**UCI Bank Marketing**

**Abstract**

The goal of this project was to use classification models to predict if a client subscribe to the bank term deposit or not. The data is related with direct marketing campaigns of a banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed.

**Design**

In this project we will use the [Bank Marketing Dataset](https://archive.ics.uci.edu/ml/datasets/Bank%2BMarketing) from UC Irvine Machine Learning Repository which is derived from One of the Portuguese banking institution conducted a marketing campaign based on phone calls from 2008 to 2010. The records of their efforts are available in the form of a dataset. The objective here is to build a model to predict whether someone is going to make a deposit or not depending on some attributes. We will try to build 4 models using different algorithm Decision Tree, Random Forest, Naive Bayes, and K-Nearest Neighbours. After building each model we will evaluate them and compare which model is the best for our case. apply machine learning techniques to analyse the dataset and figure out most effective tactics that will help the bank in next campaign to persuade more customers to subscribe to banks term deposit.

**Data**

The dataset contains 45211 examples with 17 features for each, 10 of which are categorical and 7 numeric. A few feature highlights include Duration wich is the duration of call in seconds when the client was contacted last time and poutcome which is The outcome of previous marketing campaign.

**Bank client data**

1. age (numeric)
2. job: type of job (categorical)
3. marital: marital status (categorical)
4. education (categorical)
5. default: has credit in default? (categorical)
6. housing: has housing loan? (categorical)
7. loan: has personal loan? (categorical)

**Related with the last contact of the current campaign**

1. contact: contact communication type (categorical)
2. month: last contact month of year (categorical)
3. day\_of\_week: last contact day of the week (categorical)
4. 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**

1. campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
2. 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)
3. previous: number of contacts performed before this campaign and for this client (numeric)
4. poutcome: outcome of the previous marketing campaign (categorical)
5. Social and economic context attributes

**Output variable (target)**

1. y - has the client subscribed a term deposit? (Binary: 'yes','no')

**Algorithms**

Feature Engineering

1. Converting categorical features to binary variables
2. Numerical transformations (scaling) after splitting
3. Category encoder (one-hot)
4. Apply [SMOTE](https://www.cs.cmu.edu/afs/cs/project/jair/pub/volume16/chawla02a-html/chawla2002.html) (Synthetic Minority Oversampling TEchnique) on data after spliting.

**Models**

The algorithms we used are k-nearest neighbours, Naïve Bayes , Decision Tree and random forest classifiers .

**Model Evaluation and Selection**

The entire training dataset of 45211 records was split into 80/20 train. and the model’s assessment is conducted using essential measurements, such as accuracy and recall, F1 and ROC are also used. the result for each method is:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | Accuracy | Precision | recall | F1 | ROC |
| Decision Tree | 0.86 | 0.43 | 0.53 | 0.48 | 0.72 |
| Random Forest | 0.90 | 0.58 | 0.46 | 0.51 | 0.90 |
| Naive Bayes | 0.85 | 0.39 | 0.55 | 0.46 | 0.80 |
| K-Nearest Neighbours | 0.85 | 0.41 | 0.62 | 0.50 | 0.82 |

We select Random forest because it has the highest ROC, accuracy , Precision and F1.

**Tools**

* Numpy and Pandas for data manipulation
* Scikit-learn for modelling
* Matplotlib and Seaborn for plotting