INDIVIDUAL ASSIGNMENT

STATISTICAL AND MACHINE LEARNING

BY: NITHESH RAMANNA

Table of Contents

INTRODUCTION

Machine learning is one of the parts of Artificial Intelligence. Machine learning algorithms learn from the data and make the predictions on data. The Machine learning algorithms' performance increase with experience, which means as the experience of ML increases with more and more data it will make it more effective and efficient.

There are 3 types of machine learning. Supervised, unsupervised and Reinforcement.

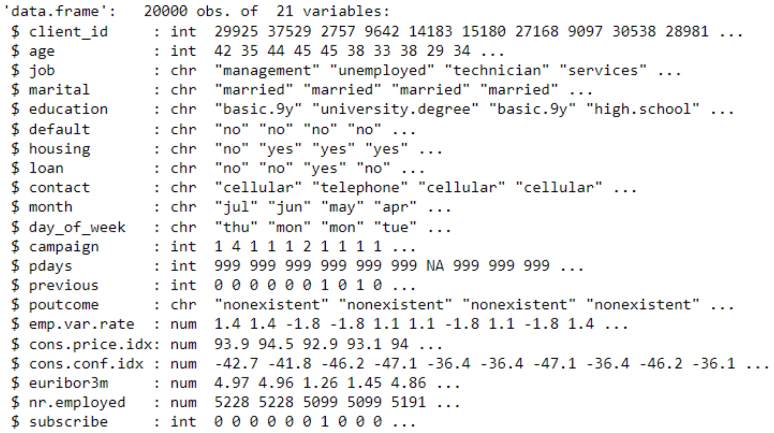
In supervised machine learning model, the model learns from the labeled dataset and generates the predictions for the new datasets as response. There are 2 types of supervised machine learning, classification, and regression. Classification is used predict the discrete values, for example probability of winning game, male or female, subscribed or not subscribed etc. Regression on the other hand is used to predict the continuous values such as salary of the employee, marks, weather prediction etc.

In unsupervised machine learning model, the model learns from the unlabeled dataset and generates the predictions for the new datasets as response. These types of models are used for clustering, anomaly detection etc.

Reinforcement learning is a type of machine learning where the model learns to behave in an environment by performing some actions and analyzing the reactions.

DATA DESCRIPTION

The dataset given is bank marketing dataset. It has 21 variables out of which 'subscribe' is the target variable (0 and 1). The dataset has 20000 observations and out of that 2271 are observations for subscribed (1). Figure 1 shows the structure of the given dataset. The given data set has 10 categorical variables.



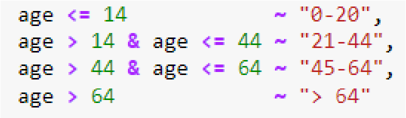
```
'data.frame':	20000 obs. of  21 variables:
 $ client_id     : int  29925 37529 2757 9642 14183 15180 27168 9097 30538 28981 ...
 $ age           : int  42 35 44 45 45 38 33 38 29 34 ...
 $ job           : chr  "management" "unemployed" "technician" "services" ...
 $ marital       : chr  "married" "married" "married" "married" ...
 $ education     : chr  "basic.9y" "university.degree" "basic.9y" "high.school" ...
 $ default       : chr  "no" "no" "no" "no" ...
 $ housing       : chr  "no" "yes" "yes" "yes" ...
 $ loan          : chr  "no" "no" "yes" "no" ...
 $ contact       : chr  "cellular" "telephone" "cellular" "cellular" ...
 $ month         : chr  "jul" "jun" "may" "apr" ...
 $ day_of_week   : chr  "thu" "mon" "mon" "tue" ...
 $ campaign      : int  1 4 1 1 1 2 1 1 1 1 ...
 $ pdays         : int  999 999 999 999 999 999 NA 999 999 999 ...
 $ previous      : int  0 0 0 0 0 0 1 0 1 0 ...
 $ poutcome      : chr  "nonexistent" "nonexistent" "nonexistent" "nonexistent" ...
 $ emp.var.rate  : num  1.4 1.4 -1.8 -1.8 1.1 1.1 -1.8 1.1 -1.8 1.4 ...
 $ cons.price.idx: num  93.9 94.5 92.9 93.1 94 ...
 $ cons.conf.idx : num  -42.7 -41.8 -46.2 -47.1 -36.4 -36.4 -47.1 -36.4 -46.2 -36.1 ...
 $ euribor3m     : num  4.97 4.96 1.26 1.45 4.86 ...
 $ nr.employed   : num  5228 5228 5099 5099 5191 ...
 $ subscribe     : int  0 0 0 0 0 0 1 0 0 0 ...
```

Figure 1

DATA PREPROCESSING

The given dataset has lot of NAs. All the NAs in numerical variables are filled with the average values of the respective columns and all the NAs in categorical variables are filled with the most occurred values in the respective columns.

Variable age is categorized to 4 groups and called the new variable as age_group. The below figure 2 shows the age groups we have after creating the variable.



```
age <= 14              ~ "0-20",
age > 14 & age <= 44 ~ "21-44",
age > 44 & age <= 64 ~ "45-64",
age > 64               ~ "> 64"
```

Figure 2

All the categorical variables are ordinally encoded because many machine learning algorithms cannot work on categorical variables directly. They need all the predictors and target variables to be numeric. Ordinal encoding assigns each categorical values an integer value.

After encoding all the categorical variables, they are dropped and assigned the data with rest of the features to new dataframe.

FEATURE SELECTION

For feature selection, forward stepwise and best subset selection method have been conducted. Both gave the same results. Forward stepwise selection is a stepwise approach that starts with the null model and adds a variable one at a time that improves the performance of the model until the given criterion is met. Best subset selection method aims to find the best subset of independent variables that predicts the best outcome (Dependent variable). This is done by taking the combinations of all independent variable possible.

For this, regsubsets() method for leaps library is used with nvmax = 20. Figure 3 shows the plot Mallow Cp against number of variables. The best subset with lower Mallow Cp values is selected. The model selected 13 variables as best features. Those are as follows,

1. 'pdays'
2. 'previous'
3. 'emp.var.rate'
4. 'cons.price.idx'
5. 'cons.conf.idx'
6. 'euribor3m'
7. 'nr.employed'
8. 'marital_encoded'
9. 'default_encoded'
10. 'contact_encoded'
11. 'month_encoded'
12. 'poutcome_encoded'
13. 'age_group_encoded'

