Classifying Bank Fraud: Protecting Our Customers

Why Fraud Classification Matters

Protect Customers

Safeguard sensitive financial data and prevent identity theft.

Reduce Losses

Minimize financial damage caused by fraudulent transactions.

Ensure Trust

Build customer confidence in our security measures and services.

Motivation

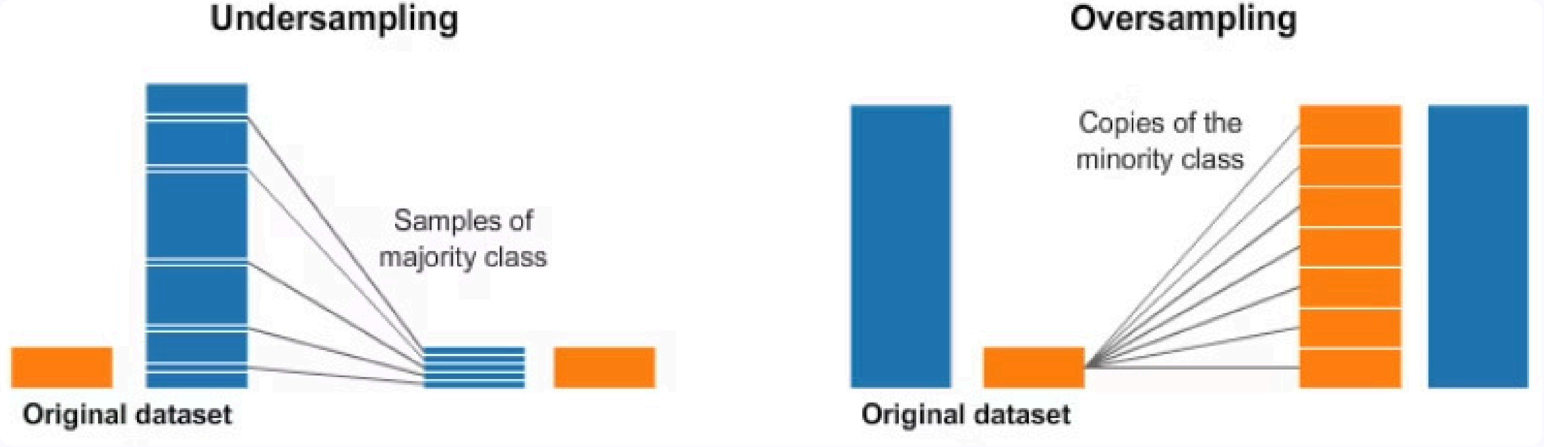
- Credit card fraud is escalating with the growth of online transactions, while traditional rule-based systems fail to keep up.
- This creates a need for real-time, intelligent detection to prevent financial loss.
- Banks are often hesitant to make major system changes, so it is important to offer solutions that are both effective and easy to integrate with their existing systems.

Introduction

Imbalanced data occurs in classification tasks when one class significantly outnumbers the other(s), leading to a skewed class distribution (e.g., 1:100 or 1:1000), which can bias model performance.

There are several approaches to solving class imbalance problem:

- More samples from the minority class should be acquired from the knowledge domain.
- Changing the loss function to give the failing minority class a higher cost.
- Oversampling the minority class.
- Undersampling the majority class.
- Any combination of previous approaches.



Problem Definition

Near Real-Time Classification and Accessibility

Near real-time fraud detection is essential for minimizing financial losses, as prompt alerts allow banks to respond quickly and effectively.

Imbalance Data

In credit card fraud dataset, data is naturally imbalanced, with rare label = "fraud" overshadowed by normal cases. Addressing this is crucial for building effective detection models.

Business Goal

1 **Early Detection**

Identify high-risk transactions before they cause harm.

2 **Loss Reduction**

Decrease financial losses due to fraud by at least 15% in the next quarter.

3 **Customer Protection**

Safeguard customer data to maintain trust and brand reputation.

