CREDIT CARD FRAUD DETECTION MODEL

CIS 5450 – Big Data Analytics

Project – Spring 2024

Group Members:

- **Priyam Shah**
- **Muskaan Beriwal**
- **Saaransh Pandey**

OBJECTIVE

Credit card fraud is a pervasive issue that affects millions of individuals and businesses worldwide, resulting in substantial financial losses and undermining trust in payment systems.

Early detection and precise identification of fraudulent activities are crucial in minimizing the financial impact and enhancing the security of credit transactions. Advances in technology, data analytics, and machine learning have significantly enhanced our capabilities in detecting and predicting credit card fraud, enabling more effective preventive measures and safeguarding both consumers and financial institutions against these illicit activities.

VALUE PROPOSITION

The Credit Card Fraud Detection Model offers a strategic advantage by leveraging cutting-edge technology, sophisticated data analytics, and advanced machine learning algorithms to proactively identify and prevent fraudulent transactions.

This model significantly reduces financial losses, protects consumer information, and maintains the integrity of payment systems. By implementing this robust solution, businesses can ensure secure transactions, enhance customer trust, and adapt swiftly to evolving fraud tactics, thereby maintaining a competitive edge in the marketplace.

TABLE OF CONTENTS

01 THE DATASET

02 MODELING

03 EXPLORATORY DATA ANALYSIS

04 IMPLICATIONS & INSIGHTS

05 CONCLUSION & FUTURE WORK

DATASET

The dataset comprises credit card transactions details, including user and card details, transaction timestamps, amounts, chip usage, merchant information, error indicators, and fraud flags.

Feature name	About the Feature
User	Identifier for the individual cardholder
Card	Unique identifier for the credit card used in the transaction.
Year	The year in which the transaction occurred
Month	The month in which the transaction took place.
Day	The day of the month on which the transaction was made
Time	The time at which the transaction was recorded.

Feature name	About the Feature
Amount	The monetary value of the transaction.
Use Chip	Describes the transaction method, indicating whether the card was swiped or a chip was used.
Merchant Name	A unique numeric code representing the merchant name who performed the transaction.
Merchant City	The city in which the merchant is located.
Merchant State	The state in which the merchant is located and if its international then it also contains name of the Country
Zip	The postal code of the merchant's location.
MCC	Merchant Category Code, which categorizes the merchant based on the type of goods or services provided.
Errors?	Indicates if there were any errors during the transaction (this column seems to contain no data in the provided entries).
Is Fraud?	Indicates whether the transaction was fraudulent or not.

THE DATASET-COLUMNS/FEATURES

- The dataset contains 24,386,900 rows (individuals) and 15 columns (features).
- The dataset has many NULL values, but had no rows that had been duplicated.
- The target feature is ‘Is Fraud’. It takes 1 in case of fraud and 0 otherwise.
- The non-fraudulent transactions dominate with 24,357,143 instances, comprising approximately 99.88% of the total. Conversely, the fraudulent transactions are a mere 0.12%, with only 29,757 cases.

We will now move on to the Exploratory Data Analysis part and see how we can get more insights about our data which will help us to answer some of the important questions we may have related to the problem.