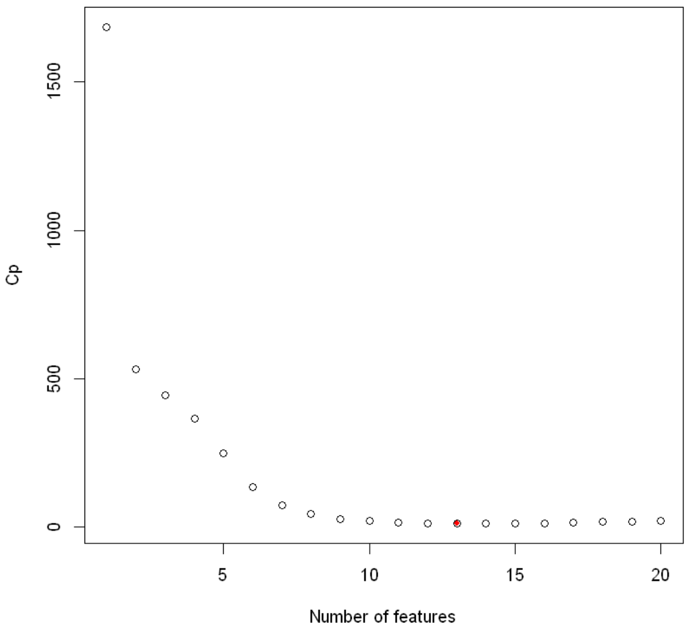


Figure 3

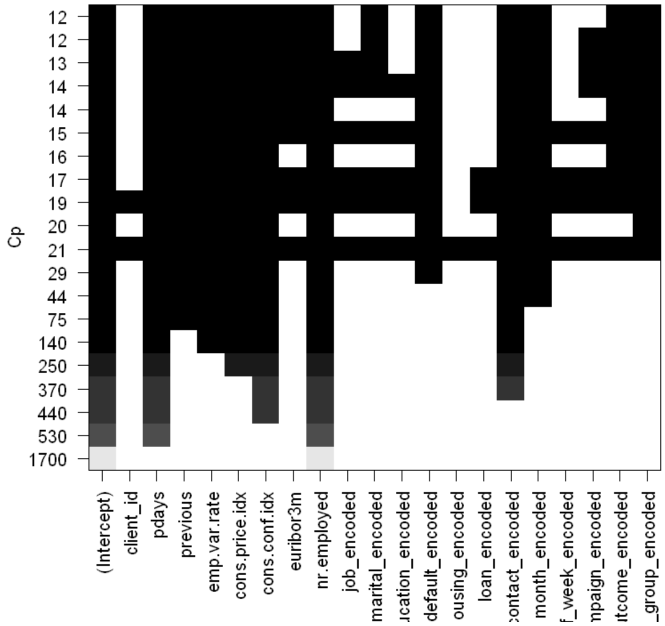
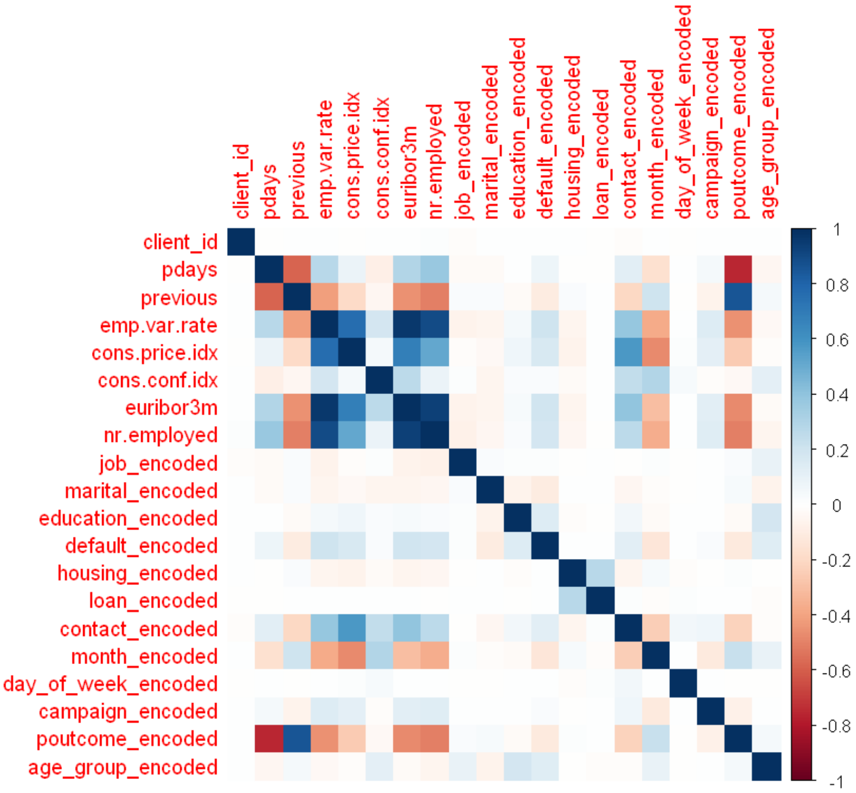


Figure 4

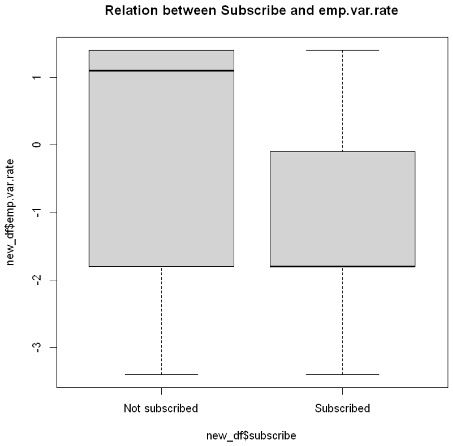
DATA EXPLORATION



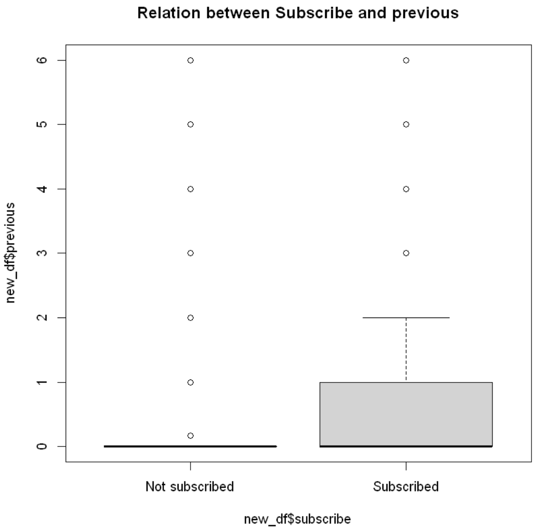
Looking at the above correlation matrix we can see that emp.var.rate has higher positive correlation with cons.price.idx, euribor3m, nr.employed. poutcome_encoded has negative correlation with pdays and positive correlation with previous.

RELATIONSHIP BETWEEN TARGET VARIABLE WITH INDEPENDENT VARIABLES

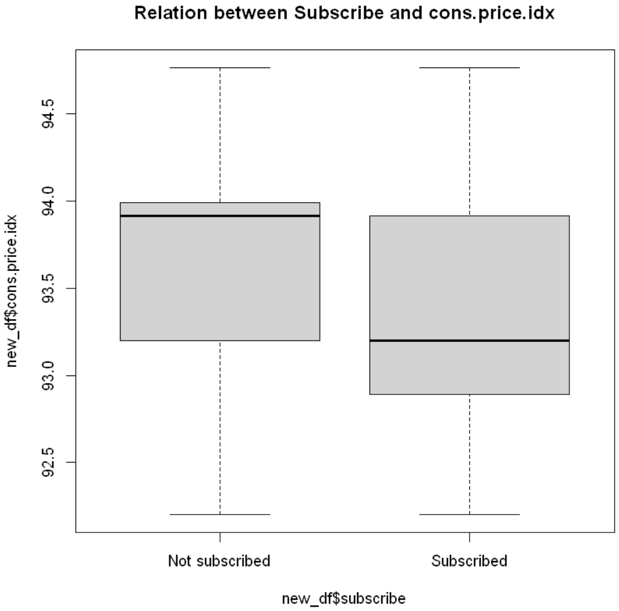
From the figure below we can see that there are more subscription when emp.var.rate is less than zero. For emp.var.rate less than 0 and more than -2 there are more of subscriptions.



