GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Final Capstone Report

Bank Term Deposit Prediction

Submitted towards partial fulfillment of the criteria for award of PGPDSE by GLIM

- **Group - IV: PGPDSE (Feb 2022 Great Learning Gurgaon Batch)**
- **Students: Aayush Kandhari, Shubham Jain ,Palak Kumar, Vinayak Chaudhary, Shivam Goel , Mrinal Dhasmana**
- **Mentor: Ms Anjana Aggarwal**

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Abstract

The increasingly vast number of marketing campaigns over time has reduced its effect on the general public. Economic pressures and competition have led marketing managers to invest on directed campaigns with a strict and rigorous selection of contacts: lesser contacts should be done, with better success rate. Although telemarketing is a direct mode of communication with the prospective customer, this may make customers grumpy.

In this Project, Exploratory Data Analysis, Statistical Tests, Imbalanced Data treatment and Predictive modeling is used in determining the main characteristics that affect success and selection of potential buying customers. Classification algorithms like Logistic Regression, CART, KNN, and ensemble algorithms like random forest and XG Boost were used to build the model using the most popular tool python and the appropriate model is selected based on F1-score, ROC, Accuracy value and False Negative value (FN). Further, this project also attempts to provide model interpretability that may help bank target the right customers.

Keywords: Exploratory Data Analysis, Null Value Imputation, Predictive Modeling, scikit learn, bootstrap sampling, predict probability, etc

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Table of Content

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Chapter 1

Introduction

1.1. Objective:

The purpose of this study is to build a robust classification model to predict the success of telemarketing calls for selling bank long-term deposits. The data which is used in this study consists of 20 independent variables. Feature selection approach has been used to select the best subsets of variables and then different type of classification algorithms have been used to check their performance. The base line model is then compared with the final model, ob- tained through a comprehensive exploratory data analysis, feature selection, model compari- son and hyper parameter tuning.

1.2. Data Source:

In this project we study different approaches to predict the success of bank telemarketing. As instrument we have a dataset related with direct marketing campaigns based on phone calls of a Portuguese banking institution. Often, more than one contact to the same client was re- quired, in order to access if the product (bank term deposit) would be (yes) or not (no) sub- scribed.

The data under study here is called Bank Marketing Dataset (BMD) and was found in the Machine Learning Repository (UCI). The data is public available in https://archive.ics.uci.edu/ml/datasets/Bank+Marketing .The size of the dataset is considera- bly large, especially if we consider its origin. Data from clients of financial institutions are usually difficult to find, and when found, are rarely available in this quantity. In the BMD data we have 41188 observations, with twenty one features.

Table 1: Features description of the Bank Marketing Dataset (BMD).

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Num.	Attribute Name	Description	Type
1	Age	Age of Client	Numeric
2	Job	Type of Client's Job	Categori-cal
3	Marital	Marital Status of client	Categorical
4	Education	What is the highest education of client?	Categori-cal
5	Default	Does client has Credit?	Categori-cal
6	Housing	Does client has Housing Loan?	Categori-cal
7	Loan	Does client has personal Loan?	Categori-cal
8	Contact	What is a contact communication type of client?	Categori-cal
9	Month	What is the last month of the year contracting to the client?	Categori-cal
10	day_of_week	What is the last day of the week contracting to the client?	Categori-cal
11	Duration	How long does it contact to the client?	Numeric
12	Campaign	Number of contacts performed dur-ing this campaign and for this client	Numeric
13	Pdays	Number of days that passed by after the client was last contacted from a previous campaign	Numeric
14	Previous	Number of contacts performed be-fore this campaign and for this client	Numeric
15	Poutcome	Outcome of the previous marketing campaign	Categori-cal
16	Emp.var.rate	employment variation rate - quarterly indicator	Numeric
17	Cons.price.idx	consumer price index - monthly indi-cator	Numeric

18	cons.conf.idx	consumer confidence index - monthly indicator	Numeric
19	Euribor3m	3 month rate - daily indicator	Numeric
20	Nr.employed	Number of employees - quarterly indicator	Numeric
21	Y	Has the client subscribed a term de-posit?	Categorical

The twenty one features are briefly described in Table 1, where in the left column we have the original feature names as in the dataset, and in the right column is its description, men- tioning also if the feature is numeric, categorial, and with how many levels (if categorical, of course). The last variable- „y" is the response or target variable . The other features are pre- sented in the same order that they appear in the dataset

Chapter 2

Literature Review

This section explains the previous research work which have been already done in classification using ML techniques.

The data which is used in this study work is the data of customers of a Portuguese banking institution. The similar data set has been used in Moro et al. (2011, 2014). In Moro et al. (2011), the aim of this study was to find the model that can increase the success rate of tele- marketing for the bank. The statistical techniques of data mining which have been used in their research are Support Vector Machine (SVM), Decision Tree (DT) and Naive Bayes. The performance of these models was checked through the Receiver Operator Characteristics (ROC) curve (detail of ROC curve is given in section 5). Among all these statistical tech- niques, SVM comes up with the most efficient results. Regarding attributes, Call duration was the most relevant feature which states that longer calls tend increase the success rate. Af- ter that month of contact, num- ber of contacts, days since last contact, last contact result and first contact duration attributes respectively.

In Moro et al. (2014), objective of the study was to predict the success of bank telemarketing. Data set which they used in their research was consists of 150 attributes and is complete data

set of the period 2008 to 2013. They compare the 4 data mining models i.e. Logistic Regression (LR), Decision Tree, Support Vector Machine and Neural Network (NN). The best result was obtained by the neural network while decision trees discloses that probability of success in inbound calls are greater.

Statistical learning algorithms have successfully been used in many research problems for classification. For example, Qi et al. (2018) conducted a research to find out the fault diagnosis system for reciprocating compressors. Reciprocating compressors are extensively used in petroleum industry. Data was taken from oil corporation (5 years operational data) and uses the Support Vector Machine to analyse it. They come up with the results that SVM accurately predicts the 80% right classification to find the potential faults in compressor.

Similarly, Gil & Johnsson (2010) did a research in medical field for diagnosing the urological dysfunctions using SVM. In this research data was collected from the 381 patients who are suffering from a number of urological dysfunctions to check the overall worth of decision support system. The fivefold cross validation has been utilised for the robustness. The outputs

of this study describe that for the purpose of classification SVM attained the accuracy of 84.25% .

Nogami et al. (1996) utilised the machine learning in decision support system. In their research they introduce the air traffic management for the future which can manage the flight schedule and flow of air traffic professionally. Their system involves many decision makers and utilised it with the neural network. They require such system which can make the optimal decision in the critical situation. Their simulation studies prove that system which is based on neural network is performed more efficiently than the current air traffic system.

Another research by Cramer et al. (2017) the machine learning methods are used in time series for rainfall prediction. Data was derived from the 42 cities including climatic features.They tried Support vector regression, NN, and k nearest neighbours. After performing these methods they come up with the results that machine learning methods have predictive accuracy.

Wang & Summers (2012) used the machine learning in field of radiology. They used it for the neurological disease diagnosis images, medical image segmentation and MRI images. They come-up with the results that machine learning identifies the complex patterns. It also helps the radiologists to make right decisions. Furthermore, they suggest that development of technology in machine learning is an asset for long term in the field of radiology.

Machine learning algorithms are also used in the field of applied mathematics. For instance, Barboza et al. (2017) did a research to predict the models for developing of bankruptcy by using the SVM and random forest methods. The data was taken from the Salomon Center database & Compustat about North American firms from period 1985 to 2013 with observations of more than 10,000. After applying SVM and RF techniques they compare the results with the ordinary used methods such as discriminant analysis and logistic regression.

To find the risk factors about failure of banks Le & Viviani (2017) conducted a research. In their study, a sample of 3000 US banks was analysed by using 2 traditional statistical methods i.e. discriminant analysis and logistics regression. Then they compare these methods with the machine learning methods i.e. SVM, ANN and k-nearest neighbours. The results of this study illustrate that ANN and k-neighbours method gives the accurate predictions as compared to the traditional methods.

Data Preparation

3.1. Data Cleaning

On observing the data, we got to know that data consists of 41,188 rows and 21 columns out of which 'y' is the target column describing whether the client has subscribed a term deposit? (binary : 'yes', 'no').

Data consists of 11 categorical features - ['job', 'marital', 'education', 'default', 'housing', 'loan', 'contact','month', 'day_of_week', 'poutcome', 'y'] and 10 numerical features - ['age', 'du- ration', 'campaign', 'pdays', 'previous', 'emp.var.rate', 'cons.price.idx', 'cons.conf.idx', 'euri- bor3m', „nr.employed'].

