Model

Azeez

24/12/2021

## This workbook is to practice Machine Learning Algorithm using caret package.

The initial step is to call the library caret

## Dataset

Next is to load the data. The data to be used is iris dataset. Reason being that the data has both numerical and categorical data to compute the supervised machine learning.

head(iris)

## Sepal.Length Sepal.Width Petal.Length Petal.Width 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

## Understanding the structure of the data

As data analyst, we need to check the structure of the data.

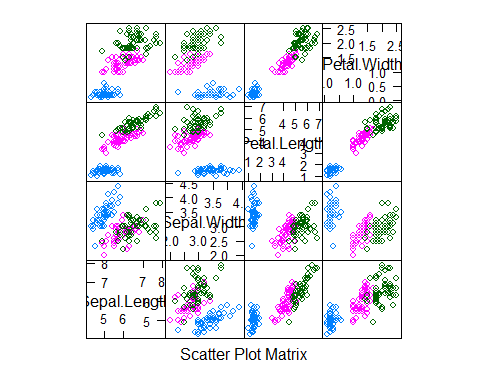
## 'data.frame': 150 obs. of 5 variables:  
## $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...  
## $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...  
## $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...  
## $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...  
## $ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

## [1] "data.frame"

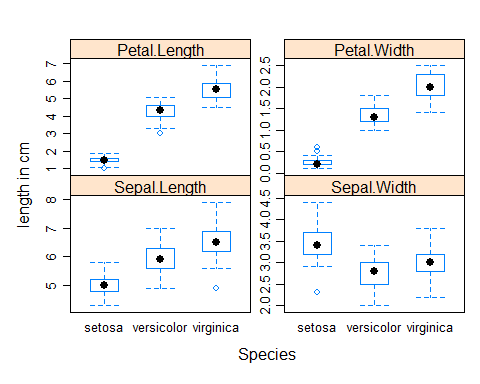
## Data Exploration

This is done to get the trend and information to explore from the data. We do this through visualization,

featurePlot(x=iris[,1:4],y=iris$Species, plot = "pairs")

 Let use Box plot to show the distributions of the numerical data for the species.

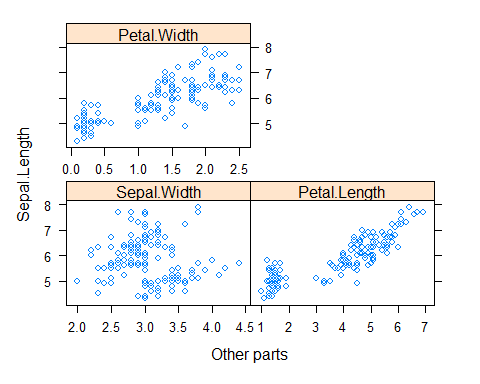
featurePlot(x=iris[,1:4], y=iris$Species, plot = "boxplot", auto.key = list(columns = 3), scale=list(y=list(relation="free")), labels=c('Species', 'length in cm'))

 ##Interpretation of the Visualisation The result shows the distributions of the flower parts in all the three flowers. It was observed that in petal.length, setosa flower has the low mean value while virginica flower has high value of petal.length with high mean value. setosa and versicolor have outliers.

## Scatter plot

Let check the relationship between the numerical data. To do this we use scatterplot.

featurePlot(x=iris[,2:4],y=iris$Sepal.Length, plot = "scatter", labels=c("Other parts", "Sepal.Length"),scale=list(x=list(relation="free")))

 ## Observation It was observed from the visualization that there is a relationship between the all the flower parts. Sepal.length and petal.length shows promiscuous relationship. To further shows this relationship we conduct correlation to determine the correlation coefficient.

##Correlation

cor\_iris = cor(x=iris[,1:4], use = "everything", method = c("pearson"))  
cor\_iris

## Sepal.Length Sepal.Width Petal.Length Petal.Width  
## Sepal.Length 1.0000000 -0.1175698 0.8717538 0.8179411  
## Sepal.Width -0.1175698 1.0000000 -0.4284401 -0.3661259  
## Petal.Length 0.8717538 -0.4284401 1.0000000 0.9628654  
## Petal.Width 0.8179411 -0.3661259 0.9628654 1.0000000

For proper understanding of the correlation result, we visualize it ## Correlation visualization

#corrplot(cor\_iris, method = "ellipse", type =c("full"))

We may like to check the summary of our flower parts.

summary(iris[1:4])

## Sepal.Length Sepal.Width Petal.Length Petal.Width   
## Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100   
## 1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300   
## Median :5.800 Median :3.000 Median :4.350 Median :1.300   
## Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199   
## 3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800   
## Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500

summary(iris)

## Sepal.Length Sepal.Width Petal.Length Petal.Width   
## Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100   
## 1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300   
## Median :5.800 Median :3.000 Median :4.350 Median :1.300   
## Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199   
## 3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800   
## Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500   
## Species   
## setosa :50   
## versicolor:50   
## virginica :50   
##   
##   
##

From the exploration of the data we see a trend and we can go ahead with our model.