EXPLORATORY DATA ANALYSIS

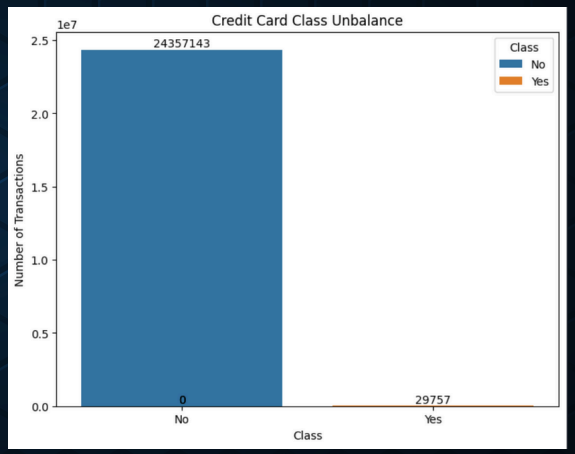
1) INITIAL ANALYSIS

- For our exploratory data analysis (EDA) of the credit card fraud detection model, we began by preprocessing the data to ensure accuracy and usability in our analysis.
- We also refined the 'Time' column by splitting it into separate 'Hours' and 'Minutes' columns, enhancing the granularity of our time analysis.
- In assessing data completeness, we found numerous null values in the 'Errors?' column, primarily because the majority of transactions did not report any errors.
- Our next step involved univariate analysis, where we examined each column individually to understand the distribution of data, specifically looking for patterns or anomalies that could indicate fraudulent behavior.
- Moving to bivariate analysis, we explored relationships between key variables and the target variable 'Is Fraud?'. This involved creating visualizations such as box plots, column charts, and line graphs to observe how different transaction characteristics—like the transaction amount, time of day, and merchant details—behaved in relation to fraudulent transactions.

2. Class Imbalance

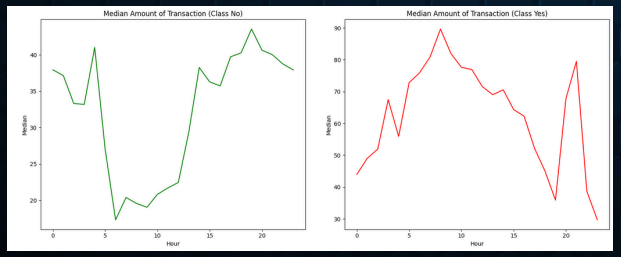
The non fraudulent transactions dominate with 24,357,143 instances, comprising approximately 99.88% of the total transactions. Conversely, the fraudulent transactions are a mere 0.12% with only 29,757 cases.

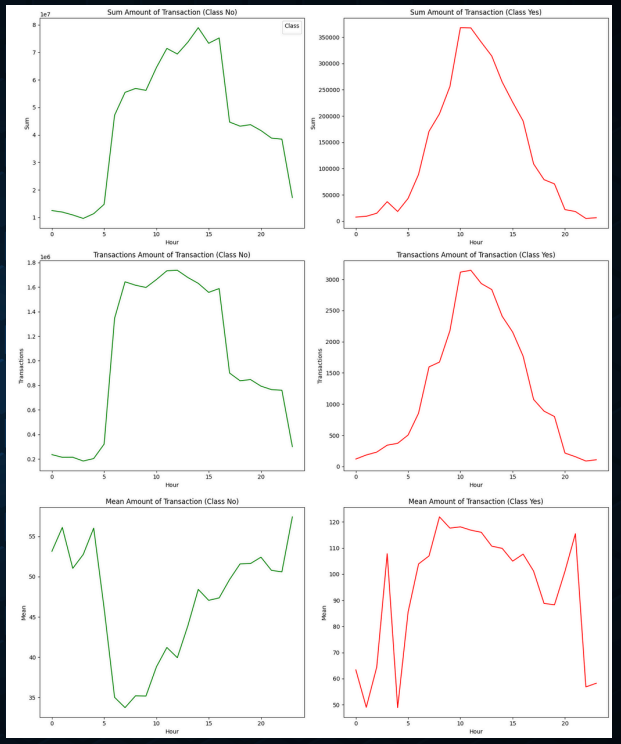
This severe imbalance can skew the predictive model, as standard algorithms will likely predict the majority class for all inputs, leading to high accuracy that's not indicative of true performance.



3. Analysis of Transactions by Time

Fraudulent activities peak between 10:00 and 14:00, indicating a high risk of fraud during midday hours, while legitimate transactions also peak at similar times but extend more evenly throughout the day. Fraudulent transactions typically involve higher average, maximum and median amounts compared to legitimate transactions, with notable spikes around 05:00, 09:00, 16:00, and 18:00.

