Project 1 (Final) - DataLab mentorship program - 18/10/2023

Credit card risk analysis

Bruno Ortega Goes
Mentor: Carlo(s)

serasa experian™

Credit card risk analysis

- **Goal:** predict if an applicant is **'good' or 'bad' client**,
- The definition of 'good' or 'bad' is not given → **I choose the criterium for the label**.
- **Unbalance data** problem is a big problem in this task.
- Dataset: two tables that can be merged by the client ID
  - application_record.csv
  - credit_record.csv

Application record: 17 features

- **ID: Client number**
- CODE_GENDER: Gender (🚫 not used to train for ethical concerns)
- FLAG_OWN_CAR: Is there a car
- FLAG_OWN_REALTY: Is there a property
- **CNT_CHILDREN: Number of children**
- **AMT_INCOME_TOTAL: Annual income**
- NAME_INCOME_TYPE: Income category
- NAME_EDUCATION_TYPE: Education level
- NAME_FAMILY_STATUS: Marital status
- NAME_HOUSING_TYPE: Way of living
- **DAYS_BIRTH: Birthday. Count backwards from current day (0), -1 means yesterday**
- **DAYS_EMPLOYED: Start date of employment. Count backwards from current day(0). If positive, it means the person currently unemployed.**
- FLAG_MOBIL: Is there a mobile phone.
- FLAG_WORK_PHONE: Is there a work phone
- FLAG_PHONE: Is there a phone
- FLAG_EMAIL: Is there an email
- **OCCUPATION_TYPE: Occupation (a lot of missing values: inputed as "not provided")**
- **CNT_FAM_MEMBERS: Family size**

- 💡**Engineered features**
  - AMT_INCOME_PER_PERSON
  - AMT_INCOME_PER_CHILD
  - "EMPLOYED"

Credit record: 2 features

- **ID: Client number**
- **MONTHS_BALANCE:** Record month. The month of the extracted data is the starting point, backwards, 0 is the current month, -1 is the previous month, and so on
  - **Some clients have a record of 62 months**

- **STATUS:**
  - **C: paid off that month**
  - **0: 1-29 days past due**
  - 1: 30-59 days past due
  - 2: 60-89 days overdue
  - 3: 90-119 days overdue
  - 4: 120-149 days overdue
  - 5: Overdue or bad debts, write-offs for more than 150 days
  - X: No loan for the month

The good, the bad and the Julius

- **Good clients**
  - Looking at the 3 last months they
  - Never paid in retard: C
  - Paid at most with 29 days in retard: 1
- **Class 0**

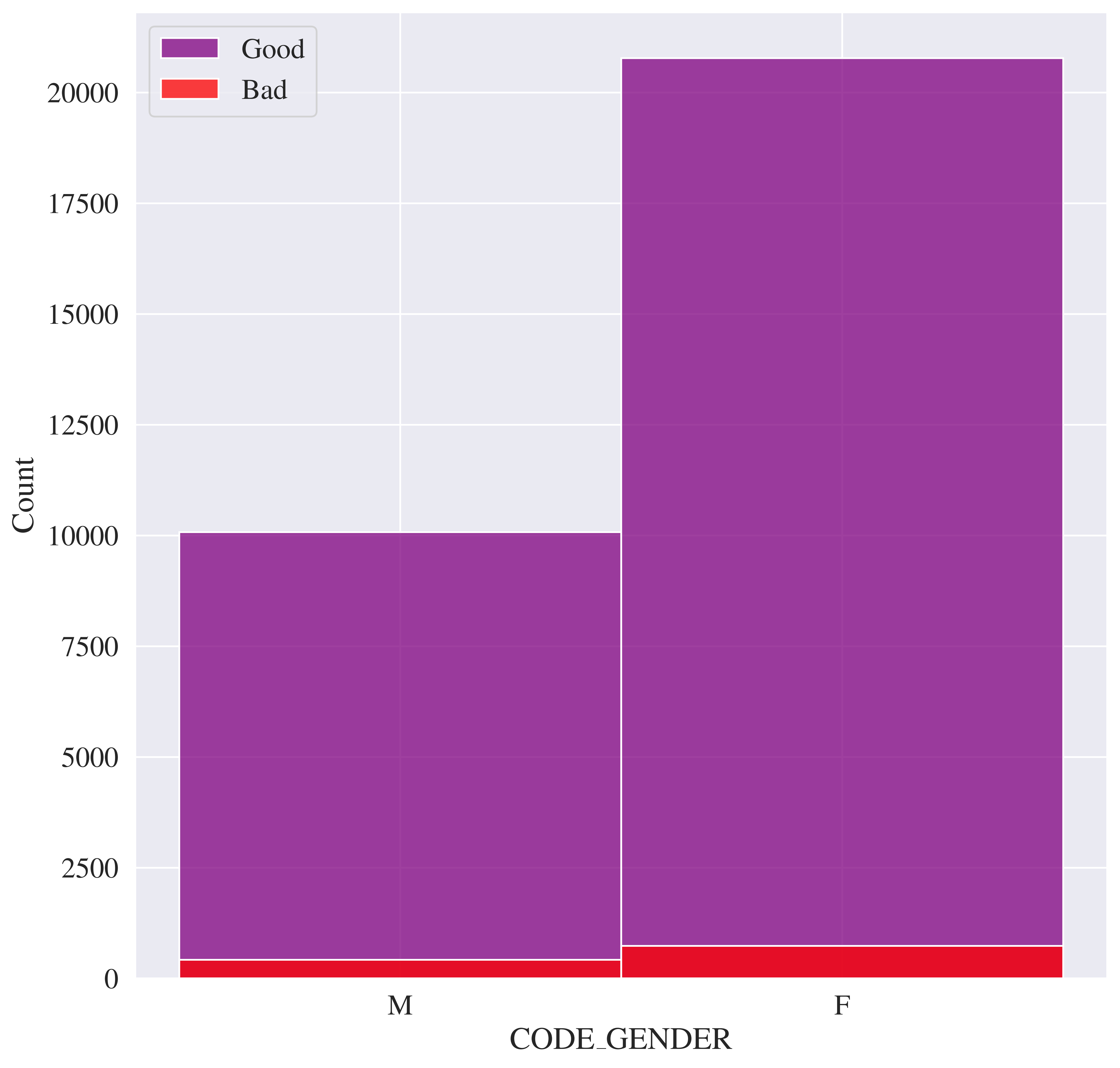
- **Bad clients**
  - Looking at the 3 last months they were in retard of more than 29 days
  - **Class 1**

