227-0689-00L: System Identification HS2023

Michele Zaffalon (Bruker BioSpin AG)

Yannic Hofmann (ETHZ)

December 16, 2023

Unreviewed notes for Prof. Smith's class. Use at own risk!

Contents

Chapter 1

Least-Squares Estimation

A word about notation: in class we used Y and Φ for the measurements and regressor matrix and ϵ for the (possibly correlated) noise. However because the noise is Gaussian, the equations below show that the terms scaled by the Cholesky decomposition C of the noise correlation matrix R are the relevant ones. With the scaling, the scaled noise e becomes uncorrelated and has variance $\sigma^2 = 1$. I personally like this approach.

Consider the model

$$\tilde{Y} = \tilde{\Phi}\theta_0 + \tilde{\epsilon} \tag{1.1}$$

where $\tilde{Y} = \begin{bmatrix} y_0 & \dots & y_{N-1} \end{bmatrix}^\top$ is a vector containing the measurements $\{y_0, \dots, y_{N-1}\}$, $\tilde{\Phi} \in \mathbb{R}^{N \times p}$ is called the *regressor* and $\theta \in \mathbb{R}^p$ is the model parameter to be estimated; p is the model order. The noise vector $\tilde{\epsilon} = \begin{bmatrix} v_0 & \dots & v_{N-1} \end{bmatrix}^\top$ has zero mean $\mathbb{E}\{\tilde{\epsilon}\} = 0$ and covariance[1] $\mathbb{E}\left\{\tilde{\epsilon}\tilde{\epsilon}^\top\right\} = R$, a symmetric positive definite matrix.

The maximum likelihood (ML) probability[2] requires us to minimize

$$(\tilde{Y} - \tilde{\Phi}\theta)^\top R^{-1} (\tilde{Y} - \tilde{\Phi}\theta) = ||C^{-1}(\tilde{Y} - \tilde{\Phi}\theta)||_2^2 = ||Y - \Phi\theta||_2^2 \tag{1.2}$$

[1] Note also that while the matrix $vv^\top$ has rank 1, the noise covariance matrix R for Gaussian distributed white noise is full rank because it is the sum of random vectors that span the whole space.

[2] The expectation value of eq. (1.2) is $N - p$. This can be seen using eq. (1.5)

$$Y - \Phi\hat{\theta}_{\mathrm{ML}} = (\Phi\theta_0 + e) - \Phi\left(\theta_0 + K\Phi^\top e\right) = \left(I - \Phi K\Phi^\top\right) e.$$

The average

$$\mathbb{E}\left\{\left|\left|Y - \Phi\hat{\theta}_{\mathrm{ML}}\right|\right|_2^2\right\} = \mathbb{E}\left\{e^\top \left(I - \Phi K\Phi^\top\right)^\top \left(I - \Phi K\Phi^\top\right) e\right\} = \mathbb{E}\left\{e^\top \left(I - \Phi K\Phi^\top\right) e\right\}.$$

where C is the Cholesky decomposition of $R = CC^\top$ (and $R^{-1} = \left(C^{-1}\right)^\top C^{-1}$), $Y = C^{-1}\tilde{Y}$, $\Phi = C^{-1}\tilde{\Phi}$ and $e = C^{-1}\tilde{\epsilon}$. Note that the scaled noise e is uncorrelated and has unit variance:

$$\mathbb{E}\left\{ee^\top\right\} = \mathbb{E}\left\{C^{-1}\tilde{\epsilon}\tilde{\epsilon}^\top\left(C^{-1}\right)^\top\right\} = C^{-1}R\left(C^{-1}\right)^\top = I.$$

The *mathematical* solution to eq. (1.2) is found by setting the gradient of the expression with respect to θ to zero, which gives the normal equation

$$\left(\Phi^\top\Phi\right)^\top\hat{\theta}_{\mathrm{ML}} = \Phi^\top Y \;\rightarrow\; \hat{\theta}_{\mathrm{ML}} = \left(\Phi^\top\Phi\right)^{-1}\Phi^\top Y. \tag{1.3}$$

The solution exists provided Φ has full column rank[3]: $\mathrm{rank}\,(\Phi) = p$. When this is the case, the system is said to be *persistently excited*, see Sect.. and when given the freedom, one chooses Φ so that it is well conditioned.

Had we not scaled $\tilde{Y}$ and $\tilde{\Phi}$ by the noise covariance, the solution would have been

$$\hat{\theta}_{\mathrm{ML}} = \left(\tilde{\Phi}^\top R^{-1}\tilde{\Phi}\right)^{-1}\tilde{\Phi}^\top R^{-1}\tilde{Y}.$$

Numerically one should not form the normal equation directly because it squares the condition number of Φ and rely either on the QR decomposition or on the SVD to solve eq. (1.2). This is taken care automatically by MATLAB when using the backslash $\backslash$ operator

$$\hat{\theta}_{\mathrm{ML}} = \Phi\backslash Y. \tag{1.4}$$

The term $\mathbb{E}\left\{e^\top e\right\}$ evaluates to N. The other term evaluates to p: using $\Phi = U\Sigma V^\top$,

$$(\Phi^\top\Phi)^{-1} = \left(V\Sigma^\top U^\top U\Sigma V^\top\right)^{-1} = V\left(\Sigma^\top\Sigma\right)^{-1}V^\top$$

and

$$\Phi V\left(\Sigma^\top\Sigma\right)^{-1}V^\top\Phi^\top = U\Sigma\left(\Sigma^\top\Sigma\right)^{-1}\Sigma^\top U$$

where the term between the two U is a tall matrix of dimension $N \times p$ with ones on the top block's main diagonal (of dimension $p \times p$) and zeros in the bottom block.

[3]The usual warning holds for the rank. Instead we want to have the matrix Φ with the smallest condition number for the estimate to be numerically stable, which is a stronger condition than full rank.

1.1 Bias, Covariance and MSE of the Least Squares Estimation

The linear estimator eq. (1.3) is unbiased[4] (but see also Sect. 2.1.2):

$$\mathbb{E}\left\{\hat{\theta}_{\mathrm{ML}}\right\} = \theta_0.$$

The covariance[5]

$$\mathrm{cov}\left(\hat{\theta}_{\mathrm{ML}}\right) = \left(\Phi^\top \Phi\right)^{-1}. \tag{1.6}$$

Lastly, we consider the mean squared error

$$\mathrm{MSE}\left(\hat{\theta}_{\mathrm{ML}}\right) = \underbrace{\left|\left|\mathrm{Bias}\left(\hat{\theta}_{\mathrm{ML}}\right)\right|\right|_2^2}_{=0} + \mathrm{tr}\left(\mathrm{cov}\left(\hat{\theta}_{\mathrm{ML}}\right)\right)$$

which reduces to N for the case of uncorrelated noise (I am not sure about this anymore).

