Bank Term Deposit Subscription Classifier

Milestone: Project Report

Group 15

Sanay Shah

Pushkar Dhabe

857-318-6077 (Tel of Student 1)
857-350-5645 (Tel of Student 2)

shah.sana@northeastern.edu
dhabe.p@northeastern.edu

Percentage of Effort Contributed by Student 1: 50%

Percentage of Effort Contributed by Student 2: 50%

Signature of Student 1: Sanay Shah

Signature of Student 2: Pushkar Dhabe

Submission Date: 04/25/2022

Problem Setting:

Banks are financial institution that are licensed to grant loans and to receive deposits from its customers. For a bank to function, it must make profits so as to pay its employees as well as sustain against its competitors and the best way for it to do so is through term deposits. Term deposits are fixed term investments where someone (person or entity) deposits an amount to the bank for a certain amount of time and the bank pays back the money with some interest. This allows the bank to know for certain how much money does it have that it can lend to others at any point in the future. The bank uses this money to invest it into other projects that pay them a higher rate of return than the term deposit's interest rate thus making a profit in between. These projects could be lending money to others at a higher interest rate for the borrower or investing it elsewhere. Term deposits are extremely safe and are therefore appeal to low-risk investors who want to ensure security in their earnings.

Problem Definition:

The data is sourced from a Portuguese banking institution. This institution reached out to its clients (mostly through direct phone calls) to enquire if they'd be interested in making a Term Deposit with them and the results have been documented. It contains some information about their clients and when were they contacted regarding this campaign and if they ended up making a Term Deposit or no. The basis of our project is to build a classification model that could use the details of their clients and this previous marketing campaign results to predict if the client makes a Term Deposit or no. The model could then be used to seek out potentially clients who would make the same investment.

Data Sources:

We have sourced our data from the following public data repository –

- https://archive.ics.uci.edu/ml/datasets/bank+marketing#

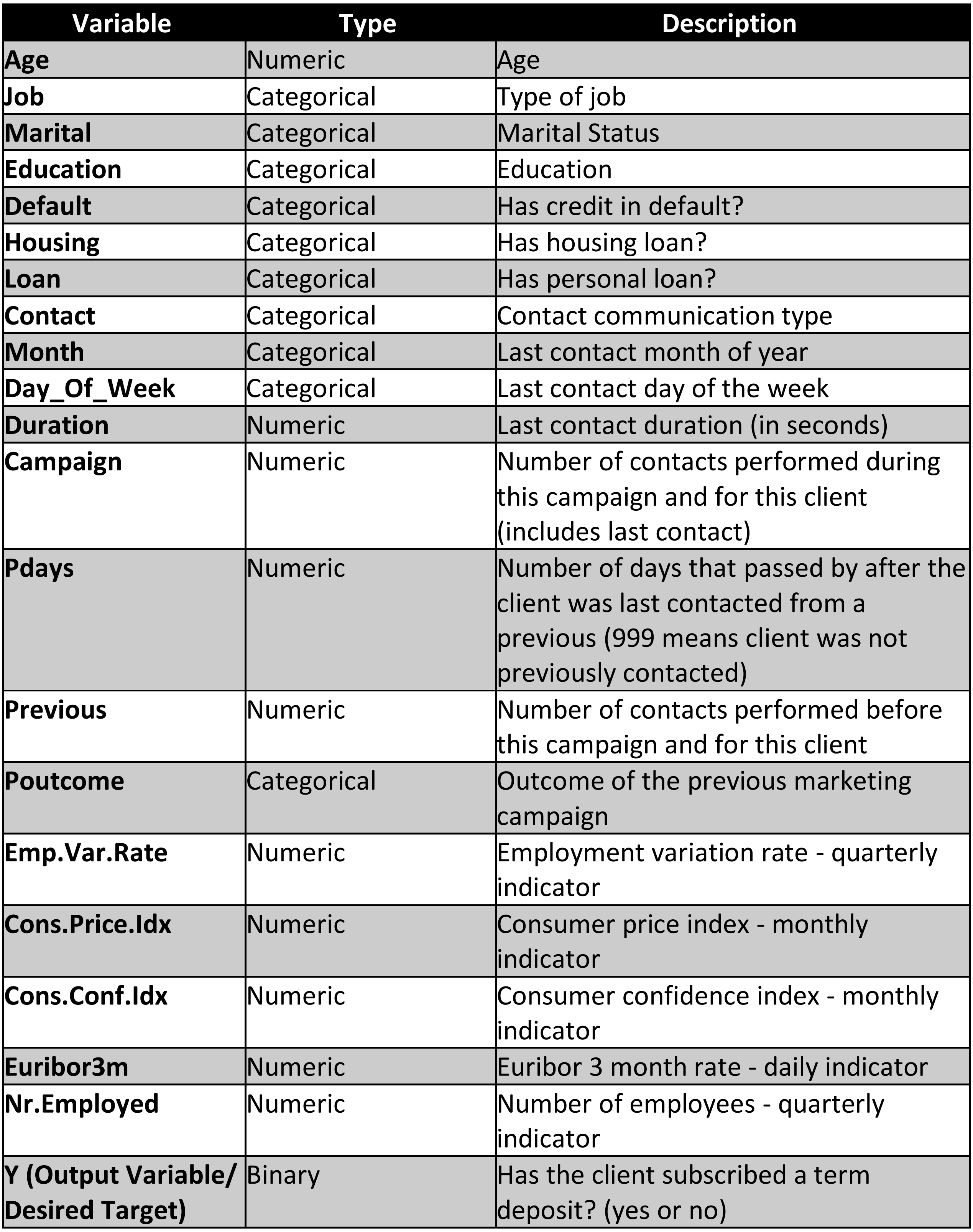
Data Description:

The data contains about 40 thousand entries and has about 20 attributes. The attributes describe their clients. This data was collected from May 2008 to November 2010.
Target variable (y) – If the client subscribed to a term deposit (binary – yes or no).