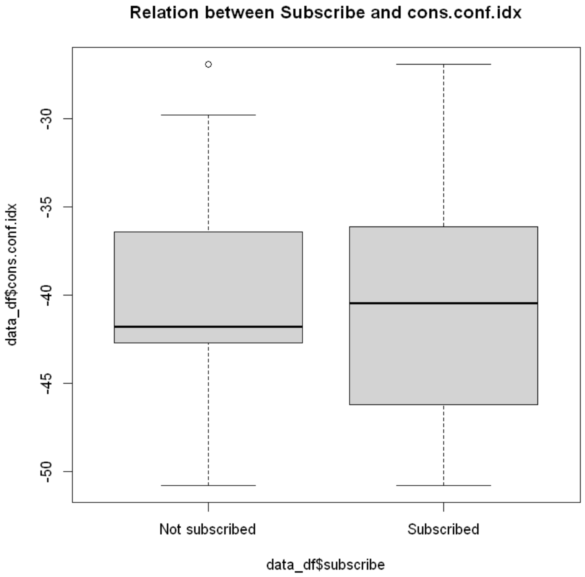
Below figure shows the relationship between subscribe and previous variables. For higher values of Previous there are more subscription. All the non-subscriptions are at previous = 0.



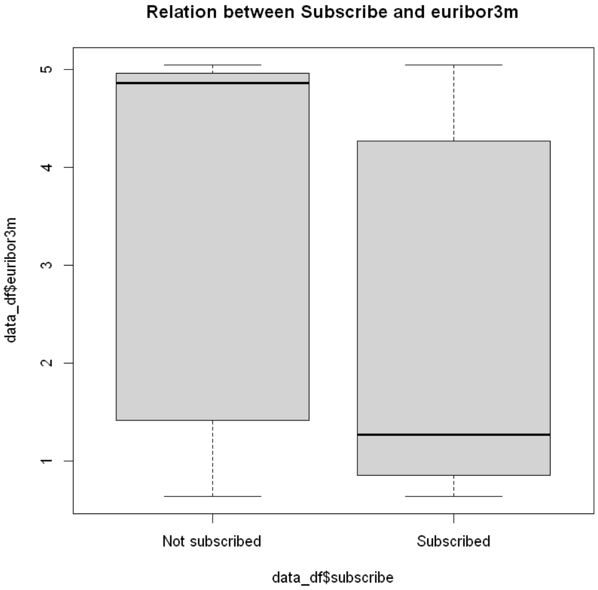
Below figure shows the relationship between subscribe and cons.price.idx. The figure directly doesn’t say how it impacts the subscription since the plot is distributed for both subscribed and not subscribed. But we can see that there are more subscriptions below the cons.price.idx = 93.5.



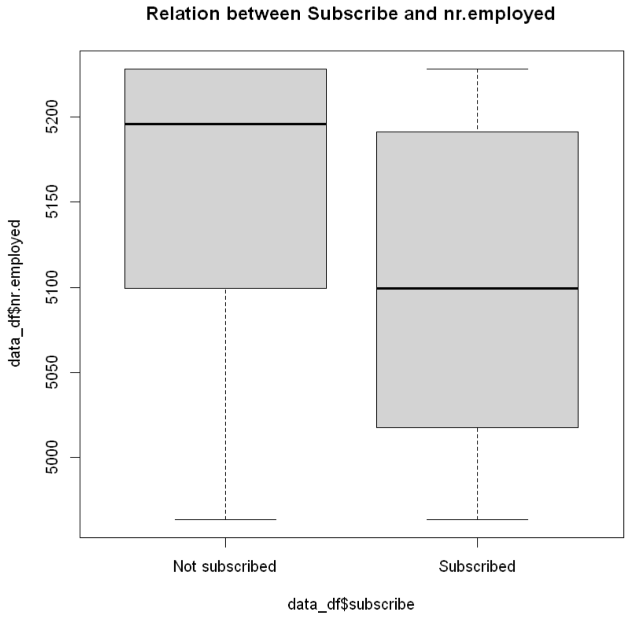
Below figure shows the relation between cons.conf.idx and subscribe. It says that the lower quartile values of cons.conf.idx have a greater number of subscriptions.



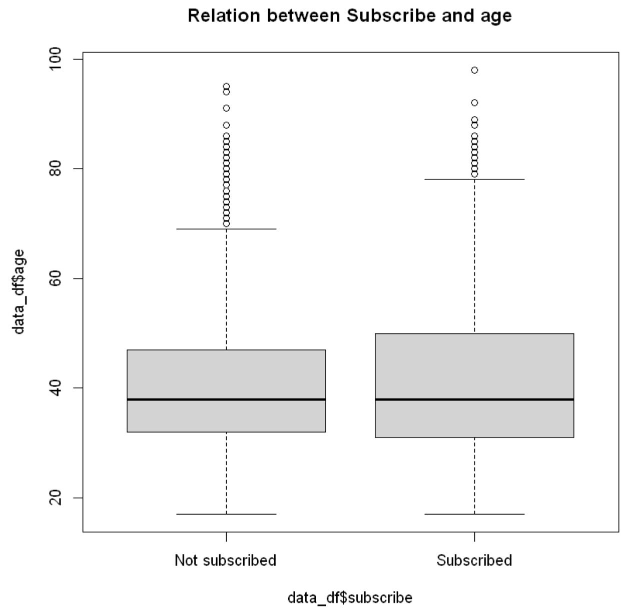
Below figure shows the relation shows the relation between subscribe and euribor3m. In the plot the subscribed and not subscribed are distributed all over. But we can see that all values for euribor3m below median of the subscribed part have the subscription.



Below figure show the relation between nr.employed and subscribe. From the plot we can seen all the values below 5100 for nr.employed have subscriptions.



Below plot shows the relation between age and subscribe. It doesn't show any direct insights as the age for both subscribed and not subscribed classes are equally distributed.



MACHINE LEARNING MODELS

LOGISTIC REGRESSION

Logistic Regression is a statistical machine learning model which is used to find the probability of a certain event or class taking place, for example probability of winning a race, winning a football match, etc. The logistic regression model is mostly used for finding the probability of binary classification.

Advantages:

- Easy to Interpret and efficient to train
- Can be extended to multiple classes
- Very fast at classifying the unknown
- Though its simple provides the good accuracy

Disadvantages:

- More number of predictors may over fit the model
- Can't be used to predict the continuous output.
- Non-linear problems cannot be solved with this model

Objective function of Simple Logistic Regression is below

$$p(X) = \frac{e^{\beta_0+\beta_1}}{1+e^{\beta_0+\beta_1}}$$

where e is Euler's Number (Math Constant), p(X) is the probability of the observation X in class 1. It returns the value in [0,1]. β_0, β_1 are the coefficient of logistic regression model.

