



UNIVERSITY OF PISA

# Predicting Bank Loan Default

Project Presentation – Data Mining & Machine Learning  
Nicolò Bacherotti

# The problem



Loan default prediction is a crucial task for financial institutions.

Identifying in advance whether a client will default on a loan allows banks to reduce financial risk, optimize credit approval processes, and minimize losses.

This project aims to build a classification pipeline that can accurately predict loan default probability based on both numerical and categorical borrower information.

This type of analysis plays a central role in **risk management strategies**, making loan default prediction a key component of modern banking and financial analytics.

# Dataset Overview

**45,000** loan applications with **14** columns, providing detailed information on a large population of loan applicants. No missing, null, or NaN values.

**Goal: Binary Classification** for loan default prediction.

## Features overview:

### Personal Information:

person\_age  
person\_gender  
person\_education  
person\_income  
person\_emp\_exp  
person\_home\_ownership

### Loan Details:

loan\_amnt  
loan\_intent  
loan\_int\_rate  
loan\_percent\_income

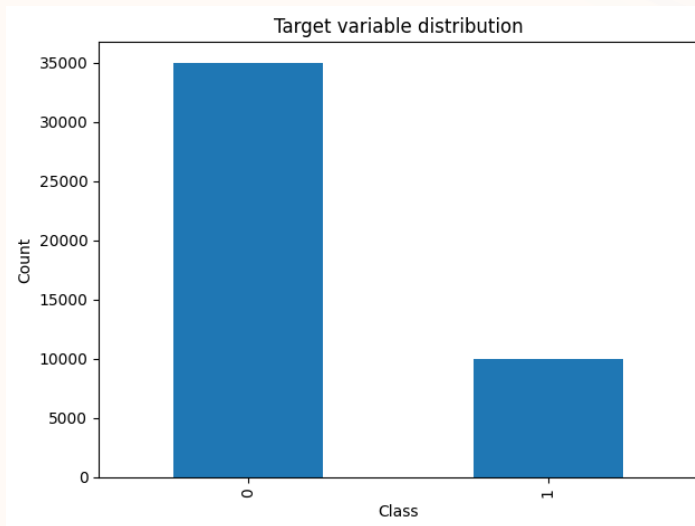
### Credit & Loan History:

cb\_person\_cred\_hist\_length  
credit\_score  
previous\_loan\_defaults\_on\_file

### Target:

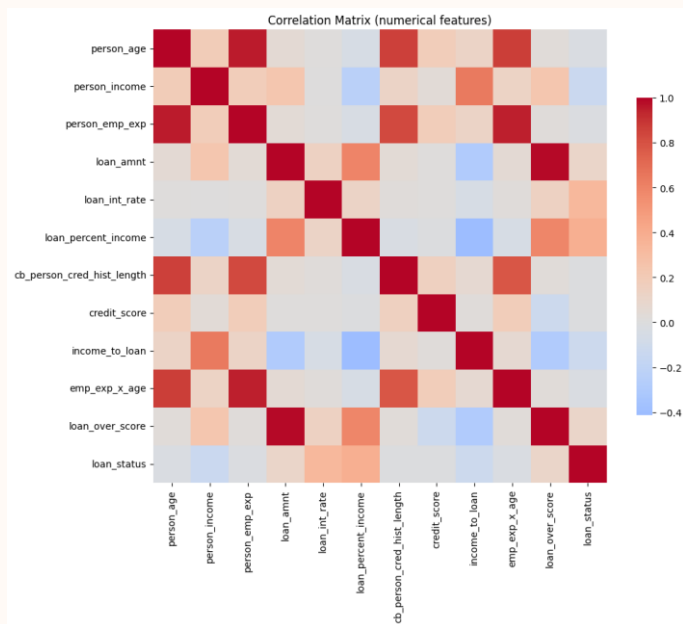
loan\_status (0 = repaid, 1 = default)

