1. **Research Question**

Financial fraud is a prevalent issue with significant consequences on individuals' daily lives. Key factors such as transaction type, amount, and changes in account balances play a critical role in identifying fraudulent behavior. Fraudulent transactions often start with amounts that appear typical to avoid suspicion, but unusual transaction patterns and abnormal account balance changes can be red flags. Investigating these features enables a deeper understanding of their contribution to fraud detection. According to a 2024 J.D. Power press release, 22% of credit card customers experienced financial fraud in the past year (Effler, 2024). Identifying significant features that lead to fraud, and gaining insights into criminal behaviors, is the first step in prevention. Recognizing these key factors helps mitigate the chances of fraud going undetected.

This analysis aims to improve the understanding of the factors contributing to fraudulent transactions. The research question guiding this study is: To what extent do transaction amount, transaction type, and account balance changes affect the classification of fraudulent transactions? The insights gained from this analysis can help financial institutions detect fraud early, preventing financial losses and protecting users. Machine learning models can automate the detection of potential fraud in large volumes of transactions (Nanda, 2024).

The alternative hypothesis for this study is that transaction amount, transaction type, and account balance changes significantly affect the classification of fraudulent transactions, as measured by performance metrics such as precision, recall, and F1-score.

**2. Data Collection**

PaySim is a payment testing provider that generates synthetic datasets mimicking real-world transactions. The dataset includes mobile payment transaction data collected over one month of financial logs in Africa. The original logs were provided by a multinational company. According to the research in the project "PAYSIM: A Financial Mobile Money Simulator for Fraud Detection," PaySim has been shown to accurately generate synthetic transactions that resemble legitimate transactions (Alonso Lopez-Rojas, 2016), making it a valuable resource for fraud detection research.

The PaySim dataset contains key features that are relevant to the research question, including transaction amount, transaction type, and account balance changes, along with the target variable, isFraud, to indicate fraudulent transactions. The advantages of using PaySim include privacy protection for customers, as the data is synthetic, and its suitability for fraud detection tasks (Alonso, 2016). Additionally, it is publicly available. However, since the dataset is synthetic, there may be potential class imbalances or missing data. To address these issues, techniques such as SMOTE (Synthetic Minority Oversampling Technique) will be employed to adjust class weights in the models. Missing data will be handled through imputation.

**3. Data Extraction and Preparation**

The PaySim dataset will be downloaded from Kaggle. The first step involves loading the dataset and removing columns that are not useful for the research. Relevant columns, such as transaction type, amount, oldbalanceOrg, newbalanceOrg, oldbalanceDest, newbalanceDest, and isFraud, will be retained. These columns will then be renamed for easier understanding. After inspecting the data, it was found that there were no null values or missing data.

A screenshot of a computer

Description automatically generated

A screen shot of a computer code

Description automatically generated

To handle missing values, imputation was used. One advantage of imputation is that it ensures no valuable data is lost, which is crucial when working with a large dataset. However, a potential disadvantage of imputation is that it can introduce bias into the data. Feature engineering was employed to create new, unique features that enhance the analysis. This approach is important because it helps establish relationships within the data, allowing the model to better understand and interpret the underlying patterns. For instance, the "balance change" feature was created to capture changes in the originating and destination accounts, providing insights into fluctuations in customer balances. Additionally, the "transaction ratio" was introduced to calculate the relationship between the originating balance and the transaction amount, offering further context to the data.A screen shot of a computer code

Description automatically generated

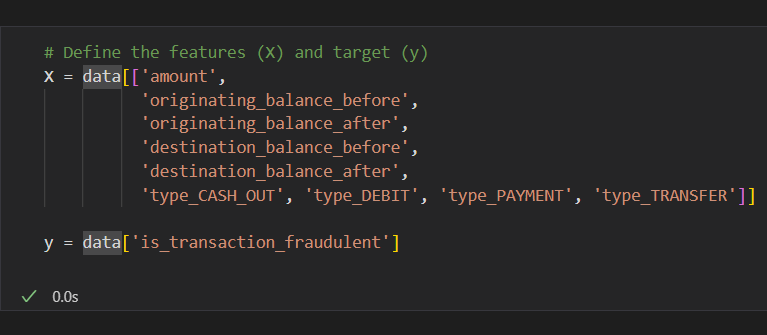
The advantage of feature engineering lies in its ability to create relevant variables from existing data, enhancing the model’s performance by identifying fraud detection patterns. However, a potential disadvantage is that incorrect or miswritten code could introduce errors, complicating the data analysis further along the project.

