

PROJECT REPORT

## ANALYTICAL APPROACH TO PREDICT CUSTOMER RESPONSE TO BANK TELE-MARKETING CAMPAIGNS

# ABSTRACT

The banking industry along with various other financial institutions indulge in varying custom- tailored marketing strategies to reach out to the community and provide their various offerings. The various marketing strategies include email marketing, advertisements, digital marketing, telemarketing etc. But all these methods come with a heavy cost associated with them in terms of planning, time, campaign expenses etc. Thus, to overcome the costs/challenges, it is of utmost importance to try and identify the customer/client groups who are more likely to say yes to subscribing to a bank’s offerings. In line with this issue, we have chosen a dataset detailing a marketing campaign carried out by a Portuguese Banking Institution via phone calls from 2008 to 2010 to persuade customers to subscribe to their bank’s term deposit offering. We intend to apply various machine learning techniques like Decision Trees, Logistic Regression, Neural Networks and build models to be able predict customer response to a bank’s telemarketing campaign. We believe that our efforts would enable a bank to identify the right customers to be targeted through their campaign and increase the campaign’s success rate.

# SUMMARY

The objective of the project is to predict the customer response on subscribing for banks term deposit based on the telemarketing campaign. The outcome variable is binary categorical variable, so the data mining techniques used are Decision Tree, Logistic Regression and Neural Network models. Firstly, the decision tree model is built with complete tree and improved parameters from grid search and compared based on their performance. Then the Logistic regression model are built with all predictors and forward selection reduced predictors and compared based on their performance. Finally, the Neural Network model is built with initial parameters and grid search improved parameters to compare their performance. Eventually best models are compared from the above three techniques to analyze different performance parameters. The project concludes by determining the best suited classification technique for the data under study.

### Table of Contents

[ABSTRACT 1](#_TOC_250010)

[SUMMARY 2](#_TOC_250009)

[INTRODUCTION 4](#_TOC_250008)

[DATASET 5](#_TOC_250007)

[DATA PREPROCESSING 6](#_TOC_250006)

[DECISION TREE CLASSIFICATION MODEL 7](#_TOC_250005)

[LOGISTIC REGRESSION 10](#_TOC_250004)

[NEURAL NETWORKS 15](#_TOC_250003)

[FINAL COMPARISON 17](#_TOC_250002)

[CONCLUSION 18](#_TOC_250001)

[APPENDIX 19](#_TOC_250000)

# INTRODUCTION

The bank dataset was obtained from the UCI Machine Learning Repository. The data set has 4k+

records with 21 columns including the outcome variable. The outcome is a binary variable - whether the client subscribed for term deposit or not. First few columns provide personal information details of the client, next few columns provide clients financial information. Furthermore, some columns provide details if client was previously contacted. Finally, some columns specify the market condition during that timeframe. The aim of this project is to build an efficient predictor model based on this information, which would help the bank target customers who have high probability of subscribing for term deposit. Since outcome is a categorical variable

* the different predictor models built are Decision Tree, Logistic Regression and Neural Network. The performance of these models is analyzed, compared, and concluded with best model for this case.

# DATASET

|  |  |
| --- | --- |
| **Variables** | **Description** |
| age | Age of Client |
| job | Type of job that client has |
| marital | Marital status of the client |
| education | Education level of the client |
| default | If client has defaulted in credit before? |
| housing | If client has housing loan? |
| loan | If client has personal loan? |
| contact | Client contact communication type |
| month | Client last contact month of year |
| day\_of\_week | Client last contact day of the week |
| duration | Client last contact duration, in seconds |
| campaign | Number of contacts performed during the current campaign and for the client |
| pdays | Number of days that passed by after the client was last contacted from a  previous campaign |
| previous | Number of contacts performed before current campaign for the client |
| poutcome | Outcome of the previous marketing campaign |
| emp.var.rate | Employment variation rate - quarterly indicator |
| cons.price.idx | Consumer price index - monthly indicator |
| cons.conf.idx | Consumer confidence index - monthly indicator |
| euribor3m | Euribor 3-month rate - daily indicator |
| nr.employed | Number of employees - quarterly indicator |
| y | Has the client subscribed for the term deposits (YES/NO) - Outcome |

