Puja Patidar

Final Report: Credit Card Fraud Detection

1.Problem statement

The rapid growth of digital payment ecosystems has significantly increased the risk of credit card fraud, resulting in substantial financial losses and operational challenges for banks and financial institutions. Fraudulent transactions constitute only a very small fraction of total transactions, creating an extreme class imbalance that makes traditional rule-based and accuracy-driven detection methods ineffective. The objective of this capstone project is to develop a robust machine learning–based credit card fraud detection system that can accurately identify fraudulent transactions from highly imbalanced data. The solution prioritizes maximizing fraud detection (recall) while maintaining a manageable false alert rate, ensuring the model is both effective and operationally feasible. The project leverages advanced tree-based models, imbalance-aware training strategies, appropriate evaluation metrics such as PR-AUC, recall and threshold tuning to align model predictions with real-world fraud monitoring requirements.

2. Dataset Description

The dataset represents simulated credit card transactions recorded between 1st January 2019 and 31st December 2020, covering transaction behavior for 1,000 customers interacting with approximately 800 merchants. The data was generated using Sparkov Data Generation, a GitHub-based simulation tool developed by Brandon Harris, which is commonly used for creating realistic transaction patterns for fraud detection research and experimentation.

Source : https://www.kaggle.com/datasets/kartik2112/fraud-detection/data?select=fraudTrain.csv

- Number of rows: 1,852,394
- Number of features: 23
- Target variable: is_fraud (0,1)

3.Data Wrangling

The raw dataset initially contained 1,852,394 rows and 23 columns. The data types across columns included 5 numerical features of type float64, 6 integer features (int64), and 12 categorical features (object). As a first step, the dataset was explored to understand the meaning, role, and data type of each column in the context of credit card transactions.

Subsequently, the dataset was examined for data quality issues such as null values, missing values, and duplicate records. No null, missing, or duplicate values were observed in the dataset. This outcome is expected, as the data is synthetically generated rather than collected from real-world transaction systems, where such inconsistencies are common.

Histograms and box plots were generated for numerical features to examine their distributions and identify potential outliers. After completing these data wrangling steps, the final dataset contained 1,852,394 rows and 29 columns. Overall, the data wrangling process ensured that the dataset was clean, consistent, and analysis-ready.

4.Exploratory Data Analysis

4.1 Plotted histogram for all numeric data type and understand data distribution

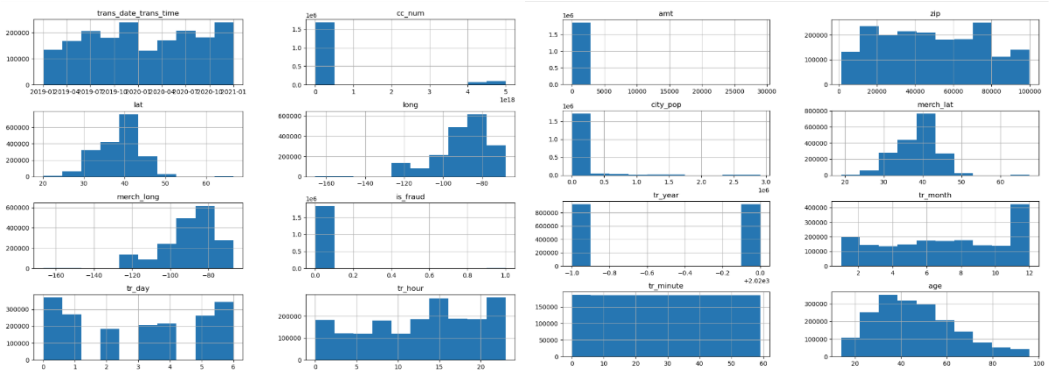


Figure1. Histogram of all numerical variables

From above histogram we can analyse:

1. Transaction amount (amt) is highly right-skewed, indicating most transactions are of low value with few high-value outliers.
2. Target variable (is_fraud) is extremely imbalanced, confirming fraud is a rare event.
3. Age follows an approximately bell-shaped distribution, with most customers in the working-age group.
4. Time features (hour) show uniform patterns, suggesting continuous transaction behaviour.
5. Geographical features (ZIP, lat/long) are widely distributed, indicating a diverse customer base.