1.1.1 Geometric Interpretation of Least-Squares

The least squares problem

$$||b - Ax||_2$$

has the following geometric interpretation: the solution is that for which the residuals $v \doteq b - Ax$ are outside (i.e. orthogonal) of the space spanned by A.

[4] Recalling that $\Phi^\top \Phi$ is a symmetric matrix, letting $K \doteq (\Phi^\top \Phi)^{-1}$ and using eq. (1.1) into eq. (1.3), we obtain

$$\hat{\theta}_{\mathrm{ML}} = K\Phi^\top(\Phi\theta_0 + e) = \theta_0 + K\Phi^\top e \tag{1.5}$$

from which $\mathbb{E}\left\{\hat{\theta}_{\mathrm{ML}}\right\} = \theta_0$ since $\mathbb{E}\{e\} = 0$. Moreover

$$\begin{aligned}
\mathrm{cov}\left(\hat{\theta}_{\mathrm{ML}}\right) &\doteq \mathbb{E}\left\{\left(\hat{\theta}_{\mathrm{ML}} - \mathbb{E}\left\{\hat{\theta}_{\mathrm{ML}}\right\}\right)\left(\hat{\theta}_{\mathrm{ML}} - \mathbb{E}\left\{\hat{\theta}_{\mathrm{ML}}\right\}\right)^\top\right\} \\
&= \mathbb{E}\left\{\left(K\Phi^\top e\right)\left(K\Phi^\top e\right)^\top\right\} \\
&= \mathbb{E}\left\{K\Phi^\top e e^\top \Phi K\right\} = K\Phi^\top \mathbb{E}\left\{ee^\top\right\}\Phi K = K.
\end{aligned}$$

[5] Is there a way to understand the form of covariance matrix without going through the calculation?

In other words, we require[6] the scalar product $\langle Az, v\rangle$ to be zero for all z:

$$\begin{aligned} 0 = \langle Az, v\rangle = (Az)^\top (b - Ax) = z^\top \left(A^\top b - A^\top Ax\right) \qquad \forall z \\ \rightarrow A^\top Ax = A^\top b \end{aligned}$$

which is the normal equation.

1.2 Random Notes: The Covariance Matrix and the Choice of the Measurement Points

These are my considerations that are not part of the lecture.

- The *off-diagonal* elements of the covariance matrix $\operatorname{cov}\left(\hat{\theta}_{\mathrm{ML}}\right)$ eq. (1.6) represent the correlations between the errors of the variables θ. It is therefore not justified to discard them and take θ's standard deviations as the square root of $\operatorname{cov}(\theta)$'s diagonal elements because one discards the correlations: the ball of probability is in general an ellipse with the axes not parallel to the variable directions.

- Given the freedom to choose the measurement points, is there a "best" way of placing them? Is this done in practice?

  In the context of experiment design, this is done.

  There are two factors that determine the covariance matrix: the choice of basis and the choice of points. The choice of basis is determined by the variables that one wants to extract: a linear transformation between one basis and the other will also transform the covariance matrix and the only concern may be the numerical stability (although one should expect that the measurement errors dominate).

  Determining the position of the best measurement points by minimizing one (or more elements of the covariance) is in general a non-convex problem.

[6] I believe the proof given in class is not correct: to span the full column space of A, one has to multiply by a generic vector z; in class $z = x$ was taken.

Chapter 2

Regularized FIR Models

We have seen that the solution $\hat{\theta}_{\mathrm{ML}}$ of the least squares problem eq. (1.2)

$$\arg\min_{\theta} ||Y - \Phi\theta||_2^2$$

is unbiased, see Sect. 1.1.

We can however choose to have a biased estimate to reduce the mean square error

$$\mathrm{MSE}\left(\hat{\theta}\right) = E||\hat{\theta} - \theta_0||_2^2.$$

This can be achieved if we modify the minimization problem by adding a regularization term

$$||Y - \Phi\theta||_2^2 + \gamma\theta^\top P^{-1}\theta \tag{2.1}$$

where γP^{-1} is a positive definite matrix. P is called the *kernel* or the regularization matrix[1].

Regularization prevents overfitting, reduces the sensitivity to noise and can improve the estimate by a proper choice of the kernel (*e.g.* for instance if the system is known to be stable, this information can be used to improve the estimation.)

The closed form mathematical solution to eq. (2.1) is given by

$$\hat{\theta}^{\mathrm{R}} = \left(\Phi^\top\Phi + \gamma P^{-1}\right)^{-1}\Phi^\top Y = \left(P\Phi^\top\Phi + \gamma I\right)^{-1}P\Phi^\top Y. \tag{2.2}$$

[1]When $P = \mathbb{I}_N$ the technique is called ridge regression, otherwise it goes under the name of Tikhonov regularization. Compared to the ridge regression which only tries to decrease $||\theta||^2$, the Tikhonov regularization can use information about the system, see Sect. 2.1.1.

2.0.1 The James-Stein Estimator

The James-Stein estimator was the first biased estimator that had a smaller MSE compared to least squares on all the [1, page 3]. It can be cast into a ridge regression problem

$$||Y-\theta||^2+\gamma||\theta||^2, \qquad \gamma=\frac{(N-2)\sigma^2}{||Y||^2-(N-2)\sigma^2}.$$

2.1 Bias, Covariance and MSE of the Regularized Least Squares Estimation

We assume uncorrelated noise with constant variance 1: $\mathbb{E}\left\{ee^\top\right\}=I_N$.

