Lab 4: Model Order Selection

Ali Al-Zawqari

2024-2025

1 Lab objectives

The goal of this lab is to:

- Select a "good but parsimonious" model (low number of model parameters).
- Analyze the effects of the number of model parameters on the value of the cost function.
- Determine the dependence of the model order on
  - the signal-to-noise ratio of the measured data
  - the number of data points used to extract the model.
- Perform a large number of repeated model order estimations to assess the robustness of the AIC criterion and the use of validation data set.

2 Introduction

2.1 Exact model

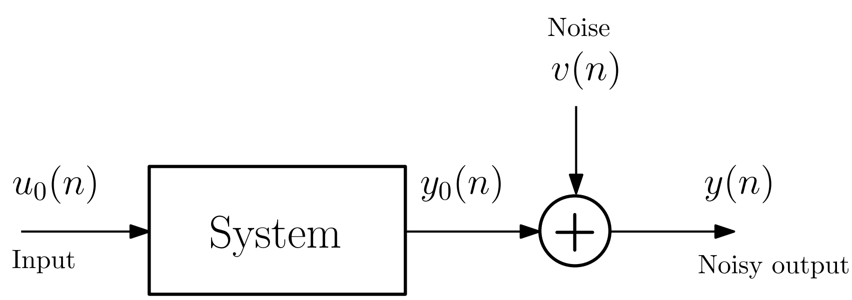

Figure 1: Measurement setup

We consider a setup as described by Figure 1, in which an input signal $u_0(n)$ is applied to an LTI system, and the signal $y_0(n)$ is the output signal. The measured output $y(n)$ is corrupted by additive noise $v(n)$, resulting in $y(n) = y_0(n) + v(n)$.

The dynamics of the system under test can be represented by an infinite length impulse response, and we assume that they can be well approximated by a finite length impulse response.

This truncated impulse response is considered to be the exact model $G_0(z)$ that needs to be estimated. It is described by:

$$G_0(z) = \sum_{i=0}^{n_H-1} g_{0i} z^i. \tag{1}$$

2.2 Fitted model

The model class selected to model $G_0(z)$ is a discrete-time FIR:

$$G(z) = \sum_{i=0}^{n_p-1} g_i z^i. \tag{2}$$

The model parameters g_i $(i = [0, n_p - 1])$ need to be estimated. Furthermore, we want to estimate an optimal value n_p for the model order. This value is to be optimal in the sense that it explains the system dynamics, but does not model the perturbation noise.

Given the FIR model, the input-output relationship is expressed by:

$$y_s(n) = \sum_{i=0}^{n_p-1} u_{0s}(n-i)\,\hat{g}_i \tag{3}$$

To vectorize the estimator, the impulse response coefficients $\hat{g}_k$ are stacked in the parameter vector θ_n:

$$\theta_n = \begin{bmatrix} \hat{g}_0 \\ \vdots \\ \hat{g}_{n_p-1} \end{bmatrix} \tag{4}$$

Equation (3) can be vectorized as follows:

$$\mathbf{Y}_s = \mathbf{H_n}(\mathbf{u_{0s}})\theta_n \tag{5}$$

With:

$\mathbf{Y}_s$:	size $N_s \times 1$	The vector of *measured output* values
$\mathbf{H_n}(\mathbf{u_{0s}})$:	size $N_s \times n_p$	The *regressor* or *observation* matrix of the measured input samples

As the matrix $\mathbf{H_n}(\mathbf{u_{0s}})$ is independent of the parameters θ_n, the model is said to be *linear-in-the-parameters.*

2.3 Impulse response estimation for an arbitrary n

Starting from a set s of input-output measurements, the matrices $\mathbf{Y}_s$ and $\mathbf{H_n}(\mathbf{u_{0s}})$ are built, and an estimate of the impulse response coefficients can be obtained by minimizing the normalized least squares cost function:

$$V_{\mathrm{LS}}\left(\theta_n, N_s\right) = \frac{1}{N_s \sigma_v^2} \left|\mathbf{Y}_s - \mathbf{H_n}\left(u_{0s}\right)\theta_n\right|^2 \tag{6}$$

Since the model is linear-in-the-parameters, the least squares estimate $\hat{\theta}_{n,\mathrm{LS}}$ can theoretically be obtained as follows:

$$\hat{\theta}_{n,\mathrm{LS}} = (\mathbf{H_n^T H_n})^{-1}\mathbf{H_n^T Y_s} \tag{7}$$

In practice this equation is never used. A numerically stable alternative way to calculate the minimizer is obtained by using the matlab \ operator instead.

