A Comparison Between Models and Sampling Methods for Imbalanced Fraudulent Credit Card Data:

Niall Anthony McNulty

School of Computing and Mathematics
Keele University

24/05/2022

CSC-40094 Applications of A.I. and Machine Learning
Domingo Salgado

Table of Contents

1. Abstract

Given that credit card fraud in the UK has risen 55.7% during the years 2016 – 2020, the need for better fraud detection solutions is paramount. The aim of this paper is to find an answer to this predicament. Credit card transactional data by nature is imbalanced, biased and skewed due to the instances of fraud being much lower than that of non-fraud. To counter this bias, this research paper uses resampling methods to redistribute the data, in an attempt to build robust solutions. First, a comparison between Logistic Regression, Random Forest and Sequential DNN models will deduce which model generalizes best on imbalanced data. Synthetic Minority Oversampling Technique (SMOTE), Random Under-Sampling (RUS), class weighting, and a hybrid approach, will then be utilised on the best performing model, in an effort to provide maximum MCC, Recall and F1 performance, with an emphasis on recall.

2. Introduction

Recent technological advances have heralded in a new era of convenience. People can shop and spend from the comfort of their homes, resulting in a massive increase in online spending. In 2006 online spending as a percentage of total retail sales was only 2.8% in the United Kingdom. Fast forward to Jan 2021 during the peak of the Covid-19 crisis, and that number peaks at 37.8% (Lewis, 2022). With this dramatic change in shopper dynamics, criminals saw an opportunity to profiteer by means of identity theft / credit card fraud. Credit card fraud generally falls into one of six categories: Card-not-present (CNP), counterfeit, lost and stolen, card-never-arrived, false application fraud, and what is colloquially known as 'skimming'. Skimming is where business' employees illicitly access customers' credit card information, with the purpose of using it themselves or selling it on to third parties (Credit Card Fraud, n.d; Barkved, 2022). As a result, fraud in the UK has risen from £1,820,726 in 2016 to £2,835,622 in 2020, equating to a 55.7% increase. Remote purchase, or CNP fraud alone, saw a staggering 12% increase over the years 2019 and 2020 (Fraud - The Facts 2020, 2021, p. 21).

To fight this increasing criminal trend; merchants, businesses, third-party payment vendors, and governments have been employing state of the art artificial intelligence solutions. Artificial intelligence leverages computers and machines to mimic the human mind, and it's problem-solving and decision-making abilities (IBM, 2020). Past solutions however, relied entirely upon a rules based system. This approach can be troublesome, as rules tend to result in a high number of false positives, resulting in poor customer satisfaction and the potential loss of customers. In addition to inefficiency, rule based solutions are hard to scale due to the evolving nature of fraud (Machine learning for fraud detection, 2022). Current solutions, however, include not only a rules based approach, but incorporate machine and deep learning (neural networks) systems, aided with graph analysis and human insight. This approach is efficient, more accurate, faster, and scalable (Machine learning for fraud detection, 2022). Given that fraud cases are continuing to rise, current systems could be further researched and improved upon to increase efficacy in the models to catch out fraudsters.

This paper aims to research whether resampling methods on credit card data will increase the efficacy of credit card fraud models. The dataset to be utilised was collected and analysed during a research collaboration of Wordline and Machine Learning Group of Université Libre de Bruxelles on big data mining and fraud detection (MACHINE LEARNING GROUP, 2018). First, a comparison will be conducted between Logistic Regression, Random Forest, and Deep Neural Network (DNN) models to deduce which model generalises best to new unseen data. Resampling methods will then be employed on this specific algorithm to see if there are any noticeable improvements in the models performance. Key to the methodology is that resampling will be conducted purely on the training data, as validation and testing data should replicate real world input.

The purpose for investigating various sampling methods and their effectiveness arises from the natural imbalance in credit card fraud datasets. The number of non-fraud vs fraud instances is extreme, and most machine learning algorithms work best when the number of samples in each class are relatively equal (Vidhya, 2020).

During this research paper, in-depth exploratory data analysis will be conducted to gain deeper insight into the data, and to highlight any bias and class imbalance found in the dataset.

The evaluation metrics that will be used to evaluate the models will be the Matthews Correlation Coefficient (MCC), Recall, and the F1 score, with most emphasis being on the recall score. To conclude, a comparison of results between other notable and relevant research papers will be conducted.

3. Literature Review

There are noticeable contributors to this particular focus of research on credit card fraud. Parekh, Rana, and Nalawade in (Parekh et al., 2021, pp. 1–7) propose the Synthetic Minority Oversampling Technique (SMOTE) and the Random Undersampling (RUS) methods to improve the efficacy of KNN, Random Forest and Artificial Neural Network models. The results showed that Random Forest with SMOTE produced the most promising results despite high training time, with a recall score of 0.83. The authors, however, use a RobustScaler to reduce the influence of outliers, which removes crucial signal from models (Vladimiro, 2016). In addition, the provided dataset has already been standardized during principal component analysis (PCA), with the aim of bringing all features under uniform scaling, and to ensure data privacy (MACHINE LEARNING GROUP, 2018). This will mean there are two different scales in their dataset.

Aktar, Masud and Sakib in (Aktar et al., 2021, pp. 1–4) focus specifically on Random Forest to classify credit card fraud whilst utilising SMOTE and RUS on the training data. The best recall score of 95% occurs when using RUS with a repeated stratified k-fold, although with a significant reduction in precision (55%). Stratified k-fold is useful when working with imbalanced datasets, as validation folds are split according to the class ratio, to ensure representative data (sklearn.model_selection.StratifiedKFold, 2022).

