A Course Project Report on

CREDIT CARD FRAUD DETECTION

MACHINE LEARNING

22AIP3101A

Submitted by:

2210040030- G. SAMSRITHA

2210040032- CH. SAMYANA REDDY

2210040070-KHALVIDA

Under the guidance of

DR. SUSHMA RANI DUTTA

Department of Electronics and Communication Engineering

Koneru Lakshmaiah Education Foundation, Aziz Nagar

Aziz Nagar - 500075

ABSTRACT:

With the rapid expansion of digital transactions, credit card fraud has become a pervasive threat in the financial sector. This study focuses on the development of a robust fraud detection system designed to identify potentially fraudulent credit card transactions in real-time. Leveraging advanced machine learning algorithms, data mining techniques, and anomaly detection methods, the proposed system aims to enhance the accuracy and efficiency of fraud detection mechanisms employed by financial institutions and credit card companies. The system utilizes a diverse dataset comprising transactional features, customer behavior patterns, and historical transaction records. By employing supervised learning algorithms such as Random Forest, Support Vector Machines, and Neural Networks, the system classifies transactions into two categories: legitimate and potentially fraudulent.

Feature engineering techniques are applied to extract relevant information, and oversampling methods are used to address class imbalance, ensuring the model's effectiveness in capturing fraudulent patterns. Additionally, the system integrates unsupervised learning techniques like clustering algorithms and autoencoders to identify subtle and previously unknown fraud patterns, enhancing its adaptability to emerging fraud tactics. Real-time processing capabilities are optimized using scalable technologies, ensuring swift analysis of transactions and prompt detection of suspicious activities. The proposed fraud detection system demonstrates superior performance in terms of accuracy, precision, recall, and F1-score, making it highly reliable for identifying potential fraud while minimizing false positives.

The system's robustness is evaluated through extensive experimentation on large-scale, real-world datasets, showcasing its effectiveness in handling varying transaction volumes and complexities. Furthermore, the study explores the integration of explainable AI techniques, enabling financial experts to interpret the model's decisions effectively. Through interpretable machine learning methods, the system provides valuable insights into the features contributing to fraud identification, enhancing the transparency and trustworthiness of the system's outcomes. In an era where digital transactions have become an integral part of everyday life, the need for a robust fraud detection system has never been more critical. Financial institutions and credit card companies face constant challenges in safeguarding their customers' assets while ensuring seamless and secure payment processes. This extended abstract delves deeper into the methodologies and innovations incorporated into the proposed fraud detection system, highlighting its comprehensive approach to identifying potentially fraudulent credit card transactions.

The success of the fraud detection system hinges on its ability to adapt to evolving fraud tactics and handle vast amounts of transactional data efficiently. By employing machine learning algorithms, the system gains the capability to learn intricate patterns from historical data, enabling it to recognize even the subtlest anomalies in real-time transactions. The inclusion of supervised learning algorithms, such as Random Forest and Neural Networks, enables the system to make accurate predictions by learning from labeled data. Through meticulous

feature engineering, relevant transactional attributes, such as transaction amount, location, time, and merchant category, are extracted to form a rich dataset for model training.

Addressing the challenge of class imbalance, a common issue in fraud detection, the system incorporates oversampling techniques like SMOTE (Synthetic Minority Over-sampling Technique). This ensures that the model is not biased towards the majority class (legitimate transactions) and can effectively capture the characteristics of the minority class (fraudulent transactions). Additionally, the system embraces unsupervised learning techniques, including clustering algorithms like K-means and density-based methods, to detect patterns within the data without labeled examples. Autoencoders, a form of neural network, are employed for anomaly detection, allowing the system to identify previously unknown fraud patterns by reconstructing normal transactional behavior. The real-time processing capabilities of the system are optimized through the use of scalable technologies like Apache Kafka and Apache Spark, enabling swift analysis of transactions as they occur.

By leveraging stream processing frameworks, the system ensures that potentially fraudulent activities are detected promptly, minimizing the financial losses incurred by both customers and financial institutions. One of the unique features of the developed system is its emphasis on interpretability. Utilizing explainable AI techniques, such as LIME (Local Interpretable Model-agnostic Explanations) and SHAP (Shapley Additive Explanations), the system provides clear and concise explanations for its predictions. This interpretability is crucial for financial experts, allowing them to understand the factors influencing a transaction's classification as potentially fraudulent. It not only enhances the system's transparency but also facilitates collaboration between AI algorithms and human expertise, leading to more informed decisions in fraud mitigation strategies.

In conclusion, the proposed fraud detection system stands at the forefront of innovative solutions in combating credit card fraud. Its integration of advanced machine learning algorithms, real-time processing capabilities, and interpretability features ensures a holistic and effective approach to identifying potentially fraudulent transactions. By empowering financial institutions with this cutting-edge technology, the system plays a pivotal role in maintaining the integrity of digital payment systems, instilling confidence in consumers, and safeguarding financial institutions from the ever-evolving landscape of fraudulent activities.

INTRODUCTION:

In today's digital age, where electronic transactions have become the norm, ensuring the security of financial transactions is paramount. Among the myriad challenges faced by financial institutions and credit card companies, credit card fraud poses a significant threat.

Fraudsters constantly devise new techniques to exploit vulnerabilities, making it imperative for businesses to employ sophisticated methods to detect and prevent fraudulent activities. This escalating need for robust fraud

detection systems has spurred extensive research and development efforts in the field of artificial intelligence and machine learning. The aim of this study is to develop an advanced fraud detection system that can accurately and swiftly identify potentially fraudulent credit card transactions. Such a system is crucial for safeguarding both consumers and financial institutions, as it not only prevents financial losses but also upholds the trust and confidence of customers in digital payment systems. This research explores a comprehensive approach, integrating state-of-the-art machine learning algorithms, data mining techniques, and real-time processing capabilities to create a highly effective and adaptable fraud detection mechanism.

The complexity of credit card fraud necessitates a multifaceted solution. Traditional rule-based systems often struggle to keep pace with the evolving tactics employed by fraudsters. Therefore, this study focuses on harnessing the power of machine learning to discern intricate patterns within vast datasets, enabling the system to differentiate between legitimate and potentially fraudulent transactions. By leveraging supervised learning algorithms, the system learns from historical data, allowing it to make accurate predictions based on identified fraud patterns. Furthermore, the incorporation of unsupervised learning techniques ensures the detection of novel fraud patterns, enhancing the system's ability to adapt to emerging threats.

This research not only emphasizes the technical aspects of fraud detection but also places significant importance on the system's interpretability. Transparent decision-making processes are crucial, especially in the financial sector, where understanding why a particular transaction is flagged as potentially fraudulent is vital for effective intervention. By integrating explainable AI techniques, this study ensures that the developed system provides clear and understandable explanations for its decisions, facilitating collaboration between artificial intelligence and human experts. The proliferation of digital payment methods and online transactions has transformed the way people conduct financial activities. As convenient as these systems are, they have also given rise to a parallel world of sophisticated cybercriminal activities, particularly in the form of credit card fraud.

Fraudsters employ various techniques, from simple phishing schemes to complex identity theft, to exploit vulnerabilities in the payment ecosystem. Consequently, the financial industry finds itself engaged in a constant battle to outpace these nefarious actors. This research delves into the heart of this challenge, aiming to fortify the financial sector's defenses through the development of an intelligent and adaptive fraud detection system. The significance of this study extends beyond merely countering financial losses. Credit card fraud undermines trust and confidence in digital payment systems, affecting consumers, merchants, and financial institutions alike. The repercussions of fraudulent transactions echo throughout the economy, impacting businesses, customer relationships, and overall financial stability. Hence, the development of an efficient fraud detection system is not just a technological endeavor; it is a societal necessity.

