Hand in assignment 2

Johannes Lundahl Daniel Söderqvist

September 27, 2023

First part - Estimators and Least squares.

1 Unbiased estimate.

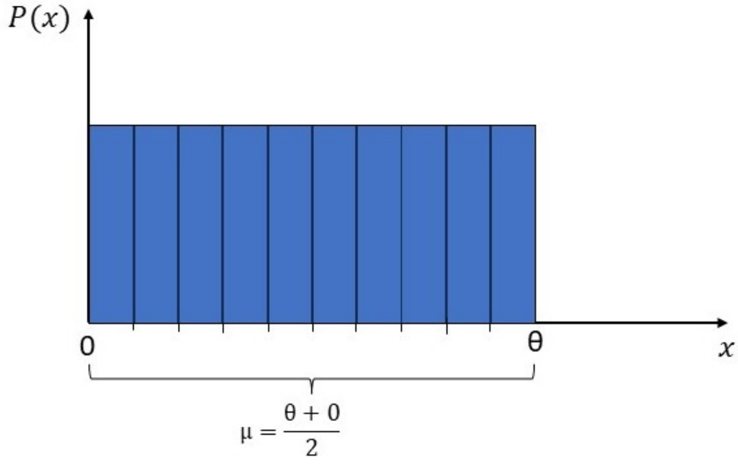

Figure 1: Uniform distribution $\mathcal{U}[0, \theta]$

a) In this task we are going to find an estimator $\hat{\theta}$ for the population given data samples that are uniformly distributed as $\theta \in [0, \infty]$. The unbiased estimator $\hat{\theta}$ of the population mean, μ, can be said to be the mean of the IID samples, $\mathbb{E}[X]$ as follows:

$$\mathbb{E}[X] = \mu \rightarrow \frac{1}{N} \sum_{i=1}^{N-1} x_i = \frac{\theta}{2} \tag{1}$$

$$let \;\; \theta = \hat{\theta} \;\; then \;\; \hat{\theta} = \frac{2}{N} \sum_{i=1}^{N-1} x_i \tag{2}$$

For the estimator, $\hat{\theta}$, to be considered unbiased it must full fill:

$$\mathbb{E}[\hat{\theta}] = \theta \tag{3}$$

$$\mathbb{E}[\hat{\theta}] = \mathbb{E}[\frac{2}{N} \sum_{i=1}^{N-1} x_i] \tag{4}$$

Since the Expected value, $\mathbb{E}$, of a constant can be moved freely inside or outside of the summation and also since sum of $\mathbb{E}[x_i]$ was shown to be the same as the mean value μ we get:

$$\mathbb{E}[\hat{\theta}] = \frac{2}{N}\sum_{i=1}^{N-1}\mathbb{E}[x_i] = \frac{2}{N}N\frac{\theta}{2} = \theta \tag{5}$$

Thus we see that the estimator of theta is unbiased.

2 Linear least squares.

a) In this task we are supposed to compute the least square estimate of θ considering a given linear system. The least square estimate for a linear regression can be derived from the least square criterion which can be found in equation (5.7) in the lecture notes book. To achieve the least square estimate we minimize and rewrite the least square criterion and the approach can be viewed on page 109 in the lecture notes after equation (5.7). The resulting equation for the least square estimate can be found in equation (5.8) in lecture notes.

   The next step would be to decide the regression vector which we simply can see from the task description to be: $\varphi = u(k)$.

   Using the least square estimate equation (5.8) from the lecture notes previously described with the given regression vector φ we simply end up with the result:

$$\hat{\theta}_N = \frac{\frac{1}{N}\sum_{k=0}^{N-1}u(k)y(k)}{\frac{1}{N}\sum_{k=0}^{N-1}u(k)^2} \tag{6}$$

b) The LS estimate can be written as equation (5.15) as explained previous in a). The LS estimate $\hat{\theta}_N$ can be said to be biased if:

$$\mathbb{E}[\hat{\theta}_N] = \theta_0, \;\; where\ \theta_0\ is\ a\ true\ parameter \tag{7}$$

   To show that the estimation is unbiased or not, we can used the derived formula (5.23) from the lecture notes were we obtain a sum of the true parameter and two terms containing $\varphi(i)$ and $e(i)$. $\varphi(i)$ in this case contains our $u[k]$ and is not equal to zero. Normally, the estimation of LS is unbiased since the error term is normally distributed around 0 and thus the Expected value of the error $\mathbb{E}[e(i)]$ becomes zero. In this case, we have an error normally distributed around mean 1, $e[k] \sim \mathcal{N}[1, \sigma^2]$, which means that the second term in equation (5.23) is not equal to zero and thus equation 7 from above does not hold.

$$\mathbb{E}[\hat{\theta}_N] = \theta_0 + (\frac{1}{N}\sum_{i=1}^{N}\varphi(i)\varphi(i)^\top)^{-1}\cdot\underbrace{\mathbb{E}[\frac{1}{N}\sum_{i=1}^{N}\varphi(i)e(i)]}_{\neq 0} \neq \theta_0 \tag{8}$$

c) We can see that if $u(k) = 0$ then the estimation of θ also becomes 0. $E[\hat{\theta}] = 0$

3 Computer exercise. Curve fitting.

a) (The code for computing and ploting all the figures and parameters are attached.)