Attribute Information:

The data attribute information is taken from UCI Machine Learning Repository

Input variables:



Variable	Type	Description
Age	Numeric	Age
Job	Categorical	Type of job
Marital	Categorical	Marital Status
Education	Categorical	Education
Default	Categorical	Has credit in default?
Housing	Categorical	Has housing loan?
Loan	Categorical	Has personal loan?
Contact	Categorical	Contact communication type
Month	Categorical	Last contact month of year
Day_Of_Week	Categorical	Last contact day of the week
Duration	Numeric	Last contact duration (in seconds)
Campaign	Numeric	Number of contacts performed during this campaign and for this client (includes last contact)
Pdays	Numeric	Number of days that passed by after the client was last contacted from a previous (999 means client was not previously contacted)
Previous	Numeric	Number of contacts performed before this campaign and for this client
Poutcome	Categorical	Outcome of the previous marketing campaign
Emp.Var.Rate	Numeric	Employment variation rate - quarterly indicator
Cons.Price.Idx	Numeric	Consumer price index - monthly indicator
Cons.Conf.Idx	Numeric	Consumer confidence index - monthly indicator
Euribor3m	Numeric	Euribor 3 month rate - daily indicator
Nr.Employed	Numeric	Number of employees - quarterly indicator
Y (Output Variable/ Desired Target)	Binary	Has the client subscribed a term deposit? (yes or no)

Table 1.1

Data Exploration

The dataset consist of 41187 records and 20 attributes and 1 target variable Y .The data consist of 9 categorical variables and 11 numeric variables. The categorical variables housing, loan ,contact are binary in nature but rest of the attributes are multiclass variables. The target variable ‘Y’ is also binary in nature. Consist of only yes and no ,where the value yes is the class of interest. There are 4641 values in class of interest i.e. client subscribed to a term deposit and 36549 instances where the client hasn’t subscribed to term deposit.

Data Pre-Processing

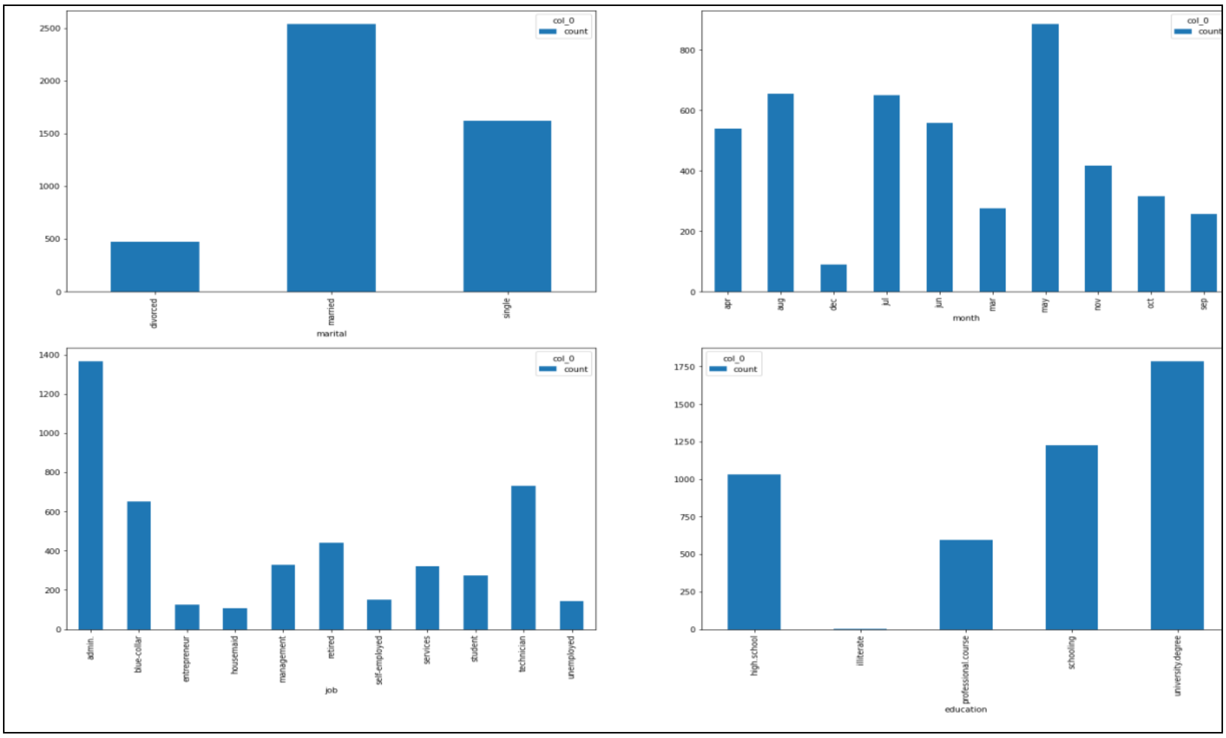
The dataset had multiple instances of where the word “unknown” was populated for any attribute value that wasn’t filled with a value. This was only seen in 6 categorical variables ~ Job, Marital, Education, Default, Housing and Loan. The attribute “Default” was removed as there were only 3 rows with a “yes” and the rest of the 32 thousand rows were “no” (8500 – unknown). “Duration had 2 unknown instances and so did “Pdays” from the numeric variables. “Pdays” was later dropped because 96% of this column was populated with the number 999 which indicated that the client was not previously contacted. We assumed that the variable “Age” was entered perfectly and is the basis on which we impute the unknowns. An “age_group” variable was created based on the age of the client (20-year-olds to 30-year-olds ~ 20, 30-year-olds to 40-year-olds ~ 30 and so on). The unknowns were then replaced by the mode of the categorical variable at hand, grouped by the variable “age_group”. Dummy variables were then created for n-1 levels for each categorical variable.

Data Visualization

The numerical and categorical variables were analyzed separately.

i) Frequency Count for “yes”

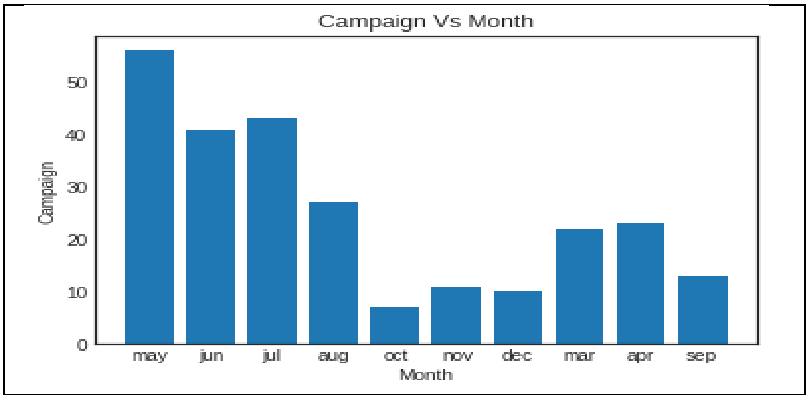
The frequency counts for “yes” on the dependent for the categorical variables are as follows:



Insights:

- Married leads have made high deposits followed by single
- There were much deposits made during may month as it is the start of bank period
- Leads who work in administrative position made deposits followed by technicians and blue-collar employees
- Leads who had atleast university degree had made the deposits followed by high school

ii) Number of Campaigns vs Month

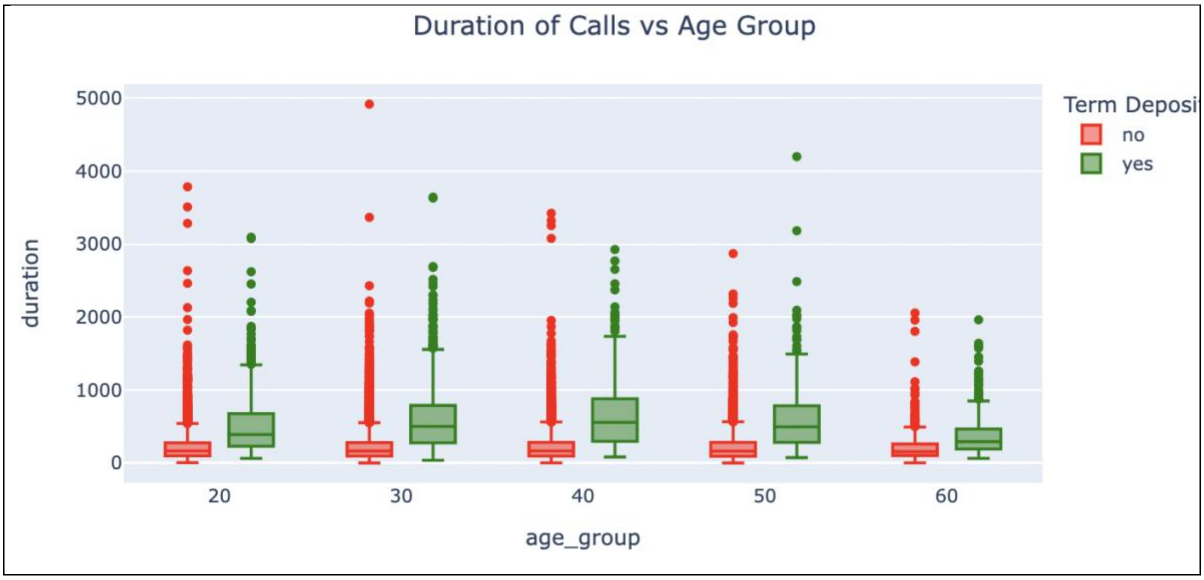


Insights:

- We can see the campaign were mostly concentrated in the starting of the bank period (May, June and July)
- Usually, education period starts during that time so there is a possibility that parents make deposits in the name of their children
- They also have made their campaign in the end of the bank period.

As observed, every variable has more than 2 levels. With "job" and "marital" being nominal and "education" and "month" ordinal variables. These counts have been obtained after imputing the unknowns with their respective age group's median for every row.
Let's look at the relationship between the categorical variables and the dependent variable too.

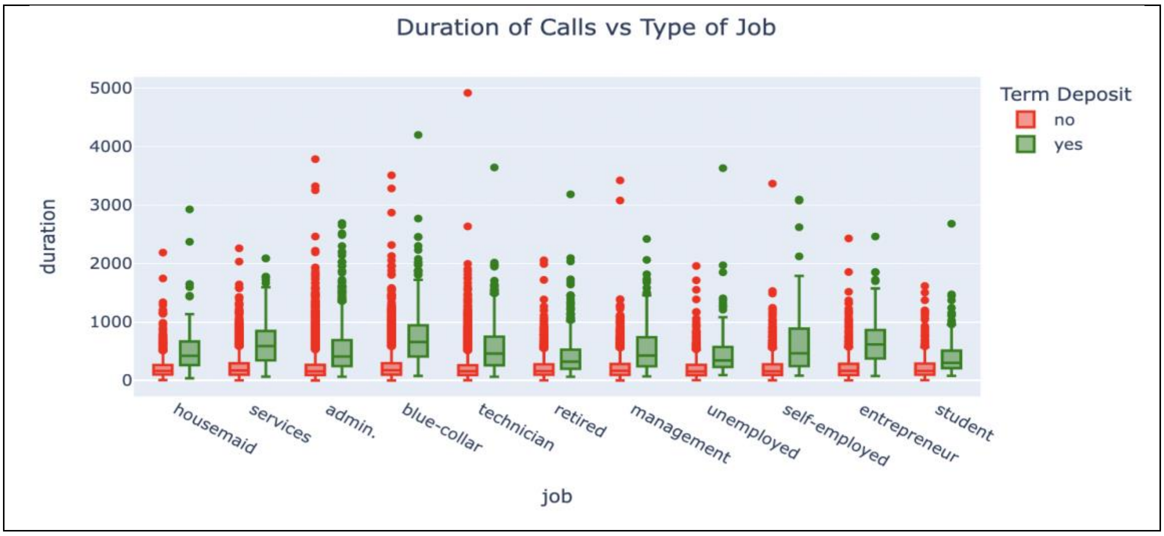
iii) Duration of Calls vs Age Group



Insights:

- Leads who have made a deposit majorly belong to age group 40 followed by age group 30 and 50
- Potential clients were distributed in wide age group of 30 to 50 years.
- The medians for all age groups as for the people who made the term deposit is higher than the people that didn't make a term deposit.