Maximum likelihood of Logistic Regression is shown below

$$L(\beta_0,\beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1-p(x_i))$$

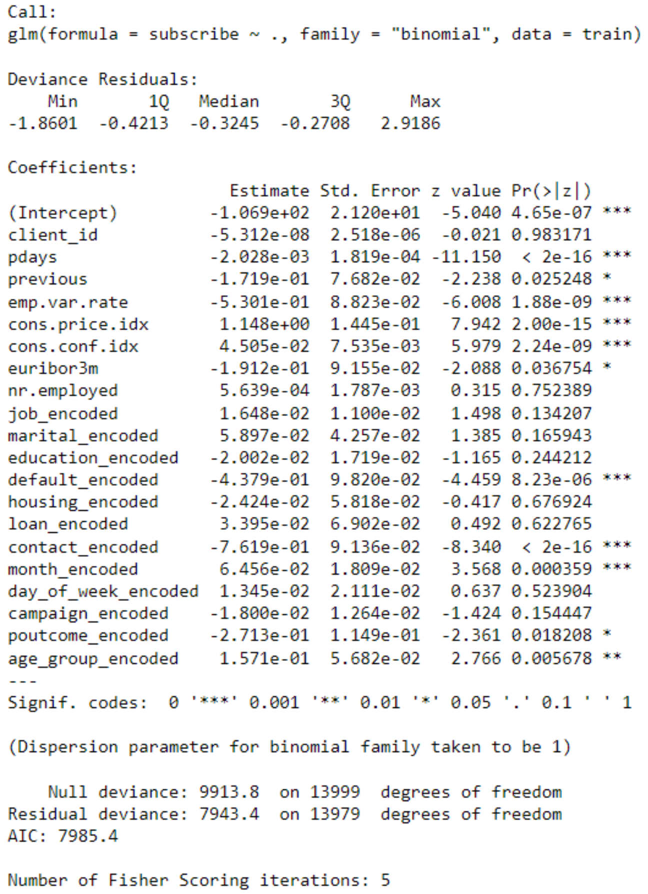
To get the best result for logistic regression, need to pick the best values for β_0, β_1.

To check the AUC of the logistic regression for our given dataset 10-K Cross Validation is conducted with 'subscribe' as target variable and rest all the variables as predictors. The cross validation conducted gave the result of 0.7753709 AUC.

Later the data set is split into train and test set with train size equal to 70% of the data and test set equal to 30% of the data. *'subscribe'* being the target variable, I fitted logistic regression with train set with all the features. It resulted to the following summary.

Figure 5 is the summary of logistic regression with all the features in the data frame. It says features, *pdays, emp.var.rate, cons.price.idx, cons.conf.idx, default_encoded, contact_encoded, month_encoded* are highly significant features as they have lowest p-values. Features, *previous, euribor3m, poutcome_encoded, age_group_encoded* have some significance and rest of the feature have no significance. The model gave the training AUC of 0.7796 and Accuracy of 0.7829.

The model is validated with the test set. The model gave the test AUC of 0.771 and Accuracy of 0.7805. Figure 6 shows the ROC of test for logistic regression and figure 7 shows the confusion matrix of test predictions.



```
Call:
glm(formula = subscribe ~ ., family = "binomial", data = train)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.8601  -0.4213  -0.3245  -0.2708   2.9186  

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)    
(Intercept)        -1.069e+02  2.120e+01  -5.040 4.65e-07 ***
client_id          -5.312e-08  2.518e-06  -0.021 0.983171    
pdays              -2.028e-03  1.819e-04 -11.150  < 2e-16 ***
previous           -1.719e-01  7.682e-02  -2.238 0.025248 *  
emp.var.rate       -5.301e-01  8.823e-02  -6.008 1.88e-09 ***
cons.price.idx      1.148e+00  1.445e-01   7.942 2.00e-15 ***
cons.conf.idx       4.505e-02  7.535e-03   5.979 2.24e-09 ***
euribor3m          -1.912e-01  9.155e-02  -2.088 0.036754 *  
nr.employed         5.639e-04  1.787e-03   0.315 0.752389    
job_encoded         1.648e-02  1.100e-02   1.498 0.134207    
marital_encoded     5.897e-02  4.257e-02   1.385 0.165943    
education_encoded  -2.002e-02  1.719e-02  -1.165 0.244212    
default_encoded    -4.379e-01  9.820e-02  -4.459 8.23e-06 ***
housing_encoded    -2.424e-02  5.818e-02  -0.417 0.676924    
loan_encoded        3.395e-02  6.902e-02   0.492 0.622765    
contact_encoded    -7.619e-01  9.136e-02  -8.340  < 2e-16 ***
month_encoded       6.456e-02  1.809e-02   3.568 0.000359 ***
day_of_week_encoded 1.345e-02  2.111e-02   0.637 0.523904    
campaign_encoded   -1.800e-02  1.264e-02  -1.424 0.154447    
poutcome_encoded   -2.713e-01  1.149e-01  -2.361 0.018208 *  
age_group_encoded   1.571e-01  5.682e-02   2.766 0.005678 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9913.8  on 13999  degrees of freedom
Residual deviance: 7943.4  on 13979  degrees of freedom
AIC: 7985.4

Number of Fisher Scoring iterations: 5
```

Figure 5

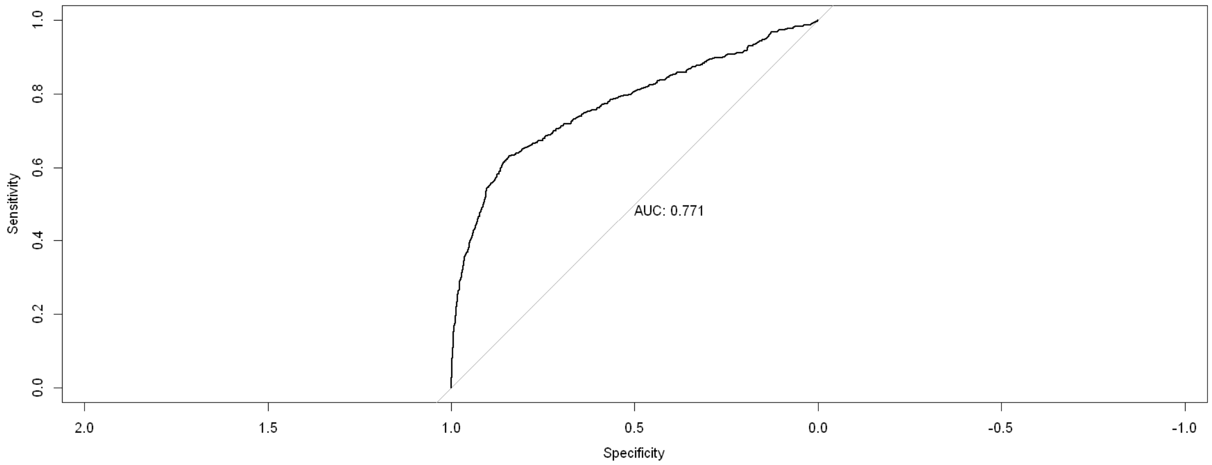
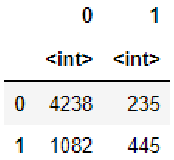


