Machine Learning Project, 2024/2025

Margarida Silveira, Catarina Barata, Alexandre Bernardino

September 13, 2024

1 Introduction

The project is split into two parts: regression and image analysis, and each part comprises two problems with deliverables for evaluation.

The programming language used in the project is Python because it has powerful libraries for building Machine Learning applications, and it is a widespread language in industry. The first problem session in the Machine Learning course is devoted to the basics of programming in Python.

The Machine Learning project should be done in groups of two. Each group should work alone. The exchange of ideas or software is not allowed and may invalidate the work.

2 Student Evaluation

Student evaluation will take into account:

- interaction with the teacher during the laboratory sessions - this is an individual assessment of each group member, thus both students must actively solve the project and be acquainted with the work;
- statistical performance of the algorithms developed by the group, evaluated on independent data (test) sets - 25% of the lab grade;
- final report (maximum length of 10 pages, font size 12 pt) describing the methodologies adopted by the group, including figures and statistical evaluation - 75% of the lab grade

Each group should submit the output of the proposed algorithms using an independent data set (test set) for each of the four questions until the end of the deadlines, as well as the Python code used to solve them. The outputs will be compared with the ground truth by the teaching team, and the results (a leaderboard with the scores achieved by each group) will be published on the Fenix web page. **Attendance to the laboratory sessions is mandatory, and submissions from groups who fail to do so will not be evaluated.**

3 Datasets and Project Submissions

The training and test data for each problem will be available through the course webpage on Fenix. For each problem, the students will have access to a training set (feature vectors and real outputs) and a test set (just the feature vectors).

All data will be stored in *numpy* (.npy) format.

There are no restrictions on the number of machine learning models that students can research and try, but **the minimum is two models to be discussed in the report**. However, in each of the project questions, they must **pick only one model** to apply to the test set and perform the submission. Project submissions should be made through Fenix, in the appropriate section. For each question, the students must submit a zip file containing:

- the output of their model of choice on the test set; and
- the Python code for all the experiments.

The predictions must respect the same format as that of the output within the training set. The teaching team will assess performance on the test set using appropriate statistical metrics. The scores achieved by each group will be made available on a leaderboard on the course webpage.

4 Part 1 - Regression with Synthetic Data

4.1 First Problem - Multiple Linear Regression with Outliers

The first problem is a linear regression problem that illustrates some characteristics of real applications.

In a coastal location, a high incidence of toxic algae that endangers marine life has been observed. A group of researchers is trying to verify a potential linear (affine) relationship between a set of 5 independent variables $x_1, \cdots, x_p,\ p = 5$ and a dependent variable y:

$$y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i \tag{1}$$

Independent variables are daily averages of air temperature x_1, water temperature x_2, wind speed x_3, wind direction x_4 and illumination x_5, and the dependent variable y is the concentration of toxic algae in water samples.

During several consecutive days, measurements of both the dependent and independent variables were taken with real instruments. Those variables are often affected by two main types of noise: (i) instrument noise η, which researchers have found to be well approximated by a zero mean white Gaussian noise, and (ii) human error ξ in handling the instruments and the resulting data, which researchers cannot precisely model but usually results in larger errors than instrument noise. Human error has been found to affect about 25% of the samples of the dependent variables, but it is negligible in the measurement of the independent variables. Therefore, the real data samples acquired in the process can be modeled as:

$$\hat{y}^{(j)} = \beta_0 + \sum_{i=1}^{p} \beta_i x_i^{(j)} + \eta^{(j)} + \xi^{(j)} \tag{2}$$

$$\hat{x}_i^{(j)} = x_i^{(j)} + \eta_i^{(j)} \tag{3}$$

where j is the sample (day) index.

A dataset of $n = 200$ samples was acquired

$$\mathcal{T} = \left\{(\tilde{\mathbf{x}}^{(1)}, \tilde{y}^{(1)}), \cdots, (\tilde{\mathbf{x}}^{(n)}, \tilde{y}^{(n)})\right\} \tag{4}$$

with $\tilde{\mathbf{x}}^{(j)} = \left[\tilde{x}_1^{(j)}, \cdots, \tilde{x}_5^{(j)}\right]^t$

To estimate the parameters β_i, use linear regression with outlier removal and regularization techniques that you consider suitable for this problem. For outlier removal, you can read the introductory chapter of [Wilcox, 2023] (Sections 1.1 to 1.5). If you want to explore this topic further, you can study more advanced methods in [Fischler and Bolles, 1981], [Rousseeuw and Hubert, 2017] and [Björklund et al., 2022].