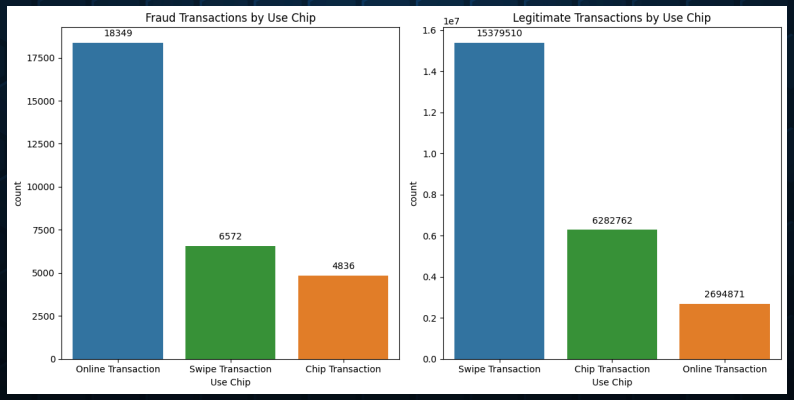




4. Mode of Transaction vs Fraud?

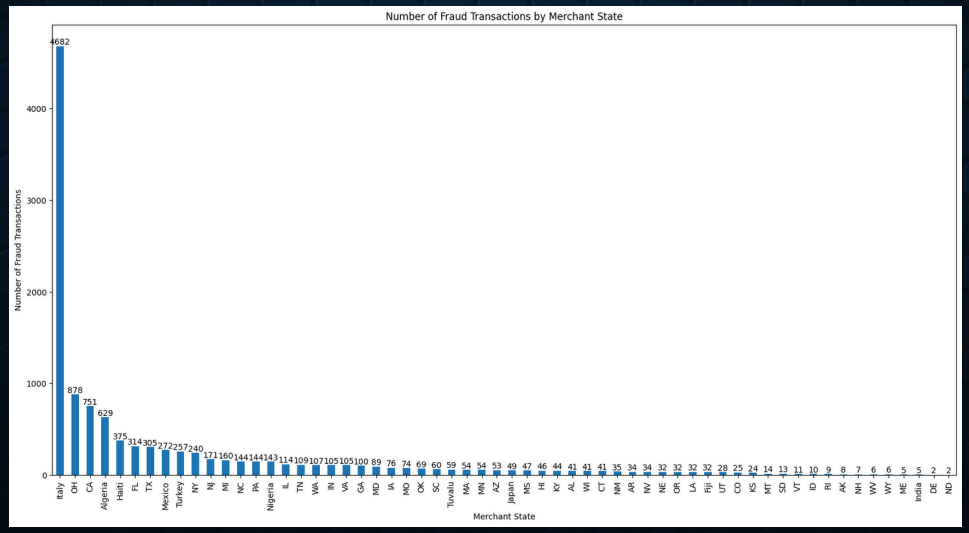
Online transactions are most susceptible to fraud, accounting for the highest count, followed by swipe transactions and chip transactions, suggesting a higher risk of fraud in less secure transaction methods.

Swipe transactions dominate in legitimacy, with chip transactions also common, while online transactions are significantly less frequent, indicating higher security and trust in physical transaction methods.



5. Distribution of Fraudulent transactions by Merchant State

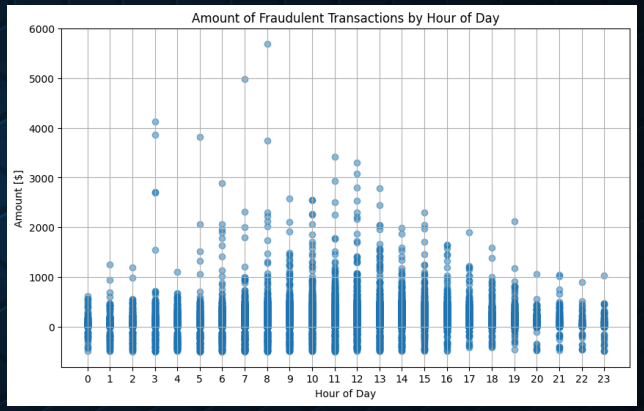
The skewed distribution raises questions about the efficiency of fraud detection systems across different states. It may indicate that certain regions are better at detecting and reporting fraud than others, or alternatively, that fraudsters find some states more appealing due to weaker detection systems.