4 **Operational Efficiency**

Utilize machine learning to streamline fraud monitoring processes.

Exploratory Data Analysis (EDA)

Data

- Dataset containing 1,852,394 credit card transactions.
- Target label: "is_fraud" (1 = fraudulent, 0 = legitimate)

Link to data- https://www.kaggle.com/datasets/kartik2112/fraud-detection/data

Made with GAMMA

Data Fields: Understanding Customer Transactions

- **High Importance:** amt, category, merchant, lat/long, trans_date_trans_time, merch_lat/long.
- **Moderate Importance:** gender, job, dob, city_pop.
- **Low Importance:** first, last, street, zip, cc_num, trans_num.

Data Distribution

Class Imbalance

Only 0.52% of transactions are fraudulent.

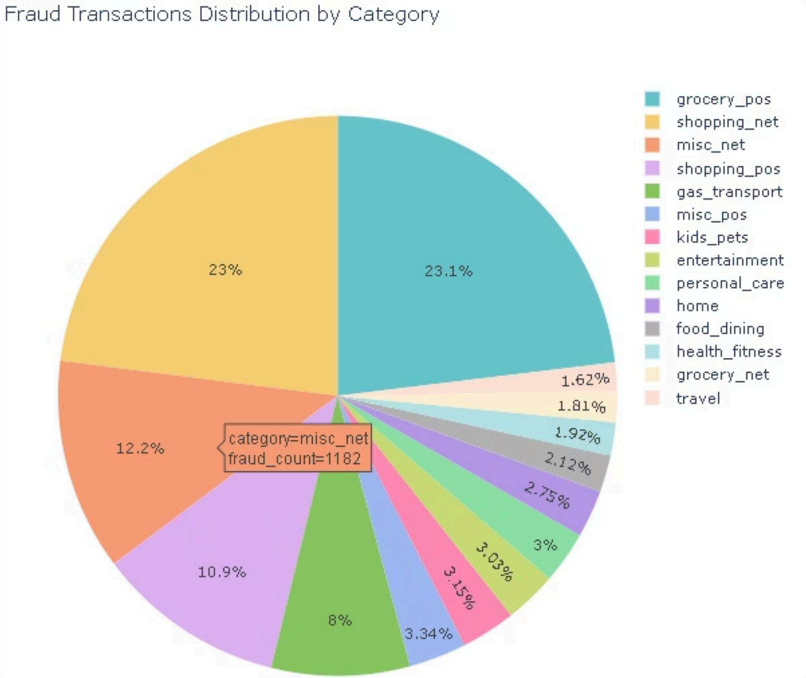
Visualization

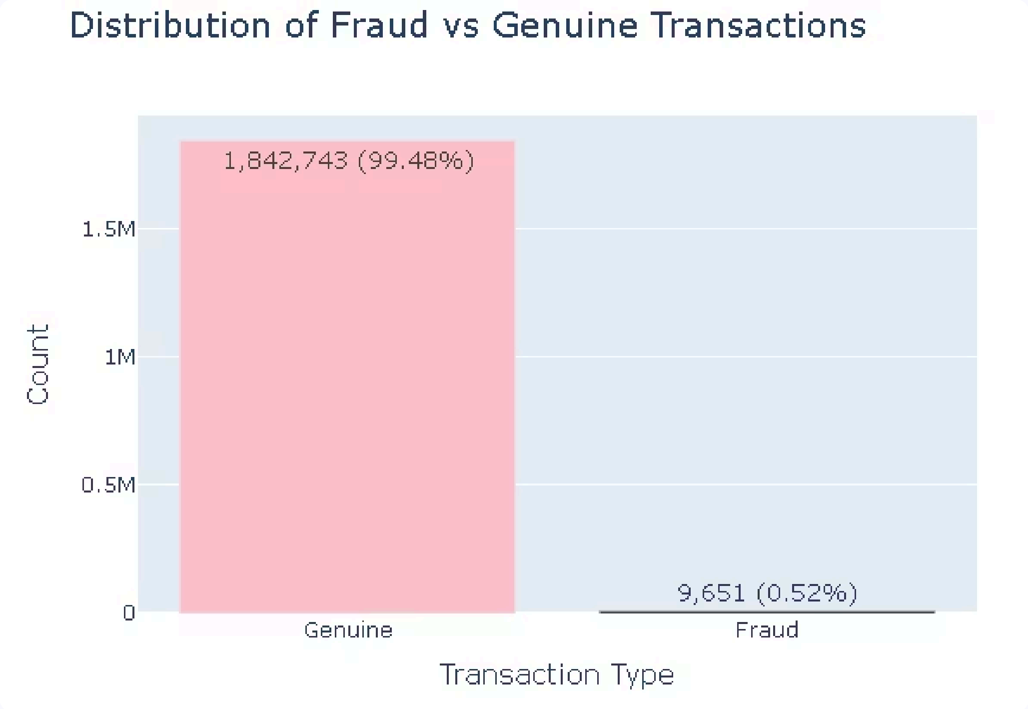
Histograms and box plots reveal patterns by class.

