CIS 563: Introduction to Data Science
Term Project Report

" Predict whether client subscribes long term deposit through telemarketing "

Instructor

Prof. Reza Zafarani

Compiled by

Prathma Rastogi

409745940

Table of Contents

1. INTRODUCTION

Marketing to potential clients has always been a challenging task in attaining success for the banking institutions. Telemarketing is an interactive technique of direct marketing via phone calls which are widely used by banks to sell long term deposits. With the inception of machine learning, reaching out to specific groups of people has been revolutionized by using data and analytics.
In this study, the bank telemarketing data used here is directly related with direct marketing campaigns of the Portugese bank institution.
I have performed extensive modeling experiments on the data using several supervised machine learning algorithms for binary classification. Through this I will explain how the Portugese bank can use predictive analysis to help prioritize the clients who would subscribe to long term deposits.

2. PRIOR WORK

The analysis on telemarketing data has always resulted in increasing campaign efficiency[1] by identifying the main factors that can affect the success of a campaign. It can help in predicting the type of client interested in subscribing the service and determining how the campaign can be successful to certain clients. The analysis on this dataset can be helpful in stabilizing economic depression[1] that leads to difficulty in retention of customers in banking institutions.
This paper[2] is based on predicting whether the client will subscribe to a long-term deposit by implementing machine learning models such as k-NN, random forest, feedforward neural network and logistic regression on resampled dataset.

3. METHODOLOGY

3.1 Data Description and Source

The dataset is taken from direct marketing campaigns (phone calls) of the Portugese Bank. It consists of 41188 instances and 20 features. The response is whether the client will subscribe to the term deposit.
Dataset is downloaded from *https://archive.ics.uci.edu/ml/datasets/bank+marketing#*
The features in data are of 3 attribute types, i.e., Binary, Discrete and Continuous.
The features in data can be broadly classified into 3 categories:

- Bank client data such as age, job, marital, education, has credit in default, balance, housing, loan.
- Attributes related to last contact of current campaign such as contact type, contact day of the month, month, duration.
- Social and economic context variables such as employment variation rate, consumer price index, consumer confidence index, no. of employees, previous outcome, previous day.

The output, i.e., “does this client subscribe to term deposit” is a two-class binary variable “yes” or “no”.

3.2 Exploratory Data Analysis

From the exploratory data analysis, two significant issues were observed: Unknown values and Imbalanced data.

3.2.1. Handling 'unknown' values

Since the data is collected from phone calls, many clients refused to provide their personal information due to personal issues. The existence of missing data can blur the real data hidden in the pattern thus making it difficult to extract important information. Hence, the unknown values present in the attributes were replaced with the value that appeared more often (mode).

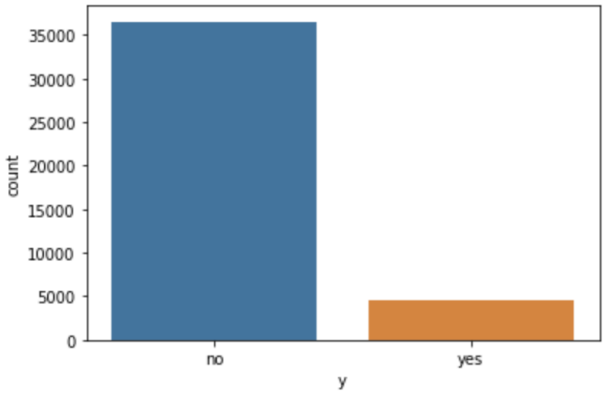
Replacing unknown with most frequent value in column

```
attributesWithUnknown = ['job', 'marital', 'education', 'default', 'housing', 'loan']
```

```
for col in attributesWithUnknown:
    mostFrequent = df[col].mode()[0]
    df[col] = df[col].replace('unknown', mostFrequent)
```

3.2.2. Balancing data

The presence of imbalance data may distort the algorithms and its predicting performance. This problem often exists in real world dataset. The dataset is highly imbalanced with 4640 instances of Class *'yes'* which is approximately 10% of data and 36548 instances of Class *'no'*.



Imbalanced Data

This problem was addressed by oversampling examples in the minority class[4]. The technique used is referred as ***Synthetic Minority Oversampling Technique (SMOTE)*** which selects the minority class instance at random and finds its *k* nearest minority class neighbors and generates new examples that combine features of minority class with its class neighbors.