Ileberi, Sun and Wang in (Ileberi, Sun and Wang, 2021, pp. 165286–165294) compare Extra Trees, Decision Trees, Support Vector Machines and Extreme Gradient Boosting models with and without SMOTE resampling. The results are indicative that SMOTE improves the performance of the models. The authors used MCC, Area Under the Curve, and Confusion Matrices' for their evaluation metrics. Random Forest with and without SMOTE returns the best MCC scores, at 0.88 and 0.99 respectively. In contrast to this research papers methodology, Ileberi, Sun and Wang apply SMOTE resampling before their train/test split. This suggests testing on resampled data, which is not representative of real world credit card transactional data.

Deshpande, Kamath, and Joglekar in (Deshpande, Kamath and Joglekar, 2019, pp. 1056–1063) chose to compare resampling techniques with four types of classifiers: K-Nearest Neighbours, Support Vector Machines, Logistic Regression and Decision Trees. The authors also chose to further reduce the dimensionality of the dataset using t-Distributed Stochastic Neighbour Embedding (t-SNE), and to remove outliers after using RUS. The results highlight that the Logistic Regression model with RUS returned recall of 99%. Moving forward Deshpande, Kamath, and Joglekar also want to test how SMOTE in conjunction with outlier removal will impact model performance.

4. Methodology

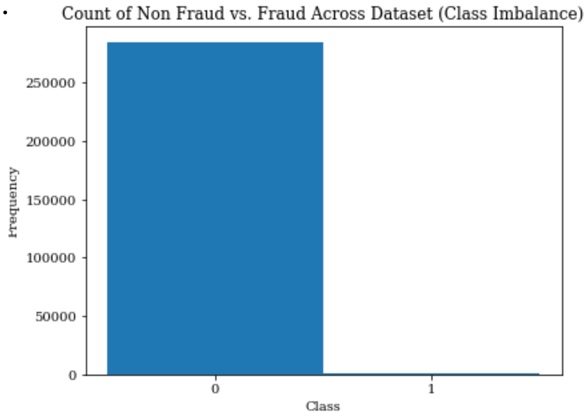
4.1 Dataset Description

The dataset to be utilised was collected and analysed during a research collaboration of Wordline and Machine Learning Group of Université Libre de Bruxelles on big data mining and fraud detection. The dataset contains transactions made by credit cards in September 2013 by European cardholders. The samples in this dataset occurred over a two day timeframe, where a total of 492 frauds were committed out of a total of 284,807 transactions. The dataset is highly imbalanced, with positive classes (frauds) accounting for 0.172% of all transactions. The dataset contains only numerical input variables which are the result of a PCA transformation (including standardization to ensure uniform scaling). Due to confidentiality and data privacy laws, the authors of the dataset could not provide the original feature names. Features V1 to V28 are the principal components obtained via PCA. The only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount. Feature 'Class' is the dependant/output variable and it takes value 1 in case of fraud and 0 otherwise (MACHINE LEARNING GROUP, 2018). There were no null or irregular values in the dataset, as a result data cleaning was unnecessary.

4.2 Exploratory Data Analysis

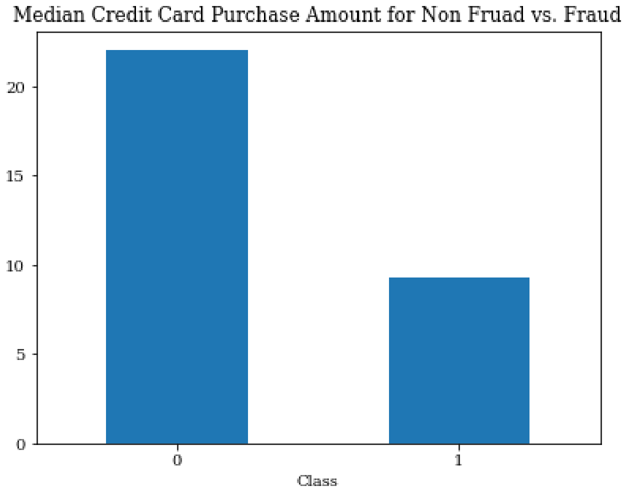
To visualise the extent of the class imbalance in the dataset, a distribution graph was plotted. Fig.1 below emphasises just how significant the bias towards non-fraud instances is in comparison to fraud instances. When dealing with such highly skewed datasets, machine/deep learning models will tend to bias towards the majority class as there is not enough signal for the model to learn any patterns in the minority class. Due in part, because most classification models are designed for problems that assume an equal distribution of classes (Vidhya, 2020). As a result, the model will neglect examples from the minority class, which is in fact, of more interest and whose predictions are more valuable (Brownlee, 2020). This justifies why this research paper aims to utilise sampling methods, by redistributing the classes, it will allow the models to more effectively find patterns between both classes.

Fig 1.