4.2 Feature Engineering:

- Converted the trans_date_trans_time column to datetime format and extracted time-based features such as year, month, day, hour, and minute to capture detailed temporal transaction patterns.
- Calculated customer age using the date of birth and subsequently bucketed age into meaningful age groups to capture demographic spending behaviour.
- Extracted transaction day and day of the week from trans_date_trans_time to identify weekly spending patterns.
- Created binary indicators is_weekend and is_night to model behavioural differences during weekends and nighttime transactions.
- Computed the geographical distance between customer and merchant locations using Haversine distance method to identify anomalous transaction locations.

- Applied a log transformation to the transaction amount to reduce right skewness and stabilize variance.
- Engineered transaction velocity features by counting the number of transactions per card within the last 24 hours to capture burst activity.
- Created rolling statistical features by calculating the rolling mean of transaction amounts over the last 20 transactions per card (with a minimum of three observations) to represent recent spending behaviour.
- Derived an amount deviation ratio feature to quantify how unusual the current transaction amount is relative to the card's recent average spending.

4.3 Target Variable Distribution (Class Imbalance)

The dataset exhibits a severe class imbalance, with **1,842,743 legitimate (non-fraud) transactions** and **9,651 fraudulent transactions**. This corresponds to approximately **99.5% non-fraud** and **0.5% fraud** records, highlighting the need for imbalance-aware modeling and evaluation techniques.

4.4 Numerical Feature Analysis: Explored distributions of key numerical features and their relationship with fraud. Plotted a correlation heatmap to examine linear relationships among numerical variables.Transaction amount (amt) has a weak positive correlation with fraud, suggesting higher amounts are slightly more risky. Strong correlations exist among geographical features (e.g., lat–merch_lat, long–merch_long, zip–location), indicating redundancy.Time-based features (hour, day, month) show minimal correlation with fraud individually

4.5 Fraud rate by hour: The below graph shows that the **fraud rate is higher during night hours compared to daytime**.

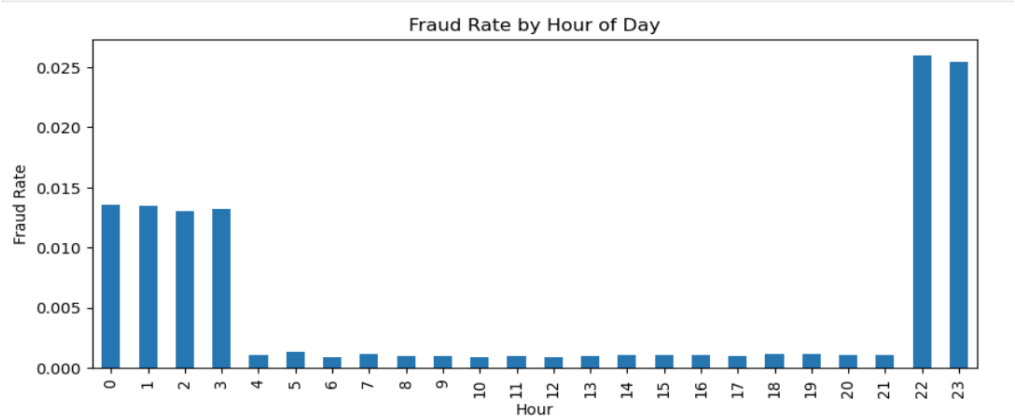


Figure 2 fraud rate by hour

4.6 Fraud rate by day: The below graph shows that Fraud rates vary by day, with higher incidence on Thursdays and Fridays and lower fraud activity on Mondays, indicating a mild day-of-week effect