Encoding categorical data is essential for transforming it into numerical values that machine learning models can process efficiently. This step is critical for data normalization, ensuring a uniform scale across features. For instance, the 'type' column was split into binary columns representing each transaction type, allowing the model to recognize and interpret each customer’s transaction more effectively**.**

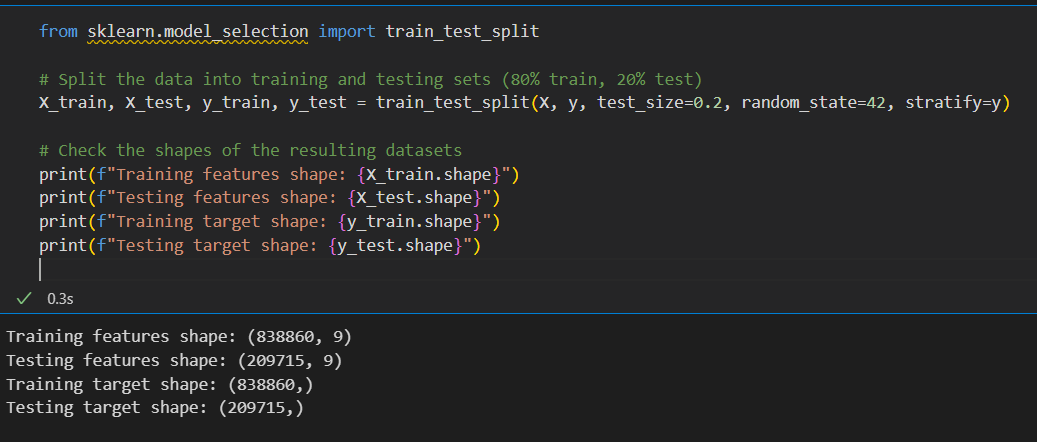
A screenshot of a computer

Description automatically generatedAn advantage to one hot encoding allows the model to easily read the dataset and have all information in the transformation. A disadvantage includes an increase in dimensionality which could result in longer training times.

Once the dataset is prepared, splitting the data into features (x) and target variable (y) is the next step. The target variable is “ isFraud” indicating if the transaction is fraudulent while the features include: “amount”, “type\_CASH\_OUT”, “type\_DEBIT”, “type\_Payment”, “type\_TRANSFER”, “balance\_change\_origin”, “balance\_change\_dest”, “transaction\_ratio”.



This step below allows for the model to focus on answering the research question on how the features interact with the target variable. Though if features are chosen poorly or there is a high correlation between features it can lead to overfitting. That would result in the need for regularization techniques. The dataset was split with 80% used for training and 20% was kept aside for testing.



Stratified sampling ensures fraudulent and authentic transactions are similarly being split into training and test sets. Fraud detection usually contains imbalanced classes and having stratified sampling provides a semblance of more balanced classes. A disadvantage to this tool would be is the dataset is not large enough the split could create more biased results.

SMOTE( Synthetic Minority Over-sampling Technique) addresses the class imbalance, it creates synthetic examples of fraudulent transactions. This is the best tool for this dataset because not only is fraud datasets already imbalance but this dataset is also synthetic which means it potentially has more imbalance. SMOTE ensures that the model doesn’t only predict authentic transactions, it creates an equal opportunity for fraudulent transactions as well.

A computer screen shot of a black background

Description automatically generated

SMOTE is effective for algorithms that are sensitive to class imbalance, such as decision trees, k-nearest neighbors, and neural networks. However, a disadvantage of using SMOTE is that it does not account for the complexity within the fraudulent data (minority class). SMOTE may generate synthetic data that does not accurately reflect the characteristics of the minority class, which could lead to misrepresentation of fraud patterns. This issue is particularly relevant if the minority class is sparse (Koluguri,2023) .

Before training the model, scaling the features ensures that all variables are on the same scale. In this project, the Standard Scaler was used. This step is crucial for improving model performance since K-Nearest Neighbor and Neural Networks are employed. Feature scaling ensures that all features contribute equally to the model, allowing the algorithm to find the optimal solution more effectively. Additionally, scaling mitigates the negative impact of skewed data and outliers, transforming the data into a standard range and reducing the influence of extreme values (Nanda,2024).

A screenshot of a computer program

Description automatically generated

An advantage of scaling the features is that it promotes faster convergence during training and enhances model performance. However, a potential disadvantage, similar to SMOTE, is that scaling can distort the inherent information within the features.

