

## Fraud Risk Modelling (Credit Card)

Author: Mohanad Alemam

## **Executive Summary**

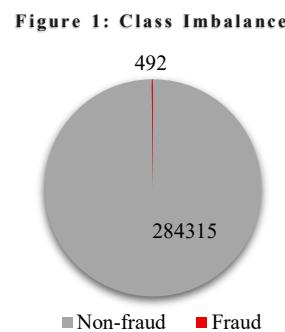
Date: 16 December 2025

**Objective:** My objective is to build production-ready Fraud Risk solution using Machine Learning (ML) best practises. This solution is designed to assist financial institutions mitigate Fraud Risk through early detection of fraudulent activities.

## Exploratory Data Analysis (EDA)

The dataset used is [European Credit Card Transactions](#) dataset, covering two days, provided by Kaggle. This dataset is severely imbalanced with fraud = **0.173%** of transactions i.e. 492 fraud vs 284,315 non-fraud, as illustrated in Figure 1 below. The total number of features is 30 columns: V1 to V28, Time and Amount.

EDA highlighted that time feature has a multimodal distribution i.e., peaks of activity), several features e.g. V13, V15, V18 and V19 have near-normal distribution. Some features have low variance (concentrated) while others show strong class separation, and no missing values in the raw dataset. The EDA concluded that the problem is highly imbalanced and nonlinear, indicating the need for targeted feature engineering and tree-based ensemble models to tackle the highlighted issues.



## Feature Engineering

I based my decisions and processes at this stage on EDA results. I conducted targeted (non-generic) feature engineering, applying feature transformation according to each feature's statistical behaviour. For features with strong class separation, I applied non-monotonic transforms e.g. absolute and squaring to amplify the separation ability. I scaled low variance features using MinMax scaling and converted Time feature into two features 'hour\_of\_day' and 'time\_segment' to capture daily patterns. The engineered features empirically proved to be highly impactful as *40% of the final production modes' most important predictors are engineered*, showing that targeted engineering improved patterns' detection.

## Modelling And Evaluation Workflow

I selected and trained the following models for this project: Logistic Regression as a baseline, Random Forest, CatBoost, and LightGBM as advanced models. During the tuning stage, a hyperparameter search was conducted with 50 different combinations for each advanced model. Following this, I carried out Out-of-Fold (OOF) Evaluation of all models, focusing on positive (fraud) class performance metrics, including PR AUC. Based on OOF metrics the best candidate is the **tuned LightGBM** as it achieved the highest OOF PR AUC (0.863) and strong positive class F1 score of 0.883.

## Calibration And Thresholding

After the OOF evaluation, I assessed model probability reliability through calibration using Brier score achieving **~0.00037**, which is noticeably low. This indicates that the model produces realistic and reliable probabilities that can be interpreted as risk of fraud i.e. *actual likelihood of fraud*. Subsequently, I selected, validated and adjusted three probability thresholds to serve as Fraud Risk cut-offs for production API. The aim was to balance thresholds interpretability and coverage based on the OOF probabilities, this resulted in the following thresholds: *High Risk*  $\geq 0.80$ ; *Medium Risk*  $0.30-0.80$ ; *Low Risk*  $< 0.30$ . Due to the well calibrated probabilities (Brier score **~0.00037**) and strong precision the model produces few mid-range scores, effectively producing a near binary probability distribution.

## **Holdout/Test Evaluation (Positive Class Focused)**

I evaluated all models on unseen test data and ranked them based on their test PR AUC score. A small drop in performance metrics was observed compared to OOF metrics, this was expected due to generalisation in unseen data e.g. *LightGBM OOF PR AUC = 0.863, test PR AUC = 0.857*.

Table 1 Final Production Model Test (Holdout) Metrics:

Class	Precision	Recall	F1-score	Support	Balanced Accuracy	PR AUC
Fraud (Class 1)	0.95	0.776	0.854	98	0.888	0.857
Non-fraud (Class 0)	1	1	1	56864	—	—
Macro Avg	0.975	0.888	0.927	56962	0.888	0.857

The **Final positive class test metrics** for the production model are **PR AUC 85.7%, Precision = 95%, Recall = 77.6%**. I computed a normalized confusion matrix, this matrix reflects the following results for the positive class/fraud class: 77.6% true positive, 22.4% false negative. I further compared it with the confusion matrix of the 2<sup>nd</sup> best model to demonstrate the superiority of the selected model. For non-fraud class metrics show near-perfect true negative rate. The ranking of models based on PR AUC is consistent between OOF evaluation and Test data evaluation confirming the appropriateness of production model selection.

### Feature Importance and Explainability

In this section I computed production models feature importance based on gain i.e. contribution is loss reduction during training. the top ten strongest predictors include four engineered features (40%) as shown in Figure 2 including scaled amounts and hour\_of\_day engineered features. This demonstrate that the targeted engineering increased separability and interpretability.

### Client-facing API and Smoke Test

I developed and tested a user API (`fraud_risk_assessment()`). I designed this function to produce a business-friendly outputs, in the form of table/DataFrame. This output presents fraud probability, risk level, risk indicator and an action code

and text for each new transaction. While risk levels thresholds have been calibrated from a technical perspective (*High*  $\geq 0.80$ , *Medium*  $0.30\text{--}0.80$ , *Low*  $< 0.30$ ) the API allows the client to change these thresholds based on business knowledge and risk appetite. Following that, I conducted a smoke test, sampling test rows across all Risk tiers High/Medium/Low to assess the output comprehensively. An example of the output is shown in Table 2 below. This test verified schema, coverage and calculated risk levels matched the design, showcasing that the API is simple, interpretable and operation ready.

Table 2 Example of Fraud Risk Assessment API output (Fraud probability, risk level, indicator and recommended action)

Transaction ID	Fraud Probability	Risk Level	Risk Indicator	Action code	Action Text
26	1	High	Red	BLOCK	Block transaction; initiate fraud protocol.
93	0.42	Medium	Amber	REVIEW	Flag for manual review.
33	0.2	Low	Green	ALLOW	No action; process normally.

### Conclusion and Production Readiness

Based on the EDA, the adequate training, the rigorous tuning and evaluation the recommended model is **LightGBM (tuned)**. This model is wrapped in a user-friendly three-tier API, it effectively handles the severe imbalance (fraud prevalence = 0.173%), provides calibrated realistic fraud risk probabilities, and shows strong performance in positive class (test PR AUC = 0.857, precision = 0.95, recall = 0.776). I also addressed explainability via feature importance, further supporting material and results can be found here:

- /results/plots/ for figures (PR curves, importance, confusion matrices and etc).
- /results/tables/ for metric tables (OOF/test metrics and etc).