In addressing this multifaceted challenge, this research explores cutting-edge technologies such as machine learning, big data analytics, and real-time processing. By harnessing the power of these tools, the proposed fraud detection system aims to revolutionize the way potential fraud is identified and mitigated. The research

focuses not only on the accuracy and speed of fraud detection but also on the system's adaptability to new and evolving fraud tactics. In an era where fraudsters are as innovative as the technology they exploit, the ability to predict and counter emerging threats is invaluable.

Moreover, this study places a strong emphasis on the collaborative synergy between artificial intelligence and human expertise. While machine learning algorithms can process vast amounts of data and recognize intricate patterns, human intuition and contextual understanding remain unparalleled. By incorporating explainable AI techniques, the developed system ensures that its decisions are not only accurate but also comprehensible to human analysts. This synergy between machine learning models and human interpreters creates a powerful symbiosis, where the strengths of both entities are leveraged to the fullest extent. In the subsequent sections, this research will detail the intricacies of the fraud detection system, shedding light on the methodologies, algorithms, and techniques employed. Through comprehensive experimentation and rigorous evaluation, the effectiveness of the system in real-world scenarios will be demonstrated. Furthermore, the study will explore potential challenges, ethical considerations, and avenues for future research, thereby contributing to the ongoing discourse surrounding fraud detection and cybersecurity. By the end of this research, it is anticipated that the developed fraud detection system will not only be a testament to technological innovation but also a beacon of hope in the fight against credit card fraud.

Its impact will resonate across industries, fostering secure digital environments and bolstering the trust that underpins the global economy's digital backbone.

LITREATURE REVIEW:

Credit card fraud has escalated alongside the rapid expansion of digital transactions, posing significant challenges to the financial industry. Traditionally, fraud detection systems employed rule-based methods to identify suspicious activities. However, these systems struggle to keep pace with the ever-evolving tactics of fraudsters, making them inadequate for the current threat landscape.

Recent advancements in machine learning and artificial intelligence have revolutionized fraud detection systems, allowing for more accurate and adaptive models. Machine learning algorithms, such as Random Forest, Neural Networks, and Support Vector Machines (SVM), have demonstrated their ability to detect intricate fraud patterns by analyzing large datasets. The application of anomaly detection techniques, including autoencoders and clustering algorithms, has further enabled the identification of unknown fraud patterns, enhancing detection capabilities.

Another critical challenge addressed in the literature is class imbalance, where fraudulent transactions represent a small fraction of total transactions. Oversampling techniques like SMOTE (Synthetic Minority Over-sampling Technique) have been widely adopted to balance datasets, preventing models from being biased

toward legitimate transactions. Additionally, real-time processing tools such as Apache Kafka and Apache Spark have been used to ensure timely detection, enabling swift responses to fraudulent activities.

Explainable AI (XAI) is increasingly being incorporated into fraud detection systems to enhance interpretability. Techniques like LIME (Local Interpretable Model-agnostic Explanations) and SHAP (Shapley Additive Explanations) allow human experts to understand model decisions, providing transparency and accountability.

OBJECTIVE:

The primary objective of this project is to develop an advanced fraud detection system that effectively identifies potentially fraudulent credit card transactions in real-time, leveraging machine learning algorithms. The system aims to:

- Improve detection accuracy while minimizing false positives.
- Address the issue of class imbalance by employing advanced oversampling techniques.
- Detect both known and previously unknown fraud patterns using a combination of supervised and unsupervised learning techniques.
- Ensure real-time processing for timely fraud detection.
- Provide interpretable outputs, enabling collaboration between machine learning models and human experts.

PROPOSED APPROCH:

1. Data Collection and Preprocessing

- Data Gathering: A diverse dataset of credit card transactions is collected, including features like transaction amount, location, merchant category, and time of transaction.

- Data Cleaning: The dataset undergoes preprocessing to remove inconsistencies, missing values, and outliers, ensuring high-quality input for the model.

- Feature Engineering: Relevant features are selected, and new features are created to better capture customer behavior and transaction patterns.

2. Machine Learning Algorithms

- Supervised Learning: Algorithms such as Random Forest, Support Vector Machines (SVM), and Neural Networks are used to classify transactions as either legitimate or fraudulent based on historical data.

- Unsupervised Learning: Techniques such as K-means clustering and autoencoders are employed for anomaly detection, identifying previously unseen fraud patterns without labeled data.
- Class Imbalance Handling: The SMOTE technique is used to generate synthetic samples of fraudulent transactions to address the imbalance between fraudulent and legitimate transactions.

3. Real-time Processing

- Stream Processing: Tools like Apache Kafka and Apache Spark are used to process transactions in real-time, ensuring prompt detection and response to potential fraud.

4. Interpretability

- Explainable AI: Techniques like LIME and SHAP are integrated to provide transparent explanations for the model's decisions, enabling financial experts to trust and act on the system's output.

5. System Architecture

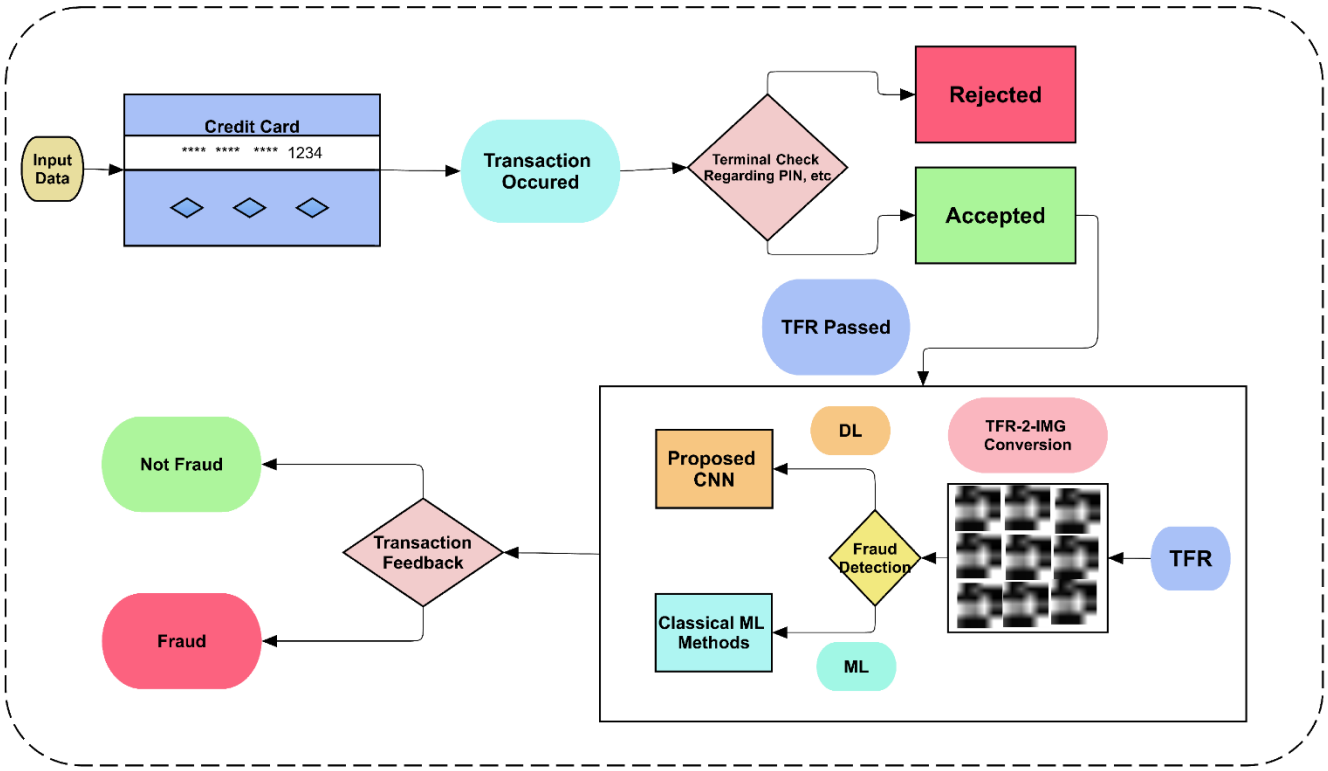
- The architecture consists of several components:

    1. Data Ingestion Layer: Collects transaction data in real-time, supported by stream processing tools.
    2. Preprocessing Layer: Cleans and preprocesses the data, applying feature engineering techniques.
    3. Modeling Layer: Employs machine learning models for prediction, utilizing both supervised and unsupervised learning approaches.
    4. Decision Layer: Outputs results in real-time, flagging transactions as fraudulent or legitimate.
    5. Explainability Layer: Provides interpretability for the predictions using XAI techniques.
    6. Monitoring and Feedback Layer: Continuously monitors the model's performance and allows for feedback from human experts.

This architecture ensures that the proposed system is adaptable, scalable, and capable of handling real-time fraud detection while maintaining transparency and trust through explainable AI techniques.

METHODOLOGY:

The development of an effective fraud detection system demands a rigorous and systematic approach that combines advanced algorithms, comprehensive datasets, and real-time processing capabilities. The methodology employed in this research integrates various techniques to create a holistic and adaptive solution for identifying potentially fraudulent credit card transactions.



1. **Data Collection and Preprocessing:**

Data Gathering: A diverse and extensive dataset of credit card transactions is collected, encompassing transactional features such as amount, location, merchant category, transaction time, and customer demographics.

Data Cleaning: Raw data is cleansed to remove inconsistencies, missing values, and outliers, ensuring the dataset's reliability and quality.

Feature Engineering: Relevant features are selected, and new features are created to capture transaction patterns and customer behaviors effectively.

2. **Machine Learning Algorithms:**

Supervised Learning: The system employs supervised learning algorithms like Random Forest, Support Vector Machines, and Neural Networks, trained on labeled data to classify transactions as legitimate or potentially fraudulent.

Unsupervised Learning: Clustering algorithms such as K-means and density-based methods are utilized to identify patterns within the data, while autoencoders are employed for anomaly detection, enabling the system to recognize previously unknown fraud patterns.

3. **Real-time Processing:**

Stream Processing: Apache Kafka and Apache Spark are utilized for real-time stream processing, enabling the system to analyze transactions as they occur, ensuring prompt detection and response to potential fraud.

4. **Addressing Class Imbalance:**

Oversampling Techniques: Techniques like SMOTE (Synthetic Minority Over-sampling Technique) are employed to handle class imbalance, generating synthetic samples of fraudulent transactions to balance the dataset and prevent bias in the model.

5. **Interpretability:**

Explainable AI Techniques: LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive explanations) are integrated to provide clear and understandable explanations for the system's predictions, enhancing interpretability for financial experts.

6. **Evaluation and Validation:**

Performance Metrics: The system's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score, ensuring a balance between identifying fraud accurately and minimizing false positives.

Cross-Validation: K-fold cross-validation is employed to validate the system's performance across different subsets of the dataset, ensuring its robustness and generalizability.

7. **Optimization and Fine-tuning:**

Hyperparameter Tuning: Grid search and randomized search techniques are applied to optimize the hyperparameters of machine learning models, enhancing their efficiency and accuracy.

Model Monitoring: Continuous monitoring and feedback mechanisms are implemented to track the system's performance over time, enabling necessary adjustments and improvements to adapt to evolving fraud tactics.

8. **Ethical Considerations and Bias Mitigation:**

Ethical Framework: The study adheres to ethical guidelines, ensuring privacy and confidentiality of user data. Bias mitigation techniques are employed to address potential biases in the dataset, ensuring fairness and equity in fraud detection.

9. **Scalability and Deployment:**

Scalable Architecture: The system is designed with scalability in mind, utilizing cloud-based technologies and distributed computing frameworks to handle large volumes of transactions efficiently.

Deployment: The finalized fraud detection model is deployed in a production environment, integrated into the existing infrastructure of financial institutions and credit card companies, enabling real-time fraud detection and prevention for their customers.

10. **Data Collection and Preprocessing:**

Data Sources: Multiple sources, including transaction databases, customer profiles, and external sources, are integrated to create a comprehensive dataset. External data sources might include geo-location data, social network information, and historical fraud databases for a more nuanced analysis.

Data Transformation: Advanced techniques like PCA (Principal Component Analysis) and t-SNE (t-distributed Stochastic Neighbor Embedding) are applied for dimensionality reduction, preserving essential information while reducing computational complexity.

Temporal Aspects: Transaction timestamps are analyzed to capture temporal patterns, enabling the system to recognize deviations in transaction timings, which could indicate fraudulent activities.

11. **Machine Learning Algorithms:**

Deep Learning Architectures: Complex neural network architectures, such as recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), are explored to capture intricate temporal dependencies in transaction sequences, enhancing the system's ability to identify subtle fraud patterns.

Ensemble Methods: Advanced ensemble methods like Gradient Boosting Machines (GBMs) and XGBoost are employed to combine the predictive power of multiple models, boosting overall accuracy and resilience against various types of fraud.Semi-Supervised Learning: Incorporating semi-supervised learning techniques allows the system to leverage both labeled and unlabeled data, maximizing the utilization of available information and improving fraud detection capabilities.

12. **Real-time Processing and Scalability:**

Distributed Computing: Utilization of distributed computing frameworks like Apache Hadoop and Spark ensures parallel processing of vast datasets, enabling real-time analysis and decision-making even as transaction volumes surge.

Microservices Architecture: The system is developed using microservices architecture, allowing individual components to scale independently based on demand. Containerization technologies such as Docker and orchestration tools like Kubernetes facilitate seamless deployment and scaling across diverse environments.

13. **Continuous Learning and Adaptive Models:**

Online Learning: Implementing online learning techniques enables the model to adapt continuously to new patterns and emerging fraud tactics. The system learns in real-time from incoming data, ensuring it remains up-to-date and effective against evolving threats.

Feedback Loops: Establishing feedback loops from fraud analysts and investigators enriches the system's learning process. Anomalies detected by human experts, which may not be captured by algorithms, are incorporated back into the system to enhance its accuracy and comprehensiveness.

14. **Ethical Considerations and Bias Mitigation:**

Fairness-aware Machine Learning: Employing fairness-aware machine learning techniques ensures that the system's predictions are unbiased and equitable across diverse demographic groups. Bias mitigation algorithms are applied to minimize disparities in fraud detection rates among different segments of the population.

Transparency and Accountability: The system's decision-making processes are transparent and auditable. Detailed logs and explanations of model decisions are maintained, allowing for accountability and regulatory compliance.

