Banking case study

André Pires - ei12058@fe.up.pt

João Bandeira - ei12022@fe.up.pt

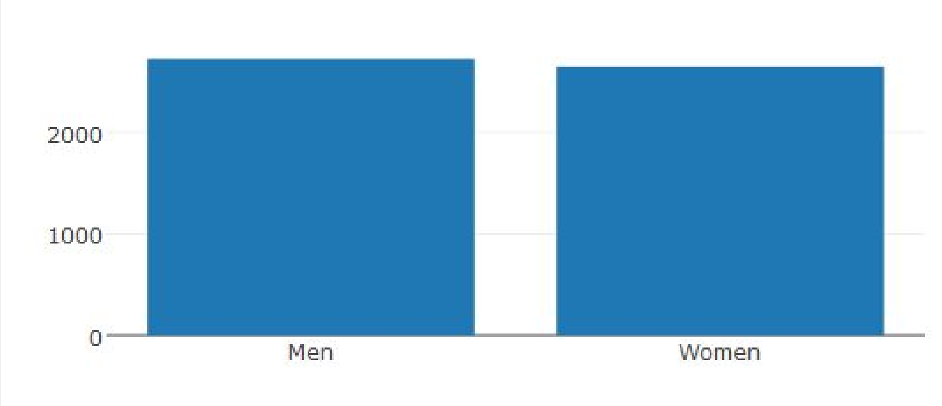
Domain description

- Develop a data mining case study: banking loans
- The dataset included information about the clients, their address, their accounts and related transactions, their credit cards and about the loans themselves
- Two tasks:
  - Descriptive: describe the clients profile
  - Predictive: predict if a loan should be provided or not
- We used Rstudio mainly for preparing the dataset
- We used RapidMiner to perform the main descriptive and predictive tasks

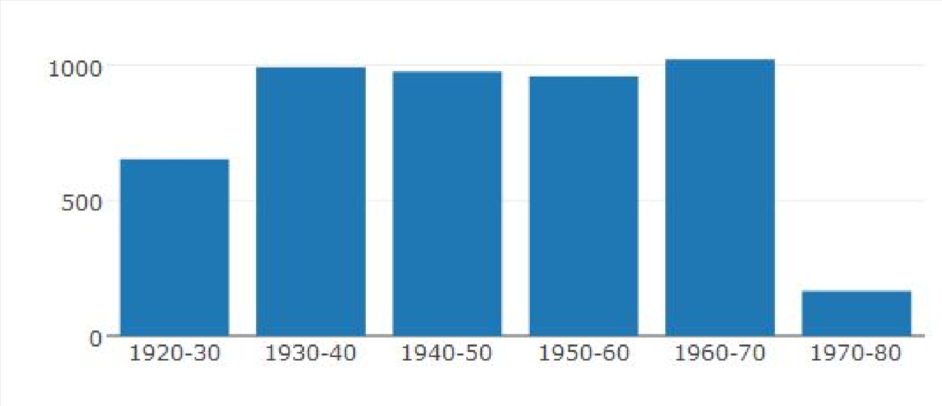
Exploratory data analysis

Client type:

- Gender
  - Almost an even number between the genders
- Age
  - Most clients were born in between the decades 1930 and 1970
  - Very few clients were born in the 70s