The estimate eq. (2.2) has bias[2], covariance[3] and MSE

$$\begin{aligned}
\text{Bias}\left(\hat{\theta}^{\text{R}}\right) &= -\left(\Phi^\top\Phi+\gamma P^{-1}\right)^{-1}\gamma P^{-1}\theta_0 \\
\text{cov}\left(\hat{\theta}^{\text{R}}\right) &= \left(\Phi^\top\Phi+\gamma P^{-1}\right)^{-1}\Phi^\top\Phi\left(\Phi^\top\Phi+\gamma P^{-1}\right)^{-1} \\
\text{MSE}\left(\hat{\theta}^{\text{R}}\right) &= \left|\left|\text{Bias}\left(\hat{\theta}^{\text{R}}\right)\right|\right|_2^2+\text{tr}\left(\text{cov}\left(\hat{\theta}^{\text{R}}\right)\right)
\end{aligned}$$

2.1.1 Choice of the Regularization Matrices

The MSE is minimized by this choice of parameters

$$\gamma^\star=1, \quad P^\star=\theta_0\,\theta_0^\top. \tag{2.3}$$

The optimal regularization matrix $P^\star$ is unknown[4] because it depends on the unknown θ_0 but the approximate knowledge of the solution helps to construct a "good" P.

---

[2] Let $K=\left(\Phi^\top\Phi+\gamma P^{-1}\right)^{-1}$ so that $\hat{\theta}^{\text{R}}=K\Phi^\top Y$. Plugging $Y=\Phi\theta_0+e$, we have $\mathbb{E}\left\{\hat{\theta}^{\text{R}}\right\}=K\Phi^\top\Phi\theta_0$. The bias is

$$\mathbb{E}\left\{\hat{\theta}^{\text{R}}\right\}-\theta_0=(K\Phi^\top\Phi-I)\theta_0=K(\Phi^\top\Phi-K^{-1})\theta_0=-\gamma KP^{-1}\theta_0.$$

[3] With the same notation as before, we have $\hat{\theta}^{\text{R}}-\mathbb{E}\left\{\hat{\theta}^{\text{R}}\right\}=K\Phi^\top e$. The covariance is

$$\text{cov}\left(\hat{\theta}^{\text{R}}\right)=\mathbb{E}\left\{\left(K\Phi^\top e\right)\left(K\Phi^\top e\right)^\top\right\}=K\Phi^\top\mathbb{E}\left\{ee^\top\right\}\Phi K=K\Phi^\top\Phi K.$$

[4] $P^\star=\theta_0\,\theta_0^\top$ has rank 1 and is therefore not invertible. To get a well-

Regularization functions are designed to encode prior knowledge on the unknown system; some choices will result in better estimates. The choice can be either subjective or based on cross-validation and different regularization functions/kernels give rise to different model classes. Technically there is no wrong regularization function, but there is an optimal one (in the sense of minimum MSE).

A quadratic regularization function based on the TC kernel is well-suited for exponentially decaying pulse responses of stable linear systems. In that case, a high order FIR model is used to estimate the first significant part of the pulse response of the unknown system, ignoring the tail. An FIR system is always stable, but its coefficients may not be exponentially decaying. An impulse response experiment may be used to reveal some properties of the system if the system and experimental configuration allow it. [Mohamed Abdalmoaty on Moodle]

A commonly used regularization matrix is the diagonally-correlated (DC) kernel

$$[P]_{ij} = c\alpha^{\frac{i+j}{2}}\rho^{|i-j|}$$

with $0 \leq c$, $0 \leq \alpha \leq 1$ and $-1 \leq \rho \leq 1$: ρ describes correlations α and the decays.

A simpler version of it is the tuned-correlated (TC) kernel[5] where one lets $\rho = \sqrt{\alpha}$ in the DC kernel

$$[P]_{ij} = c\alpha^{\max\{i,j\}} = \begin{bmatrix} \alpha & \alpha^2 & \alpha^3 & \dots & \alpha^n \\ \alpha^2 & \alpha^2 & \alpha^3 & \dots & \alpha^n \\ \alpha^3 & \alpha^3 & \alpha^3 & \dots & \alpha^n \\ \dots & & & & \\ \alpha^n & \alpha^n & \alpha^n & \dots & \alpha^n \end{bmatrix}$$

defined problem, the inverse P^{-1} is replaced by the Moore-Penrose pseudoinverse P^{+}. It turns out that the solution of the problem defined with $P^{\dagger}$ is also equal to $(P\Phi^{\top}\Phi + \gamma I)^{-1}P\Phi^{\top}Y$. [Mohamed Abdalmoaty on Moodle]

[5]Despite having entries that are exponential, the TC kernel is *not* $\theta\theta^{\top}$

$$\theta\theta^{\top} = \begin{bmatrix} a & a^2 & a^3 & \dots \\ a^2 & a^3 & \dots & \\ a^3 & \dots & & \\ \dots & & & \end{bmatrix} \quad \text{with } \theta = \begin{bmatrix} a & a^2 & \dots \end{bmatrix}$$

coming from the truncation of the IIR filter $\frac{z^{-1}}{1-az^{-1}} \approx z^{-1}(1 + az^{-1} + a^2z^{-2} + \dots)$.

The top left entries are large, the bottom right small. Since the inverse enters the minimization problem (cost function), the top left part is small whereas the bottom down are large and induce a larger cost.

The inverse of the DC kernel is a tridiagonal with the explicit form

$$[P^{-1}]_{ij} = \frac{c_{ij}}{1-\rho^2}(-1)^{i+j}\alpha^{-\frac{i+j}{2}}\rho^{|i-j|}, \qquad c_{ij} = \begin{cases} 1+\rho^2 & \text{if } i=j=2,\dots,p-1 \\ 0 & \text{if } |i-j|>1 \\ 1 & \text{otherwise} \end{cases}$$

2.1.2 Estimation Bias

> One cannot say that the linear least-squares method is unbiased in general. There are three elements at play: the data set, the model used, and the estimation method. Least-squares is just the estimation method. To check if it is unbiased, an assumption has to be made on the true system that generated the data, and the model used to get the closed form expression of the LS estimator. [Mohamed Abdalmoaty on Moodle]

In the following, the input u is known, the measurement noise e has zero-mean, the data is generated as $Y = \Phi\theta_0 + e$ with $\theta_0 \in R^p$; we estimate $\theta \in R^q$ in the linear regression model $Y = \Phi_q\theta + e$; and we use the least-squares estimation method: then $\hat{\theta} = \left(\Phi_q^\top\Phi_q\right)^{-1}\Phi_q^\top Y$.

Here some examples of source of biased estimate:

- The model does not match the generating data. One underestimates the length of the response by choosing the model order too small, $q < p = \tau_{\max}$: this is the truncation error of Sect.. (see slide 10–10). Indeed we are trying to fit the data with a model of order q but the model has order p. The estimate is biased because

  $$\mathbb{E}\left\{\hat{\theta}\right\} = \left(\Phi_q^\top\Phi_q\right)^{-1}\Phi_q^\top\Phi\theta_0 \neq \theta_0.$$

  This means the trade-off bias-variance can also be done with LS by varying the order parameter, but to a lower extent than using regularization where one has extra parameters available.

