INTELLIGENT TARGETING- BANK MARKETING DATA

Suprajah Suresh *Sucharitha Batchu* *Aneesh Shinde*

ABSTRACT:

In this project, our objective is to increase the efficiency of a campaign by identifying the main factors that affect the success of a campaign and predicting whether the campaign will be successful to a certain client, namely, whether the client will subscribe a term deposit. The data is regarding a direct marketing campaign (phone calls) of a Portuguese bank. After cleaning the data, to analyze the data we build six models: k-NN, SVC, random forest, XGBoost, gradient boosting and logistic regression. The optimal model we get is the one using k-NN.

Keywords: k-NN, SVC, random forest, XGBoost, gradient boosting, logistic regression, bank marketing campaign

1.INTRODUCTION

Banks make a profit by providing monetary services to people and one such service is term deposit. Engaging in direct marketing campaigns is one way the banks sell/provide services. Our group found a data set that was the result of a Portuguese bank direct marketing campaign to sell term deposits. The increasing number of marketing campaigns over time has reduced their effects on the general public. We try to improvise such a campaign by suggesting a better strategy based on our analyzation by building a model using the data we collected. We compare 6 different algorithms in this paper to see which model fits the best for our problem.

2.PROBLEM DESCRIPTION

The goal is to build a model that predicts whether a client subscribes to a term deposit or not, improve the strategy for the next marketing campaign and draw business insights from the data.

3.DATASET DESCRIPTION

The dataset for this paper has been downloaded from Kaggle. It is regarding Portuguese marketing campaign with term deposit subscription for 11163 clients and 17 features. There is no imputation necessary as there are no missing values in the data. This data has 6 continuous and 11 categorical variables and the target response is a binary response indicating whether the client subscribed to a term deposit or not. 'Yes' (numeric value 1) indicates the client subscribed to a term deposit and 'No' (numeric value 0) indicates the client hasn't subscribed to a term deposit. Using the DataExplorer package, the dataset is analyzed along with the distributions and the type of each variable. A report is generated for the dataset in which the categories are used as the basis for decision making and estimating the effect on the output variable.

