# Credit Card Fraud Detection

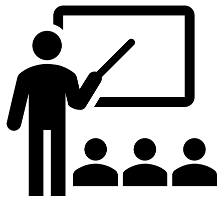
Capstone Two Project

Data Science Foundations Program

Fahad Ali

# Problem Statement

- Credit card fraud causes significant financial losses.
  - Fraudulent transactions are rare and difficult to detect.
  - Class imbalance makes traditional accuracy misleading.
  - Goal: Build a model to accurately detect fraudulent transactions.



# Dataset Overview

- European credit card transactions (September 2013).
  - Approximately 284,807 total transactions.
  - Only about 0.17% are fraudulent.
  - Features are anonymized using PCA (V1–V28).
  - Target variable: Class (0 = Non-Fraud, 1 = Fraud).

# Project Workflow

- Data Wrangling
- Exploratory Data Analysis (EDA)
- Preprocessing
- Modeling
- Evaluation and Recommendations

# Data Wrangling

- Loaded and inspected the dataset.
  - Verified data types and structure.
  - Checked for missing and duplicate values.
  - Identified special variables such as Time and Amount.
  - Created a clean dataset for analysis.

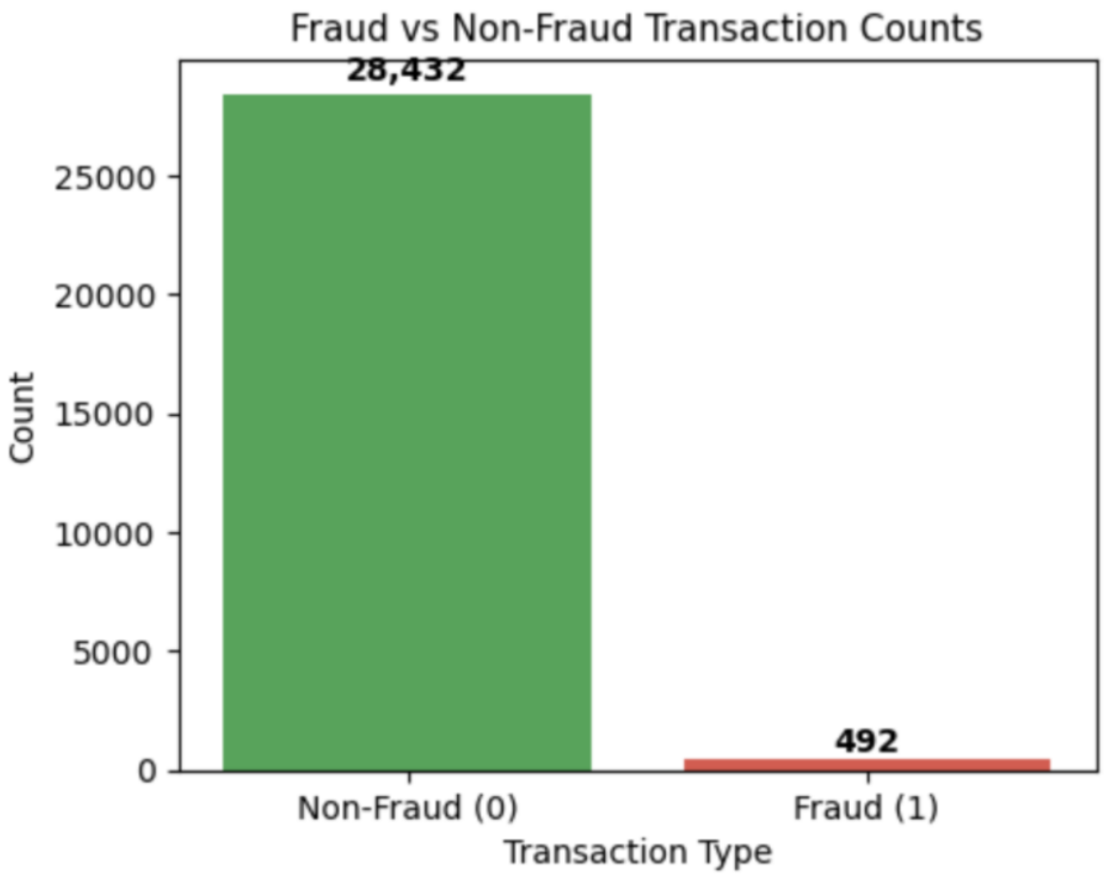
**Data Dictionary**

| Column Name | Type | Description | Notes / Range |
|---|---|---|---|
