Learning from Imbalanced Datasets (Supervised and Unsupervised Learning) Project Proposal

Viraj Kumar Dewangan, Registration number 1901181

***Abstract*—Imbalanced dataset is relevant primarily in the context of supervised machine learning involving two or more classes. Imbalance means that the number of data points available for different the classes is different: If there are two classes, then balanced data would mean 50% points for each of the class.Typically, they are composed by two classes: The majority (negative) class and the minority (positive) class.In this project, a new approach will be tested to deal with imbalanced datasets, based on a mixture of supervised and unsupervised learning.**

I. Introduction

DAta are said to suffer the Class Imbalance Problem when the class distributions are highly imbalanced. In this context, many classification learning algorithms have low predictive accuracy for the infrequent class. Cost-sensitive learning is a common approach to solve this problem.Class imbalanced datasets occur in many real-world applications where the class distributions of data are highly imbalanced. For the two-class case, without loss of generality, one assumes that the minority or rare class is the positive class, and the majority class is the negative class. Often the minority class is very infrequent, such as 1% of the dataset. If one applies most traditional (cost-insensitive) classifiers on the dataset, they are likely to predict everything as negative (the majority class). This was often regarded as a problem in learning from highly imbalanced datasets.We will know some techniques to handle highly unbalanced datasets, with a focus on resampling , Decision Trees, Random Forest. In this assignment 1, the analysis is being done till data exploration and visualisation. Imbalance is also being quantified for each of the 3 datasets.

II. Background

In order to solve the imbalanced data set problem, it is necessary to resample data sets. Different resampling techniques are available to achieve this, including undersampling and oversampling.

Undersampling is a technique wherein we reduce the number of patterns within the majority class data set to make it equivalent to other classes.

In oversampling, more data are generated within the minority class. In this study, as a result of a short number of data sets for each class consequently, oversampling is adopted.The literature below describes the fundamentals of different classification methods that will be used in the assignment.

DECISION TREE CLASSIFICATION-

Reasons for choosing Decision Tree Classification for Fake News Detection - • Decision Tree is a white box type of ML algorithm. It shares internal decision-making logic, which is not available in the black box type of algorithms such as Neural Network. • Fast training and forecasting as compared to the neural network. • The decision tree is a distribution-free or non-parametric method, which does not depend upon probability distribution assumptions. • Decision trees can handle high dimensional data with good accuracy. • Supports both numerical and categorical features. • Generation of clear human-understandable classification rules, e.g. ”if age ¡25 and is interested in motorcycles, deny the loan”. This property is called interpretability of the model. • Decision trees can be easily visualized, i.e. both the model itself (the tree) and prediction for a certain test object (a path in the tree) can ”be interpreted”. • Small number of model parameters.

K NEAREST NEIGHBOR(k-NN) CLASSIFICATION-

Reasons for choosing K Nearest Neighbor (k-NN) Classification for Fake News Detection - • Simple implementation. • Well studied. • Typically, the method is a good first solution not only for classification or regression, but also recommendations. • It can be adapted to a certain problem by choosing the right metrics or kernel (in a nutshell, the kernel may set the similarity operation for complex objects such as graphs while keeping the k-NN approach the same. • Good interpretability.

RANDOM FOREST CLASSIFICATION-

A random forest is a meta estimator 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. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if bootstrap=True (default)

III. Methodology

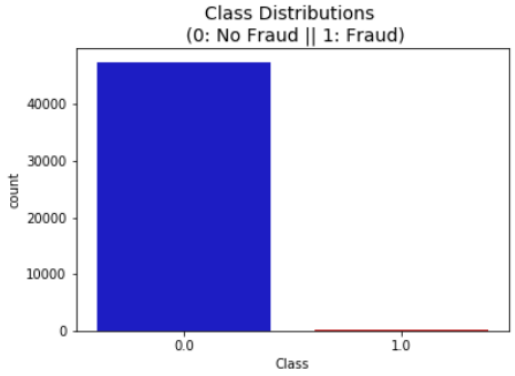
In the assignment 1, Task 1 has been carried out (i.e.)3 Imbalanced Datasets have been chosen for this project.The data sets have been forked from Kaggle and are as under -

1. Credit Card Fraud Detection data.
2. Bank Marketing data for opening term deposits.
3. Porto Seguro's Safe Driver Prediction data for determining insurance claims.