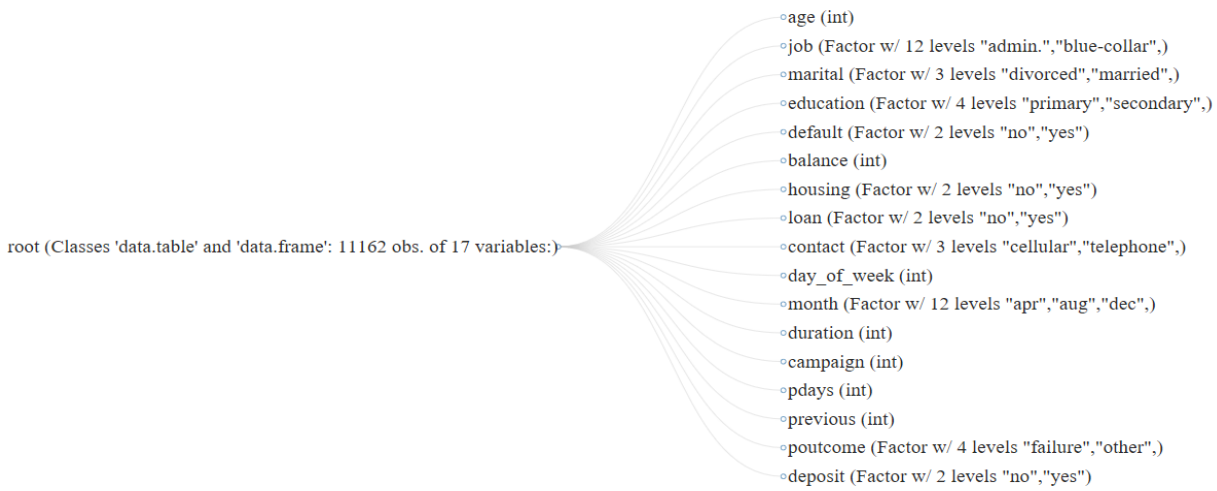


Fig 3.1 Dataset attributes

4. DATA PREPARATION

From the report generated by the Data Explorer package, it is shown t hat there are no rows with any missing values i.e.: all are complete rows. The 11 categorical variables are segregated and factored using the LabelEncoder function from the sklearn package. The LabelEncoder assigns the values to the variables in alphabetical order. Since no form of regression is being performed, LabelEncoder can be used. This facilitates the classification of the data and helps observe the relation between the categorical variables and the outcome of the marketing campaign. The socio-economic attributes of the dataset do not add any significant value to the analysis and hence are not included.

The data when plotted is shown to be left-skewed and this is a classic case where the training data and testing should be split properly. The data remains unbalanced because the model should give realistic accuracy values for the given dataset. In general, for any marketing campaign most of the clients will decline the offer for a term deposit. This shift in the balance exists in all types of marketing. The training and testing data is split in an 80%:20% ratio using the train_test_split function from the sklearn package. The data is split only after the following visualization methods to understand the customer base and the success trends.

5. DATA VISUALIZATION

The data is divided into three categories and are bank data, client data and social & economic attributes.

5.1 Client data Analysis

The client data includes the columns on ages, jobs, marital status, education level, default, if the customer owns a house or not, and loan existence. From the client data, only a few attributes provide some insight. Age is a highly dispersed variable with little to no effect on the outcome of the term deposit. As expected, people who

have not defaulted make up a majority of those who subscribe to the term deposit.

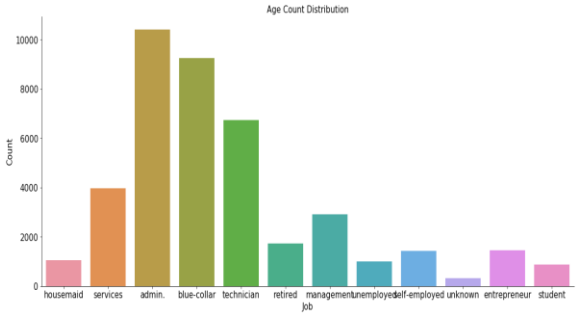


Fig 5.1.1 Occupation Distribution

Whereas, a notable unexpected result is that people subscribe to term deposits irrespective of the existence of a loan. From Fig 5.1.1 we can observe that most of the contacted clients are admins and people with blue-collar jobs. But this only gives the distribution of people and not about the success rates. When the success rates and the occupations are compared, most of the students and retired customers agreed to subscribe to the deposit.

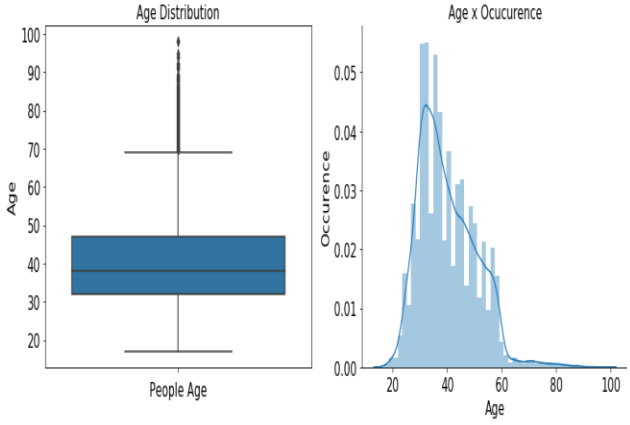


Fig 5.1.2 Age distribution of customers

Many of the potential clients that were students or retired were the most likely to subscribe to a term deposit as students tend to understand the benefits of term deposit easily and retired individuals tend to have more term deposits in order to gain some cash through interest payments. Next, we analyze the loan variable against the outcome of the term deposit. For this, a stacked bar graph is used to represent all four possibilities of the customers with their proportions.

