Score Prediction with Machine Learning

Aziza Afrin

8/8/2021

INTRODUCTION

This project is about prediction using **Supervised Machine Learning** technique.Here,I predicted the percentage score of student based on the study hours. A simple linear regression model was built to predict the percentage score.

DATA

Let's Import the data using following code. And take the data table as data frame using as.data.frame.

```
#READING DATA
score_data=read.table("https://raw.githubusercontent.com/AdiPersonalWorks/Random/master/student_scores%20-%20student_scores.csv",header=TRUE,sep=",",dec=".")
as.data.frame(score_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
## 7    9.2     88
## 8    5.5     60
## 9    8.3     81
## 10   2.7     25
## 11   7.7     85
## 12   5.9     62
## 13   4.5     41
## 14   3.3     42
## 15   1.1     17
## 16   8.9     95
## 17   2.5     30
## 18   1.9     24
## 19   6.1     67
## 20   7.4     69
## 21   2.7     30
## 22   4.8     54
## 23   3.8     35
```

```
## 24   6.9     76
## 25   7.8     86
```

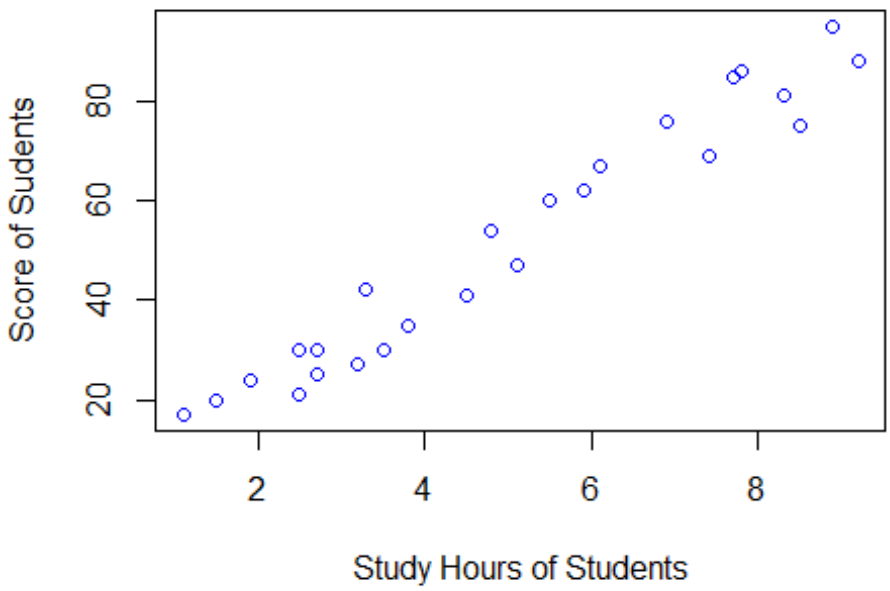
Now, to get an overview of the data we run the following code. This gives the summary statistics of the data. Here, the head function returns the first six observation of the data.

```
head(score_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

summary (score_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
```

We see our data contains two variable **Hours** and **Score**. Hours variable represent the study hour of the students. Score variable presents the percentage score of the students. We have twenty five observation in total.

SCATTER PLOT

To get more insights about the data, we present the data visually in a scatter plot.



The plot shows a relationship between the two variable **Score** and **Hours**. As the data point shows upward pattern, we can assume that there is a linear relationship between the variables.

CORRELATION

To measure the relationship between the variable ,we calculate correlation.

```
#CORRELATION
cor(score_data$Hours,score_data$Scores)

## [1] 0.9761907
```

Correlation value 0.976 suggests a strong positive relation.