In this task we are supposed to plot a noise corrupted function with our calculated version that are fitted to the noise corrupted version. The plot down below (2) shows the result where the corrupted version is a uniformly distribution illustrated with the blue dots and the calculated version is shown as the red line.

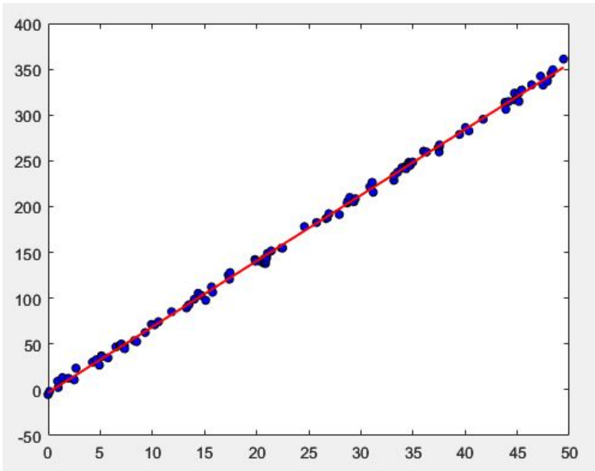

Figure 2: Figure showing a set of uniformly distributed dots along a linear line with the corresponding least square fit function.

The parameters for our calculated fitted version that corresponds to the red line in teh figure are:

$$\begin{aligned} a_0 &= -4.196 \\ b_0 &= 7.231 \end{aligned} \tag{9}$$

Which in turn summarizes to the equation:

$$y(x) = 7.231x - 4.196 \tag{10}$$

b) In this task we are supposed to as in the previous data use a noise corrupted function but in this case it's a second order polynomial function which makes it curved. We are supposed to both try out to fit the data with a 2nd order polynomial fit and a linear fit. The plots for both trials are shown down below in figures (3, 4) The corresponding parameters and equations to the poly fit and linear fit are shown down below:

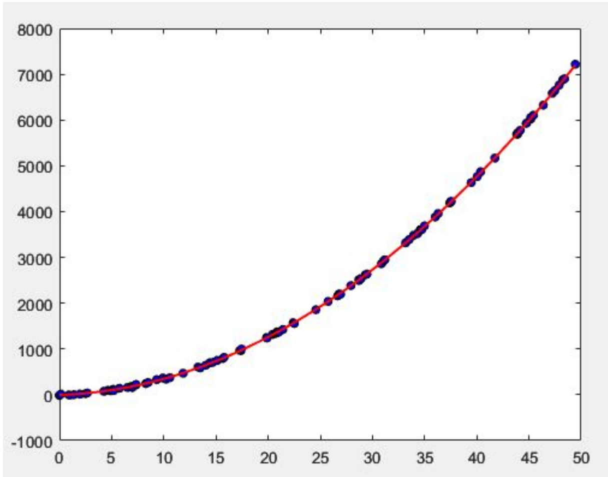

Figure 3: Figure showing a set of uniformly distributed dots along a curved line with the corresponding 2:nd order polynomial fit function.

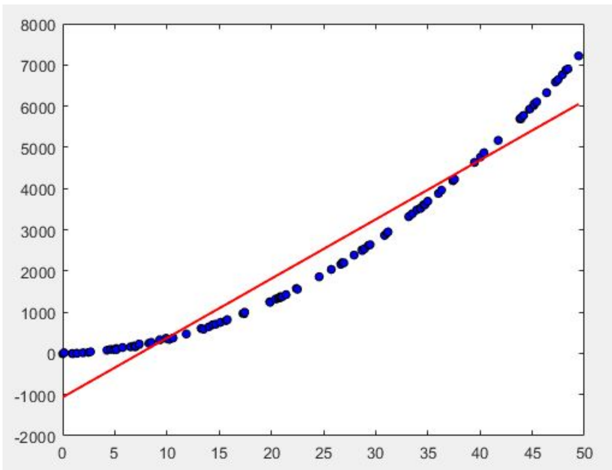

Figure 4: Figure showing a set of uniformly distributed dots along a curved line with the corresponding least square fit function.

First case, 2nd order poly fit:

$$\begin{aligned} a_0 &= -6.325 \\ b_0 &= 7.391 \\ c_0 &= 2.795 \end{aligned} \tag{11}$$

$$y(x) = 2.795x^2 + 7.391x - 6.325 \tag{12}$$

Second case, linear fit:

$$\begin{aligned} a_0 &= -1214.631 \\ b_0 &= 147.182 \end{aligned} \tag{13}$$

$$y(x) = 147.182x - 1214.631 \tag{14}$$

In this task we were also supposed to calculate the corresponding residuals which we also were given a formula to do. By using the formula in MATLAB the following residuals could be calculated:

First case, 2nd order poly fit:

$$\epsilon = 152 \tag{15}$$

Second case, linear fit:

$$\epsilon = 289739 \tag{16}$$

Second part - Identification of linear models for dynamical systems.