  Note that here the regressor involves only the known (noise-free) input u;

- The data comes from an ARX system with $p = n+m$ parameters and $\theta_0 \in R^{n+m}$ is the true parameter. As opposed to the case above, here

Φ involves also the random outputs: for this reason the resulting LS estimator is *biased* even though we have a correct model order/structure (*i.e.*, the model that matches the data generating system);

- Finally, even with the correct order parameter, one can *on purpose* bias $\hat{\theta}$ by adding the regularization term to $||Y - \Phi\theta||_2$.

2.1.3 Why the LS May Perform Badly

Least-squares performs badly when the regression matrix Φ is ill-conditioned[6], as a result of not being persistently excited and a small perturbation of the measurement data Y can have a large impact on the estimate $\hat{\theta}_{\text{ML}}$. Regularization helps in these cases. (Is there an easy way to show that $\Phi^\top\Phi + \gamma P^{-1}$ has a larger minimum singular value?)

2.2 Cross-Validation Methods

They are methods to select the discrete model orders (in least-squares or ARX) or the regularization parameters (*e.g.* γ and the kernel parameter α in regularized least squares).

A popular method is called hold-out cross-validation and consists of the following steps:

1. split the data into two (equal but not necessarily) non-overlapping parts, one for estimation and the other for validation. The way in which the data is split depends on the type of data: for data generated by dynamical systems, where the data is sequential in time, the order is important. Here the first N_{id} pairs of control inputs and output values $\{u(k), y(k)\}$ are used for identification and the remaining for validation. For static data, the order is not important: one could choose the odd-indexed pairs for identification and the even-indexed pairs for validation;

2. use the estimation data to estimate a model for each mode structure or each regularization parameter. In the case of the continuous parameter

[6]The condition number of a matrix is defined as the ratio between the largest σ_1 and the smallest σ_n singular value of the matrix

$$\text{cond}(A) = \frac{\sigma_1}{\sigma_n}.$$

Note that from a numerical point of view, the matrix being full rank is not a guarantee that the LS problem can be stably solved.

γ, one estimates the model on a set of gridded values. If there are two continuous parameters, *e.g.* γ and α, the gridding is done for both parameters;

3. select the model structure / regularization parameters that give a model with least prediction error on validation data;

4. use the selected model structure / regularization parameters and the complete data set to estimate a final model.

Numerical example in `10_lect/regularization.m`, for the moment limited to ridge regression.

2.3 Numerical Solution of the Tikhonov Regularization Problem

For the numerical solution I am aware of three methods:

1. Solve directly the normal equation

$$\left(\Phi^\top \Phi + \gamma P^{-1}\right) \hat{\theta} = \Phi^\top y \tag{2.4}$$

using MATLAB's backslash operator: although this squares Φ's condition number when constructing $\Phi^\top \Phi$, at least it does not require to explicitly construct the inverse;

2. Use the generalized SVD decomposition. We first manipulate eq. (2.1) to obtain

$$||y - \Phi\theta||_2^2 + ||D\theta||_2^2$$

by letting D be the Cholesky decomposition of γP^{-1}: that is $D^\top D = \gamma P^{-1}$.

There exists different definitions of generalized SVD: MATLAB implements the call `[U,V,X,C,S] = gsvd(Φ, D)` where U and V are unitary matrices, and C and S are nonnegative diagonal matrices such that

$$\Phi = UCX^\top \qquad D = VSX^\top \qquad C^2 + S^2 = I.$$

Inserting the decompositions into eq. (2.4) gives

$$X^\top \theta = (UC)^\top y$$

which can be solved as $\theta = X^\top \backslash (UC)^\top y$;

3. Eq. (2.1) can also be rewritten as

$$\left|\left|\begin{bmatrix}\Phi\\ D\end{bmatrix}\theta - \begin{bmatrix}y\\ 0\end{bmatrix}\right|\right|_2^2$$

and solved by backslash.

I tested the three approaches on prob. 8 and found the fastest to be the third which does not construct the normal equation and is therefore better conditioned at the small cost of increasing the problem size by the r rows of P^{-1}. The first method is twice slower and the second is 100 (!) times slower: the problem is most likely the call to `gsvd`. This is strange because this is the method normally suggested, but probably only for larger regularization matrices.

Chapter 3

Error Prediction Methods and Transfer Function Models

We assume that the complete system model is given by

$$y(k) = G(z)u(k) + H(z)e(k). \tag{3.1}$$

Prediction error-based identification methods estimate the transfer functions $G(z)$ and $H(z)$, parametrised as $G(z,\theta)$ and $H(z,\theta)$, by minimizing the objective function $J(\epsilon)$, a function of the prediction error $\epsilon(k,\theta)$

$$\epsilon(k,\theta) \doteq y(k) - \hat{y}(k,\theta).$$

The prediction $\hat{y}(k,\theta)$ for the time k is a function of the previous measurements and inputs at $k-1,\ k-2,\ldots$ only.

3.1 Prediction

We assume that $H(z)$ is monic[1] and stable[2]. In this case, given the filtered noise $v(k) = H(z)e(k)$, $e(k)$ can be reconstructed from $v(k)$ as

$$e(k) = H^{-1}(z)v(k). \tag{3.2}$$

[1] Monic means that $h(0) = 1$:

$$H(z) = 1 + \sum_{k=1}^{\infty} h(k)z^{-k}.$$

[2] $H(z)$ has only poles strictly inside the unit circle.

We now seek to predict $v(k)$ given the past values up to time $k-1$: this is called the *one-step ahead* estimate. Since $H(z)$ is monic, we split the filtered noise contribution into the term $e(k)$ and other terms up to time $k-1$

$$v(k) = H(z)e(k) = e(k) + (H(z) - 1)\, e(k) \tag{3.3}$$

The *predicted* filtered noise $\hat{v}(k|k-1)$ is

$$\hat{v}(k|k-1) = (H(z) - 1)e(k).$$

This can be intuitively understood because the error probability function distribution for $\{e\}$ has zero mean[3]: if we were left to guess $e(k)$, we would guess $e(k) = 0$. Making use of eq. (3.2), the one-step ahead estimate

$$\hat{v}(k|k-1) = \left(1 - H^{-1}(z)\right) v(k) \tag{3.4}$$

is determined only from the knowledge of the past values of v up to $k-1$.

