Fraud Detection Using Blended Modeling and Reinforcement Learning

A Data Science Approach to Identifying Fraudulent Transactions

Donn Bryan Julian

Date: 12/2/2024

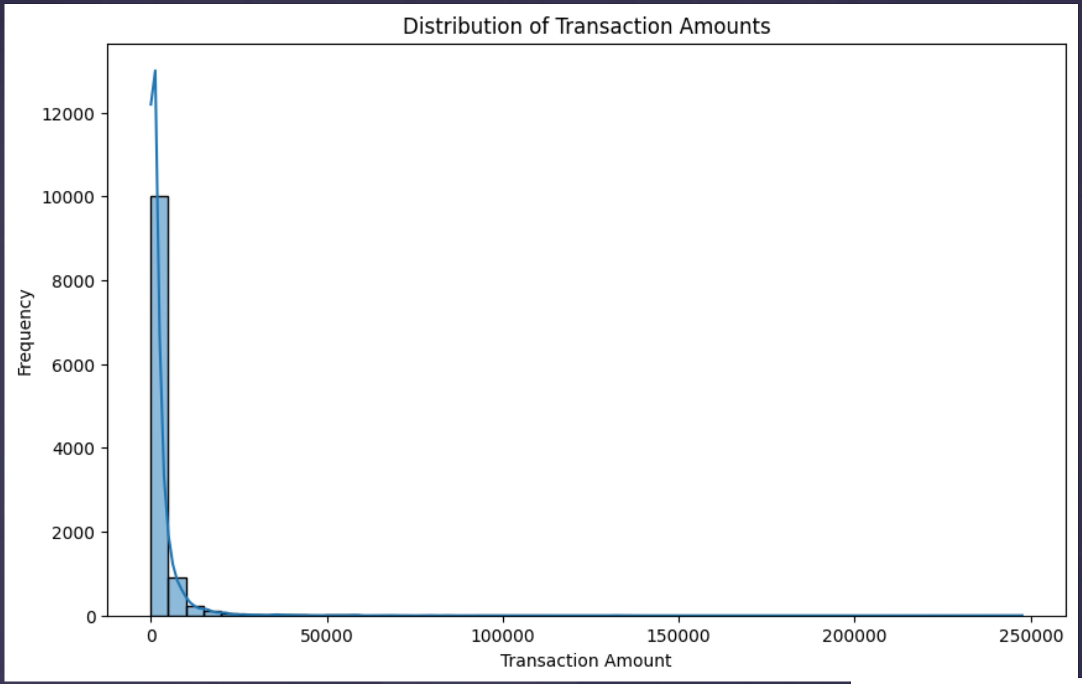
Objective:

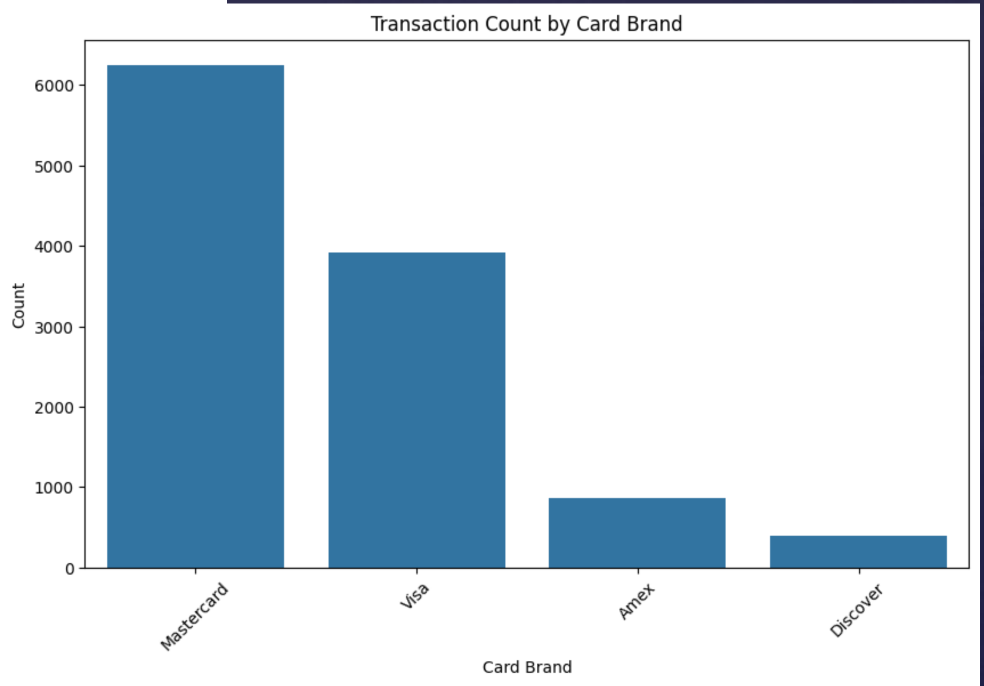
Develop a machine learning model to detect fraudulent transactions effectively.

