Fraud Detection — Technical Memo

BAF NeurIPS 2022 · XGBoost + Optuna + SHAP

Eduardo Rdgz-Á

2026-06-13

Context

This document records the design decisions behind the fraud detection model trained on the BAF dataset (NeurIPS 2022), which is made up of 1 million bank transactions with 30 features whose values were transformed to preserve privacy. The positive class (fraud) accounts for about 1.1% of the observations, and that imbalance drives the project's core decisions: the optimization metric, how the imbalance is handled, and the threshold criterion.

Design decisions

Handling class imbalance

The dataset has a roughly 98.9:1.1 ratio between legitimate and fraudulent transactions — about 90 legitimate cases for every fraud. The alternatives considered were SMOTE (synthetic oversampling), random undersampling, and weight adjustment through `scale_pos_weight`.

We chose `scale_pos_weight = 90` for three reasons: it does not change the real data distribution, it is native to XGBoost with no extra steps in the pipeline, and it preserves all of the majority-class information.

Handling missing data

The dataset has no null values. However, several columns contain `-1` values that, according to the paper, are the dataset's convention for "data not available" — not missing values in the statistical sense. The most affected column is `prev_address_months_count`, with 71.3% of records in that state.

No imputation was applied. XGBoost natively reads these values as an implicit "missing" category, and in a fraud context the absence of a value can be a signal in itself.

Encoding categorical variables

The dataset contains nominal categorical variables. The alternative considered was `OneHotEncoder`, which creates one binary column per category.

We chose `OrdinalEncoder` with `unknown_value=-1` for categories not seen in production. For tree-based models like XGBoost, ordinal encoding is enough — the splits do not assume any order between categories, and it avoids the dimensional blow-up that OneHot creates.

Optimization metric

The natural metric in binary classification is AUC-ROC. However, with a positive class of 1%, a classifier that always predicts "legitimate" scores AUC-ROC 0.50 while AUC-PR 0.01 — the random baseline. AUC-PR does not involve true negatives on either of its axes, so the imbalance does not distort the score.

We chose **AUC-PR** as the optimization metric in Optuna and as the main evaluation reference. AUC-ROC is reported as a secondary metric for comparability with the literature.

Hyperparameter search — Optuna

We ran a Bayesian search with TPE (Tree-structured Parzen Estimator) over 50 trials, maximizing AUC-PR under stratified cross-validation (5 folds). The search space:

Hyperparameter	Range
`n_estimators`	[200, 800]
`max_depth`	[3, 8]
`learning_rate`	[0.001, 0.3] log
`subsample`	[0.5, 1.0]
`colsample_bytree`	[0.5, 1.0]
`reg_alpha`	[0.0001, 10.0] log
`reg_lambda`	[0.0001, 10.0] log

Best trial: `n_estimators=581`, `max_depth=3`, `learning_rate=0.065`, `subsample=0.729`, `colsample_bytree=0.543`, `reg_alpha=0.002`, `reg_lambda=3.022`. Best AUC-PR in CV: **0.1778**. The gain over the baseline is reported in the Results section, evaluated on the test set.

Threshold selection — final model (post Optuna)

The default threshold (0.5) is not optimal under severe imbalance. The analysis is run on the probabilities of the final model (XGBoost + Optuna, 50 trials) — not the baseline.

The selection criterion depends on the relative cost of FN versus FP: a missed fraud causes a direct loss (Type II error — FN); a false alarm causes operational friction (Type I error — FP). The threshold is a business decision, not a model one.

Threshold	Precision	Recall	F1	FN	FP
0.50	0.0509	0.8005	0.0957	440	32,921
0.40	0.0408	0.8622	0.0779	304	44,724
0.30	0.0321	0.9089	0.0620	201	60,435
0.20	0.0244	0.9438	0.0476	124	83,107
0.10	0.0175	0.9787	0.0344	47	121,169
0.05	0.0141	0.9964	0.0277	8	154,024

Results

Metrics evaluated on the test set (200,000 observations, threshold = 0.5).

Metric	Baseline	Final (Optuna)	Δ
AUC-ROC	0.8444	0.8991	+6.5%
AUC-PR	0.1238	0.1800	+45.4%

AUC-PR is the optimization metric — the meaningful gain is +45.4%. AUC-ROC is reported as a secondary reference.

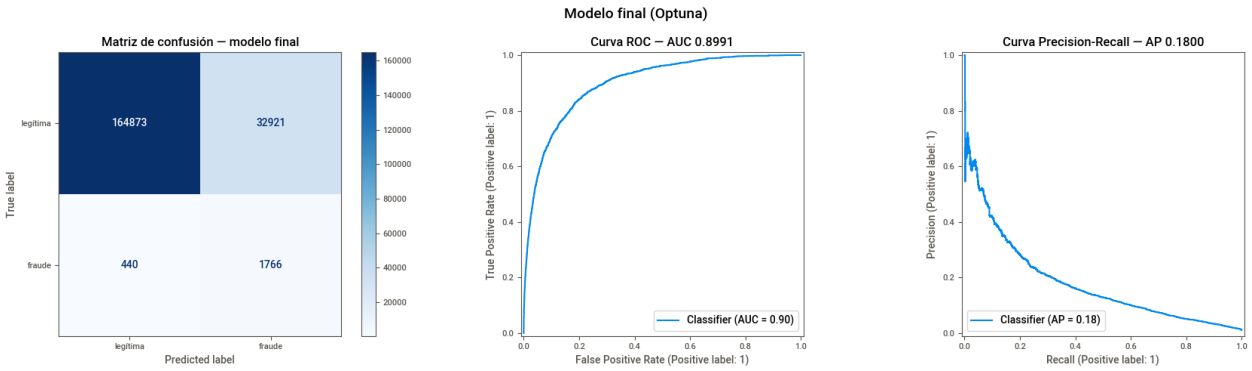

Figure 1: Curvas de evaluación — modelo final (Optuna)