On an abstract level, data does not contain any null values. But on detailed investigation, we have observed that 6 categorical columns namely ('job', 'marital', 'education', 'default', 'housing', 'loan') contain „unknown‟ class which are nothing but missing values. So, we have replaced the `'unknown'` entries with `np.nan` which we have filled by identifying pattern within the data as discussed ahead.

We looked at the percentage of missing values in each column in order to decide whether a column has to be dropped in case it contains more than 50% missing values.

Your selected dataframe has 21 columns.
There are 6 columns that have missing values.

	Missing Values	% of Total Values
default	8597	20.9
education	1731	4.2
housing	990	2.4
loan	990	2.4
job	330	0.8
marital	80	0.2

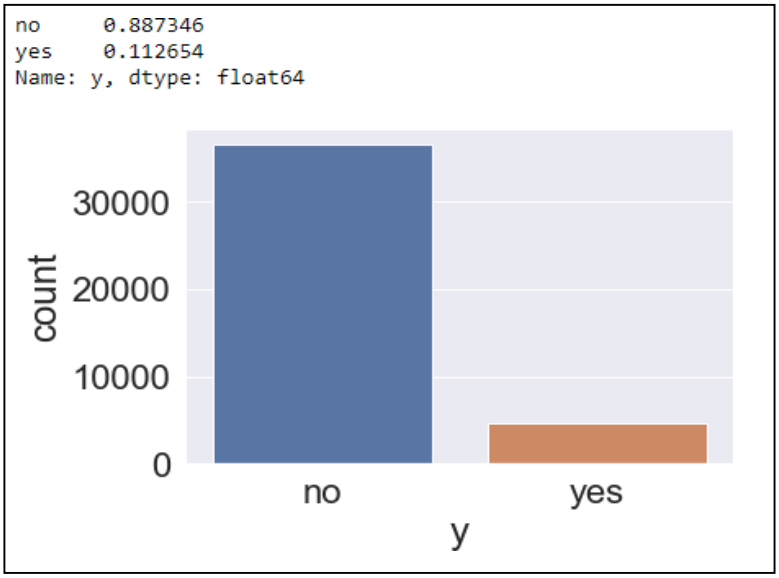
Since all the above columns have less than 50% missing values, thus we would not think of removing any columns right now (may remove later due to other factors). The missing values will have to be imputed (filled-in) using an appropriate strategy before doing machine learning.

3.2. Exploratory Data Analysis

Exploratory Data Analysis (EDA) is an open-ended process where we make plots and calculate statistics in order to explore our data. The purpose is to find anomalies, patterns, trends, or relationships. These may be interesting by themselves (for example finding a correlation between two variables) or they can be used to inform modeling decisions such as which features to use. In short, the goal of EDA is to determine what our data can tell us! EDA generally starts out with a high-level overview, and then narrows in to specific parts of the dataset once as we find interesting areas to examine.

- **Imbalanced Data:**

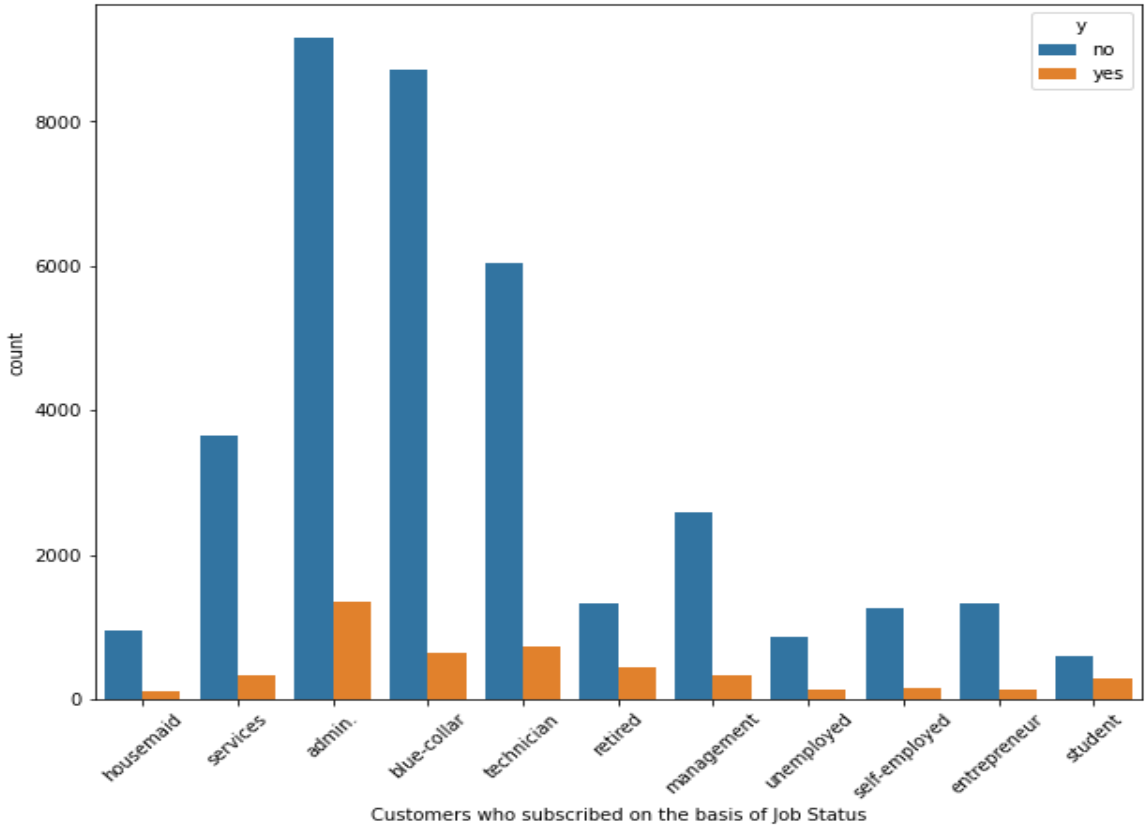
To begin the EDA, we observed the distribution of classes in our target variable, „y':



The target column is highly imbalanced with 88% 'no' as the majority class. Most of the patients have not subscribed for term deposit. If we use this data as the base for our predictive models and analysis, our algorithms would be biased towards majority class 'no' in comparison to 'yes'. Thus, handling imbalanced data is critical to model prediction. We'll have to deal with it using specialized sampling techniques later.

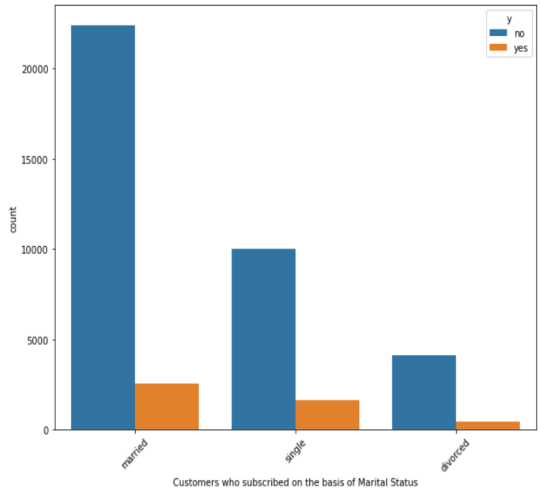
- **Insights:**

1. To understand the relation between target variable and job type, we computed the percentage of subscribed/ not subscribed customers for each job type and gained following interesting insights:



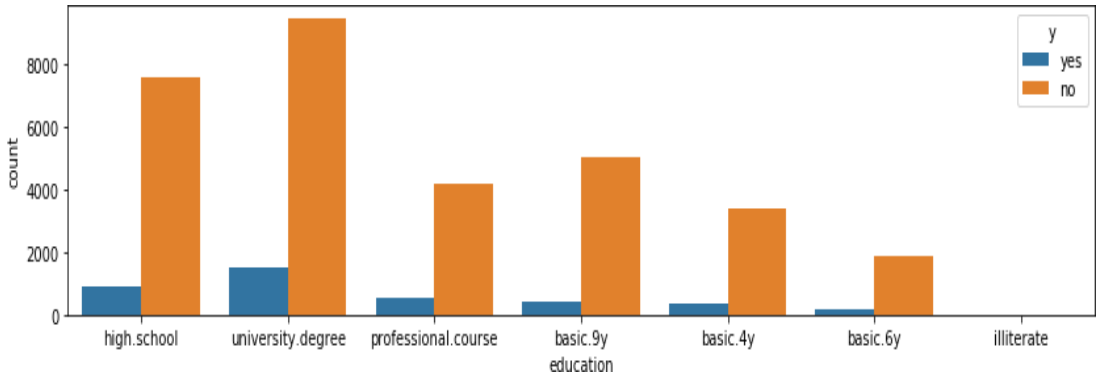
As from the given spread, we can observe that Admin & Blue Collar people was contacted most of the time, though most of the people said 'NO' to the term deposit, but also the people who said 'Yes' to the term deposit are mostly from these 2 classes.