Figure 6



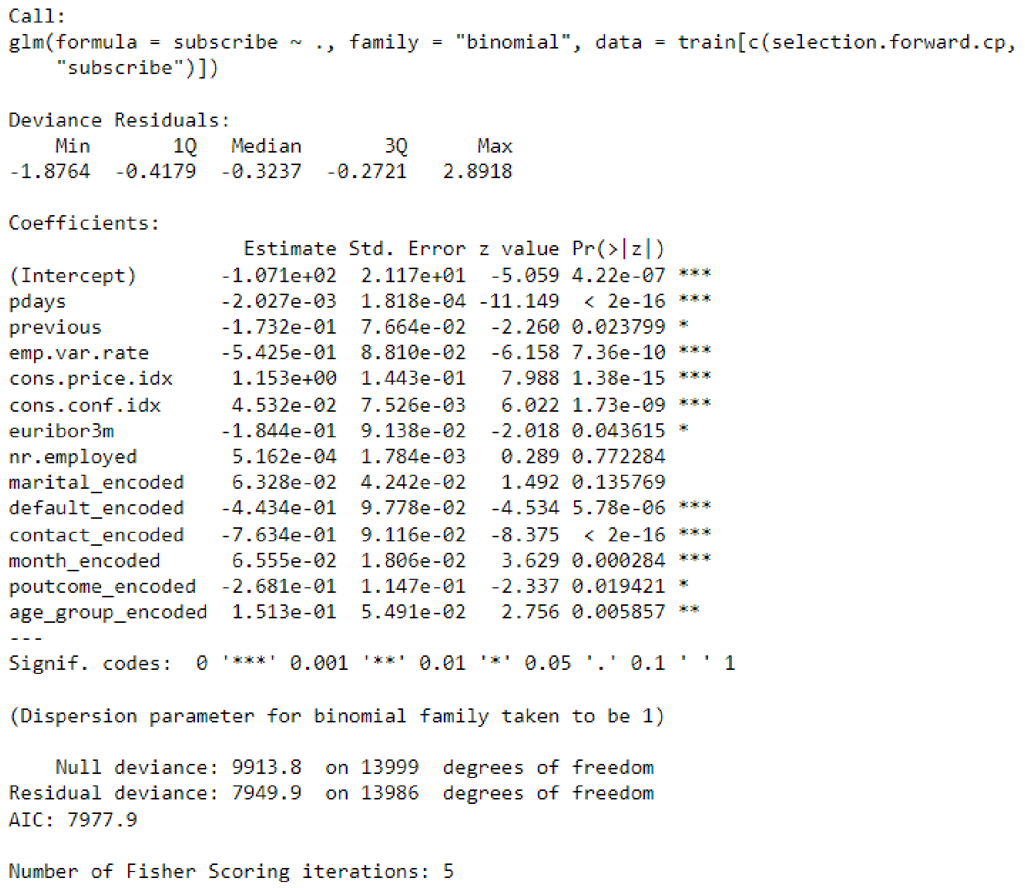
	0	1
	<int>	<int>
0	4238	235
1	1082	445

Figure 7

Figure 7 says that the model predicted 445 subscriptions properly and 4238 as not subscribed properly.

Later the model is fit with the train set by sub-setting it with the features selected using the forward stepwise selection. The summary of the model is as shown in the figure 8. With the chosen features through the forward stepwise selection all the features have significance except *'nr.employed, maritial_encoded'.* The model with selected features has training accuracy of 0.7818 and AUC of 0.7795 which are almost same as the logistic model with all the features.

Later the model is validated with the test set. The model gave the test AUC of 0.771 which is exactly same as the model with all the features and accuracy of 0.7795. Figure 9 shows the ROC of test and figure 10 shows the confusion matrix of the test.



```
Call:
glm(formula = subscribe ~ ., family = "binomial", data = train[c(selection.forward.cp, 
    "subscribe")])

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.8764  -0.4179  -0.3237  -0.2721   2.8918  

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)    
(Intercept)      -1.071e+02  2.117e+01  -5.059 4.22e-07 ***
pdays            -2.027e-03  1.818e-04 -11.149  < 2e-16 ***
previous         -1.732e-01  7.664e-02  -2.260 0.023799 *  
emp.var.rate     -5.425e-01  8.810e-02  -6.158 7.36e-10 ***
cons.price.idx    1.153e+00  1.443e-01   7.988 1.38e-15 ***
cons.conf.idx     4.532e-02  7.526e-03   6.022 1.73e-09 ***
euribor3m        -1.844e-01  9.138e-02  -2.018 0.043615 *  
nr.employed       5.162e-04  1.784e-03   0.289 0.772284    
marital_encoded   6.328e-02  4.242e-02   1.492 0.135769    
default_encoded  -4.434e-01  9.778e-02  -4.534 5.78e-06 ***
contact_encoded  -7.634e-01  9.116e-02  -8.375  < 2e-16 ***
month_encoded     6.555e-02  1.806e-02   3.629 0.000284 ***
poutcome_encoded -2.681e-01  1.147e-01  -2.337 0.019421 *  
age_group_encoded 1.513e-01  5.491e-02   2.756 0.005857 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9913.8  on 13999  degrees of freedom
Residual deviance: 7949.9  on 13986  degrees of freedom
AIC: 7977.9

Number of Fisher Scoring iterations: 5
```

Figure 8

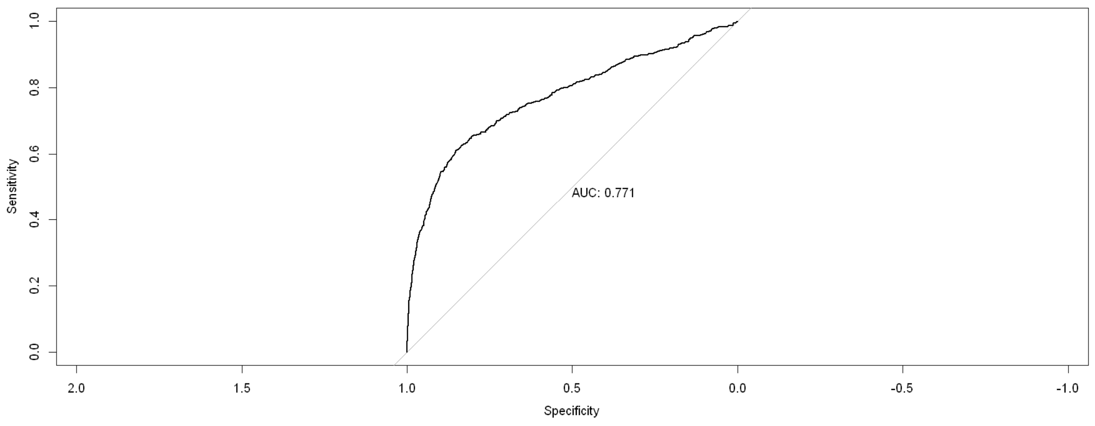
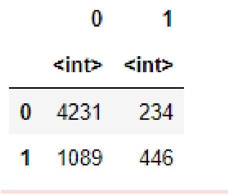


Figure 9



	0	1
	<int>	<int>
0	4231	234
1	1089	446

Figure 10

Figure 10 says the model with selected features predicted 446 subscriptions properly and 4231 as not subscribed which is better than the model with all the features.

RANDOM FOREST

Random forest is a supervised machine learning model. This black box model can be used for both regression problems and classification problems. Random forest consists of several decision tree constructed on different subsets of the data set and the average results of each decision trees are taken to improve the performance of the random forest. The greater number of decision trees in random forest gives the higher accuracy and avoids the over fitting the model.