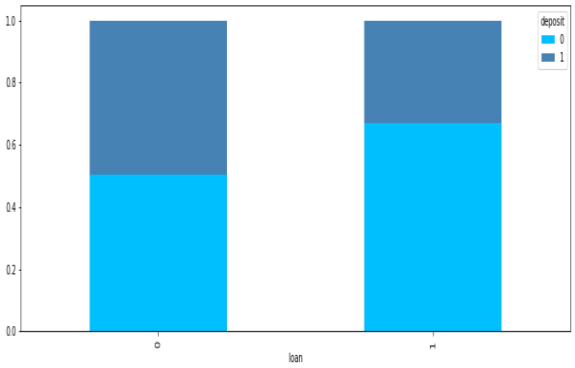


Fig 5.1.3 Loan and term deposit comparison

The above stacked bar graph shows the proportion of the people who have and have not subscribed to a deposit depending on the existence of a loan.

5.2 Bank data Analysis

The data collected by the bank includes number of days since the client has been contacted, duration of the call, marketing strategy and the month during which the call was made. The analysis is focused on the duration of the call which can tell us if the campaign is lucrative to the client. Moreover, analyzing the most successful month can give us an insight as to how the company must allocate their marketing budget. The focus is to maximize the number of people that subscribe to the term deposit.

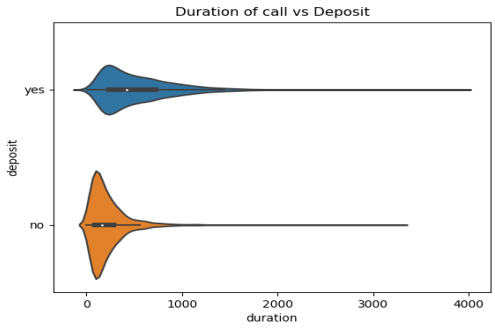


Fig 5.2.1 Duration of call and deposit comparison

The above figure represents a violin plot which is commonly used to compare a continuous variable with a categorical factor. The duration of the call is recorded in seconds and the altitude of the curve determines the extent of success of the marketing campaign. It is to be noted that when the duration of call is very low, the customer does not subscribe. This is as expected because the customer may not have listened to all the features of the campaign. The key to increasing the success rate is to keep the customers interested and make them listen to all the details about the deposit. Another point worth noting is that the altitude of the curve is not large enough when the campaign is successful.

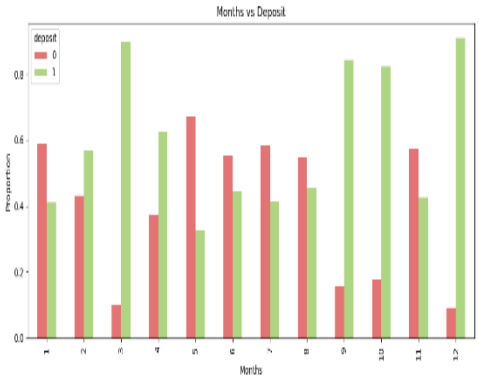


Fig 5.2.2 Months and deposit comparison

This indicates that the campaign too needs some changes like extra benefits to improve the success rates.

It is observed that highest month of marketing activity is may. However, this was the month that many potential clients tended to reject term deposits offers. It is also observed that many clients agreed to subscribe for a term deposit in the months of march and December. So, For the next marketing campaign, it would be best to advertise during the months of march and December**.** (March being financial year closing month is one of the hottest months and December is the hottest month because Portugal people get a bonus which is equal to a month's pay in the name of Restoration of Independence and Christmas).

MODEL SELECTION

Positive rate, the important factor taken into consideration is accuracy of the model. Not compromising on accuracy and False Positive rate KNN fits the best for our design.

Usage of confusion matrices

A confusion matrix is an N X N matrix, where N is the number of classes being

predicted. For the problem in hand, we have N=2, and hence we get a 2 X 2 matrix.