2. In the „marital" column, Maximum no. of customers contacted are married, then single, followed by divorced or widowed




```
marital   y
divorced  no      4136
          yes      476
married   no     22396
          yes     2532
single    no     10016
          yes     1632
Name: age, dtype: int64
```

3. In the „education" column, Illiterate customers have higher chances of buying FD followed by those with university degree



Following is the detailed percentage distribution of subscription status for each education type:

education	y	
basic.4y	no	0.897510
	yes	0.102490
basic.6y	no	0.917976
	yes	0.082024
basic.9y	no	0.921754
	yes	0.078246
high.school	no	0.891645
	yes	0.108355
illiterate	no	0.777778
	yes	0.222222
professional.course	no	0.886515
	yes	0.113485
university.degree	no	0.862755
	yes	0.137245
unknown	no	0.854997
	yes	0.145003

Name: y, dtype: float64

➢ **Correlations between Features:**

Pearson correlation coefficient also referred to as Pearson's r, is a measure of the linear correlation between two variables X and Y. According to the Cauchy–Schwarz inequality it has a value between +1 and −1, where 1 is total positive linear correlation, 0 is no linear correlation, and −1 is total negative linear correlation. This is a measure of the strength and direction of a linear relationship between two variables.

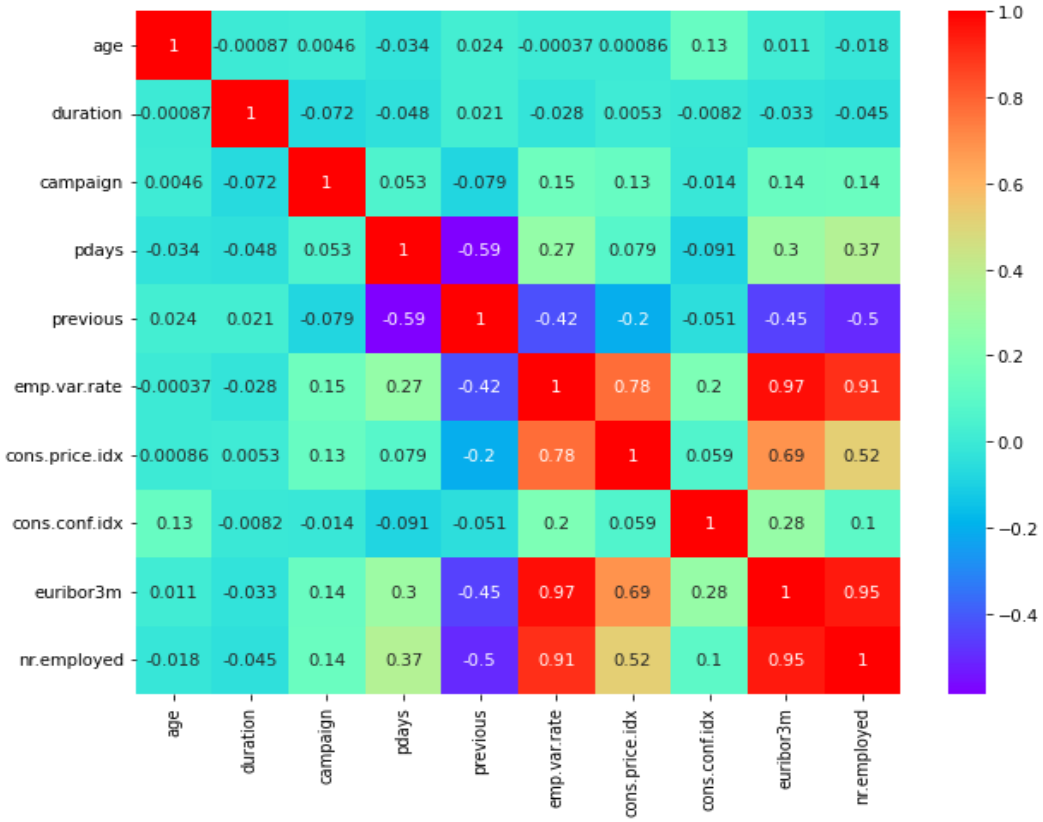
Although there can be non-linear relationships between the features and targets and correlation coefficients also account for interactions between features, linear relationships are a good way to start exploring trends in the data. We can then use these values for selecting the features to employ in our model.

Since our target variable is categorical, thus we cannot compute its correlation coefficient.

Heat map:

A heat map is a graphical representation of data that uses a system of color-coding to represent different values. We mostly use heat map to visualize correlations between (continuous) features.

Following is a heat map showing correlations between continuous features in the data:



Following are the inferences that we derived:

- Highly collinear features have a significant correlation coefficient between them. In our dataset, Following features show high multicolinearity:
  - column 'euribor3m' and 'emp.var.rate' are highly correlated i.e. 0.97
  - column 'euribor3m' and 'nr.employed' are highly correlated i.e. 0.95
  - column 'emp.var.rate' and 'cons.pri.idx' are corrrelated i.e. 0.78
  - column 'pdays' and 'previous' are correlated i.e. 0.59

Thus, in order to remove multicolinearity, out of the 3 highly collinear features (correlation > 0.7) we would drop „emp.var.rate“ for model prediction , since it has lowest correlation coefficient with the target in comparison to „euribor3m“ and „nr.employed“.

Further, pdays column would be used for feature extraction (discussed ahead), so it“s correlation with previous would not pose a problem.

3.3. Null Values Imputation

Following are the number of null values per column:

job	330
marital	80
education	1731
default	8597
housing	990
loan	990

- **Filling Missing Values in 'job':**

Job column has the following classes:

['housemaid', 'services', 'admin.', 'blue-collar', 'technician', 'retired', 'management', 'unemployed', 'self-employed', 'entrepreneur', 'student']

Pattern 1:

Taking a hint from the official retirement age in portugal as 66, we investigated further to find out the distribution of age group for „retired" job category. We observed that most of the people are choosing to retire after the age of 56 (75 percentile of retired customers are above this age group).

```
retired          0.379397
admin.           0.141820
blue-collar      0.122278
management       0.082915
technician       0.082077
housemaid        0.051089
services         0.045226
entrepreneur     0.033222
self-employed    0.029313
unemployed       0.015913
Name: age, dtype: float64
```

Following is the percentage distribution of job types for customers above the age of 56 yrs:

So we imputed the unknown jobs for the age of above 56 years old customers to 'retired'.

Pattern 2:

On the same lines, we tried to verify our intuition that job type: 'student' would mostly be for young customers by finding out the age distribution for 'student' job type, as follows:

```
count    875.000000
mean      25.894857
std        4.991334
min       17.000000
25%       22.000000
50%       25.000000
75%       29.000000
max       47.000000
Name: age, dtype: float64
```

We observed that most of the customers aged below 25 years, had job type „student".

```
student          0.273824
blue-collar      0.197226
admin.           0.189988
services         0.148372
technician       0.103136
self-employed    0.024125
management       0.022919
unemployed       0.021110
entrepreneur     0.012063
housemaid        0.006634
retired          0.000603
Name: job, dtype: float64
```

So we imputed the unknown jobs for the customers of age below 25 years to „student".

Pattern 3:

On observing correlation between education and job in the data which also resonates with our intuition we used this pattern to help us impute null values. So we computed the most frequent job type as per each education class and imputed the missing values based on it:

	education	job
0	basic.4y	blue-collar
1	basic.6y	blue-collar
2	basic.9y	blue-collar
3	high.school	admin.
4	illiterate	blue-collar
5	professional.course	technician
6	university.degree	admin.

Pattern 4:

We were still left with some null values in 'job' column because education for those observations was also NaN. Thus we imputed the remaining missing values by maintaining proportion of job classes as per the complete data.

