<u>Handling Class Imbalance in Fraud Detection</u>

A project report submitted in partial fulfilment of
the requirements for the degree of
Bachelor of Engineering

By

Clayton Almeida (8587)
Ron George (8605)
Akshay Naphade (8623)

Under the guidance of

Prof. Swati Ringe

DEPARTMENT OF COMPUTER ENGINEERING
FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING
FR. AGNEL ASHRAM, BANDRA (W),
MUMBAI - 400 050.

UNIVERSITY OF MUMBAI
(2021 – 2022)

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING

FR. AGNEL ASHRAM, BANDRA (W),
MUMBAI - 400 050.

CERTIFICATE

This is to certify that the following students working on the project "**Handling Class Imbalance in Fraud Detection**" have satisfactorily completed the requirements of the project in partial fulfillment of the course B.E in Computer Engineering of the University of Mumbai during academic year 2021-2022 under the guidance of "**Prof. Swati Ringe**".

Submitted By: **Clayton Almeida (8587)**
Ron George (8605)
Akshay Naphade (8623)

Prof. Swati Ringe	**Dr. B.S. Daga**
Guide	**Head of the Department**

Principal

CERTIFICATE

This is to certify that the project synopsis entitled **“Handling Class Imbalance in Fraud Detection”** submitted by the following students is found to be satisfactory and the report has been approved as it satisfies the academic requirements in respect of Major Project - I work prescribed for the course.

Clayton Almeida (8587)
Ron George (8605)
Akshay Naphade (8623)

Internal Examiner

(Signature)

Name:

Date:

External Examiner

(Signature)

Name:

Date:

Seal of the Institute

DECLARATION OF THE STUDENT

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources.

We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in my submission.

We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Almeida

Signature of the student **Date: 29/10/2021**

(Clayton Almeida)
(8587)

George

Signature of the student **Date: 29/10/2021**

(Ron George)
(8605)

ANaphade

Signature of the student **Date: 29/10/2021**

(Akshay Naphade)
(8623)

Table of Contents

Chapter 1: Introduction

Class Imbalance problem arises when the ratio of fraudulent activities (minority class) is very less in proportion to nonfraudulent activities (majority class) Imbalanced class distribution in datasets occur when one class, often the one that is of more interest, that is, the positive or minority class, is insufficiently represented.

- In imbalanced classes results are not always accurate which is a very standard classification problem in machine learning.
- There is always a difference in datasets with asymmetric ratio of observations in a class.

Few examples of applications which have imbalanced data sets are: reports of medical diagnosis, finance industry etc.

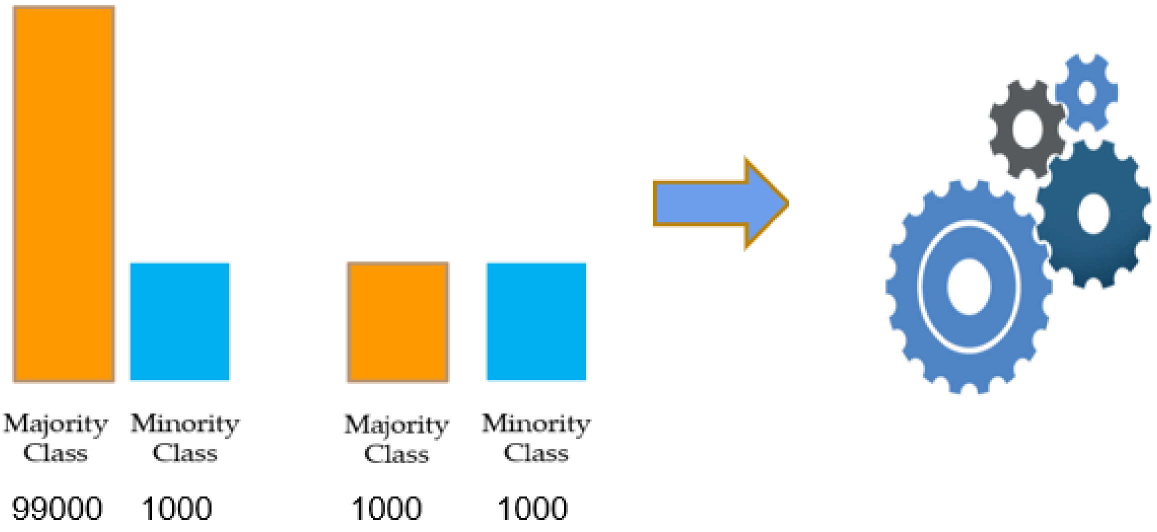
For example, in a medical diagnosis of a rare disease where there is critical need to identify a rare medical condition among the normal population, any errors in diagnostic will bring stress and further complications to the patients.

Imbalanced class distribution arises when there is a significant difference in the frequency of the outcomes when dealing with binary classification. There are different difficulties caused by imbalance classes and they also hinder the performance of machine learning techniques.