Men to women ratio: 1.029868

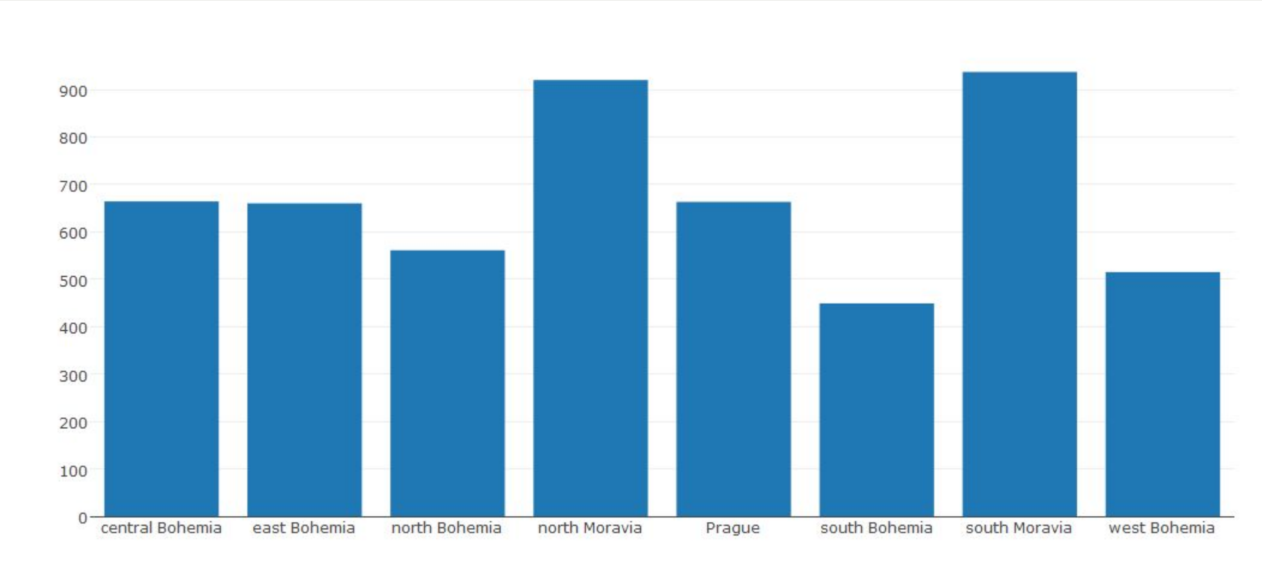


Clients birth decade year

Exploratory data analysis

Clients address:

- Region
  - No relevant information, as there is no major differences in which regions clients live in
  - The main regions are north and south Moravia

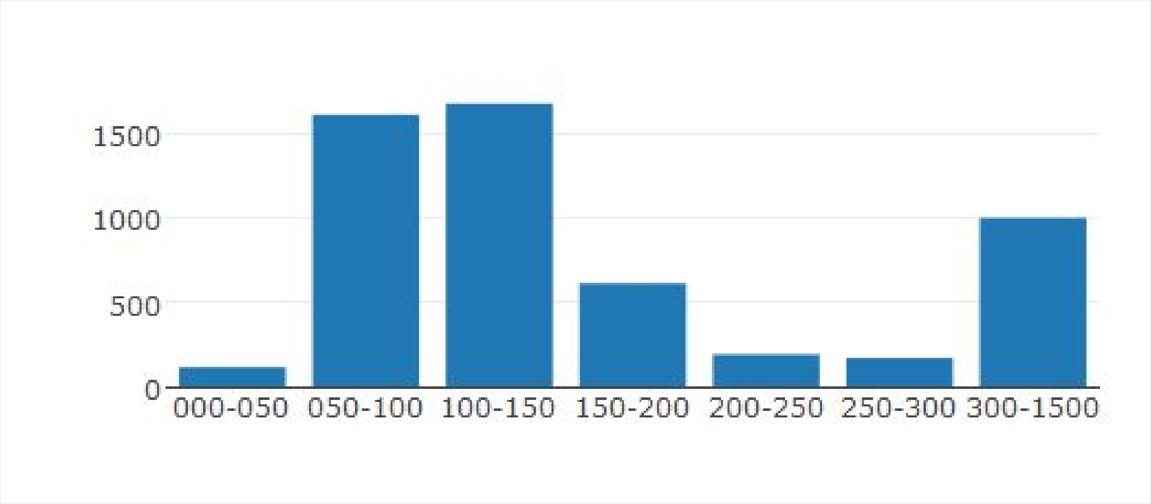


Nr. Clients by region

Exploratory data analysis

Clients address:

- District population
  - A lot of clients live in an average populated district (50-150k population)
  - A few in huge districts (population size >300k)
  - The remaining few clients live in small districts (<50k) and in average populated districts (150-300k)