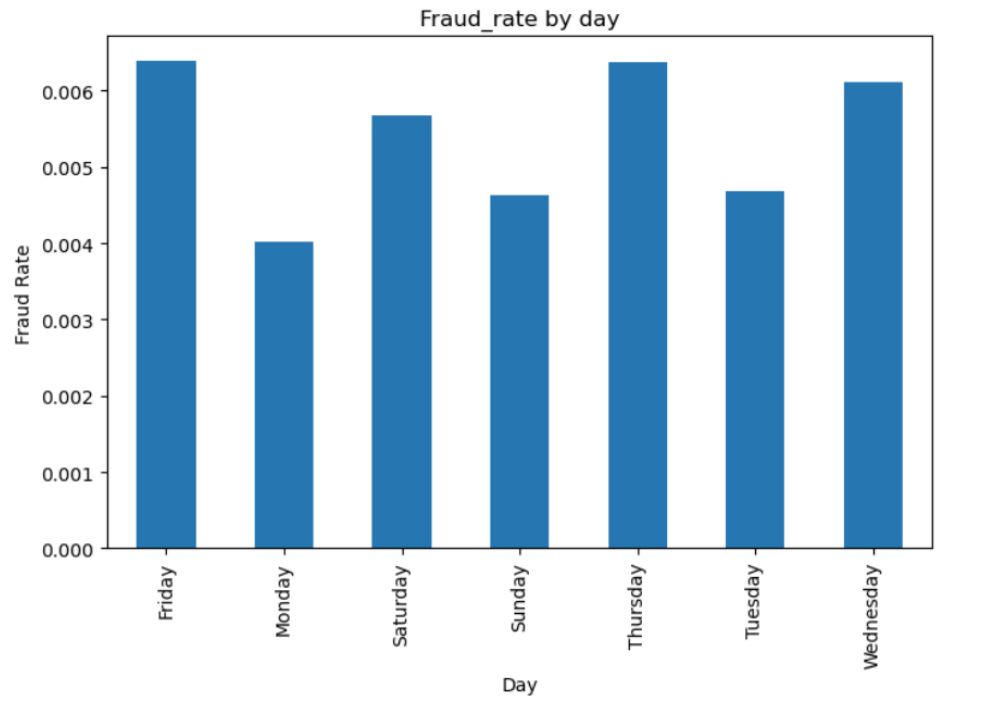


Figure 3 fraud rate by day

4.7 Fraud rate by month: Fraud rates show **seasonal variation**, peaking in **early months (Jan–Feb)** and gradually declining toward the **year end**, with the lowest fraud rate in **December**.

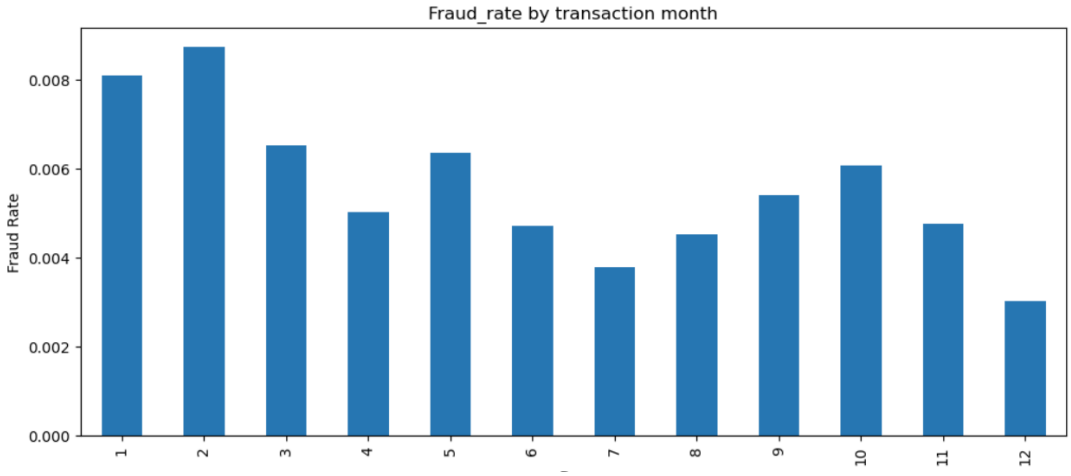
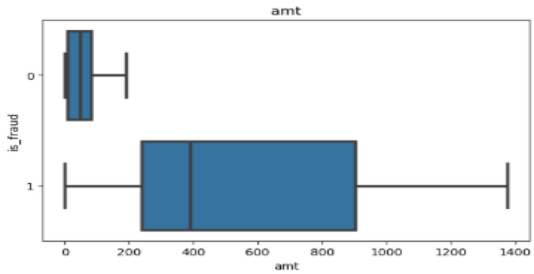
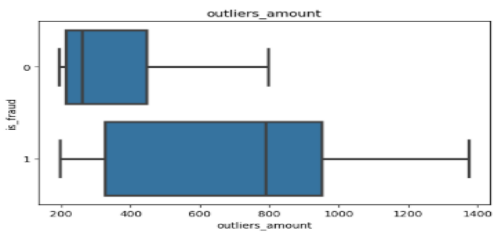


Figure 4 fraud rate by transaction month

4.8 Box plot of amount and outlier amount: Below Box plots of transaction amount and outlier amount show that fraud transactions have significantly higher values and greater variability compared to non-fraud transactions.





