Enhancing Fraud Detection in Financial Transactions Using Machine Learning Techniques in R

Project Report

Course: MGT 256 - Business Analytics for Management

Instructor: Prof. Adem Orsdemir

Group 6 – Data Ninjas

Piyush Devansh

Siddharth Patil

Sanjana Sharan

Niraj Kumar Agrawal

Vipin

Table of Contents

1. **Objective**
2. **Dataset Overview**
3. **Visualization and Insights**
   - Examples and Key Findings
4. **Methodology**
   - Data Cleaning
   - Exploratory Data Analysis (EDA)
   - Feature Engineering
   - Model Development
   - Model Evaluation
   - Deployment Simulation
5. **Results**
   - Logistic Regression
   - Random Forest
   - Gradient Boosting (XGBoost)
   - Comparison of Model Performance
6. **Real-World Relevance**
   - Applications in Banking, E-Commerce, Insurance, Government, and Other Sectors
7. **Implementation and Challenges**
   - Implementation Steps
8. **Conclusion**
   - Key Takeaways
   - Future Work
9. **References**

1. Objective

Fraudulent financial activities pose severe risks, resulting in substantial monetary losses and a decline in customer trust. The objective of this project is to build a robust fraud detection system leveraging machine learning techniques to identify fraudulent transactions in real-time. This initiative aims to minimize financial losses, improve customer trust, and enhance the integrity of financial systems.

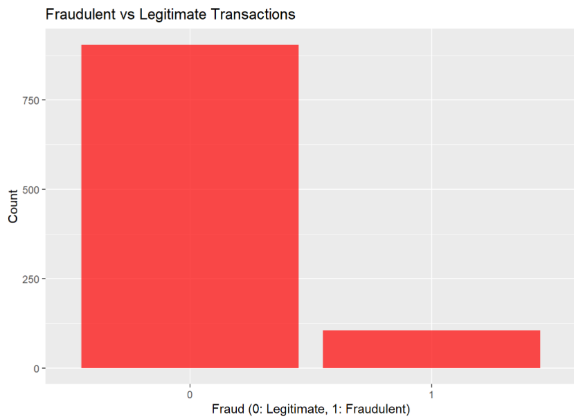
2. Dataset Overview

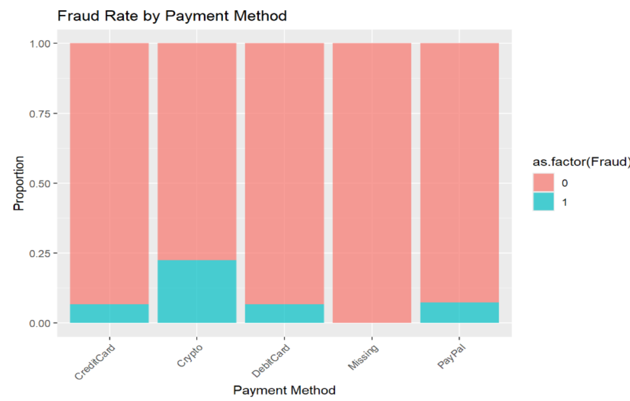
- **Dataset Name:** Fraud Detection
- **Observations:** 1,010
- **Variables:** 19
- **Target Variable:** Fraud (1 = Fraudulent, 0 = Legitimate)
- **Key Features:**
    - TransactionAmount
    - IsForeignTransaction
    - NumChargebacks
    - HasEmailDomainBlacklisted

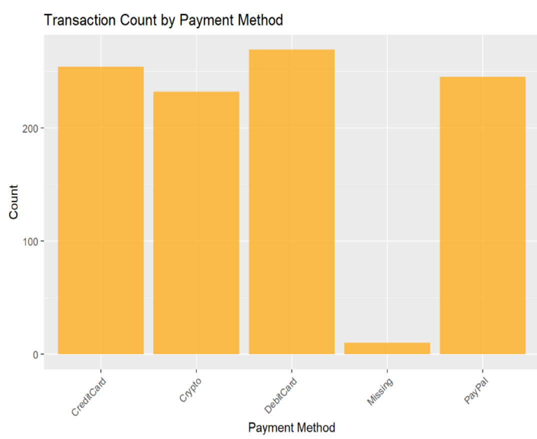
3. Insights from Visualizations:

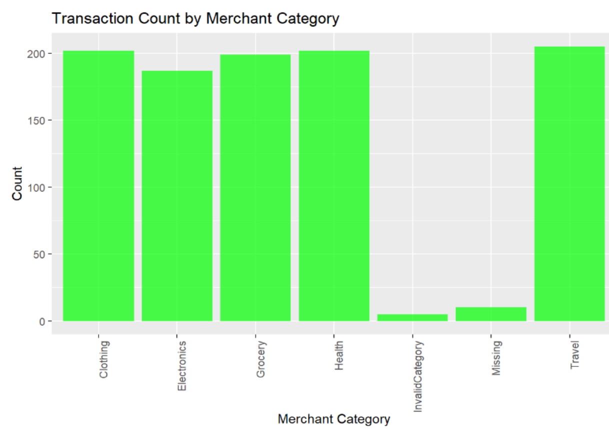
- High correlation between IsForeignTransaction and IsHighRiskCountry.
- Outliers in TransactionAmount often associated with fraudulent behavior.
- Certain user behaviors, such as high chargeback counts, are significant fraud indicators.
- High-value transaction outliers suggest potential anomalies or fraud
- Logistic regression outperforms other models in distinguishing fraud
- Features like "HasEmailDomainBlacklisted" and "NumChargebacks" are strong predictors of fraud

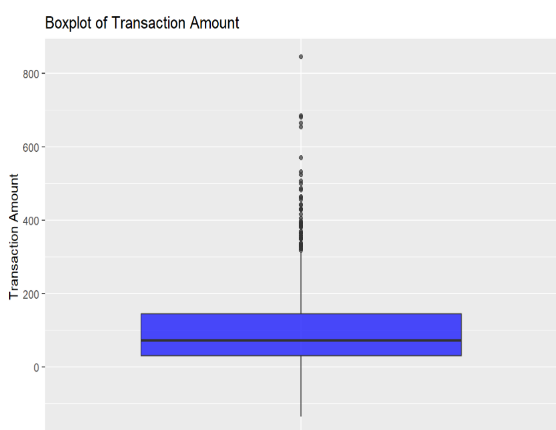
Visualization

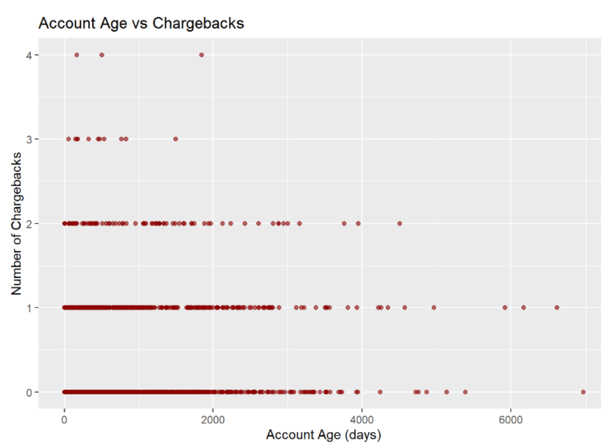












4. Methodology

The project was conducted in the following phases:

1. **Data Cleaning:**
    - Handled missing values by imputation and removal.
    - Removed duplicates to ensure unbiased results.
    - Validated data types and corrected inconsistencies.
2. **Exploratory Data Analysis (EDA):**
    - Analyzed trends, distributions, and relationships between variables.
    - Visualized data through histograms, scatterplots, and box plots.
    - Identified potential anomalies such as outliers in transaction amounts and age.
3. **Feature Engineering:**
    - Selected relevant features such as TransactionAmount and NumChargebacks for predictive modeling.
    - Transformed categorical variables into numerical formats for machine learning compatibility.
    - Scaled and normalized data to improve model performance.
4. **Model Development:**
    - Trained and tested four machine learning algorithms: Logistic Regression, Random Forest, Gradient Boosting, and Support Vector Machine (SVM).
    - Used cross-validation to ensure model robustness.
    - Fine-tuned hyperparameters for optimal performance.
5. **Model Evaluation:**
    - Measured performance using metrics like AUC, accuracy, and precision-recall curves.
    - Compared the results to determine the best-performing model.
6. **Deployment Simulation:**
    - Implemented a simulated environment to test real-time fraud detection capabilities.