These steps were carried out using Pandas for data extraction and cleaning. Pandas provided an efficient and standardized process for extracting and cleaning the data, followed by feature engineering, one-hot encoding for categorical variables, and the application of SMOTE to address class imbalance. These steps were crucial in preparing the data for model training, allowing us to assess the impact of transaction amount, transaction type, and account balance changes in detecting fraudulent transactions. The approach ensured that all necessary tasks—data exploration, preparation, and analysis—could be performed within the same tool, simplifying the workflow.

1. **Analysis**

Several machine learning models were applied to classify fraudulent and legitimate transactions. The goal, in addition to identifying the relationship between transaction details and fraud, was to understand which algorithm performs best in classifying transactions. The algorithms used include Logistic Regression, Random Forest, K-Nearest Neighbors, and Neural Networks.

Logistic Regression is particularly useful in financial fraud analysis due to its binary classification nature, where outcomes are either fraudulent or legitimate based on certain features. Logistic Regression provides a probability score for each transaction, offering a clear decision point for analysts to take appropriate action. An advantage of Logistic Regression is its simplicity, which is why it was selected for this task. However, a limitation is its inability to model complex relationships effectively, as it assumes a linear relationship between the features and the outcome.

A computer screen shot of a program code

Description automatically generated

**A screen shot of a computer

Description automatically generated**

The logistic regression preformed above for class 0 gives a precision of 1.00 meaning 100% of the predictions for the authentic transaction were correct. The Recall and F1-Score both share the same 100% correctly identified authentic transactions. For class1 the fraudulent transactions, the precision is 89% of the predicted fraudulent transaction were correct. Some authentic transactions have been misclassified as fraudulent. The recall is very low at 22% demonstrating the model is missing most of the fraud transactions. F1-Score is also low recalling. This model is good at identifying non-fraudulent transactions but not at detecting fraudulent transactions. Precision-Recall curves look at imbalanced datasets where the minority class is less then the majority class.

A computer screen shot of code

Description automatically generated

A graph of a logistic regression

Description automatically generatedThe Precision-Recall curve shows as recall increases precision decreases. The precision represents the percentage of predicted fraudulent transactions that are legitimately fraudulent and the recall represents the percentage of legitimate fraudulent transactions were correctly identified by the model. This Logistic Regression model is not the best algorithm method for detecting fraudulent transactions from authentic data.

Random Forest uses multiple decision trees to improve classification performance. It uses ensemble learning to classify problems under decision tree classifiers. They are able to improve model accuracy. Random Forest also handles imbalanced datasets, it is very good at handling large number of features and noisy data. It also automatically handles feature selection, it was chosen because of its ability to capture complex relationships in a way Logistic Regression is not able to. Although one of the disadvantages for this algorithm include larger more technical models.

A computer screen shot of a program code

Description automatically generated

A screenshot of a computer

Description automatically generated

Above is a random forest classification report analysis that shows how the model performs for authentic and fraudulent transactions. Like the logistic regression model, the class 0 (authentic transactions) shows 100% precision, recall, and F1-Score. This provides proof that the legitimate, authentic transactions were correctly identified. Class 1 (Fraudulent transactions) has 97% precision, meaning that most of the fraudulent predictions were correct, but a small percentage were misclassified. The recall was 62%; the model only correctly identified less than two-thirds of the fraudulent transactions. The F1-score is 76% a balanced measure of precision and recall. This report shows there is a class imbalance where authentic transactions dominate the model’s prediction. The accuracy of the model is calculated at 100%, indicating that the model is correctly classifying 100% of transactions. Macro Average precision is the average of the precision scores across both classes and is high at 98%. The Macro Average for recall shows that, on average, the model is performing well across both classes at 81%. The weighted average across precision, recall, and f1-score is 100%. Although this seems good, it could be misleading because the majority class dominates the overall performance. The next step with this information is to evaluate the performance of a classification model using a confusion matrix.

A screenshot of a computer screen

Description automatically generated

The confusion matrix is broken down into True positive, true negatives, false positives , and false negatives. The number of fraudulent transactions correctly classified as fraudulent is 142 transactions. The number of non-fraudulent transactions correctly classified as 209,482 non-fraudulent. In the false positives there are 5 non-fraudulent transactions incorrectly classified as fraudulent. In the false negative there are 86 of fraudulent transactions incorrectly classified as non-fraudulent. The model identifies legitimate transactions, it is better at identifying fraudulent transactions then logistic regression.

