Robust Linear Regression

Ignasi Cos – ignasi.cos@ub.edu

October 2023

Abstract

This practical focuses on data regression methods. We will focus on some of the basic methods and on the relevance of the error assessment metrics to ensure robustness against outliers. Note that this practical is concerned with unconstrained optimization only.

1 Introduction

Linear regression is an approach to model the relationship between two continuous, quantitative, variables. One variable, denoted x, is regarded as the predictor or independent variable. The other variable, denoted as y, is regarded as the response or dependent variable.

Figure 1 shows a set of data points in blue. In this example the goal is to fit these data points using a linear regression, i.e.we only have one predictor variable.

What is the best fitting line? Let $\mathbf{x}_i = \{x_i, y_i\}$ with $i = 1 \dots m$ be the data points. There are different approaches that one may take to compute the best fitting line, see figure 2. The figure shows two different methods: the standard fitting using vertical offsets, and the fitting using perpendicular offsets.

We focus here on the standard approach, easy to formulate and implement. The fitting process may threefore be reviewed as a minimization process in which $\hat{y}_i$ is the fitted value for element i. The function to minimize is the equation $\hat{y}_i = w_0 x_i + w_1$, where $w_0 \in R$ and $w_1 \in R$ are the parameters to be estimated. The parameters w_0 and w_1 may be computed in such a way that the prediction error (or residual error), $e_i = |\hat{y}_i - y_i|$, is minimised. The line that fits the data "best" will be the one in which the m prediction errors are the smallest possible. Observe that in this case we perform a fitting using vertical offsets, see figure 2. There are other kind of approaches that use the perpendicular offset, which escape the reach of this practical.

A classical way to compute the optimal parameters w_0 and w_1 is to use the least squares method (LSM), with its associated least squares error function:

$$Q = \frac{1}{2}\sum_{i=1}^{m} e_i^2 = \frac{1}{2}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2 \tag{1}$$

You way use the gradient (or Newton) descent method to obtain the optimal values for w_0 and w_1;

$$Q = \frac{1}{2}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 = \frac{1}{2}\sum_{i=1}^{m}(w_0 x_i + w_1 - y_i)^2$$

we wish to obtain the values of w_0 and w_1 that minimize Q. For this purpose let us compute the corresponding gradient

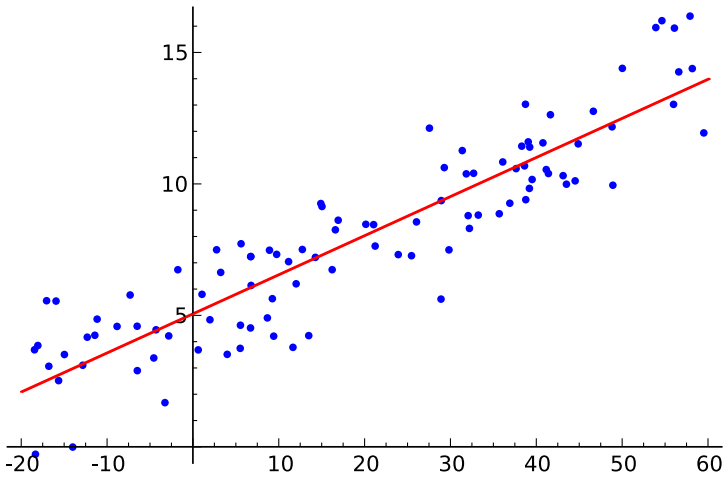

Figure 1: Linear regression example. The data points are shown in blue, and the fitted line in red. Source: https://en.wikipedia.org/wiki/Linear_regression.

$$\begin{aligned}\frac{\partial Q}{\partial w_0} &= \sum_{i=1}^{m} (w_0 x_i + w_1 - y_i)\, x_i \\ \frac{\partial Q}{\partial w_1} &= \sum_{i=1}^{m} (w_0 x_i + w_1 - y_i)\end{aligned}$$

The parameters to be estimated are $\mathbf{w} = (w_0, w_1)$. The gradient descent is

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \alpha^k \nabla Q(\mathbf{w}) \qquad \nabla Q(\mathbf{w}) = \left(\frac{\partial Q}{\partial w_0}, \frac{\partial Q}{\partial w_1} \right)$$

Your Task

Follow the next steps:

1. Implement the method proposed with gradient descent and backtracking (or a small constant α value). You may check your method with a randomly generated set of points

```
m = [0.,0.]
angle = 45*math.pi/180
rot = np.array([[math.cos(angle), -math.sin(angle)], [math.sin(angle),
math.cos(angle)]])
lamb = np.array([[100,0],[0,1]])
s = np.matmul(rot, np.matmul(lamb, rot.transpose()))
points = np.random.multivariate_normal(m,s,100)
```

   The `angle` value, as well as the `lamb` matrix, allow us to control the shape of the random values generated.