4.9 Categorical Feature Analysis:

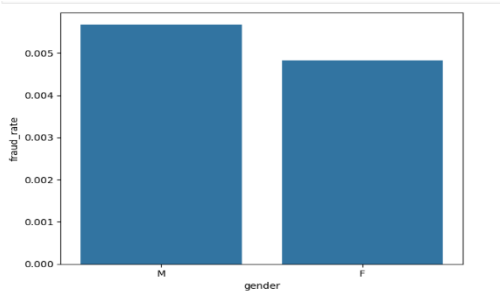
a. **Category :** Top 10 categories with highest fraud rate are as follows.

Category	total_txn	fraud_txn	fraud_rate
shopping_net	1,39,322	2219	1.59%
misc_net	90,654	1182	1.30%
grocery_pos	1,76,191	2228	1.26%
shopping_pos	1,66,463	1056	0.63%
gas_transport	1,88,029	772	0.41%
misc_pos	1,14,229	322	0.28%
grocery_net	64,878	175	0.27%
travel	57,956	156	0.27%
personal_care	1,30,085	290	0.22%
Entertainment	1,34,118	292	0.22%

b. **Merchant :** Top 10 Merchant with highest fraud rate are as follows

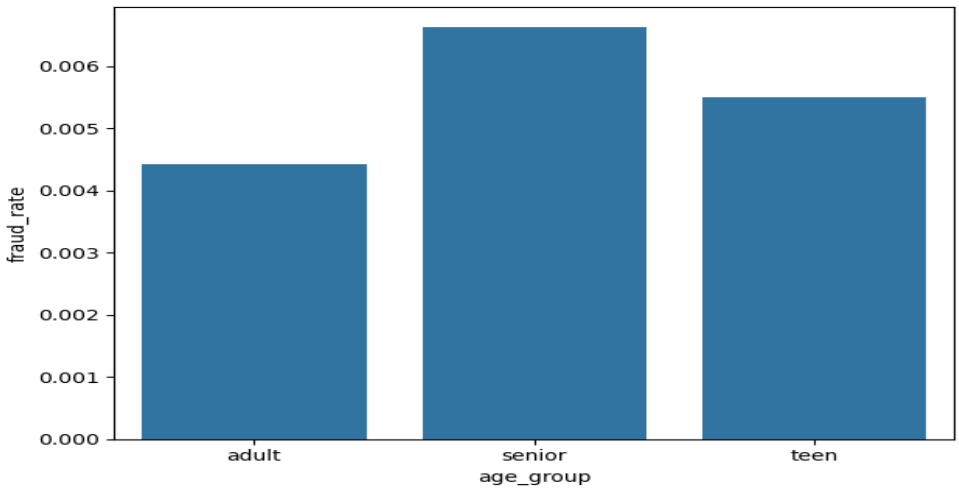
Merchant	total_txn	fraud_txn	fraud_rate
Kozey-Boehm	2758	60	2.18%
Herman, Treutel and Dickens	1870	38	2.03%
Terry-Huel	2864	56	1.96%
Kerluke-Abshire	2635	50	1.90%
Mosciski, Ziemann and Farrell	2821	53	1.88%
Schmeler, Bashirian and Price	2788	52	1.87%
Kuhic LLC	2842	53	1.86%
Jast Ltd	2757	51	1.85%
Langworth, Boehm and Gulgowski	2817	52	1.85%
Romaguera, Cruickshank and Greenholt	2767	51	1.84%

c. **Gender:** From following graph we can conclude that males have higher fraud rate than females



d. **Age Group:** <18 is teen, < 50 adult else senior

From the below graph, we can conclude that the **fraud rate is highest among senior citizens**, followed by the **teenage group**, and is **lowest in the adult age group**.



e. **Job :** 22 job categories show a 100% fraud rate, but each has fewer than 30 transactions.After excluding these low-volume categories, the top 10 job categories with the highest fraud rates are presented below.

