Predicting outcome of Term Deposit By Customers
Using Machine Learning

Rajiv Puli
Kansas State University
Fall 2021

Abstract

Borrowing or attracting public savings into banks is one way financial institutions generate revenue.

The bank invests the client's long term deposits into other sectors which brings better returns, where some is paid to the customers.

When a customer makes a fixed-term deposit, the company earns more than a savings account because the customer is not allowed to withdraw funds prior to maturity unless the bank is compensated.

The classification goal is to predict if the client will subscribe a term deposit (variable y).

1. Introduction

Recently, as of a few years ago, the data is declared as the new oil in driving the economy of a country. Data has been becoming the key asset of a company and they have been using it to solve many of the problems that have been bugging the profit, performance and long-term evolution of a company, previously.

Data analytics has become the big buzzword over the past decade, with many organizations incorporating some form of data science into their operations. And banks are no exception.

The increasing interest in the use of data analytics in the banking industry is due to the increased changes that have been happening in this sector. Changes in technology, changes in people's expectations, and changes in market structure and behavior.

The advent and application of data analytics have helped the banking industry optimize processes and streamline its operations, thus improving efficiency and competitiveness. Many banks are working on improving their data analytics, mainly to give them an edge against competition or to predict emerging trends that can affect their businesses. This blog post explores why banks need data analytics and how banks are using data analytics for various processes. Along with a case study on how Zuci Systems helped a 100-year-old bank with data engineering and analytics.

2. Related Work

There were several kaggle users who have attempted to do descriptive analysis on the same dataset but no one has gone any further into building a predictive model.

I searched extensively on 'machine learning and banking sector' topics and I found several on www.towardsdatascience.com website. Some of the people tried building models using the same dataset. But the problem is that the person achieved AUC score of 80%. My best model is XG Boost. The AUC of XG Boost Model is 85%.

3. Dataset

The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution.

The marketing campaigns were based on phone calls.

The Dataset is obtained from UCI Machine Learning Repository

There are 17 columns and 45,211 rows in the dataset.

The dataset size is 10 MB.

3.1 Data Cleaning

```
df.isnull().sum()
```

```
age          0
job          0
marital      0
education    0
default      0
balance      0
housing      0
loan         0
contact      0
day          0
month        0
duration     0
campaign     0
pdays        0
previous     0
poutcome     0
y            0
dtype: int64
```

Figure 1: Table Null Values

There are no null values present in the Dataset. No the Data cleaning was not required for this specific dataset.

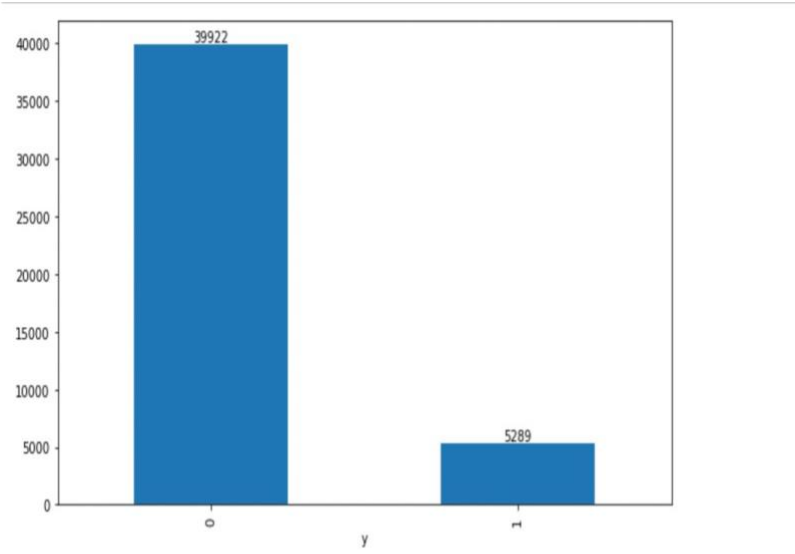
3.2 Data Exploration

These are the numerical and categorical columns present in the dataset.

Numerical columns: Age, Balance, Default, Day, Duration, Campaign, Pdays, Previous,Housing ,Loan, y.

Categorial columns: Job, Marital, Education, Contact, Month, Poutcome.

Percent of customers made Term deposit:



88% of the people did not deposited money and only 12% deposited.

Figure 2: No. Of people who subscribed for term deposit.

Correlation Analysis:

Figure 3: heatmap of feature correlation

There is strong positive correlation between Duration and Y columns. If the Duration increases or decreases then the Y also increases or decreases. So, there is positive correlation between both of the parameters.

4. Methodology

The approach for this project is to work on Anaconda Jupyter notebook with Pandas dataframe. The total project including Model Building is performed in Jupyter Notebook.

4.1 Model Selection

The problem that we are working on is a classification task. Out of all the models in the Sk learn package, I primarily focused on testing the dataset with DecisionTree, Random Forest, NaivesBayes, KNN, SVM, Logistic Regression and XG Boost.