To predict the category of the new data points, predictions of each decision trees are found and assigned the best category to the new data points based on the maximum number of votes.

Advantages:

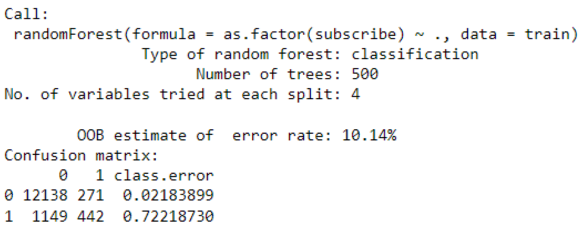
- Missing data doesn't affect the accuracy of the model
- Runs efficiently for large dataset with more number of features.
- Less prone to overfitting

Disadvantages:

- Complex computations
- High computational time

To check the AUC of the Random Forest for our given dataset 10-K Cross Validation is conducted with 'subscribe' as target variable and rest all the variables as predictors. The cross validation conducted gave the result of 0.7775 AUC.

Later the data set is split into train and test set with train size equal to 70% of the data and test set equal to 30% of the data. *'subscribe'* being the target variable, I fitted Random Forest with train set with all the features. It resulted to the following summary.



```
Call:
 randomForest(formula = as.factor(subscribe) ~ ., data = train)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 10.14%
Confusion matrix:
      0   1 class.error
0 12138 271  0.02183899
1  1149 442  0.72218730
```

Figure 11

The above figure gives the summary of the Random Forest. The model used only 4 variables for each split out of all the features. Number of trees constructed is 500. And it gives the Out of Bag error of 10.14%.

The model with all the features gave the train accuracy of 93.48% and Test accuracy of 79.53% and the AUC of train is 99.98% whereas AUC of test is 77.5% which can be seen from the figure 12. Looking at the results we can easily say that the model is overfitting. Figure 13 shows the confusion matrix on test set. It says the model predicted 440 subscribers correctly.

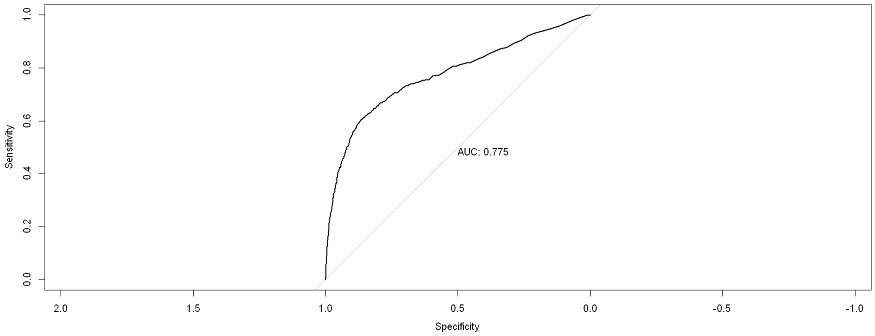
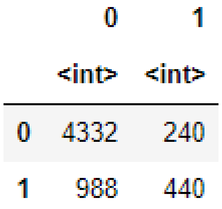


Figure 12



	0 <int>	1 <int>
0	4332	240
1	988	440

Figure 13

Later the model is fitted with train set using the features selected using forward stepwise method with mtry (Number of variables randomly sampled as candidates at each split) = 8 and ntree = 1000. The model gave the training accuracy of 90.4% and test accuracy of 86.01% and train AUC of 82.50% and test AUC of 75.41%, the AUC of the test can be seen in the figure 14. With the model with selected features, we can see that it is not overfitting much when compared with model with all the features. The accuracy and AUC of the test can be improved by tuning the hyperparameters (mtry and ntree).

The figure 15 shows the confusion matrix on test set. It says the model predicted 341 subscribers properly which is less when compared to model with all the features. But it is okay because the model is not over fitting with selected features from forward stepwise method.

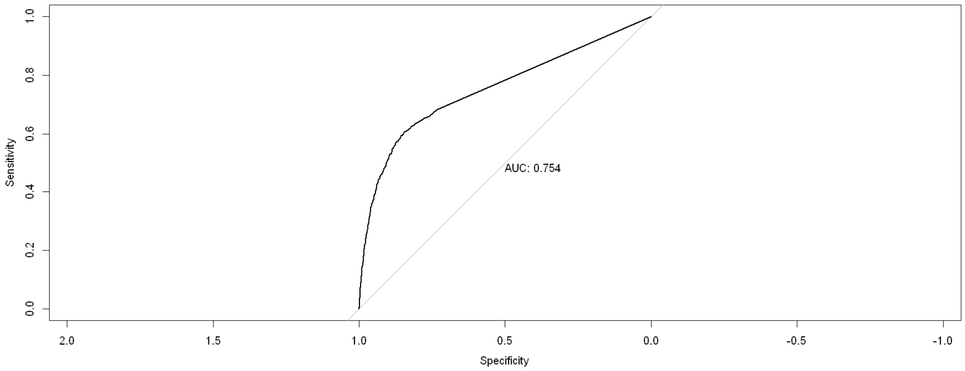
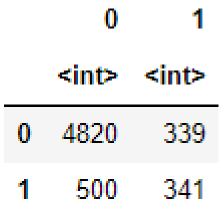


Figure 14



	0	1
	<int>	<int>
0	4820	339
1	500	341

Figure 15

K-NN MODEL

K- Nearest Neighbors model is a simple supervised machine learning model. K-NN can be used for regression as well as classification problems. K-NN is also called as lazy learner algorithm as it doesn’t learn from the training set instead stores the data and performs actions at the time of classification.

K-NN stores the data available and when the new data is encountered, the classification is performed based on the similarity between available data points and new data points.

The model selects the number of neighbors i.e., K and calculates the Euclidean distances of the neighbors. Based on the distance calculated the nearest neighbors are selected and the count of neighbors is taken in each category. When a new data point is encountered, it is assigned to the category with maximum number of neighbors based on similarity.

Advantages:

- Model is simple, so easy to implement
- Effective when training dataset is large
- Robust for noisy data.

Disadvantages:

- Choosing the optimal value for K is complex sometimes
- High Computational cost as the distance needs to be calculated between each data points

To check the mean AUC of the KNN for our given dataset 10-fold cross validation is conducted and it gave the result of 77.53% of AUC.

The model is fitted with train set with all features with k = 5 neighbors and validated with the test set. The model gave the AUC of 66.7% and Accuracy of 66.4%. The figure 16 shows the ROC curve for the test with all the features in the model with auc of 0.667. The figure 17 shows the confusion matrix for the test. It says that the model predicted 403 subscriptions and 3583 no subscriptions properly.

