1. DATA DESCRIPTION

Abstract

The bank marketing dataset consists of information related to personal details of clients of a Portuguese bank, marketing campaign (phone calls) details employed by the bank and other social and economic attributes. There are 20 input variables of both numeric and categorical types and one binary output variable. The goal of this report is to determine the best classification model to be employed which can accurately predict whether a client will subscribe a term deposit or not. All the analysis results provided have been obtained by implementing the models in SAS Enterprise Miner.

Descriptive Analytics and Findings

The initial data exploration of various input and output variables of the bank dataset available in .csv format has been performed. The following observations (Figure 1) have been made regarding data:

- The dataset contains total 21 variables, out of which 10 are interval variables, 10 are nominal input variables and 1 is binary target variable.
- Out of 10 interval variables, 9 are input variables and 1 (duration) is rejected variable. It has been discarded as it does not lead to realistic predictive model as per the metadata provided.
- The target variable (y) has 36548 and 4640 observations for clients taking and not taking term deposit respectively.

Variable Summary

Role	Measurement Level	Frequency Count
INPUT	INTERVAL	9
INPUT	NOMINAL	10
REJECTED	INTERVAL	1
TARGET	BINARY	1

Variable Levels Summary
(maximum 500 observations printed)

Variable	Role	Frequency Count
y	TARGET	2

Distribution of Class Target and Segment Variables
(maximum 500 observations printed)

Data Role=TRAIN

Data Role	Variable Name	Role	Level	Frequency Count	Percent
TRAIN	y	TARGET	no	36548	88.7346
TRAIN	y	TARGET	yes	4640	11.2654

Figure 1

Summary statistics for both class and interval variables has been obtained as can be seen below and following findings (Figure 2) have been observed:

- The maximum number of levels for any class variable is 12. Since it is considered as a normal value which does not pose any problem at later stages during modelling, class variables have been considered without any modification for analysis to be carried out later.
- For any of the interval variables, no missing values have been found.

- As is known that a variable is considered to be normal if the skew and kurtosis values fall within the range of [-2, +2]. It has been observed that the variables campaign, pdays and previous are not normal as their skew and kurtosis values fall outside the standardized range of values.

Class Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Data Role	Variable Name	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
TRAIN	contact	INPUT	2	0	cellular	63.47	telephone	36.53
TRAIN	day_of_week	INPUT	5	0	thu	20.94	mon	20.67
TRAIN	default	INPUT	3	0	no	79.12	unknown	20.87
TRAIN	education	INPUT	8	0	university.degree	29.54	high.school	23.10
TRAIN	housing	INPUT	3	0	yes	52.38	no	45.21
TRAIN	job	INPUT	12	0	admin.	25.30	blue-collar	22.47
TRAIN	loan	INPUT	3	0	no	82.43	yes	15.17
TRAIN	marital	INPUT	4	0	married	60.52	single	28.09
TRAIN	month	INPUT	10	0	may	33.43	jul	17.42
TRAIN	poutcome	INPUT	3	0	nonexistent	86.34	failure	10.32
TRAIN	y	TARGET	2	0	no	88.73	yes	11.27

Interval Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
age	INPUT	40.02406	10.42125	41188	0	17	38	98	0.784697	0.791312
campaign	INPUT	2.567593	2.770014	41188	0	1	2	56	4.762507	36.9798
cons_conf_idx	INPUT	-40.5026	4.628198	41188	0	-50.8	-41.8	-26.9	0.30318	-0.35856
cons_price_idx	INPUT	93.57566	0.57884	41188	0	92.201	93.749	94.767	-0.23089	-0.82804
emp_var_rate	INPUT	0.081886	1.57096	41188	0	-3.4	1.1	1.4	-0.7241	-1.06263
euribor3m	INPUT	3.621291	1.734447	41188	0	0.634	4.857	5.045	-0.70919	-1.4068
nr_employed	INPUT	5167.036	72.25153	41188	0	4963.6	5191	5228.1	-1.04426	-0.00366
pdays	INPUT	962.4755	186.9109	41188	0	0	999	999	-4.92219	22.22946
previous	INPUT	0.172963	0.494901	41188	0	0	0	7	3.832042	20.10882

Figure 2

2. DATA PREPARATION

2.1 Replacement

Variable pdays represents number of days that passed by after the client was last contacted from a previous campaign It has been observed that it contains value=999 which signifies that client was not previously contacted is leading to the problem of skew in the variable distribution. In order to fix this problem, we have replaced the pdays value 999 by -1 (Figure 3) since this value does not actually represent any count of days but only the representation of one of the cases. Such replacement does not make any difference to interpretation of variable but has huge numerical implications for normality of variable. This has been performed using replacement node in SAS Enterprise Miner.

month	day_of_week	duration	campaign	pdays	previous	poutcome	euribor3m	y	emp.var.rate	cons.price.idx	cons.conf.idx	nr.employed	Replacement: pdays
may	mon	261	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	151	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	307	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	139	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	222	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	137	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	293	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	146	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	312	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	440	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	353	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	195	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	38	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	342	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	99	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1
may	mon	93	1	999	0	nonexistent	4.857	no	1.1	93.994	-36.4	5191	-1

Figure 3

2.2 Transformation

Following transformation have been performed in this analysis report using Transform Variable node of SAS Enterprise Miner:

- It has been observed from the metadata that the variable nr.employed represents number of employees in the bank. Since this value can't be a decimal number, it has been transformed into an integer variable Trans_nr_employed using Floor function (Figure 4).
- For the non-normal variables observed namely campaign, pdays and previous, different functions have been applied on them in order to transform them into normal variables.
- Variable campaign has been transformed into Trans_campaign by first applying Logarithmic function to the base 10 to it and then Ceil function to convert it into a normal integer value.
- Variable previous has been transformed into Trans_previous by first applying Natural Logarithmic function to it, then Sine function and Ceil function to convert it into a normal integer value.
- Variable pdays has been transformed into Trans_pdays by first applying Logarithmic function to the base 10 to it and then Int function to convert it into a normal integer value.
- It has been observed that variables Trans_campaign, Trans_previous, Trans_nr_employed and Trans_pdays have skew and kurtosis values within standardized range of [-2,+2] and hence, have been successfully transformed into normal variables (Figure 5).

emp.var.r...	co...	...	nr.employ...	Replace...	Trans_ca...	Trans_nr_...	Tra
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	1	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	1	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	1	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	
1.4	93...	..	5228.1	-1	0	5228	