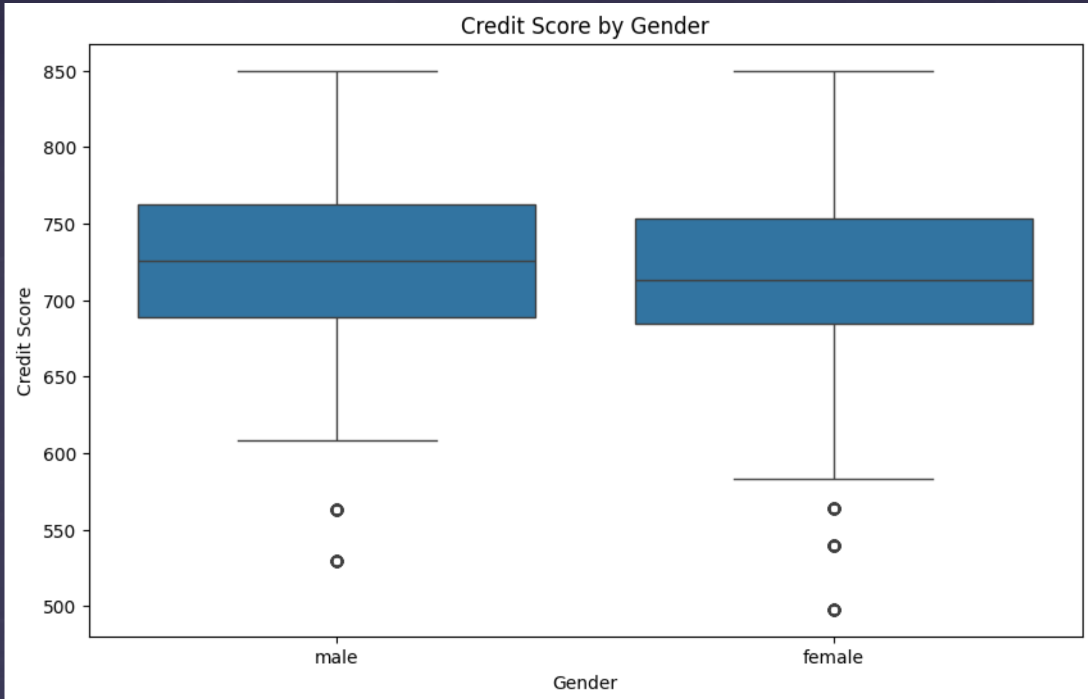
Key Takeaway:

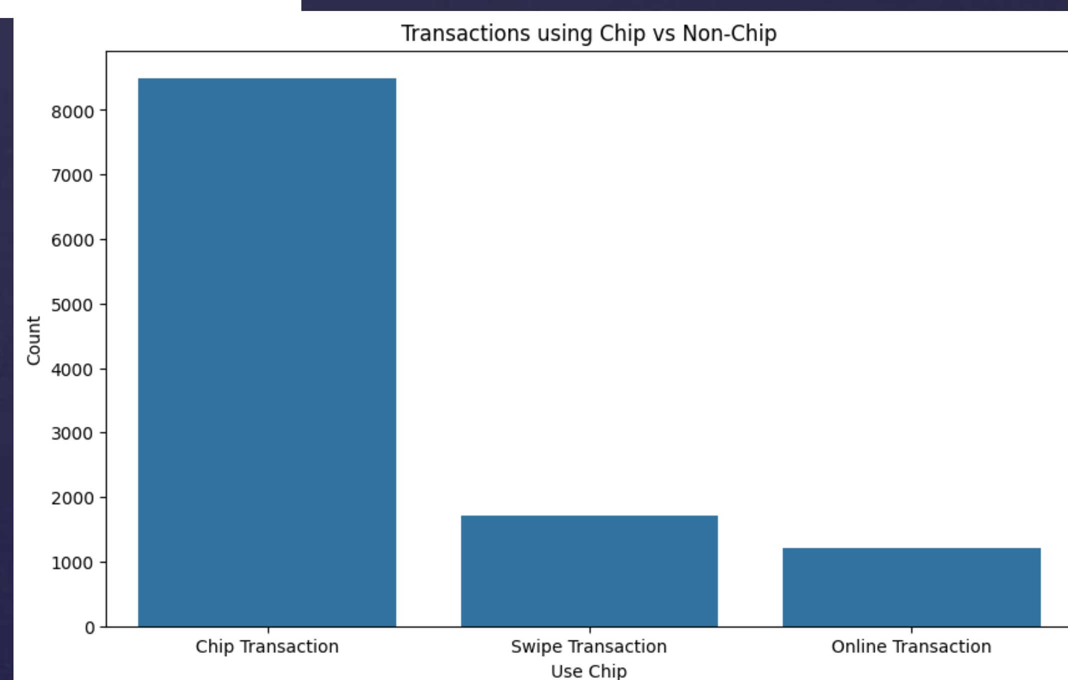
A blended model, combining ensemble learning and reinforcement learning, achieves high accuracy with enhanced feature understanding.