- **true positives (TP):** These are cases in which we predicted Yes and the actual outcome is Yes.
- **true negatives (TN):** These are cases in which we predicted No and the actual outcome is No.
- **false positives (FP):** These are cases in which we predicted Yes and the actual outcome is No.
- **false negatives (FN):** These are cases in which we predicted No and the actual outcome is Yes .

From the above four values obtained from the confusion matrix, the following parameters are evaluated.

- **Accuracy:** the proportion of the total number of predictions that were correct. (TP+TN)/total
- **Sensitivity or Recall:** the proportion of actual positive cases which are correctly identified.

  TP/ (TP+FP)
- **Specificity:** the proportion of actual negative cases which are correctly identified.

  TN/ (TN+FN)

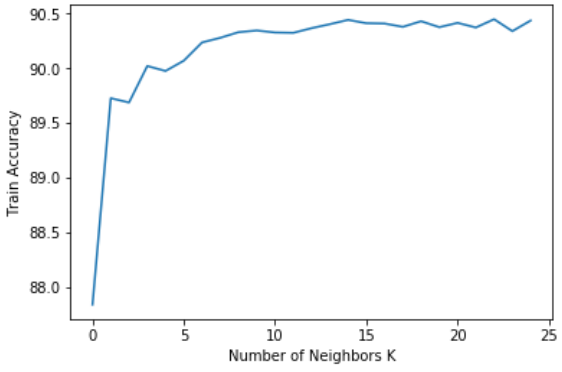
False Positive, means the client has NOT SUBSCRIBED to term deposit, but the model thinks they did. False Negative, means the client SUBSCRIBED to term deposit, but the model said he did not. Having high False Positive is the most harmful, because the company assumes that the client has already subscribed, but they haven't leading to the loss of a client even in future marketing campaigns.

k-NN

In pattern recognition, the k-nearest neighbors algorithm (*k*-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the *k* closest training examples in the feature space. The output depends on whether *k*-NN is used for classification or regression.

In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor and in k-NN regression, the output is the property value for the object. This value is the average of the its k-nearest neighbors.

Firstly, the optimal number of clusters are obtained by running the same algorithm for values of k varying from 1 to 25 clusters of which the optimal value is 22 clusters with an accuracy of 90.4%. K-FOLD cross validation is performed with the value of k as 10 on the training set.



KNN

6962	111
684	243

Random forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees habit of overfitting to their training set. To implement this algorithm, the values in the training and testing set are first scaled using the StandardScalar function. The idea behind StandardScaler is that it will transform your data such that its distribution will have a mean value 0 and standard deviation of 1. Given the distribution of the data, each value in the dataset will have the sample mean value subtracted, and then divided by the standard deviation of the whole dataset. The accuracy of the algorithm is predicted on the scaled testing data. The confusion matrix is obtained from the testing data with the Random Forest Classifier predictor values. The algorithm has an accuracy of 90.0% and the confusion matrix is as follows.

Random Forest

6797	276
491	436

XGBoost

XGBoost is short for eXtreme gradient boosting. It is a library designed and optimized for boosted tree algorithms. Its main goal is to push the extreme of the computation limits of machines to provide a scalable, portable and accurate for large scale tree boosting. This algorithm is run using the xgboost package by first fitting the model using the scaled and unscaled training data following which the confusion matrix is obtained by running the predictors on the testing dataset. This provides an accuracy of 91% but unfortunately has a very high False Positive number.

XGBoost

6858	215
512	415

Gradient Boosting

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. Under the sklearn.emsemble package the gradient boosting classifier function is available. As with the above algorithms, the scaled and unscaled training sets are used to obtain the predictor values and the confusion matrix is from the testing set with the algorithms' predictors. The 10-fold cross validation method is use to validate the results and the mean accuracy of all the 10 runs are calculated. The accuracy of the gradient boosting algorithm is also 91% like xgboost but has a disappointingly high value for the False Positives.

