**Data Mining using Random Forest/Decision Tree and SVM**

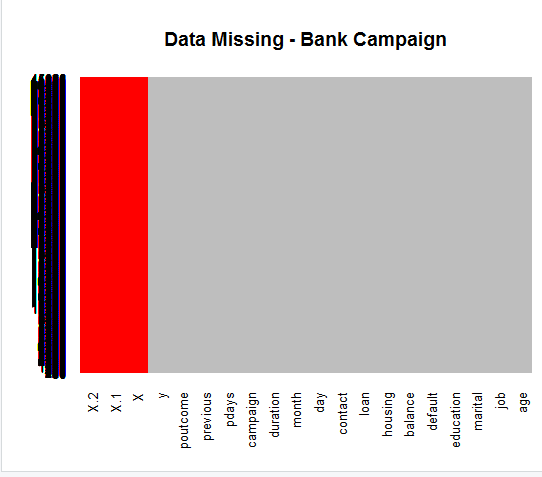
**To predict and identify the clients who all will subscribe to term deposit for the campaign held.**

**Steps involved in the code**

**1. First data loaded**

**2. Removing column having NA as the values. Part 2 of the code is doing this.**

**Below is the screenshot of the columns coloured in Red**

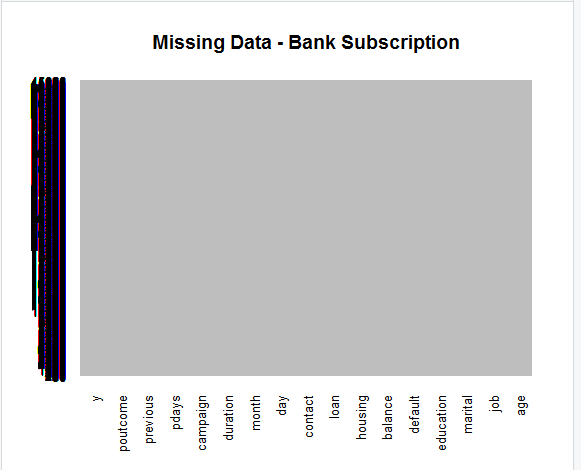
****

**Rows and columns:**

****

**3. Used Apply function where function=2 for removing the columns.**

**Below is the screenshot**

****

**#removed the non-required columns.**

**4. Part 3 of the Code removed the outlier’s and did sub selection of the data**

**i. e. Subset of the columns numbers**

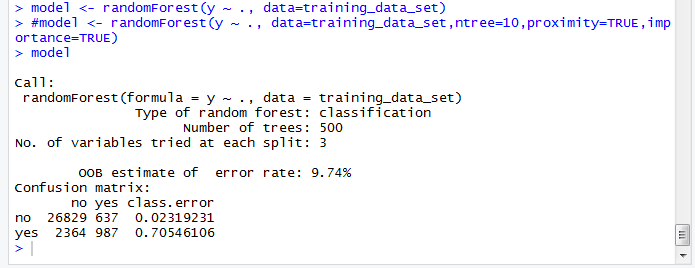
**5. Part 4 of the code performed the data transformation.**

**6. PART 5 of the code performed the Splitting of the data.**

**Rows selection for training data set in the ratio of 70:30 here prob=0.7**

**7. PART 6, Implemented the Random Forest model on the training dataset.**

**Below is the Model information**

****

**Above model is the Random Forest of type Classification.**

**Here we are classifying our output into two category which is**

**about whether the client will subscribe to the term deposit or not**

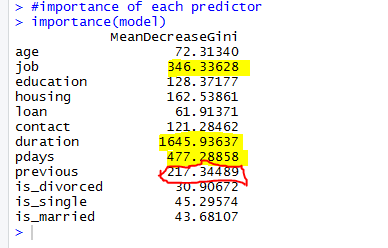
**In this model our outcome will be decided on the outcome of the 500 decision trees**