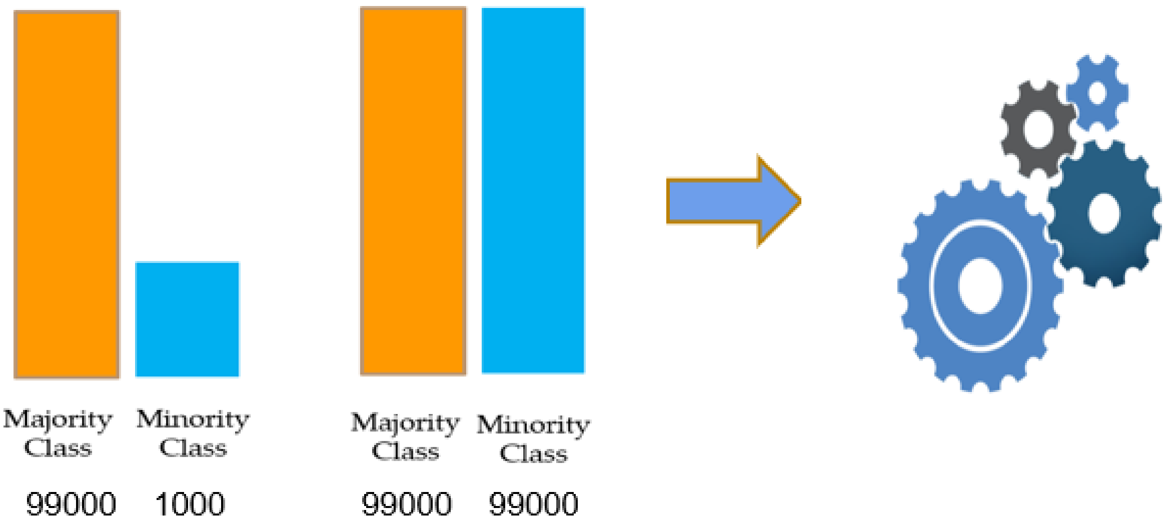
An incorrect classification can therefore lead to a major loss.

Some of the methods to handle fraud detection

1 . Undersampling majority class



2. Oversampling minority class by duplication



Chapter 2: Literature Review

The Authors in paper [1] have explained the issues that come with learning from imbalanced class data sets and various problems in class imbalance classification. A survey on existing approaches for handling classification with imbalanced datasets was also presented. The current trends and advancements which potentially could shape the future direction in class imbalance learning and classification was also discussed. They found out that the advancement of machine learning techniques would mostly benefit the big data computing in addressing the class imbalance problem which is inevitably presented in many real-world applications especially in medicine and social media.

The Authors in paper [2] discovered several shortcomings of existing methods and found that the approaches designed specially to tackle the imbalance problem are not adequately effective. They compared the performance of eight machine learning methods applied to credit card fraud detection and their weaknesses were identified and checked as to how effective they were in the case of extreme imbalance. It was found out that the LR, C5.0 decision tree algorithm, Support Vector Machine (SVM) and Artificial Neural Network (ANN) are the best methods according to the considered performance measures (Accuracy, Sensitivity).

In paper [3] it has been explained that addressing imbalanced datasets in classification tasks is a relevant topic in research studies. The main reason is that for standard classification algorithms, the success rate when identifying minority class instances may be adversely affected. Among different solutions to cope with this problem, data level techniques have shown a robust behavior.

The Authors in paper [4] have explained that the performance of fraud detection in credit card transactions is greatly affected by the sampling approaches on the dataset, selection of variables and detection technique(s) used. In this study a hybrid of under-sampling (the negative cases) and over-sampling (the positive cases) was carried out to achieve two sets of data distributions. The performances of the three classifiers Naïve Bayes, K Nearest Neighbor and Logistic Regression (LR) were examined on the two sets of data distributions

The Authors in paper [5] have explained about the under sampling which has been widely used in the class-imbalance learning area. It was found that the main deficiency of most existing under sampling methods is that their data sampling strategies are heuristic-based and independent of the used classifier and evaluation metric. Thus, the informative instances are discarded for the classifier during the data sampling. The key idea of this method was to parametrize the data sampler and train it to optimize the classification performance over the evaluation metrics.

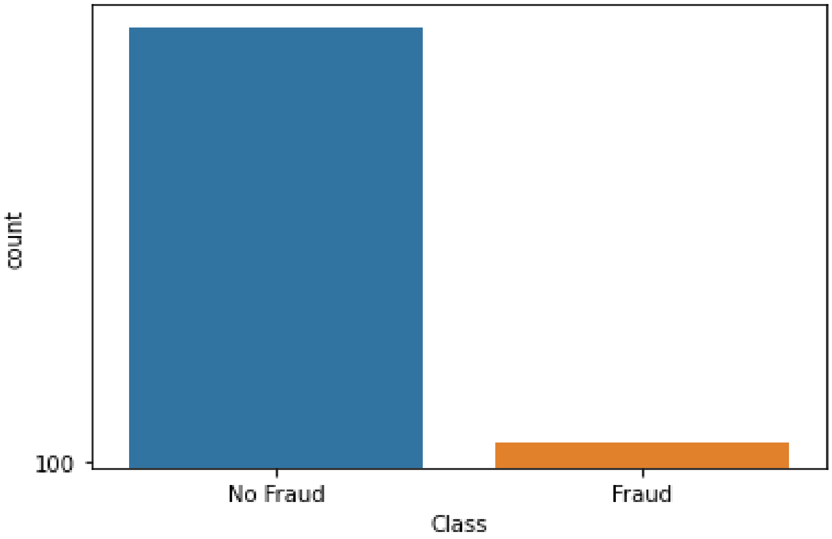
The author in paper [6] provided a large survey of published studies within the last 8 years, focusing on high-class imbalance in big data in order to assess the state-of-the-art in addressing adverse effects due to class imbalance. In this paper, two techniques were covered which include Data-Level (e.g., data sampling) and Algorithm-Level (e.g., cost-sensitive and hybrid/ensemble) Methods. Data sampling methods are popular in addressing class imbalance, with Random Over-Sampling methods generally showing better overall results. It also explains how a majority minority classification problem, class imbalance in the dataset(s) dramatically skew the performance of classifiers, introducing a prediction bias for the majority class.