Train / Test Split

90/10 ratio used for model training and evaluation.

Data Distribution - Examples





Preprocessing & Modeling

Data Fields: Understanding Customer Transactions

Transaction Details:

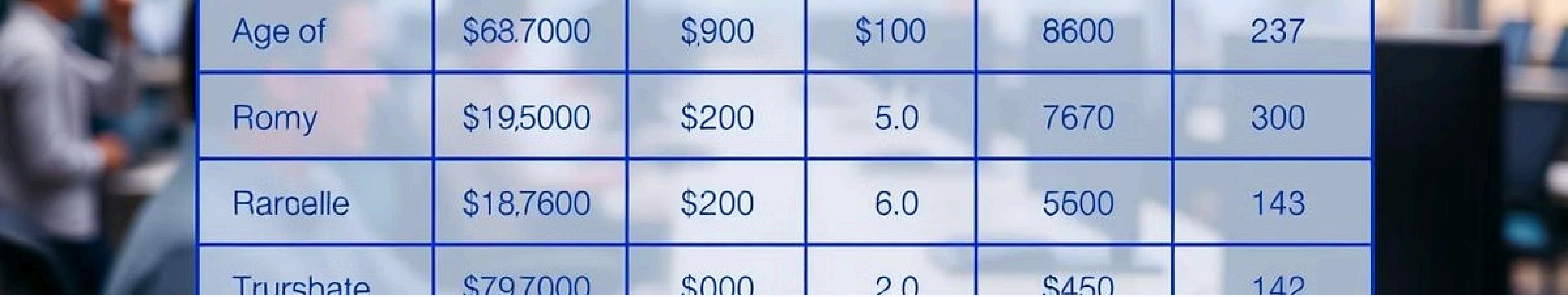
Amount, Transaction Time, and Merchant and Category – used as raw inputs for feature creation and transformation.

Customer Demographics:

DOB, Gender, Job, City, and State – used to derive age and encode categorical traits.

Time-Derived Fields:

Transaction Time available in full timestamp format – used to derive temporal indicators such as hour, day, month, and night-time flags.



Feature Engineering

Feature engineering transforms raw data into meaningful inputs for fraud detection models. In our project, we derived time-based patterns, encoded user demographics, and normalized transaction values to improve classification.

Temporal & Behavioral Features

> Derived hour, day, month, is_night, and age to model user behavior patterns.

Log Transformation & Scaling

Transformed amount using log1p(), then applied StandardScaler to normalize numerical features.

Encoding Categorical Data

> Encoded job, gender, state, city, and category using LabelEncoder.