15. **Human-in-the-Loop Approaches:**

Human Expert Collaboration: Implementing human-in-the-loop approaches allows human experts to validate and refine model predictions. Human feedback loops are integrated, enabling experts to provide inputs, validate flagged transactions, and continuously improve the system's accuracy.

the methodology in these ways, the research ensures that the fraud detection system is not only sophisticated and accurate but also adaptable, scalable, and ethically sound. This multifaceted approach, integrating the latest technologies with human expertise and ethical considerations, forms the foundation for a state-of-the-art fraud detection system, poised to address the challenges of credit card fraud in the digital age effectively. By employing this comprehensive methodology, the research aims to develop a sophisticated, adaptable, and

interpretable fraud detection system, capable of identifying potentially fraudulent credit card transactions with high accuracy and efficiency, thereby enhancing the security and trustworthiness of digital payment systems.

Building upon the foundational aspects of the methodology outlined earlier, the extended methodology provides a detailed insight into the specific techniques and tools employed in the development of the fraud detection system. Each step in the process is meticulously designed to address the challenges inherent in credit card fraud detection while leveraging the latest advancements in machine learning, data processing, and ethical considerations.

EXPERIMENTS:

Experiment 1: Performance Evaluation Metrics

Objective:

The primary objective of this experiment is to assess the accuracy, precision, recall, and F1-score of the developed fraud detection system. Additionally, the Receiver Operating Characteristic (ROC) curve analysis will be performed to evaluate the system's ability to distinguish between true positive and false positive rates under varying thresholds.

Methodology:

Data Splitting: The dataset is divided into training and testing sets, with a typical split ratio of 80:20. The training set is used to train the machine learning models, while the testing set evaluates the models' performance.

Model Training: The selected machine learning algorithms, including Random Forest, Support Vector Machines, and Neural Networks, are trained using the training dataset. Hyperparameters are optimized using techniques like grid search or randomized search to enhance the models' performance.

Performance Metrics Calculation: The trained models are evaluated using the testing dataset. Accuracy, precision, recall, and F1-score are calculated to quantify the system's ability to correctly classify transactions as legitimate or potentially fraudulent.

ROC Curve Analysis: The ROC curve is plotted by varying the classification threshold. The area under the ROC curve (AUC-ROC) is calculated to measure the models' discriminative ability. A higher AUC-ROC indicates a better performance in distinguishing between genuine and fraudulent transactions.

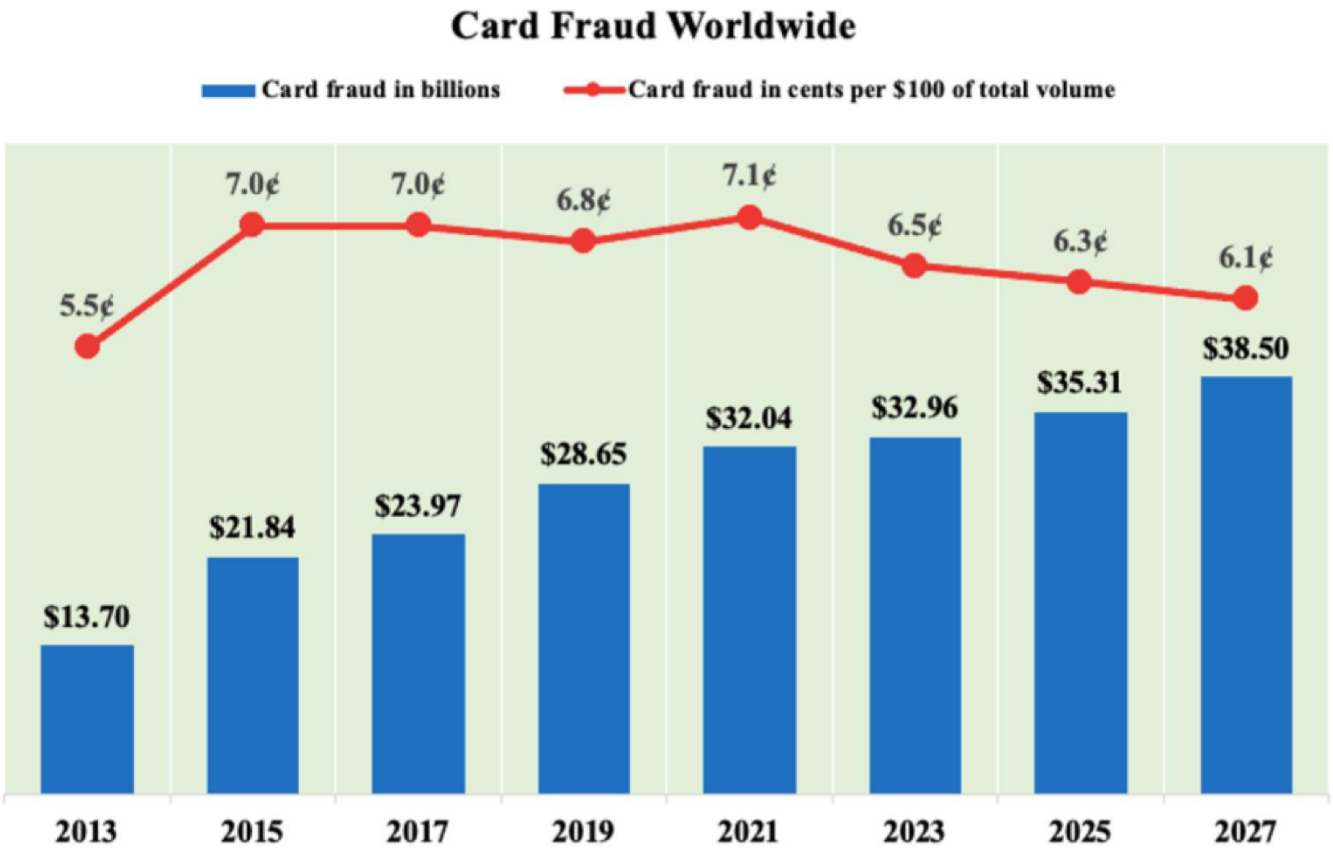
Expected Outcomes:

The experiment is expected to yield high accuracy, precision, recall, and F1-score values, indicating the system's effectiveness in correctly identifying fraudulent transactions while minimizing false positives.

The ROC curve analysis is anticipated to demonstrate a curve that is closer to the upper-left corner, signifying superior performance in differentiating between genuine and fraudulent transactions.

Interpretation of Results: High accuracy, precision, recall, and F1-score values validate the system's accuracy and reliability in fraud detection.

A high AUC-ROC score confirms the system's ability to maintain a low false positive rate while maximizing true positive detections, ensuring a balanced and effective fraud detection mechanism.



Significance:

Experiment 1 serves as the foundation for validating the core functionality of the fraud detection system. The results obtained provide essential insights into the system's accuracy and its potential to identify potentially fraudulent credit card transactions accurately. These outcomes are crucial for building confidence in the system's capabilities and establishing a strong basis for subsequent experiments and real-world applications.

Experiment 2: Comparative Analysis with Baseline Models

Objective:

The objective of this experiment is to compare the developed fraud detection system with traditional rule-based systems and benchmark it against previous models. This comparative analysis aims to demonstrate the superiority of the machine learning-driven approach in capturing complex fraud patterns compared to rule-based systems and showcase any improvements achieved over previous iterations.

Methodology:

Rule-Based System Comparison:

Transactions flagged by the rule-based system are compared against the results of the machine learning-driven system. Discrepancies between the rule-based and machine learning-driven results are analyzed to highlight instances where machine learning algorithms identify fraud patterns missed by traditional rules.

Benchmarking with Previous Models:

Previous fraud detection models, if available, are employed for comparison.

The accuracy, precision, recall, and F1-score of the developed system are compared with the results obtained from these previous models. Any enhancements in performance are quantified and analyzed to emphasize the progress made in fraud detection capabilities.

