Computational Methods for Detection,
Estimation and Identification 2022/2023

Assignment 3

Manuel Alberto Dionísio dos Santos
2019231352

1 Overview

This assignment is divided in 2 parts about robust estimation methods. The first part revolves around the experiment on performance of a robotic platform breaking system breaking system. In the second part, we are provided with a function named SimulationRobotWall() that when receives a certain noise standard deviation, percentage of outliers, duration of the experiment and sampling time as inputs, outputs a vector of measurements, a vector of timestamps of each measurement and the ground truth solution for a model that describes said measurements.

2 First Part

2.1 Mathematical Model

As of the previous assignments, it's possible to derive the structure of matrices G and m, knowing that the system has with n = 3 parameters and m = 100 observations.

$$G = \begin{bmatrix} 1 & t_1 & \frac{t_1^2}{2} \\ 1 & t_2 & \frac{t_2^2}{2} \\ & \vdots & \\ 1 & t_{100} & \frac{t_{100}^2}{2} \end{bmatrix}_{100\times 3} \tag{1}$$

$$m = \begin{bmatrix} y_0 \\ v_0 \\ a_0 \end{bmatrix}_{3\times 1} \tag{2}$$

$$d = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_{100} \end{bmatrix}_{100\times 1} \tag{3}$$

In accordance with the description of the problem, it is also possible to obtain the t matrix, containing all the time stamps of the collection of the readings.

$$t = \begin{bmatrix} 0.1 & 0.2 & 0.3 & \dots & 9.9 & 10 \end{bmatrix}_{1\times 100} \tag{4}$$

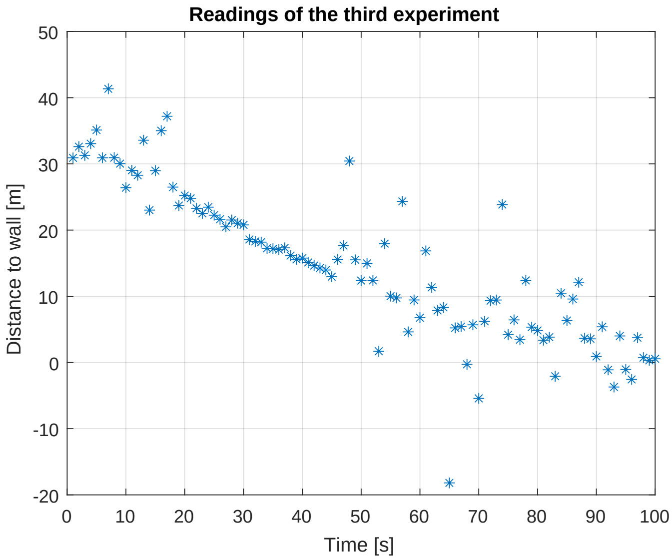

Figure 1: Visualization of measurements in d.

2.2 Visualization of Data

In order to ease the comprehension of the obtained data, and once the readings matrix was obtained, the first step was to visualize the data being worked, as shown on figure (1).

It is possible to notice that distance to the wall decreases as time passes in what appears to be a quadratic behaviour, in agreement with the obtained mathematical model. It is also important to notice that the data presents what at first glance appear to be outliers.

2.3 Least Squares solution

Using the Least Squares solution, a first estimation for the model parameters is obtained.

$$m_{L2} = \begin{bmatrix} y_0 = 35.761 \\ v_0 = -5.690 \\ a_0 = 0.448 \end{bmatrix}_{3\times 1} \tag{5}$$

Figure [2] shows a visualization of the Least Squares Solution.

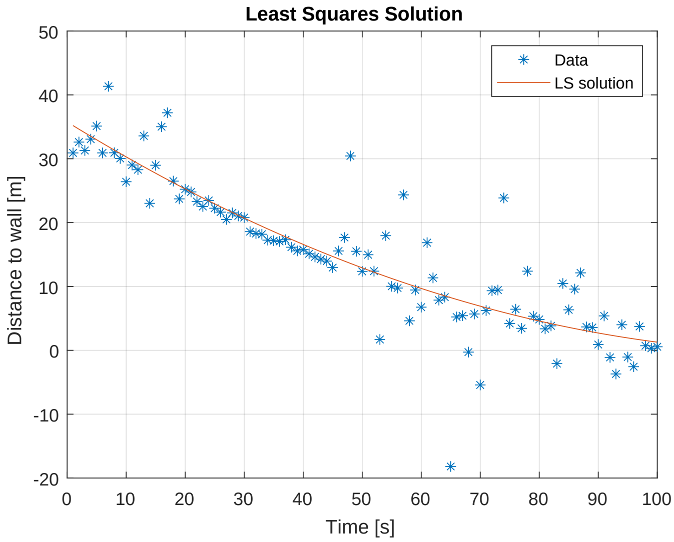

Figure 2: Visualization of Least Squares solution.

