The Task:

The purpose of this analysis is to predict the survival of passengers of the Titanic. So, to solve this problem, the nature of the relationship between of each variable with Survived must be understood along with the individual characteristic of each variable. A statistical model is then adopted to further the analysis and arrive at the results and interpretation.

The Dataset:

The dataset contains the following variables:

**PassengerId**

**Survived**

**Pclass**

**Name**

**Sex**

**Age**

**SibSp**

**Parch**

**Ticket**

**Income**

**Fare**

**Cabin**

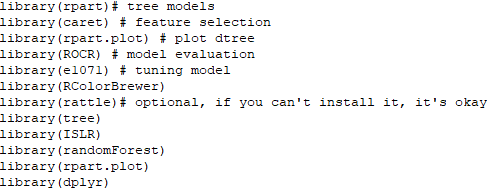
**Embarked**

The statistical model:

Classification is the method of predicting the class of a given input data point. Classification problems are common in machine learning and they fall under the Supervised learning method.

Random forest algorithm is a supervised classification and regression algorithm. As the name suggests, this algorithm randomly creates a forest with several trees. Generally, the more trees in the forest the more robust the forest looks like. Similarly, in the random forest classifier, the higher the number of trees in the forest, greater is the accuracy of the results. In simple words, Random forest builds multiple decision trees (called the forest) and glues them together to get a more accurate and stable prediction. The forest it builds is a collection of Decision Trees, trained with the bagging method.

Setting up the R model by loading the required libraries:



Data:

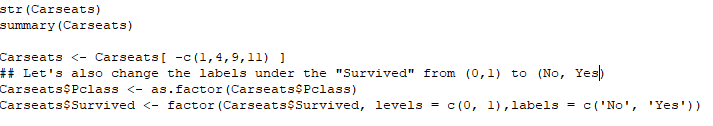
In the next step, the data is read into the R environment from the file.

setwd("C:\\Users\\ADMIN\\Desktop\\R Models\\Decision Tree")

Carseats <- read.csv("Titanic.csv")

Data selection and data type modification:

In the given dataset, the columns which are named as “PassengerId”, “Name”, “Ticket” and “Fare” contains customerid details, names of the passengers, ticket details and fare are of no relevance to our model and is thus excluded. Two of the variables, Pclass and Survived are converted to factors and Survived was in binary form so we converted 0 = “No” and 1 = “Yes”.



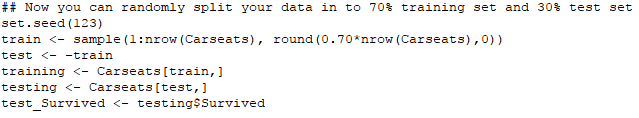
Checking for missing values:

After the data has been cleaned off all the outliers, it is then checked for any missing values in the following manner:

264 missing values were found and has been removed.

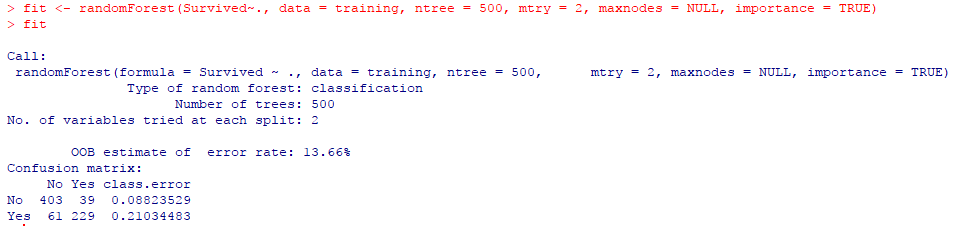
Splitting the data:

The data is then split into two parts. 70 percent of the data is split into ‘development’ and the remaining 30 percent is named ‘validation’. The development part is for training the decision tree model and the validation part is for testing the robustness of the model.



**Building the Random Forest model with default parameters:**

Once the data has been divided into two parts, a random forest has been performed with the Survived as the dependent variable. Here, we are taking the value of ntree = 500, mtry = 2 and maxnodes = NULL as these values are default values of the given parameters.