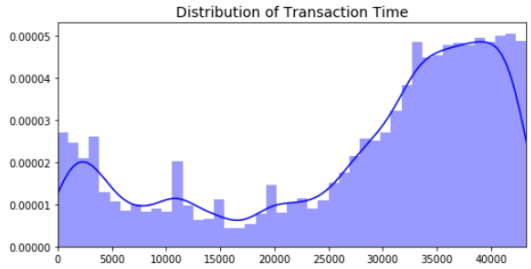
The 3 datasets were loaded and inspected in the python environment to analyse their descriptive statistics. Extensive Exploratory Data Analysis and Visualisation was being carried to properly understand the properties and features of the data set. Percentages of imbalances was also determined. The 3 datasets have been chosen in such a way that the imbalance in them are different.

IV. Results of Exploratory Data Analysis and Visualisation

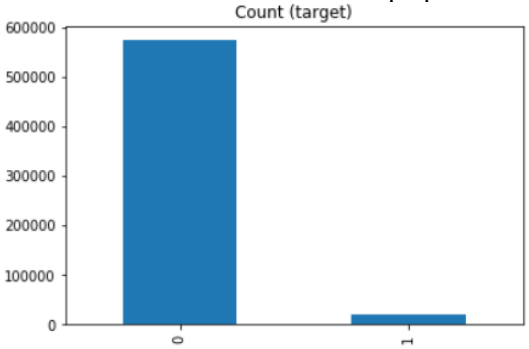
The figure below shows the imbalance of the Credit Card Fraud Detection dataset which is No Frauds 99.69 % and Frauds 0.31 % followed by distribution plots of the dataset



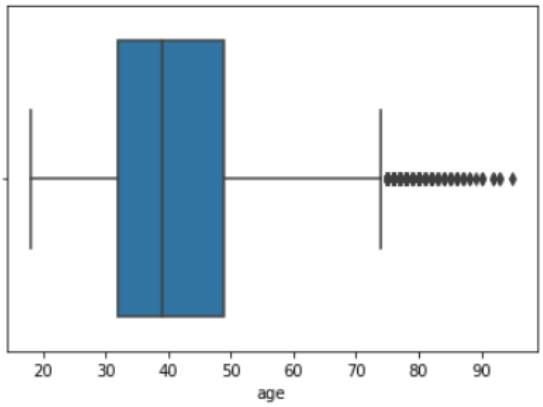
-



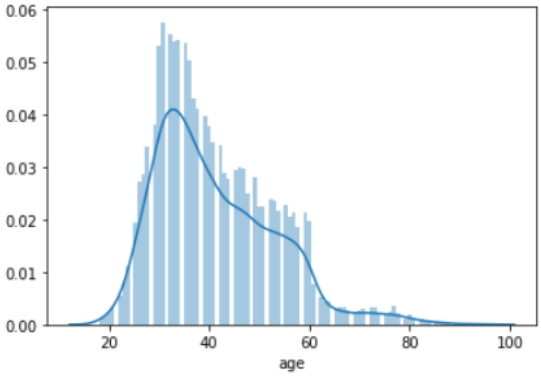
The imbalance in Porto Seguro's Safe Driver Prediction dataset is in 26.33 : 1 proportion as shown-

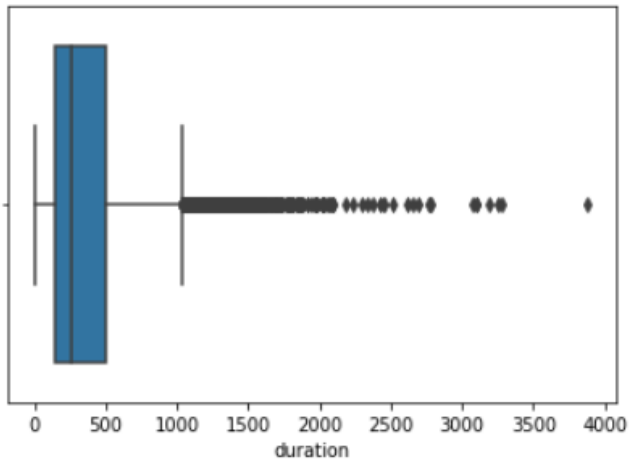


The plots below summarise the imbalance (52:58 ratio) other properties of the third dataset Bank.csv which contains labels whether a customer would take long term deposit or not.