## Imbalanced dataset

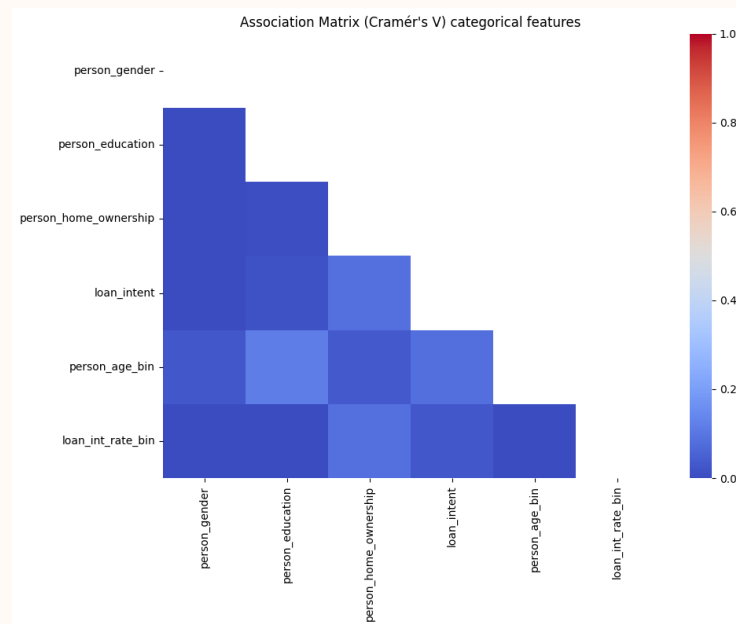


# Dataset Overview

Weak correlations across numerical variables and limited correlation with the target variable



Low association among categorical variables



# Feature Engineering & Pipeline Build



## Feature Engineering:

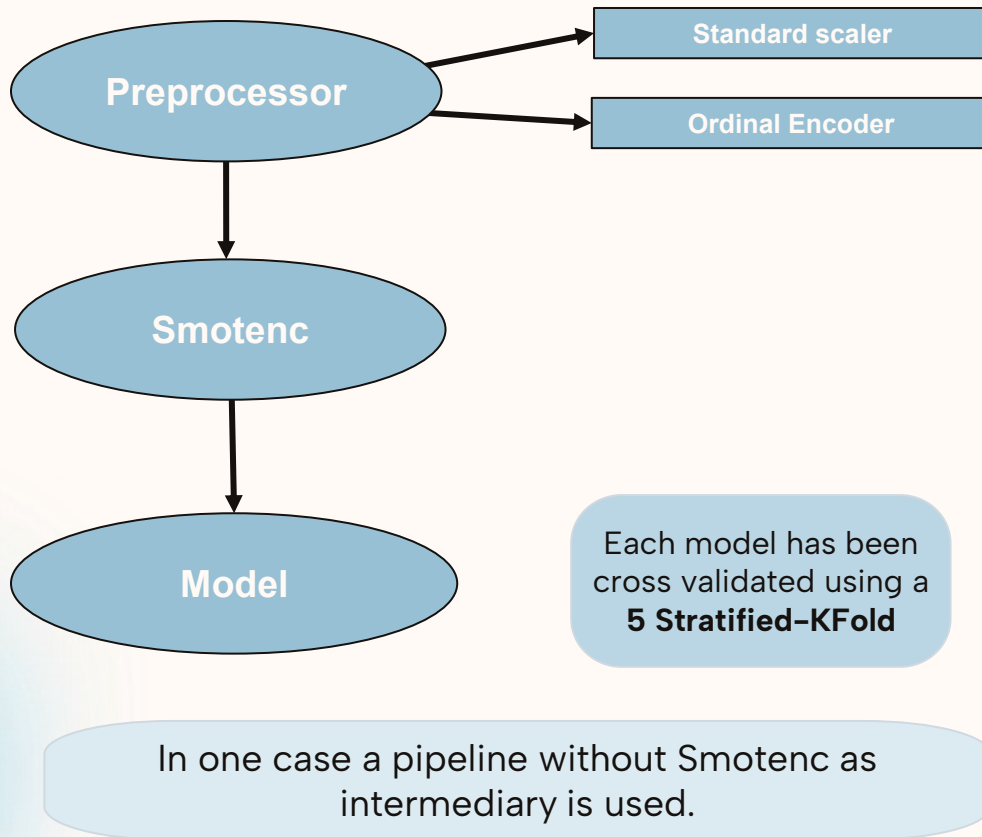
- Person\_age\_bin
- Income\_to\_loan
- Emp\_exp\_x\_age
- Loan\_over\_score
- Loan\_int\_rate\_bin

## Feature Deletion:

- previous\_loan\_defaults\_on\_file

## Applied Log-Transform for skewed features:

- Detects **heavily skewed** numeric features ( $|\text{skew}| > 1$ ) on the **train set**
- Applies  **$\log_{1p}(x)$**  to reduce skewness.