##Algorithm For our model, we will use trainControl with boot as the algorithm

trc = trainControl(method = "boot", number = 25)

## Linear Regression model

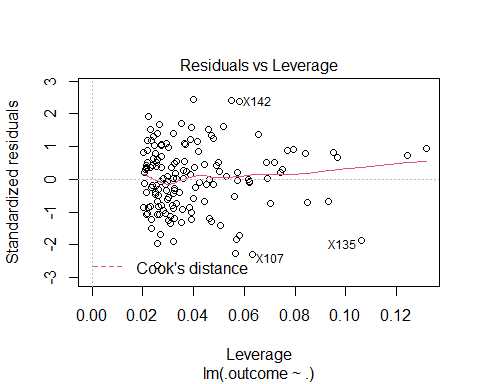
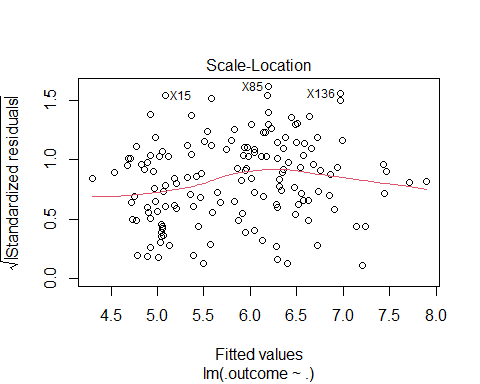
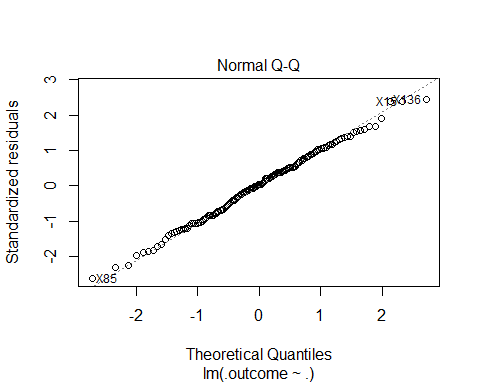
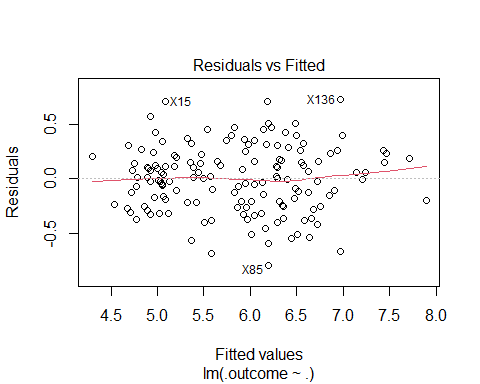
model\_ft = train(Sepal.Length~.,iris, method="lm", trControl = trc)  
model\_ft

## Linear Regression   
##   
## 150 samples  
## 4 predictor  
##   
## No pre-processing  
## Resampling: Bootstrapped (25 reps)   
## Summary of sample sizes: 150, 150, 150, 150, 150, 150, ...   
## Resampling results:  
##   
## RMSE Rsquared MAE   
## 0.3189253 0.8540669 0.2590816  
##   
## Tuning parameter 'intercept' was held constant at a value of TRUE

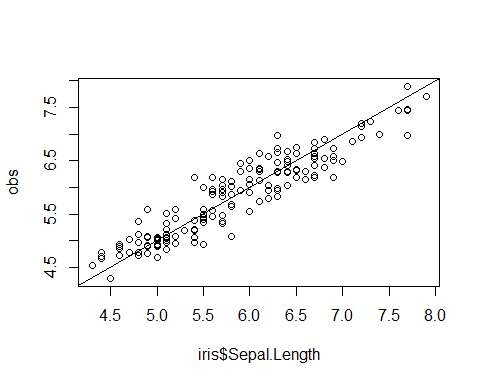
summary(model\_ft)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.79424 -0.21874 0.00899 0.20255 0.73103   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.17127 0.27979 7.760 1.43e-12 \*\*\*  
## Sepal.Width 0.49589 0.08607 5.761 4.87e-08 \*\*\*  
## Petal.Length 0.82924 0.06853 12.101 < 2e-16 \*\*\*  
## Petal.Width -0.31516 0.15120 -2.084 0.03889 \*   
## Speciesversicolor -0.72356 0.24017 -3.013 0.00306 \*\*   
## Speciesvirginica -1.02350 0.33373 -3.067 0.00258 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3068 on 144 degrees of freedom  
## Multiple R-squared: 0.8673, Adjusted R-squared: 0.8627   
## F-statistic: 188.3 on 5 and 144 DF, p-value: < 2.2e-16

