The Sparks Foundation Task Report

Kirt Preet Singh

7/18/2020

Required Libraries

```
library(caret)
library(rattle)
```

Task 1

Question 1 (To Explore Supervised Machine Learning)

In this regression task we will predict the percentage of marks that a student is expected to score based upon the number of hours they studied. This is a simple linear regression task as it involves just two variables. Data can be found at http://bit.ly/w-data What will be predicted score if a student study for 9.25 hrs in a day?

Loading and Exploring Data

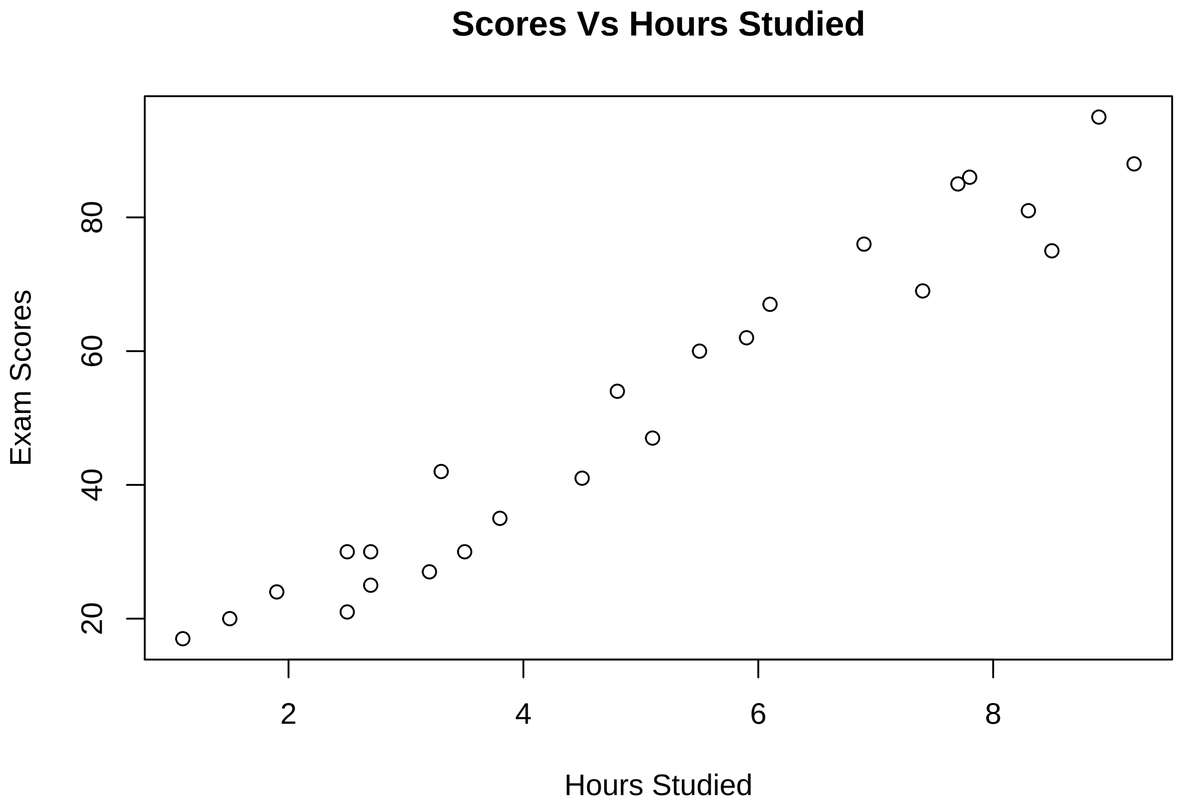
```
studied <- read.csv("tsk1.csv")
head(studied)
```

```
##   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
```

An Exploratory View

We see an outlook of the data above, to observe the relationships between the two variables we would wanna take a look a pictorial reprentation of the data

```
plot(x = studied$Hours,y = studied$Scores,main = "Scores Vs Hours Studied",xlab = "Hours Studied",ylab
```



Now we can quite confidently say that more **number of hours studied** actually yields hightened **exam scores**, and hope our prediction also yields an output alligned with the approach.