- **Julius Discarded**
  - Didn't touch their credit cards
  - **Class -1**

Outline

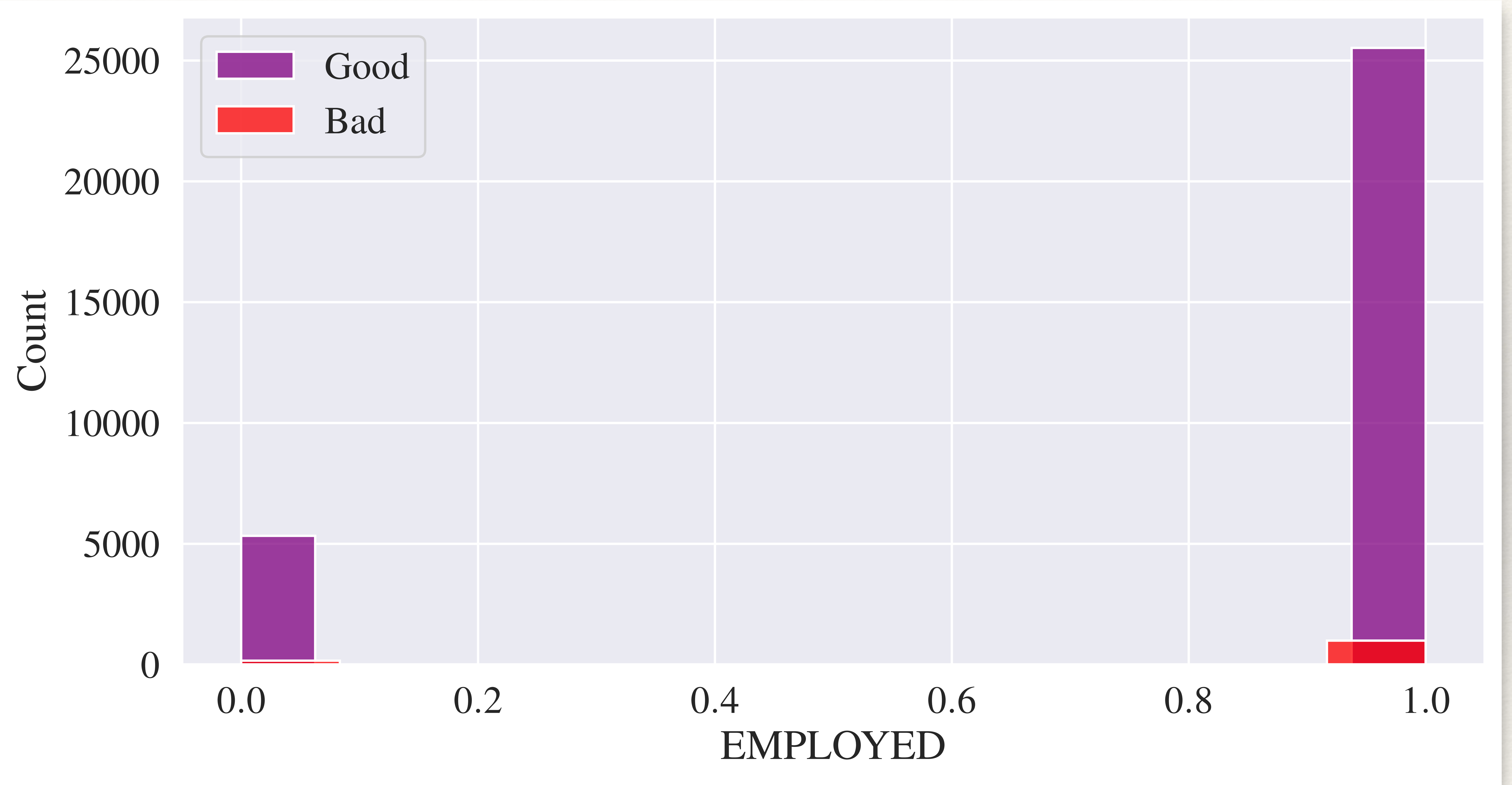
- Preliminaries
  - Data exploration
  - Data preprocessing
- Main results
  - Model trained: XGBoost
  - Rejected inference
- Conclusions and perspectives

