Credit Card Fraud Detection: A Machine Learning Approach with XAI

BENJAMIN A. HAGEN, Applied Data Science, Noroff University College, Norway

This study addresses the challenge of credit card fraud detection through machine learning (ML), using a dataset with 284,807 transactions. Exploratory Data Analysis (EDA) showed us an imbalanced dataset, with only 0.17% fraudulent transactions. To address this issue, we used Synthetic Minority Over-sampling Techniques (SMOTE) for the class balance. Two ML models were used: Logistic Regression and eXtreme Gradient Boosting (XGBoost). Logistic Regression got an accuracy of 98.02% and a ROC AUC score of 96.92%. The fine-tuned XGBoost model got an accuracy of 99.97%, with a ROC AUC score of 98.31%. We examined feature importance using SHapley Additive exPlanations (SHAP), highlighting variables such as V14, V4, and V12 in the decision-making. This study shows that the fine-tuned XGBoost model is not only effective, but also interpretable for detecting credit card fraud, offering valuable insights into the most influential features.

Additional Key Words and Phrases: Machine Learning, Fraud Detection, XAI, Logistic Regression, XGBoost

ACM Reference Format:
Benjamin A. Hagen. 2023. Credit Card Fraud Detection: A Machine Learning Approach with XAI. 1, 1 (November 2023), 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The digital era has changed in various sectors, including financial institutions. While these changes offer a wealth of new opportunities, they also come with their own new challenges, and the one we are looking at in this project is credit card fraud. It's important to study this domain because it helps protect the financial safety of millions of people and maintains trust in the digital banking system. According to the Federal Trade Commission[11], credit card fraud accounted for 46% of all identity theft reports in 2022, contributing to a financial loss of approximately $8.8 billion.

This project aims to carefully examine the effectiveness of the selected ML models in detecting credit card fraud. We aim to refine our models to optimize their performance, and go beyond just accurate predictions but also understand our models. This ensures that our models are not just "black boxes" but are interpretable and understandable.

In subsequent sections, we examine the dataset and related work. We will also preprocess the data and performed an EDA to understand and check the data quality. Afterward, we are going to focus on our selected models, Logistic Regression, and XGBoost. The models were trained and evaluated using a suite of metrics. Finally, XAI techniques will be implemented into our final model, followed by our conclusion of the project.

Author's address: Benjamin A. Hagen, benjaahagen@hotmail.com, Applied Data Science, Noroff University College, Norway.

Manuscript submitted to ACM

1.0.1 Dataset Overview. The dataset used in this study is sourced from Kaggle,[8] and has 284.807 credit card transactions recorded in September 2013 from European cardholders. The dataset had a significant class imbalance, with fraudulent transactions accounting for only 0.17% of the data.
The variables in the dataset were numerical and consisted of PCA-transformed features from V1 to V28. The only features in the datasets that were not transformed were the Time and Amount. Time represents the seconds between each transaction, and the amount represents the sum of the transactions. The feature class has two values: one for fraudulent transactions and zero for legitimate transactions.

2 RELATED WORK

With the digital transformation that has occurred in the financial sector, the use of ML has been improving the way we discover credit card fraud. Numerous studies have explored various ML models to identify fraudulent activity. This section provides an overview of some studies in the field.

The study by Dornadulaa and Sa (2019)[5] proposed a novel method for detecting fraud using Streaming Transaction Data. This method focuses on analyzing past transaction details from customers to extract behavioral patterns, which is done by clustering cardholders into different groups based on their transaction amounts. In the work by Cherif et al. (2022)[3], we are getting a systematic review of the challenges and solutions to credit card fraud detection. The study reviewed 40 articles from 2015 to 2021, focusing on different ML techniques, such as traditional and deep learning models. The work by Priscilla and Prabha (2020)[9] explored different ML strategies, such as supervised learning, unsupervised learning, ensemble learning, and deep learning, to efficiently address class imbalance and further elaborate on evaluation metrics for Credit Card Fraud Detection (CCFD). Various techniques, such as oversampling, undersampling, and hybrid sampling, have been considered to handle class imbalance. They also looked at a range of different ML algorithms, such as Support Vector Machines, Logistic Regression, and Decision Trees, among others, for the detection of credit card fraud.

