### Introduction

The case study addresses the critical problem of detecting fraudulent credit card transactions, which is an issue of immense importance due to the significant financial losses and erosion of trust in financial institutions that such fraud can cause (HackerNoon, n.d.). Financial fraud, particularly in the realm of credit card transactions, is a pervasive issue that can undermine consumer confidence and result in substantial monetary losses for banks and individuals alike. Solving this problem is essential not only for mitigating financial losses but also for maintaining the integrity and trustworthiness of financial systems (HackerNoon, n.d.).

Detecting and preventing credit card fraud is of paramount importance for both financial institutions and their customers. Effective fraud detection systems help reduce financial losses by promptly identifying and stopping fraudulent transactions. This, in turn, safeguards customer assets, maintains the integrity of financial systems, and preserves the trust that customers place in their financial institutions (HackerNoon, n.d.). The ability to detect fraudulent activities quickly and accurately also reduces the operational costs associated with manual review processes, allowing financial institutions to allocate resources more efficiently and effectively.

The dataset used in the case study contains transactions made by credit cards in September 2013 by European cardholders. This dataset includes transactions that occurred over two days, with 492 frauds out of 284,807 transactions, making it highly unbalanced (HackerNoon, n.d.). The unbalanced nature of the dataset reflects the real-world scenario where fraudulent transactions are rare but impactful, requiring sophisticated techniques to identify them accurately without generating a high number of false positives.

### Methods and Results

**Data Preparation**

The first step in data preparation involved understanding the distribution of fraud cases, which constituted only 0.173% of the total transactions (HackerNoon, n.d.). This initial analysis was crucial for developing a strategy to handle the imbalanced nature of the data. Correlation analysis was conducted to identify relationships between variables, revealing that certain variables like V11 and V17 had strong correlations with fraudulent transactions (HackerNoon, n.d.). These correlations were leveraged to improve the model's predictive power.

The data was then sampled to create a balanced dataset for training the models, including 10,000 valid transactions and 492 fraud cases. This resampling was necessary to ensure that the models could learn the characteristics of fraudulent transactions effectively. The 'Amount' variable was standardized to normalize the data, ensuring that all variables contributed equally to the model's performance (HackerNoon, n.d.). Standardizing the data helped in mitigating the impact of variables with large numerical ranges, thereby improving the model's accuracy.

**Problem-Solving Approach**

The problem was tackled by building two outlier detection models using the PyOD library: the K-Nearest Neighbors (KNN) Detector and the One-Class SVM Detector. The dataset was split into independent variables (features) and the target variable (class), followed by splitting into training and testing sets (HackerNoon, n.d.). This split ensured that the models could be trained on one part of the data and tested on another to evaluate their performance effectively.

**Modeling Techniques**

1. **K-Nearest Neighbors (KNN) Detector:** This model identifies outliers based on the distance to the kth nearest neighbor. It used parameters such as contamination rate (0.0172) and number of neighbors (5) (HackerNoon, n.d.). The simplicity and effectiveness of KNN in outlier detection made it a suitable choice for this problem.
2. **One-Class SVM Detector:** This model is effective for high-dimensional data and was also used to detect anomalies in the dataset (HackerNoon, n.d.). One-Class SVM is known for its robustness and ability to handle complex data structures, making it ideal for high-dimensional fraud detection tasks.

**Choice of Methods/Models**

The KNN model was chosen for its simplicity and effectiveness in outlier detection, while the One-Class SVM was selected for its robustness in handling high-dimensional datasets. These models were appropriate due to the nature of the fraud detection problem, which involves identifying rare but significant anomalies (HackerNoon, n.d.). The combination of these models allowed for a comprehensive approach to detecting fraudulent transactions, leveraging the strengths of both methods.

**Evaluation Metrics**

The models were evaluated using metrics such as the Receiver Operating Characteristic (ROC) curve and Precision at rank n. The KNN model achieved a ROC score of 0.9566 and a precision at rank n of 0.5482, indicating good performance in distinguishing between fraudulent and non-fraudulent transactions (HackerNoon, n.d.). The ROC curve helped in understanding the trade-off between the true positive rate and false positive rate, while precision at rank n provided insights into the accuracy of the model's top predictions.

### Conclusion

**Implementation of Results/Model**

The trained models were used to predict fraud in the test set. For KNN, 372 fraud cases were correctly identified in the training set, demonstrating the model's effectiveness (HackerNoon, n.d.). The confusion matrix plotted for the training data showed the distribution of predicted versus actual classes, providing a visual representation of the model's performance.

**Actionable Consequences**

Implementing these models in a real-world scenario can significantly reduce the manual effort required for fraud detection, allowing financial institutions to focus on more complex cases. It also enhances the accuracy and speed of fraud detection processes, leading to reduced financial losses and improved customer trust (HackerNoon, n.d.). By automating the detection of fraudulent transactions, financial institutions can allocate their resources more efficiently and respond to fraudulent activities more swiftly.

**Lessons Learned**

The case study highlighted the importance of data preprocessing, such as balancing the dataset and normalizing variables. It also underscored the need for choosing appropriate evaluation metrics to accurately assess model performance (HackerNoon, n.d.). These insights are crucial for developing effective fraud detection models and can be applied to other similar problems in the future.

**Future Approaches**

For future improvements, the team could explore ensemble methods, combining multiple models to increase robustness. Additionally, incorporating real-time data and continuously updating the models with new fraud patterns can enhance detection capabilities (HackerNoon, n.d.). Ensemble methods, which combine the strengths of multiple models, could provide a more robust and accurate solution. Real-time data integration would enable the models to adapt to evolving fraud tactics, ensuring their effectiveness over time.

### References

HackerNoon. (n.d.). Credit Card Fraud Detection via Machine Learning: A Case Study. Retrieved from [HackerNoon](https://hackernoon.com/credit-card-fraud-detection-via-machine-learning-a-case-study).

Applied Sciences. (2022). Financial Fraud Detection Based on Machine Learning: A Systematic Literature Review. MDPI. Retrieved from [MDPI](https://www.mdpi.com/).