In paper [7], a novel method that optimally adjusts the SMOTE [3] ratios for rare classes i.e number of cases for the tuple of SMOTE ratios is too large to test all the cases is presented. For that reason, an efficient method is used in which randomly generated some tuples of SMOTE ratios are used to create a model using a support vector regression (SVR). Some number of tuples are given as input for SMOTE ratios to the SVR model, and the best tuple of SMOTE ratios is chosen. Experimental results using the given method were significantly found to be satisfactory. The paper suggests an efficient method on how to find the SMOTE ratios that show good performance with very few tests. Hence, it dramatically reduces the amount of computations required to find the best SMOTE ratios.

Chapter 3: Proposed System

3.1: Problem statement Analysis

To Design a system to handle class imbalance in fraud detection using resampling methods like Random Under Sampling, Random over Sampling and SMOTE (Synthetic Minority oversampling Technique) to overcome the rare events in the dataset and get more accurate results.



Imbalanced Dataset

- Data imbalance usually reflects an unequal distribution of classes within a dataset.
- Imbalance in data is one of the major challenges when dealing with fraud detection model.
- The classifier therefore tends to favor the majority class.
- The credit card fraud detection dataset from Kaggle consists of 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions, most of the credit card transactions are not fraud and a very few classes are fraud transactions.

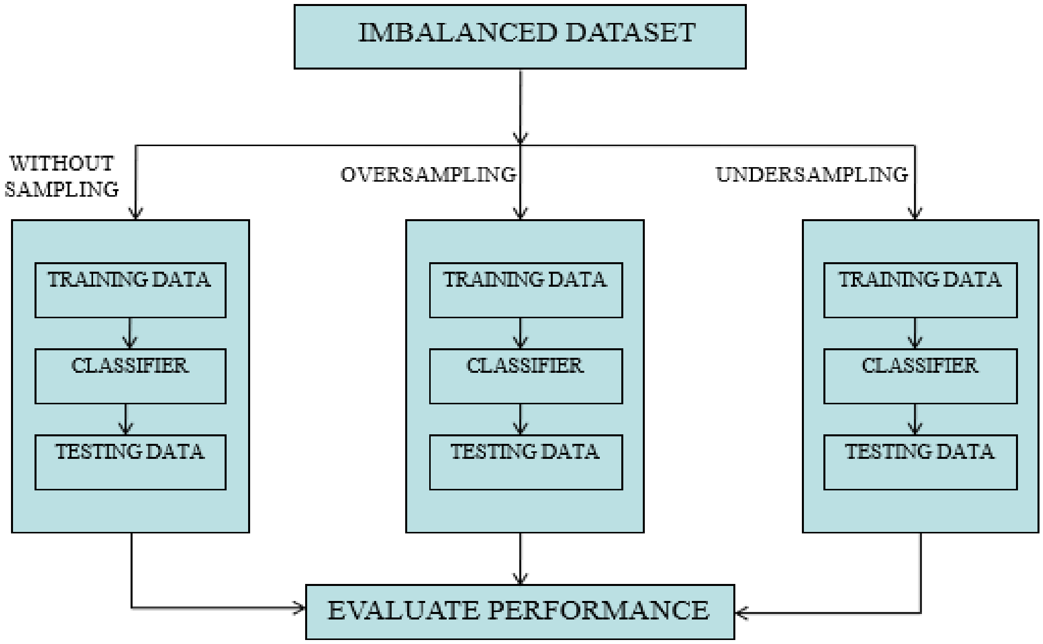
3.2: Design and Methodology of proposed system

The proposed system will help to overcome the existing constraints that lead to inaccurate conclusions and results which might be risky in many cases. We also need to understand all the limitations and the challenges that lead to poor accuracy of the system such as imbalanced data, sampling technique and classification algorithms. These challenges motivate us to provide an accurate solution which will help the user in better decision-making process and also understand the dataset in order to generate an accurate solution.