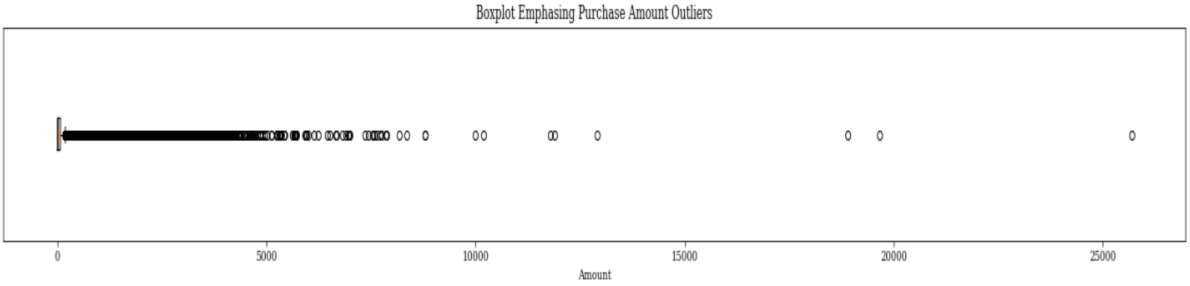
The bar chart below in Fig 2. Highlights the median amount between non fraud and fraud transactions. The median value was chosen over the mean, to minimise outlier influence. This clearly shows that fraudsters are trying to adapt to new artificial intelligence solutions, by attempting to trick models into thinking their fraudulent activities are non-fraudulent, by following similar spending patterns to normal spenders. The subplots in appendix A go further to validate this assumption, by showing the minimum, maximum, mean and median values in the upper fence (outliers). The maximum values are dominated by non-fraudulent transactions.

Fig 2.



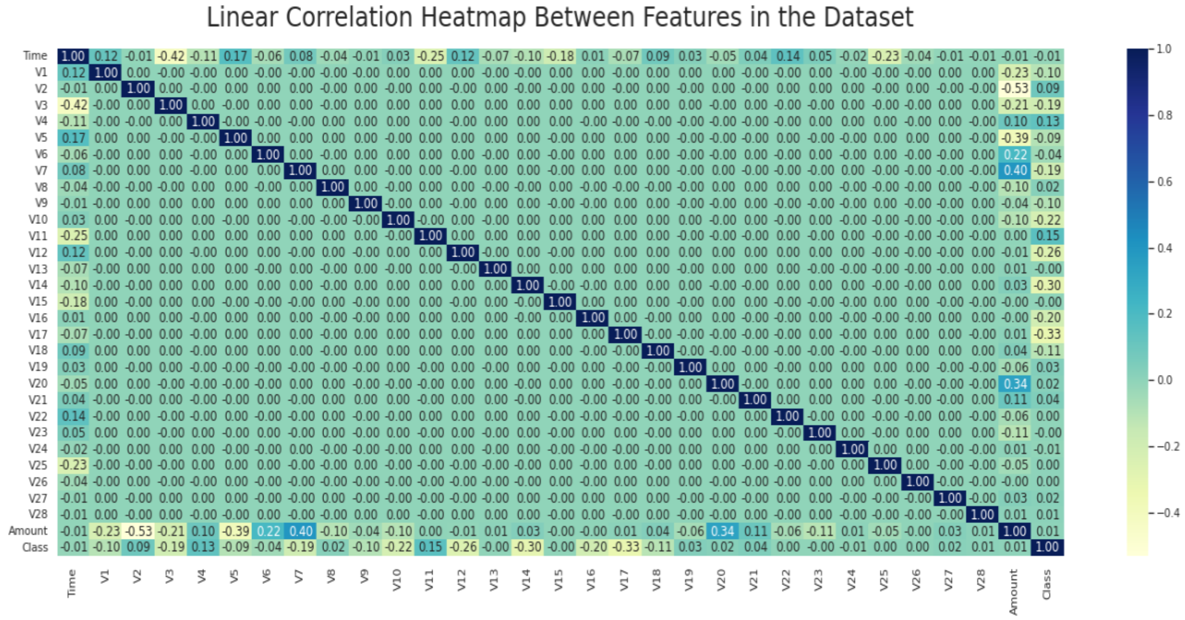
The box plot in Fig 3. shows how highly skewed transactional amounts are towards zero with the occasional extreme outlier. This does not mean outliers can be dropped from the dataset, as machine and deep learning models can still pick up on patterns from outlier values. In the case of fraud detection, outlier values can provide useful information for detecting minority class instances (Vladimiro, 2016).

Fig 3.



The heatmap below in Fig 4. shows correlation between the various features. Feature V20 has positive correlation of 0.34 with the Amount feature. Time and V3 also have a negative

Fig 4.



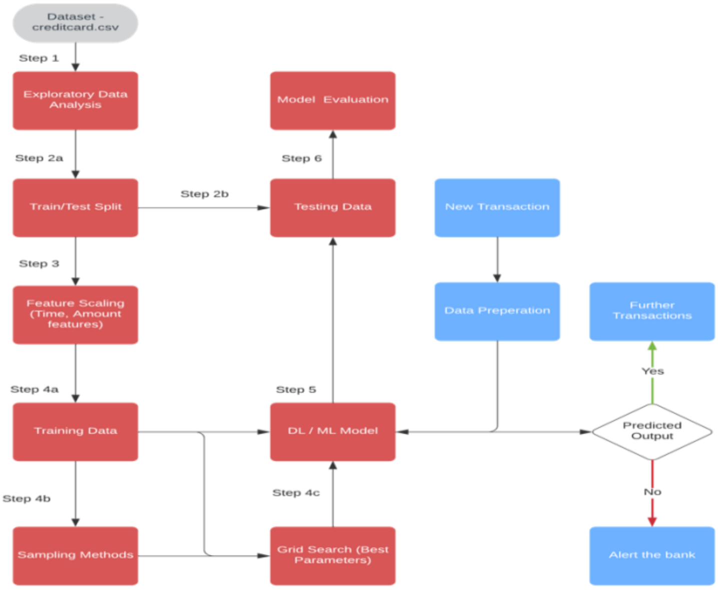
Linear Correlation Heatmap Between Features in the Dataset

