KTH Electrical Engineering

System identification with input uncertainties: an EM kernel-based approach

RICCARDO SVEN RISULEO

Licentiate Thesis in Electrical Engineering
Stockholm, Sweden 2016

KTH School of Electrical Engineering
Department of Automatic Control
SE-100 44 Stockholm
SWEDEN

TRITA-EE 2016:082
ISSN 1653–5146
ISBN 978–91–7729–034–6

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan framlägges till offentlig granskning för avläggande av Teknologie licentiatexamen i elektro- och systemteknik den 14 Juni 2016 klockan 10.15 i sal E3 Kungliga Tekniska högskolan, Osquars backe 14, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Many classical problems in system identification, such as the classical prediction error method and regularized system identification, identification of Hammerstein and cascaded systems, blind system identification, as well as errors-in-variables problems and estimation with missing data, can be seen as particular instances of the general problem of the identification of systems with limited information. In this thesis, we introduce a framework for the identification of linear dynamical systems subject to inputs that are not perfectly known. We present the class of uncertain-input models—that is, linear systems subject to inputs about which only limited information is available. Using the Gaussian-process framework, we model the uncertain input as the realization of a Gaussian process. Similarly, we model the impulse response of the linear system as the realization of a Gaussian process. Using the mean and covariance functions of the Gaussian processes, we can incorporate prior information about the system in the model. Interpreting the Gaussian process models as prior distributions of the unknowns, we can find the minimum mean-square-error estimates of the input and of the impulse response of the system. These estimates depend on some parameters, called hyperparameters, that need to be estimated from the available data. Using an empirical Bayes approach, we estimate the hyperparameters from the marginal likelihood of the data. The maximization of the marginal likelihood is carried out using an iterative scheme based on the Expectation-Maximization method. Depending on the assumptions made on the models of the input and of the system, the standard E-step may not be available in closed form. In this case, the E-step is replaced with a Markov Chain Monte Carlo integration scheme based on the Gibbs sampler. After showing how to estimate the system and the hyperparameters, we show how to specialize the general uncertain-input model to particular structures and how to modify the general estimation method to account for these particular structures. In the last chapter, we show in what sense the aforementioned classical system identification problems can be seen as uncertain-input model identification problems; we show the effectiveness of the framework in dealing with these classical problems in several numerical examples.

Contents

Acknowledgments

First and foremost, I express my profound and sincere gratitude to Håkan Hjalmarsson; thanks for your guidance, passion, and critical eye. Second, my deepest thanks go to Giulio Bottegal; thanks for all the things you taught me, thanks in advance for all the things you will teach me. Third, I thank Miguel Galrinho and Marco Molinari; thanks for proofreading this thesis and adding much-needed commas here and there.

There are many people who deserve my thanks. I thank everybody at the automatic control department at KTH: professors, colleagues, administrators, officemates, friends; thanks to all of you!

I thank my parents for making my life fantastic. I thank my brothers and the *punto di sfogo* for being my best friends. I thank Laura for being a pineapple.

Last, but not least, I thank you for reading. Please, do not stop here!

Riccardo Sven Risuleo
Stockholm, May 2016

Notation

$\langle \cdot\,, \cdot \rangle_{\mathcal{V}}$	Inner product in the vector space $\mathcal{V}$
$\| \cdot \|$	Euclidean 2-norm
$\| \cdot \|_{\mathcal{V}}$	Norm in the vector space $\mathcal{V}$
S	linear system
g	impulse response of S
a_k	kth element of vector a
n_a	number of elements in vector a
$\big[A\big]_{i,j}$	the i,jth element of a matrix A
y_t	measured system output
v_t	measured system input
ε_t	output measurement noise
η_t	input measurement noise
w_t	unknown input
σ_y^2	output measurement noise variance
σ_v^2	input measurement noise variance
ω	generic parameter vector
θ	input model hyperparameter vector
ρ	system hyperparameter vector
$\mathcal{N}(x\,; \mu, \Sigma)$	Gaussian p.d.f. in x with mean μ and covariance Σ, see (1.5)
$p(y; \theta)$	probability distribution of y, with parameters θ
$\mathbf{E}\,\{\cdot\}$	expectation
$\mathbf{T}_{m\times n}(\cdot)$	$m \times n$ Toeplitz operator
$\mathbf{cov}\,\{\cdot, \cdot\}$	Covariance
$\mathbf{cov}\,\{\cdot\}$	Variance
$\mathrm{Trace}\,\{\cdot\}$	trace of a matrix
$\mathrm{lin}\{\cdot\}$	subspace spanned by vectors (or by the columns of a matrix)
$\delta(\cdot)$	Dirac delta
$\delta_{i,j}$	Kronecker delta symbol
$\hat{a}$	Estimate of a
$\bar{a}^{(j)}$	jth realization of a from a Gibbs sampler
I_n	$n \times n$ identity matrix

Statement of contributions

This thesis is based on the work carried out over the course of two and a half years as a doctoral student. It is by no means a complete and coherent story and will probably always be a work in progress.

Chapter 1 Contains an introduction to system identification and an FIR example to introduce the bias-variance trade off. It can be skipped by the reader well versed in system identification, especially if knowledgeable Bayesian techniques and Gaussian processes for FIR modeling. The chapter is loosely based on Ljung (1999), Söderström (1981) and Pillonetto and De Nicolao (2010).

Chapter 2 Contains an introduction to Reproducing Kernel Hilbert Spaces and their link to regression and especially Gaussian-process regression. It can safely be skipped by the reader that knows Bayesian inference or kernel regularization. The chapter is based on Wahba (1990), Aronszajn (1950), and Berlinet and Thomas-Agnan (2011).