More women than man in the data set.

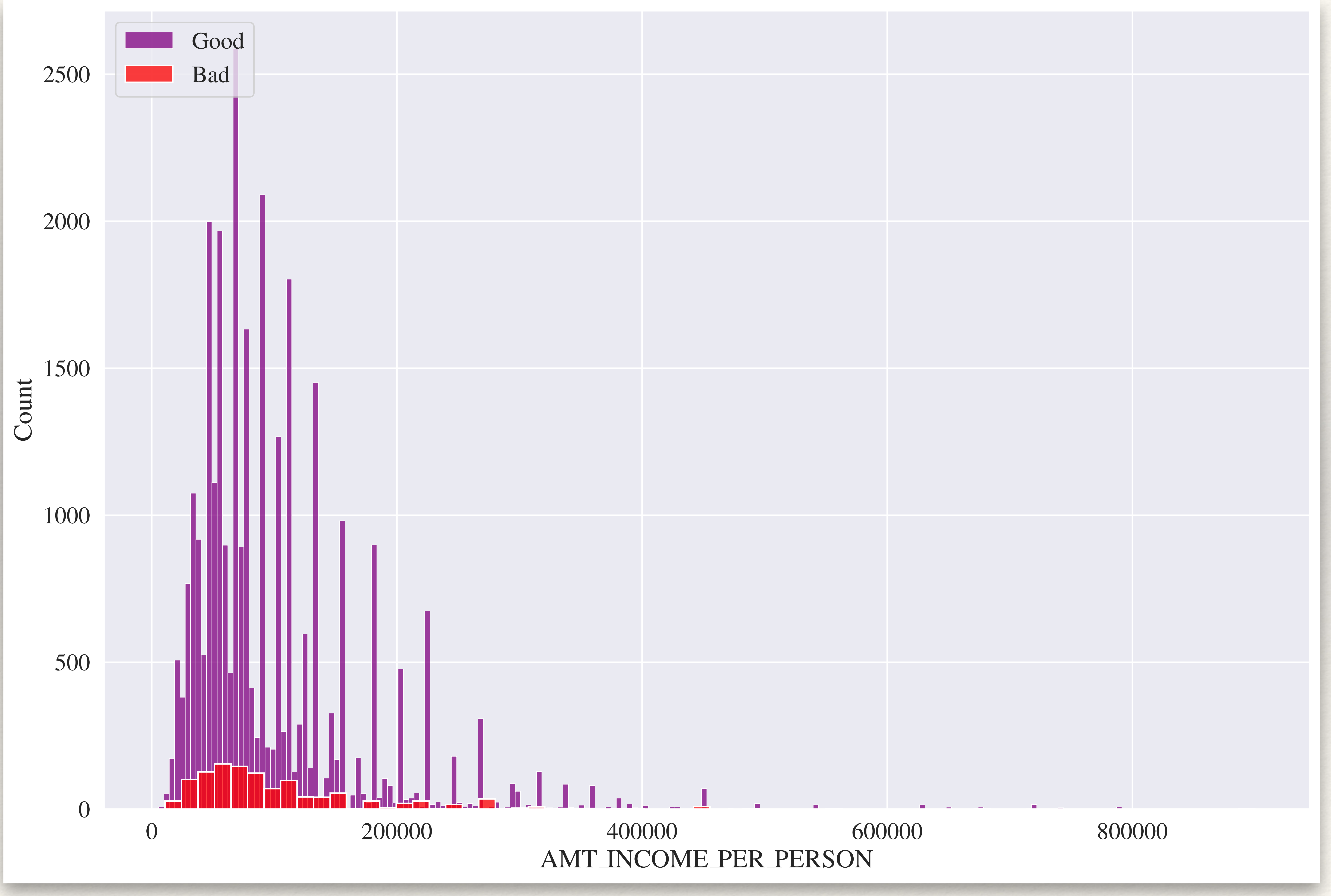


- **Feature excluded of the working dataset for a ethical reason: no sexist machine learning model.**