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
Time	1.00	0.12	-0.01	-0.42	-0.11	0.17	-0.06	0.08	-0.04	-0.01	0.03	-0.25	0.12	-0.07	-0.10	-0.18	0.01	-0.07	0.09	0.03	-0.05	0.04	0.14	0.05	-0.02	-0.23	-0.04	-0.01	-0.01	-0.01	-0.01
V1	0.12	1.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.23	-0.10
V2	-0.01	0.00	1.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.53	0.09
V3	-0.42	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	-0.21	-0.19
V4	-0.11	-0.00	-0.00	0.00	1.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.10	0.13
V5	0.17	0.00	0.00	-0.00	-0.00	1.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.39	-0.09
V6	-0.06	-0.00	0.00	0.00	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	0.22	-0.04
V7	0.08	-0.00	0.00	0.00	-0.00	0.00	0.00	1.00	0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.40	-0.19
V8	-0.04	-0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.10	0.02
V9	-0.01	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.04	-0.10
V10	0.03	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	1.00	-0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.10	-0.22
V11	-0.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.15
V12	0.12	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	1.00	-0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.01	-0.26
V13	-0.07	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	1.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.01	-0.00
V14	-0.10	-0.00	-0.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	1.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.03	-0.30
V15	-0.18	0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	1.00	0.00	0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00
V16	0.01	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.20
V17	-0.07	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.01	-0.33
V18	0.09	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	1.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.04	-0.11
V19	0.03	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	1.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.06	0.03
V20	-0.05	0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	1.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.34	0.02
V21	0.04	-0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	1.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.11	0.04
V22	0.14	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	1.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.06	0.00
V23	0.05	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	1.00	0.00	-0.00	0.00	0.00	0.00	-0.11	-0.00
V24	-0.02	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	-0.00	-0.00	0.01	-0.01
V25	-0.23	-0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	1.00	0.00	-0.00	-0.00	-0.05	0.00
V26	-0.04	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	1.00	-0.00	-0.00	-0.00	0.00
V27	-0.01	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	1.00	-0.00	0.03	0.02
V28	-0.01	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	1.00	0.01	0.01
Amount	-0.01	-0.23	-0.53	-0.21	0.10	-0.39	0.22	0.40	-0.10	-0.04	-0.10	0.00	-0.01	0.01	0.03	-0.00	-0.00	0.01	0.04	-0.06	0.34	0.11	-0.06	-0.11	0.01	-0.05	-0.00	0.03	0.01	1.00	0.01
Class	-0.01	-0.10	0.09	-0.19	0.13	-0.09	-0.04	-0.19	0.02	-0.10	-0.22	0.15	-0.26	-0.00	-0.30	-0.00	-0.20	-0.33	-0.11	0.03	0.02	0.04	0.00	-0.00	-0.01	0.00	0.00	0.02	0.01	0.01	1.00

correlation of -0.42. A lot of the other features have minimal or no correlation. They are independent from each other. This is a result of the principal component analysis performed by the authors on the dataset. To counter multicollinearity, and to reduce dimensionality, PCA reduces the features down to the same eigenvector space, whilst also maintaining the variance of the data. This allows for faster computational times, as the dimensions of our dataset have been reduced, whilst also removing noise in the data (Kumar, 2020).

4.3 System Flow Diagram

Fig 5.



4.4 Train Test Split

The dataset was split first into a train and test set by an 80/20 ratio respectively. The training set was then further split in validation sets, along the same ratio. To ensure that each set had a proportional ratio of classes, the stratify parameter was set to stratify by the dependant variable (y). Random state was also set to 42, ensuring that each instantiation of the training set returned the same data.

4.5 Feature Engineering

To improve the performance of the models in this research paper and to keep uniformity with the scale of the principle components performed by the dataset authors, StandardScalar was used on the Amount and Time feature variables. Having uniform scales helps the models generalize better to the data, as larger scaled features would dominate the rest, especially in distance based algorithms (e.g. KNN).

4.6 Evaluation Metrics

The metrics used in this research paper are the recall score, the F1 score and the Matthews Correlation Coefficient (MCC).

$$Matthews\ Coefficient = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{TP + FP \cdot TP + FN \cdot TN + FP \cdot TN + FN}}$$

MCC was chosen, as it is a reliable statistical rate which produces a high score only if the prediction obtains good results in all four confusion matrix categories proportionally. Overall it’s indicative of a high performing model in all regards. It is one of very few metrics which includes true negatives in its calculations (Chicco and Jurman, 2020, p. 6).

$$F1\ Score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

F1 score was also utilised as it is a measure that combines both precision and recall. If precision is low, it suggests that the model is poor at measuring false positives. This is not the worst-case scenario for merchants, businesses or credit card companies. Although, it could lead to poor customer satisfaction. The trade-off ultimately comes down to the user. Respectively, if recall is low, this means that the model is poor at predicting false negatives. If either recall or precision is low, the F1 score will be low, as the F1 score is the harmonic mean between the two measures (Brownlee, 2020; Kampakis, 2021).

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)}$$

The evaluation metric that has the most weight in this research paper, is the recall score. Essentially to reduce fraud, fraudulent activity needs to be identified. If a model has poor predictive power and keeps flagging fraudulent activity as non-fraudulent transactions (false negatives), then business', merchants, credit-card companies, individuals and third party payment systems will continue to lose significant amounts of money (Brownlee, 2020; Kampakis, 2021).

4.7 Resampling Methods

To combat class imbalance, SMOTE, RUS and the class weight parameter were applied to the training set in a variety of combinations to deduce which aided the models in generalising best to new unseen data.