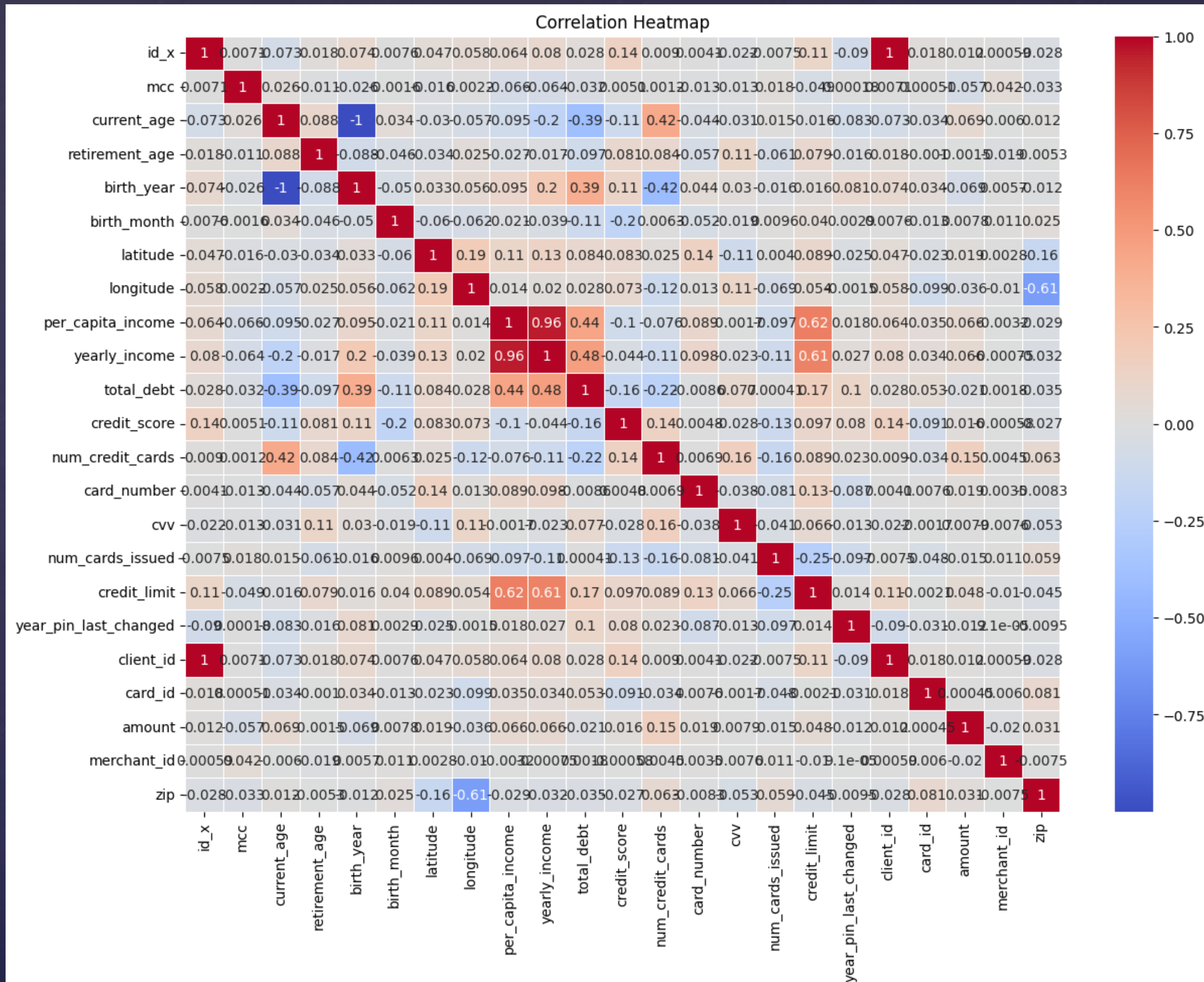
Executive Summary

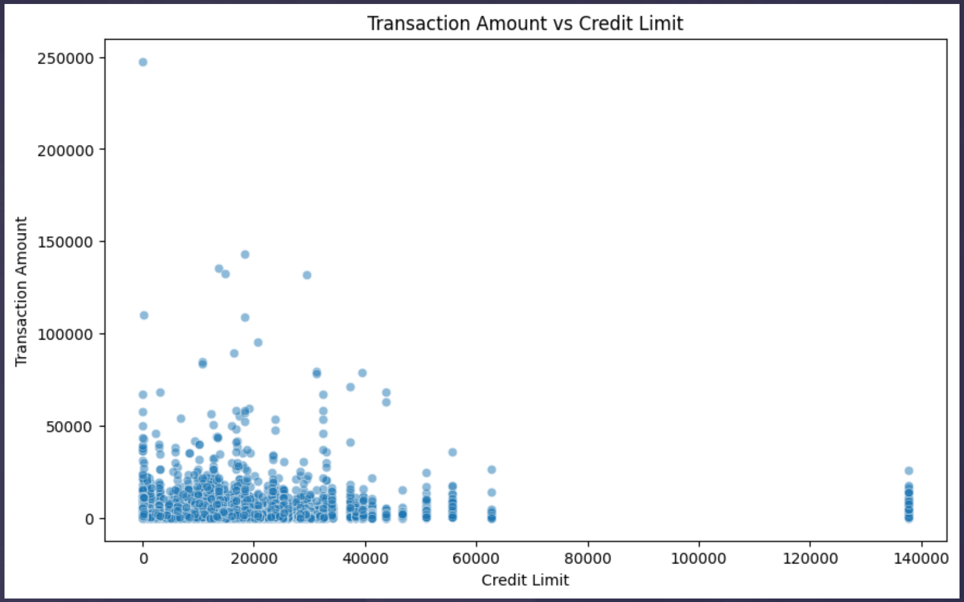

Distribution of Transaction Amounts
12000
10000
8000
6000
4000
2000
0
Frequency
0
50000
100000
150000
200000
250000
Transaction Amount