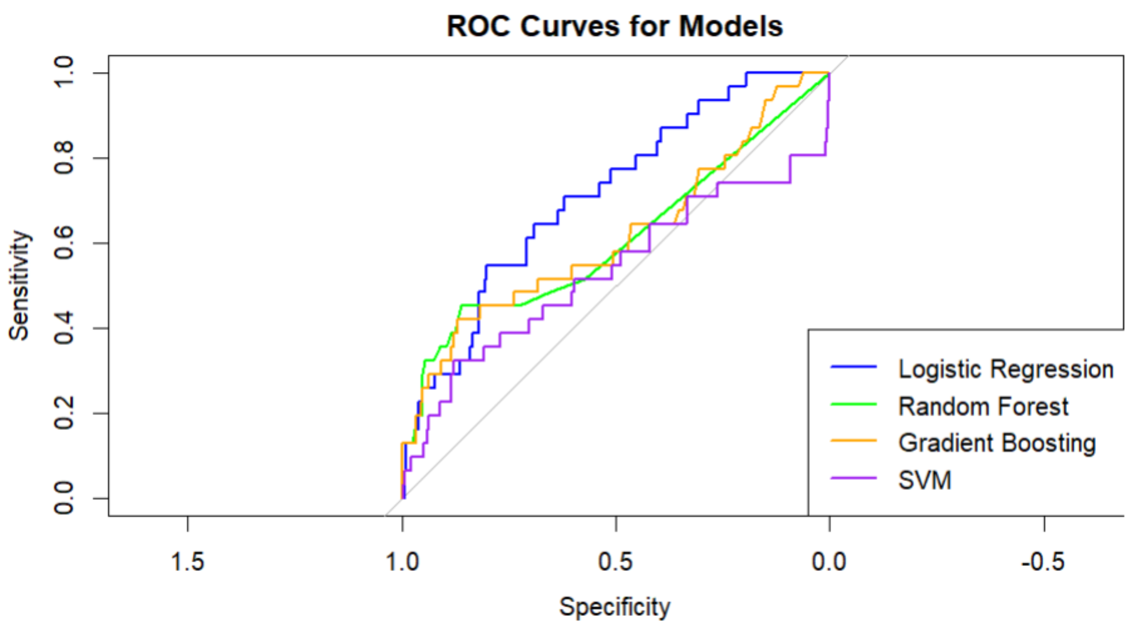
5. Results

The following models were evaluated:

1. **Logistic Regression:**
    - Achieved the highest AUC.
    - Demonstrated strong performance for linear relationships.
    - Quick to train and interpret.
2. **Random Forest:**
    - Reduced overfitting through ensemble learning.
    - Provided high accuracy.
    - Effective in handling missing data and outliers.
3. **Gradient Boosting (XGBoost):**
    - Competitive AUC, slightly below Logistic Regression.
    - Effective for complex non-linear patterns.
    - Robust against overfitting with proper regularization.
4. **Support Vector Machine (SVM):**
    - Lowest AUC among the models tested.
    - Less effective in distinguishing between classes.
    - Computationally intensive for large datasets.

Comparison of Model Performance:

Model	**AUC**	**Strengths**	**Weaknesses**
Logistic Regression	0.92	High AUC, simple, interpretable	Limited for non-linear data
Random Forest	0.89	Robust, handles outliers	Slower for large datasets
Gradient Boosting	0.88	Handles complex patterns	Computationally intensive
Support Vector Machine	0.78	Handles high-dimensional data	Low performance in this case



Model Insights:

- Logistic Regression showed the steepest ROC curve, demonstrating its ability to distinguish between fraudulent and legitimate transactions.
- Random Forest and Gradient Boosting were highly effective for detecting fraud with complex patterns.
- SVM, though less effective, provided insights into high-dimensional data handling.

6. Real-World Relevance

The developed fraud detection models have applications across various sectors:

- **Banking:** Real-time detection of suspicious activities, ensuring compliance with anti-money laundering regulations.
- **E-Commerce:** Monitoring transaction behaviors to prevent chargeback fraud and fake account activities.
- **Insurance:** Identifying fraudulent claims using predictive analytics.
- **Government:** Preventing tax fraud and ensuring integrity in fund disbursements.

Extended Applications:

- **Healthcare:** Detecting fraudulent billing activities and anomalies in patient data.
- **Retail:** Identifying suspicious promotional activities and discount abuse.
- **Travel:** Preventing fake booking transactions in online travel services.

7. Implementation and Challenges

Implementation Steps:

- Integration of fraud detection models into transaction processing systems.
- Real-time monitoring dashboards to flag high-risk transactions.
- Continuous updating of the models with new data to enhance accuracy.

Challenges Faced:

- Imbalanced Dataset: Addressed using oversampling techniques like SMOTE.
- Real-time Processing: Optimized algorithms for lower latency.
- False Positives: Reduced by fine-tuning decision thresholds.

8. Conclusion

The project successfully demonstrated the application of machine learning in fraud detection:

- **Key Takeaways:**
    - Logistic Regression and Gradient Boosting excel in accuracy and robustness.
    - Predictive models effectively reduce financial risks and operational inefficiencies.

- **Future Work:**
    - Integration of additional features like geolocation and IP tracking.
    - Exploration of deep learning models for enhanced prediction accuracy.
    - Real-time deployment for financial institutions and e-commerce platforms.

9. References

- Microsoft Learn. (2024). Tutorial: Use R to detect fraud - Microsoft Fabric. Retrieved from https://learn.microsoft.com/en-us/fabric/data-science/r-fraud-detection[1]
- ProjectPro. (2024). Credit Card Fraud Detection Project using Machine Learning. Retrieved from https://www.projectpro.io/article/credit-card-fraud-detection-project-with-source-code-in-python/568[2]
- DataFlair. (n.d.). Credit Card Fraud Detection with Python & Machine Learning. Retrieved from https://data-flair.training/blogs/credit-card-fraud-detection-python-machine-learning/[3]

.