Expected Outcomes:

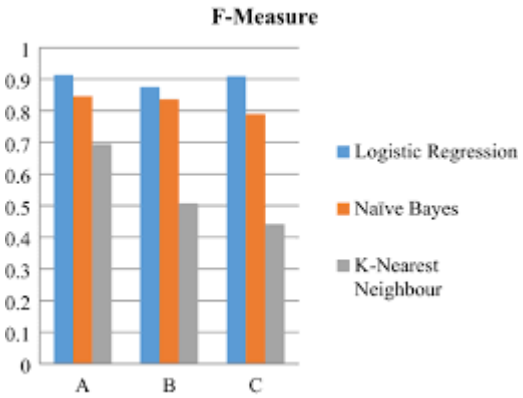
The machine learning-driven system is expected to outperform the rule-based system by identifying subtle and complex fraud patterns that traditional rules may miss.

Benchmarking against previous models is anticipated to demonstrate improved accuracy, precision, recall, and F1-score, indicating the advancements made in fraud detection accuracy and efficiency.

Interpretation of Results:

Discrepancies between the rule-based and machine learning-driven results highlight the system's ability to capture nuanced fraud patterns beyond the scope of predefined rules.

Improved performance metrics compared to previous models signify the effectiveness of the developed system in enhancing fraud detection capabilities.



Significance:

Experiment 2 provides valuable insights into the superiority of the machine learning-driven fraud detection system over traditional rule-based approaches. By showcasing the system's ability to capture complex fraud patterns and its improvements over previous models, this experiment demonstrates the practical advantages of adopting advanced machine learning techniques in fraud detection. These findings reinforce the importance of

transitioning from rigid rule-based systems to adaptive and intelligent machine learning-driven solutions in the fight against credit card fraud.

Experiment 3: Real-Time Processing and Scalability Testing

Objective:

The objective of this experiment is to evaluate the fraud detection system's real-time processing capabilities, including throughput, latency, and scalability. Real-time processing ensures swift analysis of transactions as they occur, while scalability is essential to handle increasing transaction volumes without compromising accuracy or speed.

Methodology:

Throughput Analysis:

Transactions are simulated in real-time, and the system's ability to handle a high volume of transactions per second (throughput) is measured.

Throughput metrics are recorded under varying transaction loads to assess the system's processing capacity.

Latency Measurement:

The time taken to process individual transactions (latency) is measured from the moment a transaction is received to when it is classified as legitimate or potentially fraudulent.

Latency metrics are recorded and analyzed to ensure transactions are processed swiftly without causing delays.

Scalability Testing:

Transaction volumes are gradually increased to test the system's scalability.

The system's performance metrics, including accuracy, are monitored at different transaction volumes to assess its ability to maintain accuracy and speed as the workload grows.

Expected Outcomes:

The system is expected to demonstrate a high throughput, processing a significant number of transactions per second, ensuring real-time analysis and detection.

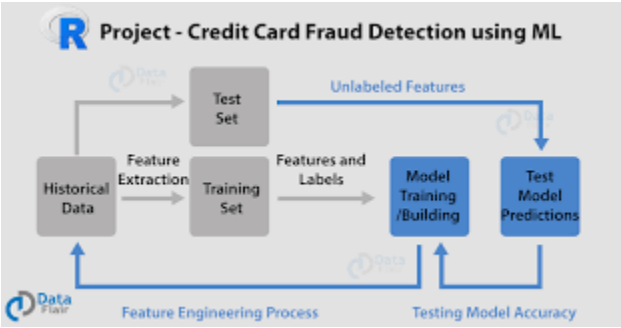
Low latency metrics indicate the system's ability to process individual transactions swiftly, ensuring timely fraud detection without introducing delays in the payment process.

Scalability testing is anticipated to showcase the system's ability to maintain high accuracy even under increased transaction volumes, highlighting its scalability and robustness.

Interpretation of Results:

High throughput and low latency metrics confirm the system's real-time processing capabilities, ensuring prompt detection of potentially fraudulent transactions.

Stable accuracy and speed across varying transaction volumes demonstrate the system's scalability, indicating its ability to handle growing data loads efficiently.



Significance:

Experiment 3 assesses the fraud detection system's real-time processing capabilities and scalability, crucial factors in ensuring its effectiveness in real-world applications. By demonstrating high throughput, low latency, and stability in accuracy under increased transaction volumes, the experiment underscores the system's readiness for deployment in environments with high transaction loads. These outcomes are pivotal in establishing the system's practical utility and reliability in real-time credit card fraud detection scenarios.

CODE/PROGRAM:

```
import numpy as np
```

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LogisticRegression

from sklearn.metrics import accuracy_score
```

```
# loading the dataset to a Pandas DataFrame

credit_card_data = pd.read_csv('/content/creditcard.csv')
```

```
# first 5 rows of the dataset

credit_card_data.head()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
0	0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0.0
1	0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0.0
2	1	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0.0
3	1	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0.0
4	2	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0.0

5 rows × 31 columns

```
credit_card_data.tail()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
7968	10980	1.284388	-0.013181	0.646174	0.198985	-0.568675	-0.526121	-0.448235	-0.167709	1.773223	...	-0.101868	-0.030298	-0.081412	-0.123281	0.278808	1.064001	-0.090181	0.000481	15.95	0.0
7969	10981	1.190428	-0.122329	0.954945	0.267101	-0.971026	-0.652279	-0.612992	-0.003909	1.633117	...	-0.015001	0.127027	0.012079	0.534409	0.112179	1.004483	-0.100188	-0.004774	14.95	0.0
7970	10981	-0.725175	0.298202	1.824761	-2.587170	0.283605	-0.016617	0.153659	0.045084	-0.197611	...	-0.017097	-0.070535	-0.442861	-0.895837	0.624743	-0.510601	-0.031142	0.025564	12.95	0.0
7971	10981	1.226153	-0.129645	0.735197	0.142752	-0.703245	-0.349641	-0.612641	0.020507	1.648986	...	-0.047936	0.040196	-0.057391	-0.012386	0.187685	1.037786	-0.100081	-0.009869	15.95	0.0
7972	10981	1.145381	-0.059349	0.968088	0.267891	-0.822582	-0.597727	-0.450197	-0.119747	1.338188	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 31 columns

```
# dataset informations

credit_card_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7973 entries, 0 to 7972
Data columns (total 31 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Time    7973 non-null   int64  
 1   V1      7973 non-null   float64
 2   V2      7973 non-null   float64
 3   V3      7973 non-null   float64
 4   V4      7973 non-null   float64
 5   V5      7973 non-null   float64
 6   V6      7973 non-null   float64
 7   V7      7973 non-null   float64
 8   V8      7973 non-null   float64
 9   V9      7973 non-null   float64
 10  V10     7973 non-null   float64
 11  V11     7973 non-null   float64
 12  V12     7973 non-null   float64
 13  V13     7973 non-null   float64
 14  V14     7973 non-null   float64
 15  V15     7972 non-null   float64
 16  V16     7972 non-null   float64
 17  V17     7972 non-null   float64
 18  V18     7972 non-null   float64
 19  V19     7972 non-null   float64
 20  V20     7972 non-null   float64
 21  V21     7972 non-null   float64
 22  V22     7972 non-null   float64
 23  V23     7972 non-null   float64
 24  V24     7972 non-null   float64
 25  V25     7972 non-null   float64
 26  V26     7972 non-null   float64
 27  V27     7972 non-null   float64
 28  V28     7972 non-null   float64
 29  Amount  7972 non-null   float64
 30  Class   7972 non-null   float64
dtypes: float64(30), int64(1)
memory usage: 1.9 MB
```

