Neural Academy

Final Project Presentation

Option 5 – Credit Risk Dataset

Enrico Sain

Problem Statement

Neural Academy

The dataset contains simulated credit bureau data

The dataset is composed of 12 columns:

- 11 Features
- 1 binary target (loan_status)

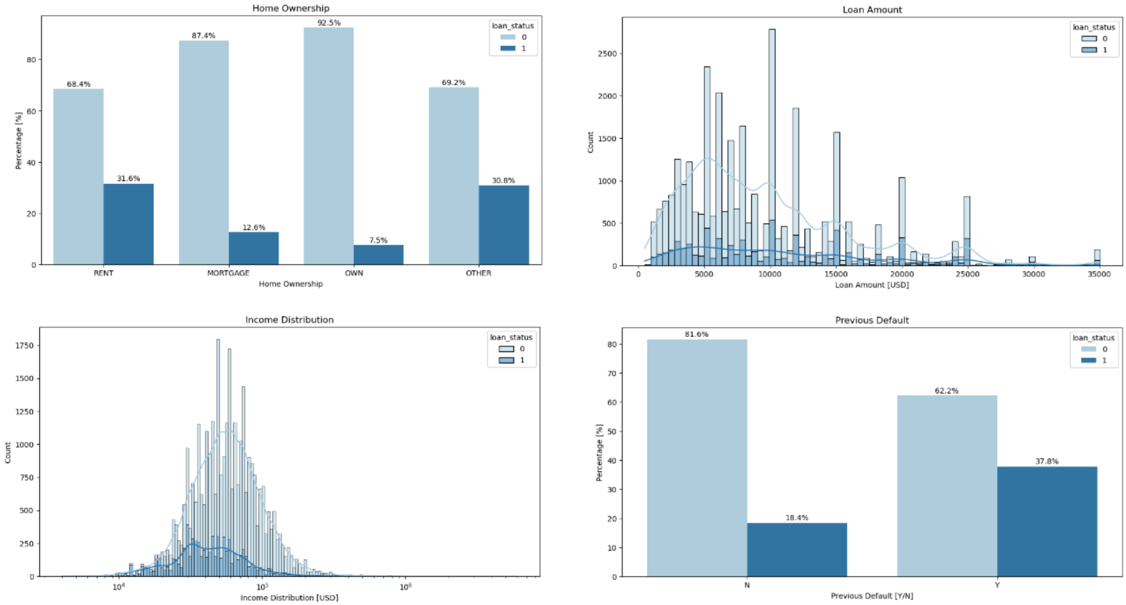
Problem statement: predict the customer loan status

Problem type: binary classification (supervised learning)

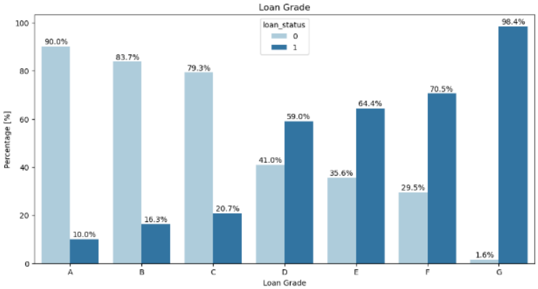
(source: https://www.kaggle.com/datasets/laotse/credit-risk-dataset)

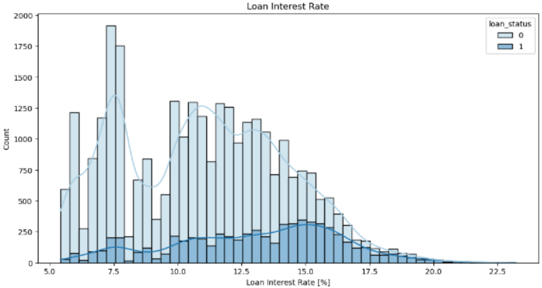
Feature Name:	**Description:**
person_age	Age
person_income	Annual Income
person_home_ownership	Home ownership
person_emp_length	Employment length (in years)
loan_intent	Loan intent
loan_grade	Loan grade
loan_amnt	Loan amount
loan_int_rate	Interest rate
loan_status	Loan status (0 is non default 1 is default)
loan_percent_income	Percent income
cb_person_default_on_file	Historical default
cb_preson_cred_hist_length	Credit history length