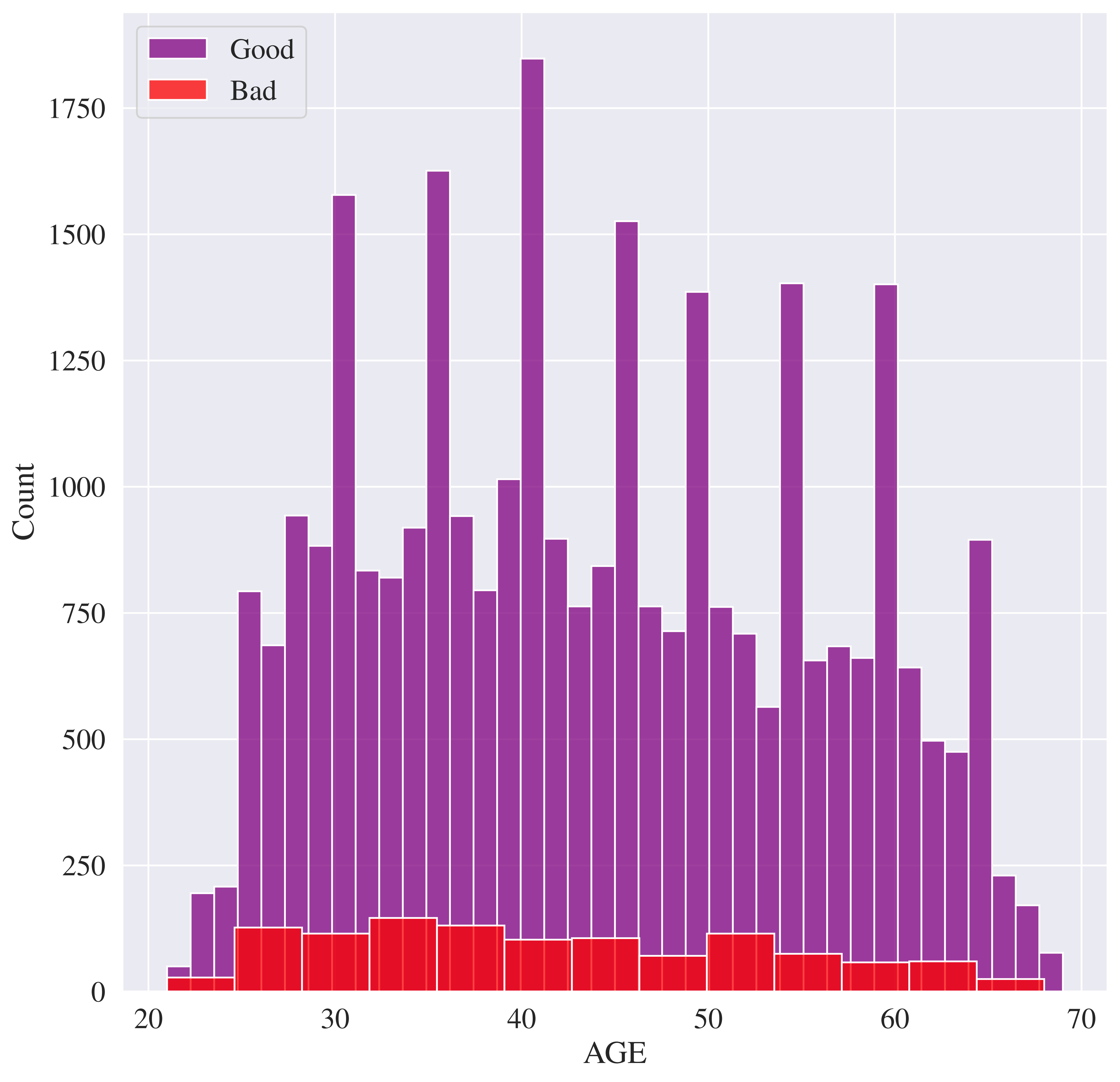
More employed (1) than unemployed people (0)



Obvious feature to create: How much of the total income is distributed among the expenses with all the family members?