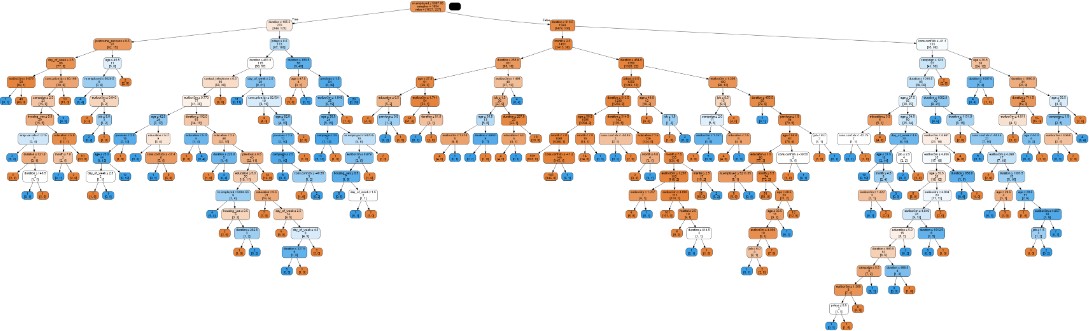
# DATA PREPROCESSING

* 1. **Converted unknown values to NaN:** Converted all the unknown values to NaN in excel with replace function.
  2. **Dropped NaN records:** Dropped all records with values having NaN in any column using dropna() function in Python.
  3. **Removed DEFAULT variable:** The default variable consisted of all NO values except for one YES value. It would not have any impact on the outcome. Hence the column was dropped.
  4. **Convert to Nominal Category:** Assigned nominal values to some categorical variables converting them to integer type. For example, replaced month name with values ranging from 1 to 12 respectively.
  5. **Converting into dummy variables:** Converted few categorical variables like housing, loan, contact, poutcome into dummy variables.
  6. **Pdays variable:** It defines the number of days since the client was last contacted. If client was not previously contacted the value taken is 999, rest all values are in the range of 0 to 21 for the previously contacted clients. The large 999 value has negative dominance over other values. Since there is another variable “previous” which states number of times the client was previously contacted, the pdays variable was converted to binary numeric value, where 0 is for 999 and 1 is for all other values. Now “pdays” will state whether the client was previously contacted or not and “previous” will tell how many times the client was contacted. This also helps to negate the impact of unnecessary large value.

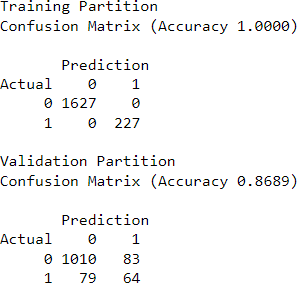
# DECISION TREE CLASSIFICATION MODEL

The outcome variable is categorical, so the first model built is Decision Tree model. The model repetitively splits the records into two subsets to achieve better homogeneity in each subset. The dataset is divided into training (60%) and validation (40%) partition. Initially a tree is built with maximum nodes with full tree.

Complete Tree:



Confusion Matrix with Complete Tree:

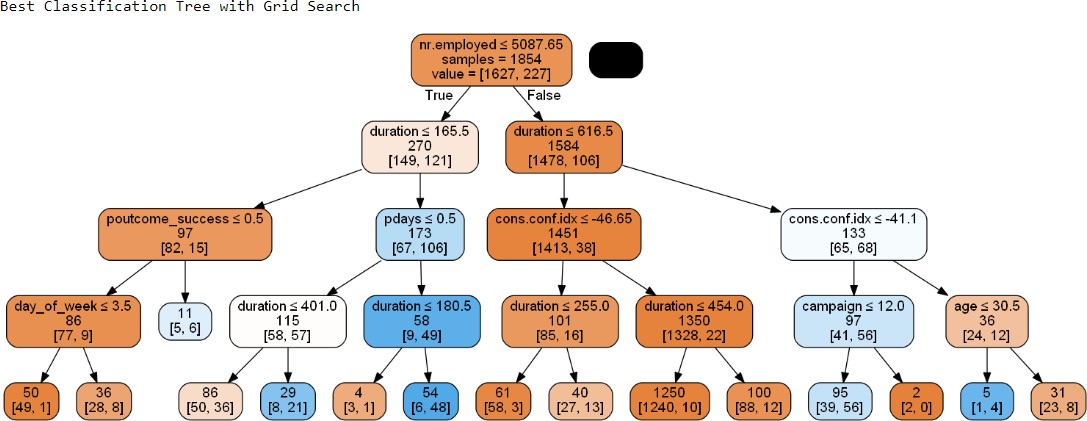


The complete tree has large number of nodes increasing the tree complexity. With complete tree, the accuracy for training matrix is 1.0 and validation matrix is 0.88. So, the complete tree has somewhat possibility of overfitting. To overcome these problems the growth of tree is limited using certain parameters like max\_depth, min\_impurity\_decrease, min\_samples\_split, etc. The GridSearchCV function is used to evaluate the optimal value of these parameters. Certain list values are provided to get initial score and initial values for these parameters. Using these initial values, the range of values is provided to get best score and best values for these parameters.

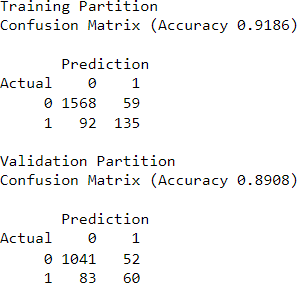
Best parameters using Grid Search function



The new decision tree is built using the best parameters which is far less complex. Improved Decision Tree



Confusion Matrix with Improved Params



Complete Tree vs Improved Decision Tree

|  |  |  |
| --- | --- | --- |
| **Model** | **Complete Tree** | **Improved Decision Tree** |
| **Complexity** | High | Low |
| **Validation Accuracy** | 0.8689 | 0.8908 |

The improved decision tree provides better validation accuracy with lower complexity. So, the improved decision tree is better model than complete tree.

# LOGISTIC REGRESSION

Next model built for the bank dataset is Logistic Regression. The main idea in logistic regression is to develop a function of the predictor variables that relates them to either zero or one outcome. Firstly, the binary categorical outcome is converted to binary numerical variable assigning zero value for one class (Not subscribing for term deposit) and one value for the other (Yes-subscribing for term deposit). Then for each record the probabilities of zero or one outcome are computed. Since outcome is binary, if the probability for zero class is more than 0.5 then the record is classified as zero and vice versa.

Initially all predictors are considered to build the logistic model. The Logit equation based on all predictors is:

**Logit** = - 0.001 – 0.011(Age) + 0.021(Job) + 0.108(Marital) + 0.011(Education) - 0.066(Month)

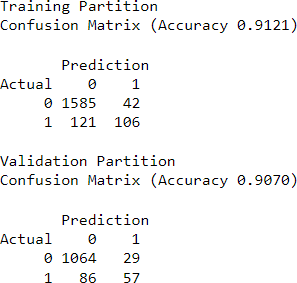
- 0.005(day\_of\_week) + 0.004(duration) - 0.054(campaign) + 0.97(pdays)

+ 0.127(previous) - 0.279(emp.var.rate) + 0.525(cons.proce.idx) + 0.071(cons.conf.idx)

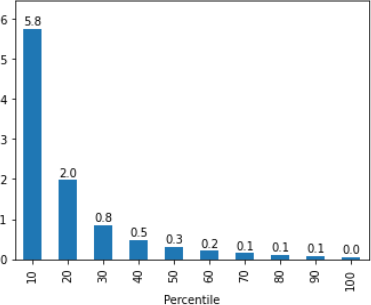
+ 0.029 (euribor3m) - 0.01(nr.employed) - (0.049)housing\_yes + 0.067(loan\_yes)