Gradient Boosting

6826	247
460	467

Logistic Regression

Logistic regression is a statistical analysis method used to predict a data value based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables. The logistic regression model comes under the linear model package of sklearn. This algorithm also has a good accuracy percentage of 90% and an acceptable number of False positives. Hence, we will find the ROC metrics for this method.

Logistic Regression

6909	164
598	329

Support Vector Classifier

In machine learning, support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.

SVC

6531	542
584	343

Comparison of accuracies

We are now comparing the accuracies of the all algorithms used

Table 4.1 Accuracy of algorithms

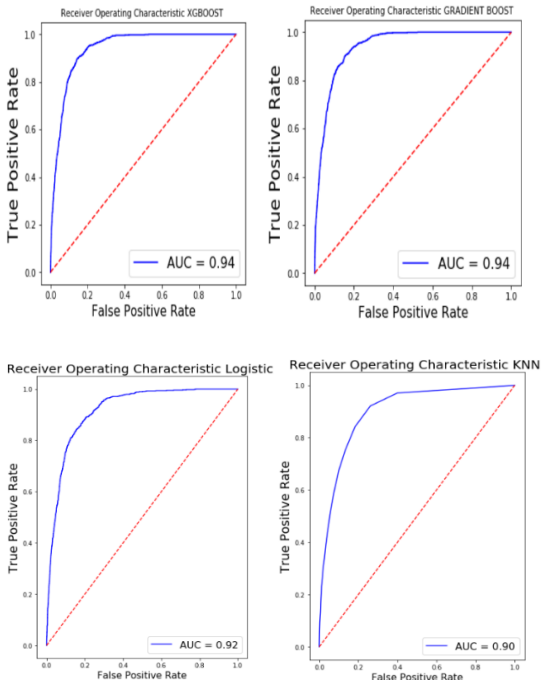
Algorithm	Accuracy
KNN	90.0
SVC	86.0
Random Forest	90.0
XGBoost	91.0
Gradient Boosting	91.0
Logistic Regression	90.0

From the above table, XGBoost and Gradient Boosting algorithms have the best accuracy. A commonly made mistake is that the algorithm with the best accuracy is chosen but does not cater to the actual requirements. This is a typical example of the same. If either the XGBoost or the Gradient Boosting algorithm is used, the False positive number will be 215 and 247 respectively. The goal of the bank should be to reduce the False positive number because the false positive will predict a client to have subscribed to the term deposit but in reality, they have not. This leads to a loss of a potential customer to advertise to and does not improve the rates of success.

7. ROC Curve Analysis

The ROC curve is a fundamental tool for diagnostic test evaluation. In a ROC curve the true positive rate (Sensitivity) is plotted in function of the false positive rate (100-Specificity) for different cut-off points of a parameter. Each point on the ROC curve represents a sensitivity/specificity pair corresponding to a particular decision threshold. The area under the ROC curve (AUC) is a measure of

how well a parameter can distinguish between two diagnostic groups.



CONCLUSION

- Focus should be on reducing the false positive rate rather than accuracy to advertise to all prospective customers.
- K nearest neighbors (KNN) has the least false positive number of just 111 customers.
- The months of March and December have the highest probability of getting customers to enroll.

References:

1. https://support.sas.com/resources/papers/proceedings17/2029-2017.pdf
2. https://arxiv.org/ftp/arxiv/papers/1503/1503.04344.pdf
3. https://mozo.com.au/term-deposits/articles
4. https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
5. https://en.wikipedia.org/wiki/Random_forest
6. https://www.quora.com/What-is-xgboost
7. https://en.wikipedia.org/wiki/Gradient_boosting
8. https://searchbusinessanalytics.techtarget.com/definition/logistic-regression
9. J. Han and M. Kamber. "Data Mining: Concepts and Techniques", Morgan Kaufmann, Text book, 2000.