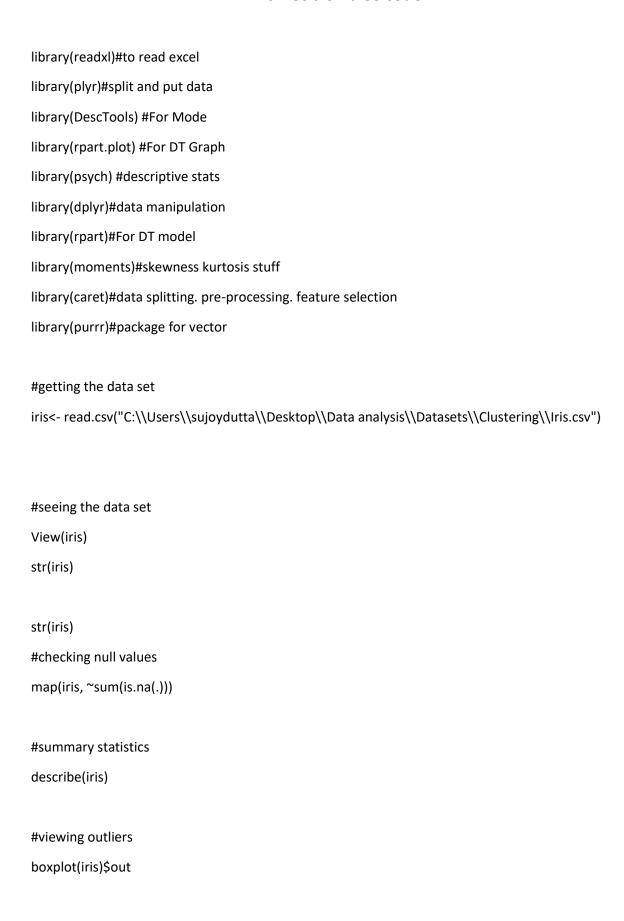
#Iris Decision tree code



```
#dropping uneccesary outliers
iris<- iris[-c(1)]
iris
# Eliminating outliers using Quartile method
iqr <- IQR(iris$SepalWidthCm)</pre>
Q <- quantile(irisk$SepalWidthCm, probs=c(.25, .75), na.rm = FALSE)
eliminated <- subset(iris, irisk$SepalWidthCm > (Q[1] - 1.5*iqr) & iris$SepalWidthCm <
(Q[2]+1.5*iqr))
iqr <- IQR(eliminated$SepalWidthCm)</pre>
Q <- quantile(eliminated$SepalWidthCm, probs=c(.25, .75), na.rm = FALSE)
irisdt <- subset(eliminated, eliminated$SepalWidthCm > (Q[1] - 1.5*iqr) & eliminated$SepalWidthCm
< (Q[2]+1.5*iqr)
remove(eliminated)
# For reproducibility
set.seed(100)
# Create index for testing and training data
indexes = sample(80, 70)
#splitting dataset into train and test
iris_train = iris[indexes,]
iris_test = iris[-indexes,]
#building the model
target = Species ~ SepalLengthCm + SepalWidthCm + PetalLengthCm + PetalWidthCm
target = Species ~.
```

```
#Build and plot model
tree = rpart(target, data = iris_train, method = "class")
rpart.plot(tree)

#predictions
predictions = predict(tree, iris_test)

#confusion matrix
table(predictions[0:150], iris$Species[0:150])
```