Chapter 3 Contains a brief introduction to Bayesian inference, in the form of *empirical Bayes*, and to the *Expectation-Maximization method*. Both methods will be used in the main contribution. It is based on Bishop (2006) and McLachlan and Krishnan (2007).

Chapter 4 Contains the first main contribution of the thesis. We present the class of *uncertain-input models*. This is previously unpublished material.

Chapter 5 Contains the second main contribution of the thesis. We present an identification method for uncertain-input models. We use empirical Bayes arguments, combined with the expectation-maximization method, to compute the maximum marginal-likelihood estimate of the model parameters. This is previously unpublished material.

Chapter 6 Contains the third main contribution of the thesis. We show how several published methods, and some novel ones, can be seen as particular cases of identification of uncertain input models.

Section 6.1 Covers the classical prediction-error method. The material in this section is previously unpublished.

Section 6.2 Covers regularized finite impulse response models. The material in this section is previously unpublished.

Section 6.3 Covers parametric and nonparametric models for Hammerstein systems. The simulation results in this section are based on

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2015a). "A kernel-based approach to Hammerstein system identication". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 1011–1016

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2015b). "A new kernel-based approach to overparameterized Hammerstein system identification". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 115–120

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2016a). "A nonparametric kernel-based approach to Hammerstein system identification". (in preparation)

Section 6.4 Covers cascaded systems. The material in this section is previously unpublished.

Section 6.5 Covers blind system identification. The simulation results in this section are based on

Bottegal, G., Risuleo, R. S., and Hjalmarsson, H. (2015). "Blind system identification using kernel-based methods". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 466–471

Risuleo, R. S., Molinari, M., et al. (2015). "A benchmark for data-based office modeling: challenges related to CO2 dynamics". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 1256–1261

Section 6.6 Covers errors-in-variables models. The simulation results in this section are based on

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2016b). *Kernel-based system identification from noisy and incomplete input-output data. arXiv:1605.03733*

Section 6.7 Covers kernel-based system identification with missing data. The simulation results in this section are based on

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2016b). *Kernel-based system identification from noisy and incomplete input-output data. arXiv:1605.03733*

Section 6.8 Covers the estimation of initial conditions in kernel-based system identification. The simulation results in this section are based on

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2015c). "On the estimation of initial conditions in kernel-based system identification". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 1120–1125

Chapter 1

Introduction

System identification is the science of building *models* of *systems* from data. When we talk about a *system*, we are referring to an abstract mechanism that transforms *inputs* (causes) into *outputs* (effects). There is an enormous amount of systems around us. For example, a car is a system. The car has a very large number of inputs, such as the angle of the steering wheel, the positions of the brake and gas pedals, the spark timing, the settings of the air conditioning and navigation system, the orientations of the rear-view mirrors. This large number of inputs influences a very large number of outputs, such as speed, acceleration, exhaust gas temperature, fuel consumption, temperature in the passenger area. Most technological products are, in fact, *systems* in this sense; they operate on the principle of giving products—that is, *effects*, *results*—in response to commands. In addition, many nontechnological entities can be seen as systems. For instance, the human body is a collection of complex systems that respond to inputs in the form of food, heat, air, external stimuli, with outputs of various kinds.

The problem with systems is that they are inherently difficult to work with. Consider the problem of building a skyscraper. The skyscraper is the output of an exceedingly complex system, consisting of workers, materials, plans, communications, trade unions, legislation, and many other components. If you were to start from a pile of bricks, glass and steel and were to build a skyscraper, you would almost surely be in great trouble. This is where *models* come to help.

When the architect designs the skyscraper, she starts by considering some aspect of the problem at hand, disregarding many others. If she is interested in the aesthetics of the building, she will build a scale model, placing miniaturized stairways and balconies, so that the final effect can be appreciated. In this way, she can make modifications, try different layouts for the windows, maybe a different slope on the roof, without having to rebuild the whole building each time. By *simulating* the building in the model, she can reduce the complexity of the problem.

Another very important aspect of modeling is that, by pruning away aspects of reality, we can *simplify* until some aspect is easy to understand. Of course the kind of welding equipment the team is using to fix the pulley in elevator number three

is very important, but when deciding whether the building should have a rooftop garden or not it can be disregarded.

The aspects of simplification and simulation are the driving forces of modeling. When we want to study some scenario, we use a model for it. What happens when the reactor core in a nuclear power plant has a meltdown? What happens when all the polar ice disappears? Questions such as these are, usually, better answered with models.

So, a model is a simplified version of an aspect of reality that is convenient for some application. It goes without saying that different applications have different models, the engineer casting the foundation will use a different model of the skyscraper than the interior designer deciding the material of the stairs. However, they are both results of a simplification process, called *modeling*.

There are two main approaches to modeling, the *physical approach* and the *identification approach* (Ljung and Glad, 1994). In the first approach, we decompose the problem into smaller components until we arrive at components for which we already have a model. An example of this is the modeling of electronic circuits. The circuit is decomposed into its basic building blocks, resistors, transistors, capacitors, among others. Each one of these basic building blocks has a well established model. Using Kirchhoff's laws, we can assemble the building blocks to get a model that describes the circuit as a whole. In the second approach, the modeling is done from experiments. We excite the system with some inputs and observe the outputs; then, we use these data to identify the process that links inputs and outputs. In other applications, the two approaches are often used together: the physical approach is used to define a model and the identification approach is used to fit the model so that it agrees with what is observed.

1.1 The system identification loop

System identification is the process of constructing models from data. The models in system identification are *mathematical models*. Inputs and outputs are represented as variables and the relationship between them—that is, the system—is represented as an operator, in the mathematical sense, that links the inputs and the outputs. The systems we consider are *dynamical systems*—that is, systems where the output and the input evolve in time according to differential (or difference) equations.