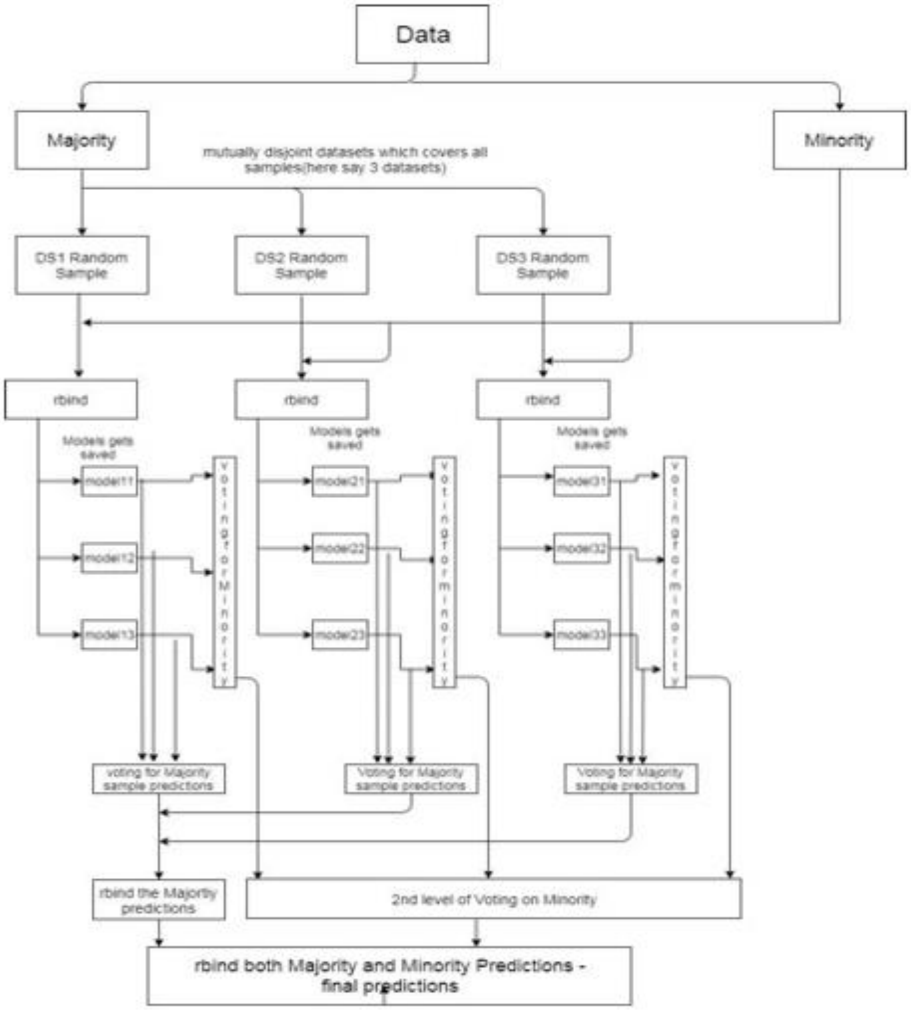
1. **The imbalanced dataset is given to the system which is then pre-processed so that the data is in useable format for analysis.**

2. **Now we shall select any data level approach to balance the data, we will be using a sampling approach.**

3. **After sampling the data it will be given to the classifier and their performance will be evaluated.**



ARCHITECTURE


Data
Majority
Minority
mutually disjoint datasets which covers all samples(here say 3 datasets)
DS1 Random Sample
DS2 Random Sample
DS3 Random Sample
rbind
rbind
rbind
Models gets saved
Models gets saved
Models gets saved
model11
model12
model13
model21
model22
model23
model31
model32
model33
votingforMinority
votingforminority
votingforminority
voting for Majority sample predictions
Voting for Majority sample predictions
Voting for Majority sample predictions
rbind the Majority predictions
2nd level of Voting on Minority
rbind both Majority and Minority Predictions - final predictions

1.Separate the samples based on the class:

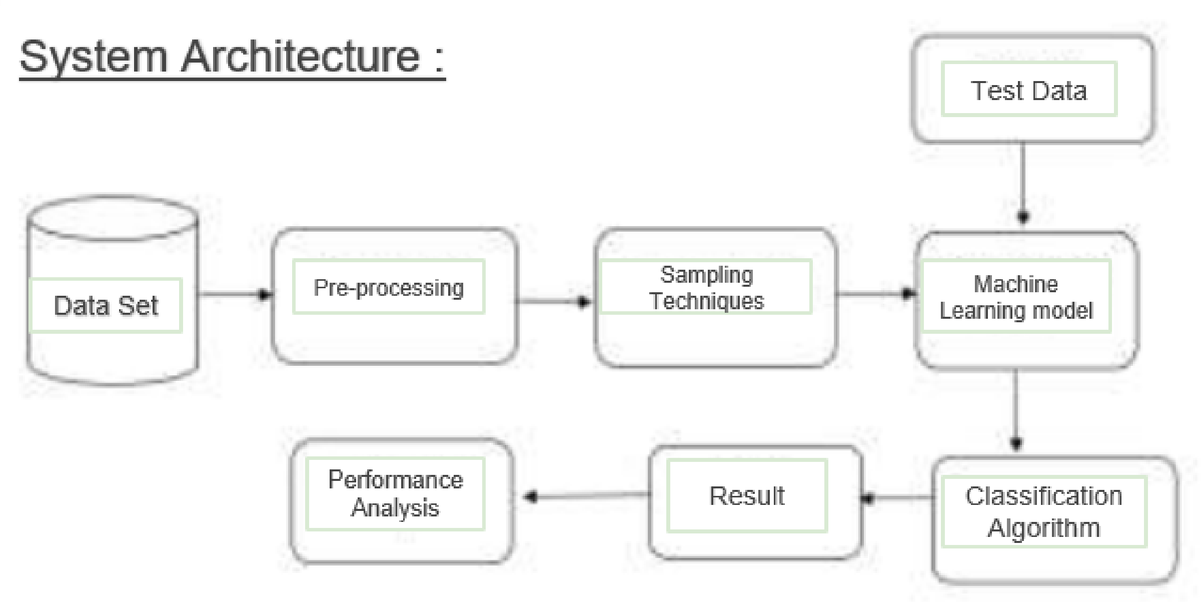
a) Dataset 1- Where all Samples belong to one class Assuming it is Majority Class.

b) Dataset 2- Where all Samples belong to Minority Class.