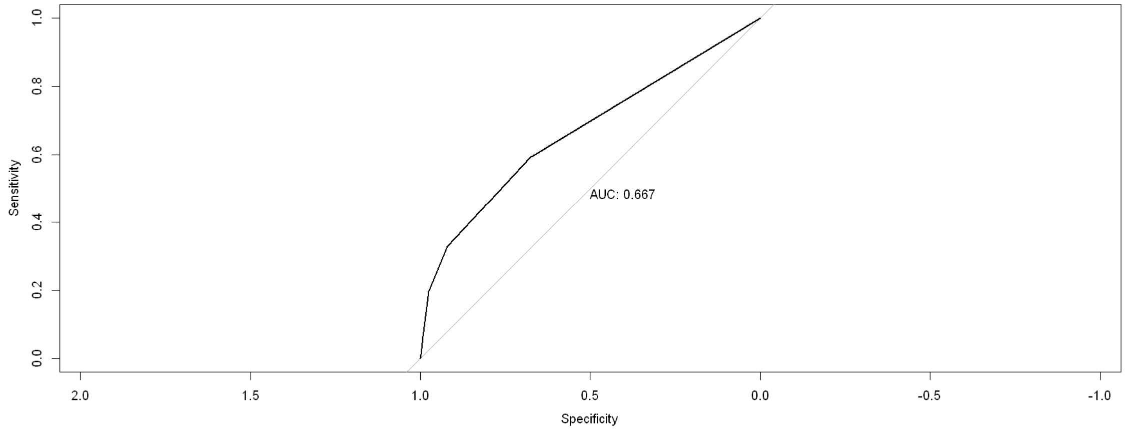
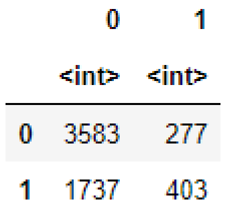


Figure 16



	0	1
	<int>	<int>
0	3583	277
1	1737	403

Figure 17

Later the model is fitted with the train set with the selected features from forward stepwise selection with k = 5 neighbors and validated with test set. The model gave the AUC of 75.3% for test and Accuracy of 78.133%. The figure 18 shows the ROC curve for the test set with the auc of

0.753. The figure 19 shows the confusion matrix for the test. It says the model predicted 434 subscriptions properly and 4254 no subscription properly.

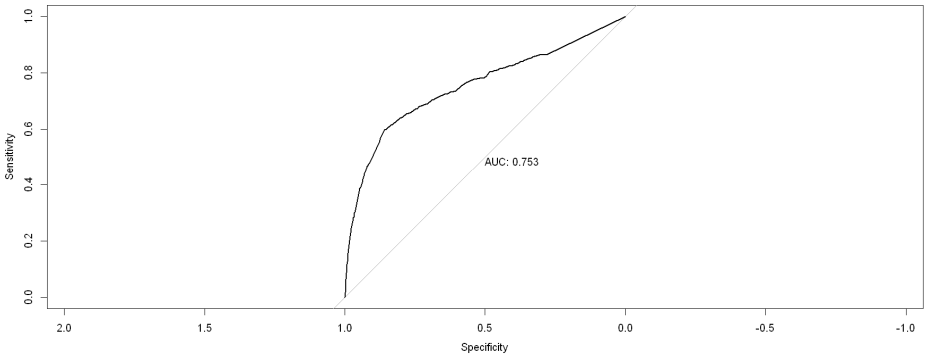
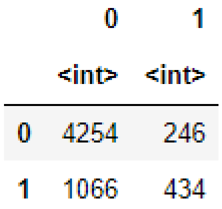


Figure 18



	0	1
	<int>	<int>
0	4254	246
1	1066	434

Figure 19

SUPPORT VECTOR MACHINE

Support vector machine is one of most popular supervised machine learning models. This model can be used for both regression and classification problems, but it is mostly used for classification problems.

SVM creates a decision boundary which can segregate the data points into classes in n-dimensional space. The decision boundary which best categorize the data points is called the Hyperplane. Whenever the new data points are encountered, they are put in the correct category in that n-dimensional space. The hyperplane in SVM is created using the extreme points and these extreme points are Support vectors.

There are 2 types of Linear SVM and Non-Linear SVM. Linear SVM is for data points in 2-dimensional. Non-Linear SVM is for data points in 3 or more dimensional.

Advantages:

- SVM creates a clear margin separating the data points into classes.

- Effective for high dimension spaces.
- Efficient memory.
- Effective when number of dimensions greater than the number of samples.

Disadvantages:

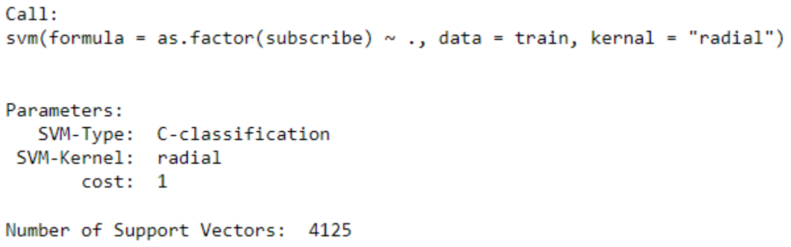
- Not suitable for large data sets.
- When data set has more noise, the SVM doesn't perform very well.

The support vector classifier function is shown below

$$f(x) = \beta_0 + \sum_{i=1}^{n} \alpha_i(x, x_i)$$

where $\beta_0$and α_i are parameters of the model, α_i is non-zero when x_i is support vector and (x, x_i) are the inner product of vector observation x and x_i

The data set is split into train and test set. The model is fitted with all the features and with default hyperparameters and kernel is set to radial. The summary of the model is as show below in figure 20.



```
Call:
svm(formula = as.factor(subscribe) ~ ., data = train, kernal = "radial")


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  4125
```

Figure 20

The model is validated with the test set. The model gave the accuracy of 89.93 % with the AUC of just 59.8%. Figure 21 shows the ROC for the test set and figure 22 shows the confusion matrix. Figure 22 says the model predicted 142 subscriptions and 5254 no subscriptions properly.

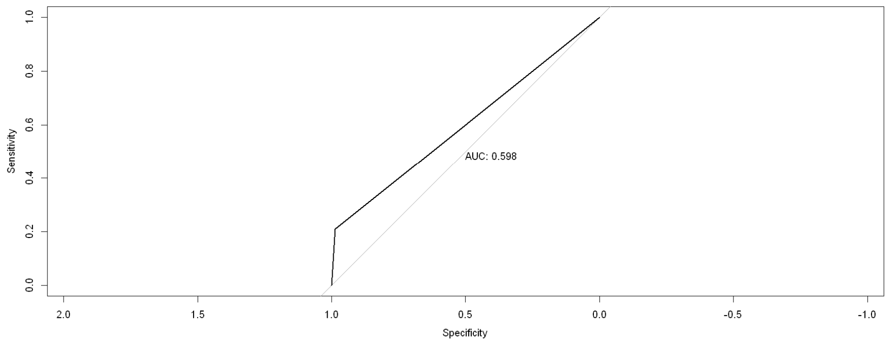
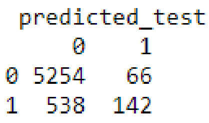


Figure 21



```
  predicted_test
      0    1
0 5254   66
1  538  142
```

Figure 22

Later the model is fitted with train set with the features selected through forward stepwise selection with radial kernel and default hyperparameters. The model is evaluated with test set with same number of features which gave 89.95% accuracy with AUC of 60.8%. The result of the second model with selected features is slightly better than first model. The ROC curve of the model is shown in figure 23 and figure 24 shows the confusion matrix.