Figure 4

Interval Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
Trans_campaign	INPUT	0.59277	0.532539	41188	0	0	1	2	0.044786	-1.12536
Trans_nr_employed	INPUT	5166.849	72.32838	41188	0	4963	5191	5228	-1.04677	0.007081
Trans_pdays	INPUT	0.170667	0.376343	1500	39688	0	0	1	1.752514	1.072733
Trans_previous	INPUT	0.189156	0.391667	5625	35563	0	0	1	1.587854	0.521466
age	INPUT	40.02406	10.42125	41188	0	17	38	98	0.784697	0.791312
cons_conf_idx	INPUT	-40.5026	4.628198	41188	0	-50.8	-41.8	-26.9	0.30318	-0.35856
cons_price_idx	INPUT	93.57566	0.57884	41188	0	92.201	93.749	94.767	-0.23089	-0.82804
emp_var_rate	INPUT	0.081886	1.57096	41188	0	-3.4	1.1	1.4	-0.7241	-1.06263
euribor3m	INPUT	3.621291	1.734447	41188	0	0.634	4.857	5.045	-0.70919	-1.4068

Figure 5

2.3 Feature Selection

In order to increase the prediction accuracy of a model, it is highly important to reduce the set of available input variables to only important variables which can make significant contribution. This helps in removing the features which are redundant or irrelevant without leading to much information loss. Since our target variable is binary in nature, we have considered Chi-Square Statistic for feature selection in SAS Enterprise Miner. This rejects the variables for which Chi-Square value is less than the minimum, hence rendering us a subset of variables based on their relative importance to be used for model construction in further analysis. As a result, the important variables obtained (Figure 6) in their order of importance are: Trans_nr_employed, poutcome, month, euribor3m, contact, day_of_week, cons_price_idx, job, education, age and Trans_campaign. Various variables have been rejected for the reasons specified as small Chi-square value.

Variable Selection

Variable Name	Reasons for Rejection ▲	Role
Trans_campaign		Input
Trans_nr_employed		Input
age		Input
cons_price_idx		Input
contact		Input
day_of_week		Input
education		Input
euribor3m		Input
job		Input
month		Input
poutcome		Input
cons_conf_idx	Varsel:Small Chi-square value	Rejected
default	Varsel:Small Chi-square value	Rejected
emp_var_rate	Varsel:Small Chi-square value	Rejected
housing	Varsel:Small Chi-square value	Rejected
loan	Varsel:Small Chi-square value	Rejected
marital	Varsel:Small Chi-square value	Rejected
Trans_pdays	Varsel:Small Chi-square value, Exceed the missing percent of 0	Rejected
Trans_previous	Varsel:Small Chi-square value, Exceed the missing percent of 0	Rejected

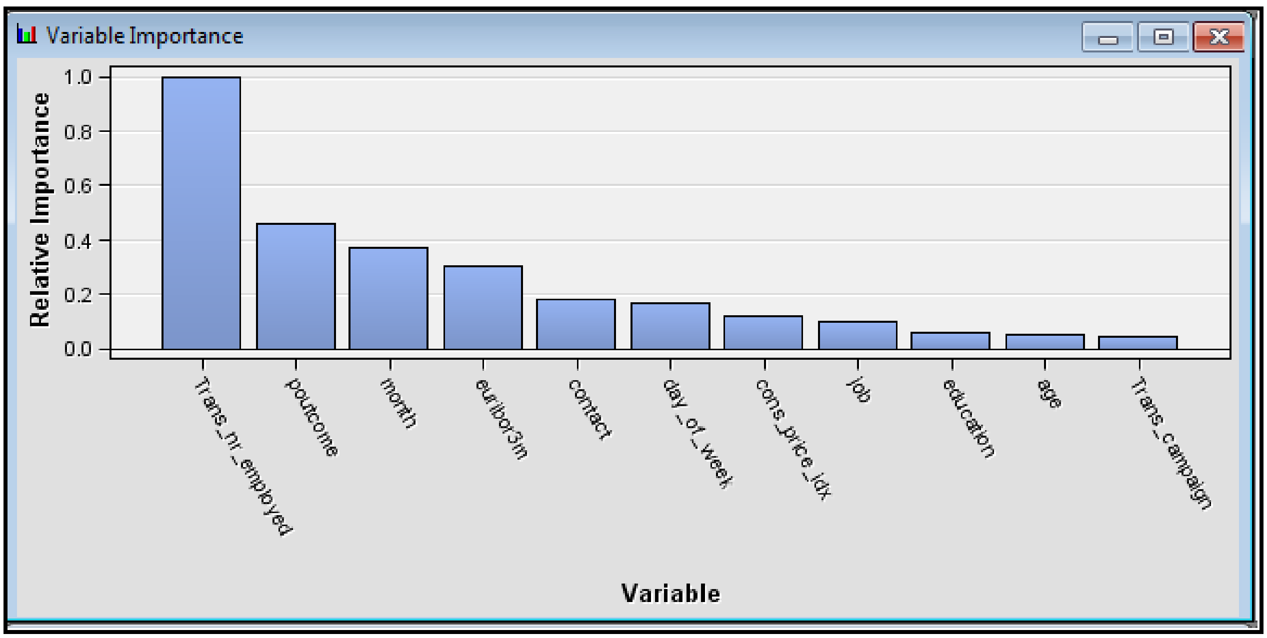


Figure 6

Data Partition

In order to improve classification performance of models, data is usually split into various chunks for training, validating and testing classifiers. As test partition is used mainly for calculating fit statistics after completion of modelling and model selection, it is regarded as wasting data by sub-setting this way by many analysts. Also, by increasing the observations to certain extent in train data, we can improve the model stability. Based on such facts, we have partitioned the data (Figure 7) into train and validate chunks in the ratio of 50:50 using Default Partitioning Method in SAS Enterprise Miner for this analysis. This allows both train and validate sets to contain 20593 observations each.

Summary Statistics for Class Targets

Data=DATA

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
y	.	no	36548	88.7346	
y	.	yes	4640	11.2654	

Data=TRAIN

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
y	.	no	18273	88.7340	
y	.	yes	2320	11.2660	

Data=VALIDATE

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
y	.	no	18275	88.7351	
y	.	yes	2320	11.2649	

Partition Summary

Type	Data Set	Number of Observations
DATA	EMWS1.Varsel_TRAIN	41188
TRAIN	EMWS1.Part_TRAIN	20593
VALIDATE	EMWS1.Part_VALIDATE	20595

Figure 7

3. DATA MINING MODELS AND CONFIGURATION SETTINGS

3.1 Decision Tree

- Initially, a Maximal Decision Tree has been created by training the node to automatically split and generate a tree in SAS Enterprise Miner.
- A Subtree Assessment plot was analyzed for the parameter Misclassification Rate in order to check the performance of the generated tree. It has been observed that the number of leaf nodes generated for the Maximal Tree is 25. The misclassification rate curve is found to be diverging for the train and validate data chunks which implies the poor model performance for the Maximal Decision Tree (Figure 8).
- From the plot, it can be seen that for number of leaves=10, the model for train and validate chunks has minimum Misclassification Rate beyond which there is no further improvement as the curve is either constant or diverging.
- We used Number of Leaves=10 as the configuration setting in order to generate the Optimized Tree interactively based on the Logworth values for variable selection for node splitting (Figure 9).