Interpretability — Global importance (SHAP)

We used SHAP (SHapley Additive exPlanations) with `TreeExplainer` over a random sample of 2,000 observations from the test set.

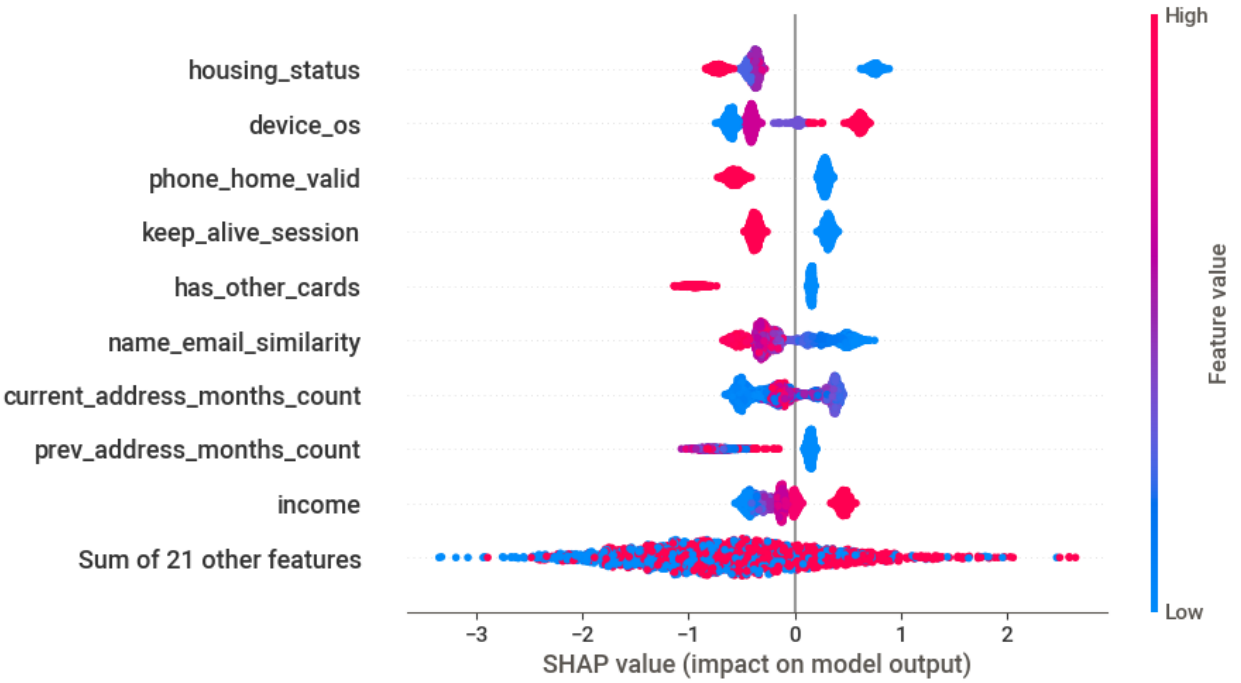

Figure 2: SHAP beeswarm — importancia global de features

Each point is one observation. The color indicates the feature value: red = high value, blue = low value. The horizontal position indicates the direction of the impact on the fraud score.

The features with a clear directional signal are `housing_status` and `device_os` (mean SHAP = 0.51 each): low values of `housing_status` push toward fraud; high values of `device_os` do too. `phone_home_valid` (+0.41), `keep_alive_session` (+0.35), and `has_other_cards` (+0.34) follow the same pattern.

`name_email_similarity` and `current_address_months_count` show a mixed signal: their distributions spread in both directions, which means the impact depends on the specific value of the observation. The cumulative sum of the remaining 21 features contributes a mean SHAP = 2.54 — larger than any individual feature, which suggests that the model spreads predictive information diffusely across the feature space.

Limitations and future work

Synthetic data. BAF NeurIPS 2022 is a synthetic dataset generated from real data. The model may not capture fraud patterns that emerge in real production — distribution drift, coordinated fraud, and new attack vectors not represented in the dataset.

Features with transformed values. The values of the 30 features are transformed to preserve privacy. This limits operational interpretability: it is possible to identify which features matter statistically, but not to translate them directly into actionable business rules without the original mapping.

Static threshold. The saved threshold (0.5) is a neutral reference. In production, the optimal threshold depends on the business's relative FN/FP cost — a parameter that varies by institution and regulatory context.

Future work. Retraining on real data with drift monitoring, threshold optimization under an explicit cost function, and exploring sequential models (RNN/Transformer) to capture temporal patterns across requests from the same device.