- 1.087(contact\_telephone) + 0.456(poutcome\_nonexistent) + 1.025(poutcome\_success)

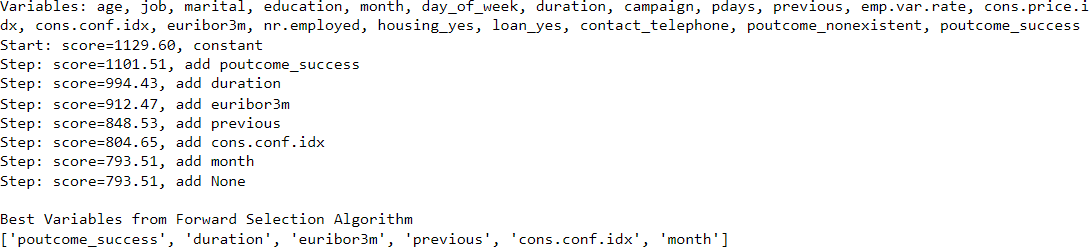
The probability for each record is calculated for both zero and one outcome. The record is classified based on the higher probability. Confusion matrix for the model is:



Another method to visually measure the performance of Logistic model is Lift chart. Below lift chart depicts that for top 10% likely one records, logistic model predicts 5.8 times better than the random model.



One of the problems with logistic model is that more the number of predictors, higher is the complexity and cost. So, the forward selection algorithm is executed to reduce the number of predictors and then logistic model is applied for predictions. Best predictor list achieved with forward selection algorithm is:

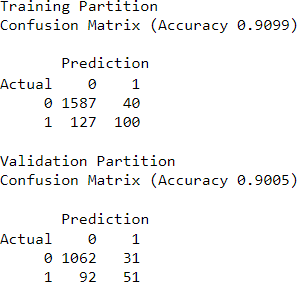


The logit function using forward selection predictors in the logistic model is computed as:

**Logit** = - 0.022 + 1.88(poutcome\_success) + 0.004(duration) – 0.619(euribor3m)

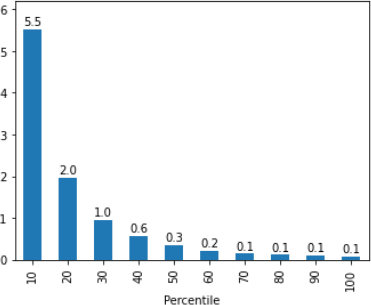
+ 0.174(previous) + 0.055(cons.conf.idx) + 0.041(month)

Confusion matrix values using reduced predictors



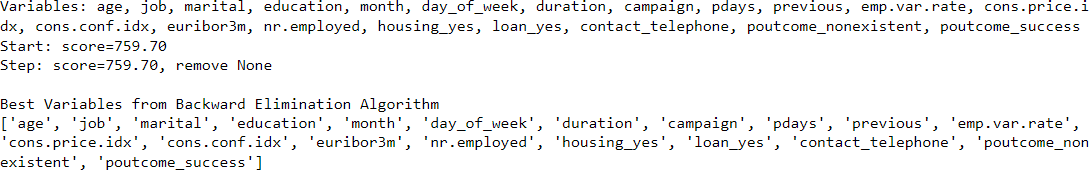
Lift chart performance with reduced predictors logistic model for top 10% likely one records is

* 1. times better than random model.



Another algorithm applied to reduce predictors is backward elimination. But it did not reduce any predictors, thereby making the resultant model the same as all predictor’s logistic regression model. Hence, it is not considered for further analysis.

Backward Selection Predictors



All Predictors Logistic vs Forward Selection Predictors Logistic

|  |  |  |
| --- | --- | --- |
| **Logistic Model** | **All Predictors** | **Forward Selection Predictors** |
| **Number of Predictors** | 20 | 6 |
| **Validation Accuracy** | 0.9070 | 0.9005 |

The accuracy for reduced predictors model is slightly less than all predictors, but number of predictors are significantly less. Also, lift chart performance is nearly similar for both the models. So, the forward selection predictors logistic model is better as it is less complex with similar performance.