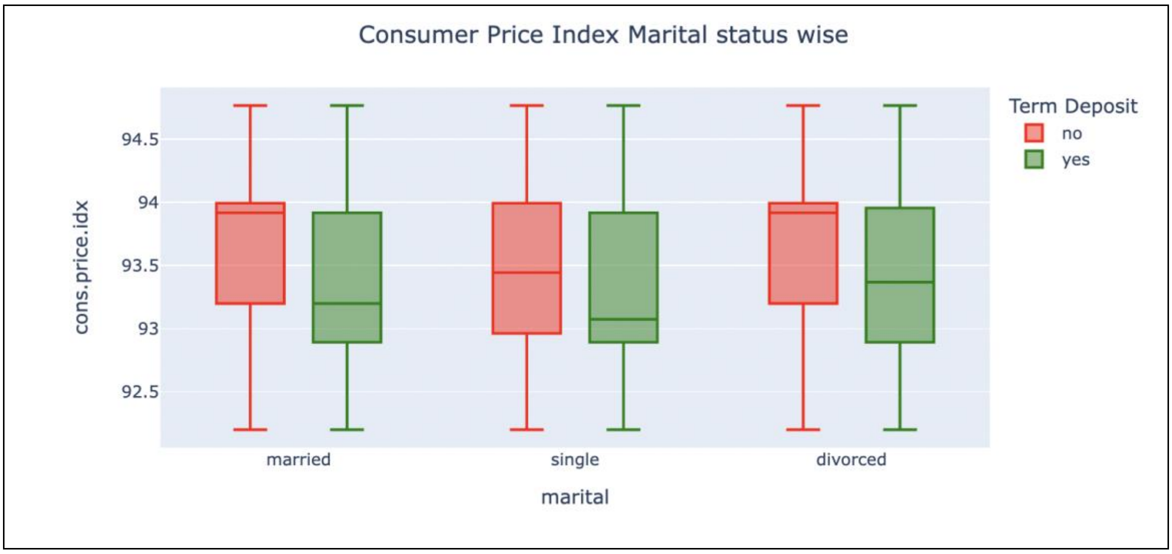
iv) Duration of Calls vs Type of Job



Insights:

- Leads who have not made a deposit have a lower call time
- Compared to average, blue-collar and entrepreneur have higher duration of calls and retired, student have lower duration on average
- Potential clients were distributed on a wide range of self-employed clients and management personnel.

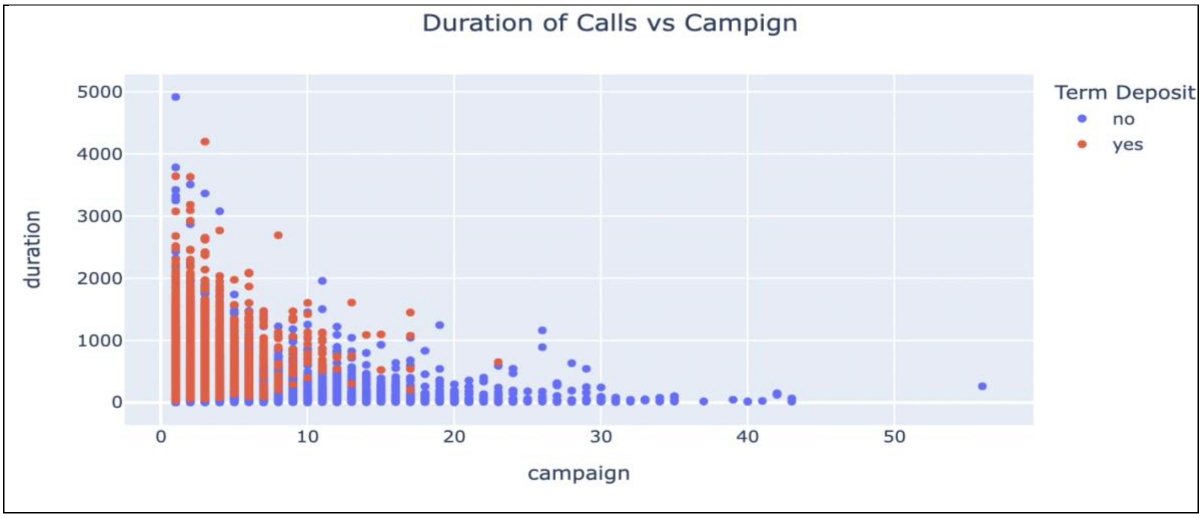
v) Consumer Price Index vs Marital Status



Insights:

- There are very minute differences among the price index.
- Married leads have considerably have an upper hand as they have index contributing as couple

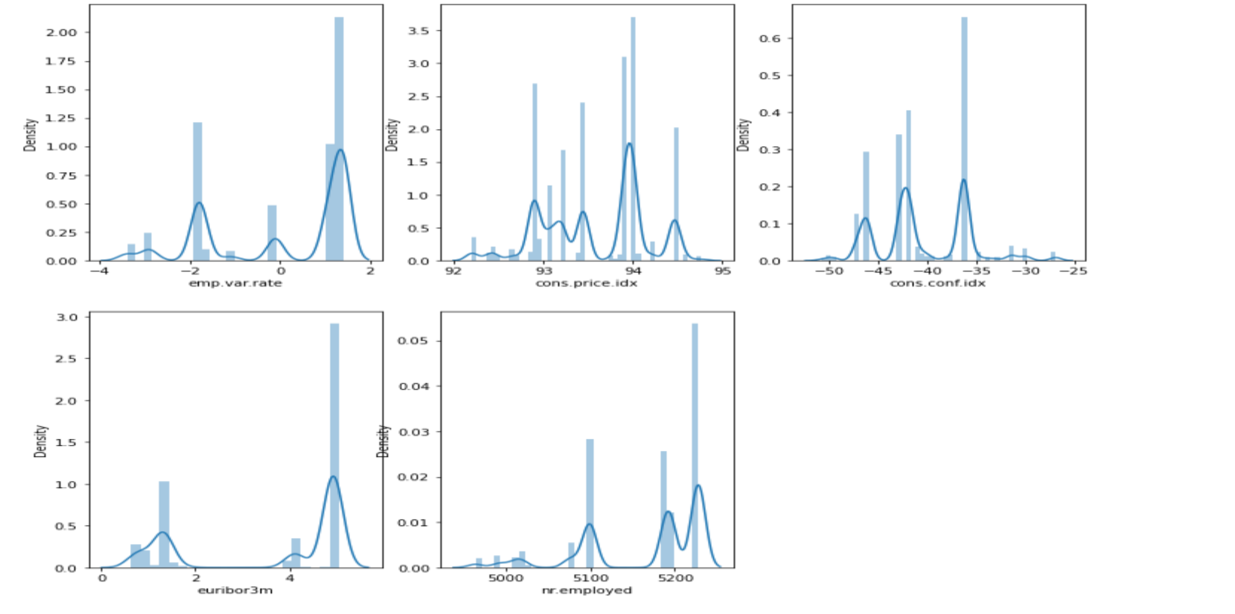
vi) Duration of Calls vs Campaign Contacts



Insights:

- The more the duration the calls were, they had higher probability in making a deposit
- Duration of calls faded as the time of campaign extended further
- There were many positive leads in the initial days of campaign

vii) Univariate plots

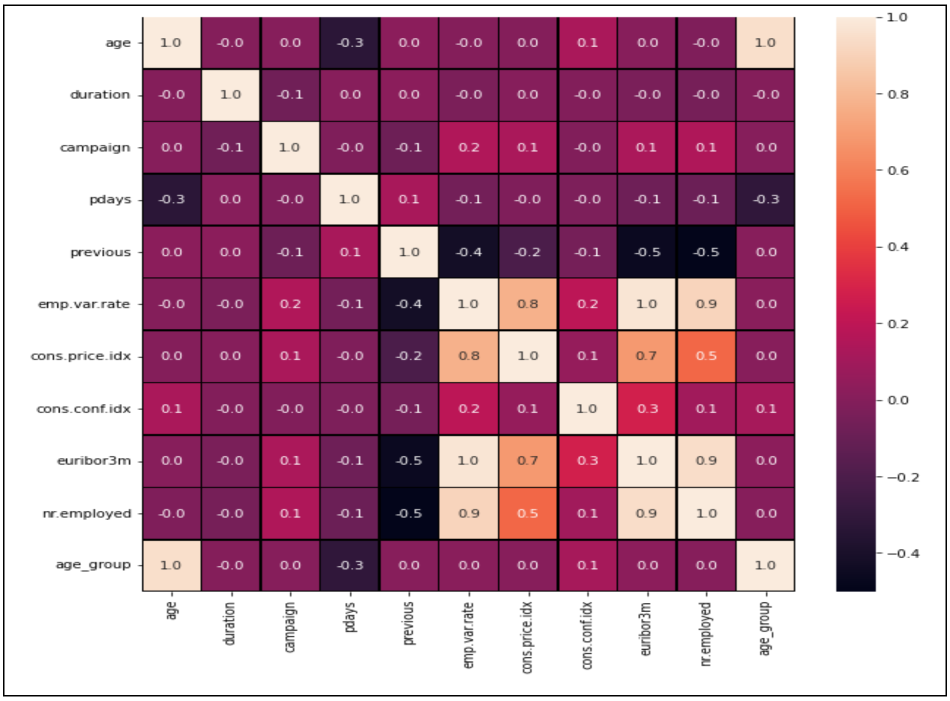


Insights:

- We can see there is a high employee variation rate which signifies that they have made the campaign when there were high shifts in job due to conditions of economy

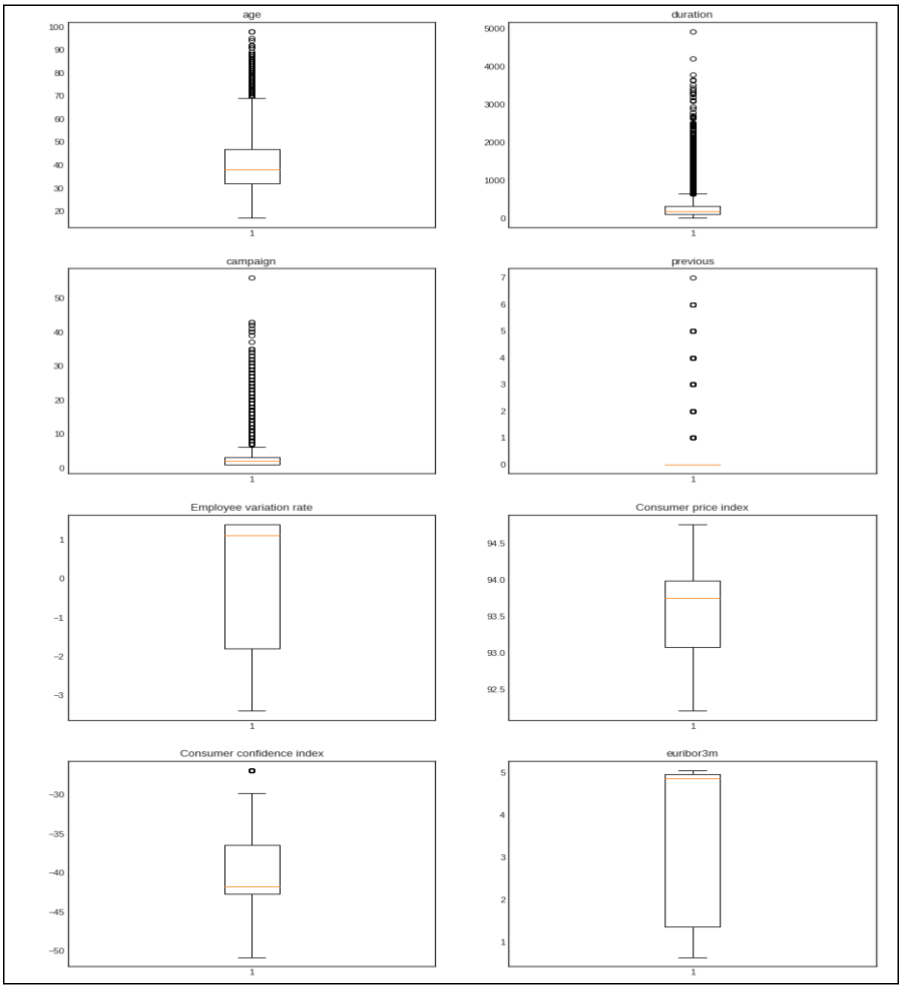
- The Consumer price index is also good which shows the leads were having good price to pay for goods and services may be that could be the reason to stimulate these leads into making a deposit and plant the idea of savings
- Consumer confidence index is low as they don't have much confidence on the fluctuating economy
- The 3-month Euribor interest rate is the interest rate at which a selection of European banks lend one another funds denominated in euros whereby the loans have a maturity of 3 months. The interest rates are high for lending their loans
- The number of employees were also at peak which can increase their income index that could be the reason the campaign targetted the leads who were employeed to make a deposit