2.Take subsets from Majority Class without repetition.

3.Merge both the Majority sample dataset and Minority dataset for each subset drawn in step 2.

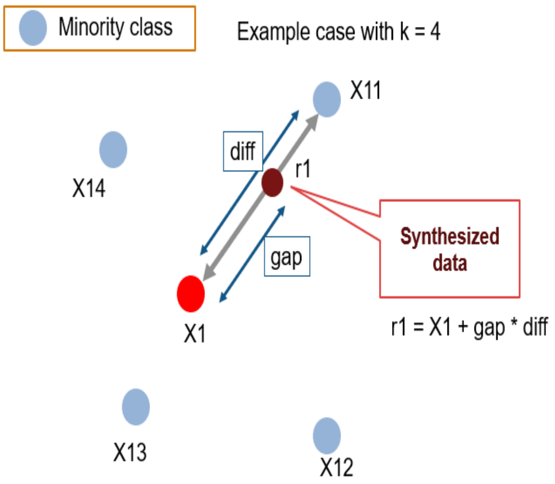
4.Repeat step3 for all Datasets and build multiple models for each.

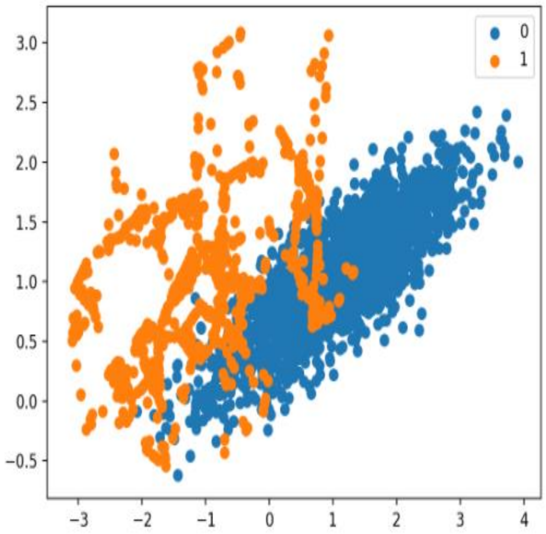


DESIGN:

SMOTE TECHNIQUE:

This technique generates new synthetic minority class data which is used to train the model. It selects minority class data points that are close in the feature space and draws a line between the selected data points. A new sample is generated at a point along this line. This procedure is used to create the required number of synthetic samples of the minority class.





Chapter 4: Hardware software requirements and Implementation

4.1: Hardware Requirements

The hardware requirements for the project are:

- Processor Intel Atom® Processor Z2520 1.2 GHz, or faster processor
- RAM Minimum of 512 MB, 2 GB is recommended
- Storage Between 850 MB and 1.2 GB, depending on the language version

4.2: Software Requirements

Python

The code will be written in Python either using Anaconda IDE or jupyter notebook and the code will be applied to a number of test cases for analysis.

Algorithms

We will be using Machine Learning Algorithms like Logistic Regression, Random Forest, SVM, etc to evaluate the performances.

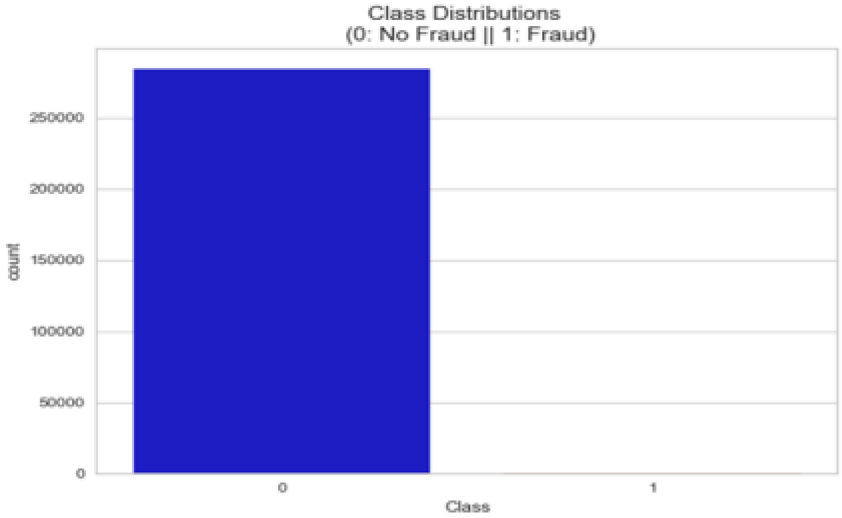
4.3: Implementation

Plotting the data to check Class Imbalance:

1. First the dataset creditcard.csv is loaded

2. The graph is being plotted showing the number of frauds and non-frauds

3. Figure shows the data imbalance where the number of frauds (Class 1) are very less when compared to non-frauds (Class 0)



```
print('No Frauds', round(df['Class'].value_counts()[0]/len(df) * 100,2), '%
of the dataset')
print('Frauds', round(df['Class'].value_counts()[1]/len(df) * 100,2), '% of
the dataset')
```

No Frauds: 99.83 % of the dataset
Frauds: 0.17% of the dataset

Splitting the dataset into training set and testing set

```
for train_index, test_index in sss.split(X, y):
    print("Train:", train_index, "Test:", test_index)
    original_Xtrain, original_Xtest = X.iloc[train_index], X.iloc[test_index]
    original_ytrain, original_ytest = y.iloc[train_index], y.iloc[test_index]
```

Label distribution

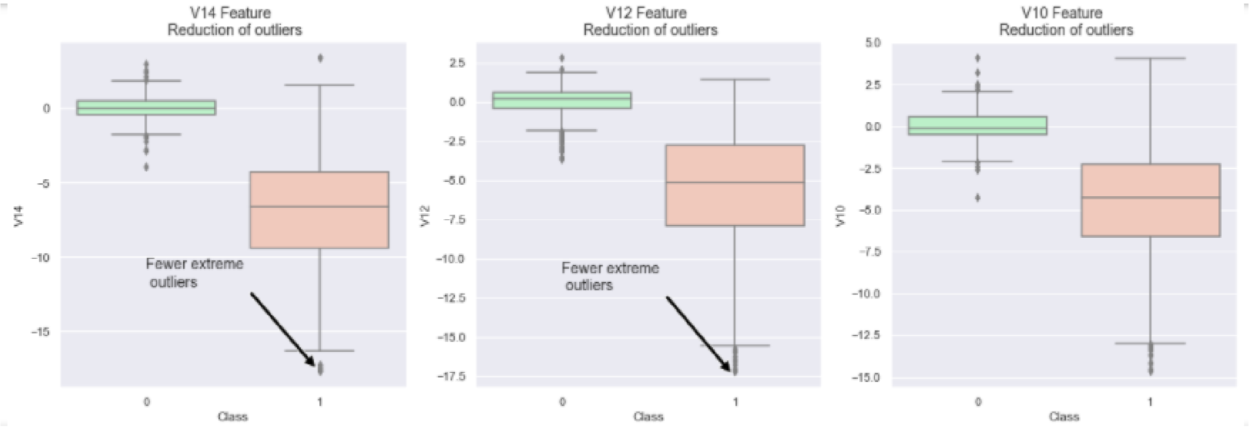
```
df = df.sample(frac=1)

fraud_df = df.loc[df['Class'] == 1]
non_fraud_df = df.loc[df['Class'] == 0][:492]
```

Removing outliers

```
v12_fraud = new_df['V12'].loc[new_df['Class'] == 1].values
```

```
q25, q75 = np.percentile(v12_fraud, 25),
np.percentile(v12_fraud, 75)
v12_iqr = q75 - q25
```



Classification algorithms like SVM, Random Forest, KNN Classification and Logistic Regression were applied on the training dataset before applying the sampling technique and the accuracies were calculated for each one of the classification techniques.

Cross-validation score

```
log_reg_score = cross_val_score(log_reg, X_train, y_train, cv=5)
print('Logistic Regression Cross Validation Score: ',
round(log_reg_score.mean() * 100, 2).astype(str) + '%')

knears_score = cross_val_score(knears_neighbors, X_train,
y_train, cv=5)
print('Knears Neighbors Cross Validation Score',
round(knears_score.mean() * 100, 2).astype(str) + '%')

svc_score = cross_val_score(svc, X_train, y_train, cv=5)
print('Support Vector Classifier Cross Validation Score',
round(svc_score.mean() * 100, 2).astype(str) + '%')

tree_score = cross_val_score(tree_clf, X_train, y_train, cv=5)
print('RandomForest Classifier Cross Validation Score',
round(tree_score.mean() * 100, 2).astype(str) + '%')
```

Chapter 5: Results and Conclusion

5.1 Result

In this dataset we have 492 out of 2,84,807 which are fraud transactions. That’s only 0.173% of all of the transactions in this dataset. Data is not balanced because less amount of fraud cases as compared to huge transaction data. For all datasets, 70% of the data is kept for the training and validation while 30% is used for the testing purpose.

```
Train: [ 30473  30496  31002 ... 284804 284805 284806] Test: [     0      1      2 ... 57017 57018 5
7019]
Train: [     0      1      2 ... 284804 284805 284806] Test: [ 30473  30496  31002 ... 113964 113
965 113966]
Train: [     0      1      2 ... 284804 284805 284806] Test: [ 81609  82400  83053 ... 170946 170
947 170948]
Train: [     0      1      2 ... 284804 284805 284806] Test: [150654 150660 150661 ... 227866 227
867 227868]
Train: [     0      1      2 ... 227866 227867 227868] Test: [212516 212644 213092 ... 284804 284
805 284806]
```

```
Label Distributions:

[0.99827076 0.00172924]
[0.99827952 0.00172048]


Distribution of the Classes in the subsample dataset
1    0.5
0    0.5
```

