Internship Task : Accredian

Data Science & Machine Learning

Report

By

Aditya Kumar Pandey

IIIT Bhagalpur, MTech (CSE)

AI & Data Science

Fraud Detection Project Questionnaire

1. Data Cleaning Including Missing Values, Outliers, and Multi-Collinearity

Missing Values: The dataset was thoroughly checked for missing values at the outset, and none were found. All columns contained complete records, so no imputation or row removal was necessary.
Outliers: Exploratory data analysis (EDA) was performed to inspect the distributions of transaction amounts and balances. While some high-value transactions exist, no extreme or obviously erroneous outliers were detected that required removal. The log-transformation of the 'amount' feature was applied to reduce the impact of skewness, further mitigating the effect of large values.
Multi-Collinearity: Initial feature engineering introduced several derived features (e.g., balance deltas, discrepancy features), which led to perfect or near-perfect multicollinearity, as revealed by infinite or extremely high Variance Inflation Factor (VIF) values. To resolve this, redundant and perfectly collinear features—such as balance deltas and multiple time-based features (step, day, hour)—were systematically removed. After this cleanup, a follow-up VIF check showed no more infinite values, and while some features (like newbalanceDest and oldbalanceDest) still showed moderate multicollinearity, all VIFs were brought to acceptable levels for modeling. This ensures the models' coefficients and feature importances are reliable and interpretable.

2. Describe Your Fraud Detection Model in Elaboration

- Three different machine learning models were developed to detect fraudulent transactions: Logistic Regression, Random Forest, and XGBoost.
- Logistic Regression served as a baseline, providing a simple, interpretable model that is easy to explain to stakeholders.

- Random Forest and XGBoost are advanced tree-based algorithms capable of capturing complex, non-linear patterns in the data.
- All models were trained using class weights or scale_pos_weight to address the extreme class imbalance (fraud cases are very rare).
- The dataset was split into training and test sets (70:30) stratified by the fraud label to ensure both sets had similar fraud rates.
- Hyperparameter tuning was considered, but the focus was on comparative performance using default settings for clarity. Each model's performance was evaluated using precision, recall, F1-score, and ROC-AUC, with particular attention to the fraud class.

Direct Comparison Table

Metric	Logistic Reg.	Random Forest	XGBoost
Recall (Fraud)	0.91	0.79	0.99
Precision (Fraud)	0.06	0.97	0.55
F1-Score (Fraud)	0.10	0.87	0.71
ROC-AUC	0.987	0.994	0.998
False Alarms	Many	Very few	Moderate
Missed Frauds	Few	Moderate	Very few

3. How Did You Select Variables to Be Included in the Model?

- Variables were selected through a combination of domain knowledge and exploratory data analysis.
  Given that fraud occurs almost exclusively in "TRANSFER" and "CASH_OUT" transactions, irrelevant transaction types were filtered out early.

- Feature engineering was used to create new variables that capture meaningful patterns: for example, whether the originator's account was emptied, transaction type, transaction amount, log-transformed amount,

and account balances before and after transactions.

- Features that were direct linear combinations of others (e.g., balance deltas, discrepancy features) were identified and removed to eliminate multicollinearity.

- Statistical checks (VIF) and model-based feature importance rankings (from Random Forest and XGBoost) were used to confirm which features were most predictive and which could be safely excluded.

4. Demonstrate the Performance of the Model by Using Best Set of Tools

The performance of each model was evaluated using standard metrics for imbalanced classification problems:

- Logistic Regression: Achieved a recall (true positive rate) of 0.91 for fraud, meaning it catches 91% of actual frauds, but with a precision of only 0.06, resulting in many false positives. ROC-AUC was 0.987.
- Random Forest: Achieved a precision of 0.97 for fraud, meaning almost all transactions it flags as fraud are real, but its recall was 0.79, so it misses some frauds. ROC-AUC was 0.994.
- XGBoost: Achieved the highest recall (0.99), missing almost no frauds, with a precision of 0.55, which is much better than logistic regression. ROC-AUC was 0.998, the highest of the three.