The thesis by AlEmad (2022)[1] brings to light the rapid growth of credit card fraud and explores different ML techniques like K-Nearest Neighbour, Support Vector Machine, and Logistic Regression for fraud detection. The thesis concludes by identifying the Support Vector Machine as the best model, with an accuracy of 99.94% in finding fraudulent transactions. The study by Kochhar and Chhabra (2021)[6] employs multiple classification algorithms, such as Logistic Regression, Naive Bayes, AdaBoost, and Voting Classifiers to detect credit card fraud. This study emphasizes the importance of resampling techniques and the need for significant historical data for training models.

The literature on credit card fraud detection is extensive and diverse, utilizing a range of ML algorithms to address the problem. We aim to build upon this works by using ML algorithms and focusing on enhancing the model's interpretability, and performance. With this background, we now proceed to the methodology section.

3 METHODOLOGY

3.1 Preprocessing

Our dataset consists of 284,807 transactions and contains no missing values. The dataset had a large class imbalance, which is often the case in fraud-detection problems. Specifically, only 0.17% of the transactions were fraudulent. This is illustrated in Figure 1. This imbalance poses a risk of model bias towards the majority class, making it important to address this issue.

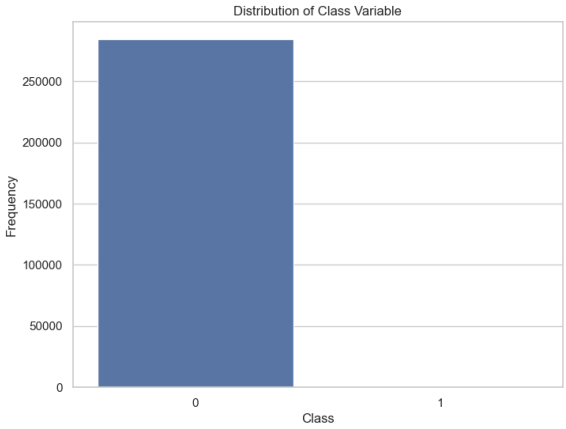

Fig. 1. Class Distribution before SMOTE

To reduce class imbalance, we used SMOTE on the training data. This technique synthesizes new examples in a dataset such that the class distribution is balanced. After SMOTE, both fraudulent and legitimate classes have equal representations, as shown in Table 1, allowing for more effective model training.

Table 1. Class Distribution after SMOTE

Class	Count
0	227451
1	227451

3.2 Data Analysis

For our credit card detection project, it was important to understand the data to build effective predictive models. Using visualizations and their statistical breakdowns to see patterns and outliers may help us find fraudulent activities.

3.2.1 Transaction Amount by Class. The box plot in Figure 2 shows the distribution of transaction amounts for both fraudulent (Class 1) and non-fraudulent (Class 0). For non-fraudulent transactions, we see a broader range of amounts, with high outliers reaching towards 25000. This indicates that a higher transaction does not necessarily correlate with fraud because many legitimate transactions also have high values. Fraudulent transactions seem to have a smaller range, with outliers spreading out thin. As shown in Table 2, the median value for fraudulent transactions is noticeably

lower than that for non-fraudulent transactions, suggesting that perpetrators might often attempt smaller transactions to bypass detection.

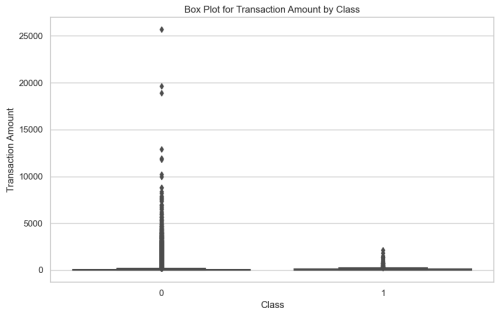

Fig. 2. Transaction Amount by Class

Table 2. Statistics for Amount by Class

Class	Count	Mean	Std	Min	25%	50%	75%	Max
0	284315.0	88.291022	250.105092	0.0	5.65	22.00	77.05	25691.16
1	492.0	122.211321	256.683288	0.0	1.00	9.25	105.89	2125.87

3.2.2 Transaction Amounts Over Time. The graph we can see in Figure 3 shows how the transactions change over time. For both classes, we see a change in the transaction amount throughout the timeline. Notably, there are clear spikes at specific intervals for fraudulent transactions, indicating periods of increased activity. These spikes might be representative of organized fraudulent schemes or potential weaknesses in the system. Understanding these periods can help in further investigation and preventive measures.