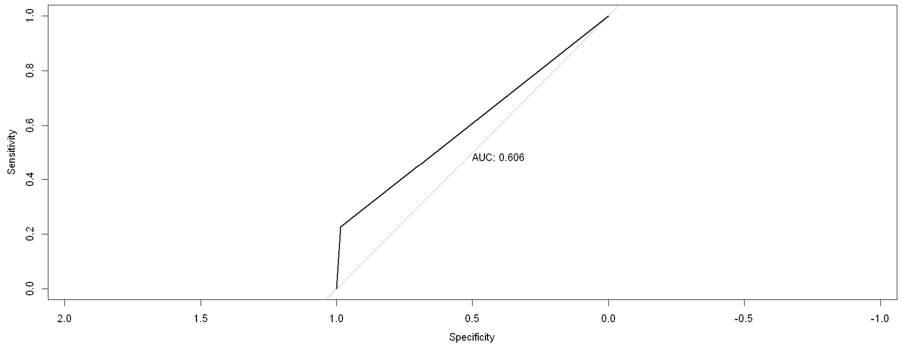
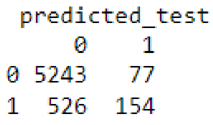
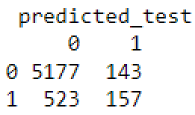


Figure 23



```
  predicted_test
      0    1
0 5243   77
1  526  154
```

Figure 24



```
  predicted_test
      0    1
0 5177  143
1  523  157
```

Figure 25

Figure 25 shows the confusion matrix for the model with cost parameter is equal to 100 and gamma is equal to 0.2. With the best combinations of Cost and gamma, the performance of the model can be increased.

DECISION TREE

Decision Tree is a supervised machine learning model which is used to perform classification or regression analysis, but it is mostly used to perform the classification analysis. It is a tree structured classification in which internal nodes are the features of the dataset used and decision rules are the branches of the tree and outcome is the leaf node of the tree.

There are 2 kinds of nodes in decision tree, decision nodes and leaf nodes. Decision nodes makes the decisions and have many branches associated with them and leaf nodes are the output nodes which have no branches.

The algorithm starts with the root node which represents the whole dataset. Later the best attribute or feature is selected from the data set using the Attribute Selection Measure and then it divides the root node into subsets with all possible values of the attribute selected and decision nodes are created. This process is continued until the stage is reached where it can't further classify the nodes and those nodes are output nodes and called as leaf nodes.

Advantages:

- It doesn't require much data preprocessing
- There is no need for normalized data
- Missing values doesn't affect the model much

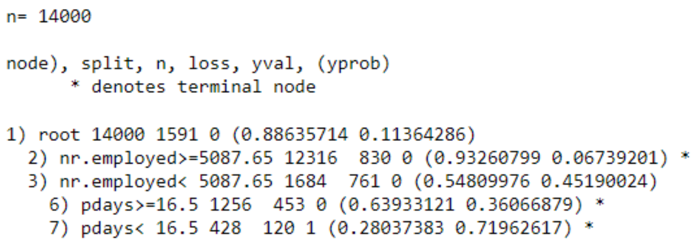
Disadvantage:

- Less flexible because a small change in data makes the model unstable
- Higher time to train the model

To check the AUC of the Decision Tree Classification for our given dataset 10-K Cross Validation is conducted with 'subscribe' as target variable and rest all the variables as predictors. The cross validation conducted gave the result of 0.7045370 AUC.

Later the data set is split into train and test set with train size equal to 70% of the data and test set equal to 30% of the data. *'subscribe'* being the target variable, I fitted Decision Tree with train set with all the features. It resulted to the following summary.

The figure 26 gives the summary of decision tree with all the features. Out of all the features the decision tree used only features in the tree, nr.employed and pdays. Figure 27 shows the dendrogram of the model. It can be seen that if nr.employed >= 5087.65, the model classify that data point as 0 i.e no subscription and if nr.employed < 5087.65 the model considers pdays to classify. In that case if pdays >= 16.5 the model classifies it as 0 else it classifies the data point as 1, i.e., subscribed.



```
n= 14000

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 14000 1591 0 (0.88635714 0.11364286)
  2) nr.employed>=5087.65 12316  830 0 (0.93260799 0.06739201) *
  3) nr.employed< 5087.65 1684  761 0 (0.54809976 0.45190024)
    6) pdays>=16.5 1256  453 0 (0.63933121 0.36066879) *
    7) pdays< 16.5 428  120 1 (0.28037383 0.71962617) *
```

Figure 26

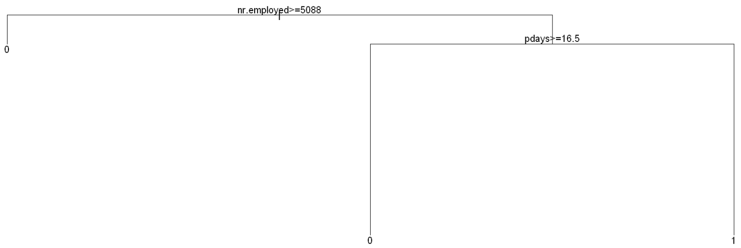
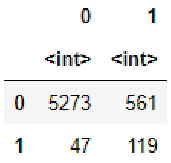


Figure 27

Figure 28 shows the confusion matrix for the test set. It says the model predicted 119 subscriptions and 5273 no subscription properly. Stating that the model gave the accuracy of 89.86% with the AUC of 69.8%. The ROC curve for the test set is shown in figure 29



	0 <int>	1 <int>
0	5273	561
1	47	119

Figure 28

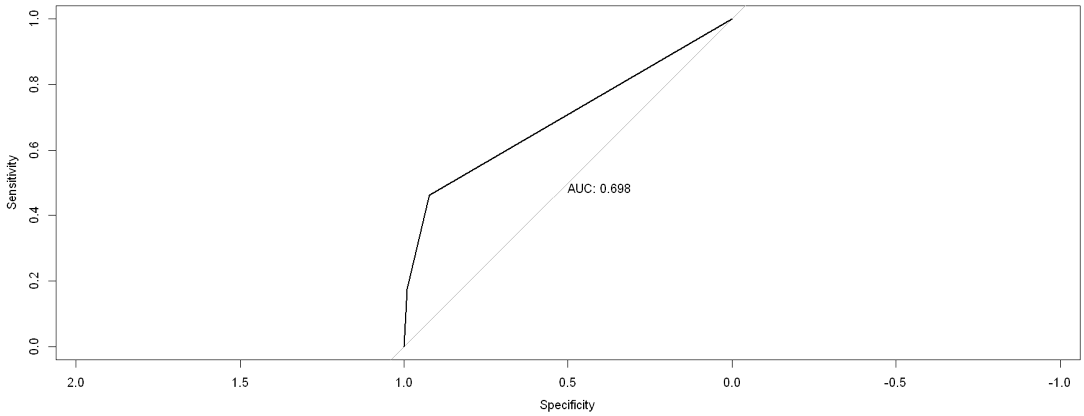


Figure 29

The model is fitted with the features selected using forward stepwise selection. The model neither improved nor deteriorate. The model gave the same out put as the model with all the features.

Later the model is pruned, i.e., the decision tree is allowed to grow with maxdepth of 8 and minsplit of 100. Though the model is pruned it gave the same result as before.

CONCLUSION

Out of all 5 models used, Logistic Regression and K-NN gave the best accuracy, in terms of True positive rate and AUC. Both the models gave the accuracy of around 78% with AUC of 77.1% and 75.3% for Logistic Regression and K-NN respectively.

REFERENCES:

- Study Material by professor
- Machine Learning Tutorial | Machine Learning with Python - Javatpoint