Executive Summary:

This report presents a machine learning approach to detect credit card fraud using the Credit Card Fraud Detection dataset available on Kaggle. The dataset contains anonymized credit card transactions labeled as fraudulent or genuine, and the goal is to develop a model that can accurately identify fraudulent transactions to prevent financial loss. The proposed approach involves data exploration, feature selection using PCA, and model selection using oversampling techniques like SMOTE and cross-validation with k-fold. The selected models include logistic regression, random forest, Support Vector Machine, and XGBoost. The proposed model achieved a high AUPRC score of 88.57% in detecting fraudulent transactions.

Introduction:

Credit card fraud is a prevalent problem that affects financial institutions and customers worldwide. In this project, I will analyse customer-level data that has been collected and analysed during a research collaboration of Worldline and the Machine Learning Group. According to Kaggle, the dataset contains transactions made by credit cards in September 2013 by European cardholders.This dataset presents transactions that occurred in two days, where I have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions. Features V1, V2, ... V28 are the principal components obtained with 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, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

Problem Definition:

The Credit Card Fraud Detection dataset contains transaction details such as the transaction amount and time, as well as whether the transaction is fraudulent or not. The goal is to develop a model that can accurately predict whether a transaction is fraudulent or not based on these features. The dataset is highly imbalanced, with only 0.172% of transactions labeled as fraudulent, which poses a challenge to traditional machine learning algorithms.
Dataset contains only numeric input variables which are the result of a PCA transformation. PCA stands for Principal Component Analysis, which is a technique used to transform a dataset with a large number of variables into a smaller set of variables while retaining most of the original information. This is achieved by identifying the principal components, which are linear combinations of the original variables that explain the maximum amount of variance in the data. The PCA-transformed information provided for the remaining transaction details except transaction amount and time is essentially a condensed version of the original data that still retains most of the important information. However, one drawback of using PCA is that the transformed variables may not have a clear interpretation in terms of the original variables. This can make it difficult to interpret the results of the analysis and may limit the usefulness of domain-specific knowledge in the analysis. Additionally, the process of selecting the number of principal components to retain can be somewhat subjective and may require some trial and error.

Data exploration and description:
Initially, I investigated the distribution of the 'time' and 'amount' features that were not transformed by PCA. I found that in the case of legitimate transactions, they tended to follow a normal distribution with lunchtime and afternoon as the average, whereas in the case of fraudulent transactions, there was a higher possibility of them occurring outside of business hours. In particular, many transactions were attempted late at night or early in the morning. Additionally, I observed that the transaction amounts for fraud were generally small. After data exploration, I used PCA for feature selection and chose the major components that preserved 80% of the explained variance. I also confirmed that the 'time' and 'amount' variables had little correlation with other variables, and thus I added them as independent variables after scaling.
In data such as fraud detection, where data is unbalanced, you should be careful about removing outliers. Removing an outlier can help improve the performance of the model on a given dataset, but it lacks generalization performance because the outlier in the normal transaction may be incorrectly trained as a fraudulent transaction or the outlier in the fraudulent transaction may be trained as a normal transaction.
Removing an outlier may prevent the model from properly recognizing the minority class, so it is recommended to use all data to train the model, and to evaluate the minority class of data samples using appropriate metrics

Model:
In the real world, learning models from these imbalanced datasets is a challenge because data is constantly updated. Generating a large number of minority class data or overfitting the model with hyperparameter tuning optimization can increase performance on the current training set, but it is likely that generalization performance will decrease. If the data is constantly updated, the model must also be re-evaluated and updated periodically. This ensures that the model always adapts to the latest data and maintains generalization performance.
Therefore, the purpose of modeling is to experiment with various feature selection and resampling methods and to make the model more logical and robust, rather than increasing the model performance on a given dataset easily due to overfitting. Feature selection allows the model to focus on important features without considering non-critical features. Resampling resolves the imbalance in the dataset and helps ensure that the model is not biased. However, increasing the amount of minority classes too much or decreasing the majority classes too much when resampling can negatively affect the model's learning and generalization. If you increase the amount of minority classes too much, the model will learn more minority class data. However, this can result in excessive reflection of fake minority class data, making it easier to overfit and lower generalization performance. On the other hand, if too many classes are reduced to match the ratio in the unbalanced data, the data loss can be large, resulting in a lower overall predictive performance for the model.
In this study, I propose a novel approach to feature selection by comparing four different methods: principal component analysis (PCA) based on main component contribution, analysis of variance (ANOVA) F-test, distribution-based feature selection, and mutual information-based feature selection. Our objective is to determine which method is most effective at identifying the most informative variables for use in a machine learning model.

