THE SPARKS FOUNDATION

TASK 1 - Prediction using Supervised Machine Learning

Predict the percentage of a student based on the no. of study hours.

#### Done By: Gopika O

### **Importing the Packages**

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.0.3

library(corrplot)

## Warning: package 'corrplot' was built under R version 4.0.3

## corrplot 0.84 loaded

### **Importing the Dataset**

# loading the dataset from the url provided  
  
data = read.csv(url("http://bit.ly/w-data"))  
head(data)

## Hours Scores  
## 1 2.5 21  
## 2 5.1 47  
## 3 3.2 27  
## 4 8.5 75  
## 5 3.5 30  
## 6 1.5 20

#taking the dimension of the dataset  
  
dim(data)

## [1] 25 2

*The dataset contains 25 observations with 2 variables.*

#finding the Column names  
  
colnames(data)

## [1] "Hours" "Scores"

*Hours and Scores are the two variables present in the dataset.*

### **DATA PREPROCESSING**

#checking for NAN values  
  
colSums(is.na(data))

## Hours Scores   
## 0 0

*The given dataset contains no NaN values.*

#checking NULL values  
  
is.null(data)

## [1] FALSE

*The given dataset contain no NULL values.*

### **EXPLORATORY DATA ANALYSIS**

#structure of the dataset  
  
str(data)

## 'data.frame': 25 obs. of 2 variables:  
## $ Hours : num 2.5 5.1 3.2 8.5 3.5 1.5 9.2 5.5 8.3 2.7 ...  
## $ Scores: int 21 47 27 75 30 20 88 60 81 25 ...

*The variable Hours is of type Numeric and Scores is of datatype Integer.*

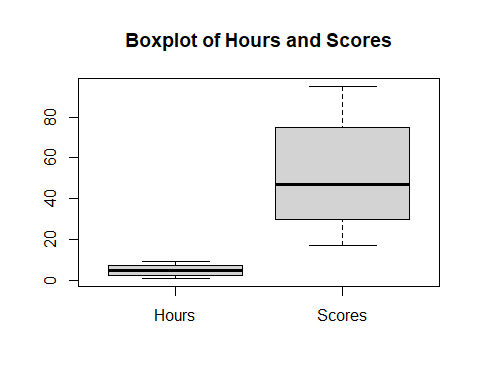
#SUMMARY OF THE DATASET  
  
summary(data)

## Hours Scores   
## Min. :1.100 Min. :17.00   
## 1st Qu.:2.700 1st Qu.:30.00   
## Median :4.800 Median :47.00   
## Mean :5.012 Mean :51.48   
## 3rd Qu.:7.400 3rd Qu.:75.00   
## Max. :9.200 Max. :95.00

* The Minimum value of Hours is 1.100 and maximum value is 9.200. Mean value is greater than median value. Hence it is right skewed.
* The Minimum value of Scores is 17.00 and maximum value is 95.00. Mean value is greater than median value. Hence it is right skewed.

**BOXPLOT**

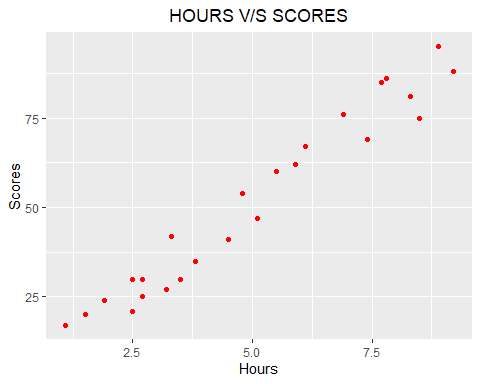
# Checking the outliers  
  
library(ggplot2)  
boxplot(data,main='Boxplot of Hours and Scores')



*From the boxplot, we can understand that No outliers are present in the dataset.*

