Credit Card Fraud Detection: *A Machine Learning Approach*

Presented by Riché FLEURINORD – Akademi Education (2025)

Overview

- **Context**: Credit card fraud costs millions of dollars and damages customer trust.
- **Objective**: Use transaction data to identify fraudulent transactions.
- **Method**: Exploratory analysis, data preparation, and classification models.
- **Expected Impact**: Reduce financial losses and improve customer safety.

Business Understanding

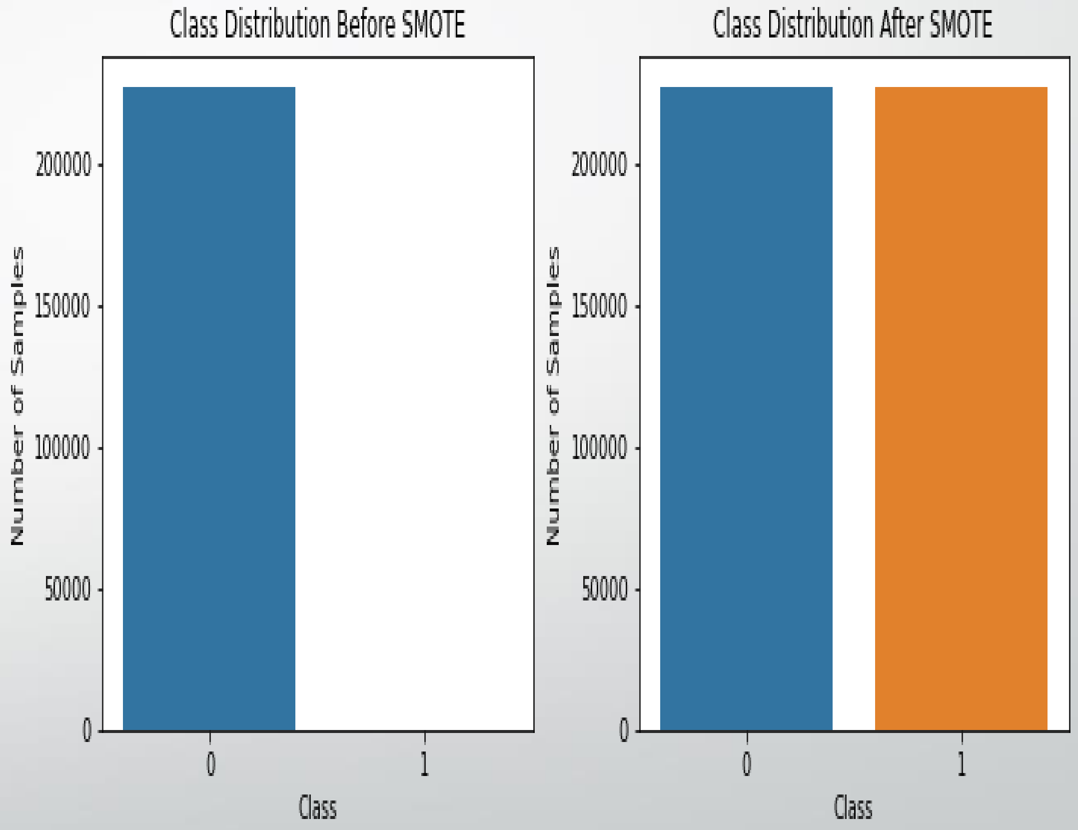
- **Problem**: How can we automatically detect fraud in real time?
- **Stakeholders**: Banks, risk management teams, customers.
- **Benefits**:
  - Reduced financial losses
  - Fewer false positives (legitimate customers not wrongly blocked)
  - Increased customer confidence

Data Understanding

- **Data Source**: Kaggle/Credit Card Fraud Detection/creditcard.csv (284,807 records, 492 frauds → 0.17%).
- **Key Features**:
- **Amount = transaction amount**
- **Time =** time sequence
- **V1-V28 =** anonymized variables (PCA components)
- **Main Challenge**: Highly **imbalanced dataset** (fraud ≈ 1 in 600 transactions).

