SML 2024, big programming assignment 2

Dorota Celińska-Kopczyńska, Adam Izdebski, Krzysztof Gogolewski

The goal of this assignment is to provide as precise as possible prediction.

Data description Organs such as the pancreas are made up of many types of tissues, which in turn are made up of many types of cells. Within the pancreas, we can distinguish cells typical only for this organ, such as cells alpha or beta, but also cells related to blood supply or the immune system. The data in this problem comes from multimodal single-cell sequencing (multimodal single cell RNA sequencing, scRNA-seq). The use of scRNA-seq allows samples to be studied in high resolution and to separate cells of different types from each other. Especially, we are able to compare pathological cells taken from cancer patients with healthy patients' cells. In the multimodal scRNA-seq technology, we obtain two types of readings for each cell:

- **Counts of RNA transcripts** corresponding to the expression (activity) of genes in a given cell;
- **The amount of surface protein (protein abundance)**, which is directly related to the type of the cell.

ScRNA-seq experiment return matrices in which for each cell we assign the RNA signal from many thousands of genes (in our task X) and the signal from several dozen several surface proteins (in our task, for simplicity, we chose a single protein CD36, y).

According to the central dogma of biology, we know that genetic information flows from RNA to proteins. Thus, we should expect a correlation between the amount of protein and the expression of the gene coded by that protein. For technical and biological reasons, this relationship often degenerates. The problem in this task is to predict the signal from surface proteins on the basis of gene expression. Predicting the protein abundance signal is crucial to most publicly available collections for which only RNA matrix is available. Analysis of the signal about gene expression and the number of surface proteins significantly facilitates the identification and naming process of the cells in the sample.

The data comes from the bone marrow of human donors. The collected cells are mostly cells of the immune system. Correct identification of T cells based on both types of reads in a set of this type could be the basis for developing targeted therapies cancer (for the curious: CAR T cell therapy).

Dataset: On the Moodle, you will find files in .csv format. There are three files:

- `X_train.csv` and `X_test.csv`, containing RNA matrices. Each row corresponds to a cell, the column is the gene, while the values are the levels of expressions. Those files contain our potential *independent variables* (predictors).
- `y_train.csv`, corresponding to the amount of a certain type of surface protein in cells, our *dependent variable* (response) related to the predictors from `X_train.csv`.

Onwards, we will treat data in `X_train.csv` and `y_train.csv` as a training set, and data in `X_test.csv` as a test set.

Desired output: You are supposed to submit by Moodle:

- a report in .pdf format with the summary of the results and your commentary (file name template: `IndexNo._report.pdf`)
- a python file or jupyter notebook containing code (accompanied by a basic commentary) that you used to obtain the results. Please make sure your notebook opens and works in Google Colab, it will not be graded otherwise (file name template: `IndexNo._code.ext`, where `ext` is the appropriate extension)
- Prediction results on test data (see task 4) in the form of a .csv file containing the column `Id` with the numbers of observations and the column `Expected` with the values of the prediction separated with ";" (file name template: `IndexNo._prediction.csv`)

They will be graded by lab assistants of respective groups. Make sure your report is self-standing – it contains necessary information without the need to read carefully your code.

Deadline: 25.01.2025 11:59PM CET

Total points to obtain: 20

Details of grading: In parentheses, we provide the maximum number of points for each of the tasks below. The assessment of tasks 1 to 4 will take into account:

- whether the goals of the task were met,
- the quality of the report (visualizations, clarity of the text, description of the results),
- the quality of the code used for this purpose — it should be easy to follow and reproduce the results.

When in doubt, ask your lab assistant for the details of the grade.

Contents of the task

1. **Exploration** *(2p.)*
   (a) Check how many observations and variables are there in the loaded training and test data. Take a look at the types of the variables and, if necessary, make the appropriate conversions before further analysis. Make sure the data is complete. (0.5p.)
   (b) Investigate the empirical distribution of the response variable (e.g., present some basic statistics, attach a histogram or graph of the density estimator to the analysis). Discuss the results. (0.5p.)
   (c) Compute the appropriate correlation coefficients between the predictors and the response variable. Visualize the results in the compact way (we suggest violin plots). Select the 100 independent variables that are the most correlated with the response variable. Calculate the correlations for each pair of these variables, and provide a compact visualization of your choice (e.g., a heatmap) in search of multicolinearity. Discuss the results. (1p.)

   Note: the selection of variables described here is only for the purposes of this task, the analysis described in the following tasks should be performed on **the full** training set

2. **ElasticNet** *(5p.)* The first model to train is *ElasticNet*. During the lecture, we introduced its special cases: ridge regression and lasso.

   (a) Provide a short description of the ElasticNet model. Present the parameters that are estimated and the optimization function. Provide some information on the hyperparameters, especially for what values of hyperparameters the model reduces to ridge regression or lasso. (1p.)

   (b) Define a *grid* of hyperparameters based on at least three values for each hyperparameter. Make sure that you included the hyperparameter configurations corresponding to the ridge and lasso regression. Use cross-validation to select appropriate hyperparameters (the number of subsets used in cross-validation is up to you to decide, but you have to justify your choice). (3p.)

   (c) Specify the training and validation error of the model (the result should be averaged over all subsets from the the cross-validation). (1p.)

3. **Random forest** *(6p.)* In this part of the project, you train the random forest model and compare its performance with the ElasticNet model from the previous task.

   (a) From the many hyperparameters that characterize the random forest model, choose three different ones. Define a three-dimensional grid of hyperparameter combinations to be searched and select their optimal (in the context of the prediction) values using cross-validation. The data division used for cross-validation should be the same as in the case of ElasticNet model. (3p.)

   (b) Provide a tabular summary of the cross-validation results of the methods in both models under consideration. (This comparison is why we make you use the same divisions.) Specify which model seems to be the best (justify your choice). Include a basic reference model for the comparison, which assigns the arithmetic mean of the dependent variable to any independent variable values. (3p.)

4. **Prediction on a test set** *(7p.)* This part of the project is open-ended. Use the training data to choose the "best" predictive model, and then use it to predict values of the dependent variable in the test set. The methods of selecting and building the models, as well as the motivations behind such choices, should be described in the report. The number of points you earn will depend on the quality of prediction, measured by the root of the mean squared error, RMSE.

   Tip: We recommend trying various techniques to support machine learning (e.g. appropriate selection of a subset of independent variables, variable transformations or dimension reduction).

   Scoring details:

   (1p.) – for an error lower than that of the basic reference model described earlier.

   (2p.) – for an error lower than the one from the ElasticNet model trained by the lab assistants.

   (4p.) – this bonus is calculated as $\frac{1}{2}\lfloor 8\widehat{F}(e)^3 \rfloor$, where e is the student's -RMSE from prediction (error), $\widehat{F}$ is the empirical distribution of errors of all reported predictions, and $\lfloor \cdot \rfloor$ is the integer part. Bonus is calculated based on all submitted solutions.