LINEAR REGRESSION
Assignment 2 of Machine Learning I, A.Y. 2021/2022

Marco Macchia
DIBRIS - Dipartimento di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi
Università Degli Studi di Genova

This report describes the second assignment of Machine Learning I, what were the requirements and what are the obtained results.

I. INTRODUCTION

REGRESSION means approximating a functional dependency based on measured data. *Linear* means that the approximation uses a first order polynomial function to approximate the data. This implies that, in the case of a one-dimensional problem, the approximation will be a line.

In machine learning, the *linear regression* is then a good approximation method to predict the value of an observation. In particular, linear regression aims to find a correlation between the *target value* t of an observation x.

II. THEORY OF LINEAR REGRESSION

A. One-dimensional problem

Keeping in mind that the *linear regression* defines a linear approximation, the problem can be formalized in the following way:

Define a linear function such that, given an input x (observation), computes the target t that best approximates the real value y.

$$t \approx y, \ where \ t = wx \tag{1}$$

w is the linear regression parameters, and associate with a linear function the input and the target.

Suppose there are two different observation x_0 and x_1. Arranging the equation 1 it is simple to get the value of w, which is

$$w_0 = \frac{x_0}{t_0} \tag{2}$$

However, with the second observation we get

$$w_0 = \frac{x_1}{t_1} \tag{3}$$

It is clear that, comparing the equation 2 and 3, the two values of w, w_0 and w_1, can not be the same if the observation are different, so we understand that the approximation can't ever be *perfect*, and it introduces a small error to the target result.

Linear regression aims to compute the best value of w that better approximates every observation. w must then minimize the mean error.

Square error (SE) is then the best solution for finding the best value of w, because it grows more than linearly, giving heavier weight to larger errors. It is also even, meaning that $(y-t)^2 = (t-y)^2$, and it is differentiable with respect to the model output:

$$\frac{\mathrm{d}}{\mathrm{d}x}\lambda_{SE}(y,t) = 2(y-t) \tag{4}$$

Since we need to take the *mean* SE, we can use the Mean Square error (MSE), that can be computed as

$$J_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2 \tag{5}$$

After some computation, it is proven that the best value of w that approximates the best the value of the set of observation is

$$w = \frac{\sum_{l=1}^{N} x_l t_l}{\sum_{l=1}^{N} x_l^2} \tag{6}$$

With the equation 6 is then possible to get the best value of w for the equation 1.

B. One-dimensional problem with intercept

It is possible to build a better regression model using an interception factor, w_0, that can move to the left or to the right the approximation line, The equation 1 becomes then

$$t = w_0 x + w_1 \tag{7}$$

the two parameters, w_0 and w_1 can then be computed as follows

$$w_1 = \frac{\sum_{l=1}^{N}(x_l - \overline{x})(t_l - \overline{t})}{\sum_{l=1}^{N}(x_l - \overline{x})^2} \tag{8}$$

$$w_0 = \overline{t} - w_1\overline{x} \tag{9}$$

C. Multi-dimensional problem

The equations above can be generalized for the multi-dimensional case. Let $\mathbf{X}$ the set of observation, each of them with multiple variables. Let $\mathbf{w}$ the vector of all the approximation parameters, one for each variable, including w_o intercept parameter

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_n \end{pmatrix} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,n} \end{pmatrix} \tag{10}$$

$$\mathbf{w} = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} \tag{11}$$

The final output oh the linear regression will be

$$y_i = w_0 + x_{i,1}w_1 + x_{i,2}w_2 + ... + x_{i,n}w_n \tag{12}$$

We then obtain

$$\mathbf{y} = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,n} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \mathbf{Xw} \tag{13}$$

Note that a constant column of ones is added in the first matrix, in order to incorporate w_0.