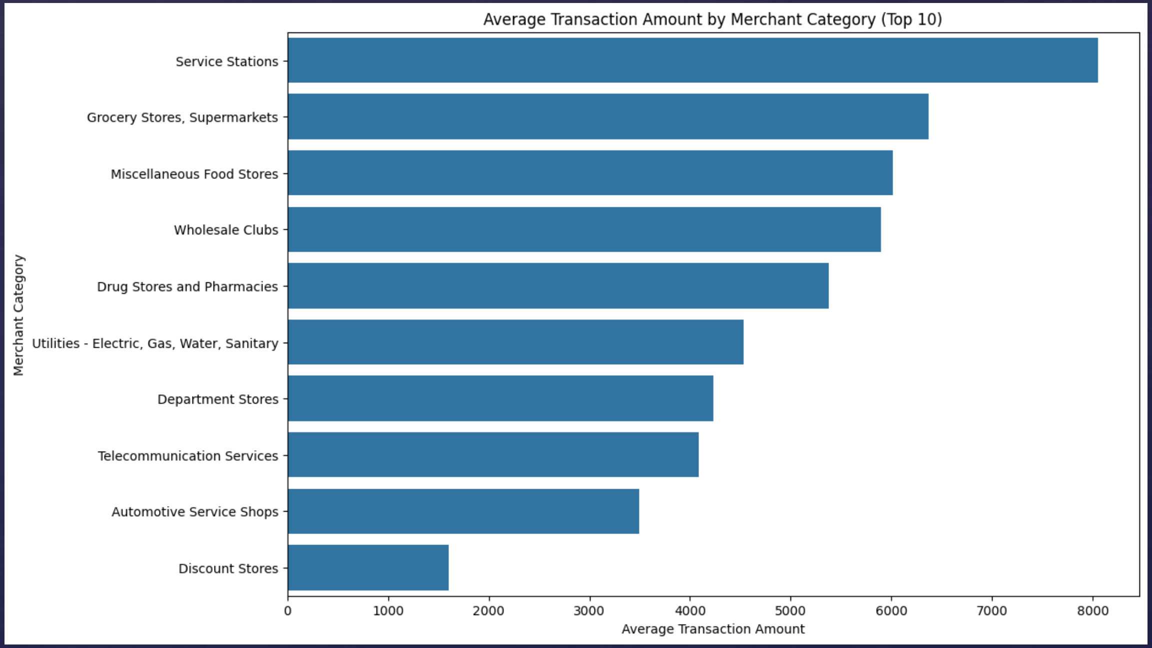

Transaction Count by Card Brand
6000
5000
4000
3000
2000
1000
0
Count
Mastercard
Visa
Amex
Discover
Card Brand


Credit Score by Gender
850
800
750
700
650
600
550
500
Credit Score
male
female
Gender


Transactions using Chip vs Non-Chip
8000
7000
6000
5000
4000
3000
2000
1000
0
Count
Chip Transaction
Swipe Transaction
Online Transaction
Use Chip


Correlation Heatmap
id_x
mcc
current_age
retirement_age
birth_year
birth_month
latitude
longitude
per_capita_income
yearly_income
total_debt
credit_score
num_credit_cards
card_number
cvv
num_cards_issued
credit_limit
year_pin_last_changed
client_id
card_id
amount
merchant_id
zip
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75


Transaction Amount vs Credit Limit
Transaction Amount
250000
200000
150000
100000
50000
0
0
20000
40000
60000
80000
100000
120000
140000
Credit Limit


Average Transaction Amount by Merchant Category (Top 10)
Merchant Category
Service Stations
Grocery Stores, Supermarkets
Miscellaneous Food Stores
Wholesale Clubs
Drug Stores and Pharmacies
Utilities - Electric, Gas, Water, Sanitary
Department Stores
Telecommunication Services
Automotive Service Shops
Discount Stores
0
1000
2000
3000
4000
5000
6000
7000
8000
Average Transaction Amount

Context: Financial fraud poses significant financial risks.

Goal: Enhance the identification of fraudulent transactions with minimal false positives, ensuring business security and customer trust.

Impact: Reduced fraud-related losses and improved customer experience.

Business Problem

Modeling Approach:

- Blended model with ensemble methods (Naive Bayes, Support Vector Classifier, Logistic Regression).
- Reinforcement Learning for fine-tuning model performance.

Feature Engineering: Focused on identifying key features like total debt, transaction amount, and credit limit.