Job	total_txn	fraud_txn	fraud_rate
TEFL teacher	760	32	4.21%
Lawyer	757	28	3.70%
Community development worker	751	22	2.93%
Accountant, chartered certified	751	21	2.80%
Horticultural consultant	746	19	2.55%
Clinical cytogeneticist	744	18	2.42%
Nature conservation officer	743	16	2.15%
Geneticist, molecular	745	16	2.15%
Writer	741	15	2.02%
Conservator, museum/gallery	743	15	2.02%

f. **State :** Top 10 states with highest fraud rate are shown in below table

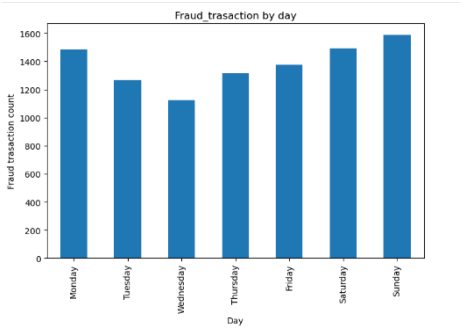
state	total_txn	fraud_txn	fraud_rate
DE	9	9	100.00%
RI	745	15	2.01%
AK	2963	50	1.69%
OR	26408	197	0.75%
NH	11727	79	0.67%
VA	41756	273	0.65%
TN	24913	159	0.64%
NE	34425	216	0.63%
MN	45433	280	0.62%
NY	119419	730	0.61%

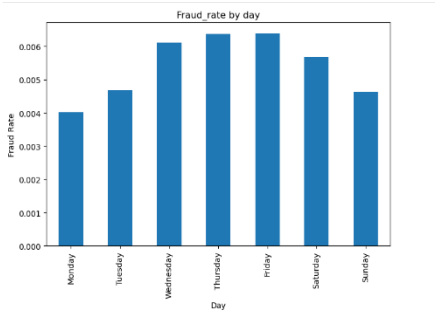
g. **City :** At the city level, 70 cities show a 100% fraud rate, but each has very few transactions (less than 20).

Excluding these low-volume cities, the top 10 cities with the highest fraud rates are shown in the table below.

city	total_txn	fraud_txn	fraud_rate
Clearwater	753	24	3.19%
Aurora	750	23	3.07%
Benton	750	20	2.67%
Jay	743	16	2.15%
Chatham	743	16	2.15%
Moscow	745	16	2.15%
Riverview	741	15	2.02%
Boulder	743	15	2.02%
Whittemore	743	15	2.02%
Howes Cave	743	15	2.02%

h. Fraud count and fraud rate by day of the week:





Fraud **counts peak on weekends**, while **fraud rates rise from Monday to Friday and peak on Thursday–Friday**, indicating higher risk on late working days despite higher weekend volumes

EDA conclusion:

- The dataset exhibits severe class imbalance, with fraudulent transactions forming a very small proportion of the data. Use imbalance-aware techniques such as SMOTE and prioritize evaluation metrics like PR-AUC and fraud recall instead of accuracy.
- Transaction amount shows a highly right-skewed distribution, with extreme values present in fraudulent transactions. Apply logarithmic transformation to stabilize variance and reduce the influence of outliers.
- Fraud occurrences vary significantly across time, particularly during nighttime hours and certain days of the week. Engineer time-based features such as hour, day of week, is_weekend, and is_night to capture temporal patterns.
- Fraudulent behavior often appears as bursts of activity within short time windows. Create transaction velocity features by counting the number of transactions per card within the last 24 hours.
- Absolute transaction amounts alone are insufficient to identify fraud, as spending behavior differs widely across customers. Generate rolling statistical features and deviation ratios to capture deviations from a card's recent spending behavior.

- Geographical distance between customer and merchant locations can indicate anomalous transactions, especially when transactions occur far from usual locations. Compute merchant–customer distance using geographical coordinates as a spatial anomaly feature.
- Fraud patterns differ across customer demographics and merchant categories, though sparse groups can produce misleading extremes.Carefully encode categorical features and avoid overinterpreting low-frequency categories.
- Fraud signals arise from complex interactions among multiple features rather than single variables. Use non-linear, tree-based ensemble models capable of capturing interactions between temporal, behavioral, and monetary features.