For the model of eq. (3.1), the one-step ahead predictor

$$\hat{y}(k|k-1) = G(z)u(k) + \hat{v}(k|k-1) \tag{3.5}$$

can be rewritten with the help of the expression for $\hat{v}(k|k-1)$ as[4]

$$\hat{y}(k|k-1) = H^{-1}(z)G(z)u(k) + \left(1 - H^{-1}(z)\right) y(k). \tag{3.6}$$

The prediction error[5]

$$\epsilon(k) = y(k) - \hat{y}(k|k-1) = e(k) \tag{3.7}$$

[3]Had the probability distribution $f_e(x)$ not had a zero mean, we would have to modify the predition according to

$$\hat{v}(k|k-1) = \arg\max_x f_e(x - m(k-1)), \qquad m(k-1) = (H(q) - 1)e(k).$$

[4]Using $v(k) = y(k) - G(z)u(k)$

$$\hat{y}(k|k-1) = G(z)u(k) + \left(1 - H^{-1}(z)\right)(y(k) - G(z)u(k)).$$

Note that $y_0(k) = G(z)u(k)$ is the evolution of the noiseless true system, so that eq. (3.5) could also be written as $\hat{y}(k|k-1) = y_0(k) + \hat{v}(k|k-1)$. However we seek an expression that involves the measured outputs and expressing the one-step head predictor as a function of the unknown true system is of no use to us.

[5]Using eq. (3.6)

$$\begin{aligned} y(k) - \hat{y}(k|k-1) &= y(k) - H^{-1}(z)G(z)u(k) - \left(1 - H^{-1}(z)\right) y(k) \\ &= H^{-1}(z)(y(k) - G(z)u(k)) \\ &= H^{-1}(z)v(k) = e(k) \end{aligned}$$

is the noise $e(k)$ that cannot be predicted: the *innovation* is the part of the output prediction that cannot be estimated from past measurements. This result is not really a surprise: it was the term that was discarded from $v(k)$ to compute the predicted noise $\hat{v}(k|k-1)$.

3.1.1 Example: Moving Average

The model

$$v(k) = e(k) + ce(k-1) \rightarrow H(z) = 1 + cz^{-1}$$

is invertible when $|c| < 1$. The one-step ahead predictor eq. (3.4) can be expressed in terms of the error $e(k-1)$ using eq. (3.7)

$$\hat{v}(k|k-1) = \left(1 - H^{-1}(z)\right) H(z)e(k) = cz^{-1}e(k) = ce(k-1)$$

3.2 Family of Transfer-Function Models

The advantage of the transfer-function models is that they can be described by fewer parameters and that the inputs required to identify the system do not have to be persistently exciting as it is the case when one wants to identify frequency or time-response: we have seen in Sect. **??** that for frequency domain methods, the order of excitation must be double the number of complex estimates of the transfer function $G(e^{j\omega_n})$ since gain and phase must be determined for each frequency; for a time response one requires the same persistency order as the number of impulse response terms.

If we control the input, this requirement is easy to satisfy. If on the other hand the data is given, this may not be the case and in these situations, one is better off looking for transfer functions/state space representations because of the reduced numbers of parameters to identify.

Prediction error-based identification methods construct the prediction error[6]

$$\epsilon(k, \theta) = y(k) - \hat{y}(k, \theta) \tag{3.8}$$

from the (one-step ahead) predictor $\hat{y}(k, \theta)$ which is based on the guesses $\hat{G}(z) = G(z, \theta)$ and $\hat{H}^{-1}(z) = H^{-1}(z, \theta)$, the guesses being parametrized by

[6]Here we are seeking the unknown parameters θ for a known model for which we can construct the one-step ahead predictor $\hat{y}(k, \theta)$, parametrized by θ. Since θ is unknown, we cannot say just yet that $\epsilon(k, \theta) = e(k)$.

I also do not know if $\min_\theta ||\epsilon(\theta)|| = e$ but I suspect this is the case when $e(k)$ is Gaussian-distributed.

θ. The optimal $\theta^\star$ is the argument that minimizes the cost function $J = J(\epsilon)$

$$\theta^\star = \arg\min_\theta J(\epsilon(k,\theta)).$$

Typical choices for the objective functions $J(\epsilon)$ are the 2-norm $||\epsilon||_2^2$ or the maximum deviation, the ∞-norm $||\epsilon||_\infty$. The kind of minimization depends on how the models $G(q)$ and $H^{-1}(q)$ are parametrised: in general, the parametrization will not be linear and the optimization may not be convex. Note moreover that the minimization of $||\epsilon||_2^2$ is not equivalent to the least squres method, unless $H(z) = 1$:

$$y(k) = G(z)u(k) + e(k).$$

Since this approach does not require an a-priori knowlege of the system, it is also called the *black-box* approach.

3.2.1 Equation Error Model Structure (ARX)

The ARX model

$$y(k) = B(z)u(k) + (1 - A(z))\, y(k) + e(k) \tag{3.9}$$

is a simple input-output relationship where the error enters as a direct term: this model covers a limited set of real-world problems, for instance those where the perturbation act as a force. We take

$$A(z) = 1 + a_1 z^{-1} + \ldots + a_n z^{-n}, \qquad B(z) = b_1 z^{-1} + \ldots + b_m z^{-m}$$

where $A(z)$ is monic and $B(z)$ does not contain a constant term, *i.e.* the model has no feed-through. It corresponds to the model of eq. (3.1) with

$$G(z) = \frac{B(z)}{A(z)}, \qquad H(z) = \frac{1}{A(z)}.$$

and generates the one-step ahead predictor

$$\hat{y}(k|k-1) = (1 - A(z))\, y(k) + B(z)u(k) \tag{3.10}$$

either by plugging $G(z)$ and $H^{-1}(z)$ in eq. (3.6) or by using the result of eq. (3.7) in eq. (3.9).