System identification can be represented as a flow of procedures. In the *modeling* phase, we postulate a *model set* for the system. This is the set of candidate models, among which we are looking for a specific model that explains the system we are looking at. The choice of the model set is based on prior knowledge about the system. Selecting the model set is by all means the most difficult step in the procedure: it requires considerable prior knowledge about the system and its workings. It involves the choice of a type of model, and the tradeoff between model complexity and model performance. Sometimes, as was the case for the circuit in the previous section, the model set is dictated by the nature of the system. In other cases a *black-box*

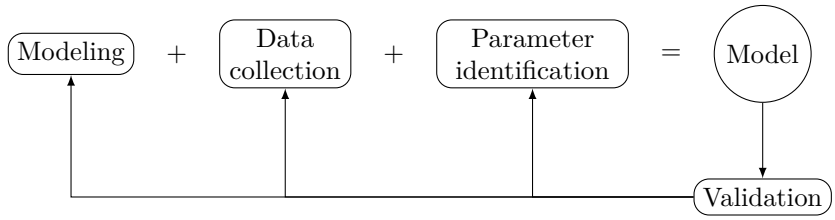

Figure 1.1: The system identification loop

approach is needed, where the model set is chosen among standard model sets, such as state-space systems or transfer functions, with no particular link to the physics of the system.

In some cases, after choosing the model set, we have the option to design an experiment that will discriminate the *best* (according to some performance criterion) model in the model set. This phase is called *experiment design*. In the experiment design, the user decides which inputs to excite and how to excite them. The user also decides which outputs are important for the specific application and which ones can be disregarded. Experiment design is the process of making all these choices so that the data collected is *the best possible*, in the sense that it contains valuable information about the system.

The experiment design is followed by a data collection phase in which we perform the designed experiment and collect data. The *data* are the measurements recorded during an identification experiment. Sometimes the experiment is dictated by the application and cannot be designed. This is the case, for instance, for systems in normal operation. In some cases it may be impossible to halt the system to perform the experiment, and we have to rely on the data from normal operation.

Once we have chosen a set of candidate models and we have collected data from the system, the problem boils down to choosing the *best* model in the set, given the current data. This phase is the *estimation* or *parameter identification* phase. In this phase, we choose a *performance metric* to discriminate among the models and an *identification method* to calculate the parameters of the model in the model set that best explains the data collected, with respect to the chosen performance metric.

Together, *modeling*, *experiment design* and *estimation* result in the formulation of a *model* of the system. In the subsequent *validation* phase, the model is compared with the true system, and its modeling performance is evaluated. If the performance is not satisfactory, we return to the modeling and identification phases, changing the model structure, the experiment or the identification method, and calculating a new model. This is repeated in a loop until we obtain a satisfactory model of the system. The whole procedure is represented in Figure 1.1. In this dissertation, we will not consider the experiment design or the model-set selection problems. We will suppose that someone has performed a good identification experiment and has given us data, in the form of a set of values of the input and of the output. We will also suppose someone has chosen the model set for us; we will use output-error models

(see Ljung, 1999, Section 4.2).

1.2 Estimating models: PEM and maximum likelihood

Consider an input signal[1] u_t. A mathematical system S is an operator, in the mathematical sense, that transforms the input signal into the output signal y_t. Given the input u_t and the output y_t, the task of system identification is to construct a model of the system, in the form of an operator $\hat{S}$, that behaves like the system, according to some pre-defined performance metric.

For instance, consider the case when we are given input and output data (u_t and y_t) collected from a system. We want to model the system using a difference equation of the type

$$y_t = g_1 u_{t-1} + g_2 u_{t-2} + \cdots + g_n u_{t-n} + \varepsilon_t. \tag{1.1}$$

This equation represents the value of the output y at time t as a function of values of the input u at the n preceding instants of time $t-1, \ldots, t-n$. The noise term ε_t represents all effects not captured by the difference equation, such as measurement noises or nonlinearities. Models such as (1.1) are often called *finite impulse response* models, or FIR models: the parameters $g_1, \ldots, g_n$ are n successive samples of the output following an impulsive input (see Figure 1.2).

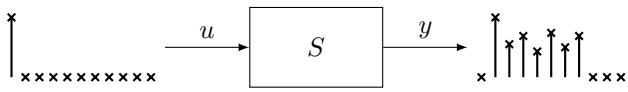

Figure 1.2: A noiseless FIR model.

The model set we are considering is the set of FIR models of order n for different values of the coefficients $g = \{g_1, \ldots, g_n\}$ (the *parameters*).

We need to define a performance metric to discriminate the best model within the model set. For every model in the model set, we can define the *predicted output*—that is, the value of the output that the model predicts using the past inputs:

$$\hat{y}_t(g) = g_1 u_{t-1} + g_2 u_{t-2} + \cdots + g_n u_{t-n}. \tag{1.2}$$

The difference between the predicted output (1.2) and the actual output y_t is called the *prediction error*, and can be used as a measure of misfit between the model and the system in the following way. We are given a set of data points $\{u_i, y_i\}_{i=1}^{N}$ (the *dataset*). For every model in the model set, we can compare the predicted output with the measured output:

$$e_t(g) = y_t - \hat{y}_t(g). \tag{1.3}$$

[1] We only consider, for notational convenience, discrete-time signals.

The prediction error is a measure of the misfit between the output predicted by the model, with parameters g, and the measured output of the true system. Averaged over the whole dataset, the prediction error can be used as a measure of the misfit between the model and the true system. This measure is called *mean square prediction error*. Minimizing the mean square prediction error, we can find an estimate of the parameters of the model:

$$\hat{g}_{\text{PEM}} = \arg\min_g \frac{1}{N} \sum_{t=1}^{N} \big(e_t(g) \big)^2. \tag{1.4}$$

This is usually called a *prediction-error method* (or PEM). A PEM is any estimation method where the parameters are found minimizing a criterion based on the prediction errors. Besides the sum of squares criterion, there are many cost functions that can be used to calculate the total prediction error. We refer the interested reader to Ljung (1999, Chapter 7) or Söderström and Stoica (1988, Chapter 7).