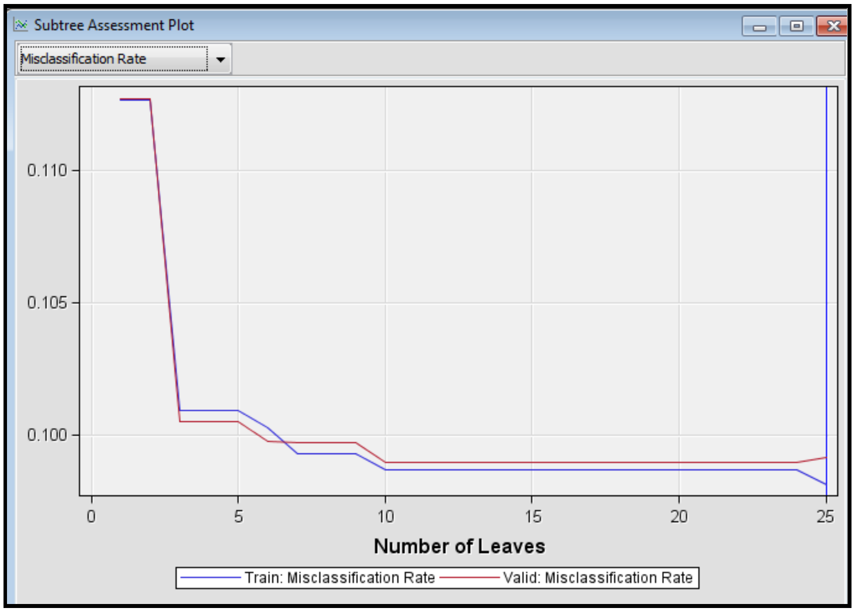


Figure 8

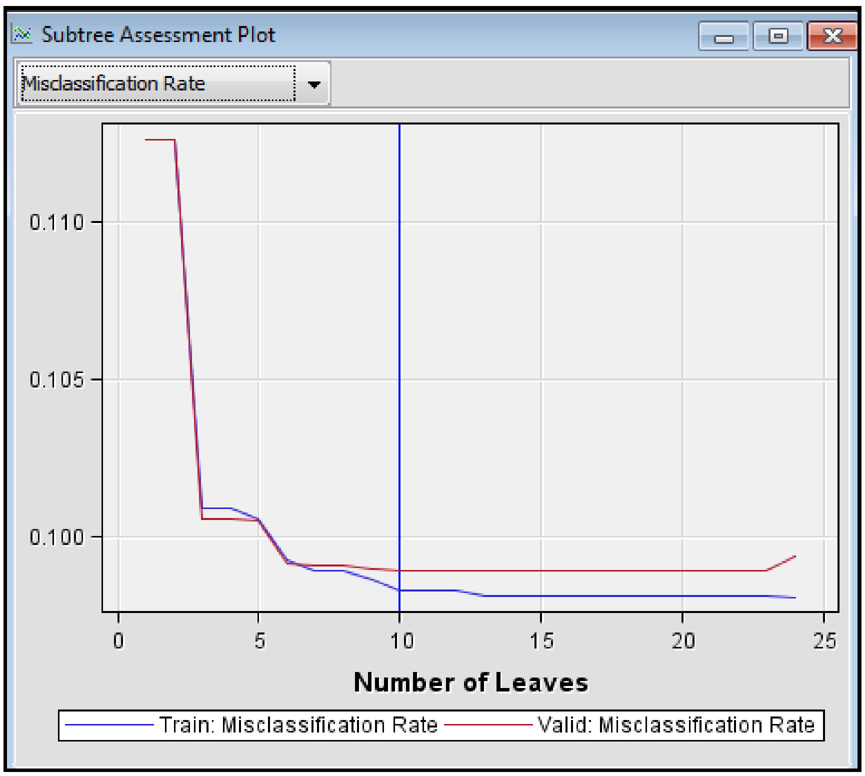


Figure 9

<u>Model Comparison (Maximal Decision Tree vs Optimized Decision Tree):</u>

- Comparison Parameter: Misclassification Rate for Validate chunk: Based on the Fit Statistics (Figure 10), it has been observed that the parameter values for Maximal and Optimized Decision Trees are .098908 and .10017 respectively. This clearly indicates that Optimized Decision Tree identified True Positives and True Negatives more accurately.

- Comparison Parameter: ROC values: It has been observed that Optimized Decision Tree has higher ROC value than Maximal Decision Tree (Figure 10) and hence, is better at predicting the target variable.
- For further model comparison at later stages in this analysis, Optimized Decision Tree has been considered for its better performance.

Fit Statistics

Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Valid: Misclassification Rate	Train: Sum of Frequencies	Train: Misclassification Rate	Train: Maximum Absolute Error
ee2	Optimized Decision Tree	y		0.098908	20593	0.098286	0.946845
ee	Maximal Decision Tree	y		0.10017	20593	0.09814	0.969231