5.Preprocessing and Modelling:

5.1. Objective of the Modelling Phase

The primary objective of the modeling phase is to build a robust machine learning model capable of accurately identifying fraudulent credit card transactions from highly imbalanced data. Given that fraudulent transactions constitute a very small fraction of the total dataset, traditional accuracy-based evaluation is inappropriate. Therefore, the modeling strategy focuses on maximizing fraud detection capability while maintaining generalization and minimizing overfitting.

5.2. Data Preparation and Preprocessing

Before model training, the dataset underwent the following preprocessing steps:

- Conversion of low cardinality categorical columns to numeric formats by using one hot encoder.
- Reduction of the dataset from **1.8 million to 100,000 records** using **stratified sampling** to preserve the proportion of fraudulent transactions
- Separation of features (X) and target variable (y), where the target represents fraudulent transactions.
- Stratified splitting of data into training and testing sets to preserve class distribution.

To handle class imbalance during training, SMOTE (Synthetic Minority Oversampling Technique) was applied only on the training data within a pipeline to avoid data leakage.

5.3. Modelling Strategy

Model Selection: Multiple models were evaluated to identify the most suitable approach for fraud detection: **Logistic Regression (baseline non-tree model), Decision Tree Classifier, Random Forest Classifier**, **XGBoost Classifier.**

Each model was trained using a consistent preprocessing and resampling pipeline to ensure fair comparison.

5.4. Evaluation Metrics

Given the imbalanced nature of the dataset, the following metrics were selected:

- **Precision-Recall AUC (PR-AUC)** – primary metric, as it focuses on minority class performance.
- **Recall (Fraud Class)** – measures the ability to capture fraudulent transactions.
- ROC-AUC was monitored but not used as the primary decision metric.

Rationale: PR-AUC provides a more informative evaluation than ROC-AUC in scenarios where the negative class dominates.

5.6. Cross-Validation Approach

To ensure model robustness, **Stratified K-Fold Cross-Validation** was employed on the training dataset. This preserves the fraud-to-non-fraud ratio across folds.

For each model, the following metrics were computed as shown below

Model	Train PR-AUC	Validation PR-AUC	Train Recall	Validation Recall
Logistic Regression + SMOTE	0.2075	0.2066	0.9203	0.8536
Decision Tree	0.2478	0.2411	0.8759	0.8489
Random Forest	0.734	0.6711	0.8843	0.8585
XGBoost	0.7742	0.7163	0.9442	0.8705

5.7. Baseline Model Performance

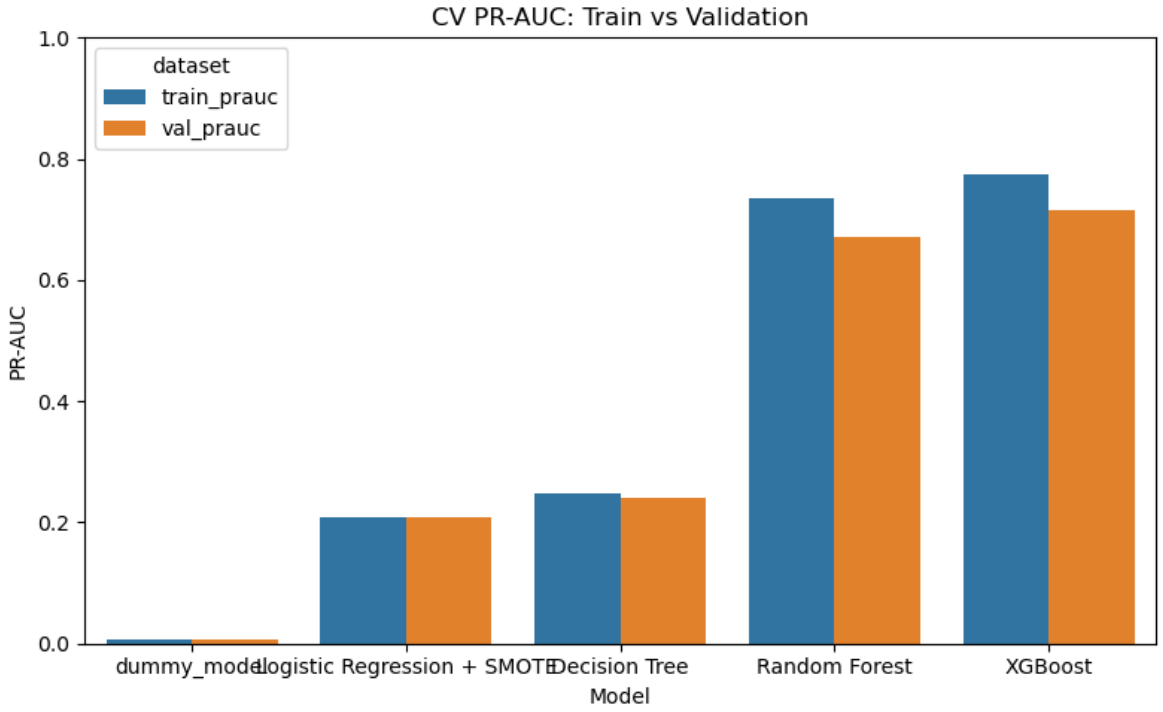
A Dummy Classifier (Most Frequent strategy) was used as a baseline.