Reduction of outliers

```
V14 Lower: -17.807576137625002
V14 Upper: 3.8320323233750013
Feature V14 Outliers for Fraud Cases: 4
V10 outliers:[-19.21432549, -18.04999769, -18.49377336, -18.82208674]
---------------------------------------------------------------------------
---------------------------------------------------------------------------
------------------------------------
V12 Lower: -17.343037158875
V12 Upper: 5.776973386124998
V12 outliers: [-18.55369701, -18.68371463, -18.43113103, -18.04759657]
Feature V12 Outliers for Fraud Cases: 4
```

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

Cross Validation score

```
Logistic Regression Cross Validation Score:  94.44%
Knears Neighbors Cross Validation Score 92.86%
Support Vector Classifier Cross Validation Score 93.38%
RandomForest Classifier Cross Validation Score 92.19%
```

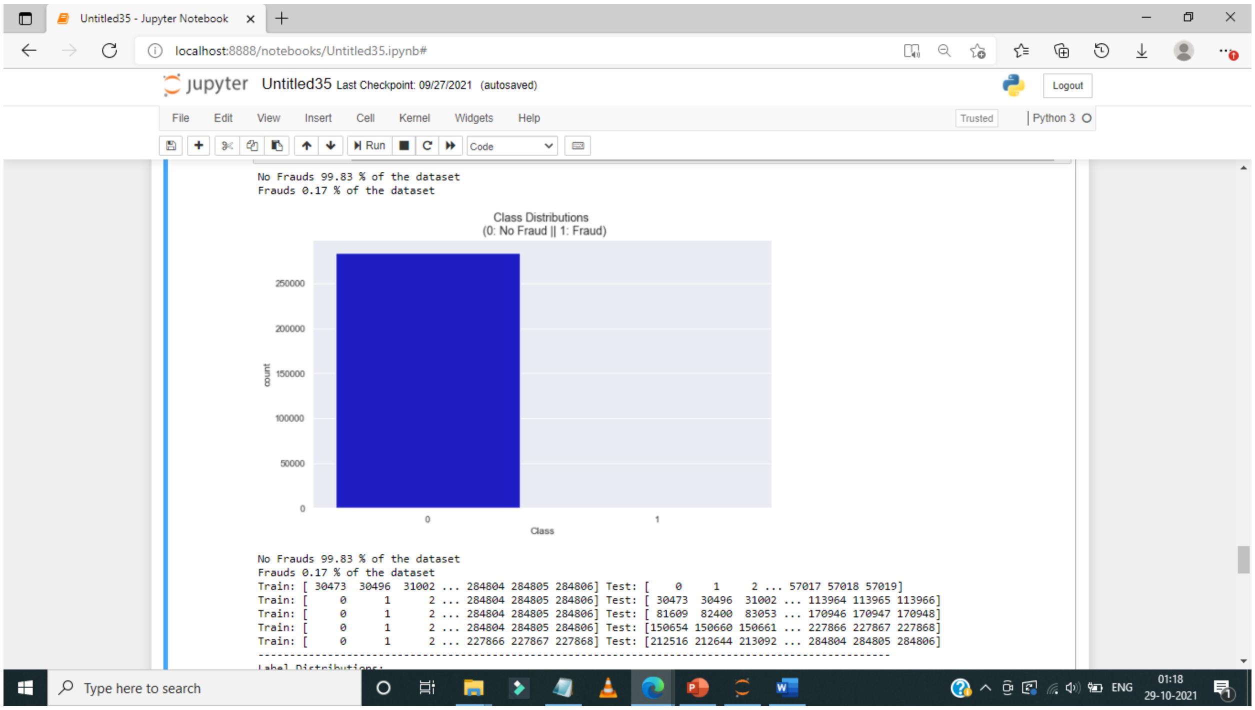
The accuracies for each one of them are :

Logistic Regression has a 94.0 % accuracy score.

KNN Classifier has a 93.0 % accuracy score.

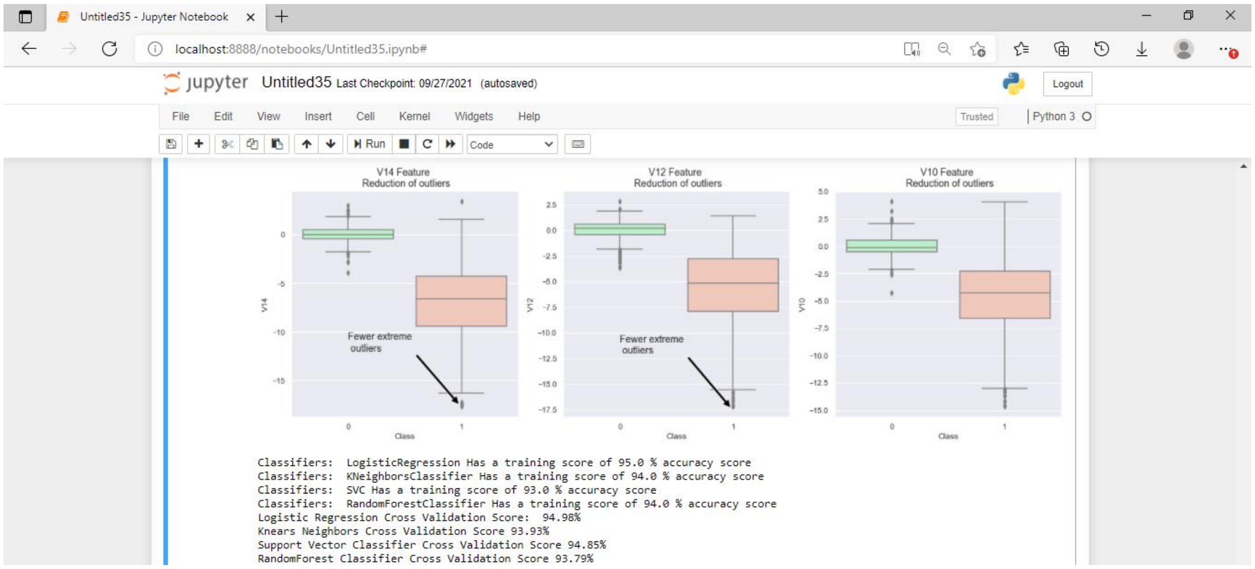
SVM has a 92. 0 % accuracy score.