2.4 Covariance Matrix Estimation

In the second part, the objective is to obtain an estimation for the covariance matrix. Since the noise statistical parameters are unknown, there is no way to analytically estimate it.

In order to compare the results, a ground Truth of the covariance matrix was given []

$$CovM_{gt} = \begin{bmatrix} 1.851 & -0.797 & 0.134 \\ -0.797 & 0.419 & -0.076 \\ 0.134 & -0.076 & 0.015 \end{bmatrix}_{3\times3} \tag{6}$$

To show that the methods employed in previous assignments would not solve this problem, a first try was made, using expressions [7] and [8].

$$s = \frac{||r||_2}{\sqrt{\nu}} \tag{7}$$

$$\tilde{C} = s^2(G_w^T G_w)^{-1} \tag{8}$$

This expressions lead to the following result:

$$\tilde{C} = \begin{bmatrix} 2.690 & -1.070 & 0.177 \\ -1.070 & 0.562 & -0.104 \\ 0.178 & -0.104 & 0.021 \end{bmatrix}_{3\times3} \tag{9}$$

Analysing this result, is possible to infer that it is not a good estimation.

As so, an alternative method was employed to reach this objective. Due to the characteristics of the problem, and since there's no statistical parameters for the noise, an algorithm of the Bootstrap family was used. There was an idea to use an algorithm of the Monte Carlo family, but since, as previously said, there's no statistical parameters for the noise, there's no way to correctly generate synthetic values for a new samples.

The employed Bootstrap algorithm consisted of 10000 repetitions of solving a weighted system with the Least Squares solution.

First, a weight matrix constructed from the values calculated with [7] was calculated. Next, for each repetition of the algorithm, a sub-sample d_i with possible repetitions of the initial d_0 vector were calculated. Using the weighted Least Squares solution, a m_i associated with each d_0 was calculated.

The m_i estimations were then used to calculate a matrix A, according to equation [10], in which q is the number of repetitions.

$$A_{i,q} = m_{L1,i}^T - \tilde{m}_{L1,i}^T \tag{10}$$

Finally, using matrix A in conjunction with equation [11], a new estimation for the Covariance Matrix was obtained [12].

$$Cov(m_{L1}) = \frac{A^T A}{q} \tag{11}$$

$$\tilde{C_{boot}} = \begin{bmatrix} 1.888 & -0.896 & 0.149 \\ -0.896 & 0.525 & -0.093 \\ 0.149 & -0.093 & 0.017 \end{bmatrix}_{3\times3} \tag{12}$$

As is possible to conclude by analysing it, $\tilde{C_{boot}}$ is closer to $CovM_{gt}$ then $\tilde{C}$ is.

3 Second Part

3.1 Planning

In the second part of the assignment the objective is to evaluate the performance of three different estimation approaches

- normal least squares
- robust M-estimation using the L1 norm
- RANSAC based approach with normal least squares

when faced with data presenting different levels of noise standard deviations (from 0 to 5, unitary increments) and different proportions of outliers (10%, 30% and 50%), while using the Monte Carlo Simulation.

The setup will consist of two nested for loops (one for the outliers and the other for the noise), in which at each iteration a new call to the SimulationRobotWall() function is made, which generates a new "experiment" so to speak (a new d_0 and ground truth solution). Each of this iterations displays a specific combination of noise standard deviation and proportion of outliers (for example, 1 and 30% respectively). To apply each method, the G matrix established on the first part of the assignment will be used.

3.2 Normal Least Squares

The first method to apply is the Normal Least Squares. This is the method applied since the first assignment. It basically consists of applying the Pseudo-Inverse of Moore-Penrose, using the G previously calculated and the d_0 obtained from the call to the SimulationRobotWall() function made.

3.2.1 Results

As expected, the Normal Least Squares solution proved to have difficulties getting a correct estimation as the percentage of outliers arrises.