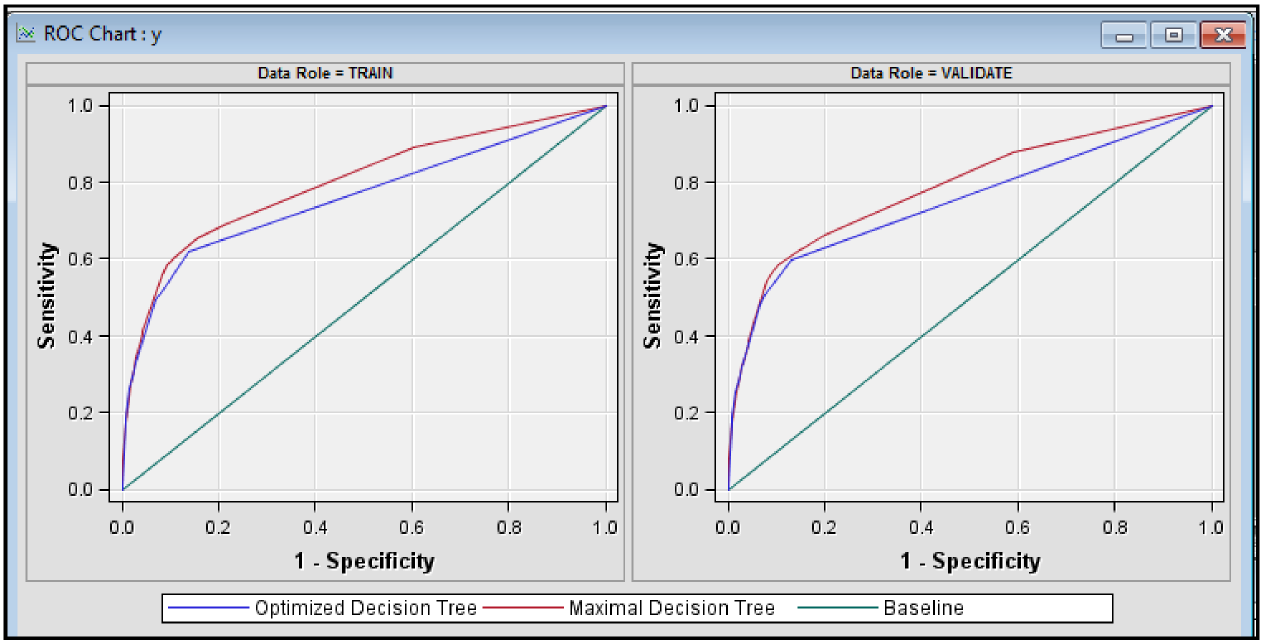


Figure 10

3.2 <u>Regression</u>

- For this analysis, we have performed Logistic Regression as the output variable is categorical. Within this, three different types have been considered namely Forward, Backward and Stepwise and three different models have been generated accordingly for respective regressions.

<u>Model Comparison (Forward vs Backward vs Stepwise)</u>

- Comparison Parameter: Misclassification Rate. As per the Fit statistics (Figure 11), parameter values observed for Forward, Backward and Stepwise Regression are .09978, .09997 and .09978 respectively. This clearly indicates that Stepwise and Forward regression models are better at classifying the true positives and negatives accurately.

- Comparison Parameter: ROC Chart: Since Stepwise Regression has more area under the curve compared to others, it is concluded that Stepwise regression is more accurate at predicting outcome and hence, has been considered for final model comparison to be performed at later stage.

Fit Statistics

lodel Node	Model Description	Target Variable	Selection Criterion: Valid: Misclassification Rate	arget Label	Train: Akaike's Information Criterion	Train: Average Squared Error	Train: Average Error Function
eg	Stepwise Regression	y	0.099782		11484.83	0.078861	0.277979
eg2	Regression-Forward	y	0.099782		11484.83	0.078861	0.277979
eg3	Regression Backward	y	0.099976		11472.28	0.078727	0.277577

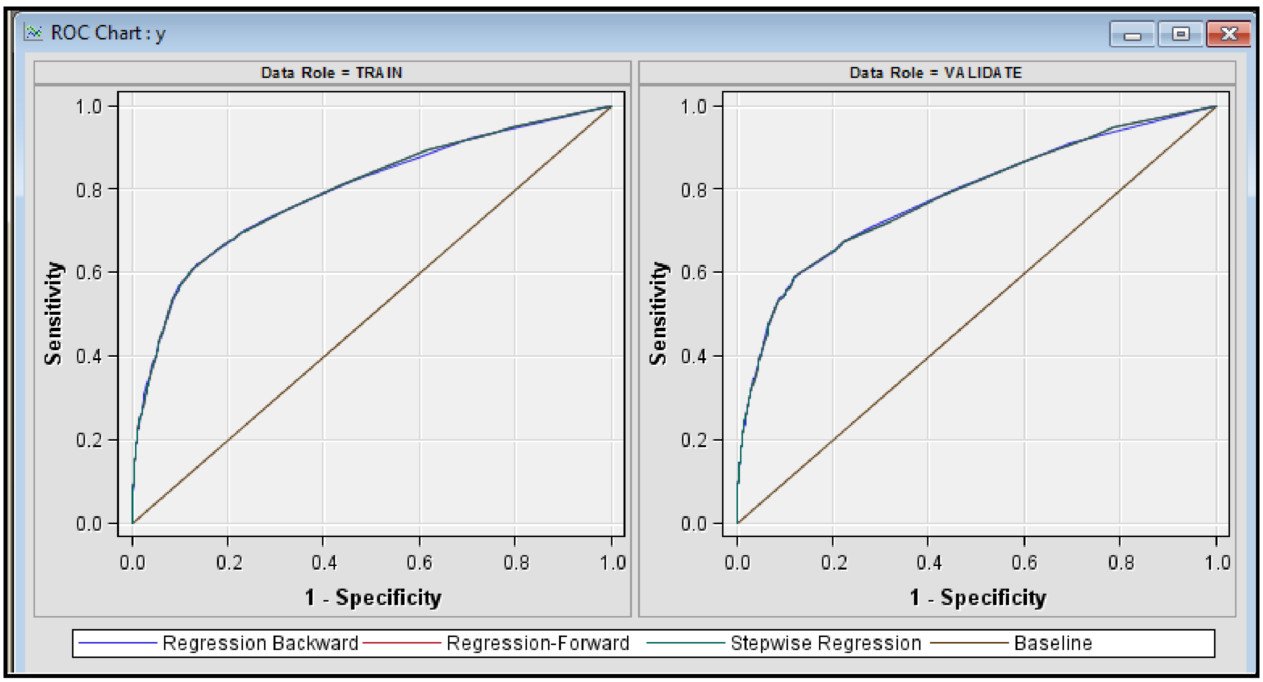


Figure 11

3.3 <u>Neural Network</u>

- Initially, Neural Network was built in SAS Enterprise Miner using the default settings. It was observed that the misclassification rate for both train and validate chunk was high and the difference between the two values was also observed to be more.
- In order to optimize the performance of the Neural Network, it is required to use only subset of input variables i.e. to reduce the number of weights used for classification. We reduced the number of input variables using Stepwise Regression and passed its results to the Neural Network where only important input variables are considered.
- To further improve the performance, we changed the number of hidden units in hidden layer from default 3 to different by hit and trial to get lowest misclassification rate for validate data.

- We found the best results at 9 hidden units by changing the network settings. As can be seen from the Fit Statistics (Figure 12), it is observed that the Misclassification Rate for train and validate chunk is .0973 and .0985 is lesser and also the difference between both the values has been observed to be lower comparatively. Hence, we have used this model for final model selection to be performed in further analysis.

Statistics Label	Train	Validation
Akaike's Information Criterion	11471.99	.
Average Squared Error	0.076584	0.078159
Average Error Function	0.270189	0.277187
Degrees of Freedom for Error	20421	.
Model Degrees of Freedom	172	.
Total Degrees of Freedom	20593	.
Divisor for ASE	41186	41190
Error Function	11127.99	11417.35
Final Prediction Error	0.077875	.
Maximum Absolute Error	0.992584	0.992584
Misclassification Rate	0.097363	0.098519
Mean Squared Error	0.077229	0.078159
Sum of Frequencies	20593	20595
Number of Estimated Weights	172	.