| **Time** | Numeric | The number of seconds between this transaction and the first transaction in the dataset. | Ranges from 0 to about 172,000 seconds (around 2 days). |
| **V1 – V28** | Numeric (anonymized) | Features created using **Principal Component Analysis (PCA)** to hide sensitive details. Each represents patterns or combinations of original transaction data such as user behavior, location, or card usage. | Can be positive or negative values. Exact meanings are unknown. |
| **Amount** | Numeric (currency units) | The transaction amount in the original currency (e.g., Euros). | Ranges from very small to very large amounts. Often right-skewed (many small, few large). |
| **Class** | Categorical (0 or 1) | Target variable: **0 = normal transaction**, **1 = fraudulent transaction.** | Highly imbalanced — frauds make up less than 1% of the data. |

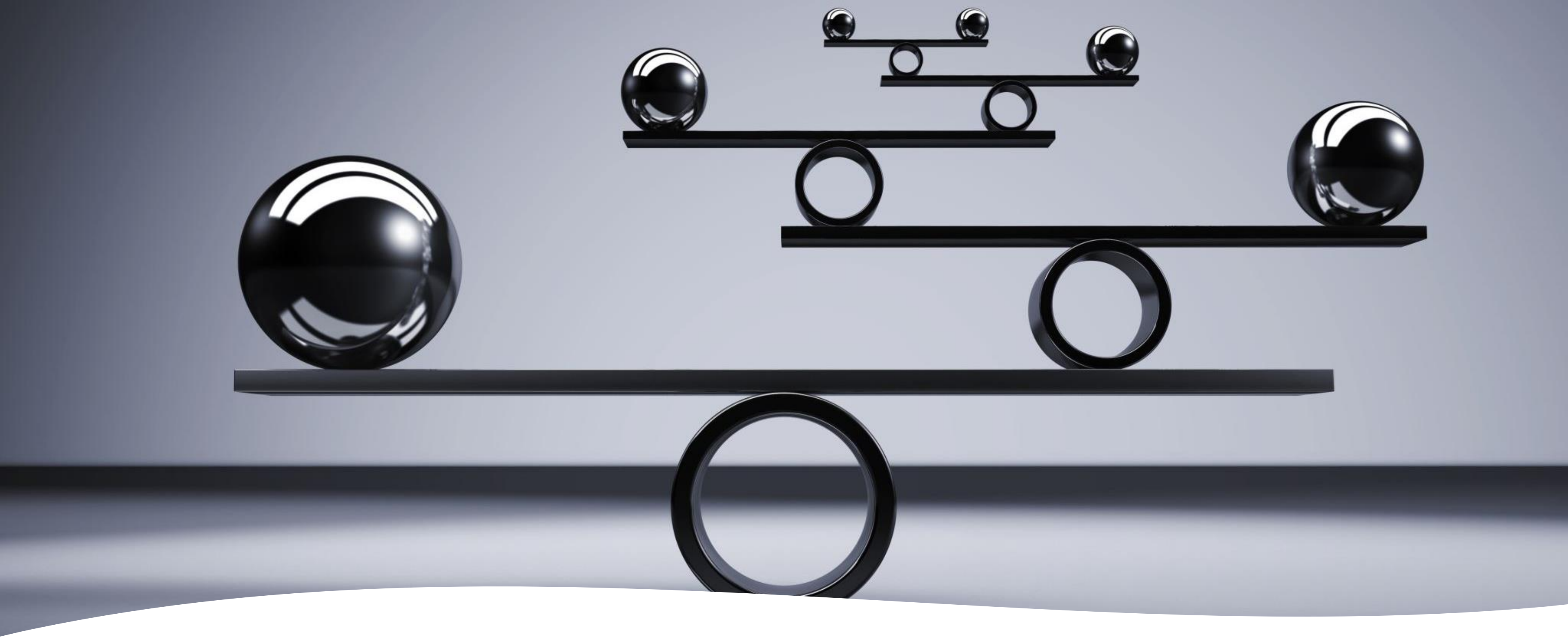
Dataset has 30 columns – 1 for time, 28 PCA (Principal Component Analysis) features, 1 for transaction amount, 1 for class

# Exploratory Data Analysis (EDA)

- Fraud transactions make up less than 2% of the data.
  - Extreme class imbalance observed.
  - Transaction amounts are highly skewed.
  - Outliers present in several features.