Builing and exploring a Regression model

```
modFit <- lm(Scores~Hours,data = studied)
summary(modFit)
```

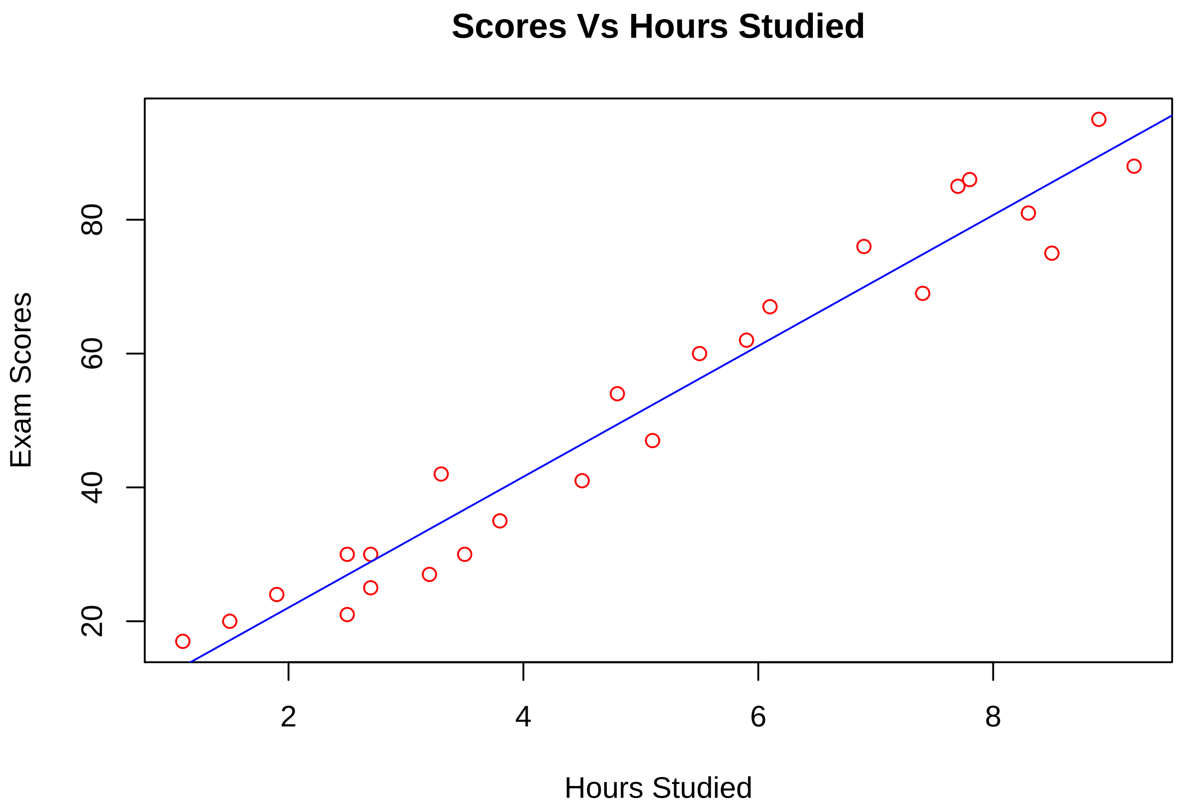
```
##
## Call:
## lm(formula = Scores ~ Hours, data = studied)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.578  -5.340   1.839   4.593   7.265
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.4837     2.5317   0.981    0.337
## Hours         9.7758     0.4529  21.583   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 5.603 on 23 degrees of freedom
## Multiple R-squared:  0.9529, Adjusted R-squared:  0.9509
## F-statistic: 465.8 on 1 and 23 DF,  p-value: < 2.2e-16
```

The R-squared value stands in favour of the model. Now we shall see how the regression line fits in our data.

Regression Analysis

```
plot(x = studied$Hours,y = studied$Scores,main = "Scores Vs Hours Studied",xlab = "Hours Studied",ylab =
abline(modFit, col="blue")
```



Our Regression line fits perfectly alligned with our data points

Prediction

Now we shall exploit the regression model, to yield a prediction of exam score for a student who studied 9.5 hours.

```
nwdat <- data.frame(Hours = 9.5)
predict(modFit,newdata = nwdat)
```

```
##        1
## 95.35381
```

Our model yields a prediction of 95.1 exam score, a value we can settle with, considering the outlook of our data and the strong 95% fit of our regression model.

Task 2

Q3. (To Explore Decision Tree Algorithm)

For the given 'Iris' dataset, create the Decision Tree classifier and visualize it graphically. The purpose is if we feed any new data to this classifier, it would be able to predict the right class accordingly. Dataset : https://drive.google.com/file/d/11Iq7YvbWZbt8VXjfm06brx66b10YiwK-/view?usp=sharing

Loading and Exploring the data

```
irisdata <- read.csv("iris.csv")
head(irisdata)
```

```
##   Id SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm     Species
## 1  1           5.1          3.5           1.4          0.2 Iris-setosa
## 2  2           4.9          3.0           1.4          0.2 Iris-setosa
## 3  3           4.7          3.2           1.3          0.2 Iris-setosa
## 4  4           4.6          3.1           1.5          0.2 Iris-setosa
## 5  5           5.0          3.6           1.4          0.2 Iris-setosa
## 6  6           5.4          3.9           1.7          0.4 Iris-setosa
```

We see that our data contains some unnecessary variables such as the indexing variable **id** , and the Species variable for which the a machin learning algorithm is to be build, contains *Iris-* before mentioning the actual species.