6. Per Transaction Amount by Hour of Day

The scatter plot shows that fraudulent transactions occur at all hours, with the highest transaction amounts frequently exceeding $2,000 and peaking around $5,000.

There’s a noticeable increase in the frequency and value of transactions between midnight and the early morning hours, underscoring a vulnerability during times typically associated with lower vigilance and fewer active security measures.

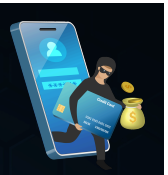


MODELING

Feature Engineering

- We started by performing one-hot encoding of the categorical variables, to create unique numerical representations, prevent misinterpretation as well as create independent, non-weighted entities.
- Next, we applied cyclical transformation on columns like month, day, time_in_hours and time_in_minutes. They were transformed into values of sin and cos.
- In the next step, we adjusted the 'Merchant City' column to better suit our analysis. Recognizing that a significant portion of transactions occurred online, transactions made online were encoded as '1', while all other transactions, irrespective of the actual city, were coded as '0'.
- The 'Errors' column primarily contained null values, indicating that most transactions were completed without any reported errors. We identified the unique non-null entries in the 'Errors' column. Subsequently, we replaced all null values with 'no_error' to explicitly denote transactions that occurred without complications.

Model Selection



To optimally analyze the data, we employed four distinct modeling techniques, each selected to provide a comprehensive representation of the dataset given its unique characteristics.

We initiated our approach with Logistic Regression, chosen for its straightforwardness and effectiveness in binary classification tasks.

Subsequently, we applied the Random Forest Classifier. This choice was driven by several advantages: its intuitive use and straightforward interpretability, robustness in capturing complex, non-linear relationships within the data, and its ability to process various data types effectively.

Following our initial modeling efforts, we progressed to incorporating XGBoost into our analysis. This advanced modeling technique builds upon multiple Decision Trees, enhancing their core characteristics while addressing some of their limitations.

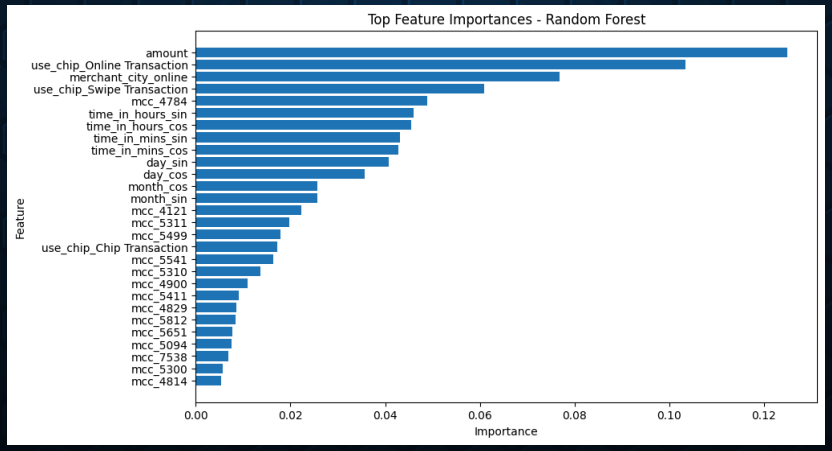
Additionally, we explored the potential of Neural Networks, to further our understanding of the data's underlying patterns.

Feature Importance

The transaction amount and method (online and swipe transaction) are the most significant predictors of fraud, highlighting the critical role these factors play in distinguishing fraudulent activities.