Filling Missing Values in 'marital':

'Marital' column has the following classes:

['married', 'single', 'divorced']

Observing the relation between age and marital status, our basic intuition says that age and marital status should be related to each other which could help us impute null values. So we computed the most frequent marital status as per each age_group (new feature created to convertage from numerical to categorical field) and imputed the missing values based on it:

	age_cat	marital
0	Young	single
1	Adult	married
2	Mature	married
3	Old	married

Filling Missing Values in 'education':

Education column has the following classes:

['basic.4y', 'high.school', 'basic.6y', 'basic.9y', 'professional.course', 'university.degree', 'illiterate']

We imputed the missing values in education by exploiting the relation between „education" and „job". We computed mode of „education" as per each „job" type and filled in missing values using this pattern.

	job	education
0	admin.	university.degree
1	blue-collar	basic.9y
2	entrepreneur	university.degree
3	housemaid	basic.4y
4	management	university.degree
5	retired	basic.4y
6	self-employed	university.degree
7	services	high.school
8	student	high.school
9	technician	professional.course
10	unemployed	university.degree

Filling Missing Values in 'default':

Since default column contains 99.9 % 'no' .Thus, we simply imputed missing values with 'no'. Further, since this column captures almost no variation (99.9% „no" implies almost every column has the same value), thus we may even consider dropping it for modeling.

Filling Missing Values in 'housing' and 'loan':

We imputed the missing values in housing and loan by maintaining the proportion of their respective classes as per the complete data.

Feature Engineering

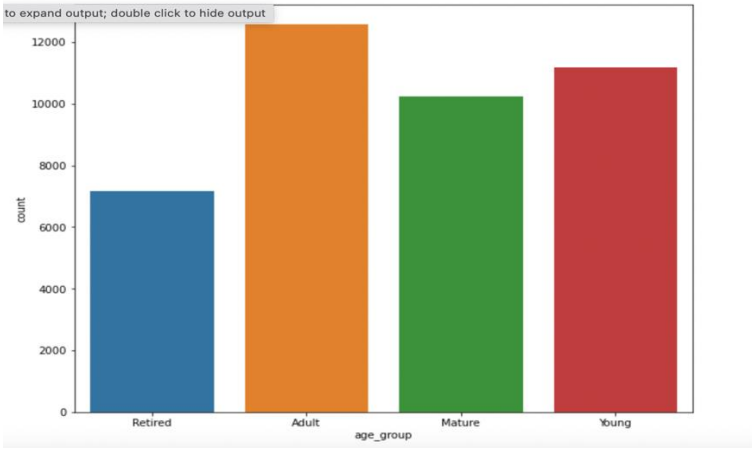
4.1. Data Transformation

Feature Engineering: The process of taking raw data and extracting or creating new features that allow a machine learning model to learn a mapping between these features and the target. This might mean taking transformations of variables, such as we do with the log and square root, or one-hot encoding categorical variables so they can be used in a model. Generally, feature engineering as additional features derived from the raw data.

Now that we have explored the trends and relationships within the data, we worked on engineering a set of features for our models. In particular, we learned the following from EDA which can help us in engineering/selecting features:

- **Age**: is not contributing significantly to term deposit, but intuitively it can be one of the important features. This might be due to specificity in age numbers. So we decided to bin age into different age categories to generalize the data to get the more insights about the data.

```
Adult      12592
Young      11176
Mature     10240
Retired     7180
Name: age_group, dtype: int64
```



- **Duration**: last contact duration is mentioned in seconds. So, we converted it into minutes in order to reduce the bandwidth of duration values which would make them more comprehensible.

- **P-Days**: implies number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted). This feature can be used to create a new generalized feature- whether the customer was contacted for previous campaign or not i.e., (if pdays=999 then 0 else 1)

4.2. Feature Selection using Backward Elimination and Statistical Tests

➢ Backward Feature Elimination using p-value approach :

Backward feature elimination is a stepwise regression approach, that begins with a full (saturated) model and at each step gradually eliminates variables from the regression model to find a reduced model that best explains the data.

Below is the list of variables obtained after performing backward feature selection on our data using pvalue approach (ie removing feature with maximum pvalue one by one while recursively fitting the model):

	coef	std err	z	P>|z|	[0.025	0.975]
age	0.0055	0.002	2.919	0.004	0.002	0.009
marital	0.0748	0.037	2.010	0.044	0.002	0.148
education	0.0612	0.010	6.061	0.000	0.041	0.081
contact	-0.7010	0.059	-11.835	0.000	-0.817	-0.585
month	-0.1140	0.008	-13.684	0.000	-0.130	-0.098
day_of_week	0.0565	0.015	3.891	0.000	0.028	0.085
duration	0.2747	0.004	63.100	0.000	0.266	0.283
campaign	-0.0335	0.011	-2.932	0.003	-0.056	-0.011
pdays	0.8605	0.090	9.515	0.000	0.683	1.038
poutcome	0.5196	0.053	9.745	0.000	0.415	0.624
emp.var.rate	-0.9216	0.063	-14.596	0.000	-1.045	-0.798
cons.price.idx	0.6834	0.030	22.744	0.000	0.625	0.742
cons.conf.idx	0.0193	0.004	4.324	0.000	0.011	0.028
euribor3m	0.6601	0.073	9.003	0.000	0.516	0.804
nr.employed	-0.0136	0.001	-22.842	0.000	-0.015	-0.012
age_cat	0.0982	0.018	5.369	0.000	0.062	0.134

➢ Statistical Tests :

A **statistical test** is a way to evaluate the evidence the data provides against a hypothesis.

This hypothesis is called the **null hypothesis** and is often referred to as **H0**. H0 is usually opposed to a hypothesis called the **alternative hypothesis**, referred to as **H1.**

For Example-
H0: Corn fields submitted to fertilizers A, B, C or D produce equivalent yields.
H1: at least one fertilizer induces a difference in corn yield.

How to interpret the output of a statistical test: the significance level alpha and the p-value?

When setting up a study, a risk threshold above which H0 should not be rejected must be specified. This threshold is referred to as the **significance level alpha** and should lay between 0 and 1. Low alpha's are more conservative. The choice of alpha should depend on how dangerous it is to reject H0 while it is true. For example, in a study aiming at demonstrating the benefits of a medical treatment, alpha should be low. On the other hand, when screening the effects of many attributes on the appreciation of a product, alpha's could be more moderate. Very often, alpha is set at 0.05 or 0.01 or 0.001.

The statistical test produces a number called p-value (that is also bounded between 0 and 1). The p-value is the probability of obtaining the data or more extreme data under the null hypothesis.
More practically, the p-value should be compared to alpha:

- If **p-value < alpha**, we reject H0 and accept H1.
- If **p-value > alpha**, we fail to reject H0.

In this dataset, we take alpha value as 0.05.

Different types of statistical tests

There are many types of statistical test:

1. Z-Test
2. T-Test
3. ANOVA(Analysis of Variance) Test
4. Chi-Square Test

In this dataset, we have the list of **categorical columns** namely as ['job', 'marital', 'education', 'default', 'housing', 'loan', 'contact', 'month', 'day_of_week', 'poutcome', 'y'] and **numerical columns** as ['age', 'duration', 'campaign', 'pdays', 'previous', 'emp.var.rate', 'cons. price.idx', 'cons.conf.idx', 'euribor3m', 'nr.employed'].
We are using ANOVA Test for numerical features and Chi-Square Test for categorical features with respect to the target variable.

Findings of the Statistical Test with respect to Numerical Features-

All numerical features have a significant effect on the target variable.

Findings of the Statistical Test with respect to Categorical Features-

Except 'default' and 'loan', all categorical features have a significant effect on the target variable.