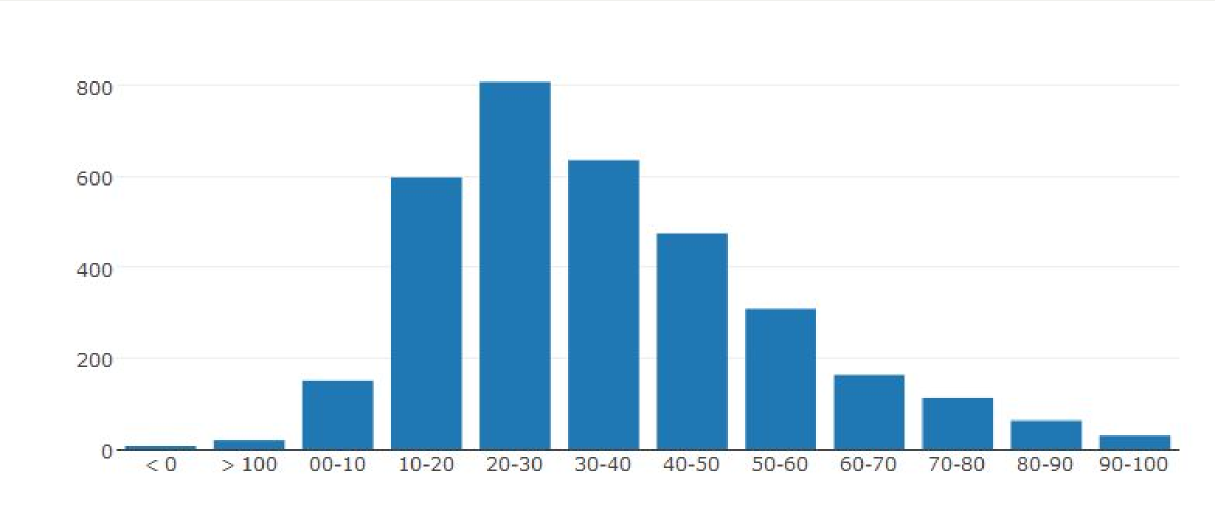


Nr. Clients by district population size

Exploratory data analysis

Clients address:

- Balance amount
  - Most clients have an average balance (in this dataset) (between 10k-50k)
  - The remaining few clients either have large sums of money or small sums of money