To evaluate the performance of these methods, I conducted a series of experiments on several datasets with different features. Specifically, I compared the precision and recall of models trained using each of the four feature selection methods.
Additionally in this study, two resampling methods were compared for their effectiveness in improving the performance of a machine learning model. The first method involved resampling only the training set using k-fold cross-validation, while keeping the validation set untouched. The second method involved splitting the dataset into training and test sets from the outset, and then applying oversampling and undersampling techniques to the training set to address class imbalance. The resampling was done at appropriate ratios, and the resulting augmented data was used to train the model, followed by validation using a small subset of the original data.
To prevent data leakage, resampling was done exclusively on the training set in the first method, while in the second method, the test set was kept separate from the resampling process. The two methods were evaluated based on their ability to improve the performance of the machine learning model, using appropriate evaluation metrics.
Furthermore, showing that pre-data segmentation oversampling, which often occurs as a false example of resampling, can lead to overfitting.
Synthetic Minority Over-sampling Technique (SMOTE) was used for over-sampling. To increase the minority class data, smote uses the k-nearest neighbors (KNN) algorithm to generate synthetic data.
Undersampling methods such as ENN, which adjust the decision boundaries between classes, can easily improve performance. However, this approach removes some of the information from majority classes and induces the model to focus on only a few classes, posing an over-consensus risk. So I chose random undersampling for generalization performance.
Algorithms considered for model evaluation included logistic regression, random forest, support vector machine, ensemble model, and XGBoost.
Among them, the XGBoost algorithm, known for its outstanding performance, was chosen to compare the performance of various models. I also compared the XGBoost with logistic regression, random forests, support vector machines, and their ensemble models using the best-selected model. Deep learning models were not chosen because they have high computational costs, difficult learning processes, and low flexibility for new data inflows, while not guaranteeing better results due to the nature of this dataset. The model was evaluated using precision, recall, F1 and AUPRC.
It is not appropriate to use accuracy as an evaluation metric in an unbalanced amount of class data sets. This is because the negative class data is much larger, resulting in relatively high classification accuracy for positive classes. Therefore, other metrics such as precision and recall are more important in these situations.
Precision represents the percentage of actual fraud samples predicted by the model.
Recall represents the percentage of actual fraudulent samples in the actual fraud class samples.
The F1-score is calculated as the harmonic mean of precision and call. Therefore, it represents the model's ability to predict both precision and recovery rates.
The AUPRC (area under the precision-call curve) calculates the area under the precision-call curve. Precision call curves are graphs used to display precision and call values for various thresholds. The AUPRC calculates the area under this precision call curve, and the higher the

value, the better the model. AUPRC represents the proportion of predicted results that correspond to the actual amount of data.
Hyperparameter tuning can have a significant impact on the performance of the model, but may not be effective for models with few hyperparameters or with proper defaults. In addition, selecting hyperparameters based on a validation data set can result in models optimized only for that data set, which can degrade the generalization performance of the new data set. Therefore, the need for hyperparameter adjustment depends on the structure of the model and the characteristics of the dataset, and in some cases, the defaults can actually yield better performance. In this regard, I selected a model that omitted hyperparameter tuning, and the result of hyperparameter tuning for this model was not better than the default value. Therefore, I decided that it would be appropriate to use the default values for the hyperparameters of this model.
In summary, this work proposes a framework for detecting fraudulent transactions on an unbalanced dataset by combining various machine learning models with feature selection methods, resampling and validation techniques. The performance of the model was comprehensively evaluated by evaluating the model using AUPRC, precision, recall, and F1 scores.

Results and Findings:
The results of this study suggest that the second method about resampling, which involves explicit separation of the test set and resampling only the training set, is more effective in improving the model's performance. This approach not only prevents data leakage, but also provides a more robust evaluation of the model's generalization ability. Therefore, I recommend this approach for future studies that involve resampling techniques to address class imbalance.
Moreover, the research demonstrates that distribution-based feature selection outperforms other methods in terms of positive discrimination model performance. In distribution-based feature selection, I selected features by examining their distributions and comparing the distributions of the positive and negative classes. I selected features with significant differences in the variances of the probability density functions for each class.
In summary, our optimized model extracted features using distribution-based feature selection, explicitly separated the test set, oversampled only the training set, and utilized XGBoost. This model achieved a high AUPRC score of 88.27%, precision of 77%, recall of 88%, and F1 score of 82%, proving to be effective in preventing financial losses. I tested the generalization performance by using original data that was not exposed to training or resampling.
Using all 30 linearly independent and unrelated variables in the same model resulted in an AUPRC score of 88.57%, precision of 88%, recall of 86%, and F1 score of 87%. Despite increasing the number of variables by almost double and increasing complexity, the fraud detection performance was weakened by a 2% decrease in recall score, which is in trade-off relationship with precision.
However, depending on the purpose of use, it may be worth considering allowing slight overfitting using more features to increase precision in order to reduce false positives for normal customers and minimize unnecessary customer inconvenience.
Furthermore, data exploration revealed that fraudulent transactions were more prevalent during off-hours, such as late at night or early in the morning, and that they typically involved smaller

transaction amounts. The practical application of this model involves financial institutions using it to detect fraudulent transactions and prevent financial losses.

Conclusions and Future Work:

In conclusion, The proposed approach utilizing various machine learning models and resampling and validation techniques demonstrated high generalization performance in detecting fraudulent activities in imbalanced credit card transaction datasets. However, as fraudsters continually develop new tactics, it is essential to continuously enhance the model's performance to stay ahead. Future work may include integrating domain knowledge to analyze the most critical features for fraud detection using the original data and further improving interpretability.