Insights of the Statistical Tests-

1. **Age Category:** Most of the clients who say 'yes' for term subscription they belong to YOUNG(13%) and RETIRED(15%) category. Clients who say 'yes' for the term deposit subscription have MORE mean age than compared to the clients who say 'no' for the term deposit subscription.

y	no	yes
age_group		
Adult	0.905257	0.094743
Mature	0.918262	0.081738
Retired	0.849304	0.150696
Young	0.863278	0.136722

```
P_value is 6.739434944311709e-10
```

2. Job: Most of the clients who say 'yes' for term subscription they are working as RETIRED(25%) and STUDENT(31%).

y	no	yes
job		
admin.	0.870772	0.129228
blue-collar	0.931391	0.068609
entrepreneur	0.914835	0.085165
housemaid	0.900000	0.100000
management	0.887825	0.112175
retired	0.748315	0.251685
self-employed	0.895144	0.104856
services	0.918619	0.081381
student	0.681767	0.318233
technician	0.891820	0.108180
unemployed	0.857988	0.142012

```
job
Chi2 198.53256818316754
PValue 4.365802887818895e-45
DOF 1
```

3. Education: Most of the clients who say 'yes' for term subscription they are ILLITERATE(22%) and UNIVERSITY.DEGREE(14%).

y	no	yes
education		
basic.4y	0.890294	0.109706
basic.6y	0.917976	0.082024
basic.9y	0.923607	0.076393
high.school	0.887205	0.112795
illiterate	0.777778	0.222222
professional.course	0.886459	0.113541
university.degree	0.862765	0.137235

```
education
Chi2 6.957703850566789
PValue 0.008345907290855374
DOF 1
```

4. Housing: There is a **good balance** between the clients who say 'yes' for term subscription as having clients housing loan(~11%) and clients not having house loan(~10%).

y	no	yes
housing		
no	0.893277	0.106723
yes	0.882233	0.117767

```
housing
Chi2 0.0003259765284992322
PValue 0.9855951204147657
DOF 1
```

5. Contact: Most of the clients who say 'yes' for term subscription their type of contact is CELLULAR (15%).

y	no	yes
contact		
cellular	0.852624	0.147376
telephone	0.947687	0.052313

```
contact
Chi2 862.3183642075705
PValue 1.5259856523129964e-189
DOF 1
```

6. Month: Most of the clients who say 'yes' for term subscription they are last contacted in the month of MARCH(51%) and DECEMBER(49%).

y	no	yes
month		
apr	0.795213	0.204787
aug	0.893979	0.106021
dec	0.510989	0.489011
jul	0.909534	0.090466
jun	0.894885	0.105115
mar	0.494505	0.505495
may	0.935653	0.064347
nov	0.898561	0.101439
oct	0.561281	0.438719
sep	0.550877	0.449123

```
month
Chi2 152.82925780495017
PValue 4.1743464815318017e-35
DOF 1
```

7. Day of Week: Most of the clients who say 'yes' for term subscription they are contacted in TUESDAY (12%) and THURSDAY (12%).

y	no	yes
day_of_week		
fri	0.891913	0.108087
mon	0.900517	0.099483
thu	0.878812	0.121188
tue	0.882200	0.117800
wed	0.883329	0.116671

```
day_of_week
Chi2 3.158825118304431
PValue 0.07551751375783097
DOF 1
```

8. POutcome: Most of the clients who say 'yes' for term subscription their last campaign outcome was SUCCESS (65%).

y	no	yes
poutcome		
failure	0.857714	0.142286
nonexistent	0.911678	0.088322
success	0.348871	0.651129

```
poutcome
Chi2 129.12809264596456
PValue 6.357994063290516e-30
DOF 1
```

9. Marital: Most of the clients who say 'yes' for term subscription they are having SINGLE (14%) as marital status.

y	no	yes
marital		
divorced	0.896791	0.103209
married	0.898427	0.101573
single	0.859890	0.140110

```
marital
Chi2 0.09677292636058074
PValue 0.7557371658860872
DOF 1
```

10. Duration: Clients who say 'yes' for the term deposit subscription have MORE mean of last call duration of clients than compared to the clients who say 'no' for the term deposit subscription.

11. Campaign: Clients who say 'yes' for the term deposit subscription have slightly LESS mean number of contacts performed to a particular client than compared to the clients who say 'no' for the term deposit subscription.

12. PDays: Clients who say 'yes' for the term deposit subscription have MORE mean number of days that passed by after the client was last contacted from a previous campaign than compared to the clients who say 'no' for the term deposit subscription.

13. Previous: Clients who say 'yes' for the term deposit subscription have slightly MORE number of contacts performed before this campaign for a particular client than compared to the clients who say 'no' for the term deposit subscription.

14. emp.var.rate: Clients who say 'yes' for the term deposit subscription have slightly LESS mean employment variation rate than compared to the clients who say 'no' for the term deposit subscription.

15. cons.price.idx: Clients who say 'yes' for the term deposit subscription have slightly LESS mean consumer price index than compared to the clients who say 'no' for the term deposit subscription.

16. cons.conf.idx: Clients who say 'yes' for the term deposit subscription have slightlyMORE mean consumer confidence index than compared to the clients who say 'no' for the term deposit subscription.

17. euribor3m: Clients who say 'yes' for the term deposit subscription have LESS mean euribor 3 month rate than compared to the clients who say 'no' for the term deposit subscription.

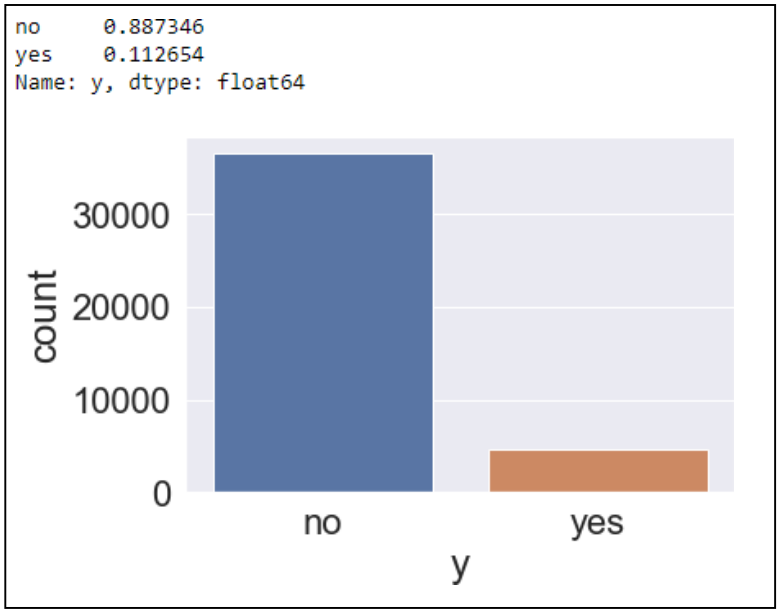
18. nr.employed: Clients who say 'yes' for the term deposit subscription have less mean number of employees than compared to the clients who say 'no' for the term deposit subscription.

Chapter 5

Imbalanced Data Treatment

5.1. Imbalanced Data Treatment

The Data we have at hand is highly imbalanced as stated above. The majority of the class is the one where the customers are not willing to subscribe to the fixed term deposits based on the most recent marketing campaign. The imbalance in the data is most probably going to lead in having the predictions tilted towards one particular class -



As we can see from the above count plot that 88% of the customers are not willing to subscribe to the fixed term deposit and only 12% of the customers actually subscribed to the fixed term deposits.

The first most important thing to do after identifying the imbalance in the data is to choose the correct metric in order to check the prediction power of our model. Since almost 88% of the data is already heavily bent towards the 0 class the accuracy of our model is naturally going to be high. So we would be using the f1 score metric in order to judge the prediction power of our model.

F1 Score is equal to 2 * ((precision * recall)/(precision + recall)). It is also called the Precision-Recall Score or the F Measure. Put another way, the F1 score conveys the balance be- tween the precision and the recall which would tell us how well the model is predicting the True Positives as well as the True Negatives.

According to the problem statement, 'yes'- i.e., customers subscribing to term deposit is an important target class for us to filter out potential customers along with devising new marketing plans for class „no"- i.e., customers not subscribing to term deposits. Hence, F1 Score is going to be a better metric for us to test the prediction power of our dataset and checking howwell the model can classify whether a customer will be subscribing to our bank's product or not.

Let's see how the model is predicting on the cleaned data without treating the imbalance in the data –

	precision	recall	f1-score	support
0	0.93	0.97	0.95	10969
1	0.67	0.43	0.52	1388
accuracy			0.91	12357
macro avg	0.80	0.70	0.74	12357
weighted avg	0.90	0.91	0.90	12357

We can see that the f1 score is at 52% and the accuracy is 91% (as explained above) now let's apply various imbalance data treatment techniques to see how the model is reacting to them.

5.2. How to treat Imbalanced Data

5.2.1 Over-Sampling- SMOTE

A technique similar to up sampling is to create synthetic samples. Here we will use imblearn's SMOTE or Synthetic Minority Oversampling Technique. SMOTE uses a nearest neighbors algorithm to generate new and synthetic data we can use for training our model.

Again, it's important to generate the new samples only in the training set to ensure our model generalizes well to unseen data.