We now seek the estimates $\hat{B}(z,\theta)$ and $\hat{A}(z,\theta)$ parametrized by the *parameter vector* θ which contains the unknown coefficients of the polynomials $A(z)$ and $B(z)$

$$\theta = \begin{bmatrix} a_1 & \ldots & a_n & b_1 & \ldots & b_m \end{bmatrix}^\top$$

This generates the one-step ahead predictor

$$\hat{y}(k,\theta) = \left(1 - \hat{A}(z,\theta)\right) y(k) + \hat{B}(z)u(k).$$

that can be written in linear form

$$\varphi^\top(k) = \begin{bmatrix} -y(k-1) & \dots & -y(k-n) & u(k-1) & \dots & u(k-m) \end{bmatrix}.$$

using the *regressor vector* $\varphi(k)$ that contains the outputs and inputs.

θ is found by minimization of the prediction error $\epsilon(k,\theta) = y(k) - \hat{y}(k,\theta)$: when θ is the true parameter vector θ_0, then $\epsilon(k,\theta_0) = e(k)$. Generally, one minimizes the 2–norm $||\epsilon(k,\theta)||_2^2$: this is solved in MATLAB by

$$\hat{\theta} = \Phi \backslash Y, \qquad \Phi \doteq \begin{bmatrix} \varphi^\top(1) \\ \varphi^\top(2) \\ \dots \\ \varphi^\top(N) \end{bmatrix}, \quad Y \doteq \begin{bmatrix} y(1) \\ y(2) \\ \dots \\ y(N) \end{bmatrix} \tag{3.11}$$

The first element of the regressor is $\varphi(1)$ that depends on the past indeces up to $k = 0$: these are the initial conditions. They are usually specified by the system being at rest; otherwise, one must drop a certain amount of entries (how many exactly?)

3.2.2 ARX: Estimate Bias

The numerical example in class (slides 9-30 to 9–34) confused me. In the first situation we have

$$\frac{B(z)}{A(z)} = \frac{bz^{-1}}{1 + az^{-1}}, \qquad H(z) = 1.$$

The one-step ahead predictor is

$$\hat{y}(k|k-1) = y(k) - e(k) = \frac{B(z)}{A(z)} u(k) = \varphi_0^\top(k)\theta_0$$

is nothing else than the evolution of the true system. The regressor Φ_0 must be constructed from the true $y_0(k) = \frac{B(z)}{A(z)}u(k)$: using instead the measured $y(k)$ gives the bias of slide 9–30.

In the second situation, we take the noise to be $H(z) = \frac{1}{A(z)}$. In Fig. 3.1 I compare the solutions of the least square problems

$$\hat{\theta}_{tn} \doteq \Phi_0 \backslash Y, \qquad \hat{\theta}_{nn} \doteq \Phi \backslash Y$$

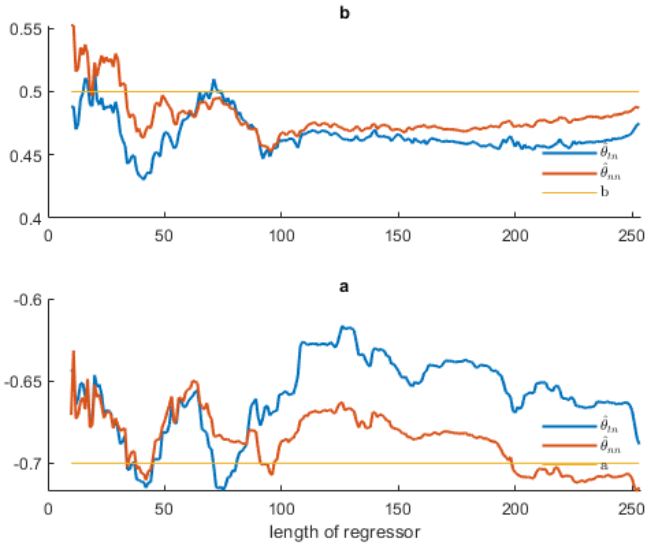

Figure 3.1: Comparison of estimates with true and noisy regressors and data for a PRBS input signal with $O = 7$ and $N = 127$. The estimates start from a minimum regressor length of 10. The code is in `09_lect/SysID_ARX.m`.

as a function of the number of elements in the regressor, just as it was done in class for the first problem. Only by looking at the graph, it is not possible to tell whether an estimate is biased, but repeating the experiment reveals that neither $\hat{\theta}_{nn}$ (as expected) nor $\hat{\theta}_{tn}$ (perhaps unexpectedly) are biased.

The explanation is the following: the (noisy) system's evolution and the parametrised predicted error are

$$\begin{aligned} y(k) &= (1 - A(z))y(k) + B(z)u(k) + e(k) = \varphi^\top(k)\theta_0 + e(k) \\ \hat{y}(k,\theta) &= \varphi^\top(k)\theta. \end{aligned}$$

Their difference, the prediction error

$$y(k) - \hat{y}(k,\theta) = \varphi^\top(k)(\theta_0 - \theta) + e(k)$$

is minimized when $\theta = \theta_0$ since $e(k)$ has zero mean: this is the case $\Phi\backslash Y$. On the other hand, we can also write

$$y(k) = y_0(k) + \frac{1}{A(z)}e(k) = \varphi_0^\top(k)\theta_0 + \frac{1}{A(z)}e(k)$$

their difference

$$y(k) - \varphi_0^\top(k)\theta = \varphi_0^\top(k)(\theta_0 - \theta) + \frac{1}{A(z)}e(k)$$

is minimized when $\theta = \theta_0$ since $\frac{1}{A(z)}e(k)$ has also zero mean. This means

$$\mathbb{E}\{\Phi \backslash Y\} = \mathbb{E}\{\Phi_0 \backslash Y\} = \theta_0.$$

As mentioned already in footnote 4, the noiseless regressor Φ_0 cannot be constructed unless (I think) $H(z) = 1$, because $y_0(k)$ evolution coincides with the $\hat{y}(k|k-1)$.

3.2.3 ARMAX Model Structure

The ARX model structure is not very flexible with regards to the noise model requiring it to have the particular structure $\frac{1}{A(z)}$. The ARMAX transfer function model partially relaxes this: it has the form

$$A(z)y(k) = B(z)u(k) + C(z)e(k) \tag{3.12}$$

where $A(z)$ and $B(z)$ are as in ARX and $C(z)$ is monic. It corresponds to the model of eq. (3.1) with

$$G(z) = \frac{B(z)}{A(z)}, \qquad H(z) = \frac{C(z)}{A(z)}$$

and with the one-step ahead predictor[7]

$$\hat{y}(k|k-1) = (1 - A(z))y(k) + B(z)u(k) + (C(z) - 1)\left(y(k) - \hat{y}(k|k-1)\right) \tag{3.13}$$

Introducing the regression vector

$$\varphi^\top(k,\theta) = \begin{bmatrix} -y(k-1) & \dots & u(k-1) & \dots & \epsilon(k-1,\theta) & \dots \end{bmatrix}$$

eq. (3.13) induces the pseudolinear regression

$$\hat{y}(k,\theta) = \varphi^\top(k,\theta)\theta \tag{3.14}$$

the equation is non-linear because the unknown appears also in the regressor $\varphi^\top(k,\theta)$.

