

Digital Lending E-Sign Prediction System

Predicts which loan applicants will finish e-signing, so outreach goes where it works.

BUILT WITH

Fintech

Predictive Modeling

Scikit-learn

Feature Engineering

Python

Model Deployment

XGBoost

0.60 to 0.69

ROC-AUC, baseline to winning model

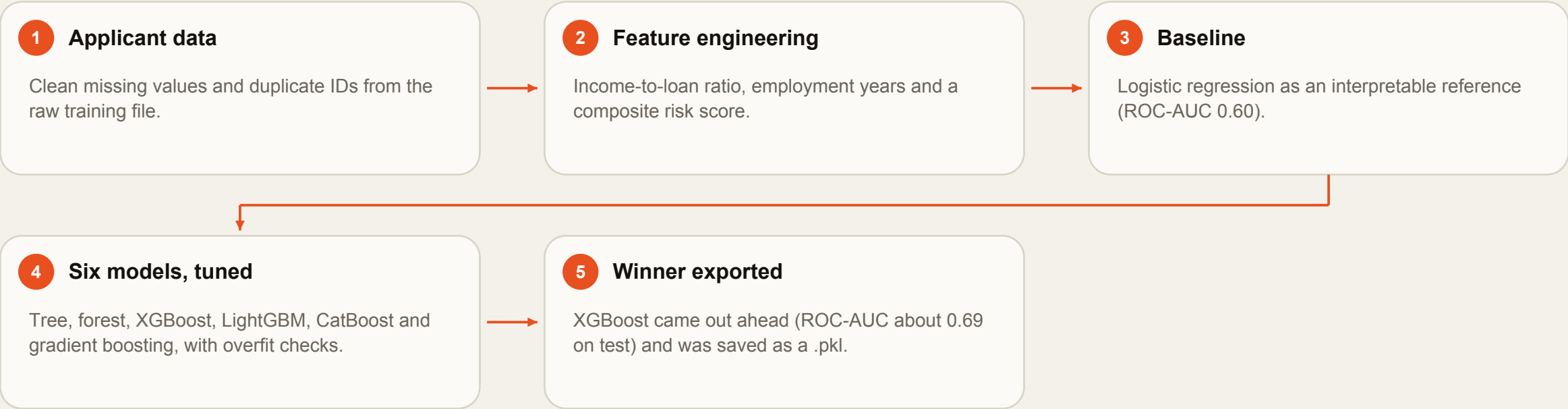
6

models compared and tuned

1

.pkl file ready for integration

How it works



WHAT EACH PART DOES

Data cleaning	Fixes missing values and duplicate applicant IDs.
Feature engineering	Total employment years, income-to-loan ratio and a composite risk score.
Model bench	Six models, each tuned with GridSearchCV and RandomizedSearchCV.
Overfit check	Compares training and validation scores for every model.
Export	Winning XGBoost model and feature importances saved as a .pkl.



Key decisions

01

Keep the weak baseline in the loop

Logistic regression scores only 0.60 but is easy to explain. Every boosted model is judged by its lift over it, which keeps the improvement honest.

02

Compare six models, do not assume

Tree, forest, XGBoost, LightGBM, CatBoost and gradient boosting were each tuned separately. XGBoost won on the held-out test set.

03

Check overfitting before choosing

Train versus validation gaps were checked per model, not just the final test score, to catch a model that memorised the data.

04

Be honest about the ceiling

At about 0.69 ROC-AUC the model helps prioritise outreach. It is not for automatic approve or deny decisions.

Stack, status and what is next

BUILT WITH



KNOWN LIMITS

- ROC-AUC is moderate; not suitable for high-stakes decisions yet.

NEXT STEPS

- Add user interaction and behavioural data beyond the static applicant fields.
- Build a pipeline for continuous model evaluation as new data arrives.

SOURCE

<https://github.com/wailethewave/Digital-Lending-E-Sign-Prediction-System>