Temporal aspects and merchant category codes are also key features, emphasizing the importance of when and where transactions occur in fraud detection efforts.

After analyzing the plot, we decided to drop error columns, as it did not play much role in predictions.



Base Model – Checking for best performing method

We have used Logistic Regression as our base model and we try to fit only 10% of the data to our model due to computational limit. We inspected three different methods here:

1. using imbalanced dataset
2. under-sampling method
3. over-sampling method via SMOTE process

Based on below analysis, we decided to move forward with undersampling of the data,as it gave the best test recall.

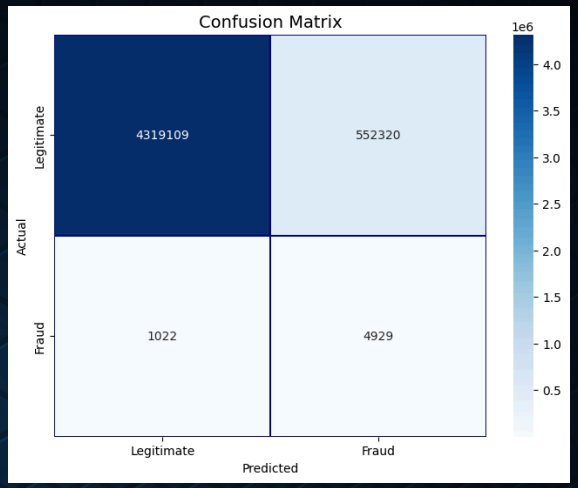
Oversampled Data	Undersample Data	Full Data
Training Accuracy: 88.95% Testing Accuracy: 92.52% Training Recall: 0.86 Testing Recall: 0.75 Training Precision: 0.91 Testing Precision: 0.11 Training ROC AUC Score: 0.89 Testing ROC AUC Score: 0.84	Training Accuracy: 85.72% Testing Accuracy: 88.46% Training Recall: 0.83 Testing Recall: 0.83 Training Precision: 0.88 Testing Precision: 0.08 Training ROC AUC Score: 0.93 Testing ROC AUC Score: 0.93	Training Accuracy: 99.17% Testing Accuracy: 99.17% Training Recall: 0.36 Testing Recall: 0.35 Training Precision: 0.88 Testing Precision: 0.90 Training ROC AUC Score: 0.89 Testing ROC AUC Score: 0.89

Modelling – Performance

*As we are focused on detecting fraudulent transactions correctly, we evaluated all the models using recall score.

We are pleased to report performance with our pre-processing, feature engineering and choice of models.

The Logistic Regression model gave us a baseline performance of 0.83 recall value. This indicated to us that it is well suited for the dataset at hand, with sufficient room for improvement. Further, AUC-ROC metric (particularly well suited to ignoring class distribution) also gave us a score of 0.86, indicating good starting trade-off between True Positive and False Positive Rates.

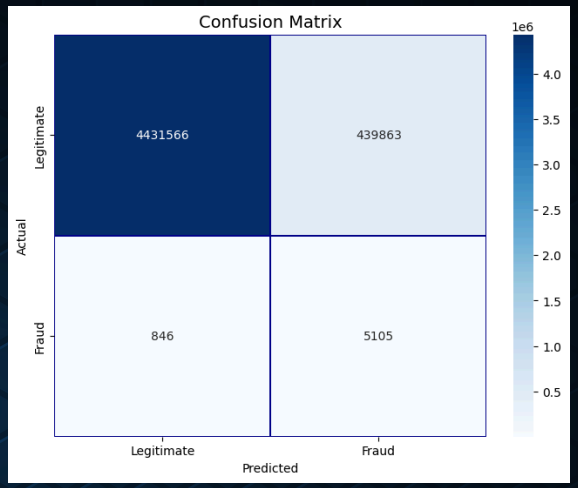


Training Accuracy: 85.60%
Testing Accuracy: 88.65%
Training Recall: 0.83
Testing Recall: 0.83
Training Precision: 0.88
Testing Precision: 0.01
Training ROC AUC Score: 0.86
Testing ROC AUC Score: 0.86

Modelling – Performance

The Random Forest model gave us test recall of 0.86 value. This indicated a considerable improvement over the baseline model, with further room for improvement. AUC-ROC metric gave us a score of 0.96, indicating a very good performance in identifying True Positives and False Positives.