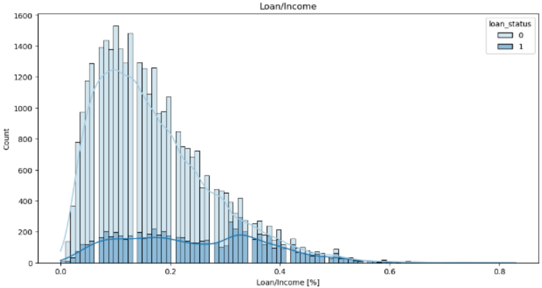
Exploratory Data Analysis

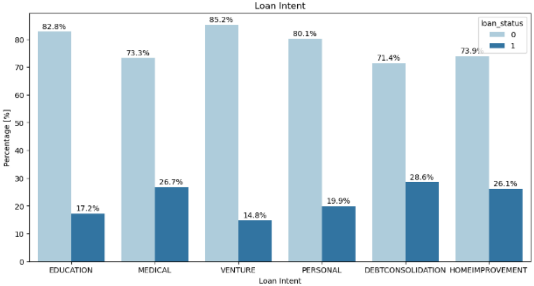


Exploratory Data Analysis









Default Customer Profile:

- Defaulted customers have a lower income than non defaulted customers.
- Most of the defaulted customers rent a house, and only a few of them are home owners.
- Most of the defaulted customers apply for a loan for debt consolidation, medical reasons or home improvement.
- Loan grade for defaulted customers is generally lower than all other customers, a reason why this feature could be strictly correlated with the target.
- Interest rates and loan/income ratio are generally higher for defaulted customers.
- Defaulted customers show also a higher number of previous defaults than non defaulted customers.

Other Insights:

- Moderately imbalanced dataset (78% class 0, 22% class 1)
- 165 duplicate rows
- **person_age** highly correlated with **cb_person_cred_hist_length**
- Outliers in **person_age** (age above 122 years) and **employment_length**
- 3981 Nans (887 in **person_emp_length** and 3094 in **loan_int_rate**)

Preprocessing

Neural Academy

Features

Ordinal Categorical

- loan_grade

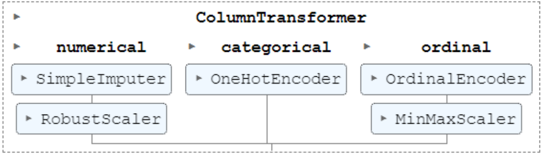
Nominal Categorical

- person_home_ownership
- loan_intent
- cb_person_default_on_file

Numerical

- person_age
- person_income
- person_emp_length
- loan_amnt
- loan_int_rate
- loan_percent_income

Pipeline



1. Data Cleaning (outliers, duplicate rows, highly correlated features)
2. Train test split with stratify option in order to create train and test subset with the same target class ratio
3. Preprocessing pipelines for each different kind of feature

Model Evaluation and Selection

Neural Academy

Baseline
- Decision Tree
- KNN
- Logistic Regression

Ensemble
- Random Forest
- XGBoost
- CatBoost

Ensemble Tuned*
- Random Forest
- XGBoost
- CatBoost

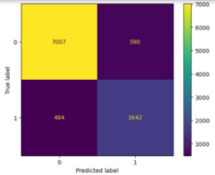
Voting Classifier
- Hard Voting
- Soft Voting

1. Classification Report

	precision	recall	f1-score	support
0	0.94	0.92	0.93	7597
1	0.74	0.77	0.75	2126
accuracy			0.89	9723
macro avg	0.84	0.85	0.84	9723
weighted avg	0.89	0.89	0.89	9723