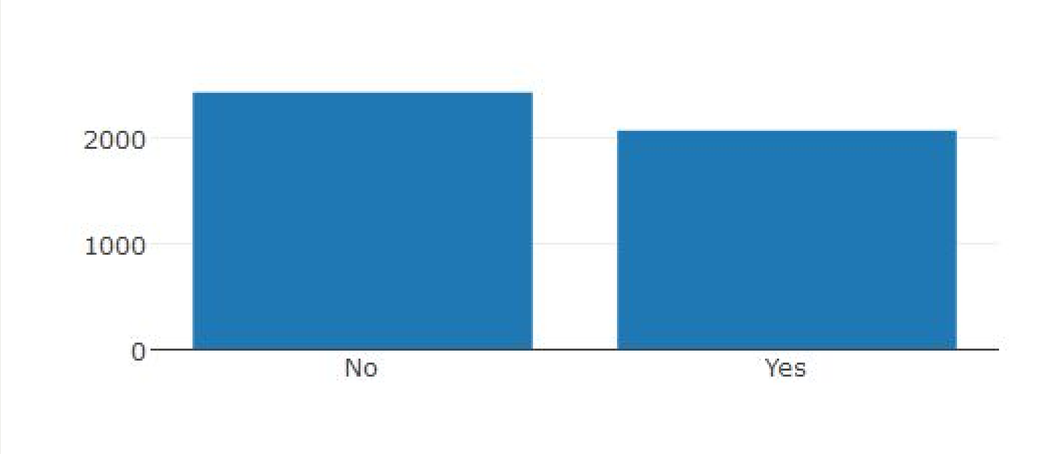


Clients balance amount

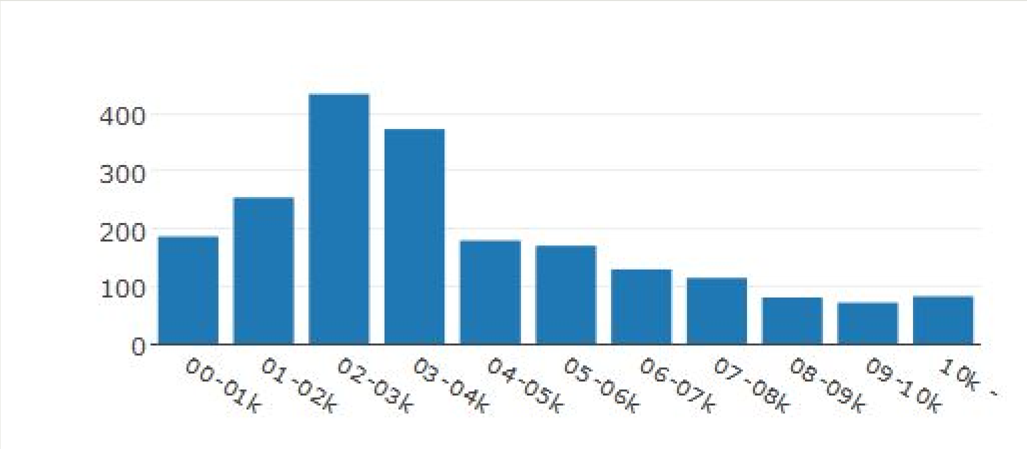
Exploratory data analysis

Client type:

- Household
  - More than half of the clients don’t pay household
  - Those who do, mostly pay between low to average amounts monthly, and few pay high amounts



Pays household?



Household amount

Descriptive problem

Problem definition

The group decided to analyse the data about the **clients of the bank**. The information analysed varied from **personal details**, such as gender, age and district, to **information from their accounts**, based on the transactions they made, such as average balance and number of transactions.

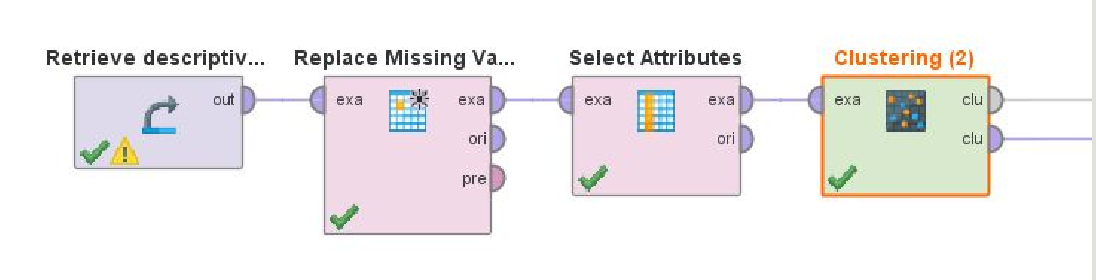
The main focus of the descriptive problem was to know **how does the clients' balance vary according to age**, and also **how much did it vary** during the timespan of analysis of the transactions.

Data preparation

- Data was prepared using **R**
- Filter clients by gender
- **Fix** women **birth date**
- Setting birth numbers as date type
- Getting **household amount**
- Getting **average account balance**
- Getting **number of transactions** of the accounts

Experimental setup

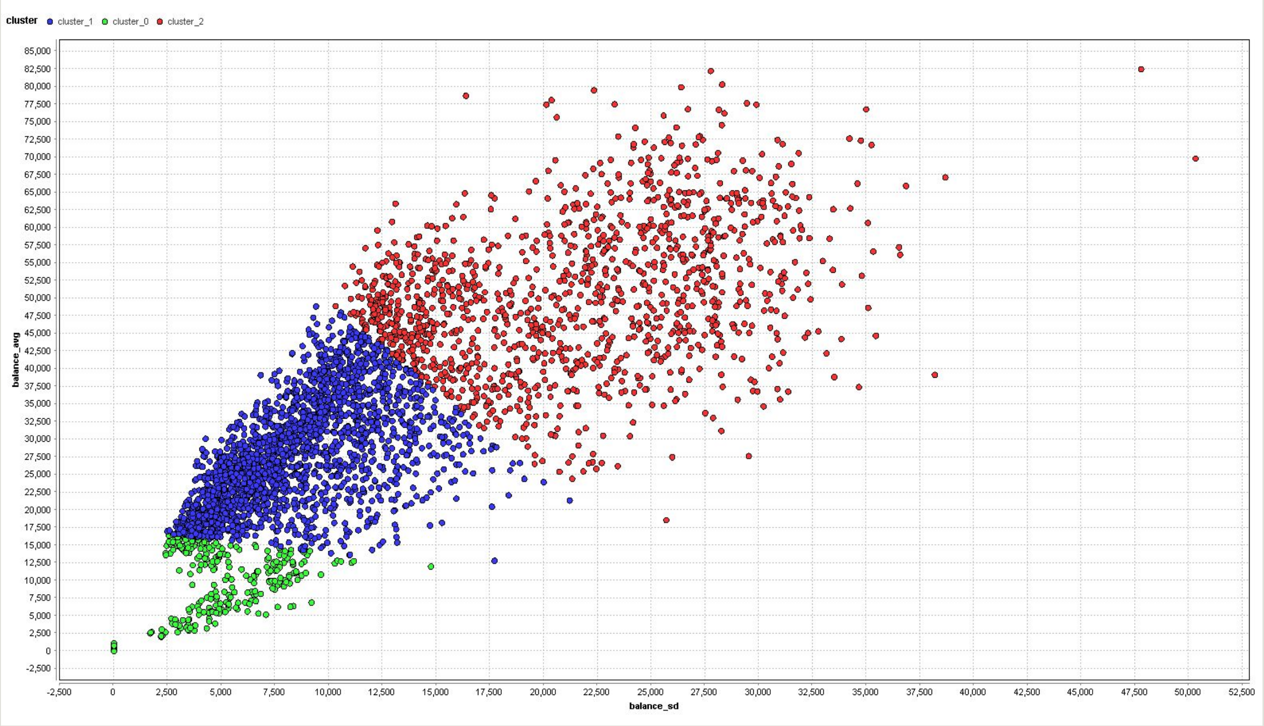
- Use of RapidMiner
- Retrieve dataset
- Select attributes
- Use *k-Medoids* algorithm for clustering (k=3)