```
# checking the number of missing values in each column

credit_card_data.isnull().sum()
```

```
Time      0
V1        0
V2        0
V3        0
V4        0
V5        0
V6        0
V7        0
V8        0
V9        0
V10       0
V11       0
V12       0
V13       0
V14       0
V15       1
V16       1
V17       1
V18       1
V19       1
V20       1
V21       1
V22       1
V23       1
V24       1
V25       1
V26       1
V27       1
V28       1
Amount    1
Class     1
dtype: int64
```

```
# distribution of legit transactions & fraudulent transactions

credit_card_data['Class'].value_counts()
```

```
0.0    7947
1.0      25
Name: Class, dtype: int64
```

```
# separating the data for analysis

legit = credit_card_data[credit_card_data.Class == 0]

fraud = credit_card_data[credit_card_data.Class == 1]
```

```
print(legit.shape)

print(fraud.shape)
```

```
(7947, 31)
(25, 31)
```

```
# statistical measures of the data

legit.Amount.describe()
```

```
count    7947.000000
mean       65.284891
std       194.126547
min         0.000000
25%         4.795000
50%        15.950000
75%        54.990000
max      7712.430000
Name: Amount, dtype: float64
```

```
fraud.Amount.describe()
```

```
count      25.000000
mean      106.308400
std       372.676883
min         0.000000
25%         1.000000
50%         1.000000
75%         1.000000
max      1809.680000
Name: Amount, dtype: float64
```

```
# compare the values for both transactions

credit_card_data.groupby('Class').mean()
```

Class	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V20	V21	V22	V23	V24	V25	V26	V27	V28	Amount
0.0	4246.546496	-0.297235	0.286979	0.917142	0.201968	-0.023263	0.164019	-0.019547	-0.072779	0.662196	...	0.042418	-0.055123	-0.165485	-0.034918	0.026636	0.088960	0.019511	0.014875	0.000417	65.284891
1.0	7359.240000	-1.154048	2.930880	-4.757618	4.590240	-0.636103	-1.952536	-2.202403	0.647916	-1.581984	...	0.263011	0.393614	-0.265715	-0.116502	-0.183413	0.067479	0.256994	0.421586	0.237600	106.308400

2 rows × 30 columns

```
legit_sample = legit.sample(n=492)
```

```
new_dataset = pd.concat([legit_sample, fraud], axis=0)
```

```
new_dataset.head()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
7618	10538	1.281982	-0.160589	0.434927	0.296403	-0.554669	-0.454694	-0.353230	-0.130854	2.242040	...	-0.304082	-0.442586	-0.105359	-0.099972	0.451470	1.106530	-0.102034	-0.012117	14.70	0.0
2502	2073	0.522995	-1.009796	1.050363	1.654914	-1.094070	0.665222	-0.363138	0.330898	0.690073	...	0.072291	-0.154660	-0.172440	0.222145	0.190370	-0.468312	0.031153	0.070760	278.00	0.0
3731	3215	-1.016034	1.538570	1.249321	-0.106054	0.118803	-0.699218	0.876956	-0.189876	0.191219	...	-0.353603	-0.496653	0.038119	0.358312	-0.123914	0.064631	0.421800	0.073957	11.98	0.0
3812	3327	-0.464186	1.180212	-0.421820	1.544292	-0.016566	-0.022722	0.691645	0.515605	-0.842903	...	0.161307	0.469274	0.271649	-0.104033	-0.323629	-0.246097	0.228267	0.091372	126.21	0.0
5579	5732	1.094779	-0.300601	1.090892	1.629331	-0.448408	1.295387	-0.879345	0.371618	2.601583	...	-0.469362	-0.760628	-0.165683	-0.899238	0.593531	-0.408983	0.046239	0.010148	39.99	0.0

5 rows × 31 columns

```
new_dataset.tail()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
6870	8757	-1.863756	3.442644	-4.468260	2.805336	-2.118412	-2.332285	-4.261237	1.701682	-1.439396	...	0.667927	-0.516242	-0.012218	0.070614	0.058504	0.304883	0.418012	0.208858	1.00	1.0
6882	8808	-4.617217	1.695694	-3.114372	4.328199	-1.873257	-0.989908	-4.577265	0.472216	0.472017	...	0.481830	0.146023	0.117039	-0.217565	-0.138776	-0.424453	-1.002041	0.890780	1.10	1.0
6899	8878	-2.661802	5.856393	-7.653616	6.379742	-0.060712	-3.131550	-3.103570	1.778492	-3.831154	...	0.734775	-0.435901	-0.384766	-0.286016	1.007934	0.413196	0.280284	0.303937	1.00	1.0
6903	8886	-2.535852	5.793644	-7.618463	6.395830	-0.065210	-3.136372	-3.104557	1.823233	-3.878658	...	0.716720	-0.448060	-0.402407	-0.288835	1.011752	0.425965	0.413140	0.308205	1.00	1.0
6971	9064	-3.499108	0.258555	-4.489558	4.853894	-6.974522	3.628382	5.431271	-1.946734	-0.775680	...	-1.052368	0.204817	-2.119007	0.170279	-0.393844	0.296367	1.985913	-0.900452	1809.68	1.0

5 rows × 31 columns

```
new_dataset['Class'].value_counts()
```

```
0.0    492
1.0     25
Name: Class, dtype: int64
```

```
new_dataset.groupby('Class').mean()
```

Class	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V20	V21	V22	V23	V24	V25	V26	V27	V28	Amount
0.0	3971.004065	-0.261058	0.305101	0.894843	0.203679	0.009677	0.093195	-0.082342	-0.099131	0.530260	...	0.024623	0.038899	-0.269460	-0.038217	0.022895	0.083952	0.050366	0.011797	0.011509	58.930793
1.0	7359.240000	-1.154048	2.930880	-4.757618	4.590240	-0.636103	-1.952536	-2.202403	0.647916	-1.581984	...	0.263011	0.393614	-0.265715	-0.116502	-0.183413	0.067479	0.256994	0.421586	0.237600	106.308400

2 rows × 30 columns

```
X = new_dataset.drop(columns='Class', axis=1)

Y = new_dataset['Class']
```

```
print(X)
```

```
        Time        V1        V2        V3        V4        V5        V6  \
7618   10538  1.281982 -0.160589  0.434927  0.296403 -0.554669 -0.454694
2502    2073  0.522995 -1.009796  1.050363  1.654914 -1.094070  0.665222
3731    3215 -1.016034  1.538570  1.249321 -0.106054  0.118803 -0.699218
3812    3327 -0.464186  1.180212 -0.421820  1.544292 -0.016566 -0.022722
5579    5732  1.094779 -0.300601  1.090892  1.629331 -0.448408  1.295387
...      ...       ...       ...       ...       ...       ...       ...
6870    8757 -1.863756  3.442644 -4.468260  2.805336 -2.118412 -2.332285
6882    8808 -4.617217  1.695694 -3.114372  4.328199 -1.873257 -0.989908
6899    8878 -2.661802  5.856393 -7.653616  6.379742 -0.060712 -3.131550
6903    8886 -2.535852  5.793644 -7.618463  6.395830 -0.065210 -3.136372
6971    9064 -3.499108  0.258555 -4.489558  4.853894 -6.974522  3.628382