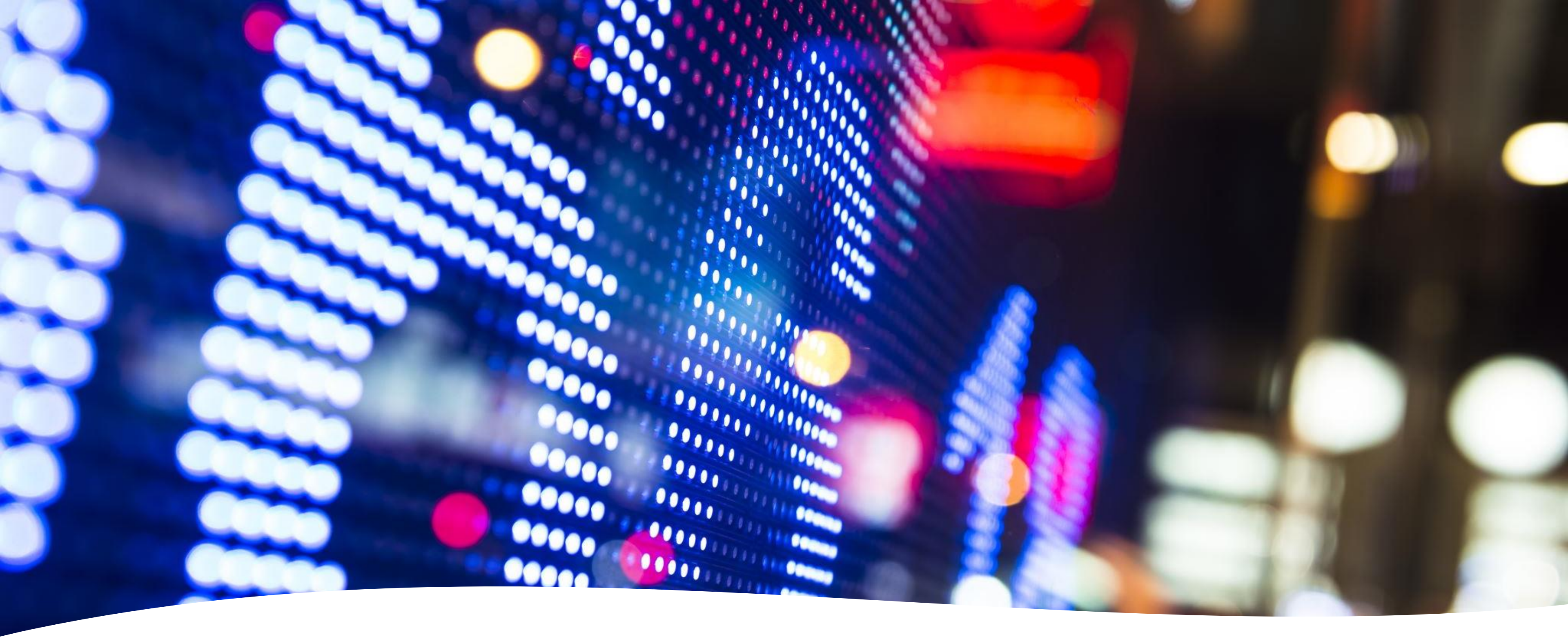



Only ~ 1.7% of transactions are fraud; the others are normal.