PARTITION OF THE DATA

Now we divide our data set into two set as train set and test set. We will build our model based on the train data set.

```
#SPLIT
set.seed(10000)
library(lattice)
library(ggplot2)
library(caret)

## Warning: package 'caret' was built under R version 4.0.5
```

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --
```

```
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
```

```r
y<-score_data$Scores
train_index <- createDataPartition(y, times = 1, p = 0.80, list = FALSE)

train_set <- score_data %>% slice(train_index)
train_set
```

```
##    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
## 7    5.5     60
## 8    8.3     81
## 9    2.7     25
## 10   7.7     85
## 11   4.5     41
## 12   3.3     42
## 13   1.1     17
## 14   8.9     95
## 15   2.5     30
## 16   6.1     67
## 17   7.4     69
## 18   2.7     30
## 19   4.8     54
## 20   3.8     35
## 21   6.9     76
## 22   7.8     86
```

```r
test_set <- score_data %>% slice(-train_index)
test_set
```

```
##   Hours Scores
## 1   9.2     88
```

```
## 2   5.9     62
## 3   1.9     24
```

So we see our train data set has twenty two rows and test data set has three rows.This partition is done based on the ratio we have given in the above code.

FORMATION OF THE MODEL

As scatter plot of the data shows a linear relationship between the variable, we will fit a linear regression model for the train data set. Then we predict the **Score** value for the test data set.

```
#LINEAR REGRESSION MODEL
fit <- lm(Scores ~Hours, data = train_set)
fit$coef

## (Intercept)       Hours
##    1.441785    9.984125

predicted_scores <- predict(fit,test_set)
as.data.frame(predicted_scores)

##   predicted_scores
## 1         93.29573
## 2         60.34812
## 3         20.41162

actual_scores <-test_set$Scores
actual_scores

## [1] 88 62 24
```

Now we will see the actual score value and predicted score value of the test data set side by side. So we can compare these value easily.

```
#SEE VALUES COMBINDLY

cbind(predicted_scores,actual_scores)

##   predicted_scores actual_scores
## 1         93.29573            88
## 2         60.34812            62
## 3         20.41162            24
```

ACCURACY

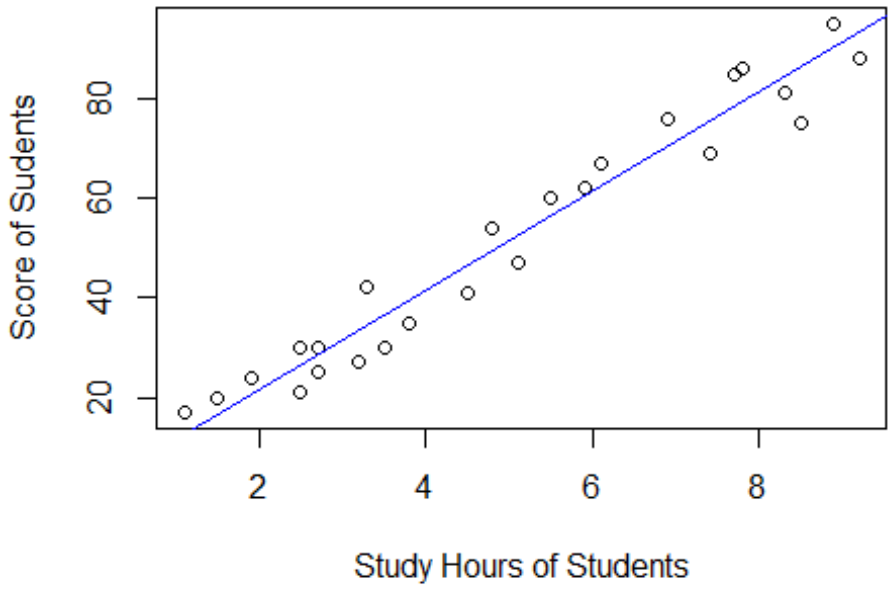
We calculate a root mean squared error to see the accuracy of our model.

```
#CALCULATE ROOT MEAN SQUARED ERROR

rmse<-mean((actual_scores-predicted_scores)^2)
rmse
```

```
## [1] 14.54999
```

At this point we again plot the data along with the fitted regression line.

```
#PLOT AGAIN
plot(score_data,xlab="Study Hours of Students",ylab="Score of Sudents")
abline(fit,col="blue")
```



TEST THE MODEL

We check our model with a given **Hours** value.

```
#CHECK MODEL
new <- data.frame(Hours=c(9.25))
predict(fit, newdata = new)

##        1
## 93.79494
```

So, predicted score value of the student who studied 9.25 hours per day is 93.79.

CONCLUSION

To form a supervised machine learning model we divide the data set into train and test data set. Based on train data set we fit a simple linear regression model. We tested the model on test data set and compare them with the actual value. From root mean squared

error we checked our model accuracy. Finally predicted a score value for a given hours value.