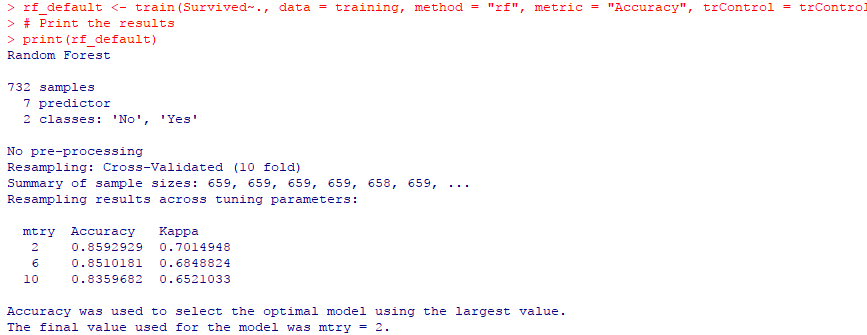


Here we can see with default parameter the out of bag estimate of error rate is 13.66%.

Define Control function has been used to train other algorithms. The grid search method is simple, the model will be evaluated over all the combination you pass in the function, using cross-validation.

C:\Users\ADMIN\Desktop\Titanic RF\define control.PNG

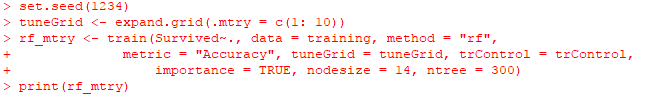
**Running** the model:

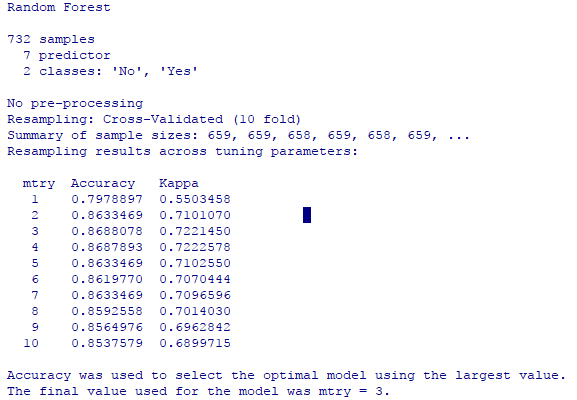


We have trained the machine using the above algorithm in such a way that it will check all the possible accuracies by evaluating over-all combinations of mtry. The final value for the model was mtry = 2 because at mtry = 2 the accuracy is 85.9% which is the highest among all other values.

**Tuning the hyperparameters of the model for better result:**

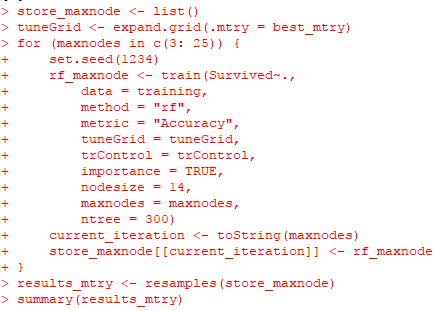
* Search the best mtry by testing the model with mtry 1 to 10.