Data Transformation

```
irisdata <- irisdata[,-1]
irisdata$Species <- gsub(pattern = "Iris-",replacement = "",x = irisdata$Species,useBytes = TRUE)
head(irisdata)
```

```
##   SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm Species
## 1           5.1          3.5           1.4          0.2  setosa
## 2           4.9          3.0           1.4          0.2  setosa
## 3           4.7          3.2           1.3          0.2  setosa
## 4           4.6          3.1           1.5          0.2  setosa
## 5           5.0          3.6           1.4          0.2  setosa
## 6           5.4          3.9           1.7          0.4  setosa
```

Now this data form is desirable.

Building a Tree Model

Making Training and Testing Datasets

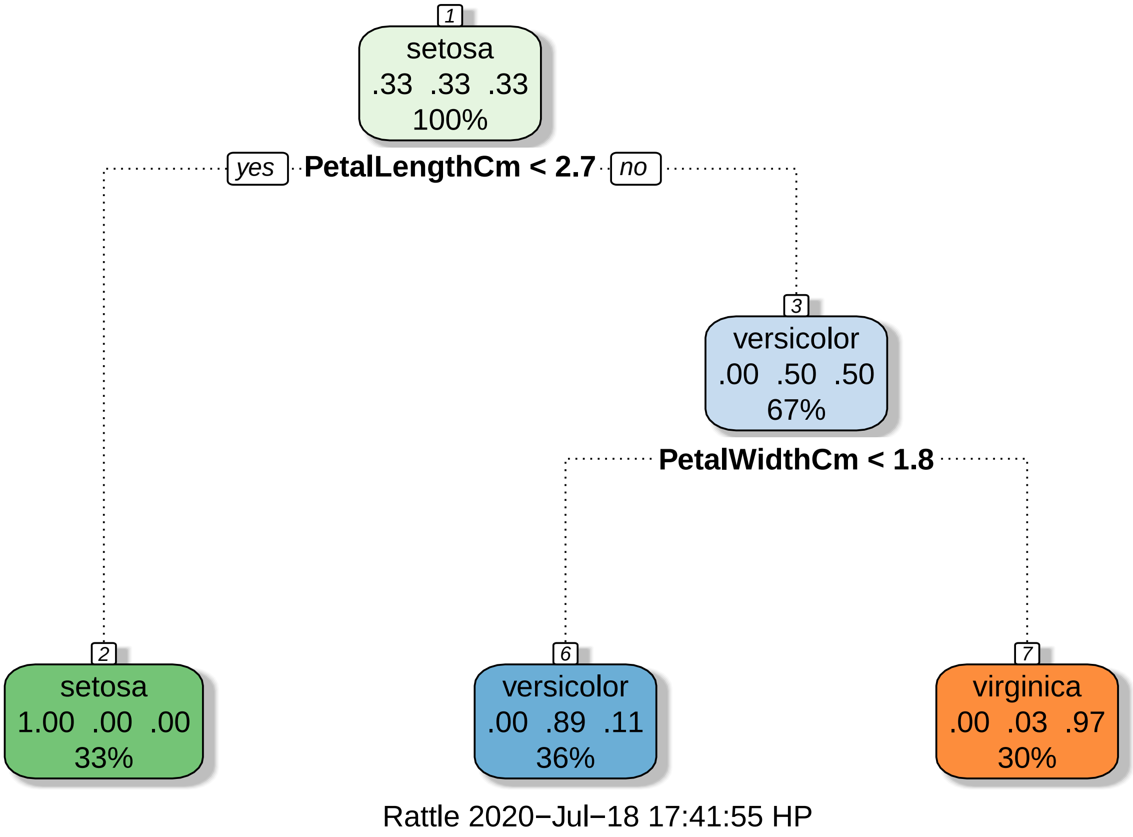
```
set.seed(72020)
inTrain <- createDataPartition(irisdata$Species,p = 0.7,list = FALSE)
training <- irisdata[inTrain,]
testing <- irisdata[-inTrain,]
testing$Species <- as.factor(testing$Species)
```

Untill now training(70%) and testing(30%) data sets have been created, and the species in testing have been factorizwd for further comparisons.

Decision Tree Model

We build our model across the training set.

```
treeFit <- train(Species~.,data = training,method = "rpart")
fancyRpartPlot(treeFit$finalModel)
```



Given above is our Decision Tree Model

Model Testing

```
pr <- predict(treeFit,testing)
cmat <- confusionMatrix(pr,testing$Species)
cmat$overall[1]
```

```
##  Accuracy
## 0.9777778
```

Our Model was fed with a set of new values from the testing set and as it comes, has an accuracy of almost 98%, so with confidence we can say that if the model is applied across a new set of values, there is a 98% chance of obtaining a correct value.