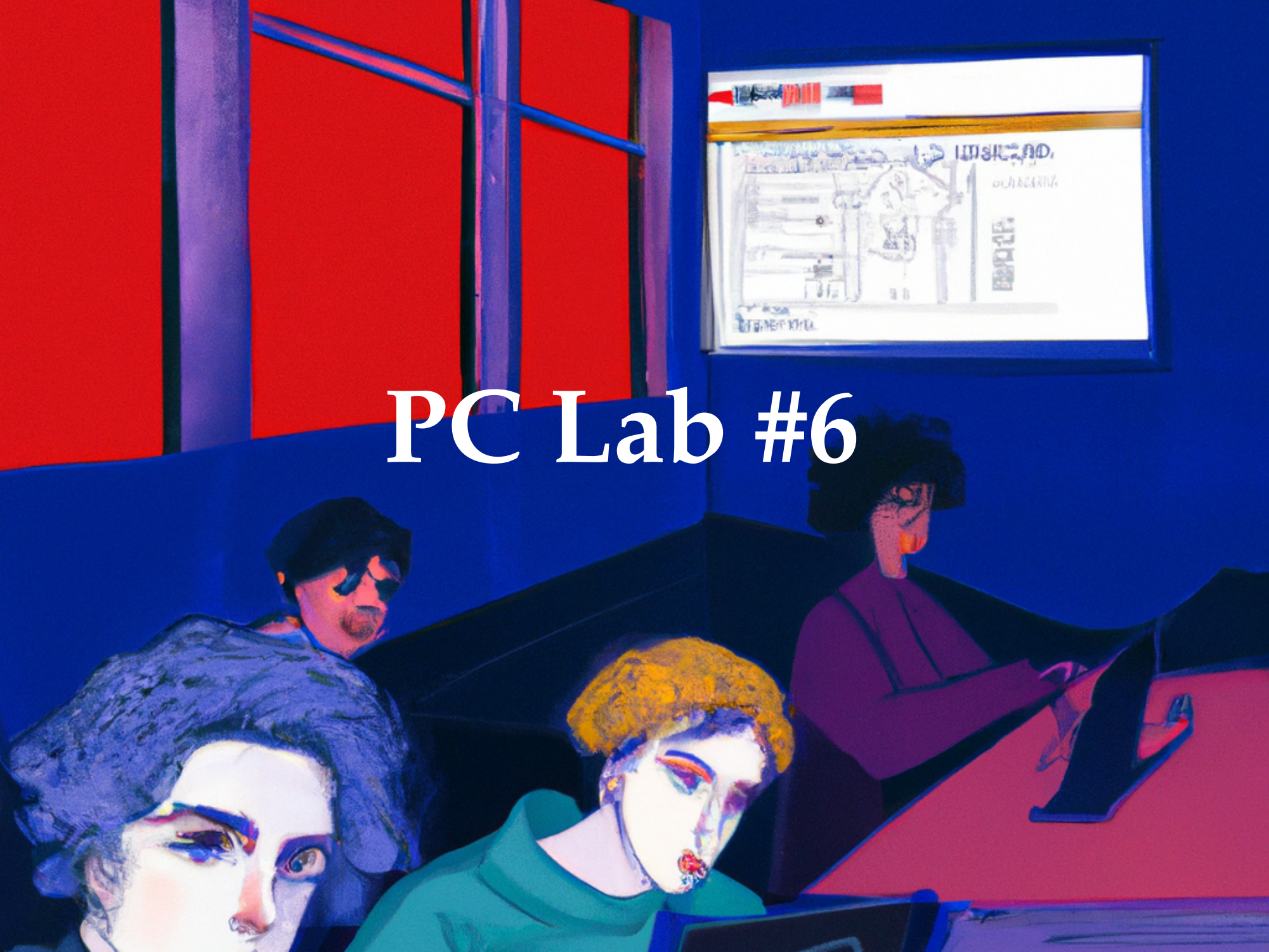
# 20598 – Finance with Big Data

## PC Lab #6: Predicting Credit Scores with ML (Week 7)

Clément Mazet-Sonilhac
clement.mazetsonilhac@unibocconi.it

Department of Finance, Bocconi University

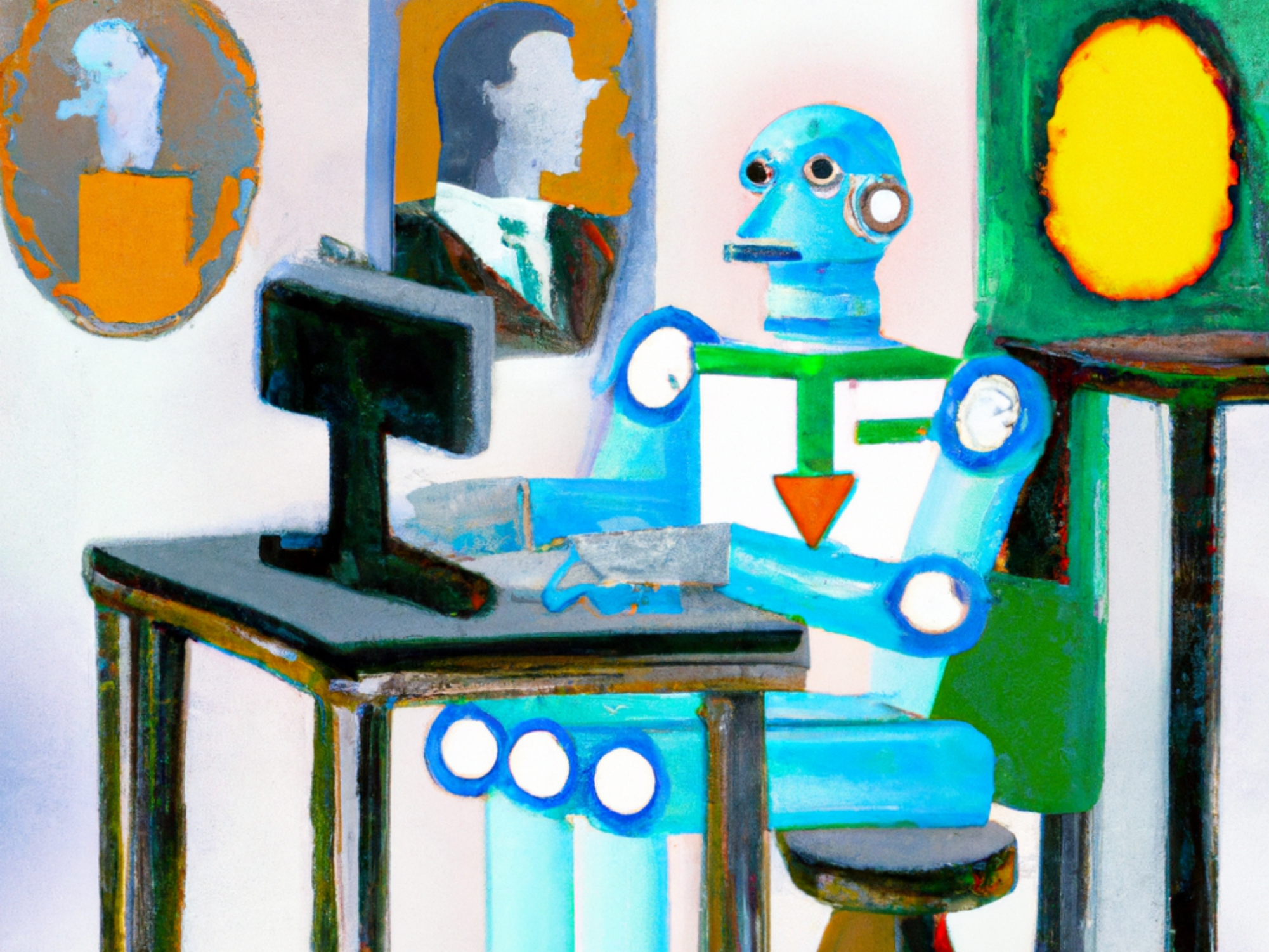

PC Lab #6

# PC Labs Grading

- PC Labs solutions are submitted as Jupyter Notebooks, via email
  - Email title : PCLab#6 - Group X - Name1 Name2 Name3
  - Your Jupyter Notebook starts in the same way (same .ipynb name)
  - Tell me (again) how long did it take

# PC Labs Grading

- PC Labs grade will depend on :
  - Your ability to submit it before the deadline
  - The quality of your code (comments, readability, use of functions, etc.)
  - The structure of the Jupyter Notebook : well organized, explain what you are doing and why
  - Your ability to complete the tasks and innovate
  - You should maybe produce less, but more *useful* outputs