The quality of the estimated impulse response $\hat{\theta}_{n,\mathrm{LS}}$ strongly depends on the model order n, the length of the measured data vectors N_s, and the signal-to-noise-ratio of the output data.

3 Practical steps

The exact model $H_0(z)$ is excited to generate input-output sequences, allowing to perform the linear least squares and estimate the parameters of $G(z)$. $G(z)$ represents the *finite impulse response* model (FIR-system) of length n_p (the model order), and we will vary the length n_p of the model between 1 and a maximum value $n_{\max}$, and estimate a new model for each model order separately. The following steps will guide you in the procedure.

3.1 Build the discrete-time test system

1. Obtain the transfer function $F_0\left(z\right)$, representing the system. It is a discrete time Chebycheff band pass filter, with a pass band ranging from $\omega = 0.3$ to $\omega = 0.6$. The ripple in the pass band is chosen to be 3 dB. and the order of the filter to be 2. Use the `cheby2` command to generate the rational form describing the transfer function.

2. Calculate the discrete impulse response of $F_0\left(z\right)$. To this end, you create a `SYS` object for the filter obtained above using the `tf(num,denom,1)` command. The impulse response is then calculated using `impulse`.

3. Truncate that impulse response to a length n_H that is sufficient to contain the main contribution to the impulse response. The truncated impulse response is now chosen to represent the exact model $G_0\left(z\right)$. **This is the model we are going to estimate.**

3.2 Generate the excitation and output signals

1. Generate the excitation signals $u_{0e}(n)$ and $u_{0v}(n)$, representing respectively the input signals of the estimation and the validation datasets. Both signals are discrete time random sequences, normally distributed and with unity standard deviation. Select the length N_e of $u_{0e}(n)$ to be at least 10 times longer than the impulse response, and the length N_v of $u_{0v}(n)$ similar to N_e.

2. Compute the exact output sequences $y_{0e}(n)$ and $y_{0v}(n)$. The signals are obtained by respectively filtering $u_e(n)$ and $u_v(n)$ using the truncate impulse response $G_0(z)$:

$$y_{0s}(n) = \sum_{i=0}^{n_H-1} g_{0i} u_{0s}(n-i) \tag{8}$$

   The index s represents the dataset considered, and it is equal to e for the estimation dataset and v for the validation dataset.

   Use the `filter` command to calculate this convolution sum and obtain the exact output sequences.

3. Add noise perturbations to the exact output sequences, to obtain the 'measured' estimation and validation output signals $y_e(n)$ and $y_v(n)$. The variance of the noise perturbation sequences is set to result in a signal to noise ratio between the exact output and the noise perturbation around 6 dB.

3.3 Implement the least squares for varying n

1. Construct the matrix $\mathbf{H_n}(\mathbf{u_{0s}})$, as defined in Equation 5. To avoid having to repeat this construction over and over again in the rest of the lab, construct the matrix for the maximum length $n_{\max}$ of θ. The regressor matrix is obtained with the input signal $u_{0e}(n)$. The definition of the regressor matrix can be obtained by rewriting Equation 3 in matrix form, as: $\mathbf{Y}_e = \mathbf{H_n}(\mathbf{u_{0e}})\theta_n$. (Hint: Look up at the matlab command *toeplitz* to build $\mathbf{H_n}(\mathbf{u_{0e}})$).