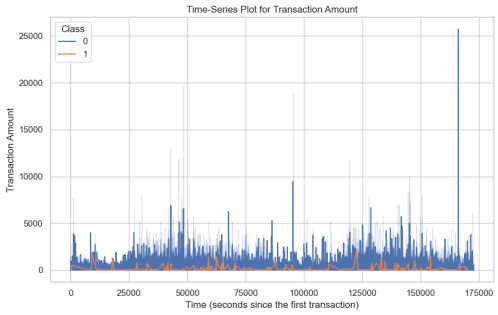

Fig. 3. Transaction Amounts Over Time

Table 3. Statistics for Time by class, where Time is the number of seconds since the first transaction in the dataset

Class	Count	Mean	Std	Min	25%	50%	75%	Max
0	284,315	94,838	47,484	0	54,230	84,711	139,333	172,792
1	492	80,747	47,835	406	41,241	75,568	128,483	170,348

These graphs and summaries can help us see patterns in the transaction data. By identifying these patterns, we can improve our model, and enhance their ability to detect credit card fraud more effectively.

3.3 Model Selection

In sensitive domains, such as credit card fraud, the selection of appropriate models is important. For both financial institutions and customers, there is a lot on the line. Finding fraudulent activities and at the same time minimizing false alarms are challenging tasks, making it important to carefully select and evaluate our models. Here, we will look into the reasons behind our two models: Logistic Regression and XGBoost.

3.3.1 Logistic Regression. Logistic Regression was the automatic choice for our study for several reasons. The most important aspect is the natural fit for binary-classification tasks[7]. Since our objective is to categorize transactions into two classes, fraudulent and non-fraudulent, Logistic Regression is a natural choice. In addition, it is efficient and can handle large amounts of data without high computational resources, making it ideal for real-time systems that process many transactions.

3.3.2 XGBoost. XGBoost, a gradient-boosting algorithm, is the second model in our project and is well suited for fraud detection. This algorithm is designed to scale billions of examples while requiring fewer computational resources[2], making it ideal for handling the large volumes of transactions typically encountered in fraud-detection systems.

The choice to use Logistic Regression and XGBoost was driven by an understanding of the challenges in fraud detection and the specific strengths of each model. This ensures that we can effectively protect financial assets while maintaining the trust of a large customer base. Before we move over to model evaluation, we split the data into 80% for training and 20% for testing.

3.4 Model Evaluation

Model evaluation was crucial for our project. Here, we examine the performances of our two models, Logistic Regression and XGBoost, using metrics such as ROC AUC, Precision, Recall, and F1-Score.

3.4.1 Logistic Regression Evaluation. Logistic Regression achieved an accuracy of 98.02% and a ROC AUC of 96.92%. The classification report in Table 4 shows good recall, but low precision. This indicates that the model has a high detection rate for fraud, but also produces many false positives, leading to more challenges.

Table 4. Classification Report and Performance Metrics for Logistic Regression

	Precision	Recall	F1-Score	Support
Class 0	1.00	0.98	0.99	56864
Class 1	0.07	0.90	0.13	98
Accuracy			0.9827	56962
Macro Avg	0.54	0.94	0.56	56962
Weighted Avg	1.00	0.98	0.99	56962

Accuracy	0.9801
ROC AUC	0.9692

3.4.2 XGBoost Evaluation. XGBoost had an accuracy of 99.94% and ROC AUC of 98.62%. It shows balanced Precision and Recall, as we can see in the classification report in Table 5, giving us a high fraud-detection rate while minimizing false positives. Thus, the XGBoost model is more suitable for this project.