A graph of a curve

Description automatically generated

The ROC Curve plots the recall (true positive rate) against the false positive rate that evaluates the model's ability to distinguish fraudulent and authentic transactions. The AUC is 97%, suggesting that the model is doing a good job of differentiating fraudulent and authentic transactions. The graph's curve indicates that the model can identify fraudulent transactions with high sensitivity. The ROC Curve plots the recall (true positive rate) against the false positive rate that evaluates the model's ability to distinguish fraudulent and authentic transactions. The AUC is 97%, suggesting that the model is doing a good job of differentiating fraudulent and authentic transactions. The graph's curve indicates that the model can identify fraudulent transactions with high sensitivity (Swastik, 2024).

K-Nearest Neighbors (KNN) is an algorithm that classifies data points based on the majority class. Its advantages include helping detect anomalies and outliers. KNN leverages users' transaction details to identify collaborative transactions. This algorithm is non-parametric and can handle multi-class problems. The disadvantage is that its computationally expensive with large datasets and sensitive to irrelevant features.

A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

The authentic transactions were predicted 100% correctly, as in the other algorithms. The fraudulent transactions precision is 89%, showing the transactions predicted as fraudulent were correctly classified. The model only identified 58% of the legitimate fraudulent transactions. Creating a moderate F1-score of 70%. The maco average was high, reflecting the ability of the model to predict each class accurately. This model is better than the logistic regression model but not as good as the random forest model.

The last algorithm I am using is a neural network. Neural networks are flexible models that can capture complex, non-linear relationships in data. This model was selected because they can handle complex data and provides higher accuracy for larger datasets. A disadvantage is that if there isn’t enough data, the neural network won't be good at generalizations.

A computer screen shot of a program code

Description automatically generated

A screenshot of a computer

Description automatically generated

The neural network classification report shows that for the fraudulent transactions, 96% of the predicted fraudulent transactions were correct. This is one of the highest precisions for class 1 that I have gotten from any of the other models. The recall, unfortunately, is the second lowest at 46%, leading to many false negatives (Prabhakaran, 2023).

The next step was a manual neural network model. The neural network model was trained over 10 epochs with a batch size of 32. The model quickly converges, reaching 0.9999 training accuracy after several epochs. The model was able to learn well on the training data. The validation accuracy also reached 0.9999 displaying the model does well at unseen data during training. The loss has a steady decrease, indicating the model has improved its predictions (Kwaku, 2023).

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

The test data shows the accuracy is at 0.9994 indicating the model predicts well on unseen data. The test loss is 0.0034 suggesting the model is making accurate predictions on the test set. The neural network did a very good job at authentic and fraudulent transaction predictions. Some challenges with the neural network are the same as most of the other models. The low class 1 recall and F-1 score show that there are many misses in fraudulent transactions (Prabhakaran, 2023).

Once all the models were run, a cross-validation was run to determine the most accurate model. Logistic Regression receives 0.9991 in cross-validation accuracy with an AUC of 0.9495. Random Forest receives a 0.9996 with an AUC of 0.9708. K-Nearest Neighbors receives a 0.9995 in cross-validation accuracy and an AUC of 0.8462. While the neural network receives an accuracy score of 0.9994 with an AUC of 0.9836. The best performances, according to AUC, were the Random Forest and Neural Network at distinguishing fraudulent from legitimate transactions (Kwaku).

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

Finally, Feature Importance answers the question that prompted this project. This technique will help identify the most influential feature in fraudulent transaction identification. In this instance, the features evaluated came from the Random Forest models due to it having the highest accuracy and AUC score (Yaqoob). This tool and Random Forest show how the decision-making process is based on the relationship between the transaction features and the target variable of fraudulent transactions. Feature importance allows for the interpretation of the model, making it easier for anyone reading the results, from analysts to customers, to understand. Along with its other advantages, feature importance helps improve decision making based on its results. A disadvantage is that feature importance does not provide causation. The fact that balance changes are important doesn’t explain why they are fraudulent (*Your guide to machine learning for fraud prevention)* .

A screenshot of a computer program

Description automatically generated

The top features include “originating\_balance\_before” – 0.2711, destination\_balance\_after” – 0.2685, “amount” – 0.2531, “destination\_balance\_before” – 0.1303, “originating \_balance\_after”- 0.0315. These features have the highest importance in predicting fraudulent transactions. This suggests that changes in transaction amount and account balances play a major role in detecting fraud. The next step is to link feature importance with logistic regression.

A screenshot of a computer program

Description automatically generated

The logistic regression coefficients directly indicate the strength and direction of the relationship between the feature and the target variable. The top coefficients in logistic regression are “originating\_balance\_before”, “originating\_balance\_after”, and “type\_CASH\_OUT”. Originating balance before has a coefficient of 12.1726 indicating a higher likelihood of fraud. Originating balance after has a coefficient of -10.6697, indicating a lower likelihood of fraud. The type(cash out) coefficient is 8.8813, indicating fraud is more likely with cashouts. From the logistic regression coefficients, features like “originating\_balance\_before” and “amount” have significant coefficients, confirming that the transaction amount and balance changes are strong fraud indicators.