Data Preparation

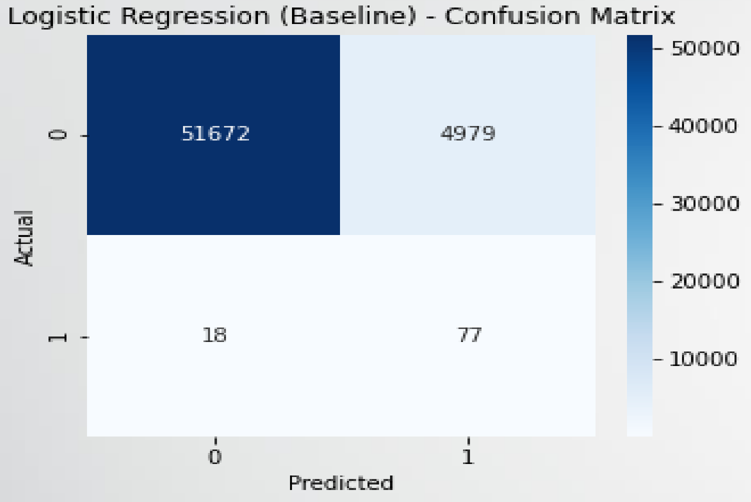
- **Key Steps:**
- Removed duplicates (1,081 dropped).
- Normalized variables for comparison.
- Split into training and testing sets.
- Applied SMOTE to balance fraud vs. non-fraud (synthetic frauds added).
- **Impact:** Models can learn fraud detection more effectively.