After estimating the parameters of the linear model β_i, the students should compute the predictions for the provided test set and save the resulting predictions of the dependent variable $\hat{y}^{(j)}$ in a *numpy* vector of 200 elements in the same order as the original data.

The comparison between the predictions $\hat{y}^{(j)}$ and the true values $y^{(j)}$ will be done by the teaching team using the SSE metric.

Note: Find ways to visualize the data and intermediate results of your computations that allow you to confirm the proper implementation of your algorithm.

4.2 Second Problem - The ARX model

The ARX model (Auto Regressive with eXogenous input) enables the representation discrete input-output dynamical systems described by linear difference equations, in which the observation noise in one sample is independent of the noise in other samples:

$$y(k) + a_1 y(k-1) + \cdots a_n y(k-n) = b_0 u(k-d) + \cdots b_m u(k-d-m) + e(k) \tag{5}$$

In the equation, k is the discrete time variable, y is the output sequence, u is the input sequence, e is the noise sequence, and $n \geq 0$, $m \geq 0$, and $d \geq 0$ are model order parameters.

To see how the problem can be formulated as a linear regression, it is more explicit to write the model (5) in regressor form:

$$y(k) = \phi(k)^T \theta + e(k) \tag{6}$$

where

$$\phi(k) = [y(k-1), \cdots, y(k-n), u(k-d), \cdots u(k-d-m)]^T$$

and

$$\theta = [-a_1, \cdots, -a_n, b_0, \cdots, b_m] + e(k)]^T$$

ARX models are used extensively for approximating dynamical systems without feedback loops. With such a model, it is possible to design controllers with model-based techniques.

It is assumed that the system is initially at rest ($y(k) = 0$, $\forall k < 0$) and that the discrete-time indexes range from 0 to $N-1$. For given n, m, d, and a sequence of input-output data of size N, to estimate the parameters a_i and b_j, the model equation is rewritten in matrix form:

$$Y = X\theta \tag{7}$$

with $Y = [y(p), \cdots, y(N-1)]^T$, $X = [\phi(p), \cdots, \phi(N-1)]^T$, $p = \max(n, d+m)$.

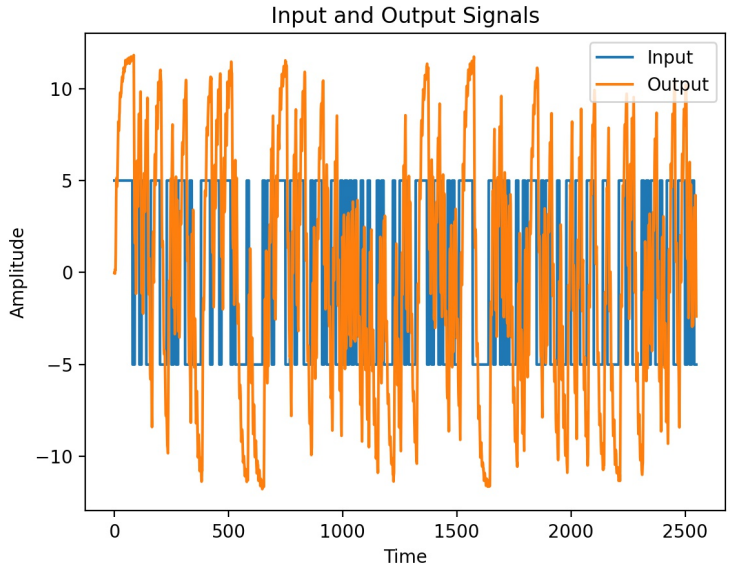

Figure 1: The input and output data of this problem

A dataset of 2550 samples (see Fig. 1) with inputs u and outputs y was acquired :

$$\mathcal{T} = \{(u(0), y(0)), \cdots, (u(2549), y(2549))\} \tag{8}$$

The data set was then divided into two parts: the training set $\mathcal{T}_{train}$ with the first 2040 samples and a test set $\mathcal{T}_{test}$ with the remaining 510 samples.

