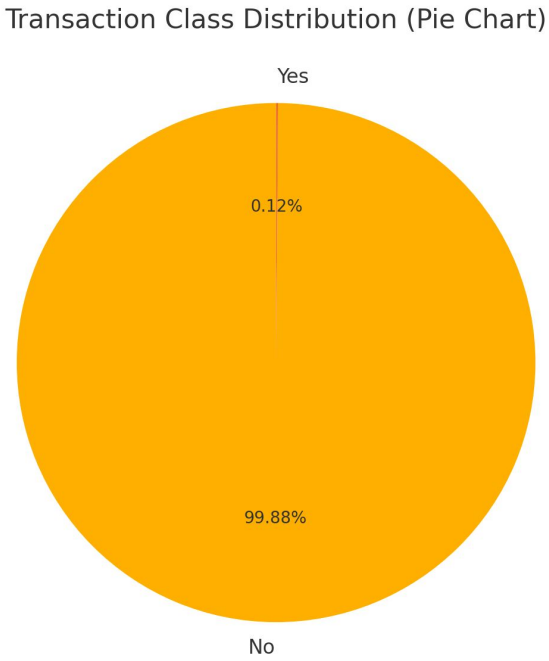
Transaction Fraud Detection - Demo Results

Hemanth Chebiyam

The Challenge - Data & Business Context

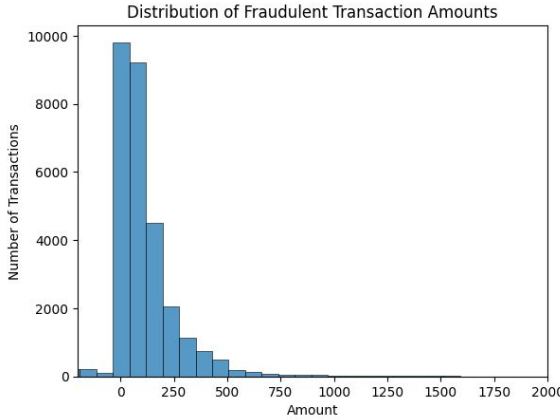
Objective: Design and implement ML models to detect fraudulent credit card transactions on the IBM Credit Card Transactions dataset

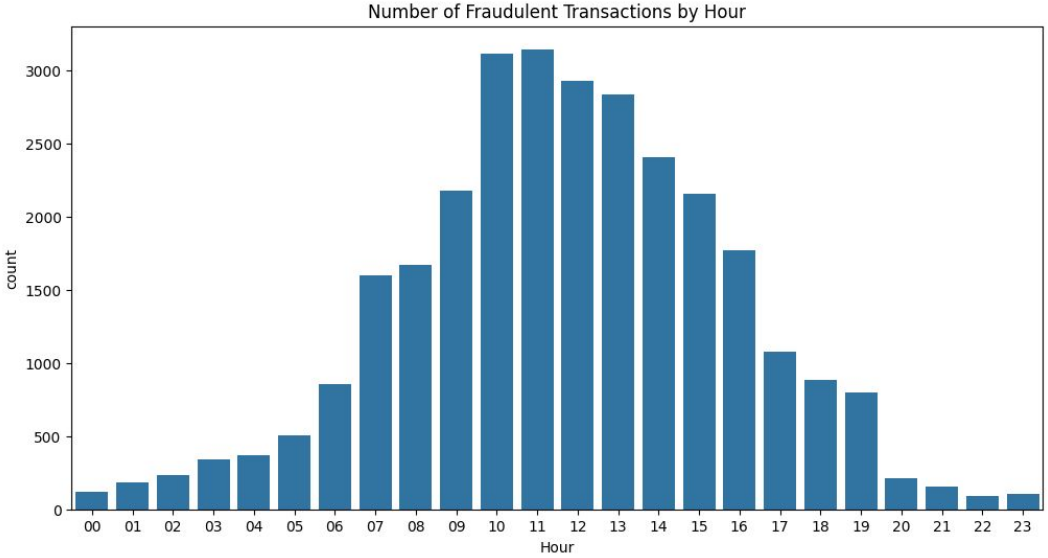
- Size: 24,386,900 transaction records
- Time Period: Multi-year historical data (2002-2010)
- Fraud Rate: 0.12% (29,757 fraud cases) - Highly imbalanced
- Key Features: User, Card, Amount, Merchant info, Location, Time, Transaction method
- Challenge: Extreme class imbalance requiring specialized sampling techniques