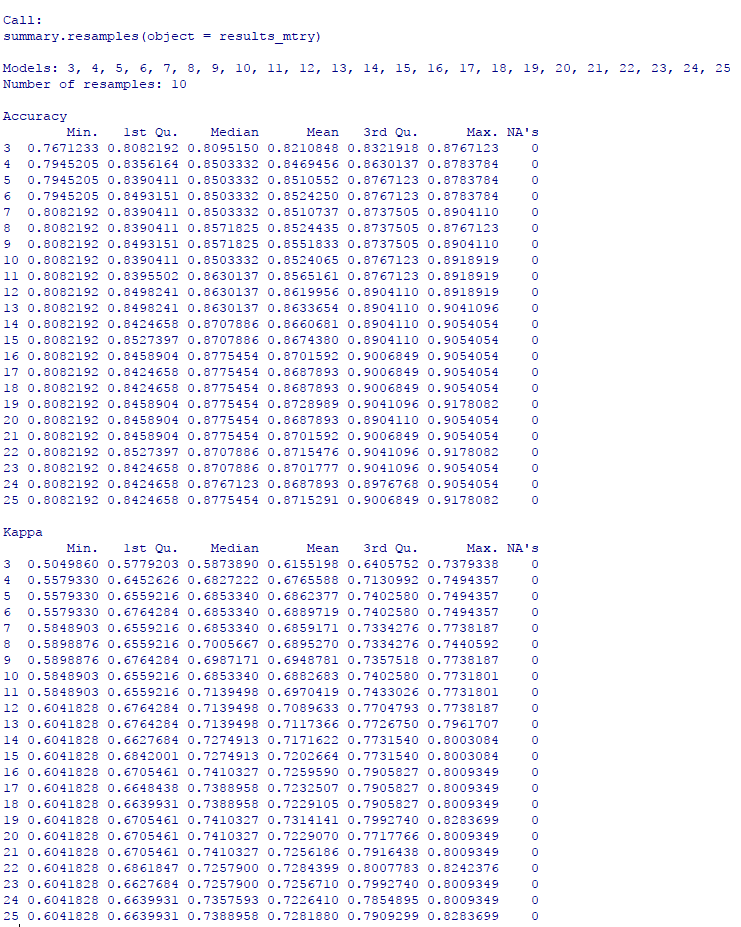




The model has given the best mtry to be 3.

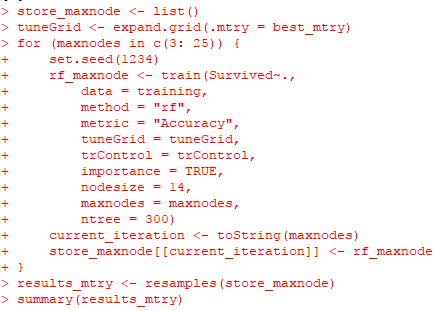
* Search the best maxnodes using the best mtry.

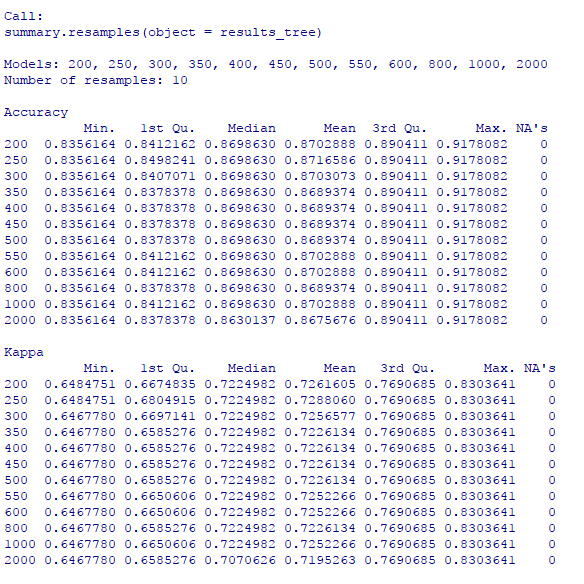




We will use maxnodes = 19 as the model has shown us that it has the highest accuracy of 87%.

* Search the best ntrees using best maxnodes

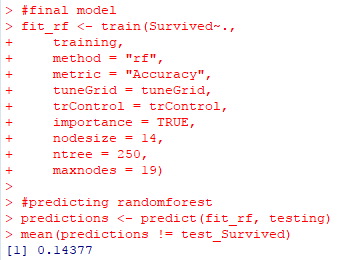




We will use ntree = 250 as the model has shown us that it has the highest accuracy of 87%.

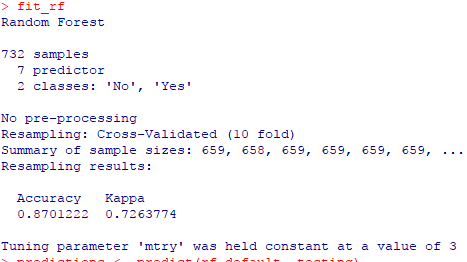
**Final Model:**

The final model is based on the tuning part of the model where we used best maxnodes, best ntree and best mtry.

****

**Validation of the model:**

Checking the confusion matrix to understand how good the model is with a tuned forest.



The model accuracy have increased from 85.6% to 87%.