DecisionTreeClassifier: A classifier that builds a single tree with greedy algorithm to predict a class for an observation. *RandomForestClassifier:* This model creates multiple trees with bootstrap and random feature selection algorithm and then aggregate the mode to assign a class to a datapoint. Similar to DecisionTreeClassifier, maxDepth, minInstancePerNode, maxBins, and impurity parameters can be customized to get the best model configuration. *NaiveBayes:* Algorithm based on Bayes' theorem of conditional probability to classify observations.

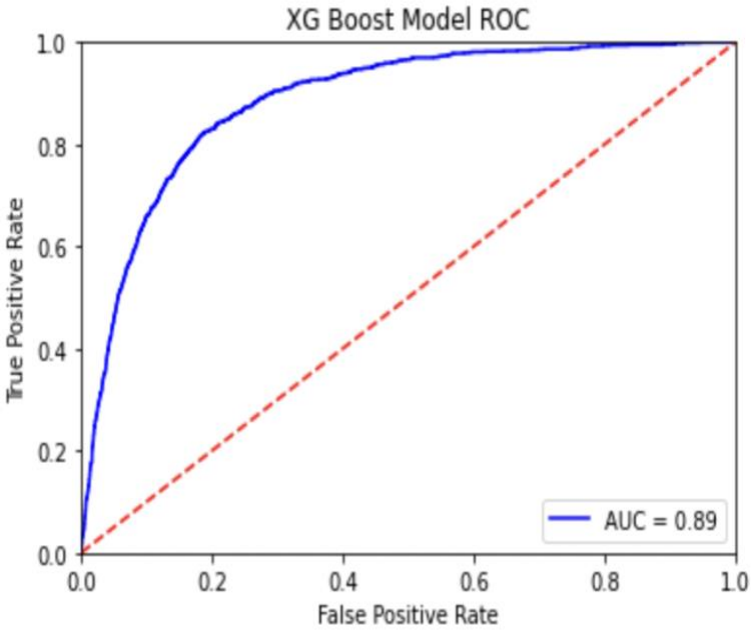
5. Evaluation

The primary metrics for evaluating the models are 'accuracy', 'precision' and 'fscore'. I relied mainly on the accuracy of the model to decide it's performance for the dataset.

Baseline: I took the Decision Tree Classification model without parameter tuning as the baseline to compare the performance of the other potential models. The Accuracy of the decision tree model is 85%.

6. Result

XG Boost model: This model works well with our dataset , the model has accuracy score of 0.89. I performed the K-Fold Cross validation for the XG Boost and it improved the model accuracy by .30% which is very less.



XG Boost Accuracy, Precision, Recall and F1-Score:

```
0.891850049762247
------------------------------------
[[7676  304]
 [ 674  389]]
------------------------------------
              precision    recall  f1

           0       0.92      0.96
           1       0.56      0.37

    accuracy
   macro avg       0.74      0.66
weighted avg       0.88      0.89

------------------------------------
```

```
dt_recall 0.9956544640505662
dt_precision 0.9956701990561545
f1_score 0.9956528488251052
Command took 5.96 seconds -- by ttopjor@ksu.edu at 12/11/2021, 1:14:07 AM on clusterOne
```

Figure 4: XG Boost model metrics scores
Figure 4.1: XG Boost model ROC Curve

Baseline: I took the Decision Tree Classification model without parameter tuning as the baseline to compare the performance of

the other potential models. The Accuracy of the decision tree model is 85%.

Decision Tree Classification model:

Decision Tree Accuracy, Precision, Recall and F1-Score:

```
0.852814331527148
------------------------------------
[[7283  697]
 [ 634  429]]
------------------------------------
              precision    recall  f1

           0       0.92      0.91
           1       0.38      0.40

    accuracy
   macro avg       0.65      0.66
weighted avg       0.86      0.85

------------------------------------
```

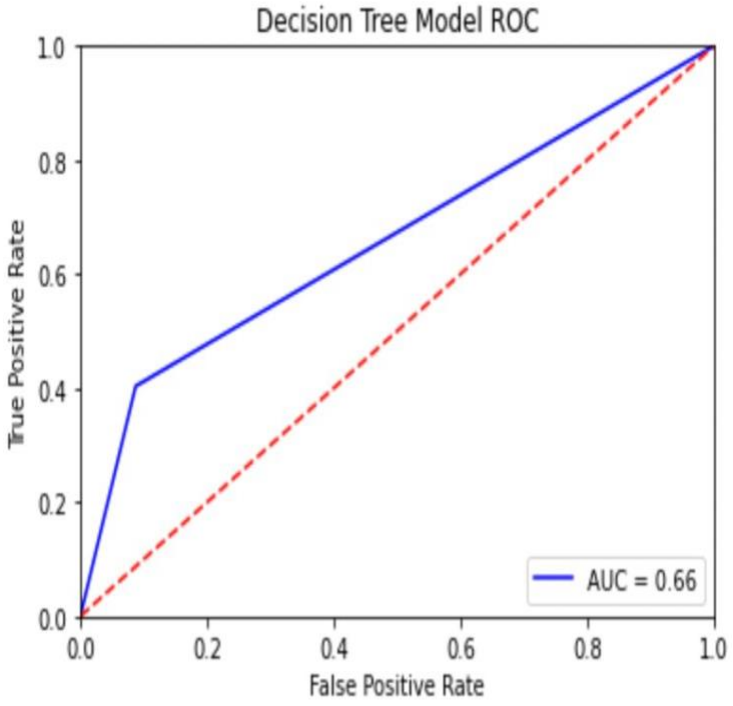


Figure 5: Decision Tree Classification model metrics scores
Figure 5.1: Decision Tree Classification model ROC Curve

7.Cross Validation:

The K-Fold Cross validation is performed for every model. I used the cross validation score of 10 for every model. For the Base Model Decision Tree the cross validation is performed. It increased the model accuracy by .20%.