Libraries and Methods

Data Handling:

Pandas, NumPy - preprocessing

LabelEncoder, StandardScaler - encoding & scaling

Datetime, Counter - date & class counts

Visualization:

Matplotlib, Seaborn

ML Models:

Scikit-learn - model training & evaluation

XGBoost, Random Forest, EasyEnsemble

Resampling Imbalanced-learn:

SMOTE, ROS, RUS, Tomek Pipeline - combine sampling with training

Utilities

Joblib, Pickle - save models

GridSearchCV, StratifiedKFold - tuning & CV

Classification Report, ROC AUC, Confusion Matrix

Model Selection & Implementation

We compared three tree-based models to identify the strongest foundation for fraud detection:

1 **XGBoost**

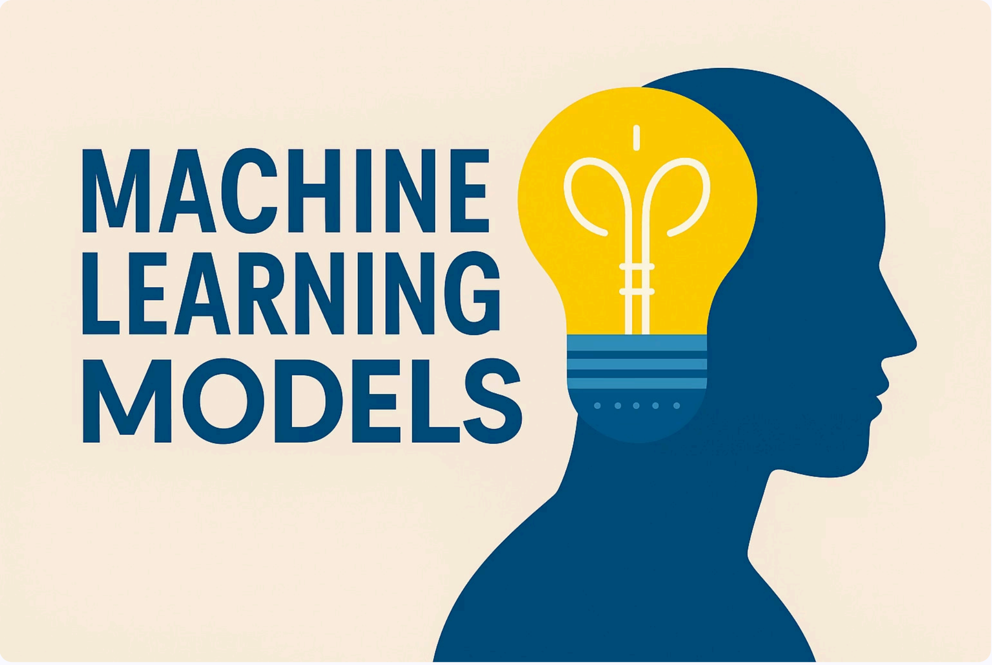
High-performing, gradient-boosted trees

2 **Random Forest**

Captures complex patterns - highlights feature importance.