[7] In alternative to plugging the expressions for $G(z)$ and $H^{-1}(z)$ into eq. (3.6), the one-step ahead predictor can be equally easily obtained from eq. (3.12) and eq. (3.7):

$$\begin{aligned}\hat{y}(k|k-1) &= y(k) - e(k)\\ &= (1 - A(z))\, y(k) + B(z)u(k) + (C(z) - 1)\, e(k)\\ &= (1 - A(z))\, y(k) + B(z)u(k) + (C(z) - 1)\left(y(k) - \hat{y}(k|k-1)\right).\end{aligned}$$

Consistency check: this expression reduces to ARX one-step ahead predictor eq. (3.10) when $C(z) = 1$.

3.2.4 Constrained Minimization

Using eq. (3.14) and eq. (3.8), we have that $y(k) = \varphi^\top(k,\theta)\theta + \epsilon(k,\theta)$: this is used as the constraint in the optimization-based algorithm for the solution to the ARMAX problem:

$$\begin{aligned} &\min_{\theta,\epsilon} \; ||\epsilon||_2^2 \\ &\text{subject to } Y = \Phi(\epsilon)\theta + \epsilon \end{aligned}.$$

A reference implementation is in `11_lect/SysID_ARMAX.m`.

3.2.5 General Family of Model Structures

The most general family of model structure is [2, Sect. 4]

$$A(z)y(k) = \frac{B(z)}{F(z)}u(k) + \frac{C(z)}{D(z)}e(k) \tag{3.15}$$

Some of the common cases that we have seen so far are summarized in Table 3.1.

Polynomials	Name of Model Structure
B	FIR
A B	ARX
A B C	ARMAX
A B C D	ARARMAX
B F C D	Box-Jenkins

Table 3.1: Some common black-box SISO models using the polynomials of eq. (3.15).

Chapter 4

Closed-Loop Identification

There are reasons to use system identification in closed loop:

- an unstable system must be operated in closed-loop. A simple controller stabilizes the system, but one may want to have a better model to improve the performance or because the system may change with time;
- operational constraints may require closed-loop: *e.g.* an operational industrial process cannot run in open loop for the purpose of identification because the specs on the final product must still be met;
- closed-loop controller maintains the system close to the operating point of interest (the system may be non-linear and closed loop linearizes it);
- this emphasizes plant dynamics close to the cross-over frequency range by removing a possibly large-scale zero-frequency response which is easy to control with a slow controller.

In open-loop, system identification can be performed in the frequency and time domain. In frequency domain for instance the estimate

$$\hat{G}(e^{j\omega_n}) = \frac{\hat{Y}}{\hat{U}}$$

with

$$\text{bias:} \quad \mathbb{E}\left\{\hat{G}(e^{j\omega_n}) - G(e^{j\omega_n})\right\} \longrightarrow 0as$$

$$\text{variance:} \quad \mathbb{E}\left\{|\hat{G}(e^{j\omega_n}) - G(e^{j\omega_n})|^2\right\} \longrightarrow \frac{\phi_v(e^{j\omega_n})}{\phi_u(e^{j\omega_n})}$$

as $N \to 0$. In closed-loop, the identification results may not be as good.

4.1 Methods for Closed-Loop Identification

The fundamental assumption to derive the results until now was that control input u and noise e were uncorrelated: $\phi_{ue}(e^{j\omega}) = 0$. In closed-loop this is no longer the case: the noise on the output is seen at the input through the feedback loop.

For identification, we assume

- a generalized reference $r(k) = r_2(k) + C(z)r_1(k)$;
- $y(k)$ and $u(k)$ are available;
- $C(z)$ stabilize the system and makes it internally stable: that is, all transfer functions (the Gang of Four)

$$\frac{GC}{1+GC}, \frac{G}{1+GC}, \frac{C}{1+GC}, \frac{1}{1+GC}$$

are stable.

There are four main methods for closed-loop identification:

- direct methods;
- indirect methods;
- joint input-output methods;
- dual-Youla parametrization.

4.1.1 Direct Methods

It applies the basic prediction error method Sect. in a straightforward manner: use the output y of the process and the input u in the same way as for open loop operation, ignoring any possible feedback, and not using the reference signal r. The method works regardless of the complexity of the regulator and requires no knowledge about the character of the feedback [2].

The closed loop transfer functions are

$$y = SGr + Sv$$
$$u = Sr - SCv$$

where $S(e^{j\omega})$ is the stable sensitivity function for the closed loop system

$$S(e^{j\omega}) = \frac{1}{1 + C(e^{j\omega})G_0(e^{j\omega})}.$$

Using spectral analysis[1], and assuming $\phi_{rv} = 0$, we have that

$$\hat{\phi}_{yu}(e^{j\omega}) = |S|^2 G\phi_r - |S|^2 \bar{C}\phi_v$$
$$\hat{\phi}_u(e^{j\omega}) = |S|^2 \phi_r - |S|^2 |C|^2 \phi_v.$$

The direct method estimates G directly from $\hat{\phi}_{yu}$ and $\hat{\phi}_u$:

$$\hat{G}(e^{j\omega}) = \frac{\hat{\phi}_{yu}(e^{j\omega})}{\hat{\phi}_u(e^{j\omega})} = \frac{G\phi_r - \bar{C}\phi_v}{\phi_r - |C|^2\phi_v}$$

which converges to G when the contribution from the reference signal dominates the noise.

The simplification of $|S|^2$ hides the fact that for frequencies for which $|S|^2 \sim 0$, *e.g.* when the loop transfer function $C(z)G(z)$ contains an integrator $\sim s^{-1}$, the measured signals $\hat{\phi}_{yu}$ and $\hat{\phi}_u$ are zero. On the other hand, in every practical control system with tracking, S has a bump at around the closed loop BW (is this true?): those are the frequencies that get emphasized and are relevant for the stability.