Results

Data used in the clustering process:

- Average account balance
- Account balance standard deviation
- Client birth year



Cluster division:

- **Cluster 0:** This cluster contains mostly **younger** clients, with **low balances**, and also clients with no transactions.
- **Cluster 1:** The second cluster includes **older** clients than the previous (in general, more than a decade older), and with a **higher ratio between the average balance and SD** than cluster 0.
- **Cluster 2:** This last cluster includes clients even **older** than the previous (the centroid indicates almost one decade older), and with **more variation of the balance** when comparing to cluster 1.

Attribute	Cluster 0	Cluster 1	Cluster 2
Balance avg	0	30870	34805
Balance sd	0	11709	21580
Birth year	1974	1962	1953

Predictive problem

Problem definition

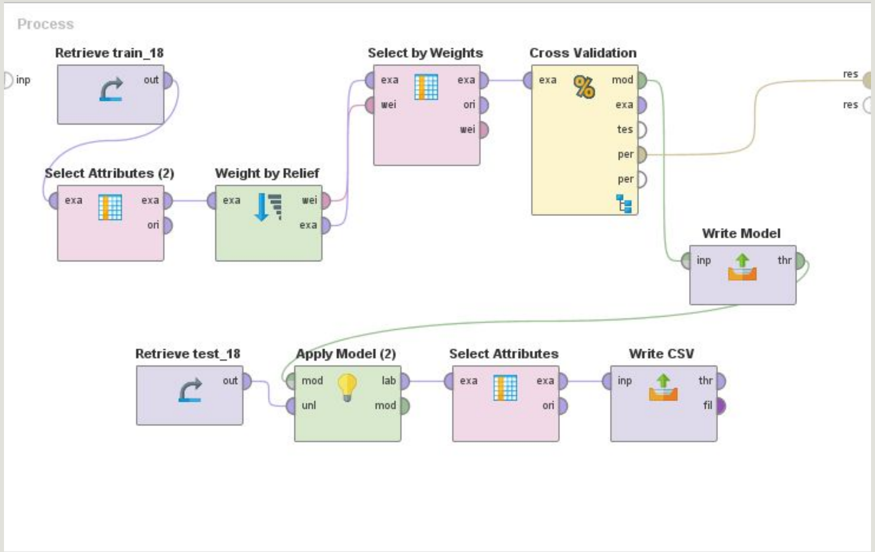
- The predictive task’s goal is to train a model that predicts whether a client should or should not be granted a loan.
- To train this model, the data available is the same used in the descriptive problem (all the information about the clients and their accounts and transactions in the past months), and also the loan details, such as duration, and total amount.