3.3 Data Transformation

3.3.1. Feature Engineering

There were 10 numerical features and 10 categorical features in the data. The machine algorithms I am going to perform in this project needs numerical data.

```
Numerical Variables:
['cons.conf.idx', 'euribor3m', 'duration', 'pdays', 'emp.var.rate', 'previous', 'campaign', 'cons.price.idx', 'age',
'nr.employed']

Categorical Variables:
['contact', 'default', 'day_of_week', 'job', 'housing', 'education', 'loan', 'poutcome', 'month', 'marital']
```

Numerical Features

These are generally discretized in modeling methods based on frequency tables. Binning the attributes helps in improving the accuracy of predictive models as it can reduce noise and non-linearity. It also helps in identifying outliers, invalid and missing values in the data efficiently. Binning technique is used on attributes such as *'age'* and *'pdays'* to transform them into categorical counterparts.

Categorical Features

One-hot encoding was performed to represent these features as binary vectors. The new column for each unique string value will be generated and mapped to *'1'* if the sample has this value or else *'0'*.

The output class variable is a binary string variable which is transformed into *0s* ('no') and *1s* ('yes').

3.3.2. Principal Component Analysis

Different sets of data are created by performing principal component analysis on original data and SMOTE data. It is used for dimensionality reduction by projecting each data point onto only the first few principal components to obtain lower dimensional data by preserving as much of the data's variation as possible.

3.4 Training Data and Test Data Split

Splitting data into training sample and test sample helps in measuring how well the mode can perform on unseen data. In this project, the data is split into 80% train and 20% test data.

3.5 Models Selection

Since the problem is binary classification, algorithms are carefully chosen with mixed hypothesis functions and infrastructures. Algorithmic experimentation was done in three categories of algorithms viz. Distance Based (KNN and SVM), Ensemble Learning (Random Forest and Gradient Boosting Method), and Logistic Regression.

GridSearchCV is used for hyper-tuning the parameters in the models. For all learning algorithms, training and validation are done on 3 sets of data, i.e., Imbalanced data (original data), SMOTE data, and PCA data (for both Imbalanced and SMOTE data).
For KNN and SVM, training and validation is done with and without Standard Scaling of data.

3.5.1. Distance Based Algorithms

K-Nearest Neighbors

Since the neighbor-based learning is a type of instance-based learning, it does not attempt to build a general internal model, but simply stores instances of training data.
The k-NN method uses the average outcome value of its k nearest neighbors based on Euclidean distance.

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i$$

The optimal choice of k is highly data dependent. The large value of k suppresses the effects of noise but makes classification boundaries less distinct. Hence, the model was tuned for different values for k. Since it is a distance-based algorithm, the data is standardized using Standard Scaler.

Support Vector Machines

This algorithm works best for classifying non-linear data by using different kernel functions. I have performed SVM with Linear and Radial Basis Function (RBF) kernel. The objective function of RBF kernel assumes that all features are centered around 0 and have variance in same order. If a feature has a variance larger than others in order of magnitude, then it might dominate the objective function. Features are standardized by removing means and scaling to unit variance. The standard score of a sample x is calculated as,

$$z = (x - u)/s ,$$

where, u is mean, and s is standard deviation of training samples.
Tuning parameters used,

Hyper-parameter	Values
C (regularization)	0.1, 1, 10
gamma	1, 0.1, 0.01

3.5.2. Ensemble Learning Algorithms

Random Forest

This algorithm is based on principle of bagging. A diverse set of classifiers are created by introducing randomness while construction. Prediction is the average prediction of all classifiers. The base classifier in random forest is decision tree. Each tree is built from sample drawn with replacement from the training set. The purpose of randomness is to reduce the variance of forest estimator. The individual decision tree typically exhibits high variance and tend to overfit the data. Random forest reduces the variance by combining diverse trees sometimes at cost of slight increase in bias.

The no. of trees in ensemble are called as n_estimators. By cross-validating the hyper-parameters, a fair model with low variance can be obtained. However, tuning RF can be computationally expensive.

Hyper-parameter	Values
criterion	gini, entropy
n_estimators	100, 200, 500
max_depth	3, 5, 8
max_features	4, 6, 8
min_samples_leaf	1, 2, 4

Gradient Boosting Method

This algorithm is based on boosting method of training. The main behind this is to make weak learners perform better. A sequence of trees is grown where each subsequent tree learns from errors of previous tree by changing weights. As a result, performance of each tree is boosted. Prediction of the final ensemble model is therefore the weighted sum of the predictions made by the previous tree models. Trees are added one at a time and existing trees in the model are not changed.