3 **Easy Ensemble**

Boosting with built-in undersampling to handle class imbalance

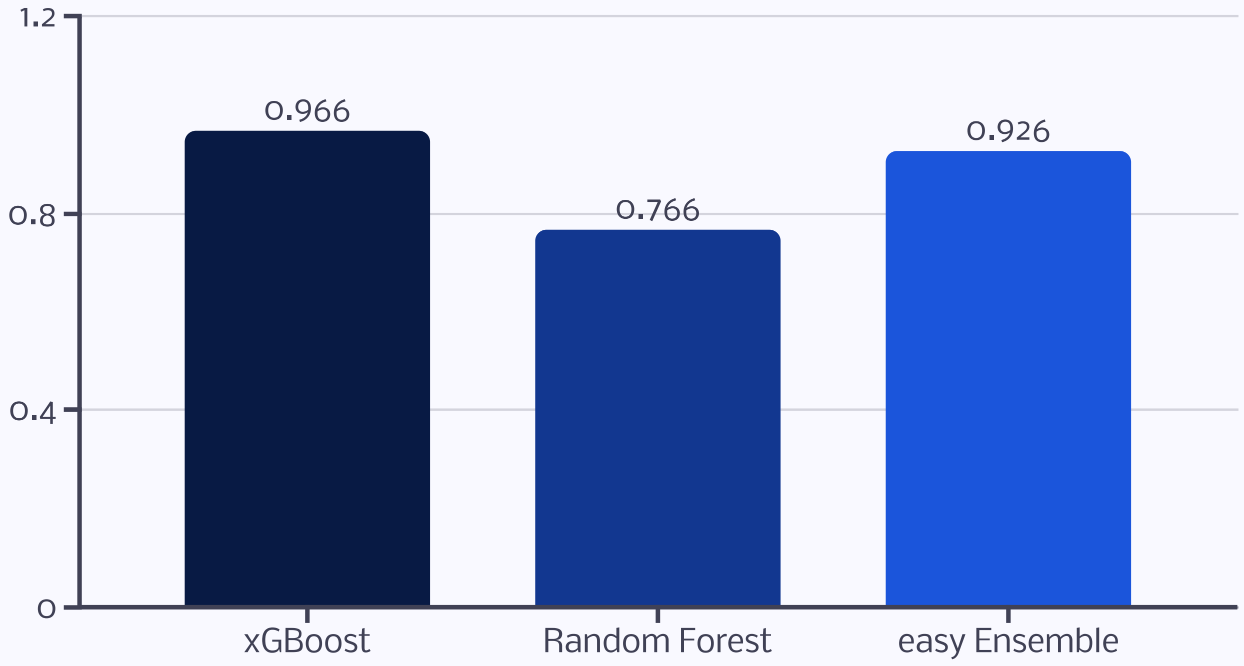


Results & Evaluation

Evaluation Metrics

We prioritized Recall for label=1 (fraud) as our main metric, since the primary goal is to maximize fraud detection and minimize missed fraudulent cases (false negatives).

Better to flag a few false alarms than miss actual fraud.



XGBoost Comparison: Model Performance Summary

- We evaluated how various resampling methods impact XGBoost's ability to detect fraud, focusing on recall, precision, and F1-score for the minority class.
- Each experiment was executed with cross-validation and grid search to identify optimal hyperparameters.
  A detailed log of all resampling experiment results has been saved in an Excel file for easy comparison.

H	G	F	E	D	C	B	A	
weighted_f1	macro_f1	accuracy	support_1	f1_1	precision_1	recall_1	model	1
0.985303837	0.65258042	0.97797452	965	0.316353887	0.188686788	0.978238342	RandomUnderSampler_XGBoost	2
0.989451921	0.699710707	0.985192183	965	0.406918919	0.257103825	0.975129534	XGBoost_scale_pos_weight	3
0.98978671	0.704111712	0.985764414	965	0.415428951	0.264241399	0.970984456	RandomOverSampling_XGBoost	4
0.986795596	0.664317664	0.980668322	965	0.338444485	0.205935252	0.949222798	SMOTE_RandomUnderSampler_XGBoost	5
0.986763609	0.663869663	0.980614338	965	0.337576093	0.205341113	0.948186528	SMOTE_XGBoost	6
0.986763609	0.663869663	0.980614338	965	0.337576093	0.205341113	0.948186528	SMOTE_TomekLinks_XGBoost	7
0.998858498	0.942993639	0.998898726	965	0.886540601	0.956782713	0.825906736	XGBoost_baseline	8

Threshold Optimization

we conducted a **threshold tuning experiment** to better balance **precision and recall**.

Rather than relying on the default threshold of 0.5,
we evaluated multiple thresholds and selected the one that yielded the **highest F1-score**,
ensuring a good trade-off between catching fraud and avoiding false alarms.

Final Results at Best Threshold (0.90):

- **Precision**: 0.4
- **Recall**: 0.94
- **F1-Score**: 0.5611

This adjustment significantly improved precision while maintaining high recall.