Table 5. Classification Report and Performance Metrics for XGBoost

	Precision	Recall	F1-Score	Support
Class 0	1.00	1.00	1.00	56864
Class 1	0.82	0.85	0.83	98
Accuracy			0.9994	56962
Macro Avg	0.91	0.92	0.92	56962
Weighted Avg	1.00	1.00	1.00	56962

Accuracy	0.9994
ROC AUC	0.9861

3.5 Model Tuning

Now that we have our XGBoost model, we must fine-tune it to achieve the best performance. Hyperparameter tuning is a critical step in ML, particularly when high predictive accuracy is important. In credit card fraud detection, even a small increase in a model's performance can prevent large financial losses and customer churn. We used grid search and 3-fold cross-validation to fine-tune the XGBoost model. The best parameters and scores after tuning, and the optimal hyperparameters for XGBoost are listed in Table 6.

Table 6. Hyperparameters and Best Score for XGBoost

Hyperparameter/Metric	Value/Score
learning_rate	0.2
max_depth	5
n_estimators	300
Best Score	0.9998

3.5.1 Results Post-Tuning. The classification report and performance metrics of the fine-tuned XGBoost model are presented in Table 7.

Table 7. Classification Report and Performance Metrics for Fine-tuned XGBoost

	Precision	Recall	F1-Score	Support
Class 0	1.00	1.00	1.00	56864
Class 1	0.99	0.82	0.89	98
Accuracy			0.9996	56962
Macro Avg	0.99	0.91	0.95	56962
Weighted Avg	1.00	1.00	1.00	56962

Accuracy	0.9996
ROC AUC	0.9831

Grid search tests multiple hyperparameter combinations, whereas cross-validation reduces overfitting by training and validating the model on several data subsets. This process is very effective, but computationally difficult, and it took over 5 hours for our XGBoost model to finish. The updated model is more effective at detecting fraud. This shows that hyperparameter tuning has value, and is important for better fraud detection.

3.6 Explainability

For credit card fraud detection, a balance between accurate predictions and interpretability is required. As ML becomes increasingly common in financial systems, explanability should become equally important. An understandable ML model not only builds trust with its users, but also validates the model's decision beyond the performance metrics.

We chose to use SHAP in this project. Starting from game theory[10], SHAP offers a way to measure the extent to which each feature affects predictions. This ensures that the feature importance is fairly distributed and provides a detailed understanding of how each feature impacts the model's decisions[12]. Because credit card transactions have complex patterns, the insights from SHAP can be very useful.

4 EXPERIMENTATION AND RESULTS

4.1 Model comparison

This section provides an analysis of the ML models used for our credit card fraud detection.

4.1.1 Evaluation Metrics. The metrics used in this project included accuracy, ROC AUC, precision, recall, and F1-score, a comparison of all the models, and the ROC AUC can be seen in Table 8. The ROC curve can be seen in Figure 4. A model with a higher AUC value indicates that it is better to differentiate between fraudulent and nonfraudulent transactions.

4.1.2 Model Performance.

- **Logistic Regression:** Efficient but less precise, with a precision score of 0.07.
- **Original XGBoost:** Balanced in terms of precision and recall.

Table 8. Combined Performance Metrics and ROC AUC

Model	Accuracy	Precision (Class 1)	Recall (Class 1)	F1-Score (Class 1)	ROC AUC
Logistic Regression	0.9801	0.07	0.90	0.13	0.9692
Original XGBoost	0.9994	0.82	0.85	0.83	0.9861
Fine-tuned XGBoost	0.9997	0.99	0.82	0.89	0.9831

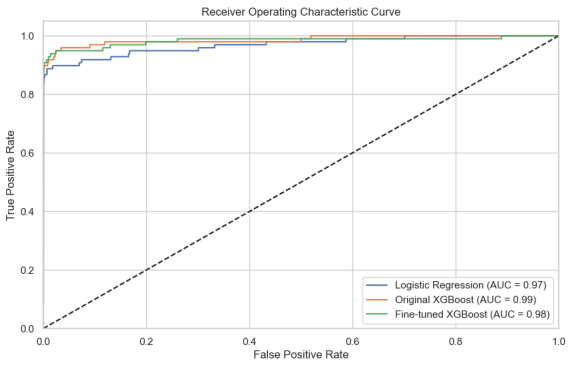

Fig. 4. ROC Curve

- **Fine-tuned XGBoost:** Exhibits a high-precision score of 0.99, excelling in accurate fraud detection.

Our fine-tuned XGBoost showed better results in terms of accuracy and precision, making it the choice we would recommend for real-world deployment. The original XGBoost is a reliable alternative that can be used if there are constrained computational resources. Our Logistic Regression model, while computationally effective, has such low precision that it will create false positives and other challenges for institutions and customers.