- **Max: 900000.0 money/person**
- **Min: 5625.0 money/person**



- Youngest client: 21 years old
- The more experienced in life: 69 years old

Outline

- Preliminaries
  - Data exploration
  - Data preprocessing
- Main results
  - Model trained: XGBoost
  - Rejected inference
- Conclusions and perspectives

General encoding and scaling

- **Numerical features→Robust scaler**
  - Good against outliers
- **Categorical features→Target encoder**
  - features are replaced with a blend of posterior probability of the target given particular categorical value and the prior probability of the target over all the training data. **(drawback: is prompt to leakage)**

Training method

- **Training set** with 23 562 lines
- **Calibration set** with 2 049 lines
  - To calibrate the model and the threshold for f_1, precision and recall scores
- **Test set** with 6 403 lines
- **Naïve model**: is it possible to correctly classify only using **only** the age?
  - $f(age) = -\, age$
  - AUC=0.55→Baseline

Outline

- Preliminaries
  - Data exploration
  - Data preprocessing
- Main results
  - Model trained: XGBoost
  - Rejected inference
- Conclusions and perspectives

Training method

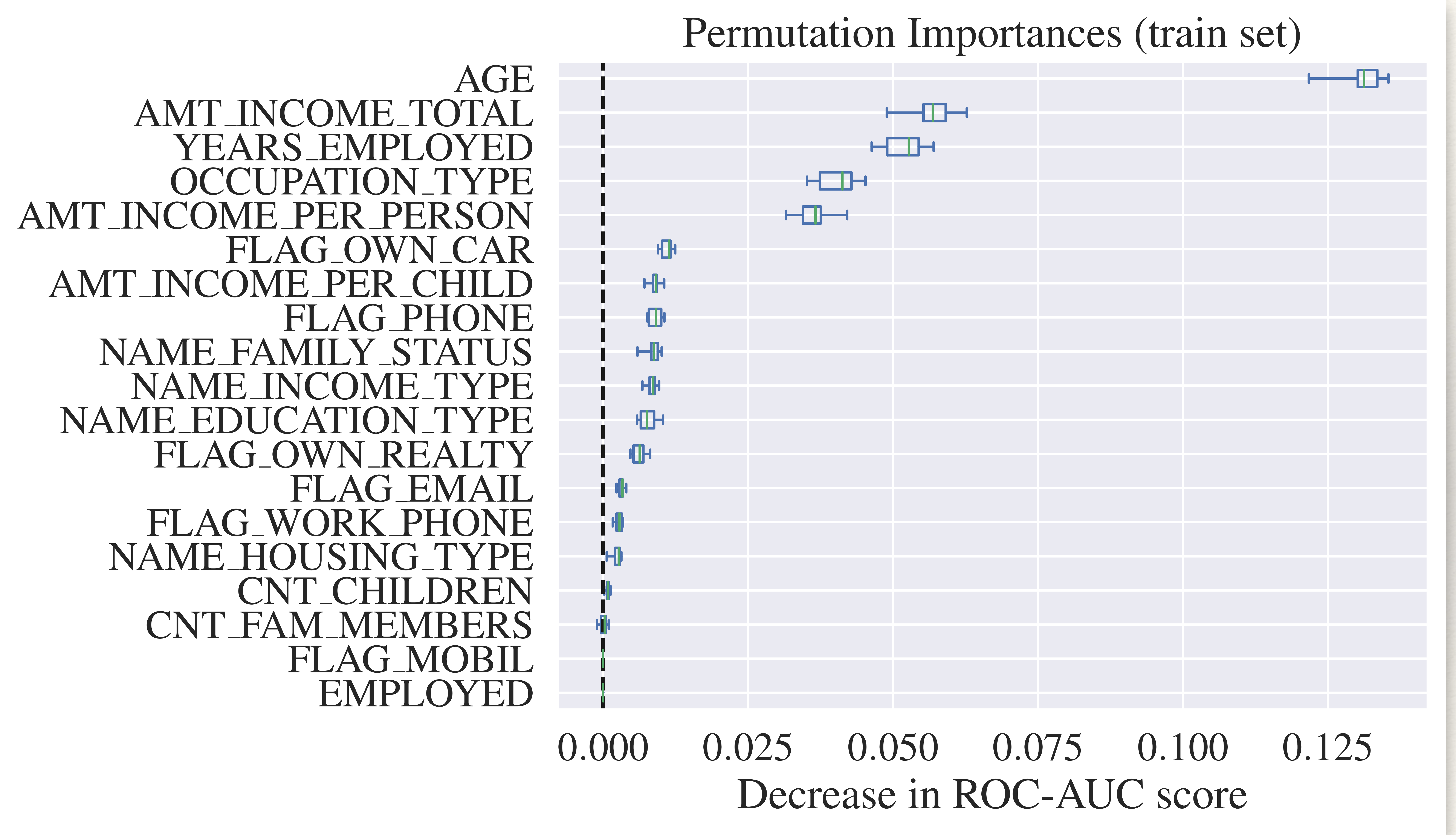
- **Machine learning models**
  - Logistic regression
  - Random Forest
  - **XGBoost**
  - KNeighrestNeighbors
- **Hyperparameter tuning** with **Random search** looking for the highest AUC in the parameter space.
  - Up to 100 iterations
- **10-fold Cross-validation** in the training set for each of the models