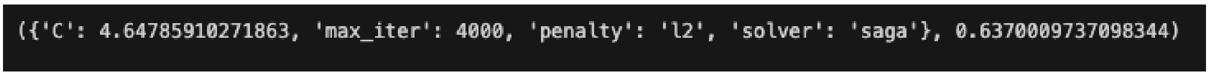
4.8 Machine/Deep Learning Models

In this paper, three models were chosen to compare their predictive power on the imbalanced dataset. To have a baseline model to compare against, Logistic Regression was chosen. Logistic Regression is a basic yet efficient model (computationally). Random Forest and a sequential DNN model were the other two respective models. Random Forest models and DNN's are effective on a wide range of problems, however they perform poorly on imbalanced classifications. By using resampling methods, the performance and metrics should improve (Brownlee, 2020).

4.9 Hyperparameter Tuning

To find the best parameters for each of the models, a Randomised Grid Search was performed. For the Logistic Regression model, cross-validation was performed on 5 splits.

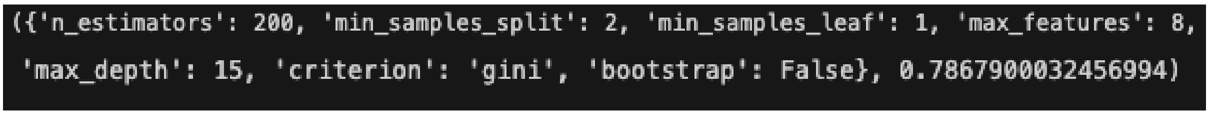
The resulting best parameters were:



```
({'C': 4.64785910271863, 'max_iter': 4000, 'penalty': 'l2', 'solver': 'saga'}, 0.6370009737098344)
```

The ‘penalty’ parameter for the best parameters came out to the l2 regularization. This means the penalty is equal to the square of the magnitude of the coefficients. Regularization is used to combat overfitting on models.

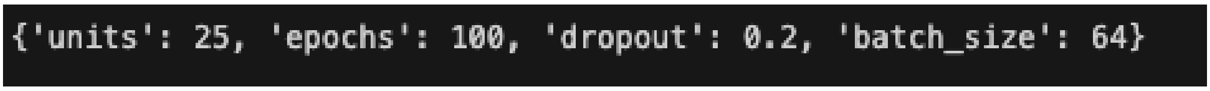
For the Random Forest model, the best parameters were:



```
({'n_estimators': 200, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 8,
 'max_depth': 15, 'criterion': 'gini', 'bootstrap': False}, 0.7867900032456994)
```

The max depth parameter for Random Forest models is defined as the longest path between a root node and the leaf node. Essentially, how many branches the tree can grow too. The larger the depth, the higher the computational power necessary.

For the DNN model, the best parameters were:



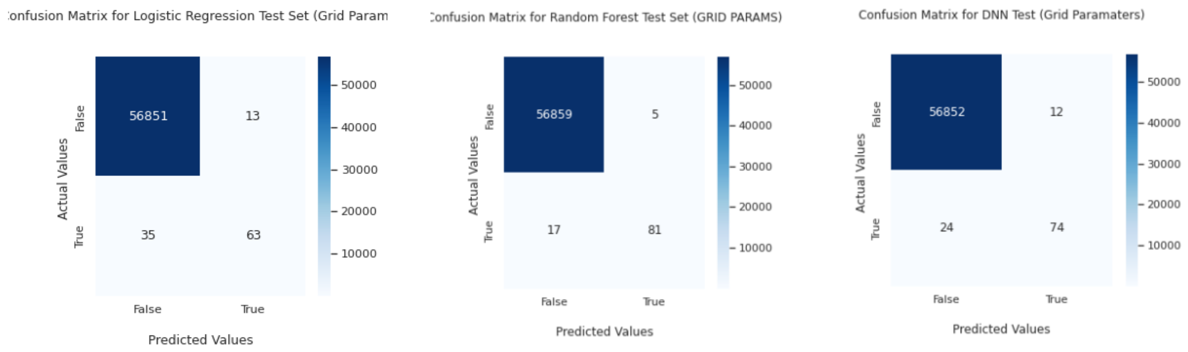
```
{'units': 25, 'epochs': 100, 'dropout': 0.2, 'batch_size': 64}
```

The epochs parameter for neural networks means training the model with all the training data for one cycle. In an epoch, all the data is used exactly once.

5. Results

Using RandomisedGridSearch, the Random Forest model performed best across the three models. As seen in Fig 6. below, the Random Forest had the least false negatives and the least false positives. It generalised best to the unseen test data.

Fig 6.



:onfusion Matrix for Logistic Regression Test Set (Grid Param

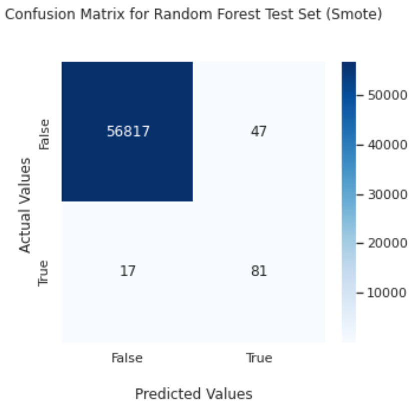
:onfusion Matrix for Random Forest Test Set (GRID PARAMS)

Confusion Matrix for DNN Test (Grid Paramaters)

Having deduced which model performed best on the imbalanced dataset, moving forward the resampling methods were only utilized on the Random Forest model using the best parameters from the grid search.

First, the SMOTE sampling was used to balance the classes. This resulted in a 50/50 split between class 1 and class 0, using synthesized data samples, based off a KNN algorithm. This resulted in an improved recall score by 0.01, although the MCC and F1 scores were reduced by almost 0.16 points for both metrics. So the model became better at identifying false negatives, but lost its predictive power for false positives. Depending on the user, this could be a necessary sacrifice.