# Key EDA Insights

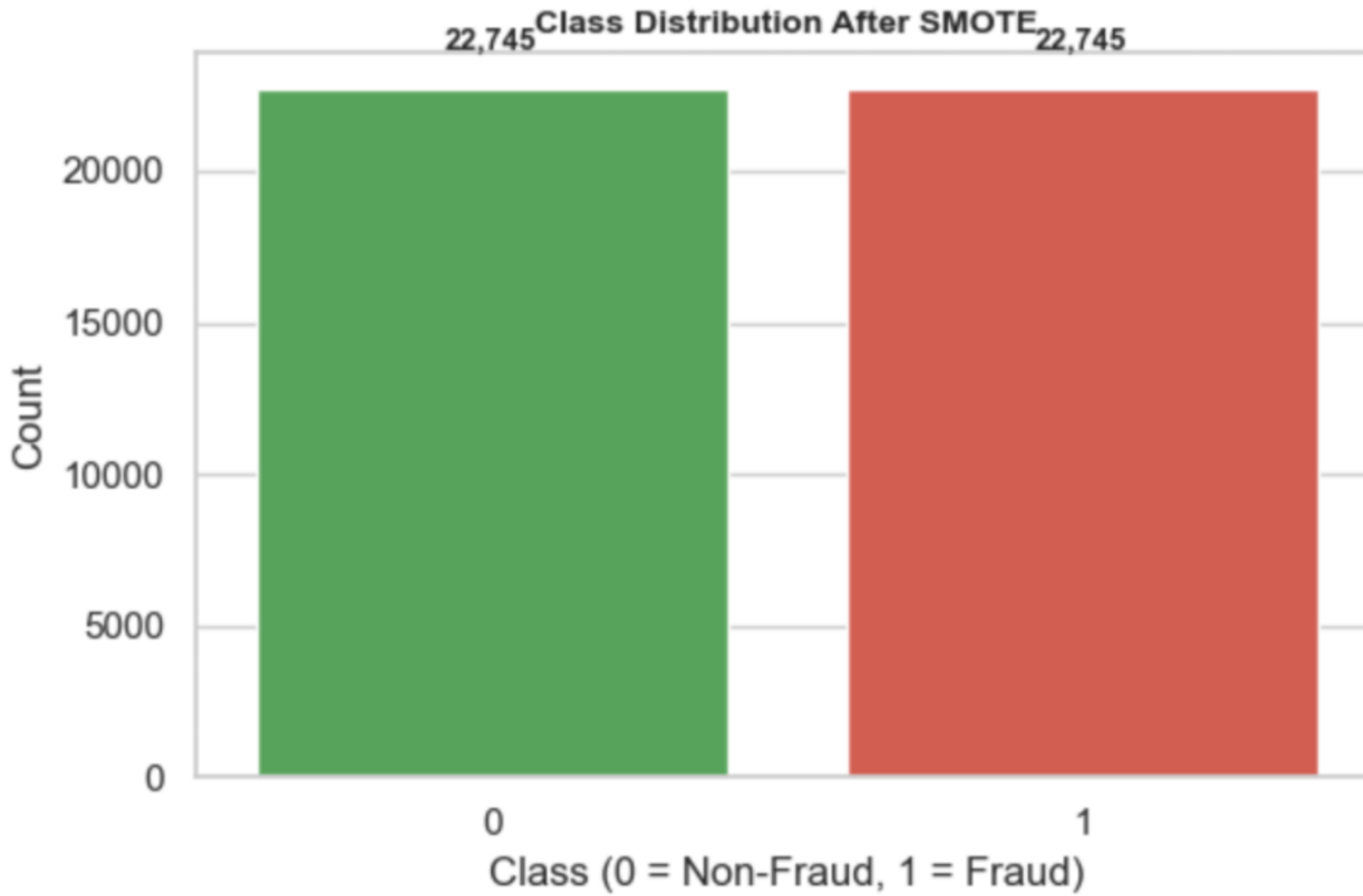
- Class imbalance must be addressed before modeling.
  - Fraud occurs more often in lower to mid-range transaction amounts.
  - PCA features limit interpretability but reveal patterns.
  - Precision and Recall are more meaningful than accuracy.



# Preprocessing

- Split data into training and testing sets.
  - Standardized numeric features using StandardScaler.
  - Handled the Time variable separately.
  - Prevented data leakage by fitting transformations on training data only.

```
Before SMOTE: Counter({0: 22745, 1: 394})
After SMOTE: Counter({0: 22745, 1: 22745})
```




The dataset is now balanced with a 1:1 ratio. (50/50).

# Handling Class Imbalance

- Applied SMOTE to the training data.
  - Balanced fraud and non-fraud classes.
  - Improved the model’s ability to learn fraud patterns.

# Models Built

- Logistic Regression (baseline model).
- Random Forest Classifier.
- XGBoost Classifier.
  - All models evaluated using the same metrics for fair comparison.