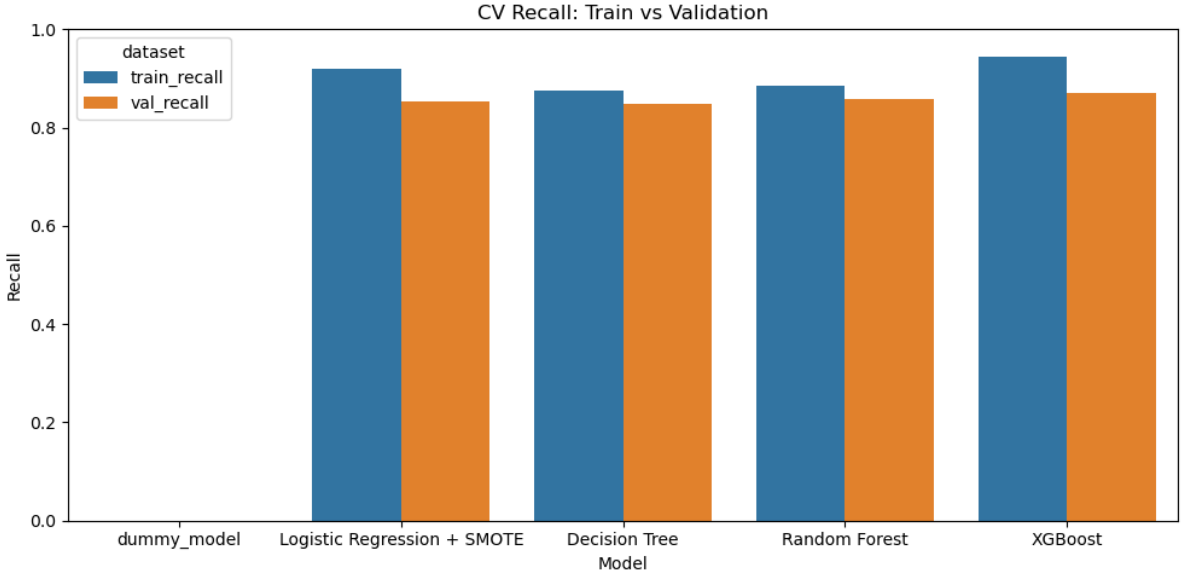
- PR-AUC ≈ Fraud prevalence
- Recall = 0%

This confirms that any meaningful model must significantly outperform the baseline to be considered useful.

5.8. Comparative Model Performance

- **Logistic Regression + SMOTE** achieves high recall but very low PR-AUC, indicating a poor precision–recall trade-off and weak fraud-ranking ability.
- **Decision Tree** shows marginal PR-AUC improvement over Logistic Regression but exhibits instability and limited generalization with modest overall performance.
- **Random Forest** significantly improves PR-AUC, reflecting strong fraud-ranking capability, though the train–validation gap indicates moderate overfitting.
- **XGBoost** delivers the highest validation PR-AUC (0.7163) and recall (0.8705) with a small train–validation gap, demonstrating strong generalization and the best overall performance.
- **Overall, XGBoost provides the best balance between fraud detection effectiveness (PR-AUC) and stable recall, making it the most suitable final model for this imbalanced fraud detection problem.**





5.9. Final Model Selection: XGBoost

Based on cross-validation performance, **XGBoost** was selected as the final model due to:

1. Highest validation PR-AUC among all models.
2. Strong recall for the fraud class.
3. Controlled generalization gap between training and validation metrics.
4. Robust handling of nonlinear feature interactions.