As for the one-dimensional case, also here it is necessary to minimize the MSE. Since we are dealing with matrices and not one-dimensional variables, now we must set $\nabla J_{MSE} = 0$. We can then find $\mathbf{w}$ as follows

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} = \mathbf{X}^\dagger\mathbf{t} \tag{14}$$

Where $\mathbf{X}^\dagger$ is the Moore-Penrose pseudoinverse of $\mathbf{X}$. Notice that computing the pseudoinverse can result in introducing a small but notable error in the calculation. This small error however can be quickly amplified by the condition number and cause large errors on the result.

III. The Assignment

The given assignment consist of three tasks:
- **Task 1:** Get data
- **Task 2:** Fit a linear regression model
- **Task 3:** Test regression model

A. Task 1 : Get data

This assignment takes into account two different sets

- turkish-se-SP500vsMSCI (Turkish stock exchange data)
- mtcarsdata-4features (1974 Motor Trend Car Road Tests)

The first set is composed by 536 observations and 1 variable. The second set is composed by 32 observations and 3 variables.

The Two sets are initially stored as .csv files. To import them in Matlab it is required to use the function *csvread()*.

B. Task 2: Build a naive Bayes classifier

In the second tasks, we have to compute the linear regression parameters in four different cases:

- One-dimensional problem without intercept on the Turkish stock exchange data
- Compare graphically the solution obtained on different random subsets (10%) of the whole data set
- One-dimensional problem with intercept on the Motor Trends car data, using columns mpg and weight
- Multi-dimensional problem on the complete MTcars data, using all four columns (predict mpg with the other three columns)

All the sub-tasks above are executed inside a main function, which is divided in sections. In every section the function computes the requires parameters (w for sub-task 1 and 2, w_0 and w_1 for sub-task 3 and $\mathbf{w}$ for the sub-task 4) and shows the results using graph, that are discussed in section IV. Note that it's impossible to show a graph for sub-task 4, so instead the value of $\mathbf{w}$ is written in the Matlab console.

Inside the training section the classifiers acquires knowledge computing every conditioned probability using the equation 4.Data are stored in a *cell matrix* in order to avoid wasting space. The cell matrix is defined and instantiated at every execution of the classifier.

C. Task 3: Test regression model

In task 3 it is required to re-run all the sub-tasks above using only 5% of the entire data set, and compute the *objective* (Mean Square Error) with the 5% subset and with the remaining 95% subset. Then it asks to repeat this procedure with 10 different random 5% subsets.

To do that, a for loop is added around every code section, and all the *objectives* are stored into variables for plotting them.

IV. Results

A. Task 2

Regarding Task 2, the function shows three graphs for the first three sub-tasks and compute and prints $\mathbf{w}$ for the fourth one.

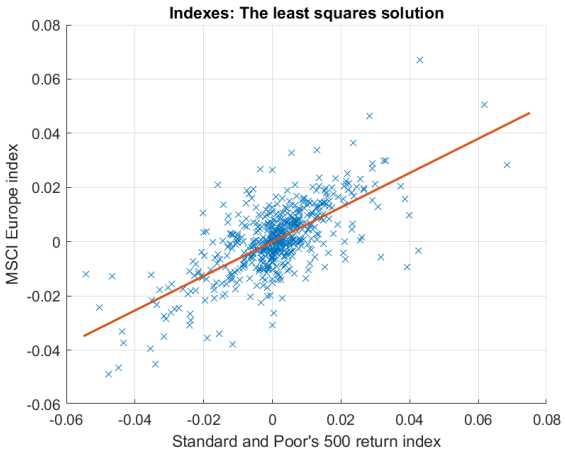

Fig. 1: Sub-Task 1 graph

As shown in the image 1, the value of w that approximates the best every observation is

$$w = 0.6639$$

The figure 2 shows instead a comparison between two values of w, one computed using the full Tuskish Stock Exchange data set (as in 2) and one computed using only 10% of the set.