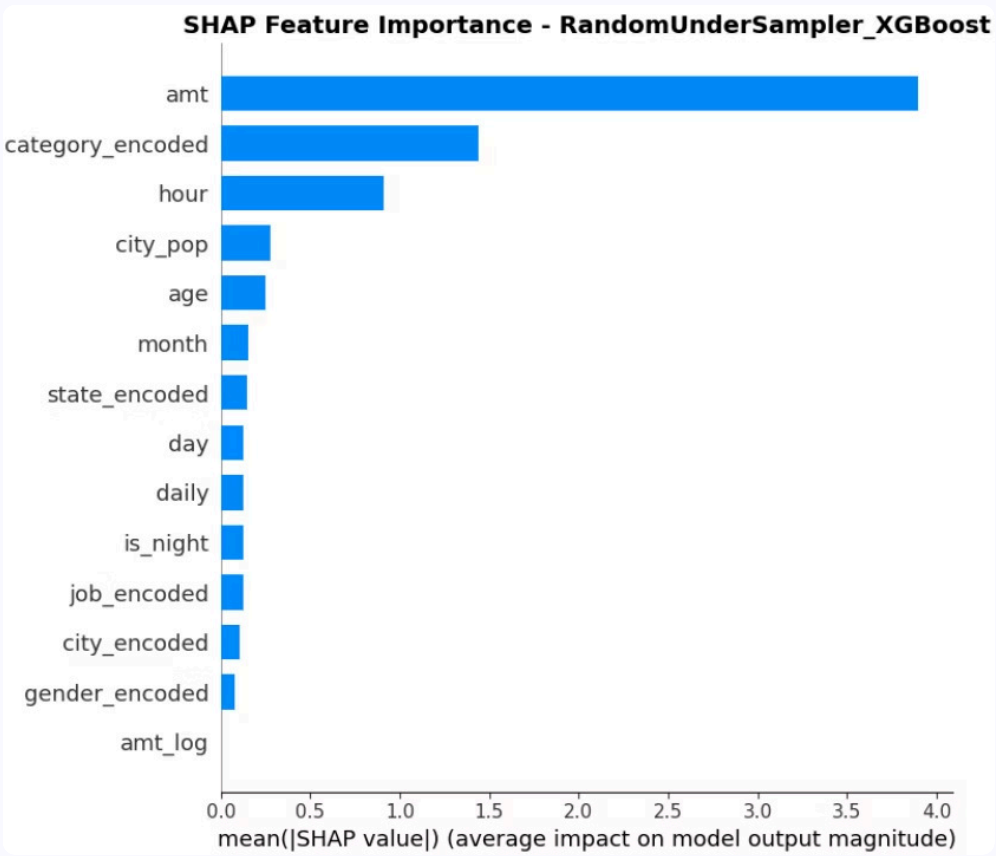
Explainability with SHAP – Global View

Why Explainability Matters

Understand why our fraud - detection model makes each decision–both in general (Global) and for individual cases (Local).

Global Feature Importance

- Top drivers:
  - **amt** (transaction amount)
  - **category_encoded** (transaction category)
  - **hour**