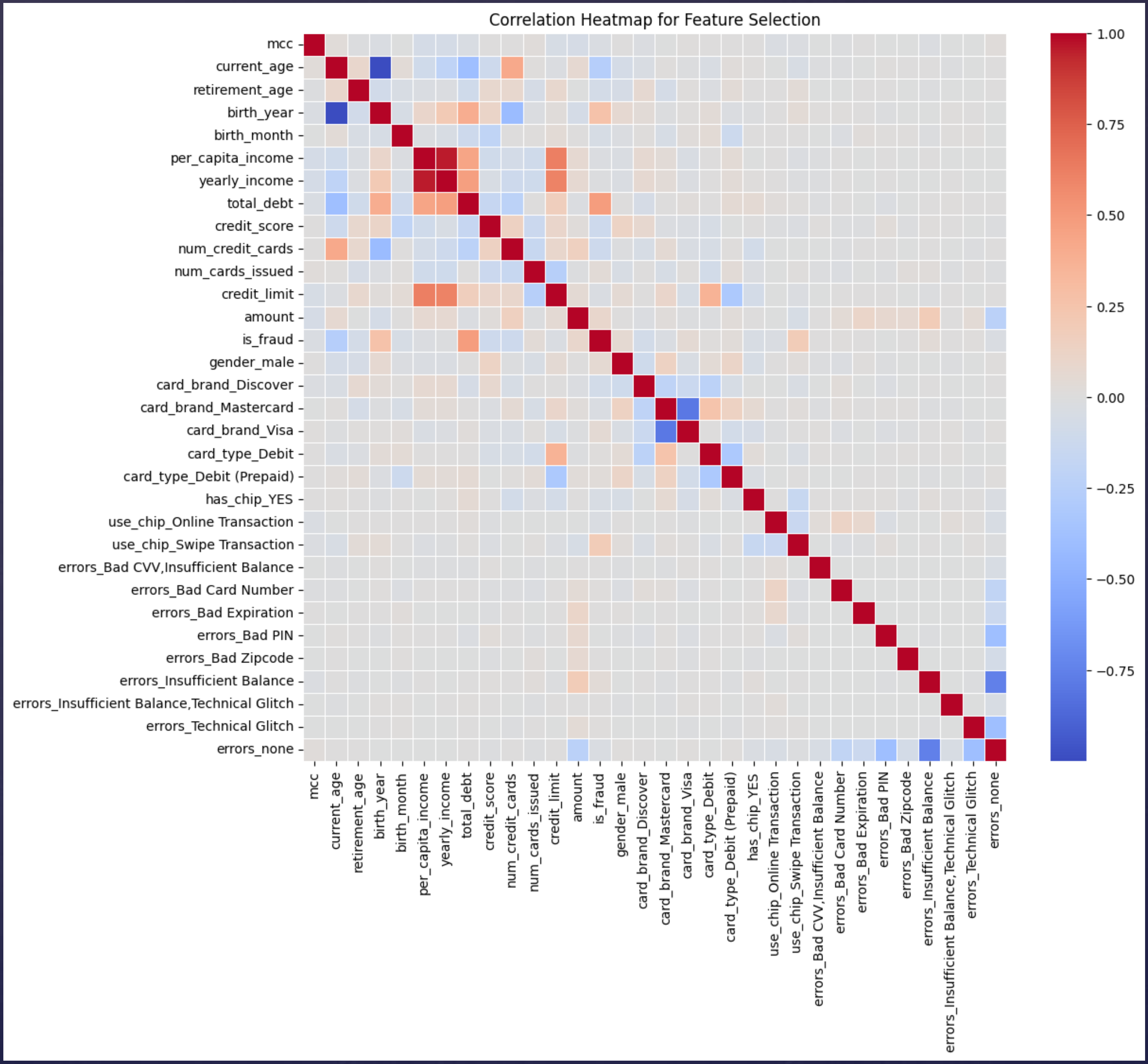
Methodology Overview

Ensemble Approach: Combined models to leverage diverse strengths.

Reinforcement Learning: Applied Deep Q-Learning to adaptively adjust model weights for optimal performance.

Regularization: Addressed overfitting by tuning hyperparameters and increasing model robustness.