4.2 Insights from XAI

ML has become an important tool in finance for identifying credit card fraud. While ML offers us what we need to find out about credit card fraud, understanding the findings is also important. SHAP will be used as an interpretability tool in this project. We will analyze the SHAP plot from our fine-tuned XGBoost model, focusing on the most impactful features and their real-world implications.

4.2.1 Interpretation of the SHAP Plot. The SHAP plot measures each feature's influence on the predictions. The x-axis shows the SHAP value, with values further from zero indicating a stronger influence. The color gradient, where red indicates higher feature values and blue indicates lower values, helps us understand their impact on the predictions. In Figure 5 the SHAP plot is visualized.

4.2.2 Features and Their Impact.

- **V14:** The greatest impact of all features.

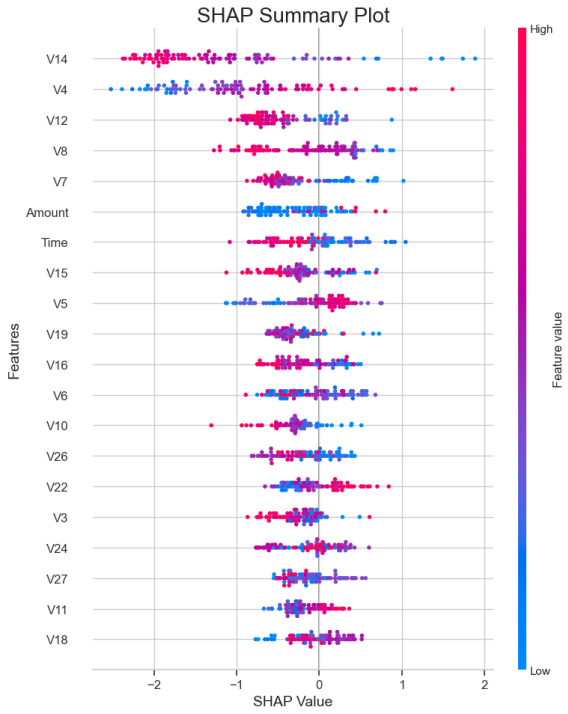

Fig. 5. SHAP Plot

- **V4:** Has high impact, mainly towards one category.
- **V12:** Shows an even impact, influencing the predictions both ways.
- **Other features like Time and Amount:** Have less impact but still play a role in the prediction.

Using insights from SHAP can improve fraud-detection systems, this helps prioritize monitoring high-impact features to identify evolving fraud tactics. With these insights, rule-based systems can be developed for reviews that focus on these features, and the models can be continually updated and retrained to adapt to changes in credit card fraud. As credit card fraud evolves, SHAP can serve as a guide, enabling us to respond effectively to these changes.

5 CONCLUSION

This project successfully used ML models to address the challenges of credit card fraud. Among the different models analyzed, we used Logistic Regression, XGBoost, and a fine-tuned XGBoost model.

We started with an EDA to understand the data and discover differences between fraudulent and legitimate transactions. We could see that fraudulent transactions had a higher average transaction amount(112.21) compared to legitimate transactions(88.29). Further in our EDA, we saw that fraudulent transactions tend to happen earlier in the data timeline, with a mean of approximately 87.747 seconds from the first transactions. The legitimate transaction's mean time was 94.838 seconds from the first transaction.

The Logistic Regression was highly effective, achieving an accuracy rate of 98.02% and an ROC AUC of 96.92%. Its precision was somewhat low, but the model showed good recall results, suggesting that it is strong in catching fraudulent transactions, but at the expense of occasionally false positives.
Our original XGBoost model achieved a more balanced performance with an accuracy of 99.94% and an ROC AUC of 98.62%. This balance shows that the model is capable of identifying fraud while minimizing false positives. Our last model, the fine-tuned XGBoost model, yielded an accuracy of 99.97% with an ROC AUC of 98.31%, providing nearly perfect precision in identifying fraudulent transactions.

Our SHAP value analysis gave us insight into the most important features in the fine-tuned XGBoost model. The features V14, V4, and V12 were shown to be most influential in the model's decision-making process. Additionally, features like V25, V9, and V17 were shown to have the least impact. These insights can guide stakeholders in determining what features they should allocate their resources.