Data preparation

- Transform *strings* to *integers*
- Set missing values
- Get clients’ gender through birth date, and normalize dates
- Get unemployment *average* and *difference* (between ‘95 and ‘96)
- Get crimes *average*, *difference* and *ratio* (between ‘95 and ‘96)
- Transform dates to *date* type

Experimental setup

- Use RapidMiner
- Retrieve dataset
- Select features by Relief weight
- Perform Cross-Validation
- Train with deep learning
- Apply the model
- Check performance
- Use model for labeling test dataset

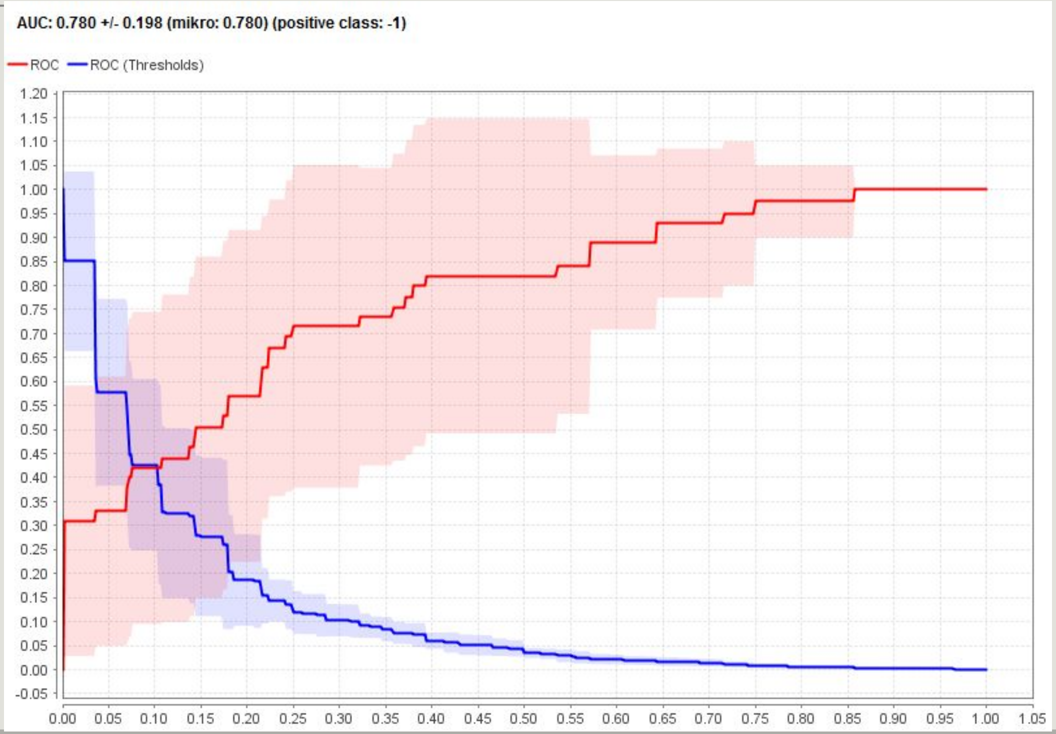


See annexes for further details

Results

The resulting model of the process was always evaluated by analysing each **class' precision and recall**, and also the **AUC** (Area Under the Curve), the evaluation metric used in Kaggle's competition.

The training data had an AUC of 78% and an accuracy of 86%. The final score from Kaggle was 75.59%.



Conclusions, Limitations and Future work

- It is important to filter not only the datasets, but also the features to use in the training process
- Some operators could be beneficial to use, such as optimise features or parameters but take too long to finish
- Some algorithms require specific types of data, because they can’t deal with some of them
- Despite having an overall predictive score of 75%, we feel this could be improved by using other features for training, which perhaps were not explored in the time given

Banking case study

André Pires - ei12058@fe.up.pt

João Bandeira - ei12022@fe.up.pt

Annexes

Features used for predictive problem, with best score:

- ratio_crimes
- ratio of urban inhabitants
- unemploymant rate '96
- average salary
- unemploymant rate '95
- no. of entrepreneurs per 1000 inhabitants

- amount
- balance_diff
- region
- balance_sd
- payments
- type_card
- frequency
- balance_avg
- Birth_number
- no. of inhabitants
- no. of committed crimes '95
- gender
- no. of committed crimes '96

Annexes

Main setup for predictive problem:

- Retrieve data
- Weight by relief - w/ normalize
- Select by weights - >= 0.02
- X-validation
  - Deep learning
    - ExpRectifier
    - Layers - 25/15
    - Epochs - 50
    - Compute variables importances
    - Early stopping
  - Performance
    - AUC - 78%
    - accuracy - 86%

Omitted default parameters