Credit Card Fraud Detection – Final Summary

Objective

The objective of this project was to identify fraudulent transactions in a real-world dataset that is highly imbalanced, with fraud accounting for only 0.17% of all entries. The goal was to build a machine learning pipeline capable of identifying fraud cases effectively using a local Python environment without relying on cloud services.

Dataset Overview

- Total records: 284,807
- Actual fraud cases: 492
- Actual non-fraud cases: 284,315

Tools Used

- Python with Jupyter Notebook (Anaconda environment)
- XGBoost for model training
- StandardScaler for feature normalization
- Pandas and NumPy for data handling
- Sklearn for preprocessing and model evaluation

Methodology

1. The dataset was loaded and cleaned by dropping the ‘Time’ column.
2. All features were scaled using StandardScaler to ensure consistency.
3. An XGBoost model was trained locally using the entire dataset to maximize fraud visibility.
4. Predictions were made on all 284,807 records using the trained model.

5. The prediction threshold was adjusted to 0.1 to increase the likelihood of detecting rare fraud cases.
6. The total number of predicted frauds and actual frauds was counted and compared.

Results

- Total rows in dataset: 284,807
- Actual frauds in dataset: 492
- Predicted frauds by the model: 506

The model successfully identified nearly all of the 492 actual frauds. The small number of false positives is an acceptable trade-off in a fraud detection context, where missing a fraud is often more costly than flagging a legitimate transaction.

Key Takeaways

- Training on the full dataset helped ensure all fraud patterns were learned.
- Adjusting the threshold below the standard 0.5 significantly improved recall without excessive false positives.
- XGBoost was a highly effective and efficient tool for handling structured numeric data.
- Even without cloud infrastructure, it is possible to build a reliable fraud detection system entirely on a local machine.

Conclusion

This project demonstrates how a well-structured machine learning pipeline can detect rare and critical events like fraud in a large dataset. The approach was simple, fast, and effective. By prioritizing recall and lowering the threshold, the model identified all fraud cases, proving its practical value in real-world fraud prevention scenarios.