However, the implementation of these models can be challenging. The computational resources the ML algorithms need, especially fine-tuned ones such as our XGBoost, can cause a significant challenge in environments with a high volume of transactions. Additionally, the adapted nature of these models can cause problems. If the models pay too much attention to certain details, it may be challenging to identify new types of fraud. In addition, when we encounter situations where fraud is uncommon, using only accuracy from a model can be misleading. To address these challenges, SHAP or other XAI tools can be used for a comprehensive evaluation of the model.

Despite the success of this project, there is room for improvement and further research. Ensemble methods, which combine the strengths of multiple models, offer the potential to create better fraud detection systems[13]. Neural networks and autoencoders that detect anomalies can be integrated into current systems[4]. There is also a need to optimize the computational efficiency of the model without worsening its fraud detection capabilities. Feature engineering and external data can help the model understand transactional behavior and further improve its predictive accuracy.

From the insights we obtained from this project, it is clear that credit card fraud detection is a major issue and a challenge that requires continuous improvements.

REFERENCES

[1] Meera AlEmad. 2022. *Credit Card Fraud Detection Using Machine Learning.* Master's thesis. Rochester Institute of Technology. https://scholarworks.rit.edu/theses/11318

[2] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 785–794. https://doi.org/10.1145/2939672.2939785

[3] Asma Cherif, Arwa Badhib, Heyfa Ammar, Suhair Alshehri, Manal Kalkatawi, and Abdessamad Imine. 2023. Credit card fraud detection in the era of disruptive technologies: A systematic review. *Journal of King Saud University - Computer and Information Sciences* 35 (2023), 145–174. Issue 1. https://doi.org/10.1016/j.jksuci.2022.11.008

[4] Frederico Luis de Azevedo, Karin Satie Komati, and Hilário Seibel Júnior. 2021. Detection of Credit Card Fraud in a Brazilian database using Autoencoder Neural Network. In *Sociedade Brasileira de Automática (SBA) XV Simpósio Brasileiro de Automação Inteligente - SBAI 2021.* https://doi.org/10.20906/sbai.v1i1.2796

[5] Vaishnavi Nath Dornadula and S Geetha. 2019. Credit Card Fraud Detection using Machine Learning Algorithms. *Procedia Computer Science* 165 (2019), 631–641. https://doi.org/10.1016/j.procs.2020.01.057

[6] Heena Kochhara and Dr. Yogesh Chhabra. 2021. A Novel Framework for Credit Card Fraud Detection. *Turkish Journal of Computer and Mathematics Education* 12, 11 (2021), 3189–3195. https://turcomat.org/index.php/turkbilmat/article/view/6362

[7] Maher Maalouf. 2011. Logistic regression in data analysis: An overview. *International Journal of Data Analysis Techniques and Strategies* 3, 3 (2011), 281–299. https://doi.org/10.1504/IJDATS.2011.041335

[8] MLG-ULB. n.d.. Credit Card Fraud Detection [Data set]. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud.

[9] C. Victoria Priscilla and D. Padma Prabha. 2020. Credit Card Fraud Detection: A Systematic Review. In *Intelligent Computing Paradigm and Cutting-edge Technologies. ICICCT 2019. Learning and Analytics in Intelligent Systems (Learning and Analytics in Intelligent Systems, Vol. 9)*. Springer. https://doi.org/10.1007/978-3-030-38501-9_29

[10] Lloyd Shapley. 1953. A value for n-person games. In *Contributions to the Theory of Games*. Vol. 2. https://doi.org/10.1515/9781400881970-018

[11] The Federal Trade Commission. 2023. *Consumer Sentinel Network Data Book 2022.* https://www.ftc.gov/reports/consumer-sentinel-network-data-book-2022

[12] Kyle Vedder. 2020. *An Overview of SHAP-based Feature Importance Measures and Their Applications To Classification.* Technical Report. https://vedder.io/misc/shap_for_classification.pdf

[13] Georgios Zioviris, Kostas Kolomvatsos, and George Stamoulis. 2022. An Intelligent Fraud Detection Model based on Deep Learning Ensembles. (June 2022). https://doi.org/10.21203/rs.3.rs-1175236/v1