Credit Card Fraud Detection

Milestone Report

1. Introduction

This project deals with the problem of building a machine learning model for fraud detection – identifying the fraudulent transactions from a set of credit card transactions, and hence the clients are financial institutions. The dataset used in this project is published by Kaggle[1], which contains credit card transactions from September 2013, made by European cardholders. This dataset presents transactions that occurred in two days, where there are 492 frauds out of 284,807 transactions. Therefore, the dataset is highly imbalanced as the positive class (frauds) account for 0.172% of all transactions.

It contains only numerical input variables, which are the result of a Principal Component Analysis (PCA) transformation. Unfortunately, due to confidentiality issues, the original features and the detailed background information about the data are not provided. Coded features V1 to V28 are the principal components obtained after applying PCA to the raw dataset. See Table 1 for more details.

Total Transactions	Legal	Fraudulent	Fraud Ratio	Number of Features
284807	284315	492	0.17%	28

Table 1: Summary of Dataset

Feature labeled as 'Class' is the response variable and it takes value 1 in case of fraud, and 0 otherwise. There are no missing values in the dataset.

2. Baseline Approach

The baseline approach simply takes the original dataset as it comes and trains the models directly on the highly skewed data. A 75%-25% training-test split of the original dataset with stratification is used. Therefore, training and test datasets have the same proportion of fraudulent transactions as the original dataset and they are summarized in Table 2.

[1] https://www.kaggle.com/dalpozz/creditcardfraud

	Training	Test
Fraud	369 (0.17%)	123 (0.17%)
Non-fraud	213,236 (99.83%)	71,079 (99.83%)

Table 2: Summary of Training and Test Datasets

Logistic regression with L2 norm regularization is trained for the baseline approach. Five-fold cross validation is applied to tune the regularization parameter for logistic regression, where 0.01, 0.1, 1, 10, and 100 are the candidates considered.

3. Performance Metrics

In learning extremely imbalanced data, the overall classification accuracy is often not an appropriate measure of performance. A trivial classifier that predicts every case as the majority class can still achieve a very high accuracy. In this case, it is clear that the performance of the model should be emphasized on its capability of detecting the fraud transactions (labeled as "1" in the dataset). Hence, the following two scores are employed:

- Recall (true positive rate or sensitivity): fraction of fraud transactions that are successfully detected, i.e. number of correctly predicted frauds divided by the total number of actual frauds.

4. Results

The training-test split process discussed in Section 2 is performed 5 times, i.e. the models are trained and tested on 5 different cases, and each time they are tested on the same samples (123 frauds and 71,079 non-frauds).

	Baseline - Logistic Regression	
Case	Training	Test
1	0.645	0.642
2	0.645	0.618
3	0.623	0.650
4	0.604	0.642
5	0.615	0.602
Avg.	0.627	0.627

Table 3: Recall for Baseline Approach

For the baseline approach using logistic regression, the recall scores for both training and test are fairly close to each other. Since only 62.7% of the fraudulent transactions in the test datasets are currently captured, the recall performance remains to be enhanced. Resampling approaches, i.e. undersampling and oversampling, will be considered and their performance will be compared the baseline approach.