Figure 12

3.4 Random Forest

- Initially, we created a Random Forest with default settings for number of trees and number of leaves and checked the Iteration Plot for the parameter Misclassification Rate. We compared the results for both train and validate data chunks and also for Out of Bag chunk, which a random set of values generated automatically.
- From the Iteration plot (Figure 13), it was observed that the lower values of Misclassification Rate upto .1 were obtained for train and validate chunks for the Number of Trees=60. Also, the Out of Bag chunk also obtained lower values at this count. Even though for Out of Bag the rate improved further for increasing number of trees, there is no improvement in rate for train and validate chunks. Hence, Number of Trees=60 has been considered for generating Optimized Random Forest.
- As the number of input variables considered is 20, the ideal number of leaves to be considered for Optimized Random Forest has been considered by taking the integer value which is nearest to the square root of 20 which comes out to be 4.
- For Optimized Random Forest, configuration settings considered are Number of Trees=60 and Number of Leaves=4 and performance of both trees has been compared.

Model Comparison (HP Forest and Optimized HP Forest)

- Comparison Parameter: Misclassification Rate. As per the Fit statistics (Figure 14), parameter values observed for HP Forest and Optimized HP Forest 9 are .0999 and, .997 respectively. This clearly indicates that Optimized HP Forest model is better at classifying the true positives and negatives accurately.
- Comparison Parameter: ROC Index and ROC Chart: The ROC Index values for HP Forest and Optimized HP Forest are .795 and .797 respectively with Optimized HP Forest having the highest value. Since Optimized HP Forest model has more area under the curve

compared to others and highest ROC index, it is concluded that Optimized HP Forest model is more accurate at predicting outcome and hence, has been considered for final model comparison to be performed at later stage.

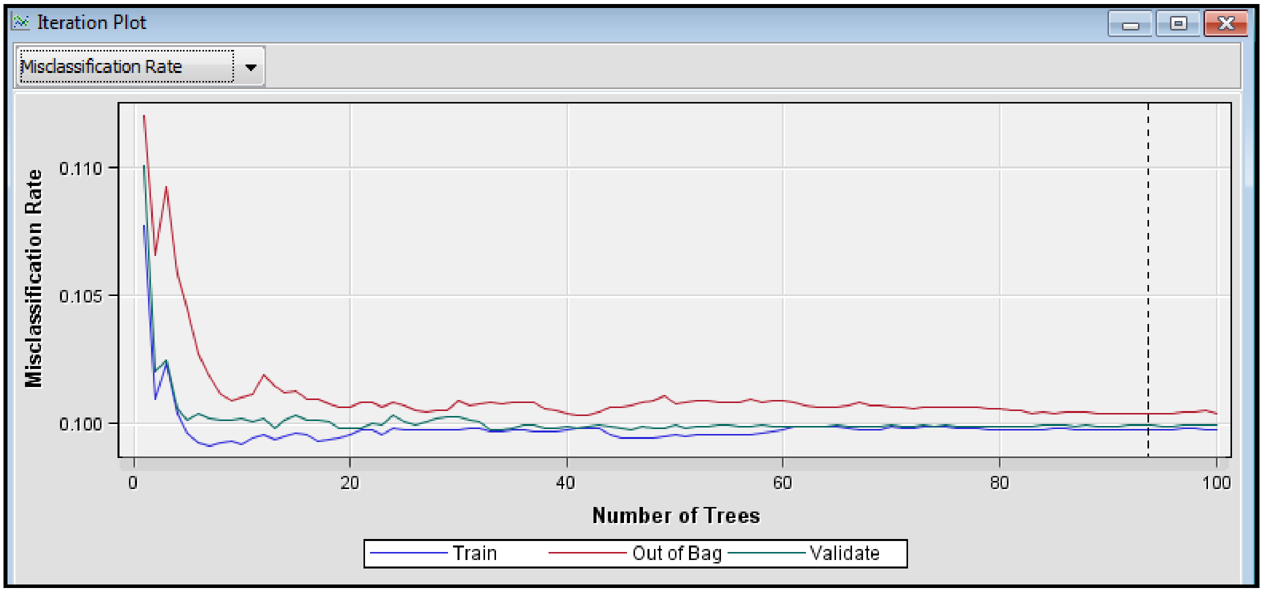


Figure 13

Fit Statistics

Model Node	Model Description	Selection Criterion: Valid: Misclassification Rate	Valid: Roc Index	Valid: Number of Wrong Classifications	Target Label	Train: Average Squared Error	Target Variable
HPDMForest2	Optimized HP Forest	0.099733	0.797	2054		0.075613	y
HPDMForest	HP Forest	0.099927	0.795	2058		0.076381	y

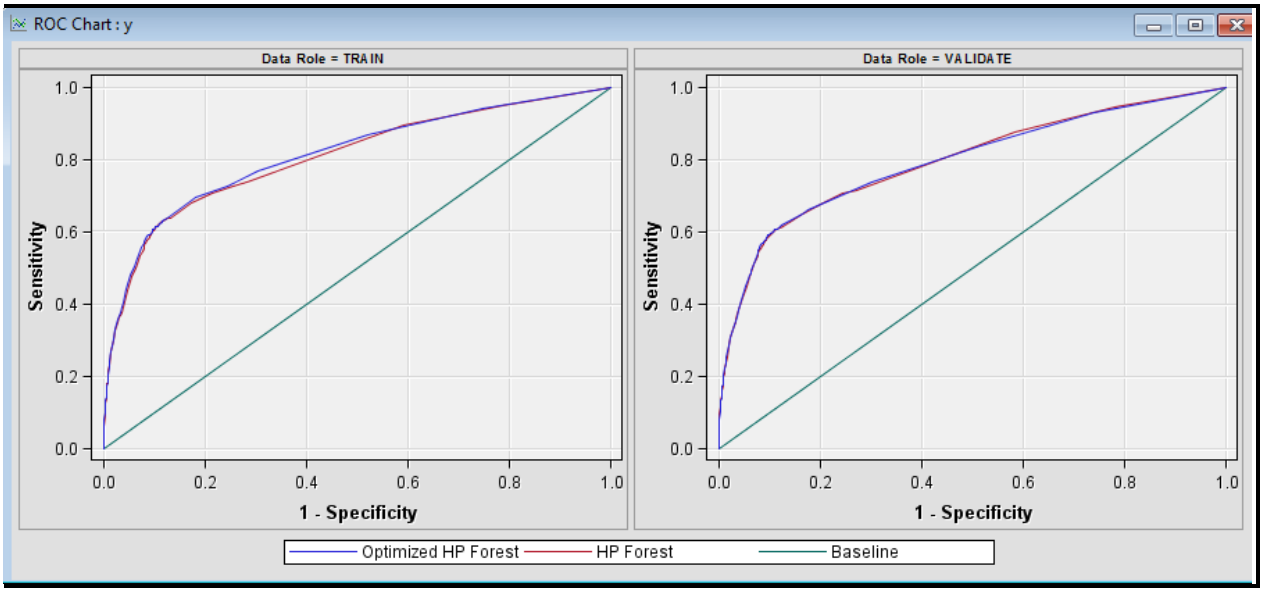


Figure 14

3.5 K-Nearest Neighbor Classification

- In order to determine the configuration settings for the KNN model that give best prediction accuracy, we created four models for the number of nearest neighbors i.e. k as 3, 5, 7, and 9 and analyzed their performance.