2. Implement the least square estimation using the estimation dataset to obtain the estimate $\hat{\theta}_{n,\mathrm{LS}}$. Make sure that variables such as the model order n_p and the length of the signals N_e are not hard coded, so that you can examine different settings easily.

3. Evaluate the value of the LS cost function described in Equation 6 for $\theta_n = \hat{\theta}_{n,\mathrm{LS}}$.

4. Repeat steps 3.3.2 and 3.3.3 for parameter order n varying from $n = 1, \ldots, n_{\max}$. Make sure to save, at every iteration, the value of the LS cost function, to obtain the curve $V_{\mathrm{LS}}\left(\hat{\theta}_n, N_e\right)$:

$$V_{\mathrm{LS}}\left(\hat{\theta}_n, N_e\right) = \frac{1}{N_e \sigma_{ve}^2} |\mathbf{Y}_e - \mathbf{H_n}(u_{0e})\,\hat{\theta}_n|^2 \tag{9}$$

3.4 Select the optimal model order using AIC and Validation

1. Obtain the cost function for the AKAIKE Information Criteria (AIC), defined below, and obtain an estimate of the optimal model order by selecting the value of n that minimizes $V_{\text{AIC}}\left(\hat{\theta}_{n,\text{LS}}, N_e\right)$:

$$V_{\text{AIC}}\left(\hat{\theta}_{n,\text{LS}}, N_e\right) = V(\hat{\theta}_{n,\text{LS}}, N_e)\left(1 + \frac{2n}{N_e}\right) \tag{10}$$

2. Use the validation dataset to build the evolution of the least squares validation cost function $V_{\text{val}}\left(\hat{\theta}_n, N_v\right)$ (obtained using $\hat{\theta}_{n,\text{LS}}$ computed in Section 3.3) and defined as below, and obtain an estimate of the optimal model order by selecting the value of n that minimizes $V_{\text{val}}\left(\hat{\theta}_n, N_v\right)$.

$$V_{\text{val}}\left(\hat{\theta}_n, N_v\right) = \frac{1}{N_v \sigma_{vv}^2}|\mathbf{Y_v} - \mathbf{H_n}\left(u_{0v}\right)\hat{\theta}_n|^2 \tag{11}$$

3. Plot together the evolution of V_{LS}, V_{val},V_{AIC} and compare the estimate of the optimal model order (the minima of each curve) between the methods. Compare and interpret the three curves.

3.5 Explore the effects of SNR and bandwidth

1. Increase the SNR by 20 dB. Repeat the estimation and the model order selection and interpret the results. What are the differences and why do they occur?

2. Decrease the bandwidth of the original system. What influence does that have on the optimal model order? Explain.

3.6 Determine the robustness of the model order estimation using V_{AIC}, V_{val}, V_{LS}

1. Repeat the complete identification run 100 times. For each run

   - Simulate a new data set adding a new perturbation noise set to the exact output data and validation data set
   - Estimate the least squares FIR model for all model orders ranging from 1 to 100
   - Calculate and save V_{AIC}, V_{val}, V_{LS} for each model order
   - Determine the value and the position of the minimum value of each vector of cost functions. The position of the minimum indicates the estimated model order. Save these values for each run.

2. Put the estimated model orders in a histogram. Make sure to use 100 classes of unity width in the histogram. This is obtained adding the vector `1:100` as a parameter to the `hist` function.

3. Discuss the results obtained for the three cost functions. Does this meet your intuitive expectations?

4. Repeat the process for a different signal-to-noise ratio, and discuss the changes you observe

4 Submission

In this lab, students are required to submit the following:

1. The solution code for the practical steps outlined in Section 3. This code must be submitted exclusively in MATLAB. Ensure that your MATLAB code is well-organized and includes comments to clarify the functionality and process flow.

2. A detailed report discussing the results obtained from each part of the lab. The report should be formatted as a maximum two-page, double-column document. This report must be submitted by each group and is due two weeks after the completion of the lab session.

Both components should address the requirements specified in Section 3 comprehensively. Ensure that your submissions are clear, well-documented, and demonstrate a thorough analysis of the lab results.