

# Task 1 - Prediction using Supervised ML

SANJEETA CHAKRABORTY

4/4/2021

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

## Loading Dataset

```
data <- read.csv("https://raw.githubusercontent.com/AdiPersonalWorks/Random/master/student_scores%20-%20student_scores.csv")
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
```

## Getting Insights from the data



### Data Summary

```
library(skimr)
skim(data)
```

#### Data summary

Name	data
Number of rows	25
Number of columns	2
Column type frequency:	
numeric	2
Group variables	None

**Variable type: numeric**

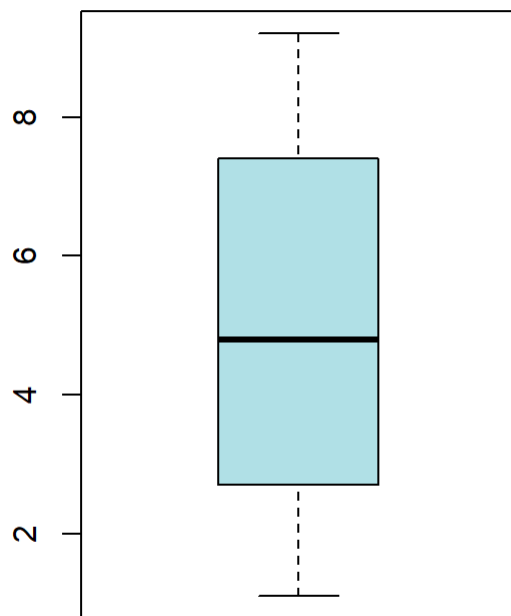
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Hours	0	1	5.01	2.53	1.1	2.7	4.8	7.4	9.2	
Scores	0	1	51.48	25.29	17.0	30.0	47.0	75.0	95.0	

## Outlier detection

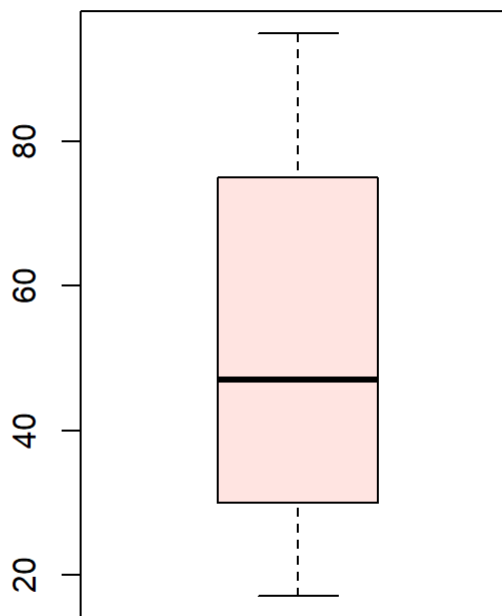
```

par(mfrow=c(1, 2))
boxplot(data$Hours, main="Number of study hours", col='powderblue', sub=paste("Outlier rows: ",
  boxplot.stats(data$Hours)$out))
boxplot(data$Scores, main="Score", col='mistyrose', sub=paste("Outlier rows: ", boxplot.stats(da
ta$Scores)$out))

```

**Number of study hours**

Outlier rows:

**Score**

Outlier rows:

**There are no outliers present in the dataset.**

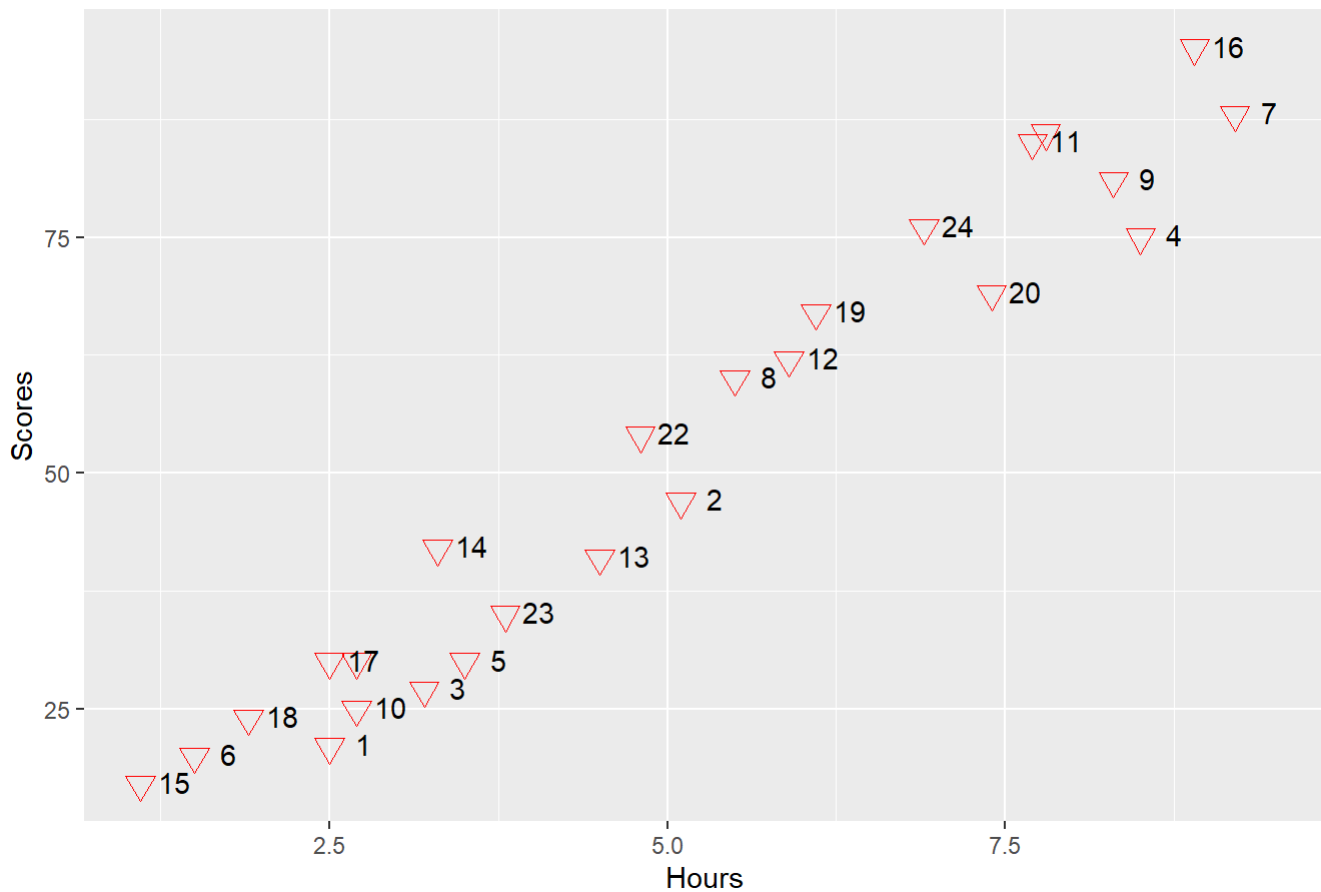
## Checking the relationship between the variables

```

library(ggplot2)
ggplot(data, aes(x=Hours, y=Scores)) + geom_point(shape=6,color="red",size=3.5) + geom_text(labe
l=rownames(data), nudge_x = 0.25, nudge_y = 0.25, check_overlap = TRUE) + ggtitle("Scatterplot")

```

## Scatterplot



From the plot, We can infer that number of study hours and score is linearly related and it follows a positive uphill trend.

## Model Building

### Splitting data into training and testing set

```
set.seed(42)
library(caTools)
split<-sample.split(data,SplitRatio=0.7)
```

```
train<-subset(data,split==T)
head(train)
```

```
##      Hours Scores
## 1      2.5      21
## 3      3.2      27
## 5      3.5      30
## 7      9.2      88
## 9      8.3      81
## 11     7.7      85
```

```
test<-subset(data,split==F)
head(test)
```

```
##      Hours Scores
## 2      5.1      47
## 4      8.5      75
## 6      1.5      20
## 8      5.5      60
## 10     2.7      25
## 12     5.9      62
```

```
lmmodel<-lm(Scores~Hours, data = train)
lmmodel
```

```
##
## Call:
## lm(formula = Scores ~ Hours, data = train)
##
## Coefficients:
## (Intercept)      Hours
##      -0.3332      10.2119
```

**Our regression equation is:  $y = -0.3332 + 10.2119X$ , that is Score =  $-0.3332 + 10.2119 \times \text{Number of Study hours}$ .**

**For every unit increase in number of study hours, the percentage scored by the student will increase by 10 units. The constant shows the percentage scored by the student when the number of study hours is zero. Generally it is considered the constant doesn't have any meaningful interpretation.**

```
summary(lmmodel)
```