Model Comparison (MBR-3, MBR-5, MBR- 7 and MBR-9)

- Comparison Parameter: Misclassification Rate. As per the Fit statistics (Figure 15), parameter values observed for MBR-3, MBR-5, MBR- 7 and MBR-9 are .1194, .1146, .1118 and .1115 respectively. This clearly indicates that MBR-9 model is better at classifying the true positives and negatives accurately.
- Comparison Parameter: ROC Index and ROC Chart: The ROC Index values for MBR-3, MBR-5, MBR- 7 and MBR-9 are .692, .722, .736 and .743 respectively with MBR-9 having the highest value. Since MBR-9 model has more area under the curve compared to others and highest ROC index, it is concluded that MBR-9 model is more accurate at predicting outcome and hence, has been considered for final model comparison to be performed at later stage.

Fit Statistics

Predecessor Node	Model Node	Model Description	Selection Criterion: Valid: Misclassification Rate	Valid: Roc Index ▼
MBR4	MBR4	MBR-9	0.11158	0.743
MBR3	MBR3	MBR-7	0.111872	0.736
MBR2	MBR2	MBR-5	0.114639	0.722
MBR	MBR	MBR-3	0.119446	0.692

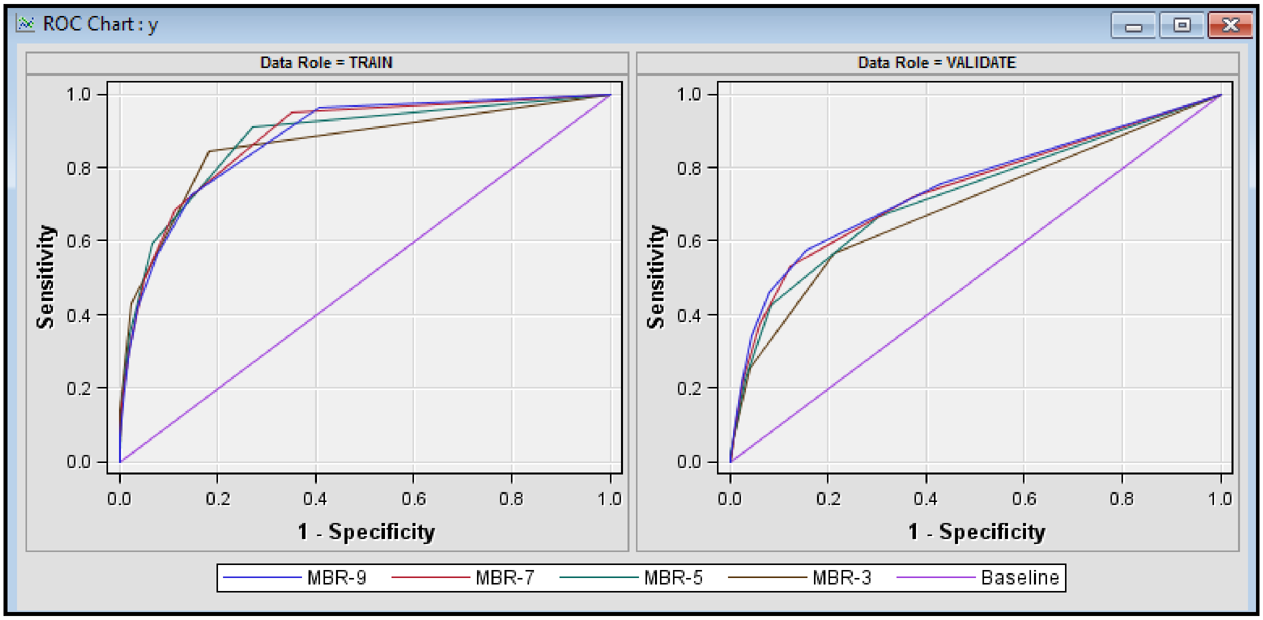


Figure 15

3.6 Support Vector Machine

- In order to determine the configuration settings for the Support Vector Machine model that give best prediction accuracy, we created four models with kernel function as Linear, Polynomial, Radial basis and Sigmoid and analyzed their performances.
- Initially, we tried to perform the model comparison with the entire dataset but due to the large size of data, SVM-Radial basis and SVM-Sigmoid were computationally very time consuming and failed to give results. Hence, we performed 60% sampling of the original dataset and used that for model comparison and obtained the following results (Figure 16).

Model Comparison (SVM-Linear, SVM-Polynomial, SVM-Radial basis and SVM-Sigmoid)

- Comparison Parameter: Misclassification Rate. As per the Fit statistics (Figure 16), parameter values observed for SVM-Linear, SVM-Polynomial, SVM-Radial basis and SVM-Sigmoid are .1011, .1024, .1029 and .1652 respectively. This clearly indicates that SVM-Linear model is better at classifying the true positives and negatives accurately.
- Comparison Parameter: ROC Index: The ROC Index values SVM-Linear, SVM-Polynomial, SVM-Radial basis and SVM-Sigmoid are .774, .679, .725 and .584 respectively with MBR-9 having the highest value. Since SVM-Linear model has the highest ROC index, it is concluded that SVM-Linear model is more accurate at predicting outcome and hence, has been considered for final model comparison to be performed at later stage.

Fit Statistics

Model Description	Selection Criterion: Valid: Misclassification Rate	Valid: Roc Index	Train: Average Squared Error	Train: Divisor for ASE	Train: Maximum Absolute Error	Train: Sum of Frequencies	Train: Root Average Squared Error
HP SVM Linear	0.101149	0.774	0.103599	24710	1	12355	0.321868
HP SVM Polynomial	0.102444	0.679	0.164428	24710	0.809226	12355	0.405497
HP SVM Radial basis	0.102929	0.725	0.069954	24710	0.895262	12355	0.264488
HP SVM Sigmoid	0.165237	0.584	0.1944	24710	1	12355	0.440909

Figure 16

3.7 <u>Naïve Bayes Classification</u>

- In order to determine the configuration settings for the Naïve Bayes model that give best prediction accuracy, we created two models with significance level to be used as cutoff for input variable selection as .2 and .05 and analyzed their performances.
- **<u>Model Comparison (BN-.2 and BN-.05 Models)</u>**
- Comparison Parameter: Misclassification Rate. As per the Fit statistics (Figure 17), parameter values observed for BN-.2 and BN-.05 Models are .1896 and .2026 respectively. This clearly indicates that BN-.2 is better at classifying the true positives and negatives accurately.
- Comparison Parameter: ROC Index As per the Fit statistics (Figure 17), parameter values observed for BN-.2 and BN-.05 Models are .783 and .781 respectively. Since BN-.2 model has more area under the curve compared to others and highest ROC index, this clearly indicates that BN-.2 is more accurate at predicting outcome and hence, has been considered for final model comparison to be performed at later stage.