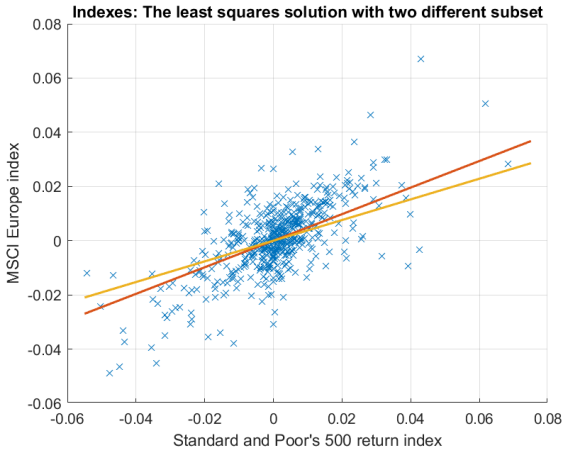

Fig. 2: Sub-Task 2 graph

The two values are

$$w_a = 0.6639 \ (red)$$

$$w_b = 0.5042 \ (yellow)$$

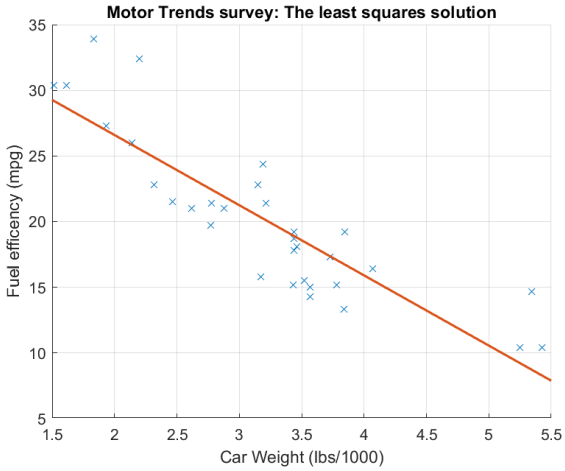

Fig. 3: Sub-Task 3 plot

In the figure 3 we can see the linear regression with the intercept factor. This is still a one-dimensional problem: only the car weight is used as variable. The two regression factor, w_0 and w_1 are

$$w_1 = -5.3445$$

$$w_0 = 37.2851$$

For the sub-task 4, the function computes the four values that composes **w**. Here all the three variables of the set are considered, making this a multi-dimensional problem. In particular, the linear regression should predict mpg using the other variables. Here's what the function has computed:

$$\mathbf{w} = \begin{pmatrix} 37.1055 \\ -0.0009 \\ -0.0312 \\ -3.8009 \end{pmatrix}$$

B. *Task 3*

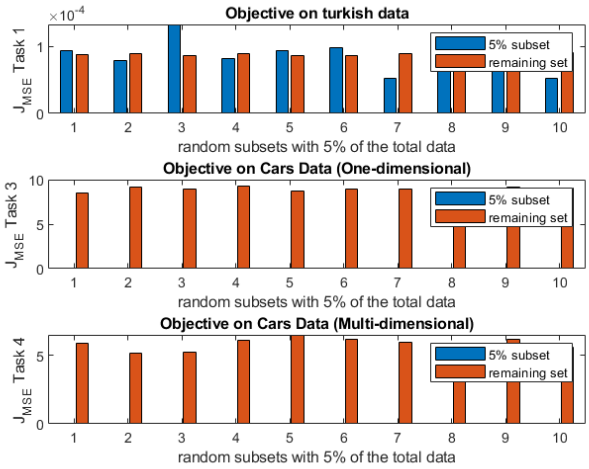

Fig. 4: Task 3 plot

If the subset should correspond to the 5% of the total set, it means that the *Turkish Stock* subset should be composed by 27 observation, while the cars data set should be composed by only 2 observations. This has some consequences on the test outcome: If a set is made of only two observation, the computed factor w will then be *perfect*. As a proof, consider that w is the slope of the correspondent regression line. Given only two points, there's one and only one line that passes through them, and that line matches the regression line. In this case

$$J_{MSE} = 0$$

The figures 4 shows a comparison of the computed *objectives* obtained with 10 random 5% subsets, and the *objectives* of their consequent remaining subsets. For what has just been told, every J_MSE related to 1974 Motor Trend subset is not shown in the figure because it is null.