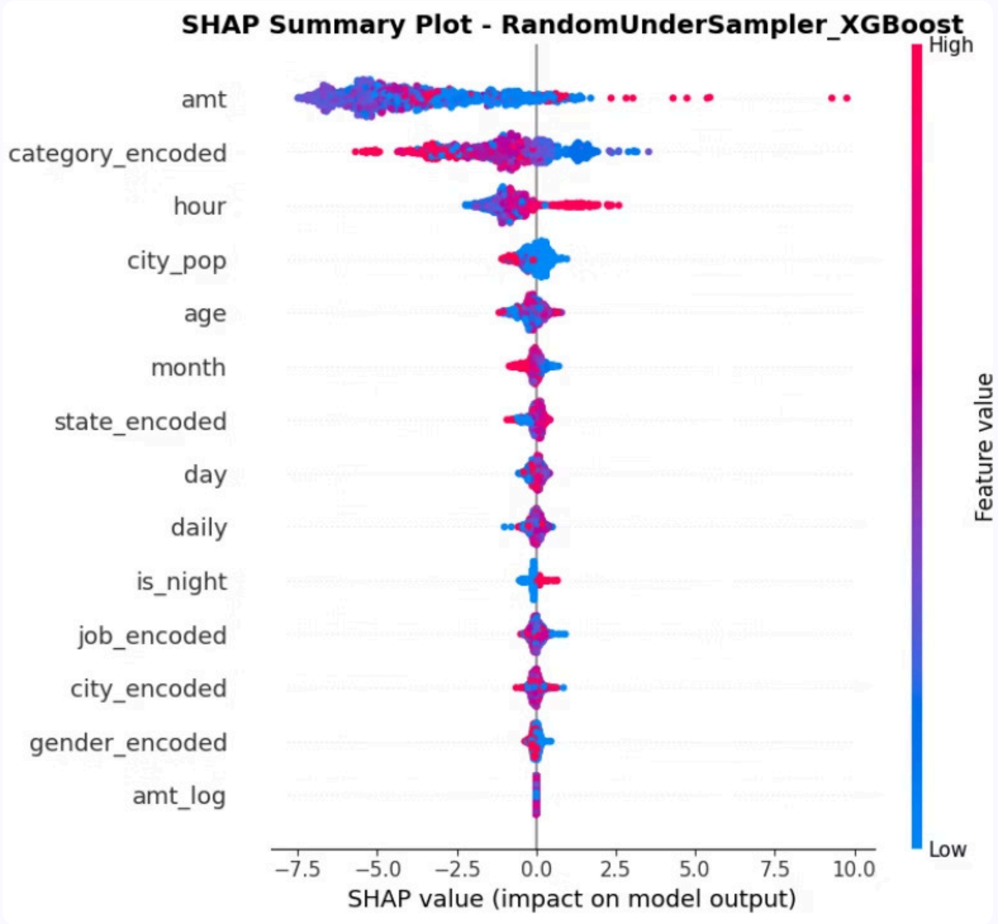


Global Feature Effects

Each dot = 1 transaction

🔴 = High value | 🔵 = Low value

🔴 = High feature value → increases SHAP (fraud risk)



SHAP Summary & Takeaways

📌 Top 3 most important features:

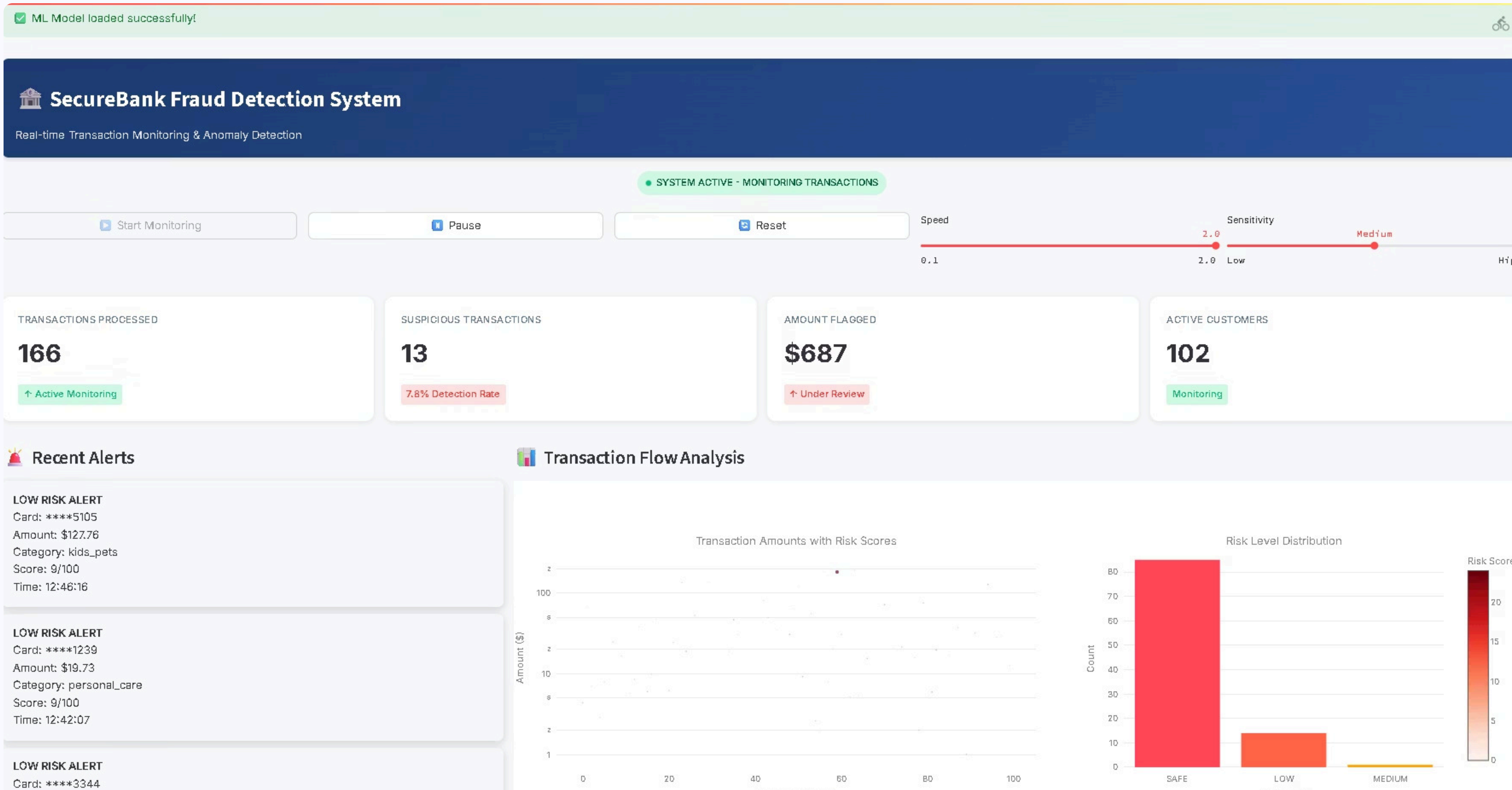
- amt, category_encoded, hour

📈 Features that increase fraud probability:

Features	fraud probability
amt	+6.2
category_encoded	+0.7
hour	+0.6

💡 Insight: Large transactions, specific categories, and late hours are strong fraud indicators according to SHAP.

Live Demo - Streamilit



Live Demo - Streamilit

Live Transaction Feed

Timestamp	Card	Amount	Category	Risk Score	Risk Level
2020-06-21 12:53:48	****9004	$6.85	travel	0/100	SAFE
2020-06-21 12:53:58	****7295	$80.80	home	3/100	SAFE
2020-06-21 12:54:15	****9996	$5.61	shopping_pos	8/100	LOW
2020-06-21 12:54:28	****8698	$7.47	shopping_pos	8/100	LOW
2020-06-21 12:54:44	****7873	$43.47	home	0/100	SAFE
2020-06-21 12:54:52	****2195	$8.32	personal_care	2/100	SAFE
2020-06-21 12:54:55	****4549	$37.00	personal_care	8/100	LOW
2020-06-21 12:55:10	****1789	$9.76	food_dining	2/100	SAFE
2020-06-21 12:55:14	****9808	$75.99	home	0/100	SAFE
2020-06-21 12:55:19	****2684	$26.54	kids_pets	8/100	LOW

Conclusion: Protecting Our Customers and Assets

Strategic Impact

Reduced fraud losses and enhanced customer trust.

Future Plans

Implement real-time detection and continuous model updates.

Commitment

Proactive fraud prevention ensures secure banking.