Random Forest Classifier has a 93.0 % accuracy score.



```
No Frauds 99.83 % of the dataset
Frauds 0.17 % of the dataset
```

```
No Frauds 99.83 % of the dataset
Frauds 0.17 % of the dataset
Train: [ 30473  30496  31002 ... 284804 284805 284806] Test: [     0      1      2 ... 57017 57018 57019]
Train: [     0      1      2 ... 284804 284805 284806] Test: [ 30473  30496  31002 ... 113964 113965 113966]
Train: [     0      1      2 ... 284804 284805 284806] Test: [ 81609  82400  83053 ... 170946 170947 170948]
Train: [     0      1      2 ... 284804 284805 284806] Test: [150654 150660 150661 ... 227866 227867 227868]
Train: [     0      1      2 ... 227866 227867 227868] Test: [212516 212644 213092 ... 284804 284805 284806]
----------------------------------------------------------------------------------------------------
Label Distributions:
```



```
Classifiers:  LogisticRegression Has a training score of 95.0 % accuracy score
Classifiers:  KNeighborsClassifier Has a training score of 94.0 % accuracy score
Classifiers:  SVC Has a training score of 93.0 % accuracy score
Classifiers:  RandomForestClassifier Has a training score of 94.0 % accuracy score
Logistic Regression Cross Validation Score:  94.98%
Knears Neighbors Cross Validation Score 93.93%
Support Vector Classifier Cross Validation Score 94.85%
RandomForest Classifier Cross Validation Score 93.79%
```

5.2 Conclusions

Fraud detection has been one of the major challenges for most organizations particularly those in banking mainly credit card fraud detection, finance, retail, and e-commerce. This goes without saying that any fraud negatively affects an organization’s bottom line, its reputation and deter future prospects and current customers alike to transact with it. It will also be beneficial to the organization so that they can prevent such type of frauds in future and any misleading results. By handling class imbalance problem, we can also overcome various other constraints in medical sectors and also in commercial sectors.

As per our research, during sampling, various techniques such as oversampling, under sampling, Near Miss etc. are used where under sampling technique leads to a lot of data loss and this elimination leads to a loss of meaningful information. So, we will be using Synthetic Minority Oversampling Technique (SMOTE) which is a very popular oversampling technique.

In the next semester we will be implementing sampling techniques focusing more on SMOTE technique to overcome class imbalance problem in the given dataset and also apply classification algorithms like Logistic Regression, Random Forest, etc. to predict and compare the accuracy of the model.

References

[1] Awoyemi J. O, Adetunmbi A. O & Oluwadare S. A, “Credit card fraud detection using machine learning techniques: A comparative analysis,” International Conference on Computing Networking and Informatics (ICCNI), doi:10.1109/iccni.2017.8123782, 2017

[2] Aida Ali1, Siti Mariyam Shamsuddin, and Anca L.Ralescu, “Classification with class imbalance problem: a review,”Int. J. Advance Soft Compu. Appl, Vol. 7, No. 3, November 2015.

[3] Makki S, Assaghir Z, Taher Y, Haque R, Hacid M.S & Zeineddine H, “An Experimental Study with Imbalanced Classification Approaches for Credit Card Fraud Detection,” IEEE Access, 1–1, doi:10.1109/access.2019.2927266, 2019

[4] Hartono, Hartono & Sitompul, Opim & Tulus, Tulus & Nababan, Erna. (2018). “Biased support vector machine and weighted-SMOTE in handling class imbalance problem.” International Journal of Advances in Intelligent Informatics. 4. 21. 10.26555/ijain.v4i1.146. Springer, Singapore,2018

[5] Zhu B, Baesens B, Backiel A & Vanden Broucke S.K, “Benchmarking sampling techniques for imbalance learning in churn prediction”, Journal of the Operational Research Society, 69(1), 49-65, 2018.

[6] Leevy J. L, Khoshgoftaar T. M, Bauder R. A & Seliya N, “A survey on addressing high-class imbalance in big data,” Journal of Big Data, 5(1), 42, 2018.

[7] Machine-Learning Approach to Optimize SMOTE Ratio in Class Imbalance Dataset by Jae-Hyun Seo 1 and Yong-Hyuk Kim 2 1 Department of Computer Science and Engineering, Wonkwang University, 460 Iksandae-ro, Iksan-si, Jeonbuk 54649, Republic of Korea.