viii) Correlation heatmap

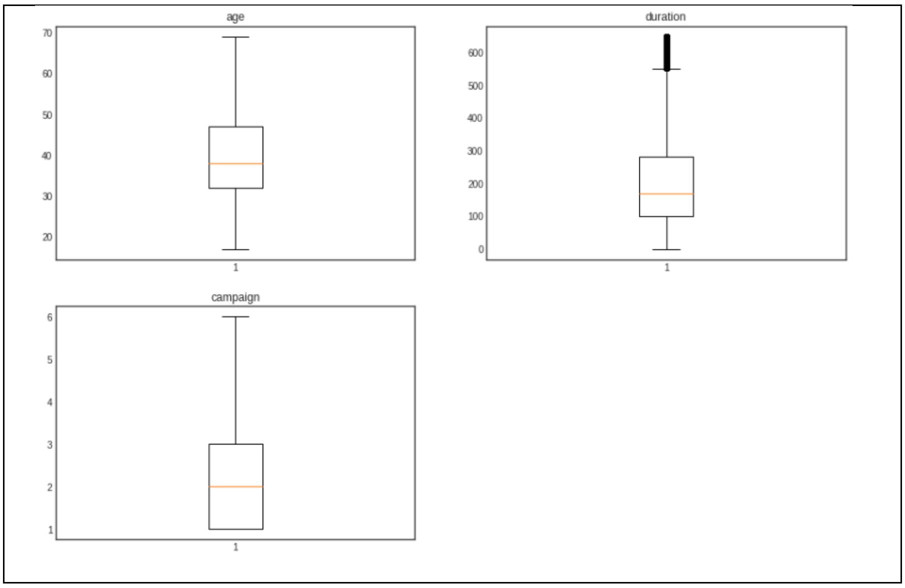


- It can be observed that euribor3m and nr employee are highly correlated. It’s the same case with nr employee and emp.var.rate.
- The people who made the term deposit needed lesser number of contacts during the campaign period. Even though lesser number of contacts had to be made, the duration of calls was a lot more.
- Dropped column nr.employed as its highly correlated to two columns emp.var.rate and euribor3m

Figuring out the outliers in each predictor variables through box plot



Majorly Outliers were observed in Age, Campaign and Duration Predictor variables which were then handled using the Interquartile range.



Model Exploration:

Before Exploring the models, we have encoded all the categorical variables using different encoding methods like ordinal encoding, frequency encoding

Encoding - Month and Day of week

Encoding the categories in month and day of week to the respective numbers.
Days: Sunday to Saturday as 1 to 7
Months: January to February as 1 to 12

Ordinal Number Encoding -Loan, Housing

Encoded the features which yes and no. Assigned Yes :1, No:0

Ordinal One -Hot Encoding -Contact, Poutcome

Performed one-hot encoding for the contact, poutcome features and dropped the original features. Created dummy columns: dummy_telephone, dummy_nonexistent, dummy_success

Frequency Encoding -Job, Education

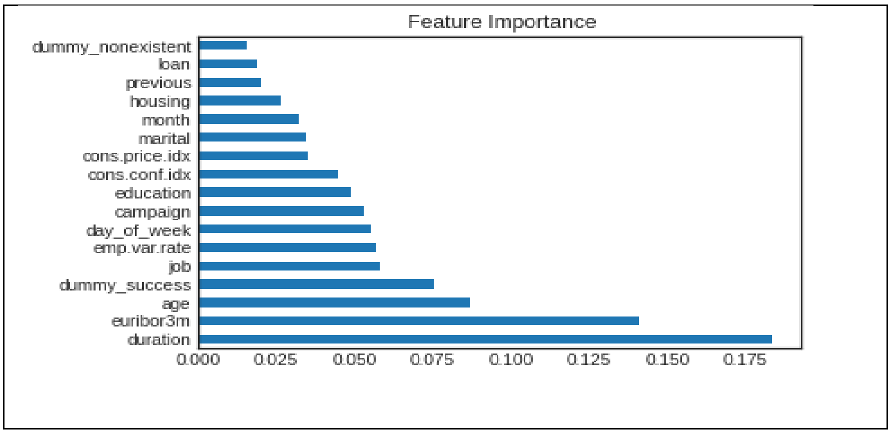
Based on the frequencies to different jobs and education levels created key value pairs and mapped them

Encoding – Marital Status

Encoded the marital features values Single, Married and Divorce. Assigned Single :1, Married :2, Divorced :3
Many features don't have much outliers except for age, duration and campaign Fixed those features using IQR method.

Feature Selection

From the bar plot we can see the importance of features based on its impact towards output. Selected the top 11 features and dropped rest of the features.
Duration, Euribor3m and Age are top 3 most important features.



Data Preparation

A big scale difference was noticed between a few predictor variables such as duration cons.price.idx nr.employed, which is a numeric value, according to the summary statistics. To account for the scaling disparity, the dataset was standardized using the scikit-learn package's StandardScaler() function to guarantee that all predictor variables were given equal weight in terms of variability.

Over Sampling

The data was biased so over sampling of the data was done using RandomOverSampler which in turn increased the number of records to 65354 records

Data Partitioning

The predictor variables were collectively represented using variable 'X' and the target variable “y” was represented using the variable “y” Variables with index 0 to 10 were defined by 'X' and the target variable with an index 11 was represented by 'y'. The Hold out Method for data partitioning was used on the dataset, such that the train and test sets generated were of the ratio 80:20. This implied that 80% of the dataset was used for training the classification models and theremaining 20% of the dataset was used as a test set for evaluation of the classification performance of those models. X_train contained 52283 records and 11 variables, while X test contained 13,071 records and 11 variables respectively. y train contained 52283 records and I variable, while y test contained 13,071 records and I variable respectively. The forward-selection method of variable selection was used for the model building phase.

Data Mining Models/Methods

There were seven Data Mining classification models that were built using the training data using Logistic Regression, Decision Tree, K-NN , Naive Bayes , Gradient Boosting Classifier ,Random Forest and SVC.

Following base accuracies were observed on training data. The models were then tuned to get the best accuracy.

Logistic Regression Test Accuracy: 0.8444809614540517

Decision Tree Test Accuracy: 0.9732608761979964

KNN Test Accuracy: 0.9307997029991076

Naive Bayes Test Accuracy: 0.7861829392918088

Gradient Boosting Test Accuracy: 0.8971941347006457

Random Forest Classifier Test Accuracy: 0.9757282217942258

From the test results, we can see high accuracy in Random Forest Classifier followed by Decision Tree, KNN, Gradient Boosting and Logistic Regression.

To improve the accuracy for class of interest YES hyper tuning the models and using Grid Search to improve its efficiency using models like Random Forest Classifier, Decision Tree, KNN, Gradient Boosting and Logistic Regression which has high accuracy .

Performance Evaluation

a) Logistic regression

Logistic regression is a parametric classification model that uses a specific model to link predictor factors to the target variable, resulting in a categorical outcome variable.

The results are estimates of the odds of belonging to each class, followed by a threshold cutoff for classifying into either of the classes. Logit is the result variable, which can be described as a linear function of the predictors.

Hyper Paramter Tuning

Grid-search was used to find the optimal model, using max _iter-5, solver= Ibfgs', which had the highest **Test set Accuracy** of **84.83%**. The **Training set Accuracy** was **84.3%**. The **Sensitivity** rating of **86.425** percent implies that the True Positives, or "yes" events, were accurately classified. It was able to correctly classify the True Negatives, or legitimate instances, with a specificity value of 83.03 percent.