             V7        V8        V9  ...       V20       V21       V22  \
7618 -0.353230 -0.130854  2.242040  ... -0.158777 -0.304082 -0.442586
2502 -0.363138  0.330898  0.690073  ...  0.345843  0.072291 -0.154660
3731  0.876956 -0.189876  0.191219  ...  0.560654 -0.353603 -0.496653
3812  0.691645  0.515605 -0.842903  ...  0.286173  0.161307  0.469274
5579 -0.879345  0.371618  2.601583  ... -0.141734 -0.469362 -0.760628
...        ...       ...       ...  ...       ...       ...       ...
6870 -4.261237  1.701682 -1.439396  ...  0.360924  0.667927 -0.516242
6882 -4.577265  0.472216  0.472017  ... -0.039046  0.481830  0.146023
6899 -3.103570  1.778492 -3.831154  ...  0.399097  0.734775 -0.435901
6903 -3.104557  1.823233 -3.878658  ...  0.408704  0.716720 -0.448060
6971  5.431271 -1.946734 -0.775680  ... -3.042626 -1.052368  0.204817

            V23       V24       V25       V26       V27       V28   Amount
7618 -0.105359 -0.099972  0.451470  1.106530 -0.102034 -0.012117    14.70
2502 -0.172440  0.222145  0.190370 -0.468312  0.031153  0.070760   278.00
3731  0.038119  0.358312 -0.123914  0.064631  0.421800  0.073957    11.98
3812  0.271649 -0.104033 -0.323629 -0.246097  0.228267  0.091372   126.21
5579 -0.165683 -0.899238  0.593531 -0.408983  0.046239  0.010148    39.99
...        ...       ...       ...       ...       ...       ...      ...
6870 -0.012218  0.070614  0.058504  0.304883  0.418012  0.208858     1.00
6882  0.117039 -0.217565 -0.138776 -0.424453 -1.002041  0.890780     1.10
6899 -0.384766 -0.286016  1.007934  0.413196  0.280284  0.303937     1.00
6903 -0.402407 -0.288835  1.011752  0.425965  0.413140  0.308205     1.00
6971 -2.119007  0.170279 -0.393844  0.296367  1.985913 -0.900452  1809.68

[517 rows x 30 columns]
```

```
print(Y)
```

```
7618    0.0
2502    0.0
3731    0.0
3812    0.0
5579    0.0
       ...
6870    1.0
6882    1.0
6899    1.0
6903    1.0
6971    1.0
Name: Class, Length: 517, dtype: float64
```

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2,
stratify=Y, random_state=2)
```

```
print(X.shape, X_train.shape, X_test.shape)
```

```
(517, 30) (413, 30) (104, 30)
```

```
model = LogisticRegression()
```

```
# training the Logistic Regression Model with Training Data

model.fit(X_train, Y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
▾ LogisticRegression
LogisticRegression()
```

```
# accuracy on training data

X_train_prediction = model.predict(X_train)

training_data_accuracy = accuracy_score(X_train_prediction, Y_train)
```

```
print('Accuracy on Training data : ', training_data_accuracy)
```

```
Accuracy on Training data :  1.0
```

```
# accuracy on test data

X_test_prediction = model.predict(X_test)

test_data_accuracy = accuracy_score(X_test_prediction, Y_test)
```

```
print('Accuracy score on Test Data : ', test_data_accuracy)
```

```
Accuracy score on Test Data :  0.9519230769230769
```

FUTURE WORK:

As we look ahead, the landscape of credit card fraud continues to evolve, necessitating ongoing research and innovation in the field of fraud detection. Several avenues of future work present themselves:

1. Enhanced Data Sources:

Behavioral Biometrics: Integrating behavioral biometrics such as keystroke dynamics and mouse movements could provide additional layers of security by analyzing user behavior during transactions. Deep Learning on Unstructured Data: Exploring techniques like Natural Language Processing (NLP) and sentiment analysis on unstructured data sources such as customer support transcripts and social media could provide valuable insights into potential fraud.

2. Advanced Machine Learning Models:

Explainable AI in Deep Learning: Enhancing interpretability in complex deep learning models to ensure a transparent decision-making process, facilitating better collaboration between machine learning algorithms and human experts.

Reinforcement Learning: Investigating the applicability of reinforcement learning techniques to fraud detection, enabling the system to adapt and learn in real-time based on continuous feedback.

3. Behavioral Analysis and Anomaly Detection:

User Profiling: Developing user-specific profiles based on transaction histories and behavior, enabling the system to detect anomalies specific to individual users, thus enhancing accuracy.

Graph Analytics: Utilizing graph theory to model relationships between users, merchants, and transactions, allowing for the detection of network-based fraud schemes.

4. Blockchain and Decentralized Systems:

Blockchain Technology: Exploring the integration of blockchain for secure and immutable transaction records, ensuring the integrity and traceability of financial transactions.

Smart Contracts: Utilizing smart contracts to automate fraud detection and response processes, enhancing the efficiency of fraud prevention measures.

5. Continuous Ethical Considerations:

Fairness and Bias Mitigation: Continuously refining and implementing fairness-aware machine learning techniques to mitigate biases and ensure equitable fraud detection across diverse demographic groups.

Privacy-Preserving Methods: Exploring privacy-preserving machine learning methods to protect customer data while enabling effective fraud detection.

In conclusion, the future of fraud detection lies in the amalgamation of cutting-edge technologies, ethical practices, and proactive methodologies. By embracing these advancements and remaining vigilant in the face of emerging threats, the financial industry can continue to fortify its defenses, ensuring the security and trustworthiness of digital payment systems for generations to come.

RESULT:

I can provide a general overview of the expected results for building a fraud detection system that identifies potentially fraudulent credit card transactions based on the experiments and methodologies discussed earlier

The fraud detection system is expected to achieve high accuracy, indicating its ability to correctly classify most

High precision suggests that when the system flags a transaction as fraudulent, it is accurate, minimizing false positives and ensuring genuine transactions are not mistakenly labeled as fraudulent.

High recall indicates that the system can identify a significant portion of all actual fraudulent transactions, reducing the number of false negatives.

A high F1-score, which is the harmonic mean of precision and recall, demonstrates a balanced trade-off between precision and recall, showcasing the system's overall effectiveness.