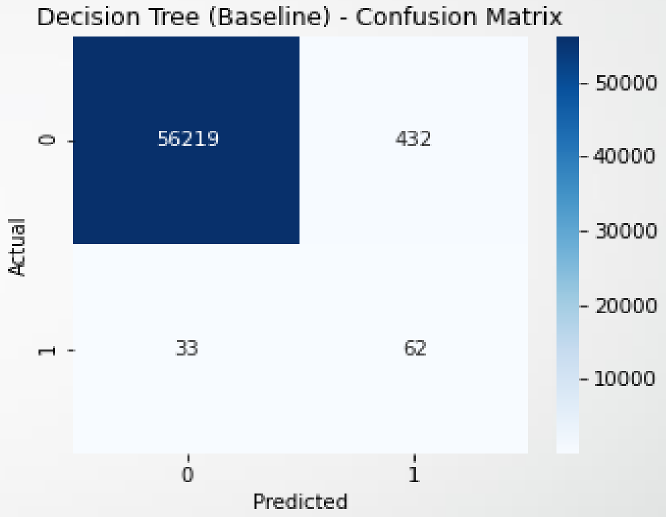


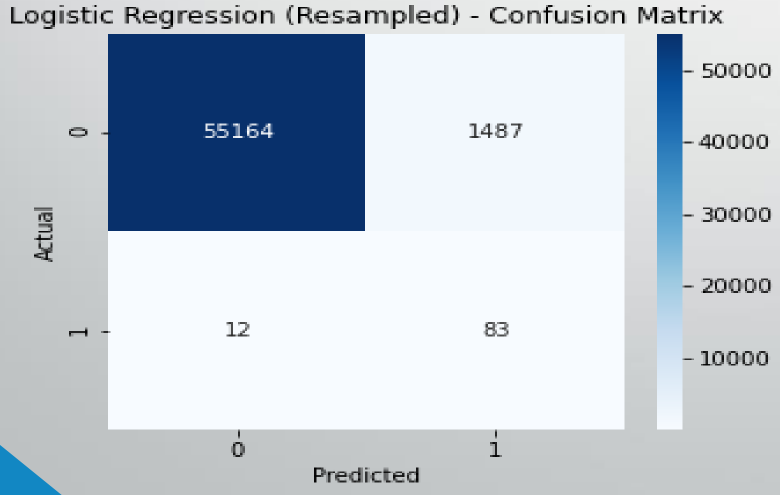
Modeling Approach

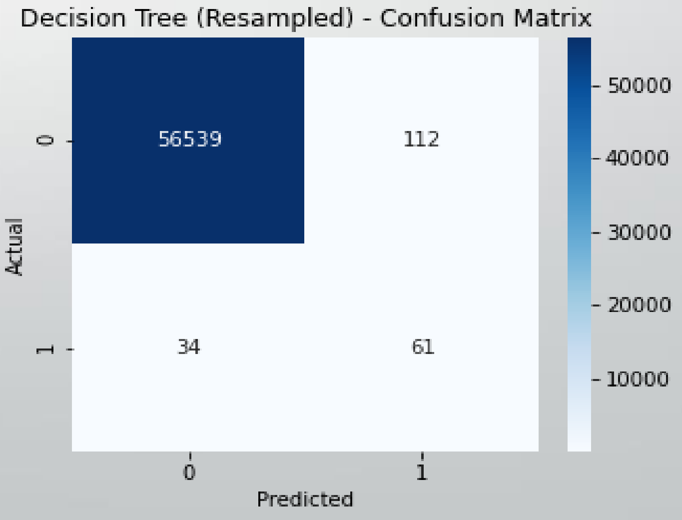
- **Models Tested:**
  - Logistic Regression
  - Decision Tree
- **Two Scenarios:**
  - Baseline (imbalanced)
  - Resampled (balanced with SMOTE)

Evaluation


Logistic Regression (Baseline) - Confusion Matrix
51672
4979
18
77
Actual
Predicted
0
1
50000
40000
30000
20000
10000


Decision Tree (Baseline) - Confusion Matrix
56219
432
33
62
Actual
Predicted
0
1
50000
40000
30000
20000
10000


Logistic Regression (Resampled) - Confusion Matrix
55164
1487
12
83
Actual
Predicted
0
1
50000
40000
30000
20000
10000

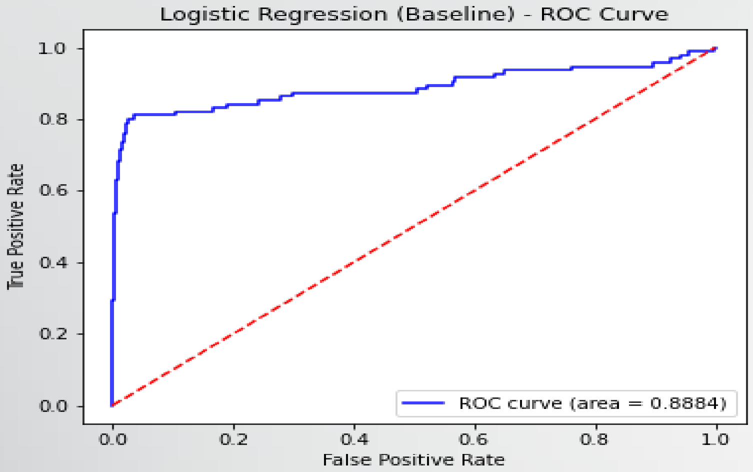

Decision Tree (Resampled) - Confusion Matrix
56539
112
34
61
Actual
Predicted
0
1
50000
40000
30000
20000
10000