PEM has an interesting link to *maximum likelihood*. If we suppose that the process ε_t in (1.1) Gaussian white noise—that is, all ε_t are independent Gaussian random variables with zero mean and the same covariance σ^2— and is independent of the input signal, then we can write a model for the measurements,

$$y_t = \hat{y}_t(g) + \varepsilon_t,$$

implying that the measurements are independent Gaussian variables, with mean $\hat{y}_t(g)$ and covariance σ^2—that is, the probability density function for y_t is

$$p(y_t) = \mathcal{N}(y_t\,; \hat{y}_t(g), \sigma^2).$$

Here, the function $\mathcal{N}$ is the Gaussian probability density function

$$\mathcal{N}(x\,; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^N \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu) \right), \tag{1.5}$$

where x is a vector in $\mathbb{R}^N$. By plugging the measurements into this probability density function, we obtain the *likelihood function* of the measurements. Because the measurement errors are independent, the measurement samples are independent, and the likelihood can be written as the product of the density functions of each measurement:

$$L(g) = \prod_{t=1}^{N} \mathcal{N}(y_t; \hat{y}_t(g), \sigma^2).$$

We can choose the parameters g that maximize the log likelihood[2]:

$$\tilde{g} = \arg\max_g \log L(g).$$

[2]When working with the exponential family of distributions it is convenient to consider the natural logarithm of the likelihood.

Writing out this expression, we get

$$\tilde{g} = \arg\max_g -\frac{1}{2\sigma^2}\sum_{t=1}^{N}\left(y_t - \hat{y}_t(g)\right)^2 - \frac{N}{2}\log\sigma^2,$$

where we recognize the prediction error (1.3). This shows that the maximum likelihood estimate $\tilde{g}$ (for Gaussian measurement noise) is equal to the PEM estimate $\hat{g}_{\text{PEM}}$.

PEM with a quadratic cost function can be used for any model structure, however it becomes computationally attractive when the model is linearly parameterized. *Linearly parameterized* means that the relationship between the parameters and the predicted output is linear. For FIR models, this is indeed the case. We can collect the input samples[3] in a vector ${\phi_t}^T = [u_t \cdots u_{t-n}]$, and the parameters in a vector g, and write

$$\hat{y}_t(g) = {\phi_t}^T g.$$

Using linearly-parameterized models, PEM (consequently, maximum likelihood with Gaussian noise) becomes *linear least squares* since

$$\hat{g}_{\text{PEM}} = \arg\min_g \sum_{t=1}^{N}\left(y_t - {\phi_t}^T g\right)^2 = \arg\min_g \sum_{t=1}^{N}\left(y_t - {\phi_t}^T g\right)^2$$

has the closed form solution

$$\hat{g}_{\text{PEM}} = \left[\sum_{t=1}^{N}\phi_t{\phi_t}^T\right]^{-1}\left[\sum_{t=1}^{N}\phi_t y_t\right].$$

This solution can be written in an equivalent matrix form by collecting the measurements y_t in a vector y and the measurements u_t in a matrix Φ:

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \qquad \Phi = \begin{bmatrix} {\phi_1}^T \\ {\phi_2}^T \\ \vdots \\ {\phi_N}^T \end{bmatrix}.$$

The parameter estimate is given by the linear regression

$$\hat{g}_{\text{PEM}} = \arg\min_g \|y - \Phi g\|^2, \tag{1.6}$$

and the solution is given by of

$$\hat{g}_{\text{PEM}} = \Phi^{\dagger} y,$$

[3]For simplicity, we suppose that we know that $u_t = 0$ for $t \leq 0$.

where $\Phi^\dagger$ is the pseudoinverse of Φ (see Kailath, Sayed, and Hassibi, 2000, Lemma 2.2.2). In well designed experiments, the input is *persistently exciting* and the matrix Φ is full rank, so $\Phi^\dagger = \left(\Phi^T\Phi\right)^{-1}\Phi^T$ (see Ljung, 1999, Chapter 14 for details). If we suppose that the true system to be identified is an FIR model with parameters g^0, we can evaluate the *bias* and the *variance* of the estimate. The bias is the difference between the expected value of the estimate and the true parameters. In this case, the estimate is unbiased:

$$\mathbf{Bias}\{\hat{g}_{\text{PEM}}\} = \mathbf{E}\left\{\hat{g}_{\text{PEM}} - g^0\right\} = \mathbf{E}\left\{g^0 + \left(\Phi^T\Phi\right)^{-1}\Phi^T\varepsilon - g^0\right\} = 0.$$

where ε is the vector of measurement noises. We can also calculate the variance of the estimator:

$$\mathbf{cov}\left\{\hat{g}_{\text{PEM}}\right\} = \mathbf{E}\left\{(\hat{g}_{\text{PEM}} - g^0)(\hat{g}_{\text{PEM}} - g^0)^T\right\} = \sigma^2(\Phi^T\Phi)^{-1}$$

In the next section, we will see how we can design an estimator that has a better performance than this one, in the sense of a lower mean square error. However, this performance increase comes at a cost: we have to allow for some bias in the estimates.

1.3 Bayesian estimation and regularization

In the previous section, we introduced PEM and showed the link to maximum-likelihood estimation with Gaussian noise. In this section, we adopt a Bayesian approach. We introduce a prior distribution for the coefficients of the FIR model and calculate their posterior distribution given the observed data.