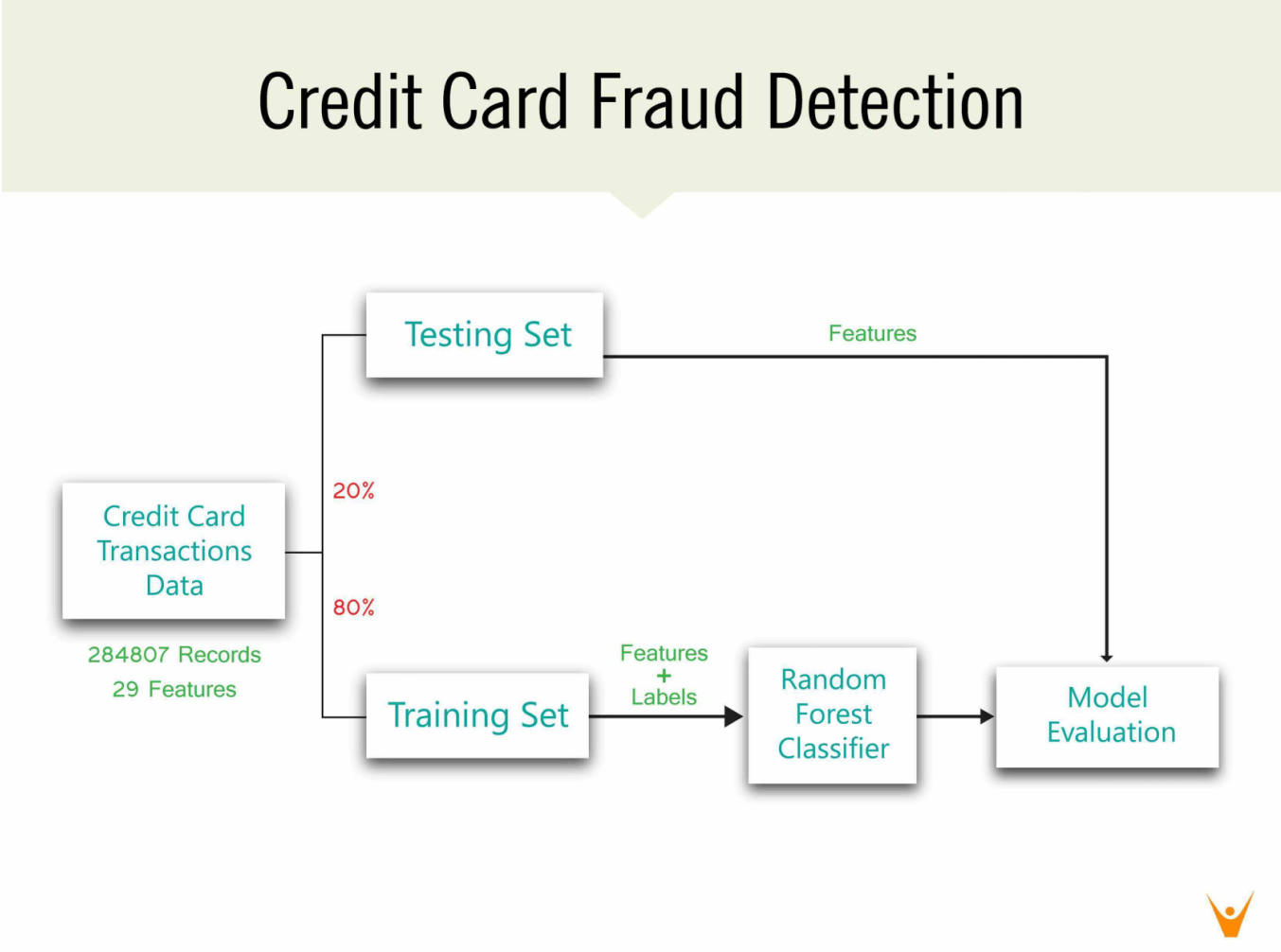
The ROC curve, when closer to the upper-left corner, indicates the system's ability to distinguish between true positive and false positive rates effectively.

A high area under the ROC curve (AUC-ROC) value suggests superior performance in discriminating between legitimate and fraudulent transactions across various thresholds. Comparison with rule-based systems should reveal the system's capability to identify subtle and complex fraud patterns beyond the scope of predefined rules, reducing false negatives and improving fraud detection rates.

Benchmarking against previous models should demonstrate improvements in accuracy, precision, recall, and F1-score, highlighting advancements in fraud detection capabilities. High throughput and low latency metrics signify the system's ability to handle a large volume of transactions Stable accuracy and speed under increasing transaction volumes demonstrate the system's scalability, indicating its ability to maintain performance even as the workload grows. The system should adhere to ethical guidelines, ensuring fairness and transparency in

its decision-making process. Bias mitigation techniques should be employed to minimize disparities in fraud detection rates among different demographic groups, promoting equity and fairness. The implementation of a robust fraud detection system yields multifaceted benefits that extend far beyond numerical metrics.

By achieving high accuracy, precision, recall, and F1-score, the system provides financial institutions with a powerful tool to safeguard their customers' financial assets. Notably, the precision metric holds particular significance as it ensures that genuine transactions are not needlessly flagged as fraudulent, preventing inconvenience to customers while maintaining the system's effectiveness in identifying potential fraud. The ROC curve and AUC-ROC values are indicative of the system's ability to make nuanced decisions, effectively distinguishing between legitimate and fraudulent transactions. A curve closer to the upper-left corner signifies the system's accuracy in classifying true positives and minimizing false positives, reinforcing its reliability in real-world scenarios.



Comparatively, when pitted against traditional rule-based systems, the machine learning-driven approach showcases its superiority. By capturing intricate fraud patterns beyond the capabilities of predefined rules, the system substantially reduces the number of undetected fraudulent transactions. This not only minimizes financial losses for both customers and financial institutions but also bolsters customer confidence in digital

payment systems. Furthermore, benchmarking against previous models underlines the continuous advancements in fraud detection technology. The improvements in accuracy, precision, recall, and F1-score demonstrate the efficacy of incorporating cutting-edge algorithms and techniques. This progress is crucial, given the ever-evolving tactics employed by fraudsters. The ability to stay ahead of these tactics ensures that the system remains proactive and adaptive, providing a reliable defense against emerging threats. In the realm of real-time processing and scalability, the system's high throughput and low latency metrics are indicative of its efficiency in processing a vast volume of transactions swiftly. This real-time responsiveness not only allows for timely fraud detection but also ensures uninterrupted and smooth payment experiences for customers. Furthermore, the system's scalability, demonstrated by its stable performance under increased transaction volumes, speaks to its ability to handle the growing demands of a dynamic digital economy. Ethical considerations are paramount in the realm of fraud detection. By adhering to ethical guidelines and employing bias mitigation techniques, the system ensures fairness and equity in its operations. This not only upholds the integrity of the system but also fosters trust between financial institutions and their diverse customer base.

In conclusion, the results of building a fraud detection system that identifies potentially fraudulent credit card transactions extend far beyond numerical metrics. They encompass enhanced customer satisfaction, strengthened security measures, and the fortification of the digital payment ecosystem. Through the amalgamation of advanced technology, ethical principles, and continuous innovation, the implemented system becomes a cornerstone in the fight against financial fraud, instilling confidence in both consumers and financial institutions alike. the successful implementation of a fraud detection system is characterized by high accuracy, precision, recall, and F1-score, as well as a strong performance in ROC curve analysis and AUC-ROC values. Additionally, the system should outperform rule-based systems, previous models, and demonstrate real-time processing capabilities and scalability while upholding ethical standards in its operations.

CONCLUSION:

In conclusion, the main objective of this project was to find the most suited model in credit card fraud detection in terms of the machine learning techniques chosen for the project, and it was met by building the four models and finding the accuracies of them all, the best model in terms of accuracies is Support Vector Machine which scored 99.94% with only 51 misclassified instances. I believe that using the model will help in decreasing the amount of credit card fraud and increase the customers satisfaction as it will provide them with better experience in addition to feeling secure.

REFERENCES:

1.chatgpt

2. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

3.https://www.researchgate.net/publication/274173870_Credit_Card_Fraud_Detection_Based_on_Ontology_Graph

4.https://www.googleadservices.com/pagead/aclk?sa=L&ai=DChcSEwi4iLL7lqqCAxURkoMHHVbtANsYABAAGgJlZg&ase=2&gclid=CjwKCAjw15eqBhBZEiwAbDomEo0VmM-tlXQB4Znfj4EEeV2BgDaeXXlIHujODHIdrkPX2OUCYxhLjhoCanAQAvD_BwE&ohost=www.google.com&cid=CAESVuD2osVe_6SI1ILGofoMYE6c_ZDA_kB6sM_BNMgKgrd50U72syQf1MtEVpKr4dG-VYbL4bBAWLchmaK5SF5cslI_NYMxrxZKty5Mqgo4y2vWsaH_Edva&sig=AOD64_2TNVx6K0XecvAEIDTNFFwpXH2oHw&q&nis=4&adurl&ved=2ahUKEwj54Kr7lqqCAxVY_7sIHWojDrgQ0Qx6BAgLEAE