F1-score average is: 0.890
Recall score (class 1) is: 0.772

2. Confusion Matrix

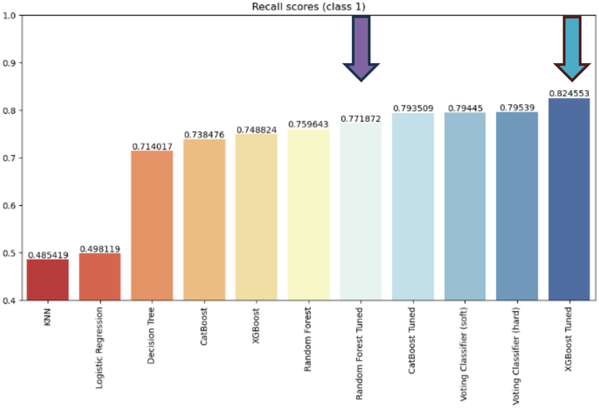


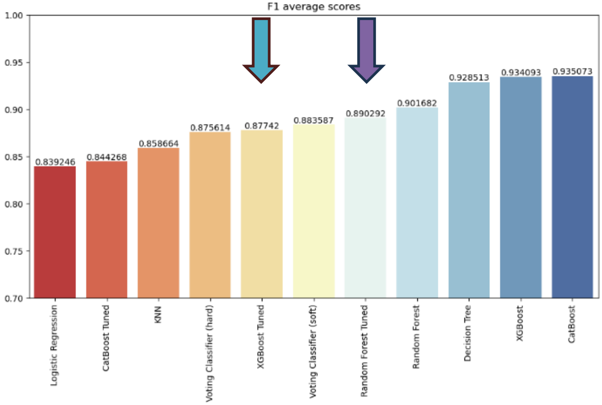
3. Overfit check

4. Precision Recall curves**

*Ensemble models tuned with RandomizedSearchCV using weight hyperparameters to cope with unbalanced dataset
**only for Ensemble tuned

Model Evaluation and Selection





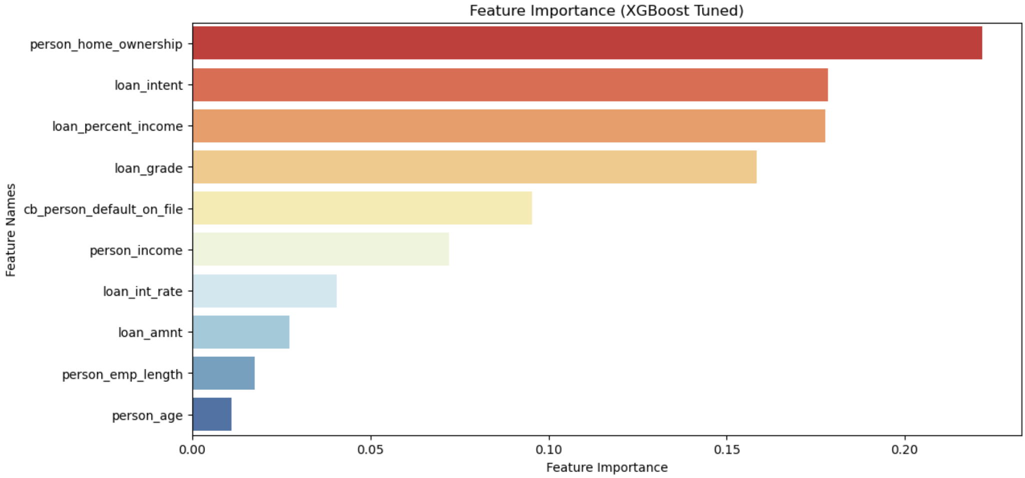
Tuning strategy: Recall score Optimization

+1,6% Random Forest

+7,4% CatBoost

+10,1% XGBoost

Feature Importance



Conclusions and Future Work Recommendations

Conclusions:

We have trained 11 different ML models on the credit_risk_dataset and compared the f1 number and recall metrics calculated on the test set for each of them. The best model if we look at recall is certainly the optimized XGBoost with a weighted average f1 score of 88%, a high recall of 82.5% on class 1 and the best precision recall curve.
Voting classifier (soft) or Random Forest tuned are still very good options if we are looking to achieve a more balanced model with slightly higher f1 and slightly lower recall.

Recommendations for future work:

- Deploy other strategies to cope with unbalanced datasets and compare the results (downsampling, upsampling, synthetic data augmentation e.g. SMOTE, use of imbalanced learn library etc.)
- Try more hyperparameters and higher ranges in order to improve the tuning results
- Try Bayesian Optimization as a more efficient way to improve hyperparameter tuning
- Train more Ensemble methods based on Decision Tree (e.g. LightGBM)
- Drop least important features from the dataset and retrain the model
- Assign weights to each loan in order to optimize models that minimize financial loss