Consider the measurement model we introduced in the previous section,

$$y = \Phi g + \varepsilon,$$

where g is now a random, n dimensional, vector with a prior distribution. Suppose, for simplicity, that

$$p(g) = \mathcal{N}\left(g\,;0,\lambda I_n\right), \tag{1.7}$$

where λ is a positive scalar.

Because the noise is Gaussian and white, the data will be Gaussian distributed according to

$$p(y|g) = \mathcal{N}(y\,;\Phi g, \sigma^2 I_N).$$

We can calculate the posterior distribution of g using *Bayes' rule*:

$$p(g|y) = \frac{p(y|g)\,p(g)}{p(y)}.$$

This posterior distribution is Gaussian[4], so the *minimum mean-square-error* (MMSE, see Anderson and J. B. Moore, 2012, Section 2.3) estimate—that is, the mean of

[4]This is the *conjugate-prior* property: a Gaussian prior with a Gaussian likelihood gives a Gaussian posterior. See, for example, Bishop (2006, Section 2.4.2).

the posterior distribution—is equal to the *maximum-a-posteriori*:

$$\begin{aligned}\hat{g}_{\text{MMSE}} &= \arg\max_g p(g|y) = \arg\max_g \log p(y|g) + \log p(g) \\ &= \arg\min_g \|y - \Phi g\|^2 + \frac{\sigma^2}{\lambda}\|g\|^2.\end{aligned}$$

In the case of maximum likelihood with Gaussian noise, we obtained the least-squares criterion (1.6); in this case, we obtain a *regularized least-squares* criterion, with the quadratic regularization term coming from the Gaussian prior distribution.

The MMSE criterion admits the closed form solution

$$\hat{g}_{\text{MMSE}} = \left(\Phi^T\Phi + \frac{\sigma^2}{\lambda}I_n\right)^{-1}\Phi^T y.$$

To evaluate the performance, we will suppose that the true system to be identified is an FIR model with parameters g^0, so the data is generated according to

$$y = \Phi g^0 + \varepsilon.$$

We evaluate the performance of the MMSE criterion applied to these measurements. Averaging over the noise realizations, we see that the MMSE estimate is biased

$$\mathbf{E}\left\{\hat{g}_{\text{MMSE}}\right\} = \left(I_n + \frac{\sigma^2}{\lambda}(\Phi^T\Phi)^{-1}\right)^{-1} g^0,$$

and we can calculate the bias and the variance of this estimator:

$$\begin{aligned}\mathbf{Bias}\{\hat{g}_{\text{MMSE}}\} &= \mathbf{E}\left\{\hat{g}_{\text{MMSE}} - g^0\right\} = -\left(I_n + \frac{\lambda}{\sigma^2}\Phi^T\Phi\right)^{-1} g^0, \\ \mathbf{cov}\left\{\hat{g}_{\text{MMSE}}\right\} &= \sigma^2\left(\Phi^T\Phi + \frac{\sigma^2}{\lambda}I_n\right)^{-2}\Phi^T\Phi.\end{aligned}$$

Observing these expressions, we see that we can control the amount of bias and the amount of variance in the estimator using the parameter λ. Large values of λ will remove the bias, effectively returning the least-squares solution $\hat{g}_{\text{PEM}}$. Reducing the value of λ will increase the bias, *shrinking* the estimates toward zero. A similar reasoning can be done with the variance. Large values of λ will give large variance. Reducing the value of λ will reduce the variance because the estimates concentrate around zero.

1.4 Mean square error of the estimators

Consider any estimator $\hat{g}$. If we want to evaluate its performance, for instance to compare it with another estimator, we can use the mean-square-error metric

$$\text{MSE}\{\hat{g}\} = \mathbf{E}\left\{\|\hat{g} - g^0\|^2\right\}.$$

The mean square error is proportional to the average error made by the estimator over the realizations of the noise. We can rewrite it as:

$$\begin{aligned}\mathrm{MSE}\{\hat{g}\} &= \mathbf{E}\left\{\left\|\hat{g} - \mathbf{E}\{\hat{g}\} + \mathbf{E}\{\hat{g}\} - g^0\right\|^2\right\} \\ &= \mathbf{E}\left\{\left\|\hat{g} - \mathbf{E}\{\hat{g}\}\right\|^2\right\} + \left\|\mathbf{E}\{\hat{g} - g^0\}\right\|^2 \\ &= \mathrm{Trace}\{\mathbf{cov}\{\hat{g}\}\} + \left\|\mathbf{Bias}\{\hat{g}\}\right\|^2.\end{aligned}$$

The error made by an estimator can be decomposed into two parts, a *variance term* that accounts for the spread of the estimates, and a *bias term* that accounts for the systematic error of the estimator. In Figure 1.3 we see a schematic representation of two estimators. The true value is in black cross and the estimates, for different realizations of the noise, are in gray. The estimator on the left has high variance and low bias; the estimator on the right has a smaller variance, but high bias.

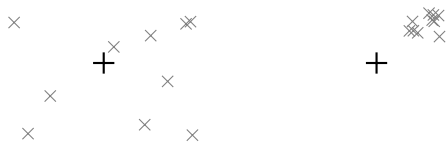

Figure 1.3: Bias and variance. The estimator on the left has high variance, the estimator on the right has high bias.

For unbiased estimators, the covariance matrix is directly linked to the mean square error:

$$\mathrm{MSE}\{\hat{g}_{\mathrm{PEM}}\} = \mathrm{Trace}\{\mathbf{cov}\{\hat{g}_{\mathrm{PEM}}\}\} = \mathrm{Trace}\left\{\sigma^2(\Phi^T\Phi)^{-1}\right\}.$$

In the case of biased estimators, the mean square error is the sum of the two components. If we return to the Bayesian estimator, we see that we can use the parameter λ to control the bias and the variance. The hope is that we are able to introduce some bias in the estimator, reducing its variance.