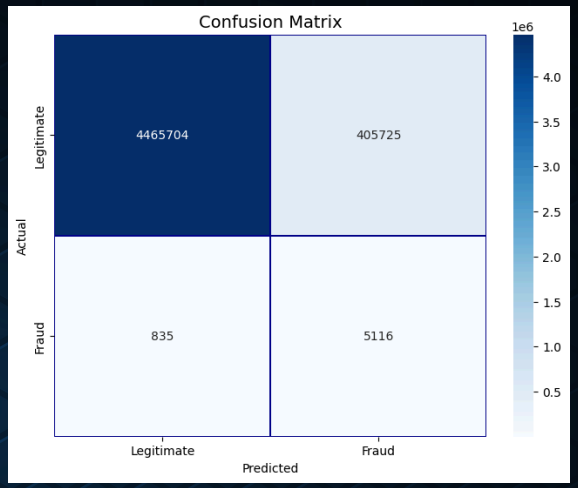
At the same time, we also observed training recall value is close to 1 and much higher than test recall value, suggesting the model might be overfitting.



Training Accuracy: 97.96%
Testing Accuracy: 91.02%
Training Recall: 0.98
Testing Recall: 0.86
Training Precision: 0.98
Testing Precision: 0.01
Training ROC AUC Score: 1.00
Testing ROC AUC Score: 0.96

Modelling – Performance

The XGBoost model, demonstrated high reliability in fraud detection with training and test recall of 0.9 and 0.86, respectively - effectively identifying a substantial proportion of fraudulent transactions. However, the model faced challenges with precision in the testing phase, where it scored notably low at 0.01, indicating a significant number of false positives. Despite this, the model excelled in distinguishing between classes, as evidenced by high ROC AUC scores of 0.98 and 0.96 in training and testing, respectively, showcasing its strong discriminative power though with room for improvement in specificity.



Training Accuracy: 91.85%
Testing Accuracy: 91.56%
Training Recall: 0.90
Testing Recall: 0.86
Training Precision: 0.94
Testing Precision: 0.01
Training ROC AUC Score: 0.98
Testing ROC AUC Score: 0.96

Modelling – Hyper-Parameter tuning

To enhance the predictive capabilities and robustness of our models, we recognized the necessity of hyperparameter tuning. The primary motivations for this process include improving model generalization, preventing underfitting and overfitting, enhancing convergence, and boosting overall model performance. These factors drove our decision to perform detailed hyperparameter tuning on several of our key models.

With the XGBoost model, we aimed to optimize its recall due to its critical role in accurately identifying true positives, especially given our focus on minimizing false negatives in fraud detection. Utilizing the RandomizedSearchCV from scikit-learn, we explored various parameter distributions, focusing on maximizing recall to ensure the detection of fraudulent transactions. The best parameters were determined after extensive testing, significantly improving the recall score, thus enhancing the model's ability to correctly identify fraudulent cases, which are the following:

Best Hyperparameters: {'subsample': 0.6, 'n_estimators': 300, 'max_depth': 10, 'learning_rate': 0.05}