**Which means every Node of the decision tree will split into three Daughter Nodes.**

**OOB compares predicted with actual it can be checked using the confusion matrix.**

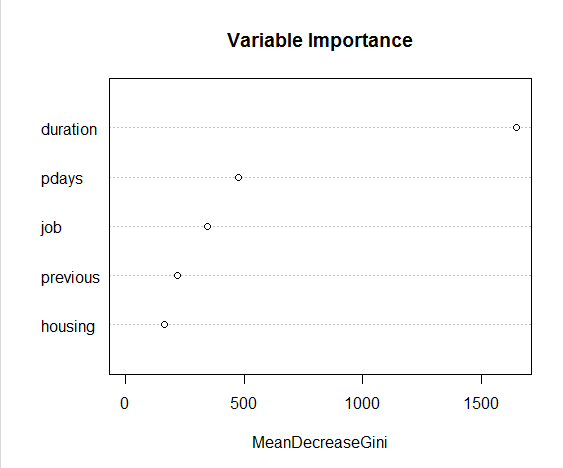
**In our model we are getting 9.74% which is pretty good and low.**

**8. Checking importance of the predictors with respect to the Target**



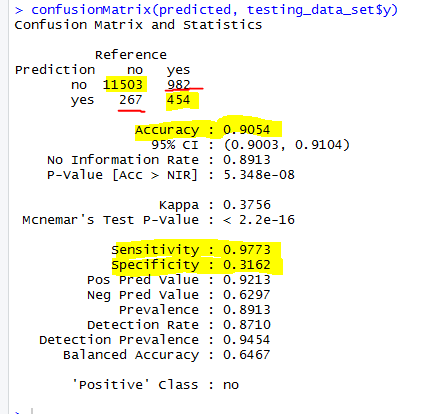
**9. Variable Importance Chart:**

**Top 5 variables are selected and plotted based on Model Accuracy and Gini value.**

****

**10. PART 7 of the code implementing Model on the Testing data set.**

**11. Checking the Confusion Matrix**



**Confusion Matrix:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Predicted Values** |  |  |
|  |  | **NO** | **YES** |
| **Observed Values** | **NO** | **TN : 11503** | **FP: 982** |
|  | **YES** | **FN: 267** | **TP: 454** |

**Sensitivity: (TP)/(TP+FN):Means what percentage is classified by the model predicted class 1.**

**Value we are getting: 0.977**

**Specificity: (TN) /(TN+FP) : It measures true negative rate**

**Value we are getting : 31 %**

**Accuracy: (TP+TN) /(TP+FP+TN+FN)**

**It measures overall accuracy. Which we are getting as 90.1 %.**

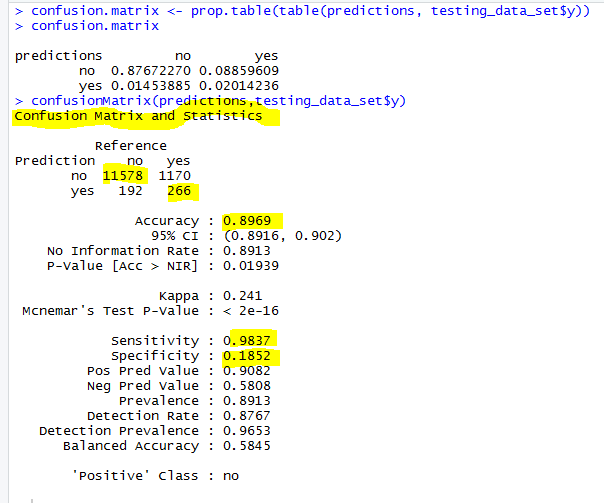
**Result:**

**So the model will predict correctly 91 % of the cases about whether the client will respond to term deposit after sell campaign.**

**DECISION TREE:**

**Here I am going to check the Accuracy of prediction between Random Forest and Decision Tree**

1. **PART 8 of the code has the Decision Tree code, in which we are finding the Accuracy of the Model using Decision Tree Algorithms**



**Result: So Random Forest has a better Accuracy of (91%) than that of Decision Tree**

**having 89%.**

1. **PART 9 of the code has the SVM linear classification method implemented.**