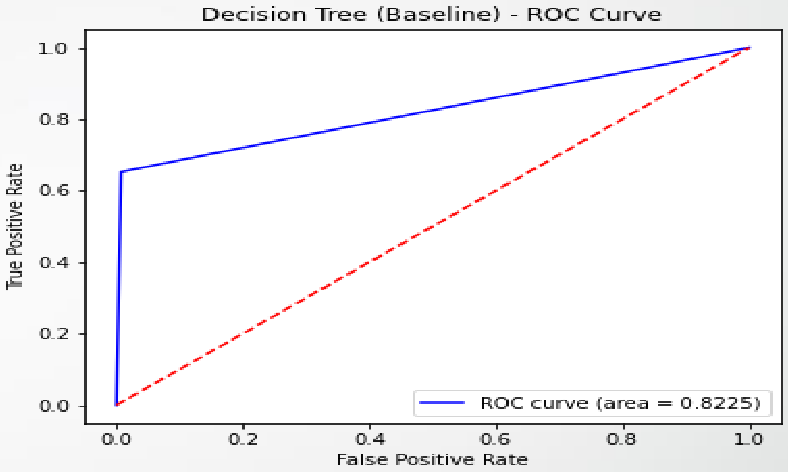
Evaluation Metrics Explained

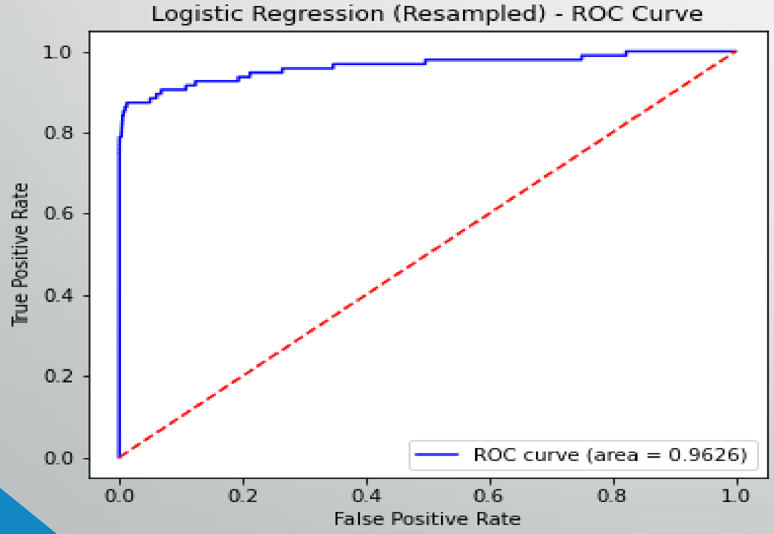
- **Accuracy alone is misleading** (~99% but misses rare frauds).
- **Key Metrics:**

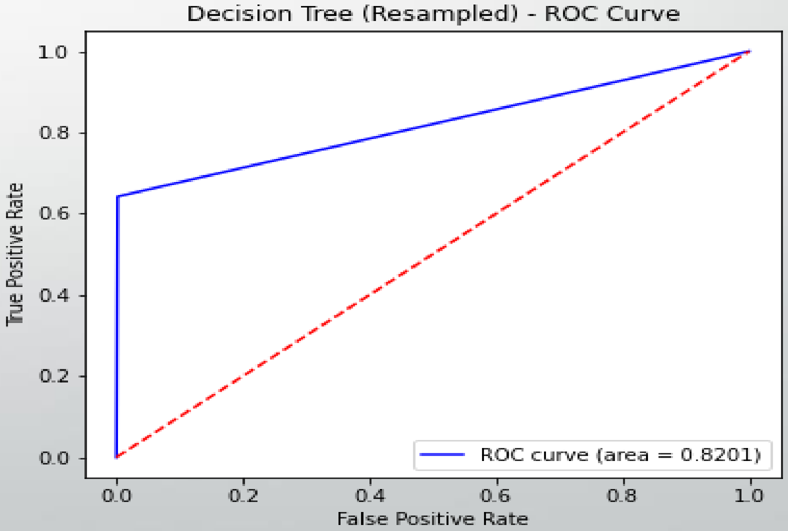
- **Precision =** % of flagged frauds that are truly frauds.
- **Recall =** % of all frauds that the model successfully detected.
- **ROC-AUC =** overall performance score.

Evaluation Metrics Explained









Results Comparison

Model	Accuracy	Recall (Fraud)	Precision (Fraud)	ROC-AUC
Logistic (Baseline)	91%	81%	1.5%	0.89
Decision Tree (Baseline)	99%	65%	12%	0.82
Logistic (Resampled)	97%	87%	5%	0.96
Decision Tree (Resampled)	99.7%	64%	35%	0.82

Key Insights:
Logistic Regression (resampled) → very strong **recall** (detects most frauds).
Decision Tree (resampled) → higher **precision** (fewer false alerts).

Recommendations

- Use a **hybrid approach:**
- Logistic Regression for **recall** (catching most frauds).
- Decision Tree for **precision** (reducing false positives).
- Deploy as a **real-time fraud alert system.**
- Continuously monitor and retrain with new data.

Next Steps

- Collect and analyze more recent data.
- Test advanced models: **Random Forest, XGBoost.**
- Integrate results into a **production banking system.**
- Work with business teams to fine-tune fraud alert thresholds.

Thank You

Together we can make financial transactions safer.

📋**Project**: *3rd Project_ MACHINE LEARNING FUNDAMENTALS - Phase 3*

👨‍🏫**Instructors**: Wedter JEROME & Geovany Batista Polo LAGUERRE

�**GitHub Repository**:
https://github.com/richefleuriord/Ds_Fraud_Detection_Project.git