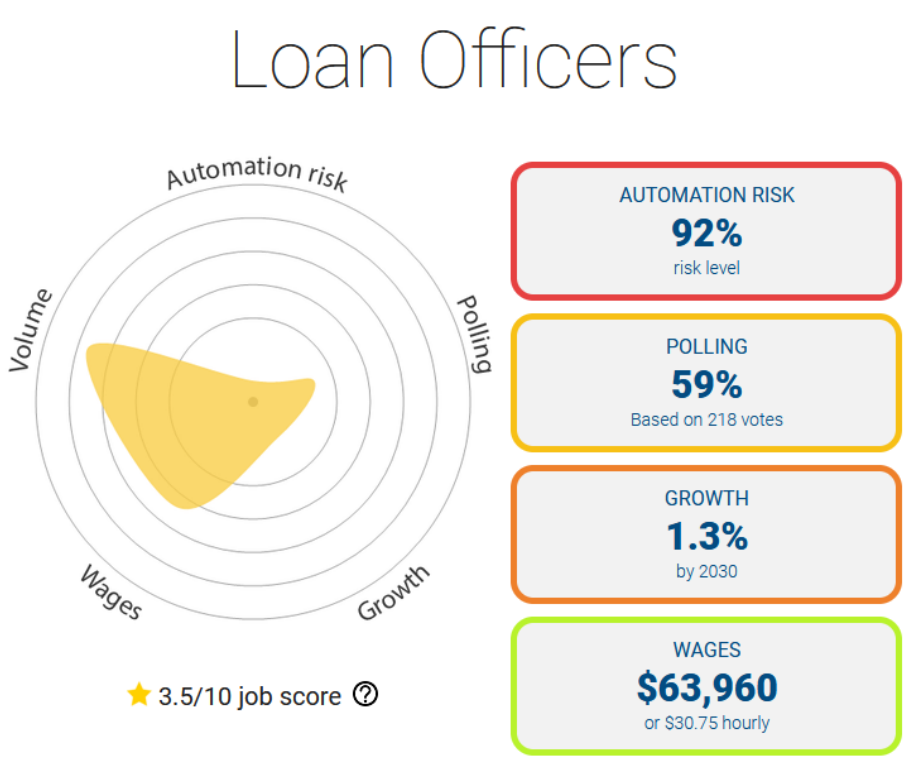


# Goals

- Manipulate and visualize balance-sheet and income statement data
- Understand which are the most important variables to predict firms' credit scores
- Try to predict credit scores as well as a loan officer

# Big picture context

- You're the new intern at JP Morgan
- JPM wants to go full digital: closing bank branches and replacing its loan officers by AI
- Needs an algorithm to screen firms : classify them w.r.t. their default's risk



# Task #1 : Basic manipulation and descriptive statistics

- Import the `Data_PCLab5_train_test.csv` data (you'll use `Data_PCLab5_pred.csv` later)
- Describe the sample :
  - What are the different credit scores in the data ? What does it mean ?
  - Are there any useless variables ?
  - Which are the key findings relative to the variables' distributions ?
- Plot the distribution of the variables against the different credit scores
- Deal with the outliers within the sample, if any.
- Create the age of the company and other variables that may be useful (ratios, etc.). Hint : use `year` and `year_creation`

# Task #2 : Feature selection

- The Head of Risk Department comes to your office asking for the factors which are the best predictors of credit scores
- Perform a feature selection over the sample. How do you interpret the results ?
- Hint: There are functions in `sklearn.feature_selection` that do exactly that, such as `SequentialFeatureSelector` or `SelectFromModel` ($\rightarrow$ better tech might be available, do whatever you think works best !)

# Optional Task #2 : Feature importance

- Considering predictive tools such as Decision Trees or Random Forests, one can retrieve the feature importance of the estimators, i.e., calculate a score for all the input features for a given model — the scores simply represent the importance of each feature.

- Pick and run a model to retrieve the features importance of each input

- Depending on the model you're running, it may happen that – if you run many times the same model – you may obtain slight different results. What is happening ? How could you avoid that ?

- What is the interpretation behind this results ? Are there any differences among the features selection and the features importance approaches ?

# Task #3 : Predict credit scores

- Train and test a machine learning model of your choice over the train sample
- Predict the credit score for the firms of the prediction sample `Data_PCLab5_pred.csv`.
- Comment : Andrea (TA) will evaluate your predictions based on the True Positive Ratio. Send a .txt file with your predicted credit scores.
- The bank policy is to accept all firms with a score above 4. What share of the loan applications are you accepting ?

# Task #4 : Further approaches

- The Head of Risk Department comes back to your office once you've just finished predicting the credit score
- Tells you that he doesn't care that much about the exact risk category, he just wants to have a general idea about firms' risk profile.
- Modify the credit scores to create broader categories
- Re-train your model(s). Does your TP ratio improves

# Task #5 : Welfare Implications : Beyond Accuracy

- Accuracy alone says little about the *social* impact of AI adoption
- It is not particularly harmful if a very good firm (3++) is classified as merely good (3), or vice versa.
- It is, however, socially costly if good firms (3 or 4+) are denied credit (classified as bad, like 5+ or worst), or if bad firms are selected $\Rightarrow$ Algorithmic discrimination.
- How frequently does your models badly misclassify firms ?
- How could you modify your performance criterion (e.g. the True Positive ratio) to incorporate this dimension ?

# Packages you may need

- Among others : `sklearn`, `catboost`, `xgboost`, `tensorflow`, `pytorch`, `sklearn.feature_selection`, `etc.`