Model Development



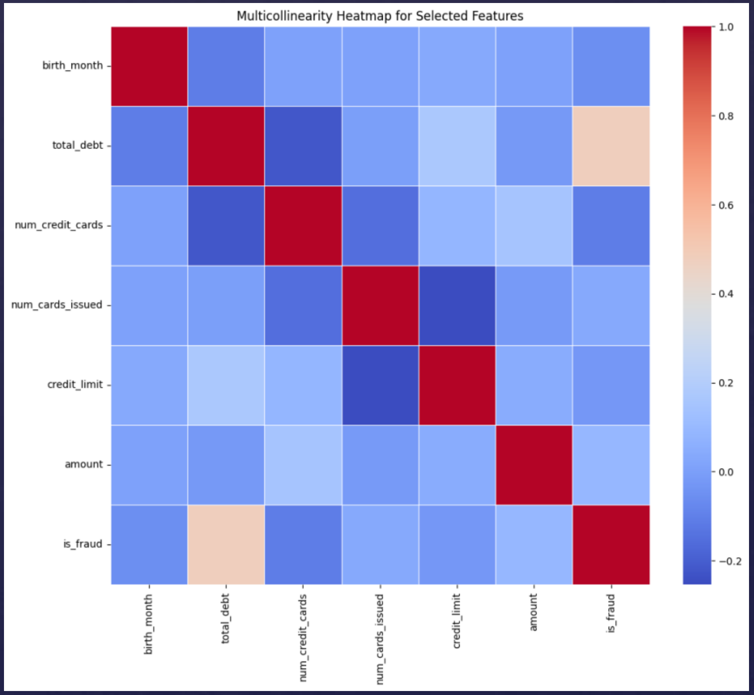
Correlation Heatmap for Feature Selection
mcc
current_age
retirement_age
birth_year
birth_month
per_capita_income
yearly_income
total_debt
credit_score
num_credit_cards
num_cards_issued
credit_limit
amount
is_fraud
gender_male
card_brand_Discover
card_brand_Mastercard
card_brand_Visa
card_type_Debit
card_type_Debit (Prepaid)
has_chip_YES
use_chip_Online Transaction
use_chip_Swipe Transaction
errors_Bad CVV,Insufficient Balance
errors_Bad Card Number
errors_Bad Expiration
errors_Bad PIN
errors_Bad Zipcode
errors_Insufficient Balance
errors_Insufficient Balance,Technical Glitch
errors_Technical Glitch
errors_none
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

```
Variance Inflation Factor (VIF) Results:
              Feature         VIF
0                 mcc   16.815939
1         current_age   18.501917
2      retirement_age  171.379854
3         birth_month    4.815610
4   per_capita_income   12.947993
5          total_debt    3.547813
6        credit_score  126.567385
7     num_credit_cards    9.754125
8     num_cards_issued    9.432587
9        credit_limit    4.269002
10             amount    1.203849

Autocorrelation Check using Ljung-Box Test:
      lb_stat  lb_pvalue
10  24.658515   0.006032

Heteroscedasticity Test using Breusch-Pagan:
{'LM Statistic': 1946.6844350997947, 'LM-Test p-value': 0.0, 'F-Statistic': 390.8437585380092, 'F-Test p-value': 0.0}

Pearson Correlation P-values with Target Variable 'is_fraud':
birth_month: p-value = 1.0394178456815732e-08
total_debt: p-value = 0.0
num_credit_cards: p-value = 4.27296447649582e-32
num_cards_issued: p-value = 4.255322676660147e-05
credit_limit: p-value = 0.0070754529397043355
amount: p-value = 1.0417764840878858e-21
```



```
                        Model  Precision (Class 0)  Recall (Class 0)  \
0         Logistic Regression             0.768827          0.737815
1               Decision Tree             0.889078          0.875630
2               Random Forest             0.894737          0.885714
3           Gradient Boosting             0.875562          0.981513
4  Support Vector Classifier             0.773179          0.784874
5                 Naive Bayes             0.602620          0.927731
6         K-Nearest Neighbors             0.876543          0.954622

   F1-Score (Class 0)  Precision (Class 1)  Recall (Class 1)  \
0            0.753002             0.945397          0.953406
1            0.882303             0.973962          0.977056
2            0.890203             0.976048          0.978115
3            0.925515             0.996016          0.970702
4            0.778982             0.954674          0.951641
5            0.730642             0.982882          0.871514
6            0.913918             0.990288          0.971761

   F1-Score (Class 1)  Accuracy   ROC-AUC          Confusion Matrix
0            0.949385  0.915986  0.966951  [[439, 156], [132, 2701]]
1            0.975507  0.959452  0.926343    [[521, 74], [65, 2768]]
2            0.977080  0.962077  0.988090    [[527, 68], [62, 2771]]
3            0.983196  0.972579  0.990243    [[584, 11], [83, 2750]]
4            0.953155  0.922695  0.966491  [[467, 128], [137, 2696]]
5            0.923854  0.881272  0.957194   [[552, 43], [364, 2469]]
6            0.980937  0.968786  0.987330    [[568, 27], [80, 2753]]
```

```
Fitting 3 folds for each of 16 candidates, totalling 48 fits
Using cpu device
Error during RL training: This StackingClassifier instance is not fitted yet. Call 'fit' with appropriate arguments before using this estimator.
Classification Report for Blended Model with Reinforcement Learning:
              precision    recall  f1-score      support   ROC-AUC
0              0.784161  0.848739  0.815174   595.000000       NaN
1              0.967672  0.950935  0.959231  2833.000000       NaN
accuracy       0.933197  0.933197  0.933197     0.933197  0.969525
macro avg      0.875917  0.899837  0.887202  3428.000000       NaN
weighted avg   0.935820  0.933197  0.934227  3428.000000       NaN
```

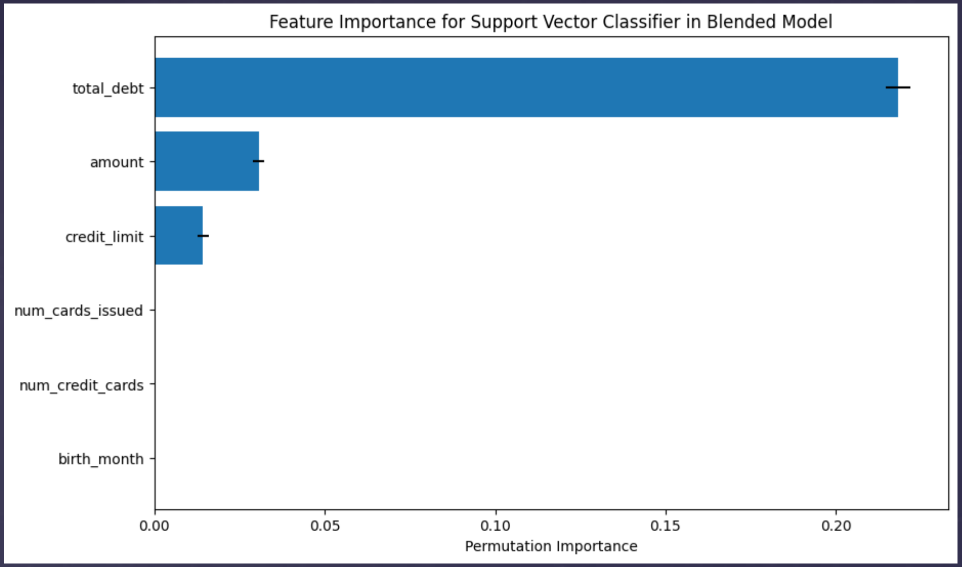
Key Results

Performance Metrics:

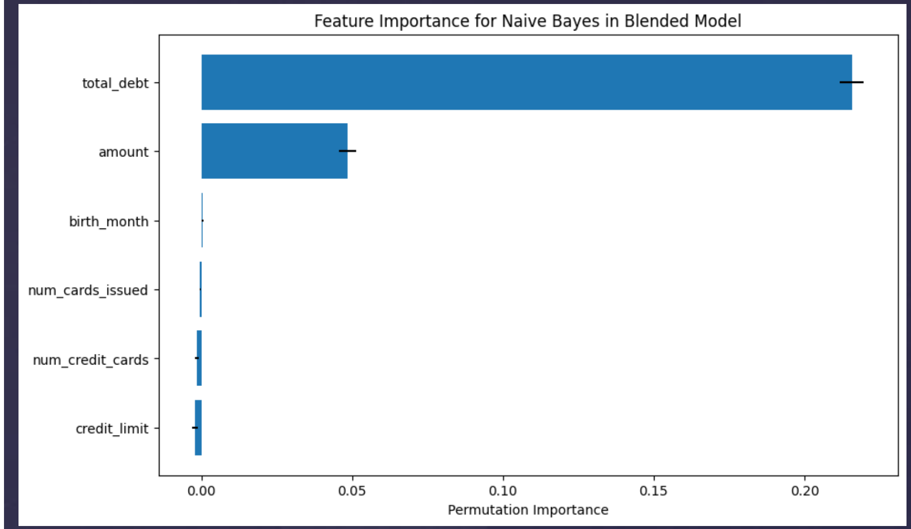
- **Accuracy**: 93.5%
- **Precision**: High focus on reducing false positives.
- **Recall**: Captured a high proportion of actual fraud cases.

Model Impact:

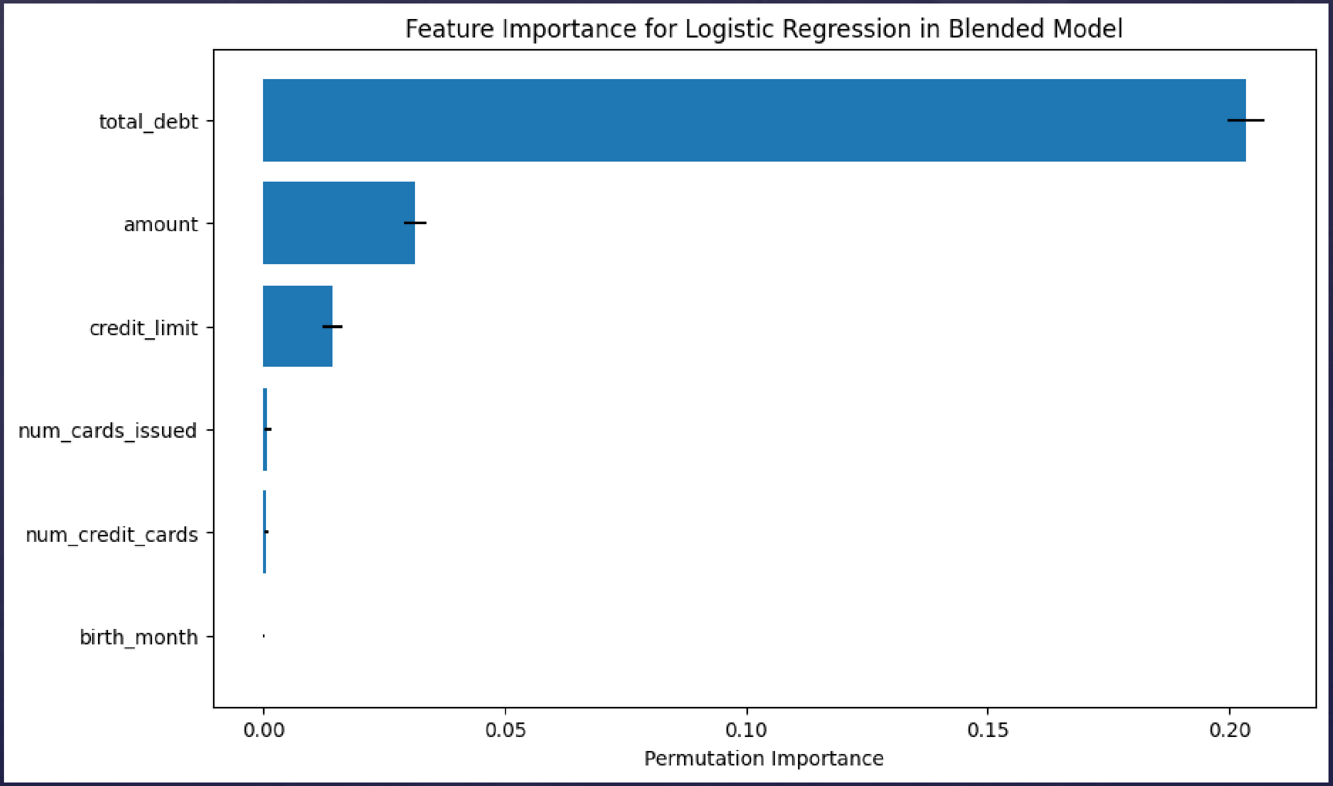
The refined blended model provided a balanced outcome, minimizing both false positives and false negatives.