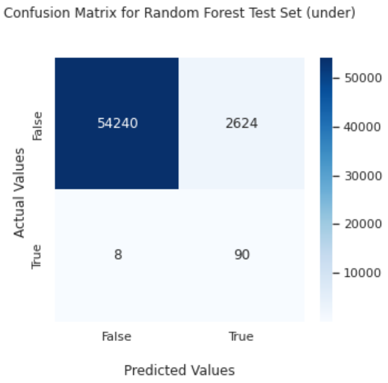
Fig 7.



Confusion Matrix for Random Forest Test Set (Smote)

Next, RUS was used, by reducing the majority class down to equal the number of minority samples. Fig 8. provides the confusion matrix for RUS on the Random Forest model. This provides the best recall score of 0.92, but both MCC and F1 drop dramatically. The model loses almost all ability to predict false positives. This could become an issue for merchants and business' if they keep flagging customers for fraudulent activity.

Fig 8.



The proceeding confusion matrix in Fig 9. represents a hybrid version combining both SMOTE and RUS. By reducing the majority class and oversampling the minority class, the data was able to keep more of its signal, enabling the model to generalize better to unseen data. The recall score for the hybrid model came out with the second highest recall score of 0.87, whilst also maintaining decent F1 and MCC scores.

Fig 9.

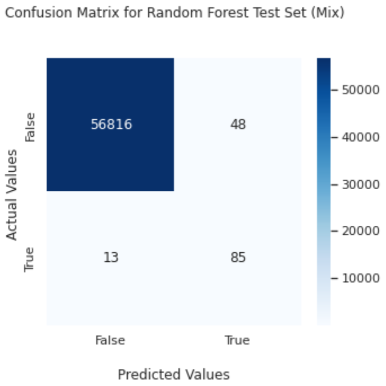


Fig 10. Show the confusion matrix for a Random Forest model using class weights to balance the classes, rather than a specific resampling method. Here, the parameters we manually input, as opposed to choosing one of the parameters more generic options, e.g. ‘balanced’. With a class weight ratio of 0.1/0.99, the model returns a recall of 0.82, with an MCC and F1 score of 0.89 each. These were the highest MCC and F1 scores out of all the sampling iterations. This could be an ideal model for those wanting to find a healthy balance between finding false negatives and false positives.

Fig 10.

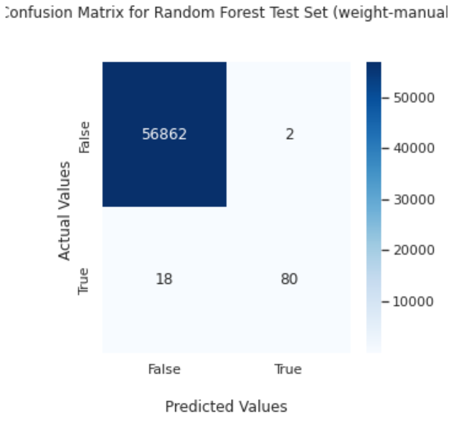
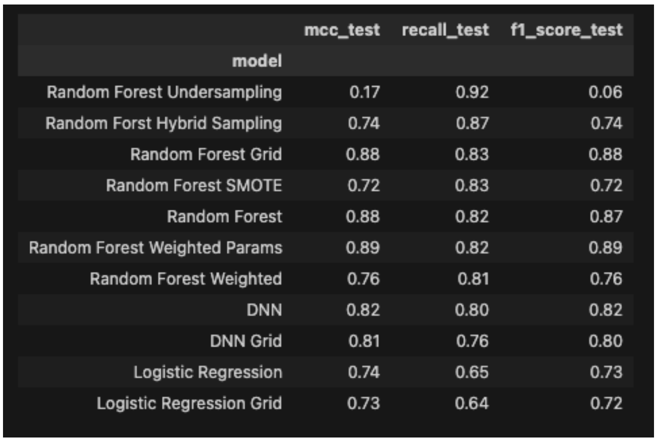


Fig 11. is a tabulation of all the models run ranked by recall score. Depending on what priorities users of these models have, they can pick and choose which sampling methods best suite their business needs. If catching fraudsters is critical at the expense of customer convenience, then using the under sampling method might be most fruitful. If however, they want to balance their losses from fraud against the potential loss of customer business, then maybe using the hybrid sampling approach would be most suitable.

Fig 11. Table of Model Metrics Ranked by Recall



	mcc_test	recall_test	f1_score_test
model			
Random Forest Undersampling	0.17	0.92	0.06
Random Forst Hybrid Sampling	0.74	0.87	0.74
Random Forest Grid	0.88	0.83	0.88
Random Forest SMOTE	0.72	0.83	0.72
Random Forest	0.88	0.82	0.87
Random Forest Weighted Params	0.89	0.82	0.89
Random Forest Weighted	0.76	0.81	0.76
DNN	0.82	0.80	0.82
DNN Grid	0.81	0.76	0.80
Logistic Regression	0.74	0.65	0.73
Logistic Regression Grid	0.73	0.64	0.72

6. Conclusion

The aim of this study was to provide a more robust methodology for detecting credit card fraud by comparing three very different machine/deep learning models, and then employing various resampling methods onto the best performing model. The main focus, however, was on resampling. The objective behind this, was to rebalance the classes. This enabled the models to better generalize to new data, whilst also removing class bias. Originally, the models were only seeing signal in the majority class, however, by resampling, the models were able to predict fraud to a higher degree, although losing the predictive ability for non fraud instances.

