Project 2

The goal is to build a model that combines two advantages: has high accuracy and is parsimonious, i.e., is based on a small number of variables.

Imagine that the company you work for collaborates with a utility provider on a new energy-saving initiative. The utility company wants to identify households that are likely to exceed a predefined electricity usage threshold next month. These households will then be offered a personalized energy-efficiency support package.

However, collecting detailed data about every customer is expensive. Some data (e.g., total monthly electricity use) is easy and cheap to access, while other variables (like appliance-level consumption, temperature sensor data, or occupancy estimates) are costly to acquire or compute.

Your team has been asked to build a model that can accurately identify households that will likely exceed the threshold—but using as few costly variables as possible.

Data

We have **5000 historical training data**. Each client is described with **500 variables** (variables are anonymized).

Your task is to build a model that predicts which customers in the test set took advantage of the offer.

The training data:

- x_train.txt - variable matrix for training data for 5000 households
- y_train.txt — labels (value `1` = usage above threshold, value `0` = usage within acceptable range)

Test data:

x_test.txt - variable matrix containing information about 5000 households.

Task:

Your goal is to build a model on the training data and then identify **1,000 households** in the test set that you predict will exceed the energy usage threshold next month.

Why 1,000? The utility company has limited capacity and can only offer the energy-saving package to **1,000 households per month**.

In addition to submitting your predictions, you must **indicate the variables** your model uses.

Technically, you should evaluate at least 5 strategies of building models: as one strategy we consider one combination of machine learning algorithm (such as gradient boosting or logistic regression, **we**

do not consider different hyperparameter configuration as different algorithm) and feature selection method.

Project evaluation (30 points)

Score – 15 points

- For details how models are evaluated, please see next section.
- Final score will be assigned according to the leaderboard of model performance attained by all teams.

Report – 8 points

- The investigated strategies and the finally selected model should be described in the report.
- The report should include key information to enable reproduction of the solution and, in addition, the results of the experiments arguing the design decisions made.
- **Maximum number of pages of the report: 5 pages**
- The report should be prepared in Latex

Presentation – 7 points

- Presentation will be given during project meeting in front of the whole group, so you should prepare slides.
- Presentation should take max 10 minutes.
- Attendance during the presentation is obligatory to get points for the presentation.

Model evaluation

- The performance of your model will be scored as follows:
- For each correctly identified household (i.e., one that did indeed exceed the threshold), the utility company pays you EUR 10.
- For each variable used in your model, you must pay EUR 200 to simulate the cost of acquiring and processing that data.

Example 1:

- Your model correctly identifies 850 out of 1,000 households.
  You used 12 variables.
- Reward: 850 × 10 EUR = EUR 8500
- Variable cost: 12 × 200 EUR = EUR 2400
- Final score = EUR 8500 − EUR 2400 = EUR 6100

Example 2:

- You only correctly identify 300 out of 1,000 households, but use just 2 variables.
- Reward: 300 × 10 EUR = EUR 3000
- Variable cost: 2 × 200 EUR = EUR 400
- Final score = EUR 3000 − EUR 400 = EUR 2600

The higher the score, the better, because it means a higher reward.

Additional remarks:

1. You can choose any programming language (Python/R are preferred), as long as the resulting files are in the correct format.
2. Projects are prepared in groups of 3 students.

How to submit a solution?

Your solution should be contained in two files:

- File STUDENTID_obs.txt should contain 1000 indexes of customers from testing data to whom you want to send the offer.
- File STUDENTID_vars.txt should contain the indexes of variables used by the proposed model.

STUDENTID is a student id of the first student from the group.

Please see example files: 123456_obs.txt and 123456_vars.txt. The submitted files must be in the same format.

Please save all results to the ZIP file, named STUDENTID.zip. The archive should contain the following files: **STUDENTID_obs.txt, STUDENTID_vars.txt, report.pdf presentation.pdf** (ppt, pptx, etc.) and folder named **code** with source codes.
Please upload your solution using the task assigned in the MS Teams channel.

Deadlines

- Solutions should be submitted until **2.06.2025 23:59**
- Final presentations:
    - **5.06.2025 - Group 3 & 4**
    - **12.06.2025 - Group 1 & 2**

Meeting schedule

Group 1 & 2

24.04.2025

15.05.2025

29.05.2025

Group 3 & 4

08.05.2025

22.05.2025

29.05.2025

If you have any questions, please send us an e-mail: katarzyna.woznica@pw.edu.pl, adam.majczyk.stud@pw.edu.pl, dawid.pludowski.stud@pw.edu.pl