Figure 1.4 represents this idea of the *bias-variance tradeoff*, for a simple scalar regression example. The figure shows the mean square error of the Bayesian estimator (solid) and the PEM estimator (dashdotted) as functions of λ. The figure also shows the bias and variance components of the error for the Bayesian estimator. There are values for λ such that the Bayesian estimator has better average performance than PEM (highlighted in gray). In this simple scalar example, a small regularization (large λ) will always have a positive effect on the estimates, giving lower mean square error than PEM. However, in more complicated applications this is not always the case and, a careful choice of the regularization parameter is needed to tune the bias-variance tradeoff.

Note that the prior model (1.7) is just intended to perform a shrinkage of the parameters (a so called *ridge regression*; see Hastie et al., 2009, Section 3.4). There are many prior distributions that can be used to do system identification. These

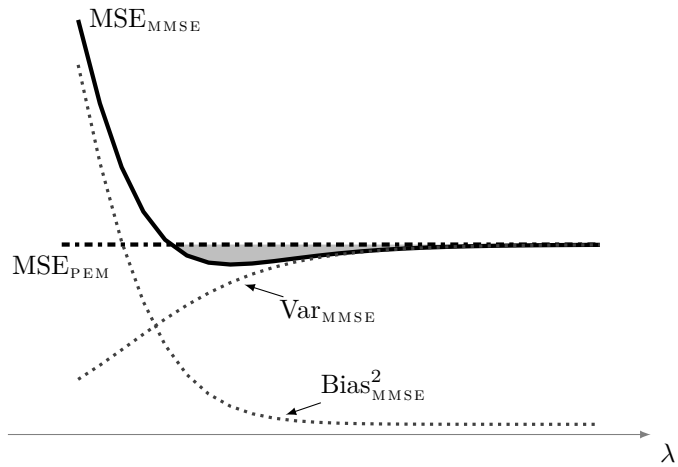

Figure 1.4: Bias-variance tradeoff. Plot of the mean square error as a function of λ. The drop in variance for small λ gives a lower mean square error, at the price of an increased bias.

priors encode more information about the system under study, such as smoothness and exponential stability (Pillonetto and De Nicolao, 2010) or resonant structures (Chen and Ljung, 2014).

1.5 Errors-in-variables models, input uncertainties and extensions

In the previous introductory sections, we saw the problem of system identification as a regression problem between inputs and outputs. We considered the case where the input samples were perfectly known, with no noise. In some applications, however, this assumption could be too restrictive because the input signal has to be measured, together with the output signal, during normal operation of the system. In the FIR case, the resulting model is called an *errors-in-variables* model (Söderström, 2007). Errors-in-variables models are more complicated to estimate than models with noiseless inputs (Söderström, 2003). In this thesis, we will consider the case where the input signal is measured, but we have some additional information about the signal itself. In this sense, the input is *uncertain*.

We refer to systems described by the equations

$$\begin{cases} y_t = \sum_{k=1}^{\infty} g_k w_{t-k} + \varepsilon_t, \\ v_t = w_t + \eta_t. \end{cases}$$

The signals y_t and v_t are measurements, corrupted by the noise processes ε_t and η_t. The objective is to identify the impulse response g and the unknown input

w_t from the measurements. This framework includes errors-in-variables models directly. Setting $w_t = f(u_t)$, where $f(\cdot)$ is an unknown nonlinear function and u_t is a known input, this framework can be used to estimate Hammerstein systems. Setting $w_t = \sum_{k=1}^{\infty} h_k u_{t-k}$, where h is an unknown impulse response and u_t is a known input, this framework can be used to estimate cascaded linear systems. In addition, this framework incorporates PEM estimation and regularized FIR estimation, as well as functional estimation problems, blind system identification, among others (see Chapter 6).

We will develop a model for the uncertain-input system that allows us to incorporate prior information about the input signal and the linear system. To do this we will use Bayesian techniques, postulating prior distributions for the input signal and the system, and using the data to estimate the parameters of the model.

1.6 Annotated bibliography

This thesis is the result of a generalization process of various contributions. In these contributions, we have addressed different—and seemingly unrelated—problems. Before going into the main part of the thesis, we give a brief survey of the different topics covered.

Expectation-maximization method

The complete treatment of the expectation-maximization (EM) method can be found in McLachlan and Krishnan (2007). The EM method was first proposed in the seminal paper Dempster, Laird, and Rubin (1977). After its introduction, EM has been applied to many different types of problems. *Iteratively reweighted least squares*, an iterative weighted least-squares method where the weight changes at each iteration, was shown in Dempster, Laird, and Rubin (1980) to be an EM method with a distributional assumption. The EM method has been used to solve regression problems with non-Gaussian noises, such as Student's t distribution in Pettitt (1985) and in Bottegal, A. Y. Aravkin, et al. (2016). In Phillips (2002), the authors study regression with Laplace noise using the EM method, and find it equivalent to Least Absolute Deviation estimation.

The proof of convergence in Dempster, Laird, and Rubin (1977) contains an error (an incorrect application of the triangle inequality). The authors do prove that the likelihood values converge, but the proof of the convergence of the parameter estimates is incorrect. Even when the parameter estimates do converge, the convergence rate, in contrast to what they state, is not necessarily linear. In Boyles (1983), the authors give an example of an EM method where the parameters do not converge to a single point but to the unit circle. In Horng (1986), there are many examples of sublinearly converging EM sequences.

The error was noticed, and corrected, in Wu (1983). After this, the convergence and invariance properties of EM have been well studied (Lansky, Casella, et al., 1992), and some ways to accelerate the sublinear convergence have been proposed

(Lansky and Casella, 1992; Jamshidian and Jennrich, 1997; C. Liu, 1998). For an overview on EM acceleration methods, see McLachlan and Krishnan (2007, Chapter 4).