Aktar, Masud and Sakib in (Aktar et al., 2021, pp. 1–4) had similar results to this study, when using Random Forest with RUS. The authors noted a recall score of 95%, whilst also seeing a big drop off in the F1 score, insinuating that they too saw an increase in false positives.

Ileberi, Sun and Wang in (Ileberi, Sun and Wang, 2021, pp. 165286–165294) saw great improvements in their MCC scores across all of their models with AdaBoost and SMOTE dealing with the class imbalance issue. However, due to SMOTE being fitted before the train/test splitting, there test data will not be representative of real world data, therefore cannot be a fair comparison to the results in this research study.

Moving forward, other more efficient and effective sampling techniques can be employed to further enhance the methodology in this research paper. Such sampling techniques include, the robust Tomek Links and Edit Nearest under sampling methods. In addition, with more computing power, a more comprehensive grid search can be performed on the various models. This might help avoid overfitting, which plagued the models in this research study.

7. References

Brownlee, J. (2020). 'Tour of Evaluation Metrics for Imbalanced Classification'. (Machine Learning Mastery), 8 January. Available at: https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/ (Accessed: 25 May 2022).

Chicco, D. and Jurman, G. (2020). 'The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation'. England: BioMed Central Ltd (BMC Genomics, 21), 21 (1), p. 6. doi: 10.1186/s12864-019-6413-7.

Kampakis, D. S. (2021). 'What is the F-1 measure and why is it useful for imbalanced class problems?' (The Data Scientist), 19 February. Available at: https://thedatascientist.com/f-1-measure-useful-imbalanced-class-problems/ (Accessed: 25 May 2022).

Kanstrén, T. (2021). 'A Look at Precision, Recall, and F1-Score'. (Medium), 19 May. Available at: https://towardsdatascience.com/a-look-at-precision-recall-and-f1-score-36b5fd0dd3ec (Accessed: 25 May 2022).

Aktar, H., Masud, M. A., Aunto, N. J. and Sakib, S. N. (2021). 'Classification Using Random Forest on Imbalanced Credit Card Transaction Data'. in. - *2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, pp. 1–4. doi: 10.1109/STI53101.2021.9732553.

Deshpande, A., Kamath, C. and Joglekar, M. (2019a). 'A Comparison Study of Classification Methods and Effects of Sampling on Unbalanced Data'. in. - *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 1056–1063. doi: 10.1109/ICSSIT46314.2019.8987801.

Deshpande, A., Kamath, C. and Joglekar, M. (2019b). 'A Comparison Study of Classification Methods and Effects of Sampling on Unbalanced Data'. in. - *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, IEEE, pp. 1056–1063. doi: 10.1109/ICSSIT46314.2019.8987801.

Ileberi, E., Sun, Y. and Wang, Z. (2021). 'Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost'. (- IEEE Access, 9). doi: 10.1109/ACCESS.2021.3134330.

Ileberi, Emmanuel, Sun, Y. and Wang, Z. (2021). 'Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost'. (IEEE Access, 9), 9. Available at: https://xploreqa.ieee.org/document/9651991

(Accessed: 26 May 2022).

Mondal, I. A., Haque, M. E., Hassan, A.-M. and Shatabda, S. (2021). ‘Handling Imbalanced Data for Credit Card Fraud Detection’. in. - *2021 24th International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6. doi: 10.1109/ICCIT54785.2021.9689866.

Muaz, A., Jayabalan, M. and Thiruchelvam, V. (2020). ‘A Comparison of Data Sampling Techniques for Credit Card Fraud Detection’. West Yorkshire: Science and Information (SAI) Organization Limited (International journal of advanced computer science & applications, 11), 11 (6). doi: 10.14569/IJACSA.2020.0110660.

Parekh, P., Rana, C., Nalawade, K. and Dholay, S. (2021). ‘Credit Card Fraud Detection with Resampling Techniques’. in. - *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–7. doi: 10.1109/ICCCNT51525.2021.9579915.

Tyagi, R., Ranjan, R. and Priya, S. (2021). ‘Credit Card Fraud Detection Using Machine Learning Algorithms’. in. - *2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 334–341. doi: 10.1109/I-SMAC52330.2021.9640822.

Brownlee, J. (2020a). ‘Random Oversampling and Undersampling for Imbalanced Classification’. (Machine Learning Mastery), 15 October. Available at: https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/ (Accessed: 25 May 2022).

Brownlee, J. (2020b). ‘SMOTE for Imbalanced Classification with Python’. (Machine Learning Mastery), 17 January. Available at: https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/ (Accessed: 25 May 2022).

Barkved, K. (2022). *Credit Card Fraud Detection and AI*. (Obviously.ai). Available at: https://www.obviously.ai/post/credit-card-fraud-detection-with-machine-learning (Accessed: 23 May 2022).

Credit Card Fraud. (n.d). (LII / Legal Information Institute). Available at: https://www.law.cornell.edu/wex/credit_card_fraud (Accessed: 23 May 2022).

Fraud - The Facts 2020. (2021), p. 21. Available at: https://www.ukfinance.org.uk/system/files/Fraud%20The%20Facts%202021-%20FINAL.pdf (Accessed: 23 May 2022).

Lewis, R. (2022). *Internet Sales as a Percentage of Total Retail Sales (Ratio) (%)*. (Office for National Statistics). Available at: https://www.ons.gov.uk/businessindustryandtrade/retailindustry/timeseries/j4mc/drsi (Accessed: 23 May 2022).

Machine learning for fraud detection. (2022). (Ravelin). Available at: https://www.ravelin.com/insights/machine-learning-for-fraud-detection (Accessed: 23 May 2022).