Modelling – Hyper-Parameter tuning

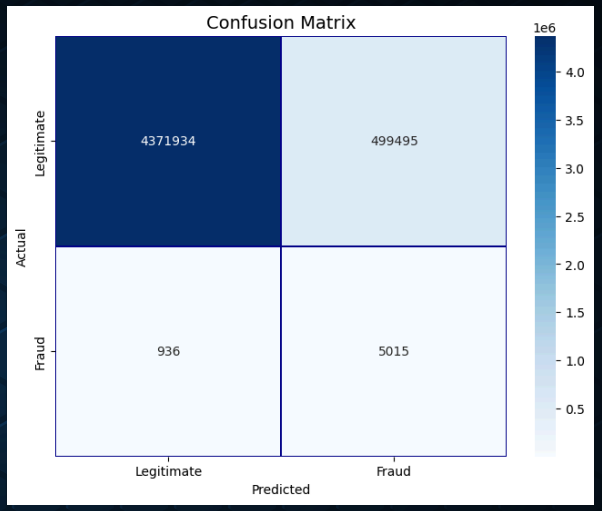
Next, we addressed the Random Forest model, recognizing its potential due to its similarity in performance to XGBoost. Here, the objective was again to enhance the model's recall. The tuning involved a randomized search across a defined range of hyperparameters, with particular attention to those affecting model depth and complexity to avoid overfitting while improving detection capabilities. The results are following;

Best Hyperparameters: {'n_estimators': 200, 'min_samples_split': 5, 'min_samples_leaf': 1, 'max_features': 'sqrt', 'max_depth': None}

To Conclude: XGBoost outperformed all the other models, keeping Recall score as the metric.

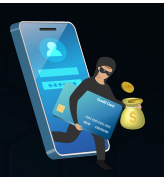
NN Modelling – Performance

The confusion matrix and performance metrics reveal significant aspects of our model's ability to detect fraud. The model demonstrated a strong ability to identify true positives, as evidenced by a recall score of 0.84, indicating that it successfully identified 84% of all fraudulent transactions. However, the precision was notably low at 0.01, highlighting a challenge in distinguishing between legitimate and fraudulent transactions without generating a high number of false positives. The ROC AUC score was robust at 0.95, demonstrating the model's effectiveness in discriminating between the classes across various threshold settings.



Testing Accuracy: 89.74%
Testing Recall: 0.84
Testing Precision: 0.01
Testing ROC AUC Score: 0.95

Results and Conclusion



INTERPRETABILITY VS PERFORMANCE

The balance between interpretability & performance is crucial; however, performance typically takes precedence to ensure high accuracy and recall in identifying fraudulent transactions, especially in operational settings where preventing fraud is critical.

High Accuracy and Recall - In credit card fraud detection, performance, specifically accuracy and recall, is paramount. High recall ensures that the majority of fraudulent transactions are detected, minimizing financial loss, while high accuracy reduces false positives, maintaining customer satisfaction and operational efficiency.

Rapid Response Capabilities - The ability to quickly identify and respond to fraudulent transactions is critical. Models with superior performance can process transactions in real time, providing immediate alerts and actions to prevent fraud before significant damage occurs.

Scalability and Adaptability - High-performance models can efficiently handle large volumes of transactions and adapt to new, sophisticated fraud techniques. This scalability is essential for maintaining security as transaction volumes and fraud schemes evolve.

CHALLENGES - LIMITATIONS - FUTURE WORK

- **Model Generalization and Data Diversity -** The robustness of fraud detection models can be improved with more diverse datasets, encompassing a broader range of transaction types and fraud scenarios. This would help in enhancing the model's generalization capabilities across different environments and conditions.
- **Mitigating Model Bias -** It's crucial to address and reduce biases that may occur due to overrepresentation or underrepresentation of certain types of transactions. Ensuring that the model performs well across all transaction segments is essential for fair and accurate fraud detection.
- **Real-Time Detection Capabilities -** Developing capabilities for real-time fraud detection can significantly enhance the effectiveness of preventive measures. This involves not only the speed of detection but also the accuracy with which models can operate in a live environment.
- **Adaptive Fraud Detection Techniques -** As fraud tactics evolve, so should our detection strategies. Future work could focus on adaptive algorithms that continuously learn and adjust to new fraudulent behaviors, potentially through the application of machine learning techniques that can evolve with changing patterns without requiring constant human supervision.


THANK
YOU