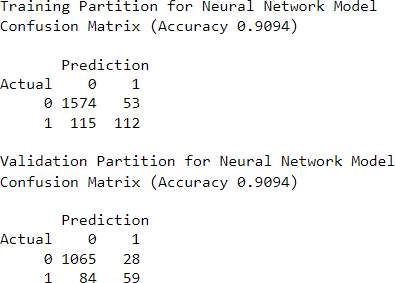
# NEURAL NETWORKS

Finally, the neural network model is built for classification. The key concept in neural network model is to combine input information to capture complex relationship between input and output. Model coefficients and weights are improved in the iterative process. Neural network tries to learn the relations like perceptron used by human brain. It is a feed-forward network with input, output, and multiple hidden layers.

Scaling: The input variables (cons.price.index, cons.conf.index, etc.) have different range of

values. To avoid dominance of one input variable on other due to large values, the input variables are scaled using standard scalar function. Scaling basically calculates z-scores for all the inputs. Z-scores are calculated by subtracting mean value from each value and then perform division by standard deviation for each input.

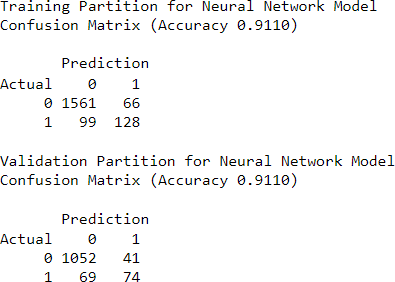
Dataset is divided into training (60%) and validation (40%) partition. Initial neural network model is built with one hidden layer consisting of 3 nodes. Confusion matrix results for this model is:



The performance of neural network model depends on the optimal number of hidden layer sizes. The grid search function is used to determine the optimal number of hidden layer nodes. The improved parameter is:



Another model is built using single hidden layer with 2 nodes as improved parameter. The confusion matrix with improved size is:



Initial Neural Network vs Improved Neural Network

|  |  |  |
| --- | --- | --- |
| **Model** | **Initial Neural Network** | **Improved Neural Network** |
| **Hidden layer nodes** | 3 | 2 |
| **Validation Accuracy** | 0.9094 | 0.9110 |

The improved neural network model has slightly better accuracy with a smaller number of nodes. So, the improved neural network model is better choice than initial neural network model.

# FINAL COMPARISON

Three prediction models with different variations are implemented for the bank dataset in this study. The best variation from each model is analyzed based on their performance.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Improved Decision**  **Tree** | **Forward Selection**  **Predictors Logistic** | **Improved Neural**  **Network** |
| **Validation Accuracy** | 0.8908 | 0.9005 | 0.9110 |
| **Number of Predictors** | 20 | 6 | 20 |

* + - The Improved Decision Tree model has lowest accuracy with all 20 predictors, so it is not a good choice.
    - The Forward selection predictors logistic model has slightly less accuracy than Improved Neural Network model but has significantly less predictors.
    - The validation accuracy is best for Improved Neural Network model, but it comes with a little higher computational cost.
    - If computational cost is not a concern, then Improved Neural Network is the best model with highest accuracy.

# CONCLUSION

Different Data mining techniques for prediction of categorical outcome are implemented in this study. The dataset is significantly larger, so the KNN model is not considered. Decision Tree, Logistic Regression and Neural Network model are implemented with different variations. The best variation of each model is evaluated using different factors. Finally, the best variation from each model is compared based on respective validation accuracies and number of predictors. Forward selection logistic regression is a good choice if low computational cost is the requirement. But the best model with highest accuracy is the Improved Neural Network model with 91.1 % accuracy. Thus, we conclude by, recommending the Improved neural network model to predict and identify the right customers to be targeted through the Banks’s campaign and increase the campaign’s success rate.

# APPENDIX

* Dataset: [Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22- 31, June 2014
* Referred Dr. Radovilsky python files for coding different algoritms
* Referenced Prof. Dr. Radovilsky Lecture notes