Fit Statistics

lecessor le	Model Node	Model Description	Selection Criterion: Valid: Misclassification Rate	Valid: Roc Index	Target Variable	Target Label	Train: Average Squared Error
NC	HPBNC	HP BN Classifier-0.2	0.189609	0.783			0.160765
NC2	HPBNC2	HP BN Classifier-0.05	0.202671	0.781			0.16333

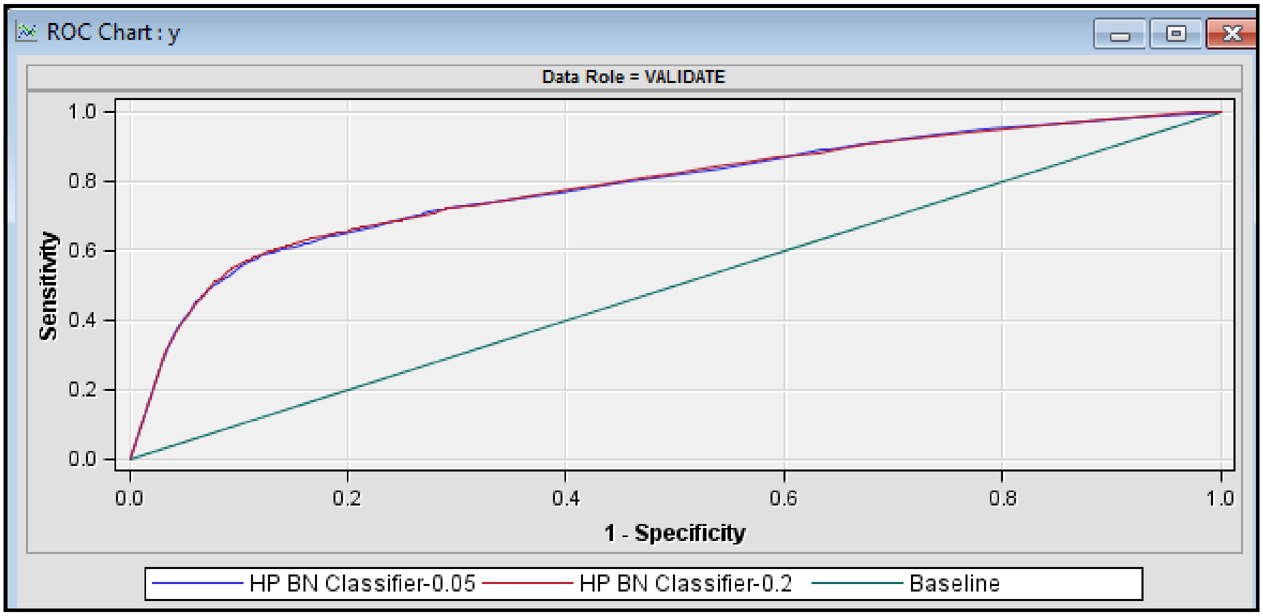


Figure 17

4. Model Results Comparison

- Once the final selection of models has been performed, we compare all the resultant models in order to select the classification model which is best accurately predicting whether a client will subscribe a term deposit or not.

Model Comparison (Neural Network, Optimized Decision Tree, Optimized Random Forest, Stepwise Regression, Linear SVM, KNN-9 and Naïve Bayes Classifier Models)

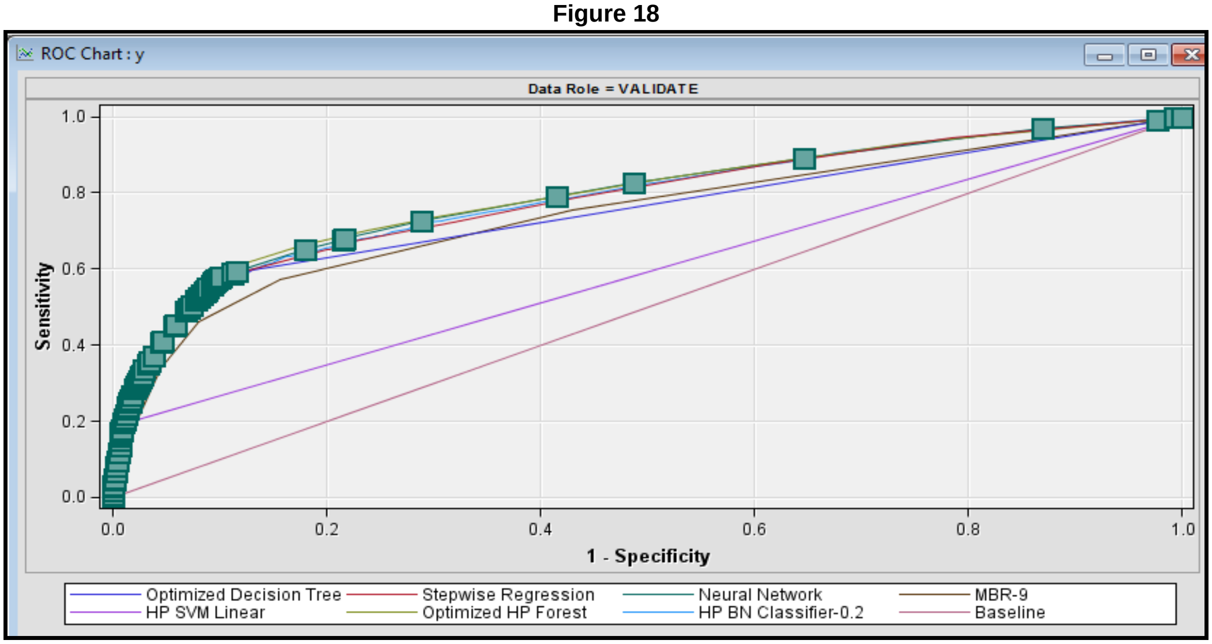
- Comparison Parameter: Misclassification Rate. As per the Fit statistics (Figure 18), parameter values observed for Neural Network, Optimized Decision Tree, Optimized Random Forest, Stepwise Regression, Linear SVM, KNN-9 and Naïve Bayes Classifier Models are .09851, .09890, .9973, .9978, .1026, .1115 and .1896 respectively. This clearly indicates that Neural Network is best at classifying the true positives and negatives accurately.
- Comparison Parameter: ROC Index As per the Fit statistics (Figure 19), parameter values observed for Neural Network, Optimized Decision Tree, Optimized Random Forest, Stepwise Regression, Linear SVM, KNN-9 and Naïve Bayes Classifier Models are .79, .75, .80, .78, .76, .74 and .78 respectively. Since Neural Network and Optimized Random Forest models have more area under the curve compared to others and high ROC index, this clearly indicates that Neural Network and Optimized Random Forest models are more accurate at predicting outcome.
- Comparison Parameter: Number of Wrong Misclassifications. As per the statistics (Figure 19), parameter values observed for Neural Network, Optimized Random Forest, Linear SVM, KNN-9 and Naïve Bayes Classifier Models are 2029, 2054, 2115, 2298 and 3905 respectively. This clearly indicates that Neural Network has misclassified minimum number of observations and hence, has higher accuracy
- Based on the all the comparison results obtained for the above discussed models, it can be concluded that Neural network has the highest accuracy in predicting the outcome variable.