4.1.2 Indirect Methods

It identifies the closed loop transfer function[2] $T_{yr}(z)$ from reference input $r(k)$ to output $y(k)$, and retrieve from that the open loop system, making use of the knowledge of the regulator $C(z)$ [2].

Given the closed loop system

$$y(k) = T_{yr}(z)r(k) + v_{cl}(k) = \frac{G(z)}{1 + G(z)C(z)}r(k) + \frac{1}{1 + G(z)C(z)}v(k)$$

the open loop transfer function estimate $\hat{G}(z)$ is retrieved from

$$T_{yr}(z) = \frac{\hat{G}(z)}{1 + \hat{G}(z)C(z)}.$$

Only the estimate $T_{yr}(z)$ is asymptotically unbiased because the reference r is known; $\hat{G}$ (probably) does not converge to G because the transformation is non-linear which does not preserve the mean.

The advantage with the indirect method is that any idenfication method can be applied to estimate $T_{yr}(z)$. On the other hand, any error in the knowledge of $C(z)$ will be reflected in $\hat{G}(z)$.

[1] One more time, I have the impression that the result could have equally well been expressed in terms of ETFE without the need of using the correlations.

[2] In class the method was described in the frequency domain; Ljung does it in the time-domain.

4.1.3 Joint Input-Output Methods

It relies on independent measurements of y and u

$$y = SGr + Sv = T_{yr}r + Sv$$
$$u = Sr - SCv = T_{ur}r - SCv$$

so that their noise is uncorrelated, to estimate (asymptotically unbiased) $\hat{T}_{yr}(z)$ and $\hat{T}_{ur}(z)$; their noise is also uncorrelated. Since

$$\frac{T_{yr}}{T_{ur}} = \frac{SG}{S} = G$$

the estimate for $\hat{G}$ follows as the ratio

$$\hat{G}(z) = \frac{\hat{T}_{yr}(z)}{\hat{T}_{ur}(z)}.$$

As before, since the estimated spectra are weighted by S or $S(z)C(z)$ and S may become small, some frequencies may not be reliably resolved when taking the ratio. Key points:

- $\hat{G}$ may not be unbiased (unless the input signal is periodic?);
- the noise enters in a complicated manner;
- one key advantage: C does not need to be known.

This method can be seen as the specific case of a more general framework [2, Sect. 13.5] that works also for large interconnected systems, where there is no measurable reference r (*e.g.* large interconnected systems where it is also not possible to model the controller). The model is

$$y = GS(r + w) + Sv = G_{cl}r + v_1$$
$$u = S(r + w) - CSv = T_{ru}r + v_2$$

When including the correlations between $v_1 = Sv + GSw$ and $v_2 = -CSv + Sw$ gives

$$\begin{bmatrix} y \\ u \end{bmatrix} = \mathcal{G}r + \mathcal{H}v.$$

When instead the correlations are ignored gives the method described at the beginning of this section.

4.1.4 Dual-Youla Methods

It relies on coprime factorization of transfer functions

$$G(s) = \frac{N_0(s)}{D_0(s)}$$

where $N_0(s)$ and $D_0(s)$ are stable and have no common zeros.

The Bezout identity states that $N_0(s)$ and $D_0(s)$ are coprime iff there exists U and V such that

$$UN_0 + VD_0 = I.$$

A coprime factorization is "normalised" if

$$D_0^* D_0 + N_0^* N_0 = I.$$

The MATLAB command is `sncfbal`.

The Youla parametrisation is a way of parametrize all stable controllers: given a controller $C_0 = \frac{X_0}{Y_0}$ with X_0, Y_0 a coprime factorization and stable (an integral controller would not work) which stabilizes G_0, all controllers C stabilizing $G_0 = N_0/D_0$ have the form

$$C_Q = \frac{X_0 + QD_0}{Y_0 - QN_0}$$

with Q stable.

The dual Youla parametrization method takes the opposite route: given the known controller C that stabilizes the system, the plant G must be one of those that can be stabilized by C. The problem can be therefore formulated as a search on stable[3] Q: find the estimate $\hat{G}$ from the set of all plants stabilized by $C(s)$.

We model the open-loop system as

$$y(k) = \frac{N}{D}u(k) + \frac{F}{D}e(k) \rightarrow Dy = Nu + Fe \tag{4.1}$$

with F stable and stably invertible[4]. Let $C_0 = \frac{X_0}{Y_0}$ any stabiling controller: the choice of X_0, Y_0 makes a difference only from a numerical point of view. The parametrization gives

$$G_Q = \frac{N}{D} = \frac{N_0 + QY_0}{D_0 - QX_0}, \qquad H_{Q,F} = \frac{F}{D} = \frac{F}{D_0 - QX_0}$$

[3] In class we used R as the stable search transfer function, but R was used for the transient in frequency-domain and r for the closed-loop reference in time-domain. To avoid confusion, I use Q.

[4] I guess this means all zeros and poles strictly inside the unit circle.

The equivalent open loop identification experiment is obtained by rewriting eq. 4.1 as

$$(D_0 - QX_0)y = (N_0 + QY_0)u + Fe$$

or equivalently, after rearranging the terms, as

$$\begin{aligned}
\beta &\doteq D_0 y - N_0 u \\
\alpha &\doteq X_0 y + Y_0 u = X_0 \left(y + \frac{Y_0}{X_0} u \right) = X_0 r \\
\beta &= Q\alpha + Fe.
\end{aligned}$$

where the quantity $r = y + \frac{Y_0}{X_0} u$ is the reference signal r.

As it is written, this is an open-loop since there is no feedback between β and α. The procedure for the dual-Youla method is the following: given a stabilizing controller C_0

- factorise $C_0 = X_0/Y_0$;
- choose the excitation r;
- run closed-loop experiments with C_0, measuring y and u.
- choose an initial model, $G_0 = \frac{N_0}{D_0}$ (must be stabilised by C_0);
- filter the measurements, $\beta = D_0 y - N_0 u$ (time or frequency domain);
- filter the excitation $\alpha = Y_0 r$;
- estimate $\hat{Q}$ and $\hat{F}$ from $\beta = Q\alpha + Fe$;
- calculate the plant estimate $\hat{G} = (N_0 + \hat{Q}Y_0)(D_0 - \hat{Q}X_0)$.

Bibliography

[1] *Regularized System Identification: Learning Dynamic Models from Data*. Springer, 2021.

[2] *System Indentification: Theory for the User*. Second edition. Prentice Hall PTR, 2009.