MACHINE LEARNING GROUP, U. (2018). *Credit Card Fraud Detection*. (Kaggle). Available at: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud (Accessed: 23 May 2022).

Vidhya, A. (2020). ‘Imbalanced Classification | Handling Imbalanced Data using Python’. (Analytics Vidhya), 23 July. Available at: https://www.analyticsvidhya.com/blog/2020/07/10-techniques-to-deal-with-class-imbalance-in-machine-learning/ (Accessed: 26 May 2022).

Vladimiro, R. (2016). ‘Is it reasonable to exclude outliers in your training dataset for your classifier?’ (Quora), 1 August. Available at: https://www.quora.com/Is-it-reasonable-to-exclude-outliers-in-your-training-dataset-for-your-classifier.

Singh, K. (2020). ‘How To Dealing With Imbalanced Classes in Machine Learning’. (Analytics Vidhya), 6 October. Available at: https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/ (Accessed: 25 May 2022).

Brownlee, J. (2020). ‘A Gentle Introduction to Imbalanced Classification’. (Machine Learning Mastery), 14 January. Available at: https://machinelearningmastery.com/what-is-imbalanced-classification/ (Accessed: 26 May 2022).

Kumar, S. (2020). *How to remove Multicollinearity in dataset using PCA?* (Medium). Available at: https://towardsdatascience.com/how-to-remove-multicollinearity-in-dataset-using-pca-4b4561c28d0b (Accessed: 23 May 2022).

8. Bibliography

Ferreria, L. (2018). ‘Credit Card Fraud Prediction - [RF + SMOTE]’. (Kaggle). Available at: https://www.kaggle.com/code/kabure/credit-card-fraud-prediction-rf-smote/notebook.

luke4u. (2021). *Credit Card Prediction / Fraud Detection Models*. (Github). Available at: https://github.com/luke4u/Credit-Card-Prediction/blob/master/Fraud%20detection%20models/Credit_card_fraud_detection-RF_gridsearch.ipynb (Accessed: 23

May 2022).

LVING. (2017). ‘SMOTE with Imbalance Data’. (Kaggle). Available at: https://www.kaggle.com/code/qianchao/smote-with-imbalance-data.

SHAIKH, P. A. (2020). ‘SMOTE and undersampling-credit card fraud dataset’. (Kaggle). Available at: https://www.kaggle.com/code/parvezahmedshaikh/smote-and-undersampling-credit-card-fraud-dataset/notebook.

Yong, M. (2022). ‘Credit Card Fraud Detection with Scikit-learn’. (Kaggle), 1 May. Available at: https://www.kaggle.com/code/ymingj/credit-card-fraud-detection-with-scikit-learn.

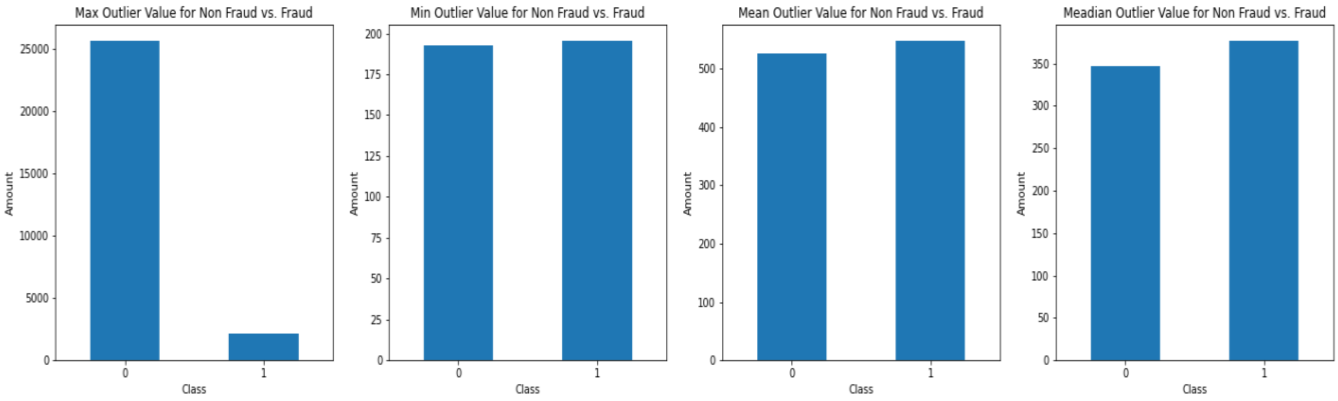
Brownlee, J. (2018). ‘Use Early Stopping to Halt the Training of Neural Networks At the Right Time’. (Machine Learning Mastery), 10 December. Available at: https://machinelearningmastery.com/how-to-stop-training-deep-neural-networks-at-the-right-time-using-early-stopping/ (Accessed: 25 May 2022).

Shmueli, B. (2019). *Matthews Correlation Coefficient is The Best Classification Metric You've Never Heard Of* . (Medium.com). Available at: https://towardsdatascience.com/the-best-classification-metric-youve-never-heard-of-the-matthews-correlation-coefficient-3bf50a2f3e9a (Accessed: 22 May 2022).

Shmueli, B. (2020). ‘Matthews Correlation Coefficient is The Best Classification Metric You’ve Never Heard Of’. (Medium), 20 May. Available at: https://towardsdatascience.com/the-best-classification-metric-youve-never-heard-of-the-matthews-correlation-coefficient-3bf50a2f3e9a (Accessed: 25 May 2022).

9. Appendices

Appendix A:



Appendix B: Github Repository

https://github.com/niall-anthony-mcnulty/Credit-card-fraud-detection-sampling-methods/blob/main/FinalAssignment-ML:DL%20Evaluation.ipynb