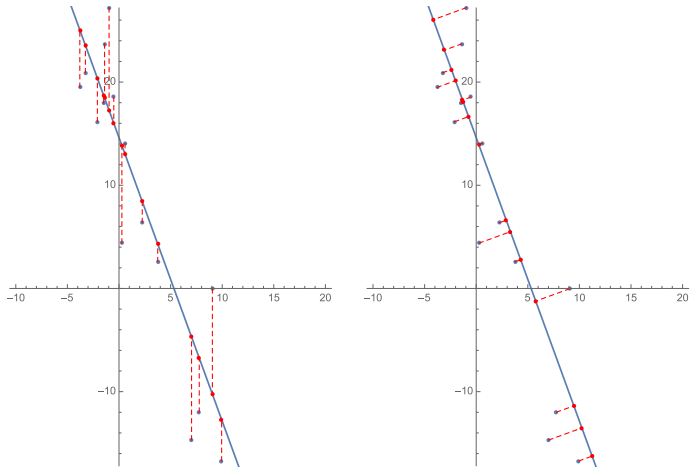

Figure 2: Left: fitting using vertical offsets, Right: fitting using perpendicular offsets. Obtained from http://toddreed.name/articles/line-fitting/.

Compute the parameters associated to the model and draw the lines that is obtained with the set of points.

2. Let us now check the sensitivity of the method to outliers, that is, points that do not follow the model. Using a set of points generate with an angle of 45 degrees, change the value of one point to a value "far away" from the set of points, for instance `points[1] = [-40,20]`. Draw the line that approximates the set of points and observe that one point may have a large influence in the obtained solution.

   Change now other points of the original dataset to assess the influence of outliers.

Outliers are common when performing most data analyses. Thus, it is important to use robust functions for outliers in order to obtain the expected result.

2 Robust functions

The LSM is known to be sensitive to outliers, i.e., samples that do not particularly contribute to fit the model (they typically have a different origin than the phenomenon we intend to study, for instance noise). The reason is that the error the LSM has to minimise in the case of the outlier is very large, which contributes more to the fit than the resulting fitting coefficients. The outliers may thus have a large influence in the numerical values of the parameters to be estimated.

One solution is to alter equation (1) to make it more robust (less sensitive) to outliers. Without entering into exhaustive details, a reasonable possibility is to minimise

$$\sum_{i=1}^{m} \rho(e_i)$$

where $\rho(u)$ is a robust error function. For the least squared method $\rho(u) = \frac{1}{2}u^2$, but one may take

other functions such as the Cauchy function, which is defined to be[1]

$$\rho(u) = \frac{c^2}{2} \log \left[1 + \left(\frac{u}{c} \right)^2 \right] \tag{2}$$

where $c \in R$.

Your Task

For simplicity, assume $c = 1$ for the Cauchy function (you may use the set of points with a 45 degrees)

1. Plot the least squares function, $\rho(u) = \frac{1}{2}u^2$, and compare it with the Cauchy function, Eq. (2), in order to see the "importance" given to each prediction error u, you may, for instance, plot the function $\rho(u)$ for $|u| \leq 10$.

2. Implement the algorithm that allows to compute the parameters w_0 and w_1 using the Cauchy function. For that issue you can use the backtracking gradient descent method (there is no need to use the Newton method).

3. Compare the results obtained with the least squares function and with the Cauchy function, assuming that there are no outliers in the dataset.

4. Compare now the results with only one outlier. You may proceed as previously proposed. The Cauchy function should be more robust than the quadratic function.

5. Test the influence of the parameter c in the parameters obtained. You may, for instance, check the results obtained with $c = 1$, $c = 100$, $c = 1/100$ and $c = 1/1000$. Can you reason why of these results? To this end, you are recommended to plot the histogram of the error function $|u|$ (its absolute value of u) and to compare with the shape corresponding $\rho(u)$ functions. Consider performing a "zoom" of the Cauchy function to see the interval at which the function behaves as a quadratic function. Which values are considered as "inliers" / "outliers"?

6. The Cauchy function is not "perfect", and it is not robust for any number of outliers. Using $c = 1$ and $c = 1/100$ you may, as before, gradually introduce more number of outliers into the dataset. You should see that for a certain number of outliers, the the Cauchy function will be sensitive to the "high" number of outliers. Can you comment on the experiments you have performed?

Report

You are requested to deliver a report (PDF, jupiter notebook preferably), to be performed in pairs. Please, do state clearly your names and niubs on the document you submit! Comment each of the steps taking your to your results and plots you obtain. Do not expect the examiner to interpret the results for you. I would like to see if you are able to understand the results you have obtained.

[1] If you are interested on this issue you may check Steward, C. "Robust parameter estimation in computer vision", SIAM Review, Vol. 41, No. 3, pp. 513–537, 1999.