K- Fold Cross Validation for Decision T

```
from sklearn.model_selection import cross_val_score
accuracies = cross_val_score(estimator = classifier, X =
print("Accuracy: {:.2f} %".format(accuracies.mean()*100))
print("Standard Deviation: {:.2f} %".format(accuracies.st
```

```
Accuracy: 85.40 %
Standard Deviation: 0.39 %
```

Figure 6: Decision Tree Classification model K-Fold Cross Validation accuracy and standard deviation

For the Final Model XGBoost the cross validation is performed

It increased the model accuracy by .30%

K Fold Cross Validation for XGBoost

```
from sklearn.model_selection import cross_val_score
accuracies = cross_val_score(estimator = classifier, X =
print("Accuracy: {:.2f} %".format(accuracies.mean()*100))
print("Standard Deviation: {:.2f} %".format(accuracies.st
```

```
Accuracy: 89.47 %
Standard Deviation: 0.22 %
```

Figure 6.1: XG Boost model K-Fold Cross Validation accuracy and standard deviation

8. Feature Engineering

The Feature Engineering is performed on the dataset. The ExtraTreeClassifier() method is used to identify the important features. I Removed 5 unimportant features manually. This process can called as semi automation. I used 5 important features to build the model. But the model performance has not improved.

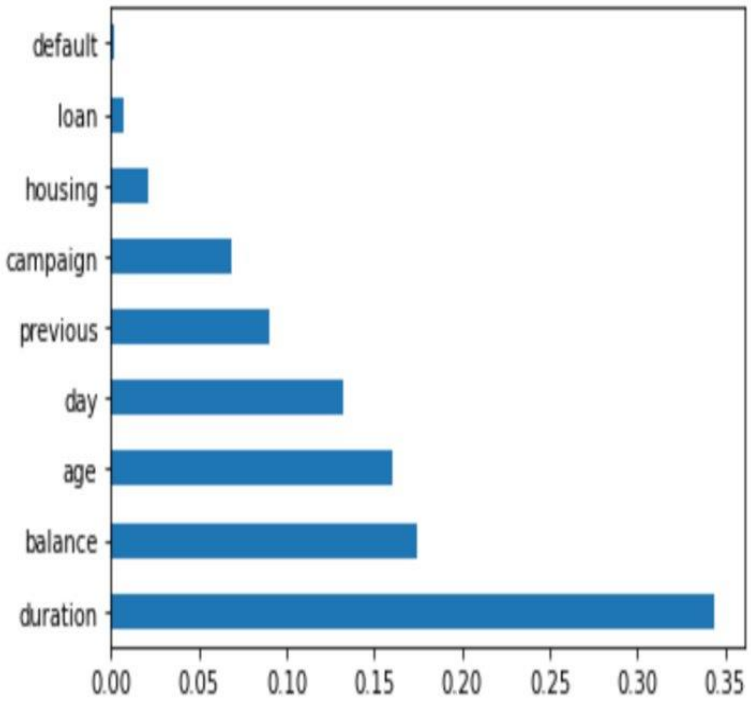


Figure 7.1: Feature Importance plot.

9.Models Comparison:

Model Name	Accuracy	ROC
Decision Tree Classification	85%	66%
Naive Bayes	88.24%	72%
SVM	88.46%	75%
KNN	88.51%	79%
Logistic Regression	88.62%	85%
Random Forest	88.64%	84%
XG Boost	89.19%	89%

Figure 8: Comparison of every model

Every model is compared on the basis of accuracy and ROC.

10. Conclusion

With this project we learnt a complete cycle of building a machine learning model; collecting raw data from various sources, cleaning the data to be fetched into the model, selecting the most appropriate machine learning algorithm, building the actual model with numerous iterations with varying parameters, evaluating the model and finally saving for deployment.

This project has put myself into a test of how much I have learnt and how much of them I was able to put into use for solving a real project. In the initial stages of the project we were doing most of the tasks on a Jupyter Notebook for cleaning and descriptive analysis and few visualizations.
My models have decent accuracy on this data. In future I will work with similar datasets and I will try to increase the model accuracy by feature engineering.

11. Reference

S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing.

Decision Support Systems, Elsevier, 62:22-31, June 2014

S. Moro, R. Laureano and P. Cortez. Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology.

In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimaraes, Portugal, October, 2011. EUROSIS. [bank.zip]

This dataset is public available for research. The details are described in [Moro et al., 2014].

[Moro et al., 2014] S. Moro, P. Cortez and P. Rita.

A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014