Important Note -

Always split into test and train sets BEFORE trying oversampling techniques! Oversampling before splitting the data can allow the exact same observations to be present in both the test and train sets. This can allow our model to simply memorise specific data points and cause overfitting and poor generalisation to the test data. –

	precision	recall	f1-score	support
0	0.96	0.90	0.93	10969
1	0.49	0.73	0.59	1388
accuracy			0.88	12357
macro avg	0.73	0.82	0.76	12357
weighted avg	0.91	0.88	0.89	12357

We can see that the model is performing better in terms of both f1 score where it has jumped up to 59%.

5.2.2. Under-Sampling-

Under-sampling can be defined as removing some observations of the majority class. Under-sampling can be a good choice when you have a ton of data -think millions of rows. But a drawback is that we are removing information that may be valuable. This could lead to un- der-fitting and poor generalization to the test set.

We will again use the resampling module from Scikit-Learn to randomly remove samples from the majority class.

Since our data has high amount 0s and very little 1s we would down sample the rows containing the target variable as 0s and make the data somewhat balanced.

	precision	recall	f1-score	support
0	0.97	0.80	0.88	8767
1	0.35	0.82	0.49	1118
micro avg	0.80	0.80	0.80	9885
macro avg	0.66	0.81	0.68	9885
weighted avg	0.90	0.80	0.83	9885

Let's have a comparison amongst both the techniques to finalize the approach to be used for our model building –

	Technique	Accuracy Score	F1 Score
0	UnderSampling (Nearmiss)	0.804047	0.486615
1	Oversampling (SMOTE)	0.863935	0.583978

Of all the methods we tried SMOTE is working the best and we could see an increase in the f1score after introducing the SMOTE data. But we should tread lightly while using the resample techniques as oversampling the data could introduce bias to our data and under-sampling could actually make the data too specific.

Although these are the available options to actually tackle the imbalance data problem there are various drawbacks of the same which should also be taken into consideration - There are known disadvantages associated with the use of sampling to implement cost-sensitive learning. **The disadvantage with under-sampling is that it discards potentially useful data**. The main **disadvantage with oversampling, from our perspective, is that by making exact copies of existing examples, it makes overfitting likely**. In fact, with oversampling it is quite common for a learner to generate a classification rule to cover a single, replicated, example. A second disadvantage of oversampling is that it increases the number of training examples, thus increasing the learning time.

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Chapter 6

Model Building

6.1. Model Building

There are various machine learning algorithms which we can use to actually make predictions on the given data set but which would work the best is almost impossible to say without having a comparison amongst them. In this section we will build, train, and evalute several ma- chine learning methods for our supervised regression task. The objective is to determine which model holds the most promise for further development (such as hyperparameter tuning).

6.1.1. Model Selection

We will compare five different machine learning models using the great [Scikit-Learn li-brary](http://scikit-learn.org/stable/):

1. Logistic Regression
2. Knn Classifier
3. Decision Tree Classifier
4. Random Forest Classifier
5. XGboost Classifier

To compare the models, we are going to be mostly using the Scikit-Learn defaults for the model hyperparameters. Generally these will perform decently, but should be optimized be- fore actually using a model. At first, we just want to determine the baseline performance of each model, and then we can select the best performing model for further optimization using hyperparameter tuning. Remember that the default hyperparameters will get a model up and running, but nearly always should be adjusted using some sort of search to find the best set-tings for your problem!

Here is what the Scikit-learn documentation [says about the de-faults](https://arxiv.org/abs/1309.0238):

> Sensible defaults: Whenever an operation requires a user-defined parameter,an
>
> appropriate default value is defined by the library. The default value
>
> should cause the operation to be performed in a sensible way (giving a baseline
>
> solution for the task at hand.)

One of the best parts about scikit-learn is that all models are implemented in an identical manner: once you know how to build one, you can implement an extremely diverse array of models. Here we will implement the entire training and testing procedures for a number of models in just a few lines of code.

1. **Logistic Regression –** Logistic regression is the basic classification model which uses the sigmoidal function to classify the new data point within a binary classification 1 and 0.

	precision	recall	f1-score	support
0	0.97	0.89	0.93	10969
1	0.46	0.77	0.58	1388
accuracy			0.87	12357
macro avg	0.71	0.83	0.75	12357
weighted avg	0.91	0.87	0.89	12357

2. **KNN Classifier** – K Nearest Neighbor is an algorithm which works on the Nearest Neighbors it is actually known as the lazy learned since it learns nothing in training and actually predict as the data points are introduced to the model based on the nearest neighbors. Below is the classification report and f1 score for the model –

	precision	recall	f1-score	support
0	0.96	0.89	0.92	10969
1	0.45	0.72	0.55	1388
accuracy			0.87	12357
macro avg	0.71	0.80	0.74	12357
weighted avg	0.90	0.87	0.88	12357

3. **Decision Tree Classifier** - The decision tree induction algorithm works by recursively selecting the best attribute to split the data and expanding the leaf nodes of the tree until the stopping criterion is met. The choice of best split test condition is determinedby comparing the impurity of child nodes and also depends on which impurity measurement is used. –

	precision	recall	f1-score	support
0	0.96	0.89	0.92	10969
1	0.43	0.67	0.52	1388
accuracy			0.86	12357
macro avg	0.69	0.78	0.72	12357
weighted avg	0.90	0.86	0.88	12357

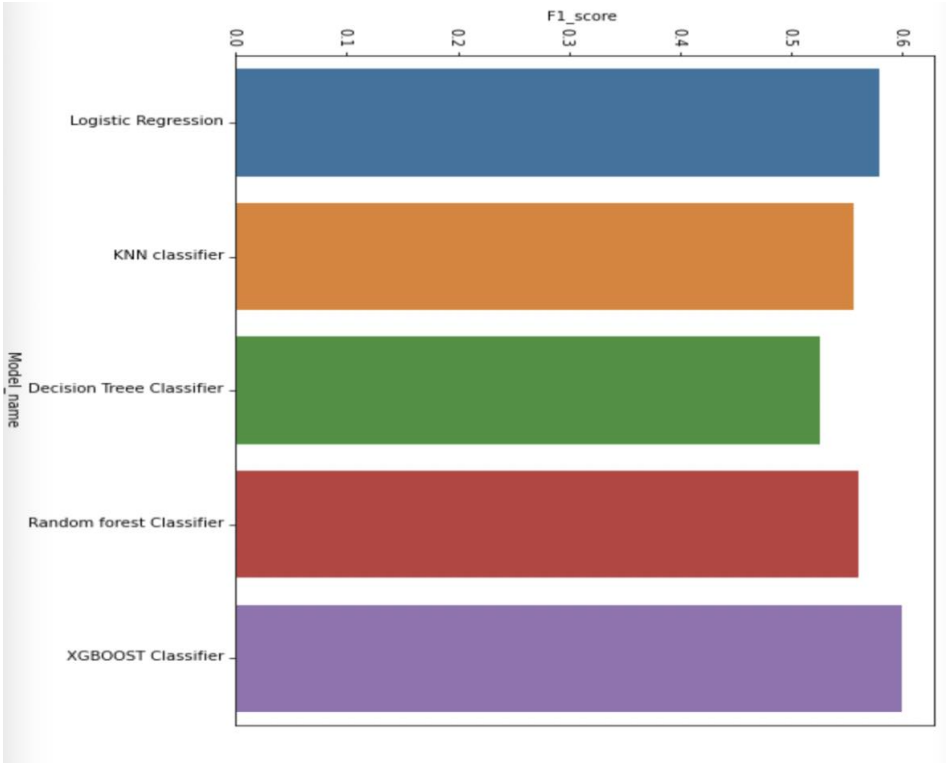
4. **Random Forest Classifier**- A random forest is a meta estimator and an ensemble bagging technique that fits a number of decision tree classifiers on various sub- samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

	precision	recall	f1-score	support
0	0.96	0.90	0.93	10969
1	0.47	0.68	0.56	1388
accuracy			0.88	12357
macro avg	0.72	0.79	0.74	12357
weighted avg	0.90	0.88	0.89	12357

5. **XGBOOST Classifier** - XGBoost stands for eXtreme Gradient Boosting XGBoost is an implementation of gradient boosted decision trees designed for speed and performance, here is the classification report and the f1 score for the same –

	precision	recall	f1-score	support
0	0.97	0.90	0.93	10969
1	0.50	0.75	0.60	1388
accuracy			0.89	12357
macro avg	0.73	0.83	0.77	12357
weighted avg	0.91	0.89	0.90	12357