XGBoost model: AUC = 0.68, f_1=0.23

- Optimal hyperparameters:
  - n_estimators=266
  - max_depth=27
  - max_leaves=5
- AUC = 0.68 (10-fold cross validated)
- f_1 = 0.24 (10-fold cross validated)
  - Threshold: 0.3
  - Precision: 0.24
  - Recall: 0.24

	Good	Bad
Predicted Good	5994	177
Predicted Bad	176	56

Feature permutation importance



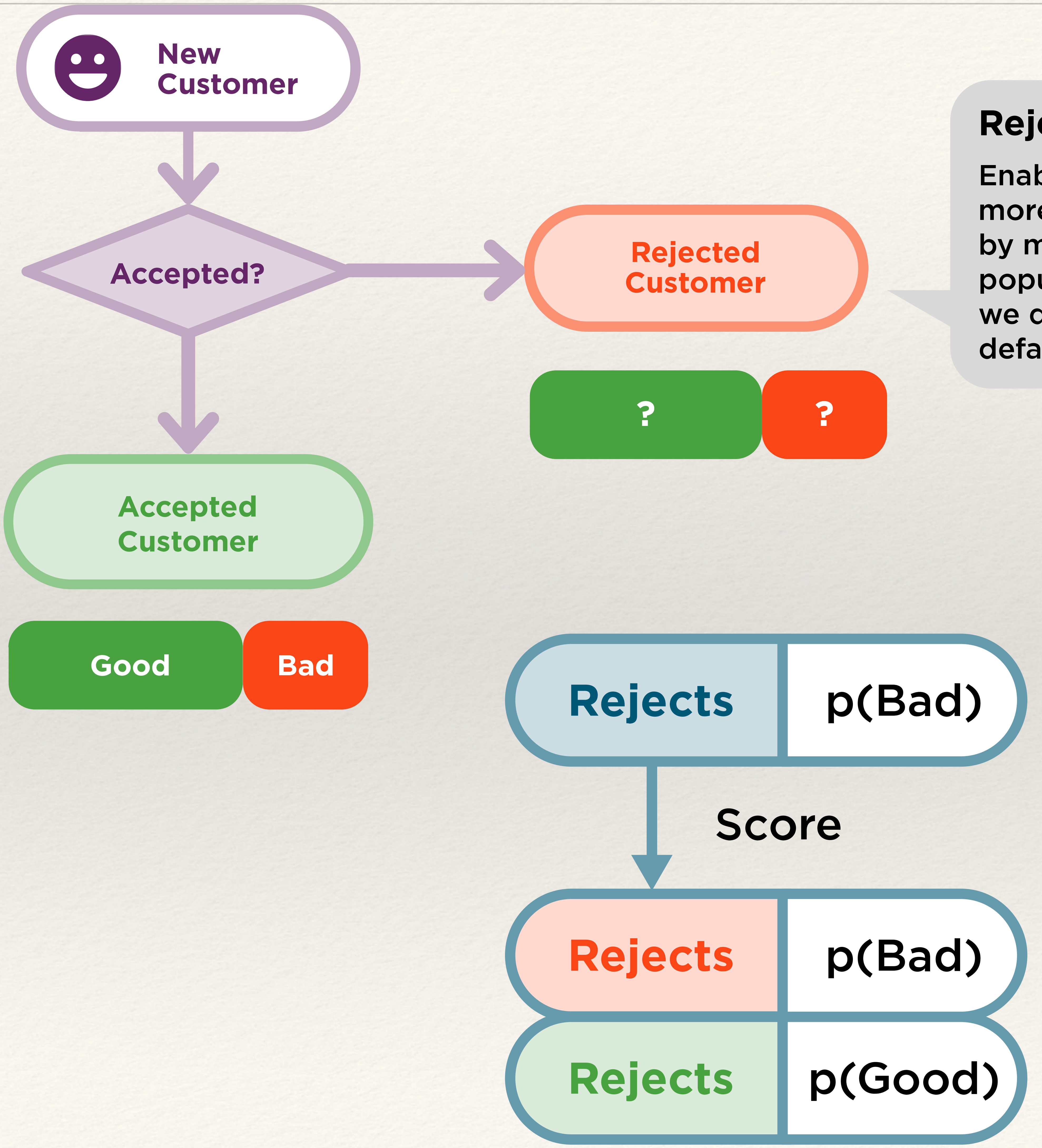
Outline

- Preliminaries
  - Data exploration
  - Data preprocessing
- Main results
  - Model trained: XGBoost
  - Model calibration
  - Rejected inference
- Conclusions and perspectives

Rejected inference

Application SC

Can be built using historically accepted customers to whom we know default status