**SCATTERPLOT**

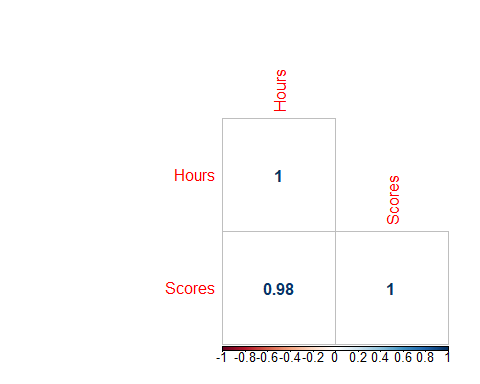
library(ggplot2)  
my\_graph <- ggplot(data,  
 aes(x =Hours, y =Scores))+  
 ggtitle('HOURS V/S SCORES')+  
 theme(plot.title = element\_text(hjust = 0.5))+  
 geom\_point(col='red')   
   
my\_graph



*From the graph we can see that there is a linear relationship between the response variable and explanatory variable. Also the direction of association seems to be positive i.e. As Hours increase, the Scores obtained also increase and vice-versa.*

**CORRELATION**

library(corrplot)  
corrplot(cor(data),  
method ='number',  
type = 'lower' # show only lower side  
)



#using pearson correlation  
  
cor.test(data$Hours,data$Scores)

##   
## Pearson's product-moment correlation  
##   
## data: data$Hours and data$Scores  
## t = 21.583, df = 23, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.9459248 0.9896072  
## sample estimates:  
## cor   
## 0.9761907

*Here, the correlation value is 0.9761907. Hence, we can understand that there exists a high positive correlation between Hours and Scores.*

### **DATA MODELLING**

*Train Test Splitting*

set.seed(100)  
rows=sample(nrow(data))  
  
#Randomly order data  
data=data[rows,]  
#Identify row to split on: split  
split = round(nrow(data) \* .80)

#Create train  
train=data[1:split,]  
#Create test  
test=data[(split+1):nrow(data),]

**Linear Regression Model**

#fitting linear regression model  
linmod = lm(Scores~Hours, data = train)  
  
#taking the summary of the model  
summary(linmod)

##   
## Call:  
## lm(formula = Scores ~ Hours, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8.983 -4.624 1.614 4.579 7.252   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.5030 2.9013 1.207 0.243   
## Hours 9.4682 0.5367 17.643 8.29e-13 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.508 on 18 degrees of freedom  
## Multiple R-squared: 0.9453, Adjusted R-squared: 0.9423   
## F-statistic: 311.3 on 1 and 18 DF, p-value: 8.29e-13

The value of intercept of the linear model is 3.5030. The slope of the model is 9.4682. Hence,the model can be interpreted as : .

* Residual standard error is the measure of the quality of the linear regreesion fit and here it is on degrees of freedom.
* R squared statistic provides measure of how well the model is fitting the actual dataset.Here, of fitting to the linear model.
* F statistic value is , which is relatively larger than 1. Hence, a good relationship is existing between Sales and Spend.

*Predicting the Scores*

Pred = predict(linmod, test)

*Comparing The Actual and Predicted Scores*

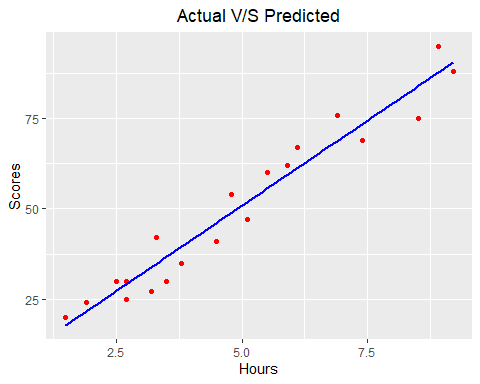
data.frame(Actual=test$Scores,Predicted=Pred)

## Actual Predicted  
## 15 17 13.91803  
## 11 85 76.40841  
## 9 81 82.08936  
## 1 21 27.17356  
## 25 86 77.35524

*Comparing The Actual and Predicted Values using Data Visualisation*

library(ggplot2)  
my\_graph <- ggplot(train,  
 aes(x =Hours, y =Scores))+  
 ggtitle('Actual V/S Predicted')+  
 theme(plot.title = element\_text(hjust = 0.5))+  
 geom\_point(col='red')+  
 stat\_smooth(method = 'lm',  
 col = 'blue',  
 se = FALSE,  
 size = 1)  
   
my\_graph

## `geom\_smooth()` using formula 'y ~ x'



#### **What will be the predicted score if the student studies for 9.25 hr/day?**

test2 = data.frame(Hours = 9.25)  
predict(linmod, test2)

## 1   
## 91.08419

*Therefore, according to the regression model, if a student studies for 9.25 hours per day he/she is likely to score 91.08419.*