Tuning GBM,

Hyper-parameter	Values
n_estimators	100, 200, 500
max_depth	3, 5, 8
max_features	7,9
min_samples_leaf	1, 2, 4

3.5.3. Logistic Regression

It is a model that fits a linear decision boundary between the positive and negative samples. The logistic function called as Sigmoid function is a S shaped function which outputs the probability value of sample belonging to a class between 0 and 1. This is the most convenient approach for binary classification. Major challenge of overfitting is overcome by adding regularization term (C) in cost function. C shrinks the effect of coefficients by penalizing with a parameter lambda. Regularization here used is l2.

Tuning LR,

Hyper-parameter	Values
solver	liblinear, lbfgs, saga
C	0.001, 0.01, 0.1, 1, 10

4. RESULTS AND FINDINGS

4.1. Models Performance Evaluation

Models	Data	Precision	Recall	f1-Score	Balanced Accuracy
k – Nearest Neighbors	*Imbalanced Data*	0.58	0.32	0.41	64.62%
	SMOTE Balanced	0.54	0.35	0.43	65.61%
	PCA	0.56	0.32	0.41	64.36%
	PCA SMOTE	0.50	0.42	0.46	68.51%
Support Vector Machine (Linear)	*Imbalanced Data*	0.11	1.00	0.20	50.00%
	SMOTE Balanced	0.66	0.33	0.44	65.55%
	PCA	0.63	0.27	0.37	62.32%
	PCA SMOTE	0.50	0.59	0.55	75.95%
Support Vector Machine (RBF)	*Imbalanced Data*	0.68	0.40	0.50	68.83%
	SMOTE Balanced	0.00	0.00	0.00	65.55%
	PCA	0.62	0.42	0.50	69.23%
	PCA SMOTE	0.61	0.47	0.53	71.53%
Random Forest	*Imbalanced Data*	0.68	0.46	0.55	71.57%
	SMOTE Balanced	0.50	0.68	0.57	79.57%
	PCA	0.64	0.48	0.55	72.11%
	PCA SMOTE	0.55	0.72	0.62	82.06%
Gradient Boosting Machine	*Imbalanced Data*	0.66	0.54	0.59	75.05 %
	SMOTE Balanced	0.58	0.63	0.60	78.40%
	PCA	0.63	0.50	0.56	73.03%
	PCA SMOTE	0.58	0.58	0.58	76.28%
Logistic Regression	*Imbalanced Data*	0.63	0.41	0.50	69.08%
	SMOTE Balanced	0.55	0.54	0.55	74.39%
	PCA	0.62	0.36	0.46	66.64%
	PCA SMOTE	0.50	0.58	0.54	75.20%

4.2. Findings

4.2.1. Tuned Models

Support Vector Machines

RBF

Hyper-parameters	Values
C (regularization)	1
gamma	0.01

Linear

Hyper-parameters	Values
C (regularization)	0.1
gamma	1

Ensemble Learning: RF and GBM

Random Forest

Hyper-parameters	Values
criterion	entropy
n_estimators	100
max_depth	8
max_features	6
min_samples_leaf	2

Gradient Boosting Machine

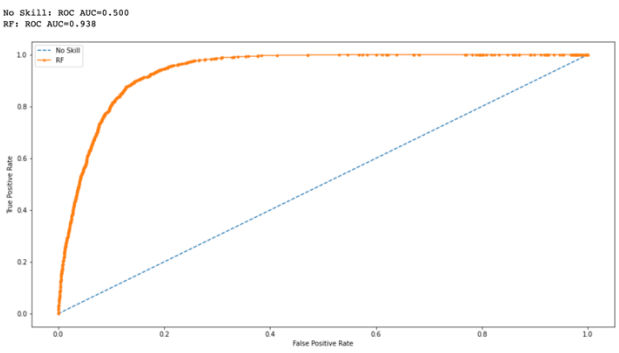
Hyper-parameters	Values
n_estimators	100
max_depth	8
max_features	7
min_samples_leaf	1