One difficulty of applying the EM method is that it relies on the computation of the mean of the complete likelihood with respect to a posterior distribution. When this mean is impossible to calculate, we can use Monte Carlo EM (Neath, 2013). Another possible approach is Stochastic EM (Nielsen, 2000). Both methods are also described in Bishop (2006, Section 11.1.6) and Murphy (2012, Section 11.4.9).

Another difficulty is the need to maximize the expected complete likelihood at each iteration. As already noted in Dempster, Laird, and Rubin, 1977, and further studied in Wu, 1983, we do not need to maximize the likelihood at each iteration, but it is sufficient to find parameter updates of higher likelihood. Methods that work on this principle are called *generalized maximum likelihood* (GEM). Particular types of GEM are the *expectation/conditional maximization* (Meng and Rubin, 1993); and the *expectation/conditional maximization either* (C. Liu and Rubin, 1994). For an overview on these, we refer to McLachlan and Krishnan (2007).

Regularization and Bayesian system identification

Regularization has a long history, it was introduced in Tikhonov and Arsenin (1977) as a means to reduce the variance and improve the performance of the least-squares solution to regression problems. Before that, Stein had devised an estimator that has always better performance than least squares. This *superefficient* estimator, that later became known as the James-Stein estimator (James and Stein, 1961), can be seen as regularized estimator (see, for instance, Pillonetto, Dinuzzo, et al., 2014, Section 6). From there on, regularization has gathered a lot of interest and it has become a standard tool in applied statistics (see Hastie et al., 2009 and Bishop, 2006 for a thorough overview).

In the functional-approximation field, regularization has been used extensively to curb the ill posedness of certain estimation problems. The classic book by Grace Wahba (Wahba, 1990) on splines gives a very good survey over spline methods and how splines can be used to solve functional approximation problems. In Wahba (1990) we also see the link between splines and approximation in reproducing kernel Hilbert spaces; this relationship is extensively studied in Berlinet and Thomas-Agnan (2011), where the authors also draw the bridges to Bayesian estimation and Gaussian processes.

In the machine learning community, regularization is often used to do *kernel learning* (see Bishop, 2006, Chapter 6). Here, the *kernel trick* is used to create new methods by replacing certain inner products with kernel functions. This is the functional approximation equivalent of changing the prior distribution (it is related to Hilbert space congruence; see Berlinet and Thomas-Agnan, 2011, Theorem 34). Support vector machines have their kernel equivalent (called *relevance vector machines*, see Schölkopf and Smola, 2002). Relevance vector machines are a particular form of Gaussian process, where the kernel function has a finite expansion in terms

of basis functions. Gaussian processes are presented in all classical books on machine learning, (for a maximum-a-posteriori view, see Murphy, 2012; for a functional-learning view, see Williams and Rasmussen, 2006).

Regularization has also been used in the system-identification community. Borrowing tools from machine learning (Pillonetto, Dinuzzo, et al., 2014), kernels have been used to do nonparametric predictor impulse-response estimation. The introduction of the *stable spline* kernel in De Nicolao and Pillonetto (2008) (or *tuned/correlated kernel* in Chen, Ohlsson, and Ljung, 2012) has sprung a series of methods that incorporate Bayesian ideas into system identification; see Pillonetto and De Nicolao (2010) and Pillonetto, Chiuso, and De Nicolao (2011). As the main prerogative of the kernel is to define a suitable Hilbert space of candidate estimates, developing good kernels is very important. See Dinuzzo (2015) for an overview of kernels for system identification. In Chen and Ljung (2014), the authors propose a constructive way of creating kernels that encode prior knowledge about the systems under study. Recently, the use of multiple kernels has been proposed (see Chen, Ljung, et al., 2012, Chen, Andersen, Ljung, et al., 2014, and Chen, Andersen, Chiuso, et al., 2014), and nonparametric impulse-response estimation has been used to do subspace identification (see Chiuso, Pillonetto, and De Nicolao, 2008). In Bottegal, Hjalmarsson, et al. (2015), outlier robust identification is studied using non-Gaussian noise distributions.

One key idea of system-identification kernels is that they have to encode prior information on the exponential stability of the system and on the smoothness of the impulse response. These features have made stable spline kernels suitable for other estimation problems, such as the reconstruction of exponential decays (Pillonetto, Chiuso, and De Nicolao, 2010) and correlation functions (Bottegal and Pillonetto, 2013).

The main advantage of regularization methods is that they circumvent the need for a model-order selection step. The tuning of the hyperparameters can be seen as a way to reduce the complexity of the estimates, much like classic model-order selection criteria such as AIC or BIC (Pillonetto and De Nicolao, 2011). Traditionally, hyperparameters were tuned using cross-validation. This is the way advocated in Wahba (1990), where the method is studied in depth. One strength of the Bayesian interpretation is that it enables hyperparameter tuning using the *maximum marginal likelihood* method (see A. Aravkin et al., 2012). In the marginal-likelihood method, the prior parameters are estimated from the marginal distribution of the data. The estimated hyperparameters are then used to compute the posterior mean of the unknowns. This method is also known as *empirical Bayes* (Maritz and Lwin, 1989), *type 2 maximum likelihood* (Berger, 1993), *generalized maxium likelihood* (Wahba, 1990), *evidence approximation* (Bishop, 2006). The robustness properties of empirical Bayes and the marginal-likelihood estimator for system identification applications have been studied in Pillonetto and Chiuso (2014) and Pillonetto and Chiuso (2015).