Reject Inference

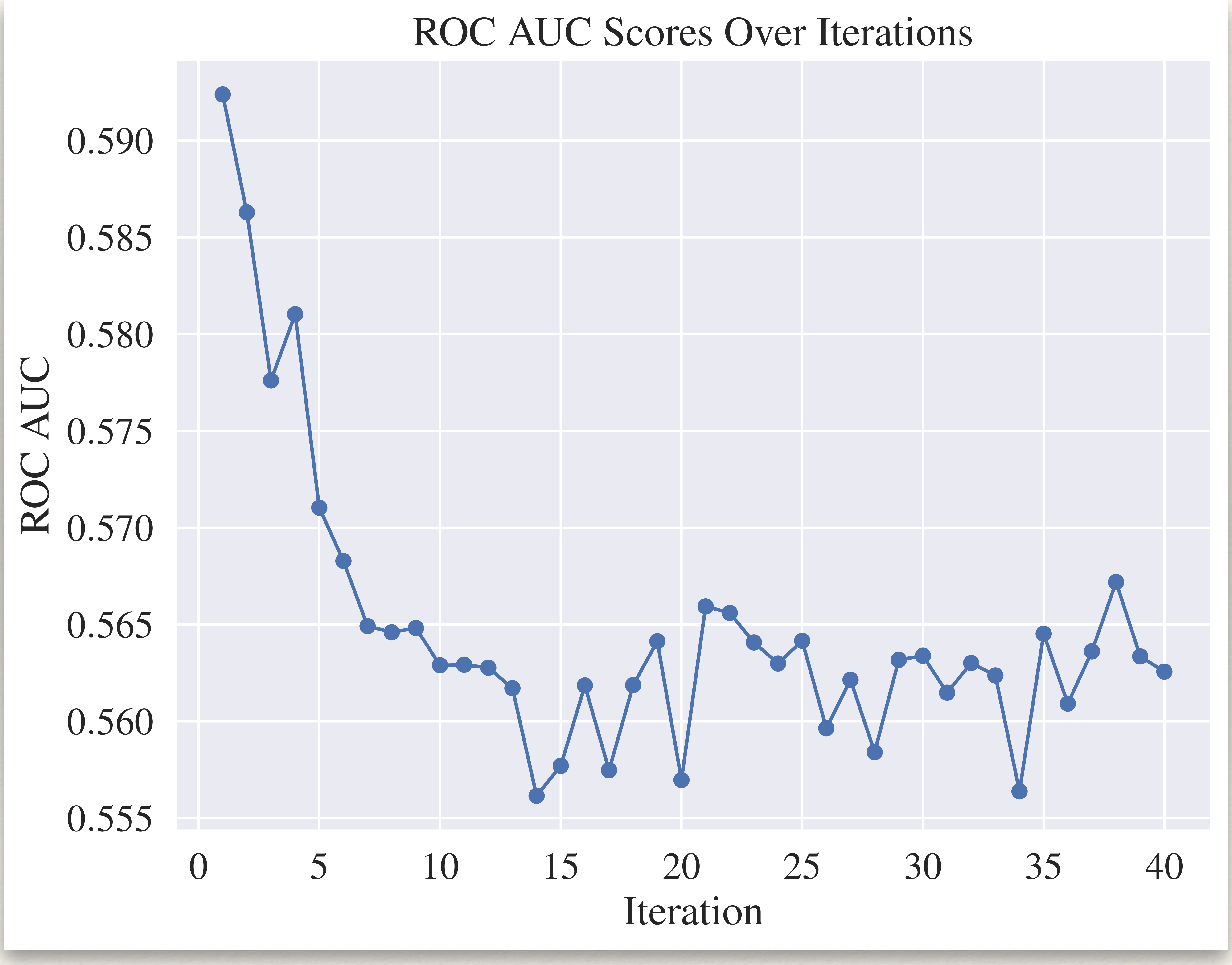
Enables building a more robust SC model by modeling the population to whom we do not know default status

Observed values

$\mathscr{M}$ rule	Good	Bad
Got the credit line	48516	156
Rejected	????	????