Comparing all the models based on the f1 score metric –



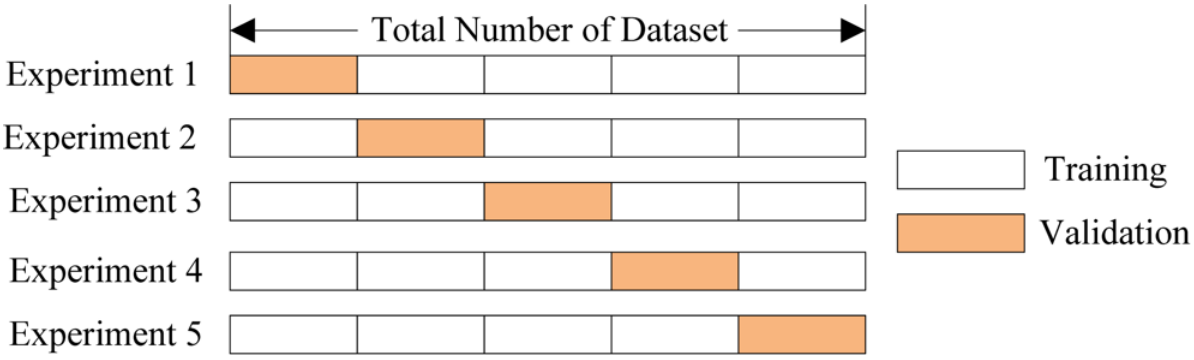
Since of all the models applied above XGBoost classifier is working the best for us and giving the most optimum f1 score we will be using the XGBoost model as our final data and tune the hyper parameters of the same using the scikit library GridSearch.

6.2 Hyper Parameter Tuning

The best way to think about hyper-parameters is like the settings of an algorithm that can be adjusted to optimise performance In the case of Logistic Regression, hyper-parameters include the number of decision trees in the forest and the number of features considered byeach tree when splitting a node. (The parameters of a random forest are the variables and thresholds used to split each node learned during training). Scikit-Learn implements a set of sensible default hyper-parameters for all models, but these are not guaranteed to be optimal for a problem. The best hyper-parameters are usually impossible to determine ahead of time, and tuning a model is where machine learning turns from a science into trial-and-error based engineering.

If we optimise the data of the and hyper-tune the training model there are chances that the model is going to over fit on the training set but isn't going to perform just as good on the testing data set this is known as the problem of overfitting to overcome this problem there is concept known as cross validation

The technique of cross validation (CV) is best explained by example using the most common method, K-Fold CV. When we approach a machine learning problem, we make sure to split our data into a training and a testing set. In K-Fold CV, we further split our training set into K number of subsets, called folds. We then iteratively fit the model K times, each time training the data on K-1 of the folds and evaluating on the Kth fold (called the validation data)



Here are various other parameters that we would be using and tuning for our model –

- **Dual** = Dual or primal formulation. Dual formulation is only implemented for l2 penalty with liblinear solver. Prefer dual=False when n_samples > n_features.
- **Learning Rate =** Learning rate is a configurable hyperparameter used in the training of neural networks that has a small positive value, often in the range between 0.0 and 1.0.
- **Max_depth =** The maximum depth can be specified in the XGBClassifier and XGBRegressor wrapper classes for XGBoost in the max_depth parameter. This parameter takes an integer value and defaults to a value of 3. We can tune this hyperparameter of XGBoost using the grid search infrastructure in scikit-learn on the Otto dataset.
- **Gamma =** Gamma is dependent on both the training set and the other parameters you use. It is a pseudo-regularization hyperparameter in gradient boosting . Mathematically you call "Gamma" the "Lagrangian multiplier" (complexity control). The higher Gamma is, the higher the regularization.

Since our data set is not that huge we would be using Grid Search CV in order to find out the best parameters in and around the default parameters.

```
Best parameters for XGBoost classifier:  {'gamma': 0, 'learning_rate': 0.5, 'max_depth': 9}
```

Above are the best hyper parameters we got after using GridSearchCV on XGBoost Regression.

Now we run the above and then analyse the results of the same on the training data set –

	precision	recall	f1-score	support
0	0.96	0.91	0.93	10969
1	0.49	0.73	0.59	1388
accuracy			0.89	12357
macro avg	0.73	0.82	0.76	12357
weighted avg	0.91	0.89	0.89	12357

The f1 score we are getting is 59% and the classification report has over all improved quite a lot. Now we would be using the same model on the Test Set to check the actual accuracy of our model in production on unseen data.

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Chapter 7

We until now are through with the Data cleaning and formatting, Exploring the data, Feature engineering and selection, comparing various models on performance metrics and finally performing the hyper parameter tuning on the final selected model, we will now run the model on the test data set and then judge the model's prediction power basis the data which is unseen. If our model gives stable results in production on unseen data only then we would be able to deploy the model in the live environment.

We are going to show the final model on the test set with techniques like fitting the best model on the SMOTE infused data.

- **Final Model with SMOTE-**

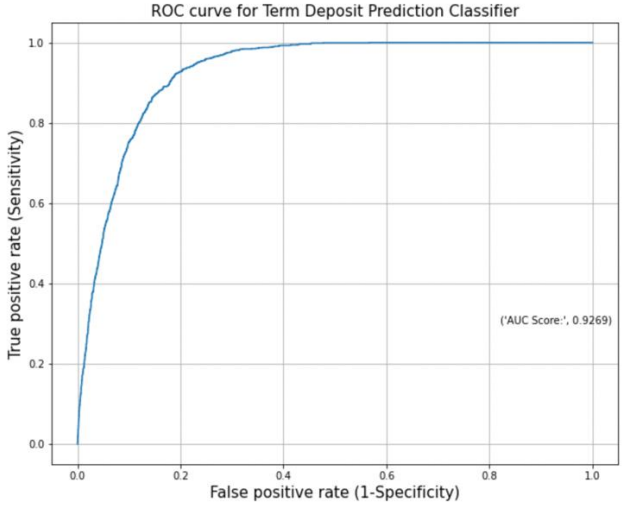
	precision	recall	f1-score	support
0	0.96	0.91	0.93	10969
1	0.49	0.73	0.59	1388
accuracy			0.89	12357
macro avg	0.73	0.82	0.76	12357
weighted avg	0.91	0.89	0.89	12357

```
array([[9874, 1095],
       [ 326, 1062]], dtype=int64)
```

We get an f1 score of 0.59 and accuracy of 0.89.

Choosing the best threshold using the ROC curve

- A common way to visualize the trade-offs of different thresholds is by using an ROC curve, a plot of the true positive rate (true positives/ total positives) versus the false positive rate (false positives /total negatives) for all possible choices of thresholds.
- A model with good classification accuracy should have significantly more true positives than false positives at all thresholds.
- The optimum position for roc curve is towards the top left corner.



Final Model Performance -

Since Model built by lowering the threshold is performing better than model with SMOTE (for handling imbalanced data), thus we will proceed with threshold shifted XGBoost classifier model as our final model.

	precision	recall	f1-score	support
0	0.96	0.91	0.93	10969
1	0.49	0.73	0.59	1388
accuracy			0.89	12357
macro avg	0.73	0.82	0.76	12357
weighted avg	0.91	0.89	0.89	12357

```
array([[9874, 1095],
       [ 326, 1062]], dtype=int64)
```