# Evaluation Metrics

- Precision
  - Recall (most important for fraud detection)
  - F1-Score
  - ROC-AUC
  - Precision-Recall AUC

# Model Performance Comparison

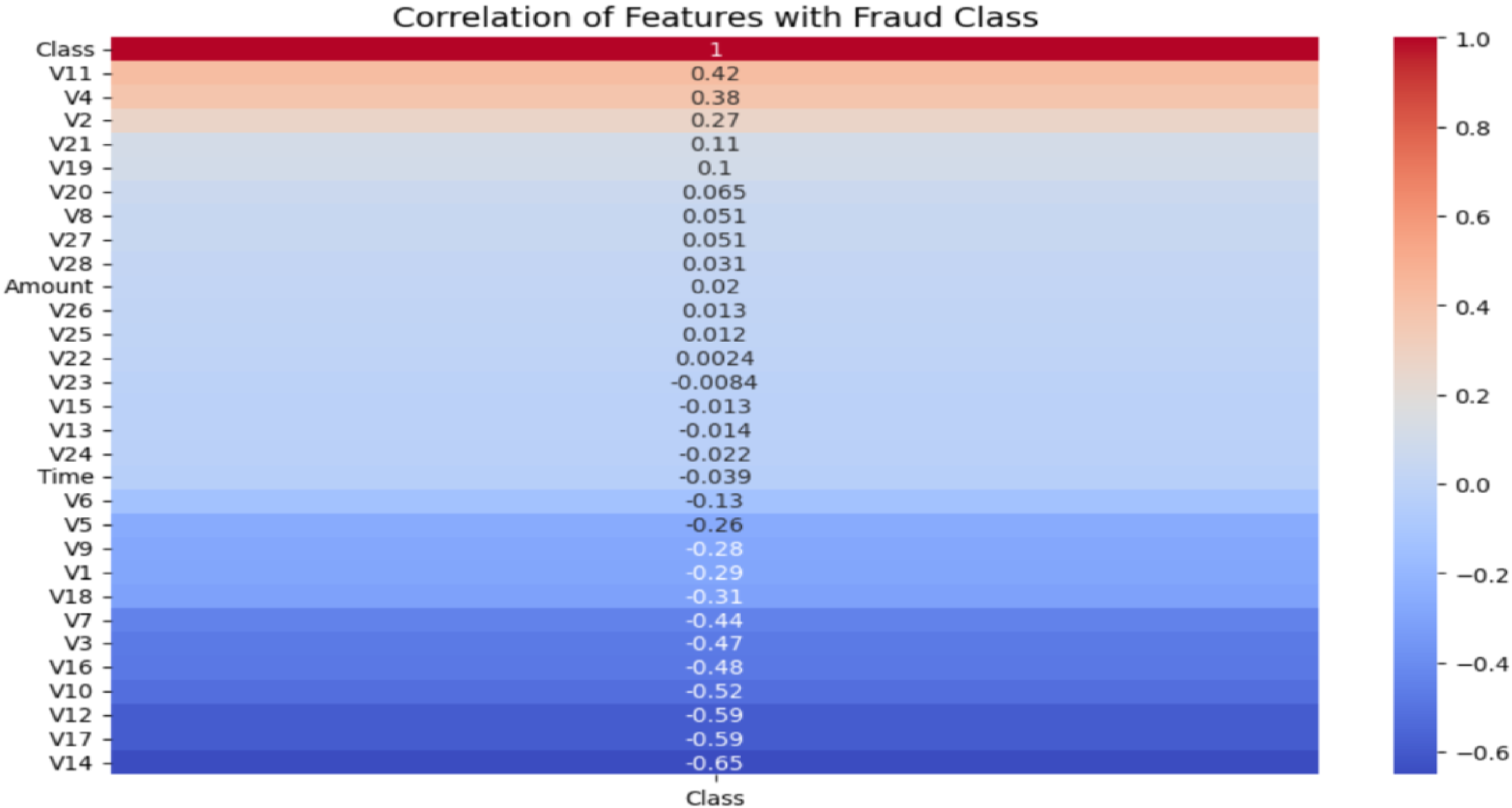
- Logistic Regression provided baseline performance.
  - Random Forest improved recall and balance.
  - XGBoost achieved the best overall results.