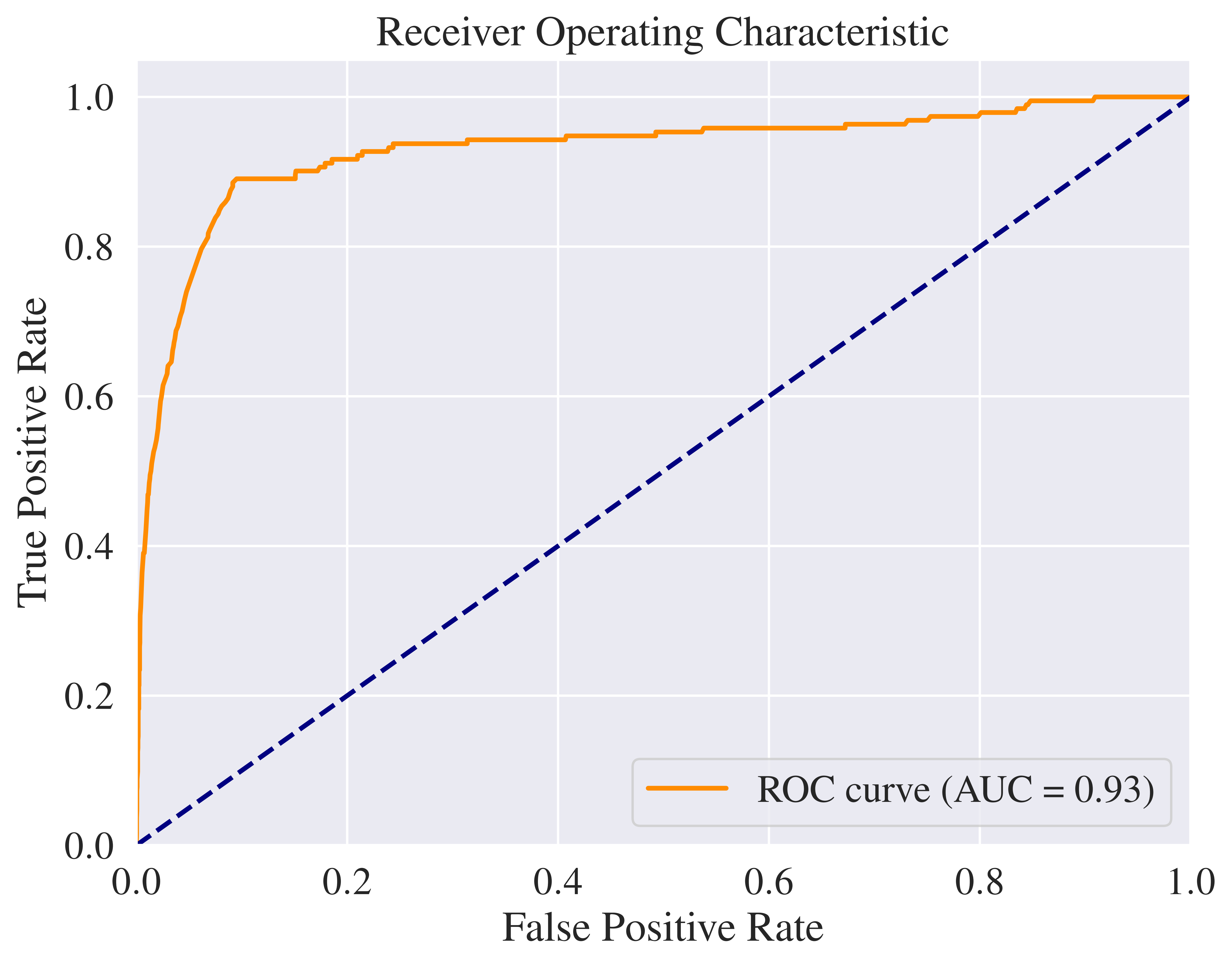
Can't be evaluated because they were … rejected!

ROC-AUC in the fuzzy prediction set



- The ROC-AUC scores comparing the prediction using **sample_weigths** on the training set against the y_fuzzy test.
- After about 40 interactions it stabilizes around 0.56

High ROC-AUC in the when comparing to the “hidden values” y_rejected



- The model is very accurate with the classification of the rejected group
- Here we compare the ROC between the “hidden” y_rejected (that in this controlled case we actually have access against the predictions.

Outline

- Preliminaries
  - Data exploration
  - Data preprocessing
- Main results
  - Model trained: XGBoost
  - Model calibration
  - Rejected inference
- Conclusions and perspectives

Conclusions and perspectives

- We performed a credit card risk analysis where we treated data and trained several models with a given rule for classification, created important features and used techniques such as hyper parameter tuning and cross-validation to corroborate the results
  - Obtained good results with XGBoost model
  - Consider other rule of good/bad clients?
- Rejected inference demonstrated high ROC-AUC for the rejected clients.

Thanks for the attention!

SCAN ME

Code available on my GitHub, click here or scan the QR code above.

Backup slides

What is the cost of the true negatives?

- **Strategy:**
  - 1) Come up with a rule $\mathscr{M}$ that **rejects** some of the individuals that we **actually have the score!**
    - My rule is: if you're below 30 years old and earns less than 40 000
  - From my dataset I create 2 other data sets:
    - Accepted: This provide the training set and I can verify the scores with the test set because I have the target
    - Rejected: This one has a **hidden target,** basically I have only the features of this population and I have no idea if they are good or not bad payers
  - I apply a predict_proba on the rejected people, and I choose a "threshold" (arbitrarily) to come up with their label
  - Then I put these guys on the accepted set and I repeat the training step with these guys included.