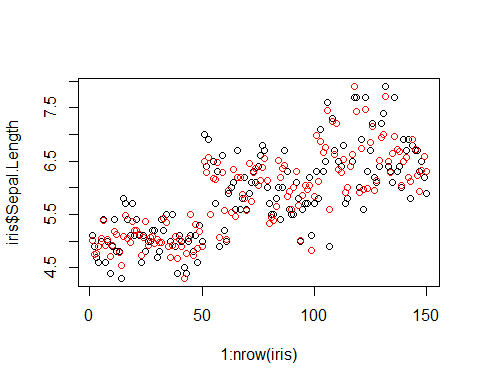
plot(model\_ft$finalModel)

 ## Test model by predicting the train data from the model

obs =predict(model\_ft, iris)  
#head(obs)  
#head(iris)  
plot(iris$Sepal.Length, obs)  
abline(0,1)



plot(1:nrow(iris), iris$Sepal.Length)  
points(1:nrow(iris),obs, col="red")



## Test model by predicting from new data

We want to know the Sepal.Length with new value of other parts of the flower. To do this, we new to create new dataset

new\_data <- data.frame(Sepal.Width=c(1.7, 3.9, 4.0),  
 Petal.Length =c(1.0, 2.8, 7.1),  
 Petal.Width = c(0.4, 2.7, 3.2),  
 Species =c("setosa", "versicolor", "virginica"))  
new\_data

## Sepal.Width Petal.Length Petal.Width Species  
## 1 1.7 1.0 0.4 setosa  
## 2 3.9 2.8 2.7 versicolor  
## 3 4.0 7.1 3.2 virginica

Now, we have the new data set, let predict the Sepal.Length from them

obs = data.frame(Predicted\_Sepal.Length=c(predict(model\_ft, new\_data)))  
obs

## Predicted\_Sepal.Length  
## 1 3.717459  
## 2 4.852635  
## 3 8.010459

## To model the flower from the flower part.

We are to perform classification, since our outcome is categorical To do this we change our linear regression model to random forest.

model\_rf = train(Species~.,iris, method="rf", trControl = trc)  
model\_rf

## Random Forest   
##   
## 150 samples  
## 4 predictor  
## 3 classes: 'setosa', 'versicolor', 'virginica'   
##   
## No pre-processing  
## Resampling: Bootstrapped (25 reps)   
## Summary of sample sizes: 150, 150, 150, 150, 150, 150, ...   
## Resampling results across tuning parameters:  
##   
## mtry Accuracy Kappa   
## 2 0.9556750 0.9327985  
## 3 0.9599324 0.9392361  
## 4 0.9548891 0.9315671  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final value used for the model was mtry = 3.

##To predict the type of flower from the model

flower\_typ = predict(model\_rf, iris)  
confusionMatrix(iris$Species, flower\_typ)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction setosa versicolor virginica  
## setosa 50 0 0  
## versicolor 0 50 0  
## virginica 0 0 50  
##   
## Overall Statistics  
##   
## Accuracy : 1   
## 95% CI : (0.9757, 1)  
## No Information Rate : 0.3333   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 1   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: setosa Class: versicolor Class: virginica  
## Sensitivity 1.0000 1.0000 1.0000  
## Specificity 1.0000 1.0000 1.0000  
## Pos Pred Value 1.0000 1.0000 1.0000  
## Neg Pred Value 1.0000 1.0000 1.0000  
## Prevalence 0.3333 0.3333 0.3333  
## Detection Rate 0.3333 0.3333 0.3333  
## Detection Prevalence 0.3333 0.3333 0.3333  
## Balanced Accuracy 1.0000 1.0000 1.0000

##To predict flower type from new data

new\_data <- data.frame(Sepal.Width=c(1.7, 3.9, 4.0),  
 Sepal.Length = c(5.0, 6.7, 9.0),  
 Petal.Length =c(1.0, 2.8, 7.1),  
 Petal.Width = c(0.4, 2.7, 3.2))  
obs <- data.frame(new\_flower=c(predict(model\_rf, new\_data)))  
obs

## new\_flower  
## 1 setosa  
## 2 virginica  
## 3 virginica