An F1-score of 85 percent implies low False Positives and False Negatives, indicating that "yes" was accurately identified. With an AUC of 0.84, the ROC curve was not that close to the top-left corner, showing average discriminating between the “yes” and “no”

LogisticRegression(C=0.08685113737513521, random_state=0)

Confusion Matrix

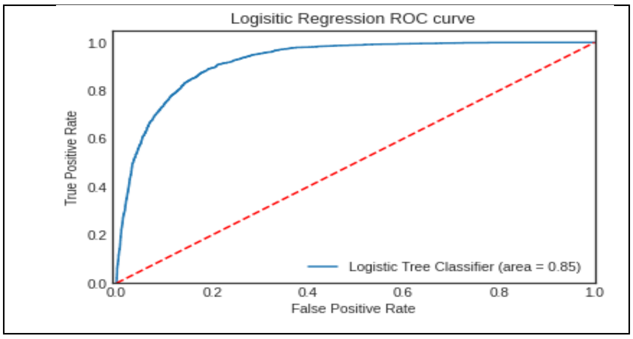
	Predicted C1	Predicted C2
Actual C1	5488	1140
Actual C2	862	5581

True Negative	False Positive
False Negative	True Positive

Classification Report

	Precision	Recall	F1-Score
No	0.86	0.83	0.85
Yes	0.83	0.87	0.85

ROC Curve



b) Gradient Boosting Classifier

One of the most powerful algorithms in the field of machine learning is the gradient boosting technique. Machine learning algorithm faults can be divided into two categories: bias error and variance error. Gradient boosting is one of the boosting strategies that is used to reduce the model's bias error. The Gradient Boost algorithm's base estimator, i.e, Decision Stump, is fixed.

Hyper Parameter Tuning

Grid-search was used to find the optimal model with parameters max_depth=3, min_samples_leaf=1, min_samples_split=2, n_estimators=100,which had the highest **Test set Accuracy** of **89.76%.** The **Training set Accuracy** was **89.99%.** The **Sensitivity** rating of **93.67%** implies that the True Positives, or "yes" events, were accurately classified. It was able to correctly classify the True Negatives, or legitimate instances, with a specificity value of 86.37 % percent.

An F1-score of 89% implies low False Positives and False Negatives, indicating that "yes" was accurately identified. With an AUC of 0.95, the ROC curve was close to the top-left corner, showing good discriminating between the “yes” and “no”

Confusion Matrix

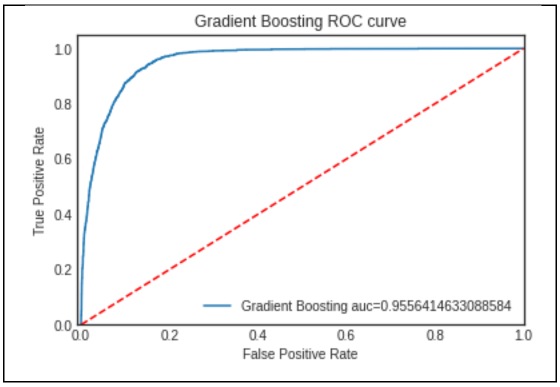
	Predicted C1	Predicted C2
Actual C1	5677	951
Actual C2	452	5991

True Negative	False Positive
False Negative	True Positive

Classification Report

	Precision	Recall	F1-Score
No	0.93	0.86	0.89
Yes	0.86	0.93	0.90

ROC Curve



c) KNN

The k-NN Classification algorithm works on the premise of categorizing a new record from the test data using related records from the training data. The k-nearest neighbors approach is used to find related entries. A malortv decision rule is then used to classify the new record as a member of the majority class of the k-neighbors.

Hyper Paramter Tuning

Grid-search was used to find the optimal model, using neighbors =1. Not training the model with neighbors = 1 as on training and testing we get to know that model is overfitting. Trained the model at neighbor =4 which had the highest **Test set Accuracy** of **95.5%**. The **Training set Accuracy** was **97.33%**. The **Sensitivity** rating of **99.75%** implies that the True Positives, or "yes" events, were accurately classified. It was able to correctly classify the True Negatives, or legitimate instances, with a specificity value of 91.52 percent. An F1-score of 95% implies low False Positives and False Negatives, indicating that "yes" was accurately identified. With an AUC of 0.97, the ROC curve was close to the top-left corner, showing good discriminating between the "yes" and "no"

Confusion Matrix

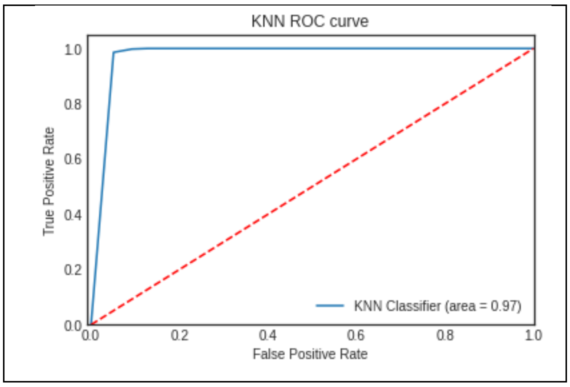
	Predicted C1	Predicted C2
Actual C1	6033	595
Actual C2	15	6428

True Negative	False Positive
False Negative	True Positive

Classification Report

	Precision	Recall	F1-Score
No	1.00	0.91	0.95
Yes	0.92	1.00	0.95

ROC Curve



d) Random Forest

The Random Forest is an ensemble approach that makes predictions by combining a large number of decision trees rather than individual models. Individual models must make predictions that are unrelated to one another, and each model should outperform a random classifier.