$$\mathcal{T}_{train} = \{(u_{train}(0), y_{train}(0)), \cdots, (u_{train}(2039), y_{train}(2039))\} \tag{9}$$

$$\mathcal{T}_{test} = \{(u_{test}(0), y_{test}(0)), \cdots, (u_{test}(509), y_{test}(509))\} \tag{10}$$

The students have access to the training set $\mathcal{T}_{train}$ for training the model, and to the input data of the test set u_{test}. The learned model should be excited with u_{test} iteratively using the regressor form (6), to obtain predictions of the test output $\hat{y}_{test}$. For model evaluation, the last 400 samples of $\hat{y}_{test}$ will be compared with real samples of y_{test} (not available to students), using the SSE criterion. Therefore, students should submit a file with only the last 400 samples resulting from the predicted

Figure 2: Mars images with (top row) and without (bottom row) craters.

$\hat{y}_{test}$, corresponding to the time indexes 2150 to 2549 (inclusive) of the original data or the indexes 110 to 509 of the test data.

Try different values for the model order parameters n, m, and d and choose those that are more likely to give the best results in the test set. You may assume $n < 10$, $m < 10$, $d < 10$.

Note: Note that linear regression methods do not impose any constraints on the stability of the dynamical system - some solutions may lead to unstable systems that you should discard.

5 Part 2 - Image Analysis

The project's second part is devoted to analyzing satellite images of Mars showing meteorite impact craters. Impact craters are an essential source of information about Mars's geology and surface characteristics, and automatizing their detection in the images is essential.

Our first goal is to distinguish between images with and without craters. The second goal is to segment the craters in the images where they are present.

5.1 First Problem - Image classification

The first classification task is binary, where we want to create a model that identifies the type of image. For this task the label is either 0 (without crater) or 1 (with crater). The images are 48x48 matrices. A few examples are shown in Fig. 2.

Note that the dataset is imbalanced since there is a big difference in the number of images with and without craters in our dataset. In addition to the training set, an extra dataset, without labels, is provided. This dataset can be used in whatever way the students find helpful. The metric the teaching team uses to evaluate the submissions for this task is the F_1 score.

5.1.1 Suggestions

- investigate which are the most suitable classifiers for image tasks;
- investigate ways to deal with imbalance in classification tasks.
- explore creative ways to take advantage of the extra dataset;

5.2 Second Problem - image segmentation

The second task consists of segmenting the craters in the images where they appear. The image pixels corresponding to the crater area are labeled with "1", whereas the background pixels have the label "0".

This task can be regarded as a pixel classification task, where we want to classify each pixel in a 48x48 pixel image. Two data formats are provided for this purpose. In the first format (format a), for each pixel, the set of pixels in a 7x7 neighborhood surrounding are given as features. In the second format (format b) the entire input and corresponding segmentation images are given and can be used in a structured prediction task.

Figure 3: A Mars image and its ground truth segmentation mask.

The segmentation masks used for training have been manually generated with circles positioned in the approximate crater location, as illustrated in Fig. 3.

Note that this dataset is also imbalanced since there is a big difference in the number of pixels within the craters and in the background.

The metric the teaching team uses to evaluate the submissions for this task is the Balanced Accuracy.

5.2.1 Suggestions

- do not use thresholding or circle detection methods, and instead use the classification methods learned in class;
- check functions **extract_patches_2d** and **reconstruct_from_patches_2d** from the skimage toolbox.

References

[Björklund et al., 2022] Björklund, A., Andres Henelius, E. O., Kallonen, K., and Puolamäki, K. (2022). Robust regression via error tolerance. *Data Mining and Knowledge Discovery*.

[Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*.

[Rousseeuw and Hubert, 2017] Rousseeuw, P. J. and Hubert, M. (2017). Anomaly detection by robust statistics. *WIREs Data Mining and Knowldge Discovery*.

[Wilcox, 2023] Wilcox, R. R. (2023). *A Guide to Robust Statistical Methods*. Springer Cham.