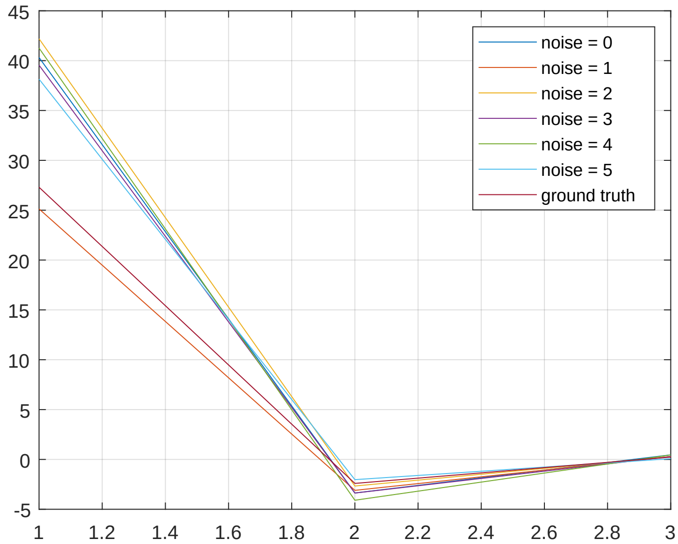

Figure 3: m obtained for outliers at 10%

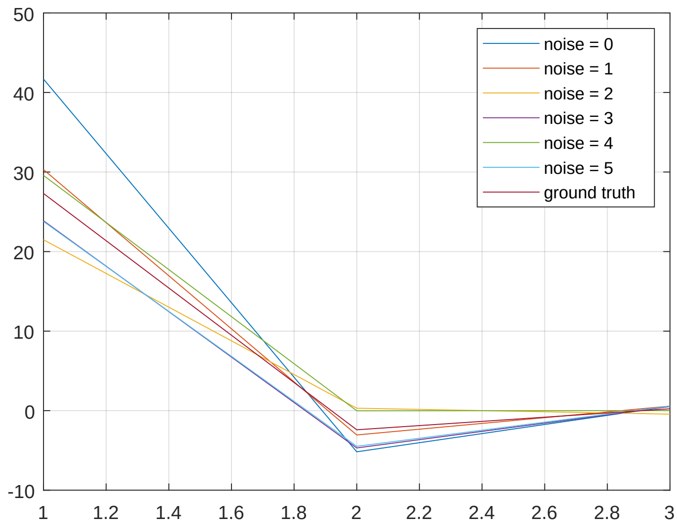

Figure 4: m obtained for outliers at 30%

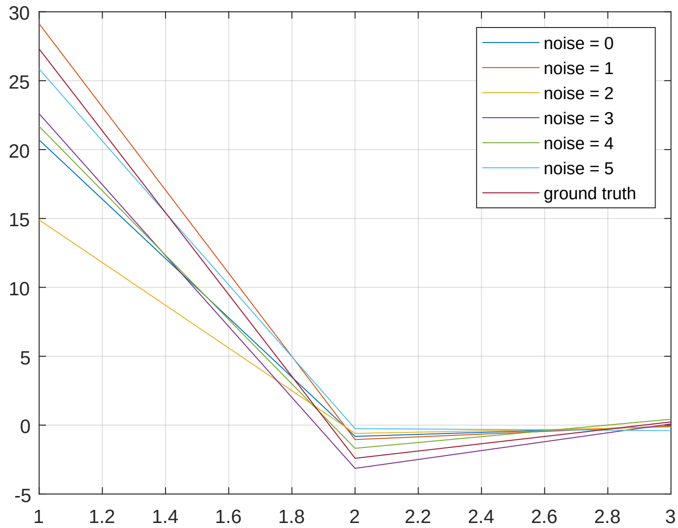

Figure 5: m obtained for outliers at 50%

AS predicted, the estimation quality has decreased with an increased percentage of outliers

3.3 Robust M-estimation using the L1 Norm

The second method is the Robust M-estimation using the L1 Norm, a robust method against outliers.

Unfortunately the implementation of the Robust M-estimation using the L1 Norm method was not successful, but the prevision is that it lead to a model with a good resistance to outliers.

3.4 RANSAC based approach with Normal Least Squares

The third and final method is RANSAC. The basic idea of RANSAC is to randomly select a subset of the data points d_0 anduse them to estimate a model. This model is then used to evaluate the remaining data points, and those that are consistent with the model are considered as "inliers". The process is repeated multiple times, and the best model that fits the most inliers is selected.

The RANSAC algorithm can be summarized in the following steps:

1. Select a random subset of the data points.
2. Use the subset to fit a model.
3. Evaluate the remaining data points and classify them as inliers or outliers based on a threshold.
4. If the number of inliers is greater than a pre-defined threshold, re-estimate the model using all inliers.
5. Repeat steps 1-4 for a fixed number of iterations or until a satisfactory model is found.

RANSAC is particularly useful when dealing with datasets that contain a significant amount of outliers, as it can effectively identify and ignore them when estimating the model parameters.

Unfortunately the implementation of the RANSAC method was not successful, but the prevision is that it lead to a model with a good resistance to outliers.

References

[1] R. C. Aster, B. Borchers, and C. H. Thurber, Parameter Estimation and Inverse Problems. Elsevier, 2018.

[2] Nuno Gonçalves, sebenta_TCDEI_chapter1_linear_inverse_problems, University of Coimbra, 2023