Feature Importance for Support Vector Classifier in Blended Model
total_debt
amount
credit_limit
num_cards_issued
num_credit_cards
birth_month
0.00
0.05
0.10
0.15
0.20
Permutation Importance

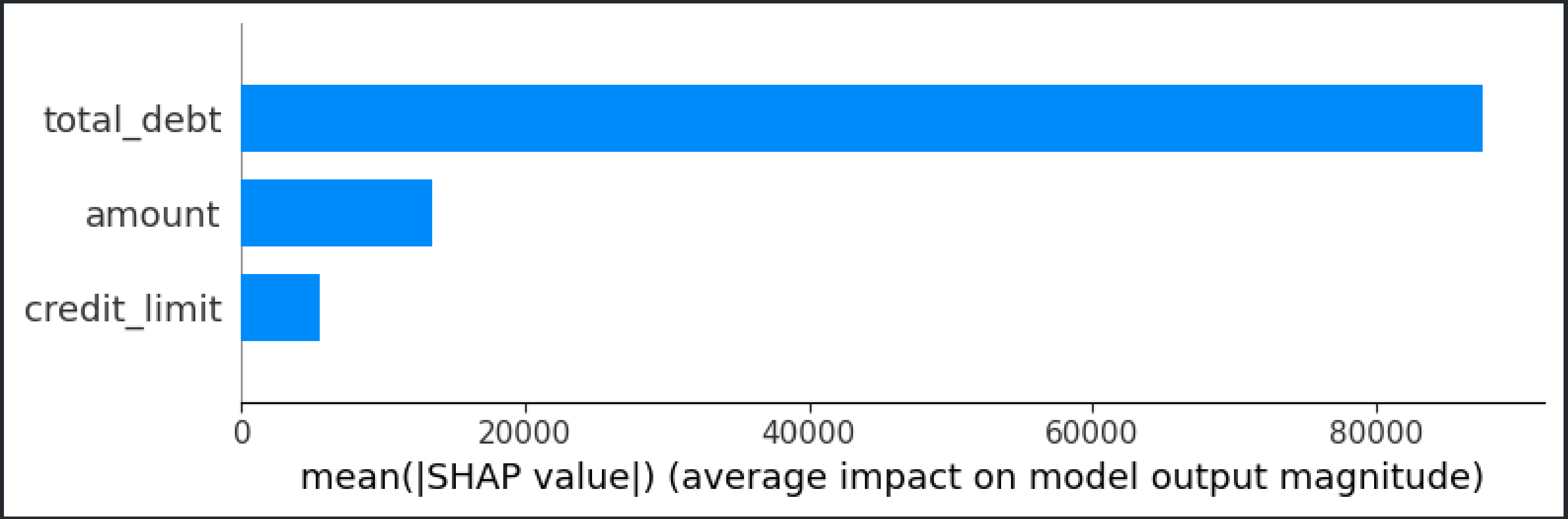


Feature Importance for Naive Bayes in Blended Model
total_debt
amount
birth_month
num_cards_issued
num_credit_cards
credit_limit
0.00
0.05
0.10
0.15
0.20
Permutation Importance



Feature Importance for Logistic Regression in Blended Model
total_debt
amount
credit_limit
num_cards_issued
num_credit_cards
birth_month
0.00
0.05
0.10
0.15
0.20
Permutation Importance

```
Fitting 3 folds for each of 16 candidates, totalling 48 fits
Cross-Validation Scores: [0.916875   0.93125    0.93746091 0.91557223 0.9318324 ]
Mean Cross-Validation Score: 0.9265981081926205
Classification Report for Refined Blended Model:
              precision    recall  f1-score      support   ROC-AUC
0              0.781627  0.872269  0.824464   595.000000       NaN
1              0.972504  0.948818  0.960515  2833.000000       NaN
accuracy       0.935531  0.935531  0.935531     0.935531  0.969992
macro avg      0.877065  0.910543  0.892489  3428.000000       NaN
weighted avg   0.939373  0.935531  0.936900  3428.000000       NaN
```



Top Features:

- **Total Debt**: Strongest predictor of fraudulent activity.
- **Transaction Amount**: Significant impact on detection.
- **Credit Limit**: Provided additional context for model predictions.

Insights:

Focus on reducing customer debt and monitoring high-value transactions for fraud indicators.

Feature Importance

Optimization: Removed less significant features to streamline the model.

Cross-Validation: Improved model consistency and reduced overfitting risks.

Conclusion: Achieved a well-balanced, reliable fraud detection model.

Refinement Process

Scalable API:

Implemented a RESTful API for easy integration into existing systems.

Real-time Predictions:

Immediate fraud detection using a Flask web service.

Business Impact:

Faster response times, improved fraud detection accuracy, and enhanced security.

Deployment Strategy

Fraud Reduction:
Significant reduction in fraudulent activities.

Customer Trust:
Improved due to fewer false alarms and better service.

Operational Efficiency:
Automated fraud detection reduces manual review costs.

Key Business Benefits

Model Monitoring:
Implement continuous model monitoring to ensure performance stability.

Scalability:
Plan for deployment in other regions or across different transaction types.

Feedback Loop:
Integrate a feedback mechanism to enhance model training with real-world data.

Next Steps

Conclusion

Summary:

The blended model effectively detects fraud, balancing recall and precision while minimizing false positives.

Call to Action:

Approve deployment to production for enhanced fraud management.