5.10. XGBoost Model Configuration

The final XGBoost model was trained using the following key hyperparameters:

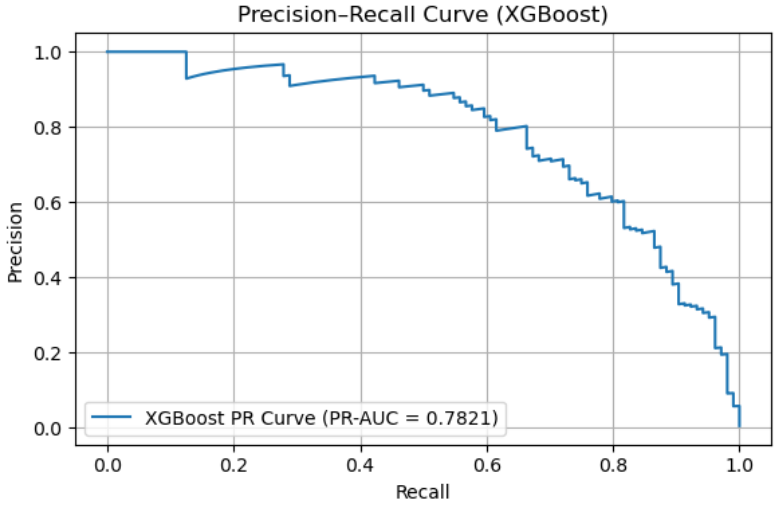
Parameter	Value
n_estimators	600
learning_rate	0.05
max_depth	2
min_child_weight	100
gamma	1.5
subsample	0.6
colsample_bytree	0.6
reg_alpha	10
reg_lambda	30
scale_pos_weight	pos_weight × 0.25
tree_method	hist
eval_metric	aucpr
random_state	42
n_jobs	-1

Hyperparameter tuning was performed to mitigate overfitting while maintaining high recall.

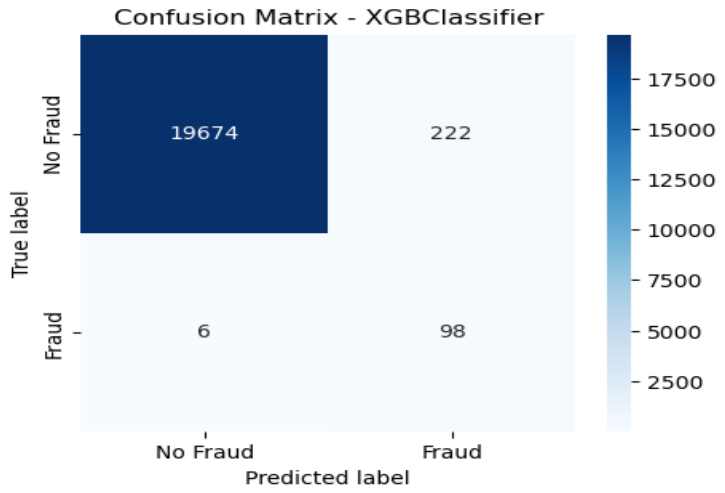
11. Final Model Evaluation on Test Data

After training the final XGBoost model on the full training set, performance was evaluated on the held-out test set.

Evaluation included:PR-AUC on test data,Recall for fraud transactions,Confusion matrix analysisPrecision–Recall curve visualization.



	precision	recall	f1-score	support
0	1.00	0.99	0.99	19896
1	0.31	0.94	0.46	104



The final model demonstrates strong fraud detection capability with acceptable false positive trade-offs.

13. Limitations

Despite strong performance, the model has certain limitations:

- Results are dependent on sampled data and may vary on full-scale deployment.
- Synthetic oversampling may not fully capture real fraud patterns.
- High recall may increase false positives, requiring operational tuning.

14. Conclusion

This modeling exercise demonstrates a systematic and metrics-driven approach to fraud detection under extreme class imbalance. Through careful preprocessing, appropriate metric selection, cross-validation, and model comparison, **XGBoost emerged as the most effective model**.

The final model provides a strong balance between fraud detection capability and generalization, making it suitable for real-world deployment with threshold customization.

15. Future Work

Future work focuses on making the model more business-aware, time-sensitive, and production-ready.