It performed better on all the comparison parameters considered in this analysis. The number of correctly classified cases are 39160 and the accuracy obtained is 95.07%.

Fit Statistics

edecessor ode	Mo	Model Description	Selection Criterion: Valid: Misclassification Rate	Target Variable	Target Label	Train: Akaike's Information Criterion	Train: Average Squared Error	Train: Average Error Function
ural	N...	Neural Network	0.098519	y		11471.99	0.076584	0.270
lComp	T...	Optimized Decision Tree	0.098908	y		.	0.078623	
lComp5	H...	Optimized HP Forest	0.099733	y		.	0.075613	
g	R...	Stepwise Regression	0.099782	y		11484.83	0.078861	0.277
SVM5	H...	HP SVM Linear	0.102695	y		.	0.102462	
lComp6	M...	MBR-9	0.11158	y		10016.16	0.072271	0.242
lComp7	H...	HP BN Classifier-0.2	0.189609	y		.	0.160765	

Figure 18



Data Role=Valid

Statistics	Neural	Tree2	HPDMForest2	Reg	HPSVM5	MBR4	HPBNC
Valid: Kolmogorov-Smirnov Statistic	0.48	0.46	0.49	0.47	0.18	0.42	0.47
Valid: Average Squared Error	0.08	0.08	0.08	0.08	0.10	0.09	0.16
Valid: Roc Index	0.79	0.75	0.80	0.78	0.76	0.74	0.78
Valid: Average Error Function	0.28	.	.	0.28	.	0.39	.
Valid: Bin-Based Two-Way Kolmogorov-Smirnov Probability Cutoff	0.15	0.12	0.15	0.16	0.00	0.22	0.69
Valid: Cumulative Percent Captured Response	44.00	44.22	45.57	43.14	44.05	39.72	43.41
Valid: Percent Captured Response	15.66	16.23	17.47	16.06	17.80	17.21	18.51
Valid: Frequency of Classified Cases	.	.	20595.00	.	20595.00	.	20595.00
Valid: Divisor for VASE	41190.00	41190.00	41190.00	41190.00	41190.00	41190.00	41190.00
Valid: Error Function	11417.35	.	.	11599.76	.	15875.64	.
Valid: Gain	339.87	342.14	355.64	331.33	340.41	297.14	333.95
Valid: Gini Coefficient	0.58	0.50	0.59	0.56	0.52	0.49	0.57
Valid: Bin-Based Two-Way Kolmogorov-Smirnov Statistic	0.47	0.46	0.49	0.46	0.46	0.41	0.47
Valid: Kolmogorov-Smirnov Probability Cutoff	0.16	0.06	0.13	0.14	0.01	0.12	0.65
Valid: Cumulative Lift	4.40	4.42	4.56	4.31	4.40	3.97	4.34
Valid: Lift	3.13	3.24	3.49	3.21	3.56	3.44	3.70
Valid: Maximum Absolute Error	0.99	0.95	0.97	0.98	1.00	1.00	1.00
Valid: Misclassification Rate	0.10	0.10	0.10	0.10	0.10	0.11	0.19
Valid: Mean Square Error	0.08	.	.	0.08	.	0.09	.
Valid: Sum of Frequencies	20595.00	20595.00	20595.00	20595.00	20595.00	20595.00	20595.00
Valid: Root Average Squared Error	0.28	0.28	0.28	0.28	0.32	0.30	0.39
Valid: Cumulative Percent Response	49.55	49.81	51.33	48.59	49.61	44.74	48.88
Valid: Percent Response	35.27	36.55	39.35	36.17	40.10	38.76	41.68
Valid: Root Mean Square Error	0.28	.	.	0.28	.	0.30	.
Valid: Sum of Square Errors	3219.39	3261.39	3176.69	3278.67	4230.00	3631.14	6395.54
Valid: Sum of Case Weights Times Freq	41190.00	.	.	41190.00	.	41190.00	.
Valid: Number of Wrong Classifications	2029.00	.	2054.00	.	2115.00	2298.00	3905.00

Figure 19

5. <u>DISCUSSION</u>

- The model analysis performed so far in this study can be compared with the model considered best in the original research (Moro, Cortez and Rita (2014)).
- For the original research carried out on the same dataset, semi-automatic feature selection was performed on the dataset where some of the features were handpicked. Also, data partition was performed in the ratio of 65:35 for train and validate chunks respectively. Different models namely Decision Tree, Support Vector Machine, Logistic Regression and Neural Network were compared based on the parameter considered as AUC (Area under Curve) and Area of LIFT Cumulative Curve. The models considered were compared by taking different samples of the dataset namely 5, 10, 20, 30, 40, 50, 60 and 70% and the parameter results were considered for evaluation. It was observed in the original research that Neural Network gave the best results for all sample size values for the input dataset.
- For our analysis, we have done feature selection based on Chi-Square statistic value on the dataset. The data partition rule that we considered was 50:50 for train and validate chunks. The parameters considered for evaluation are Misclassification Rate, ROC Index, ROC Chart and Number of Wrong Classifications. We compared the performance of various models namely Neural Network, Optimized Decision Tree, Optimized Random Forest,

Stepwise Regression, Linear SVM, KNN-9 and Naïve Bayes Classifier Models. In our analysis, we found Neural Network having highest accuracy in predicting the outcome variable.

- Based on both our analysis and the original analysis, we can conclude that Neural Network gave best results in both the cases. But it can be said that it might be accidental that we obtained Neural Network as the best model since both the model evaluations were performed under different conditions. It can be possible that Neural Network gives best results under majority of conditions for the dataset considered but the given comparative study is not sufficient enough to comment on this.
- It can be said that there is no way to conclude that one model considered in a particular analysis will be better than the other considered under different analysis. It depends upon different conditions considered for testing model performance since the data cleaning, feature selection, et. al. determine the accuracy of the model.

6. REFERENCE

S. Moro, P. Cortez and P. Rita (2014). *A Data-Driven Approach to Predict the Success of Bank Telemarketing*. Decision Support Systems, Elsevier, 62:22-31, June 2014. Retrieved from: http://media.salford-systems.com/video/tutorial/2015/targeted_marketing.pdf