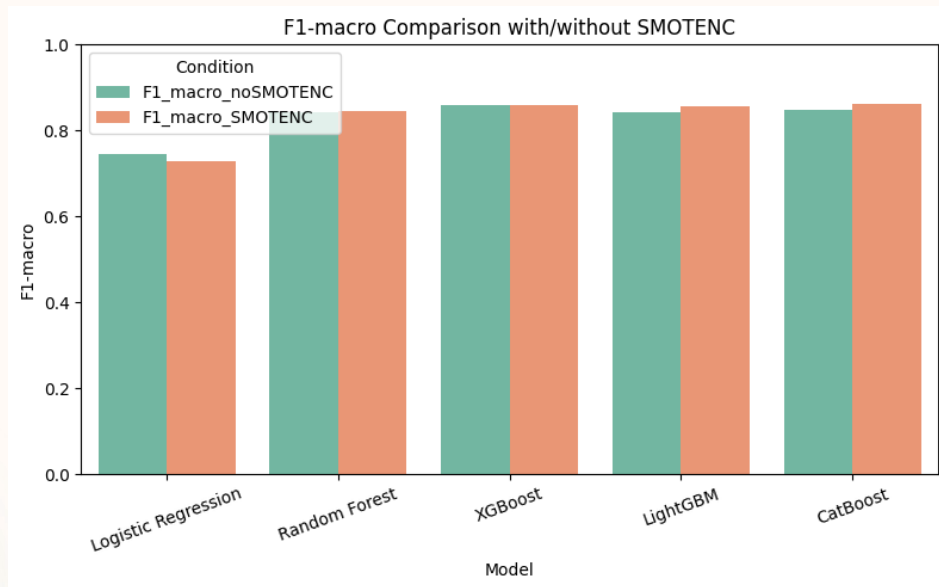
# Initial Model Selection

Five models evaluated in **two pipelines**: Baseline vs with SMOTENC:

- Random Forest
- LightGBM
- Logistic Regression
- XGBoost
- CatBoost

**XGBoost** (0.859) and **CatBoost** (0.861) are the top performers.

**Smotenc was retained** in the pipeline of the subsequent evaluations, leading in small but consistent F1\_macro gains for the best models.



# Grid Search Evaluation

## XGBoost params:

- n\_estimators: [200, 400, 600],
  - max\_depth: [3, 5, 7, 10]
- learning\_rate: [0.01, 0.05, 0.1]
- subsample: [0.6, 0.8, 1.0]



## Best XGBoost params:

n\_estimators: 600,  
max\_depth: 5,  
learning\_rate: 0.1,  
subsample: 0.8

## CatBoost params:

- iterations: [200, 400, 600]
  - depth: [4, 6, 8, 10]
- learning\_rate: [0.01, 0.05, 0.1]
  - l2\_leaf\_reg: [1, 3, 5, 7]



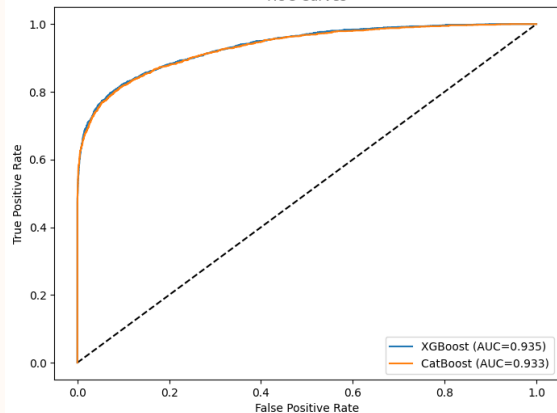
## Best CatBoost params:

iterations: 600,  
depth: 6,  
learning\_rate: 0.1,  
l2\_leaf\_reg: 1

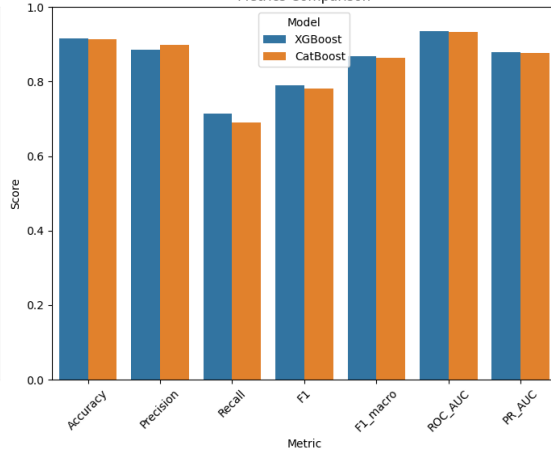
*Each model was then retrained with the optimal settings and assessed performance on the test set*

# Models Evaluation

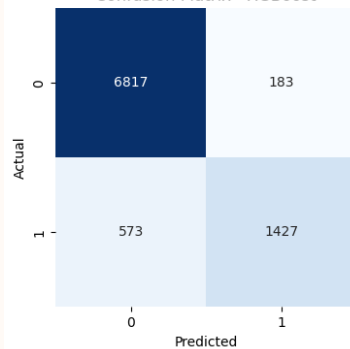
ROC Curves



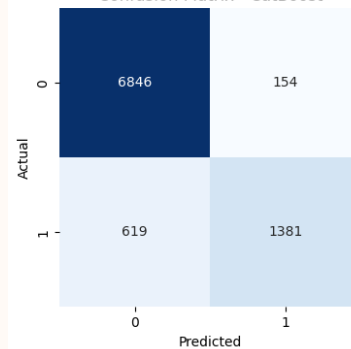
Metrics Comparison



Confusion Matrix - XGBoost



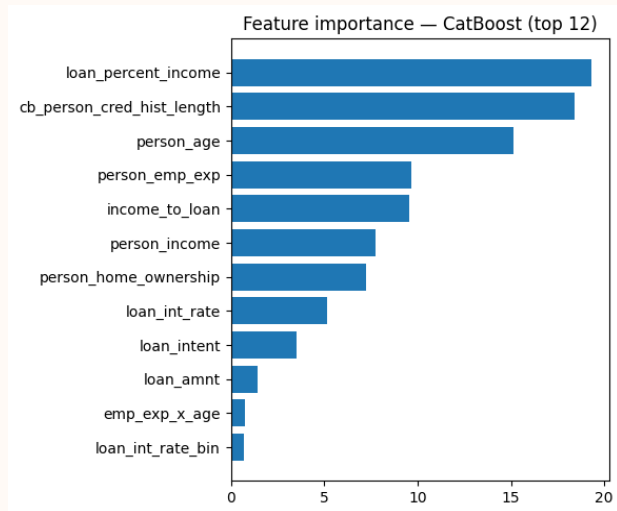
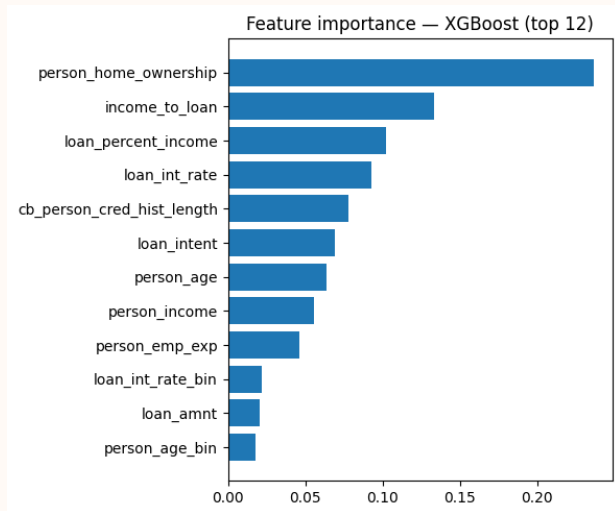
Confusion Matrix - CatBoost



	Accuracy	Precision	Recall	F1_score	F1_macro	ROC_AUC
XGBoost	0.916	0.886	0.713	0.790	<b>0.869</b>	0.935
CatBoost	0.914	0.899	0.690	0.781	<b>0.863</b>	0.933