# Best Model – Random Forest

- Highest Recall and Precision-Recall AUC.
  - Strong performance on rare fraud cases.
  - Lowest false negative rate.
  - Selected as the final model.

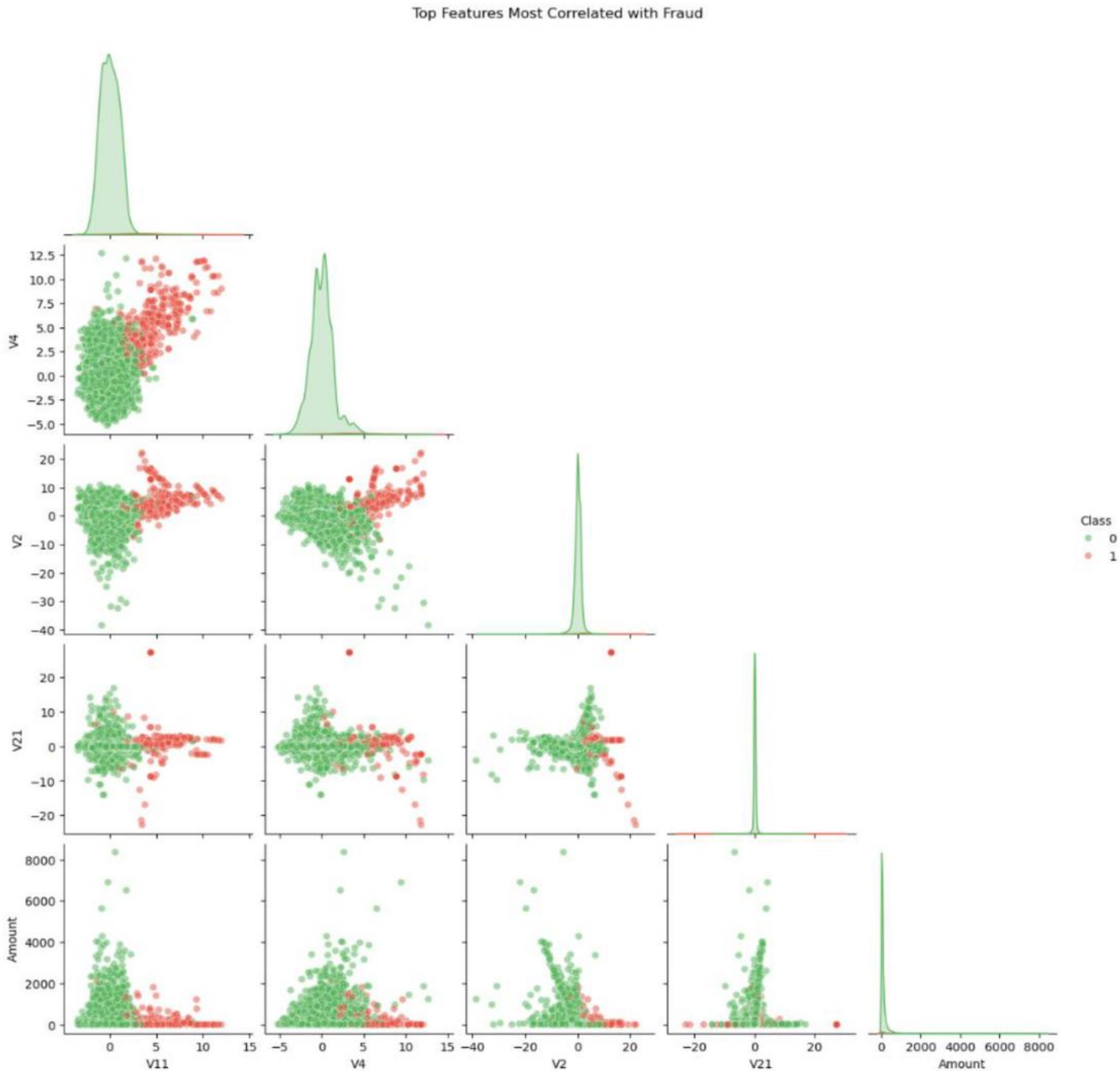
# Feature Importance

- Certain PCA components strongly influence fraud detection.
  - Ensemble models capture complex feature interactions.
  - Feature importance improves interpretability.




```
Top 10 features correlated with Class:
 Class    1.000000
V11      0.423192
V4       0.379510
V2       0.270344
V21      0.109878
V19      0.104997
V20      0.065156
V8       0.051324
V27      0.050756
V28      0.031038
Name: Class, dtype: float64
```

- Features like V11, V4, V2, and V21 show up in different areas of the data compared to the normal transactions.