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

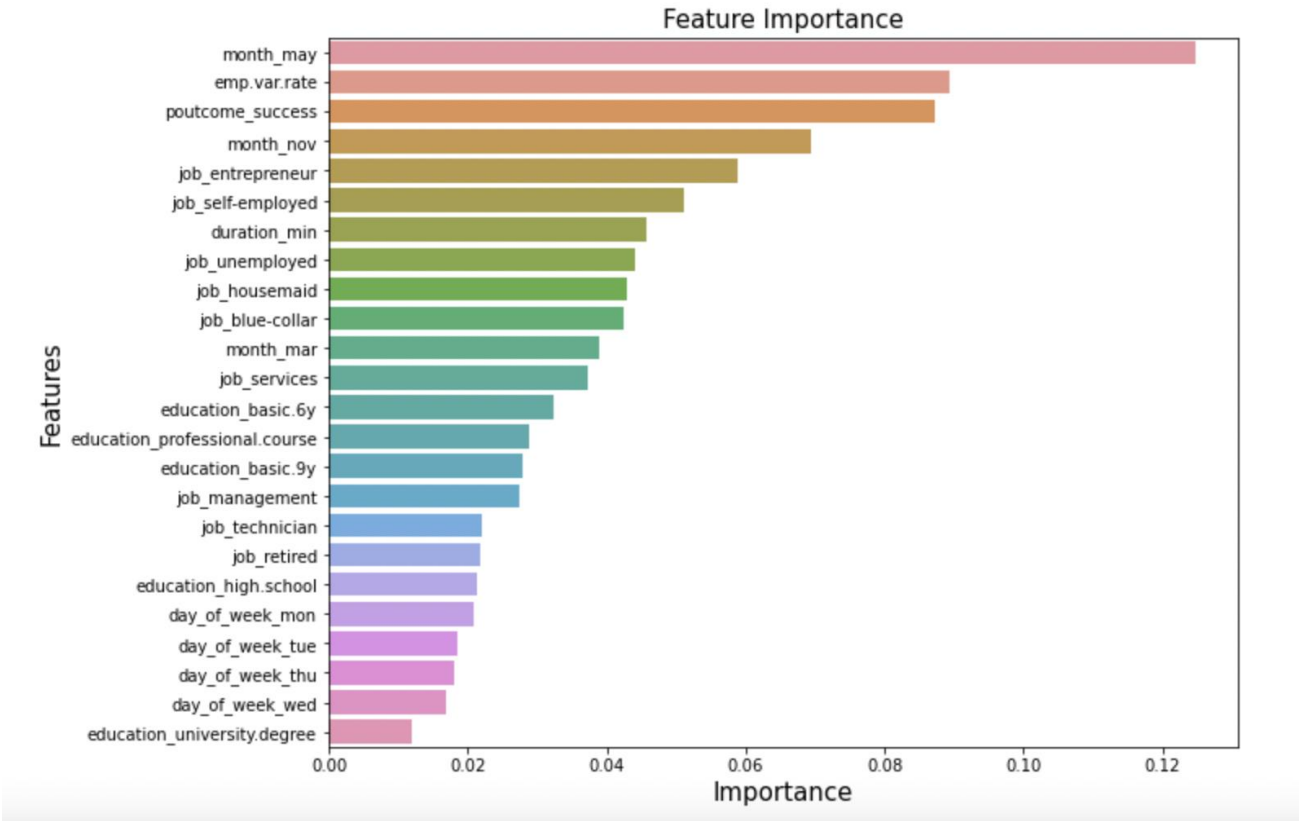
Interpreting the Model

Machine learning is often criticized as being a black-box: we put data in on one side and it gives us the answers on the other. While these answers are often extremely accurate, the model tells us nothing about how it actually made the predictions. This is true to some extent, but there are ways in which we can try and discover how a model "thinks".

We will explore several ways to interpret our model:

* Feature importance
* Interpreting the results using statistics

- **Feature Importance -**
  One of the basic ways we can interpret an ensemble of decision trees is through what are known as the feature importances. These can be interpreted as the variables which are most predictive of the target.



We calculated the feature importances using Decision Tree Extracting the feature importances from a trained ensemble of trees is quite easy in scikit-learn. We will store the feature importances in a dataframe to analyze and visualize them. Decision Tree Feature Importances shows similar results as logistic regression coefficeint interpretation. It is apparent that duration is the most important feature of the given data set and business problem.

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Conclusions Findings and Inferences

1. Using the given Bank Marketing data, our machine learning model has increased the base-line model performance from:
F1-score = 0.52 to f1_score: 0.59

2. The most important variables for determining the term deposit subscription are: month_may,emp_var_rate and poutcome_sucess.

3. From the statistical tests performed on 'job', we can infer that most of the clients who say 'yes' for term subscription they are working as **RETIRED(25%)** and **STUDENT(31%)**.

4. From the statistical tests performed on 'education', we can infer that most of the clients who say 'yes' for term subscription they are having **ILLITERATE (22%)** and **UNIVERSITY.DEGREE (14%)**

5. From the statistical tests performed on 'marital_status‟, we can infer that most of the clients who say 'yes' for term subscription they are having **SINGLE(14%)** as marital status.

6. From the statistical tests performed on 'contact' type, we can infer that most of the clients who say 'yes' for term subscription their type of contact is **CELLULAR (15%)**.

7. From the statistical tests performed on 'month', we can infer that most of the clients who say 'yes' for term subscription they are last contacted in the month of **MARCH (51%)** and **DECEMBER (49%)**.

8. Clients who say 'yes' for term subscription they are contacted in **TUESDAY** and **THURSDAY**.

9. From the statistical tests performed on 'poutcome', we can infer that the clients who say 'yes' for term subscription their last campaign outcome was **SUCCESS (65%)**.

10. For Bank to focus on future marketing campaigns, it should target above mentioned class of customers for better success rate

The accuracies and prediction power of the model can be further improved with time as we gather more data and we have more data points to work with regards to the customers who subscribed to the fixed term deposits based on the marketing campaign.

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

References

Aptéa, C. and Weiss, S. 1997. "Data mining with decision trees and decision rules", Future Generation Computer Systems 13, No.2-3, 197–210.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. 2000. CRISP-DM 1.0 - Step-by-step data mining guide, CRISP-DM Consortium.

Coppock, D. 2002. Why Lift? – Data Modeling and Mining, Information ManagementOnline (June).

Cortes, C. and Vapnik, V. 1995. "Support Vector Networks", Machine Learning 20, No.3, 273–297.

Cortez, P. 2010. "Data Mining with Neural Networks and Support Vector Machines using the R/rminer Tool". In Proceedings of the 10th Industrial Conference on Data Mining (Berlin, Germany, Jul.). Springer, LNAI 6171, 572– 583.

Cortez, P. and Embrechts, M. 2011. "Opening Black Box Data Mining Models Using Sensitivity Analysis". In Proceedings of IEEE Symposium on Computational Intelligence and Data Mining (Paris, France), 341-348.

Fawcett, T. 2005. "An introduction to ROC analysis", Pattern Recognition Letters 27, No.8, 861–874.

Hu, X. 2005, "A data mining approach for retailing bank customer attrition analysis", Applied Intelligence 22(1):47- 60.

Kohavi, R. and Provost, F. 1998. "Glossary of Terms", Machine Learning 30, No.2–3, 271–274.

Ling, X. and Li, C., 1998. "Data Mining for Direct Marketing: Problems and Solutions". In Proceedings of the 4th KDD conference, AAAI Press, 73–79.

Li, W., Wu, X., Sun, Y. and Zhang, Q., 2010. "Credit Card Customer Segmentation and Target Marketing Based on Data Mining", In Proceedings of International Conference on Computational Intelligence and Security, 73-76.

Ou, C., Liu, C., Huang, J. and Zhong, N. 2003. "On Data Mining for Direct Marketing". In Proceedings of the 9th RSFDGrC conference, 2639, 491–498.

Page, C. and Luding, Y., 2003. "Bank manager‟s direct marketing dilemmas – customer‟s attitudes and purchase intention". International Journal of Bank Marketing 21, No.3, 147– 163.

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

Bibliography

https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

https://machinelearningmastery.com/calculate-bootstrap-confidence-intervals-machine-learning-results-python/

https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/#:~:targetText=Backward%20Elimination%3A%20In%20backward%20elimination,observed%20on%20removal%20of%20features.

https://towardsdatascience.com/methods-for-dealing-with-imbalanced-data-5b761be45a18

GREAT LAKES
INSTITUTE OF MANAGEMENT, GURGAON
Global Mindset - Indian Roots

ANNEXURE

➢ Data Source:

https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

➢ Data Dictionary:

- Input variables:
  # bank client data:
  1 - age (numeric)
  2 - job : type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')
  3 - marital : marital status (categorical: 'divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)
  4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')
  5 - default: has credit in default? (categorical: 'no','yes','unknown')
  6 - housing: has housing loan? (categorical: 'no','yes','unknown')
  7 - loan: has personal loan? (categorical: 'no','yes','unknown')
  # related with the last contact of the current campaign:
  8 - contact: contact communication type (categorical: 'cellular','telephone')
  9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
  10 - day_of_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')
  11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.
  # other attributes:
  12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
  13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)
  14 - previous: number of contacts performed before this campaign and for this client (numeric)
  15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')
  # social and economic context attributes
  16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)
  17 - cons.price.idx: consumer price index - monthly indicator (numeric)
  18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)
  19 - euribor3m: euribor 3 month rate - daily indicator (numeric)
  20 - nr.employed: number of employees - quarterly indicator (numeric

- Output variable (desired target):
  21 - y - has the client subscribed a term deposit? (binary: 'yes','no')