# Feature Importance



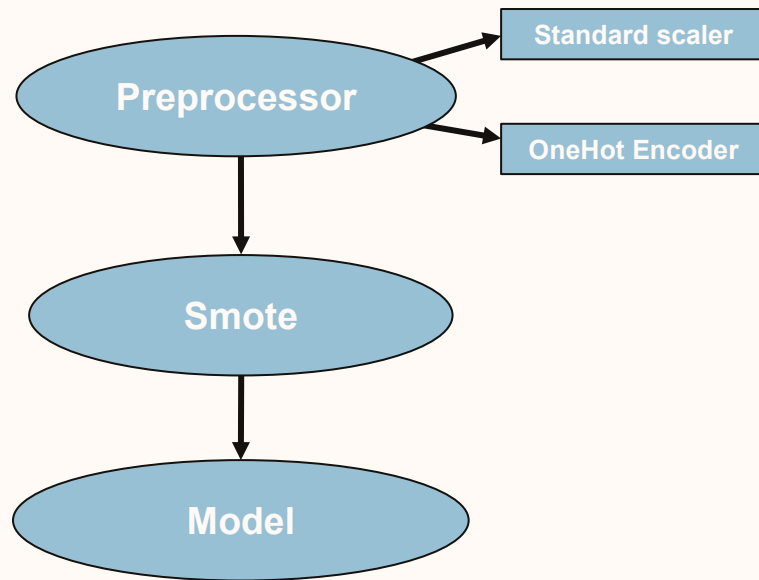
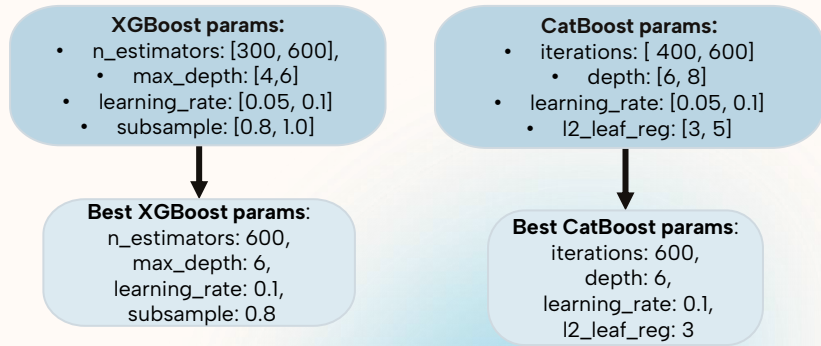
# Comparison with the State of the Art

Following the reported studies, i decided to analyze my dataset also with pipelines containing **One-Hot Encoding** and **SMOTE**.

The models tested in the cross-validation were:

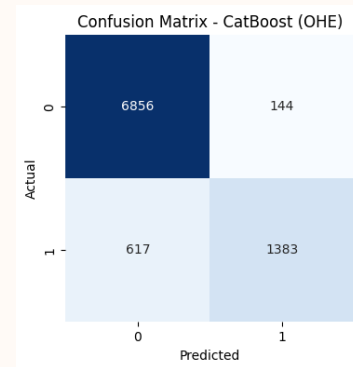
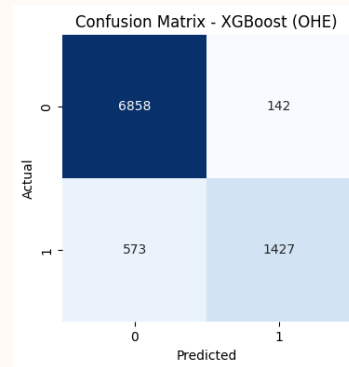
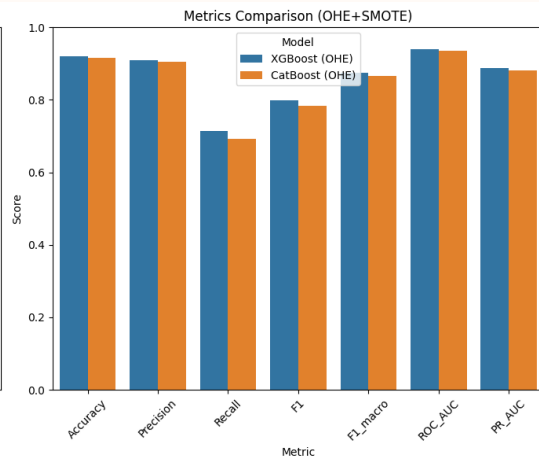
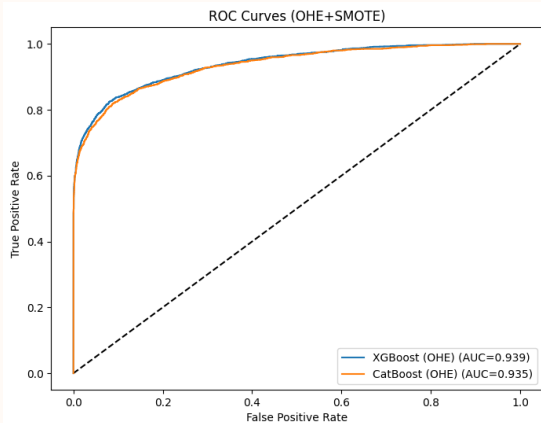
- Random Forest (F1\_macro=0.850)
- XGBoost (F1\_macro=0.863)
- CatBoost (F1\_macro=0.864)