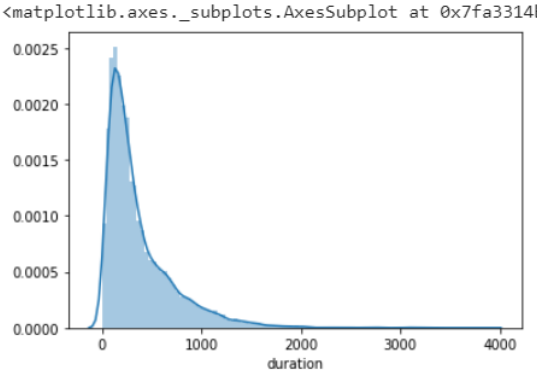


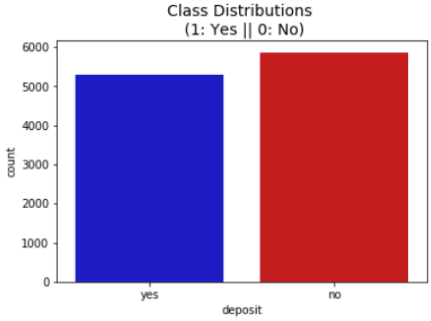

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa331b2ac18>
```




```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa3314b44a8>
```



V. Discussion

The Metric Trap -

One of the major issues that novice users fall into when dealing with unbalanced datasets relates to the metrics used to evaluate their model. Using simpler metrics like accuracy score can be misleading. In a dataset with highly unbalanced classes, if the classifier always ”predicts” the most common class without performing any analysis of the features, it will still have a high accuracy rate, obviously illusory.In this way, the choice of the metric used in unbalanced datasets is extremely important.The Normalized Gini Coefficient is a more robust metric for imbalanced datasets, that ranges from approximately 0 for random guessing, to approximately 0.5 for a perfect score.

The Class Imbalance Problem-

In practice there are cases in which one class is represented by a large number of training points while another by only of few. This is usually referred to as the class imbalance problem. Such situations occur in a number of applications such as text classification, diagnosis of rare medical conditions, and detection of oil spills in satellite imaging. It is by now well established that class imbalances may severely hinder the performance of a number of standard classifiers, for example, decision trees, multilayer neural networks, SVMs, and boosting classifiers. This does not come as a surprise, since our desire for a good generalization performance dictates the design of classifiers that are as “simple” as possible. A simple hypothesis, however, will not pay much attention to the rare cases in imbalanced data sets.
the class imbalance may not necessarily be a hindrance to the classification, and it has to be considered in relation to the number of training points as well as the complexity and the nature of the specific classification task. For example, a large class imbalance may not be a problem in the case of an easy to learn task, for example, well separable classes, or in cases where a large training data set is available. On the other hand, there are cases where a small imbalance may be very harmful in difficult-to-learn tasks with overlapping classes and/or in the absence of a sufficient number of training points. To cope with this problem, a number of approaches have been proposed that evolve along two major directions.

Data-level Approaches-

The aim here is to “rebalance” the classes by either oversampling the small class and/or undersampling the large class. Resampling can be either random or focused. The focus can be on points that lie close to the boundaries of the decision surfaces (oversampling) or far away (undersampling).A major problem with this method is how to decide the class distribution given the data set.

Cost-sensitive Approaches-

According to this line of “thought”, standard classifiers are modified appropriately to account for the unfair data representation in the training set. For example, in SVMs, one way is to use different parameters C in the cost function for the two classes.According to the geometric interpretation, this is equivalent to reducing the convex hulls at a different rate paying more respect to the smaller class, for example,the cost-sensitive modifications of the AdaBoost algorithm are also proposed, where, during the iterations, samples from the small class are more heavily weighted than those coming from the more prevalent class.

Class imbalance is a very important issue in practice. The designer of any classification system must be aware of the problems that may arise and alert of the ways to cope with it.

VI. Plan

The table below shows the timeline of different tasks laid out to be performed in project 2 of CE888 Assignment -

Plan: Breakdown of the work needed to complete this project		
Task	**Dates**	**Time**
Data Exploration & Visualisation	By 20th February 2020	Done
Cross-Validation using Decision Tree	By 29th February 2020	10 days from start date
Cross-Validation using Random	By 29th February 2020	10 days from start date
Clustering using Elbow Method	By 10 March 2020	20 days from start date
Clustering using Silhoutte Method	By 10 March 2020	20 days from start date
Running KNN	By 20 March 2020	1 month from start date
Training Random Forest	By 30 March 2020	1 month and 10 days from start date
Permutation Testing	By 10th April 2020	1 month and 20 days from start date
Reporting Results	By 20th April 2020	2 months from start date

VII. Project links to Github Repository Datasets used

The below repository contains all the python notebooks and datasets that were implemented for the completion of this Assignment 1/Project Proposal.

1. https://github.com/Viraj1901181/CE888Assignment1.git

2. https://www.kaggle.com/rouseguy/bankbalanced

3. https://www.kaggle.com/janiobachmann/credit-fraud-dealing-with-imbalanced-datasets

4. https://www.kaggle.com/c/porto-seguro-safe-driver-prediction

VIII. References

1.https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-$8_1 10$

2.$https://www.sciencedirect.com/topics/computer-science/imbalanced-data$

3.$https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html$