242 6.904548 6.421242 6.421242
```

```
## We represent the model vs the scatter plot of the data.
ggplot(data = IrisDataset, aes(x = Petal.Width, y = Petal.Length)) +
geom_point() +
stat_smooth(method = "lm", col = "dodgerblue3") +
theme(panel.background = element_rect(fill = "white"),
```

```
axis.line.x=element_line(),
axis.line.y=element_line()) +
ggtitle("Linear Model Fitted to Data")
```

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