A grid search was carried out on the 2 best models:



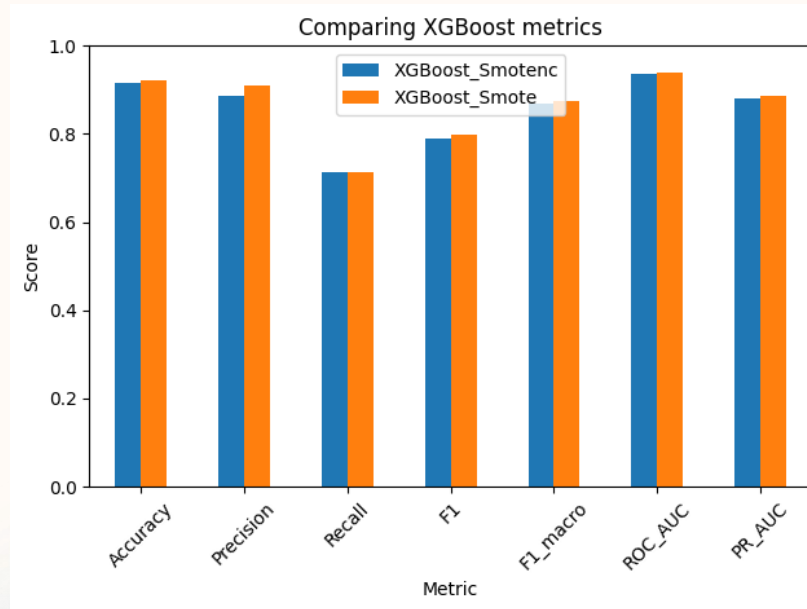
# Comparison with state of art

## Models Evaluation



	Accuracy	Precision	Recall	F1_score	F1_macro	ROC_AUC
XGBoost	0.920	0.909	0.713	0.799	<b>0.875</b>	0.939
CatBoost	0.915	0.905	0.691	0.784	<b>0.865</b>	0.935

# Comparison of the best model under the two configurations



XGBoost appears robust under both strategies;  
Configuration with SMOTE has slightly  
better performances in almost all the metrics.

# User Interface

## Credit Risk Prediction

Gender

female



Education

High School



Home ownership

RENT



Loan intent

PERSONAL



Age

30



Annual income

30000



Predict risk

Prediction: No Default

Probability of default: 12.61%

# References

Dataset: [https://www.kaggle.com/datasets/udaymalviya/bank-loan-data?select=loan\\_data.csv](https://www.kaggle.com/datasets/udaymalviya/bank-loan-data?select=loan_data.csv)

Monje, L., Carrasco, R.A. & Sánchez-Montañés, M. Machine Learning XAI for Early Loan Default Prediction. *Comput Econ* (2025).

<https://doi.org/10.1007/s10614-025-10962-9>

Fekadu, R., Getachew, A., Tadele, Y., Ali, N., & Goytom, I. (2022). Machine Learning Models Evaluation and Feature Importance Analysis on NPL Dataset. arXiv:2209.09638.

<https://arxiv.org/abs/2209.09638>