1 One-step ahead predictor.

a) The model structure in this example is an ARMAX. Since we know that time shift is a useful property of a transform we can use the expression and equation (1.58) from the lecture notes book to rewrite the equation given in the task to:

$$y(t) + a_1 y(t) q^{-1} + a_2 y(t) q^{-2} = b_0 u(t) + e(t) + c_1 e(t) q^{-1} \tag{17}$$

By rearranging we get the expression as follows:

$$y(t) = \frac{b_0}{1 + a_1 q^{-1} + a_2 q^{-2}} u(t) + \frac{1 + c_1 q^{-1}}{1 + a_1 q^{-1} + a_2 q^{-2}} e(t) \tag{18}$$

We can now clearly see that the model is an ARMAX when comparing our model above (18) with (5.47a) from the lecture notes.

b) In this task we are supposed to find the expression for plant model G and noise model H. From the previous question and equation (18) we can see that the expressions for the plant and noise model are easy to extract to the following:

$$G(q, \theta) = \frac{b_0}{1 + a_1 q^{-1} + a_2 q^{-2}}$$

$$H(q, \theta) = \frac{1 + c_1 q^{-1}}{1 + a_1 q^{-1} + a_2 q^{-2}}$$

c) In this task we start by comparing the expression for the ARMAX model from equation (5.47a) in the lecture notes book with our equation (18) above. We can then identify the different terms in the ARMAX model from our equation to:

$$A(q, \theta) = 1 + a_1 q^{-1} + a_2 q^{-2}$$

$$B(q, \theta) = b_0$$

$$C(q, \theta) = 1 + c_1 q^{-1}$$

The expression for a 1-step-ahead prediction for a ARMAX is also described in the lecture notes book equation (5.47a) and if we use our expressions for the different terms above in the equation for a 1-step-ahead prediction we get the following which is the 1-step-ahead prediction in our case:

$$\hat{y}(t \mid t-1, \theta) = \frac{b_0}{1 + c_1 q^{-1}} u(t) + \frac{c_1 q^{-1} - a_1 q^{-1} - a_2 q^{-2}}{1 + c_1 q^{-1}} y(t) \tag{19}$$

d) We can rewrite the expression for an ARMAX model as described in the lecture notes in the book equations (5.51) and (5.52). We can then see that we have values i.e. θ in the prediction error $\epsilon(t, \theta)$ that both depend on the current and previous time step which means its nonlinear. To conclude the function is **<u>NOT</u>** a linear function of the parameters.

2 Prediction or simulation?

a) The model structure in this example is an OE. We use the same argument as in second part, task 1. a) with the new given expression which gives:

$$y(t) + a_1 q^{-1} y(t) = b_0 u(t) + e(t) + a_1 q^{-1} c(t) \tag{20}$$

By rearranging we get the expression as follows:

$$y(t) = \frac{b_0}{1 + a_1 q^{-1}} u(t) + e(t) \tag{21}$$

We can now clearly see that the model is an OE when comparing our model above (21) with the expression (5.47b) from the lecture notes.

b) From equation (21) and the equation (5.47b) in the lecture notes book we can identify our terms of B and F as:

$$B(q, \theta) = b_0$$

$$F(q, \theta) = 1 + a_1 q^{-1}$$

From equation (5.47b) in the lecture notes book we also get the expression for a 1-step-ahead prediction by using our terms described above and insert them into the equation:

$$\hat{y}(t \mid t-1, \theta) = \frac{b_0}{1 + a_1 q^{-1}} u(t) \tag{22}$$

We would say that the weird part is that the 1-step-ahead prediction only depend on the input to the model and not on previous output steps this is because the expression above (22) only consider $u(t)$ and not$y(t)$.

3 Computer exercise. Identification of an ARX model.

a) (The code for generating the parameters and for calculating the RMSE for both the prediction and simulation sets are attached.)

The parameters to each of the models are presented down below in the table:

Model 1	$b_0 = 0.068808$		$a_1 = -0.95898$	$a_2 = 0.35672$	
Model 2	$b_0 = 0.011291$	$b_1 = 0.99458$	$a_1 = -0.89231$	$a_2 = 0.3116$	
Model 3		$b_1 = 0.99996$	$a_1 = -1$	$a_2 = 0.6$	$a_3 = -0.3$

b) The results from predicting and simulating are shown in the table down below, where the RMSE are printed for the prediction and simulation for each model:

	Prediction	**Simulation**
Model 1	$RMSE = 1.0582$	$RMSE = 1.3636$
Model 2	$RMSE = 0.31664$	$RMSE = 0.41933$
Model 3	$RMSE = 4.3927e - 05$	$RMSE = 6.163e - 05$

We can from this result see that the root mean squared error are the least for **Model 3** both in the predicted and simulated case.