Linear Regression

Machine Learning 2021/22: 2^{nd}Assignment
Matteo Maragliano
EMARO-European Master on Advanced Robotics
DIBRIS - Dip. Informatica, Bioingegneria, Robotica, Ing. dei Sistemi
Università degli Studi di Genova

***Abstract*—The goal of the assignment is to implement a *linear regression* among a certain amount of data given. *Regression* means approximating a functional dependency based on measured data.**
The first task to be implemented concerned getting the data and make them ready to be read and used by the program, in our case MATLAB. We were given two different data sets to be used in different ways: the first one has two columns, the first represents the input and the second the output; the second data set has four different columns representing *mpg (miles per gallon)*, *disp (displacement)*, *hp (horse power)* and *weight* respectively.
Then we had to complete the second task:

- **first of all we had to implement a one-dimensional problem without intercept on the first data set;**
- **than we had to compare it graphically with a random subset of dimension 10% of the total one and find a regression without intercept as well;**
- **the third point took the second data set and used its first and fourth column, mpg and weight respectively; we had to compute another one-dimensional problem using weight as the output and the mpg data as the input;**
- **the last one point dealt with a multi-dimensional problem using the last three columns of the second data set to predict the mpg, the first one column.**

The third task concerned the re-make of the point 1,3 and 4 of the task 2 using this time 5% of the total data given. Then we had to compute the objective, mean square error, on the training data and also on the remaining 95% of the data.
In conclusion the program had to be tested multiple times, for example 10 times, and then all the results to be shown on a graph or in a table to be discussed properly. .

I. Introduction

In statistical modeling, regression analysis is a set of statistical processes for estimating the relationships between a dependent variable, often called the *outcome* or *response* variable and one or more independent variables. The most common form of regression analysis is *linear regression*, in which one finds the line, or a more complex linear combination, that most closely fits the data according to a specific mathematical criterion.
For specific mathematical reasons, this allows the researcher to estimate the conditional expectation of the dependent variable when the independent variables take on a given set of values.

II. Getting the data

The first task to be completed is the one in which we had to get the data from given data sets and make them readable for out software, MATLAB.
We were given two different data sets, the first one containing only two columns, the first for the input and the second one for the output. In MATLAB we could read them by a specific function, *readmatrix*, which transforms the data set into a matrix ready to be used.

With the second data set it was a little be more complicated since we had also a literal columns, the first one, and other three number columns. With the same function used before we transformed the data set into a matrix and then we proceeded to remove the first column since it was not necessary for us and not readable by our software. So at this time we had all data available and ready to be processed.

III. Linear Regression Model

This task concerned the presence of four different points to complete. In the first one we had to model a linear regression of a one dimension problem, using the first data set given. In this case we did not have to consider the intercept of the linear regression, because of the zero mean value of the data set.
We proceeded by calculating the angular coefficient, stored in a vector with proper dimensions, by doing the division of all the outcome values (first column) and all the independent values (the second column). After that, we could compute the outcome of the regression by simply multiplying the angular coefficient obtained with the x-values of the data set. The formula used is:

$$y = wx$$

All results obtained were plotted in a graph.
The second point was just a comparison between the previous result obtained and an analogous problem computed with a sub set of 10% the dimension of the initial set. All the steps computed were the same and also in this case the result was plotted on a graph.

In the third point of the task the problem was still a one dimensional problem but in this case there was also the intercept to be computed. We proceeded in the same way only slightly changing the computation of *X*, all the values of the independent variable. In this case the formula for the computation of the outcome is:

$$y = w_1 x + w_0$$

where w_0 is the intercept of the line.
Then the angular coefficient and the computation of the outcome were the same as before and the result plotted in a graph gave the result attended.

The most difficult point was the last one because of the presence of a multi dimensional problem; in this case the *target* was the first column whereas the input variables were the other three columns of the data set. Also in this case there was the intercept to be computed.

The formula used to computed the angular coefficient is a little different by the one used in the software, because of more readable code. Anyway nothing changes in terms of result; the theoretical one used is the following:

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

where **t** represents the target vector. One computed the *w* vector the outcome is as always the multiplication of the input and the angular coefficient.

The graph in this case represents an approximation of the diagonal of the square having side from the minimum to the maximum value: the dot-graph is an approximation whereas the diagonal is the exact diagonal took as reference.

We give a general view of the result with all the following graphs:

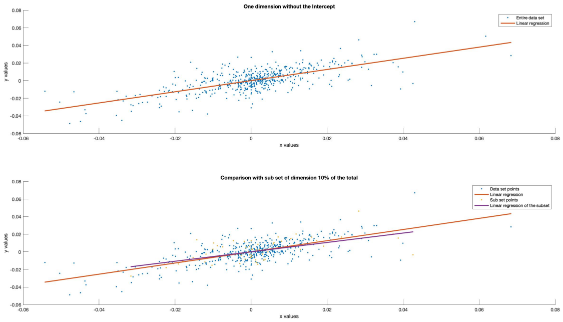

Fig. 1. Plots concerning the first and the second point of the task: the first is the entire data set; the second represents a comparison with a subset of 10% the dimension of the original one.

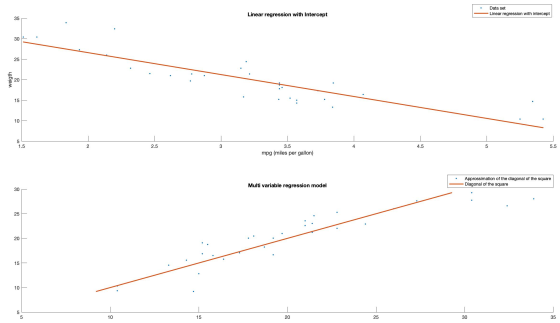

Fig. 2. Plots concerning the third and fourth point of the task: in the first graph there is the regression from *mpg* to *weight*; the second graph, as it can be seen, represents the approximation of the diagonal and the diagonal itself.

IV. Test Regression Model

In the last one task, we had to test the regression model obtained by computed errors on two sub sections of the data set. We started by dividing the set into two different parts: according to a given percentage of 5-10% we had the training test and the remaining 95-90% the test set. Since the division is randomly computed, we had to do this for a certain number of iterations in order to verify the behaviour: for example 10 different iterations.

In order to do the task we computed the angular coefficient with the test test and then, with the value obtained, we computed the outcome by using all the target values of the training set. Then the error was computed by using the Mean Squared Error among the outcome computed and the target of the set; of course both done for the test and the training set. At the end all the results were plotted by a histogram graph to show the error.

We give the plot obtained as follows:

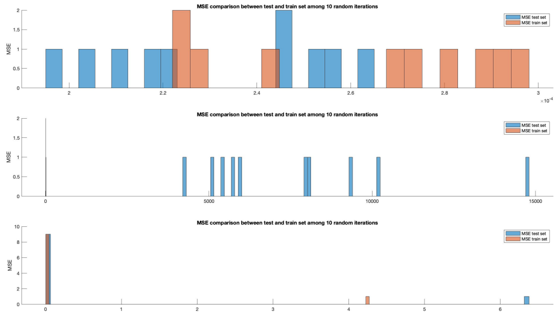

Fig. 3. Histograms for linear regression obtained

In order to make more specific the plot we indicated the number of bins to be shown, in particular: in the first one we had the division into 20 different bins while in the second and third histograms we had 100 bins. Of course, as it can be seen in the picture, the results of the training set are much lower than those of the test set because of the small percentage given to the division.