Hyper Paramter Tuning

Grid-search was used to find the optimal model, using paramters max_depth=80, max_features=3, min_samples_leaf=3, min_samples_split=8, n_estimators=1000, random_state=0 =1which had the highest **Test set Accuracy** of **96.53%**. The **Training set Accuracy** was **98.06%**. The **Sensitivity** rating of **99.88%** implies that the True Positives, or "yes" events, were accurately classified. It was able to correctly classify the True Negatives, or legitimate instances, with a specificity value of 93.53%. An F1-score of 97 % implies low False Positives and False Negatives, indicating that "yes" was accurately identified. With an AUC of 0.97, the ROC curve was very close to the top-left corner, showing great discriminating between the “yes” and “no”

Confusion Matrix

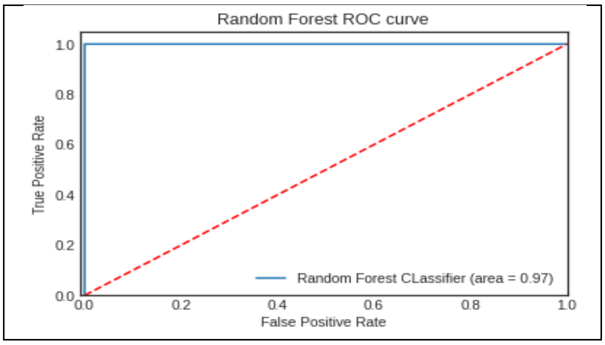
	Predicted C1	**Predicted C2**
Actual C1	6174	454
Actual C2	0	6443

True Negative	False Positive
False Negative	True Positive

Classification Report

	Precision	**Recall**	**F1-Score**
No	1.00	0.93	0.96
Yes	0.94	1.00	0.97

ROC Curve



e) Decision Tree Classifier

The Decision Tree Classifier is a tree-like data-driven or non-parametric approach for formulating predictions. Trees separate predictors in such a way that homogeneity increases with each split, dividing records into sub-groups and offering simple logical principles.

Hyper Paramter Tuning

Grid-search was used to find the optimal model, using paramters max_depth=27, max_features=5,, random_state=0 which had the highest **Test set Accuracy** of **97.59%**. The **Training set Accuracy** was **99.98%**. The **Sensitivity** rating of **100%** implies that the True Positives, or "yes" events, were accurately classified. It was able to correctly classify the True Negatives, or legitimate instances, with a specificity value of 95.52%.

An F1-score of 98% implies low False Positives and False Negatives, indicating that "yes" was accurately identified. With an AUC of 0.98, the ROC curve was very close to the top-left corner, showing ideal discriminating between the “yes” and “no”

Confusion Matrix

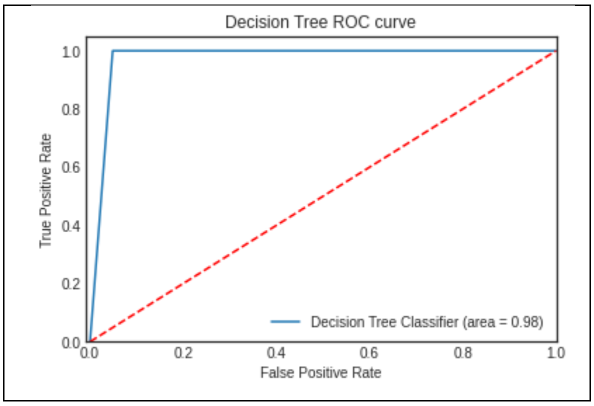
	Predicted C1	Predicted C2
Actual C1	6326	302
Actual C2	0	6443

True Negative	False Positive
False Negative	True Positive

Classification Report

	Precision	Recall	F1-Score
No	1.00	0.95	0.98
Yes	0.95	1.00	0.98

ROC Curve

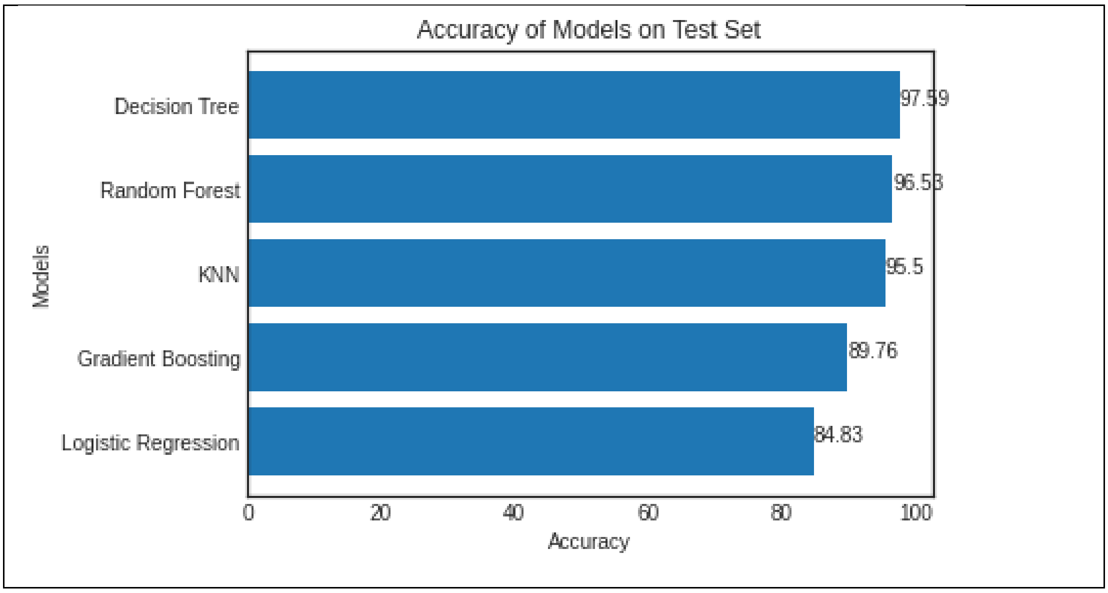


Results:

Summary of all models:

Models	Overall Test Accuracy %	Validation Error %	Sensitivity %	Specificity %	F-1 Score %
Logistic Regression	84.83	15.17	86.42	83.03	85
Gradient Boosting	89.76	10.24	93.67	86.37	90
KNN	95.5	4.5	99.75	91.52	95
Random Forest	96.53	3.47	99.88	93.53	97
Decision Tree	97.59	2.41	100	95.52	98

Accuracy of Models on Test Set:



Decision Tree model gives the highest **Test set Accuracy** of **97.59%** and gives **100 % Sensitivity** and 95.52 % specificity and outperforms all other models. Decision Tree Classifier is the best classification model which could be used to figure out potential client who would like to make a deposit.

Impact of Project Outcomes

We can see that duration is a significant attribute in deciding the outcome of our dataset from the EDA and model selection sections. The greater the number of leads interested in initiating a deposit, the more calls will be made, and the call duration will be longer than normal. We've also discovered that employment and education are important deciding factors that have a significant impact on the results.

Here are a few bank suggestions that can help enhance the deposit rate.

- Job roles should be classified according to corporate tiers, and all tier 1 employees should be contacted within a few days of the campaign starting.
- Listen to the leads and get more information to provide the best deposit plan, which can lengthen the calls and lead to a deposit.
- Approaching leads at the start of the new bank period (May-July) is a smart idea because many have showed favorable results in the past.
- Tune the campaign according to national econometrics, and don't cut campaign spending when the national economy is struggling.