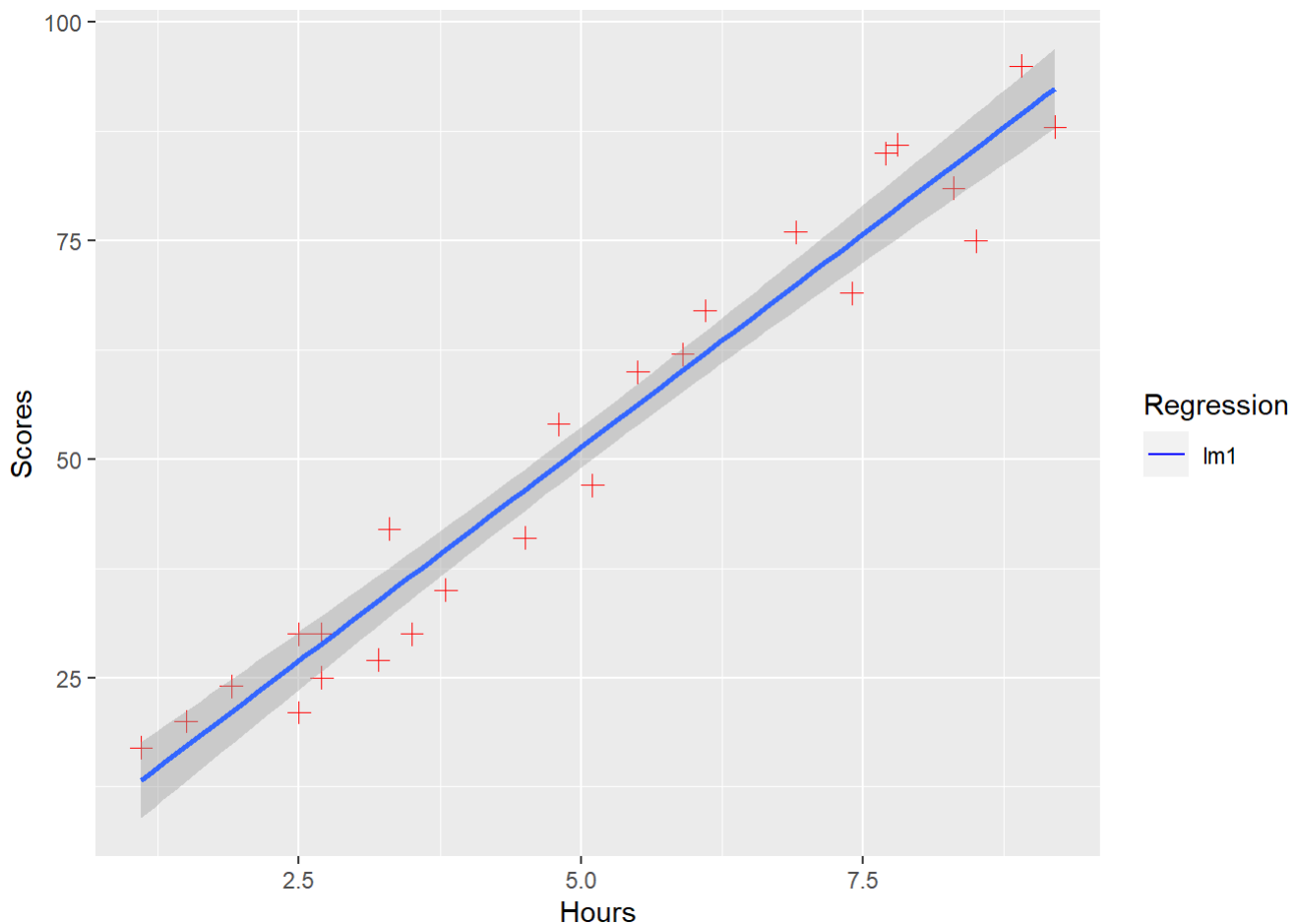
```
##
## Call:
## lm(formula = Scores ~ Hours, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.617  -4.621  -3.426   5.040   6.701
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.3332     3.2799  -0.102   0.921
## Hours        10.2119     0.5996  17.032 2.97e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.518 on 11 degrees of freedom
## Multiple R-squared:  0.9635, Adjusted R-squared:  0.9601
## F-statistic: 290.1 on 1 and 11 DF, p-value: 2.971e-09
```

The R-square value is 0.96. This implies 96% of the variation in response variable is explained by the predictor variable and the remaining 4% is due to error.

## Regression plot

```
library(ggplot2)
ggplot(data,
       aes(x = Hours, y = Scores)) +
  geom_point(shape=3,color="red",size=2.5) +
  geom_smooth(method = 'lm', formula = y~x,
             se = FALSE, size = 0.5,
             aes(color = "lm1")) +
  geom_smooth(method = 'lm') +
  scale_color_manual(name = "Regression", values = "Blue")
```

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



## Predicted values

```
predictions <- data.frame(cbind(Actual = test$Scores, Predicted = predict(lmmodel, test)))
predictions
```

##	Actual	Predicted
## 2	47	51.74774
## 4	75	86.46834
## 6	20	14.98474
## 8	60	55.83252
## 10	25	27.23908
## 12	62	59.91729
## 14	42	33.36624
## 16	95	90.55312
## 18	24	19.06952
## 20	69	75.23521
## 22	54	48.68416
## 24	76	70.12924

```
hr = data.frame(Hours=c(9.25))
pred = predict(lmmodel, hr)
sprintf("Predicted Score when number of study hours is 9.25: %s", format(round(pred, 2), nsmall = 2))
```

```
## [1] "Predicted Score when number of study hours is 9.25: 94.13"
```

**Therefore, a student who studies for 9.25 hrs will score 94.13%.**