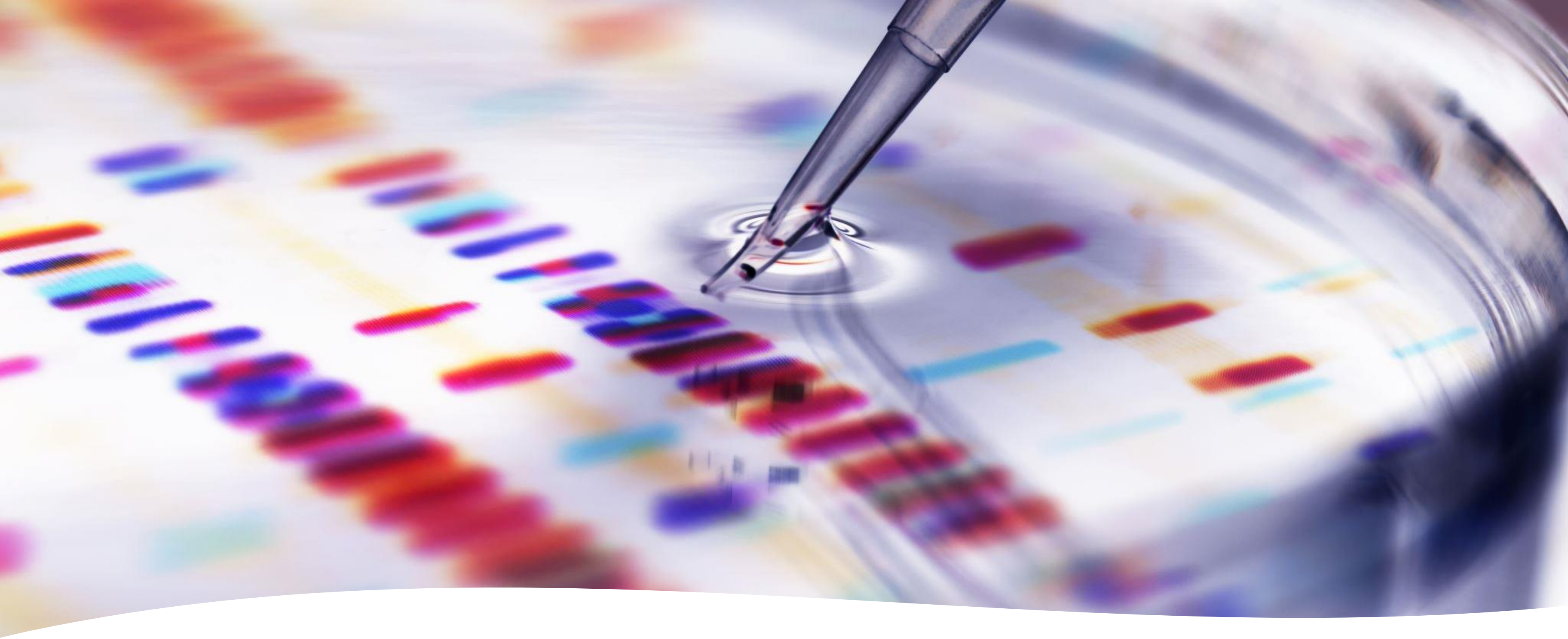



Recommendations

- Deploy  Random Forest as the fraud detection model.
  - Tune the classification threshold based on business risk.
  - Use model insights to support fraud analysts.

# Conclusion

- Applied the full data science workflow.
  - Addressed extreme class imbalance.
  - Compared multiple machine learning models.
  - Random Forest performed best for fraud detection.
  - Project provides actionable business insights.



# Future Work

- Further hyperparameter tuning.
  - Additional feature engineering.
  - Explore anomaly detection methods.
  - Develop a real-time fraud detection pipeline.

# Thank You

Questions?