Logistic Regression

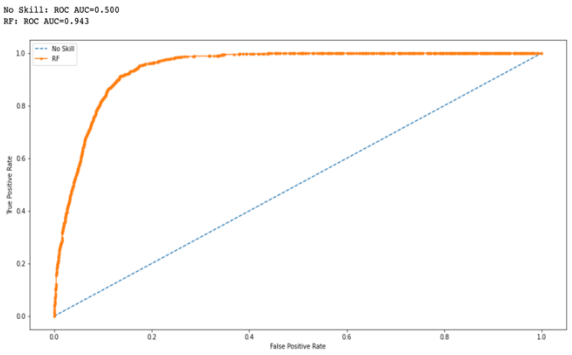
Logistic Regression

Hyper-parameters	Values
Solver	Lifts
C	0.01

4.2.2. ROC Curves

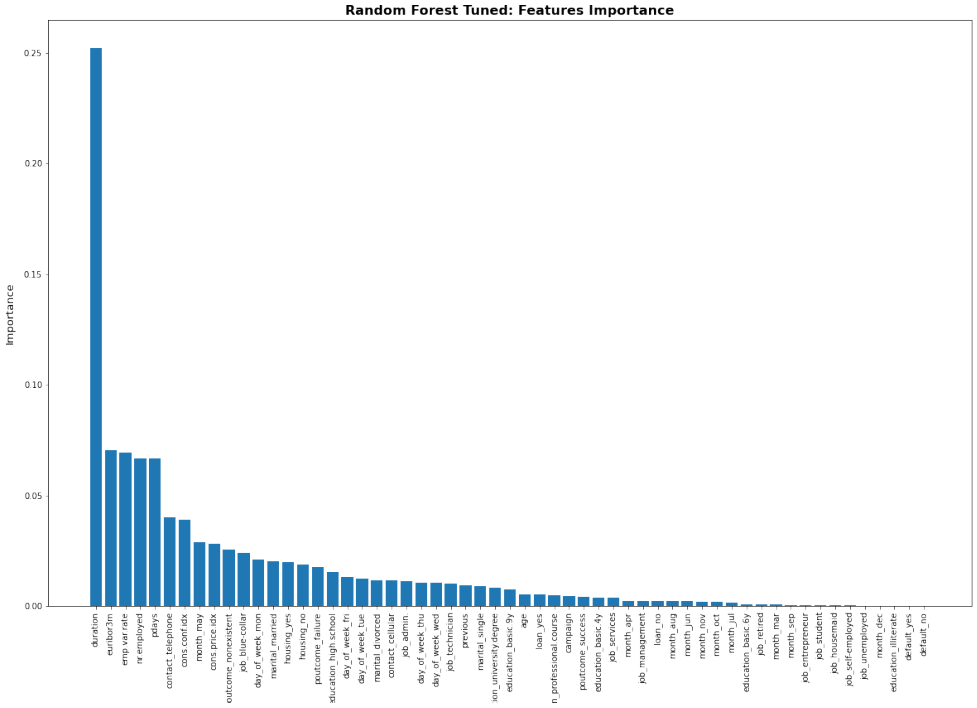


ROC: Random Forest PCA SMOTE



ROC: GBM SMOTE

4.2.3. Features Importance



5. CONCLUSION

According to the experiments, it is concluded that all the models performed better on SMOTE oversampled data rather than imbalanced data. The best model was ***Random Forest*** with balanced accuracy of **82.06%** on **PCA SMOTE** data and **79.57%** on **SMOTE Balanced** data.

The most influential features according to model are ***duration***, ***euribor3m***, ***emp.var.rate*** and ***nr.employed***. Based on above importance, "*duration*" has larger effect on clients saying yes. This is because the larger the conversations on phone, higher interest the customer will show to the term deposit. Secondly, *euribor3m* denotes the euribor 3-month rate. This is based on average interbank interest rates in Eurozone. This positively effects since higher the interest rate, more willingly customer will spend their money on financial tools. Third, *emp.var.rate* denotes employee variation rate which refers to how many people getting hired or fired due to shifts in the conditions of economy. This effects positively because when the economy is at its peak, people are more open to invest on financial tools for higher returns. And lastly, *nr.employed* denotes number of employees working. Greater the number of employees, the greater the customer reach and higher chances of subscriptions.

Therefore, if banks want to improve their lead generation, they should hire more people to work for them, improve the quality of conversation on the phone and run campaigns when interest rates are high and macroeconomic environment is stable.

6. REFERENCES

- Safia Abbas, " *Deposit subscribe Prediction using Data Mining Techniques based Real Marketing Dataset* ", https://arxiv.org/pdf/1503.04344.pdf, Cairo, Egypt, 2015.
- Jiong Chen, Yucen Han, Zhao Hu, Yicheng Lu, Mengni Sun, " *Who Will Subscribe A Term Deposit?*", http://www.columbia.edu/~jc4133/ADA-Project.pdf, Columbia University, US, 2014.
- Rikesh Patel, Urvi Chawada, " *Predicting if a Bank Client will Subscribe to a Term Deposit using Classification Algorithms* ", https://rikeshp.github.io/Projects/BankTelemarketing.pdf.
- Jason Brownlee, " *SMOTE for Imbalanced Classification with Python* ", https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/, 2020.