Classification reports, ROC-AUC scores, and feature importance tables were generated for each model. These metrics show that XGBoost provides the best overall balance between catching frauds and minimizing false alarms, though each model optimizes a different part of the precision-recall trade-off.

5. What Are the Key Factors that Predict Fraudulent Customer?

The top predictive features varied somewhat by model, but several factors consistently stood out:

- Emptied Originator Account (is_account_emptied): Transactions where the sender's balance goes to zero were highly predictive of fraud.
- Transaction Type (type_TRANSFER): "TRANSFER" transactions were much more likely to be fraudulent than others.
- Transaction Amount (amount, log_amount): The size of the transaction was important, though the relationship was not always straightforward—very large transactions were less likely to be fraudulent in this dataset.
- Account Balances (oldbalanceOrg, newbalanceDest): The sender's balance before the transaction and the recipient's balance after were also influential, especially in tree-based models.

These features align with intuitive financial fraud patterns: fraudsters often move funds out of accounts (emptying them) via transfers, and the transaction amounts and balances reflect the mechanics of these actions.

6. Do These Factors Make Sense? If Yes, How? If Not, How Not?

- Yes, these factors make sense in the context of financial fraud. Emptying an account via a transfer is a classic red flag for fraud, as criminals seek to maximize their gain quickly.
- "TRANSFER" type is specifically highlighted because fraudsters often move money out of accounts rather than spending it directly.
- Large transactions being less likely to be fraud in this dataset is plausible if the majority of large transactions are legitimate (e.g., business payments), while fraudulent transactions tend to be moderate in size.
- Account balance changes reflect the flow of funds, which is central to detecting unauthorized movements.
- While some counterintuitive patterns emerged (e.g., large transactions less likely to be fraud), these are explainable by the specific data

distribution and do not undermine the overall logic of the model. In summary, the key factors identified are consistent with financial fraud scenarios and provide actionable signals for detection.

7. What Kind of Prevention Should Be Adopted While the Company Updates Its Infrastructure?

The company should adopt a multi-layered fraud prevention strategy informed by the model's findings:

- Real-Time Transaction Monitoring: Implement systems to flag transactions that empty accounts, especially via transfers, for immediate review.
- Rule-Based Alerts: Combine machine learning alerts with simple business rules (e.g., large or rapid balance changes) to catch obvious fraud patterns.
- Behavioral Analytics: Monitor user transaction patterns over time to detect anomalies that might indicate account takeover or insider fraud.
- Human Review: Ensure that all machine-generated alerts are reviewed by trained staff to reduce false positives and adapt to evolving fraud tactics.
- User Education: Educate customers about secure practices to reduce social engineering and phishing risks.
- Continuous Model Updating: Regularly retrain models on new data to adapt to changing fraud patterns.

Updating infrastructure to support these steps—such as scalable real-time analytics, secure APIs, and integrated case management—will maximize the effectiveness of fraud detection.

8. Assuming These Actions Have Been Implemented, How Would You Determine If They Work?

To assess the effectiveness of the new fraud prevention measures, the company should:

- Track Key Performance Indicators (KPIs): Monitor the fraud detection rate (percentage of actual frauds caught), false positive rate (percentage of legitimate transactions flagged as fraud), and average time to detection.
- Conduct A/B Testing: Compare fraud rates and operational costs before and after implementation, or between groups using the new and old systems.
- Solicit Feedback: Collect input from fraud analysts and customers to identify missed frauds or unnecessary alerts.
- Regular Audits: Perform periodic reviews of model performance and update thresholds or features as needed.
- Monitor Financial Impact: Track reductions in actual fraud losses and operational costs associated with manual reviews.

By establishing a feedback loop—continuously measuring, learning, and improving—the company can ensure that its fraud prevention system remains effective as threats evolve.