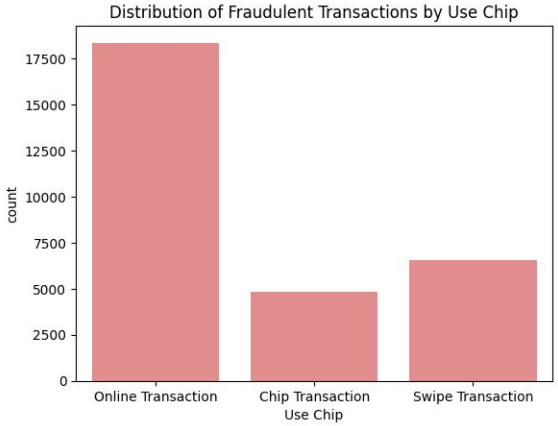


Data Distribution Highlights

- Majority of fraudulent transactions: $0-$250 range
- Peak fraud activity: 10-11 AM local time
- High-risk locations: Italy, Algeria, Haiti, Mexico
- Most vulnerable: Online transactions vs. chip/swipe







Data Preprocessing Pipeline

Phase 1 - Basic Cleaning:

- Amount: Removed '$' symbols, converted to float
- Time: Extracted Hour feature from time strings
- Target: Converted 'Yes/No' to binary (1/0)
- Missing Values: Handled null entries in Zip codes

Phase 2 - Feature Engineering (V2 Enhancement):

- Amount Categories: Binned into risk levels (0-50, 50-200, 200-1000, 1000+)
- Time Risk Score: High risk (0-6 AM), Medium risk (10-11 PM), Low risk (other)
- Online Transaction Flag: Binary indicator for online vs. physical transactions

Sampling Strategy:

- Target: 20% fraud rate for balanced training
- Method: RandomUnderSampler to address class imbalance
- Final training set: ~104K samples (V2), ~28K samples (V1)

Model Development Strategy

Why Two Models?

- Model V1 (Random Forest): Establish baseline performance with core features
- Model V2 (XGBoost): Advanced algorithm + enhanced features for improved detection

Model V1 - Foundation:

- Algorithm: Random Forest (200 trees)
- Features: Year, Hour, Amount, Use Chip, Day of Week, Merchant Name, MCC
- Preprocessing: StandardScaler, Binary Encoding
- Focus: Prove concept feasibility

Model V2 - Enhancement:

- Algorithm: XGBoost (200 estimators)
- Features: V1 features + Amount Categories + Time Risk + Online Flag
- Preprocessing: RobustScaler (better outlier handling)
- Focus: Improve feature engineering

Model Performance Comparison

Evaluation Results

Metric	Model 1 (Random Forest)	Model 2 (XGBoost)	Improvement
Accuracy	94.0%	95.4%	+1.4%
Precision	93.0%	94.0%	+1.0%
Recall	78.0%	82.4%	+4.4%
F1-Score	85.0%	87.7%	+2.7%

Business Impact:

Model V2 would detect ~320 additional fraud cases per 10,000 transactions

Feature Importance Insights

Merchant Type: Certain business categories (gas stations, online retailers) higher risk

Transaction Amount: Both very high and very low amounts suspicious

Online Vulnerability: Online transactions 3x more likely to be fraudulent

Time Patterns: Early morning transactions significantly riskier

Model Deployment

Local Development & Demo:

Framework: FastAPI for REST API development

Endpoints: 3 dedicated endpoints (/predict/v1, /predict/v2, /predict/compare)

Some Test Scenarios:

"The Big Purchase" - Amount Category Showcase

POST http://localhost:8080/predict/v2

```
{
  "Amount": 3500.0,
  "Hour": 18.0,
  "MCC": 5732,
  "Merchant_Name": 44444,
  "Year": 2021,
  "Is_Online": 1
}
```

Expected: amount_category=3, HIGH risk due to large online purchase

"The Midnight Snack" - Time Risk Focus

POST http://localhost:8080/predict/compare

```
{
  "Amount": 15.99,
  "Hour": 1.0,
  "MCC": 5814,
  "Merchant_Name": 22222,
  "Year": 2021,
  "Is_Online": 0,
  "Use_Chip_0": 1,
  "Use_Chip_1": 0,
  "Day_of_Week_0": 0,
  "Day_of_Week_1": 0,
  "Day_of_Week_2": 1
}
```

Expected: V2 shows time_risk=2, V1 focuses on small amount

All Risk Factors

POST http://localhost:8080/predict/v2

```
{
  "Amount": 3500.0,
  "Hour": 18.0,
  "MCC": 5732,
  "Merchant_Name": 44444,
  "Year": 2021,
  "Is_Online": 1
}
```

Expected: V2 model HIGH risk, strong agreement