1. **Data Summary and Implications**

In this analysis, the aim was to answer the research question: “ To what extent does the transaction amount, transaction type, and account balance change affect the classification of fraudulent transactions?”.

Several machine learning models were used throughout the application, indicating that transaction amount type and amount balance changes are influential features in classifying fraudulent transactions. Transaction amount and account balance changes both have a notable impact on the model's ability to detect fraud, indicated by logistic regression coefficients and random forest feature importance. Changes in the originating balance before, and the destination balance after a transaction also play a key role in fraud detection. Transaction types like “cash\_out are strongly associated with fraudulent behavior indicated by logistic regression coefficients. The findings suggest that fraud models based on these features can be effective. While the model performs very well, there is room for improvement in detecting fraudulent transactions (Prabhakaran,2023).

A limitation of this analysis is the class imbalance present in the dataset. Due to class 1 being outnumbered by class 0, even using SMOTE might not be enough to create a large enough dataset for class 1 with the same relationship the data has with itself. The imbalance affects the model’s ability to identify fraudulent transactions, leading to false negatives properly. A course of action based on the results is to apply calibration techniques like Platt Scaling to improve the probability estimates. For future studies it would be effective to investigate time series features. Since fraud can exhibit temporal patterns, time series features could improve the model’s ability to detect future anomalies. Another direction for future studies could be exploring anomaly detection techniques such as isolation forests to detect outliers in the data that might represent fraudulent transactions.

The analysis demonstrates that the transaction amount, transaction type, and account balance changes are critical factors in predicting fraudulent transactions. Correcting challenges with class imbalance through balancing techniques and threshold adjustments will improve fraud detection.

1. **Acknowledge sources, using in-text citations and references, for content that is quoted.**

 Alonso Lopez-Rojas, E., Axelsson, S., & Elmir, A. (2016, September). *PAYSIM: A financial mobile money simulator for fraud detection*. Research Gate. https://www.researchgate.net/publication/313138956\_PAYSIM\_A\_FINANCIAL\_MOBILE\_MONEY\_SIMULATOR\_FOR\_FRAUD\_DETECTION

Effler, G. (2024, November 7). *2024 U.S. Financial Protection Satisfaction Study | J.D. Power*. J.D Power. <https://www.jdpower.com/business/press-releases/2024-us-financial-protection-satisfaction-study>

Koluguri, R. R. (2023b, September 1). *End-to-end fraud detection analysis using CRISP-DM*. Medium.

<https://medium.com/@ruchithareddy.koluguri/end-to-end-fraud-detection-analysis-using-crisp-dm->92c3a7f2ca19

Kwaku Afriyie , J. (2023, January 14). *A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions*. Decision Analytics Journal. https://www.sciencedirect.com/science/article/pii/S2772662223000036

Lopez-Rojas, Edgar Alonso & Elmir, Ahmad & Axelsson, Stefan. (2016). PAYSIM: A FINANCIAL MOBILE MONEY SIMULATOR FOR FRAUD DETECTION.

Mishra, P. (2021a, August 9). *A realistic approach to IEEE CIS Fraud Detection*. Medium. https://medium.com/@mr.priyankmishra/a-realistic-approach-to-ieee-cis-fraud-detection-25faea54137

Nanda, A. (2024, October 12). *What is fraud analytics? techniques, workflows, and Tools*. DataCamp. https://www.datacamp.com/blog/fraud-analytics

Prabhakaran, S. (2023, September 10). *Lemmatization approaches with examples in Python*. Machine Learning Plus. https://www.machinelearningplus.com/nlp/lemmatization-examples-python/

Shivanipickl. (2023, November 16). *What is feature scaling and why does machine learning need it?*. Medium. https://medium.com/@shivanipickl/what-is-feature-scaling-and-why-does-machine-learning-need-it-104eedebb1c9

Swastik. (2024, December 20). *Smote for imbalanced classification with python*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/

Yaqoob, N. (n.d.). *K-nearest neighbors algorithm for credit card fraud detection*. PyFi. https://pyfi.com/blogs/articles/k-nearest-neighbors-algorithm-for-credit-card-fraud-detection

*Your guide to machine learning for fraud prevention*. Ravelin. (n.d.). https://www.ravelin.com/insights/machine-learning-for-fraud-detection