Hammerstein system identification

For a survey over classical nonlinear black-box system-identification methods, see Sjöberg et al. (1995). Many nonlinear systems can be approximated using block-oriented nonlinear models (see Giri and Bai, 2010). In particular, the Hammerstein model is a nonlinear cascaded model where a linear time-invariant (LTI) dynamical model is fed by a signal that is the result of a static nonlinear transformation of the input signal (Ljung, 1999). Hammerstein models are very flexible, and are able to model a range of phenomena (see, for example, Hunter and Korenberg, 1986; Westwick and Kearney, 2001; Bai, Cai, et al., 2009).

In Bai (1998), the overparameterization method was introduced. With over-parametrization, the identification problem is embedded in the larger problem of identifying a vector containing all the cross products of the parameters. In this way, the problem becomes linear and can be solved with least squares (Bai, 1998), or instrumental variables (Han and De Callafon, 2011), among others. To recover the parameters from the overparameterized vector, we need a reduction step (for instance, minimum norm in Bai, 1998, and in Han and De Callafon, 2011; consistent estimation in Boutayeb, Aubry, and Darouach, 1996; regularization in Risuleo, Bottegal, and Hjalmarsson, 2015b, and in Falck et al., 2010).

Besides overparametrization, Hammerstein models can be identified using subspace methods. In Verhaegen and Westwick (1996), MOESP is used by assuming a polynomial model for the static nonlinearity. In Goethals et al. (2005), the authors use support vector machines to adapt N4SID to Hammerstein model identification.

Separable least squares can be used to reduce the number of parameters to be estimated (see Golub and Pereyra, 2003; Westwick and Kearney, 2001; Han and De Callafon, 2012). In similar approaches, the problem is solved by alternating the estimation between two sets of variables. These methods are often called iterative methods; see, for instance, Bai and Li (2004), and Y. Liu and Bai (2007).

In Billings and Fakhouri (1978), Greblicki (2000), and Rangan, Wolodkin, and Poolla (1995), the linear component of the Hammerstein model is identified, irrespective of the nonlinearity, using correlation analysis. Also approaches based on the *best linear approximation* (see Pintelon and Schoukens, 2012, Section 3.4) or on blind identification (see Bai and Fu, 2002 and Vanbeylen, Pintelon, and Schoukens, 2009) can be used.

Frequency domain methods use the frequency content of the input and output signals. By applying various sinusoidal inputs, the model parameters can be identified (see, for instance, Baumgartner and Rugh, 1975). Using sinusoidal inputs, the harmonics of the output signal can be used to derive information about the nonlinear transformation (see, for instance, Schoukens, Dobrowiecki, and Pintelon, 1998; Schoukens, Pintelon, Dobrowiecki, et al., 2005; Pintelon and Schoukens, 2012).

Various models for the static nonlinearity have been used. For instance, the nonlinearity can be described with kernels or orthogonal series (see, for instance, Greblicki and Pawlak, 1986; Greblicki, 1989; Mzyk, 2007).

A Bayesian kernel-based method has been used, paired with the stable spline

kernel, in Risuleo, Bottegal, and Hjalmarsson (2015a). A maximum-likelihood method has been proposed in Wills et al. (2013). In Pillonetto, Quang, and Chiuso (2011), the authors propose a nonparametric Bayesian method for nonlinear models, without postulating any specific structure. Likewise, in Pillonetto and Chiuso (2009b), the authors propose a nonparametric method for the Wiener-Hammerstein model structure.

Errors-in-variables models

The first method for the identification of static errors-in-variables models was proposed in Frisch, 1934. This method, called the *Frisch scheme*, gives not one but a family of models that explain the observed data. When applied to dynamical systems, the Frisch scheme gives unique estimates (Beghelli, Guidorzi, and Soverini, 1990). For an in-depth study of the Frisch scheme, see Söderström, Soverini, and Mahata (2002). The method has later been extended to more general noise models (Fan and Luo, 2010; Ning et al., 2015; Zhang and Pintelon, 2015)

Other methods for errors-in-variables identification rely on bias compensation (for instance, Zheng and Feng, 1989) to correct the bias of least-squares. For dynamical systems, both time-domain (Söderström, 1981; Diversi, Guidorzi, and Soverini, 2007) and frequency-domain (Schoukens, Pintelon, Vandersteen, et al., 1997; Zhang, Pintelon, and Schoukens, 2013; Zhang and Pintelon, 2015) maximum-likelihood approaches have been proposed; for a survey and a comparison of the maximum-likelihood methods see Söderström, Hong, et al. (2010).

Missing data and blind system identification

The first example of identification of dynamical systems with missing data can be found in Isaksson, 1993. There, the author uses a Kalman filter for the reconstruction of the missing samples and then estimates the system using the EM method. System identification with missing data has been studied both in the frequency domain (Pintelon and Schoukens, 1999; Pintelon and Schoukens, 2000) and in the time domain (Wallin, Isaksson, and Ljung, 2000; Wallin and Hansson, 2014). Regularization techniques can also be used to identify systems with missing data; for instance, nuclear-norm approaches are used in Z. Liu, Hansson, and Vandenberghe (2013) and Markovsky and Usevich (2013), and kernel approaches are used in Pillonetto and Chiuso (2009a) and Risuleo, Bottegal, and Hjalmarsson (2016b).

In the case where all input measurements are missing, we cannot apply standard system identification tools but we need to use *blind system identification* methods (Abed-Meraim, Qiu, and Hua, 1997).

Blind system identification finds applications in a wide range of engineering areas, such as image reconstruction (Ayers and Dainty, 1988; Nakajima, 1993), biomedical sciences (McCombie, Reisner, and Asada, 2005), occupancy estimation (Ebadat, Bottegal, Varagnolo, et al., 2015; Bottegal, Risuleo, and Hjalmarsson, 2015) and communications (Gustafsson and Wahlberg, 1995; Moulines et al., 1995).

The unavailability of the input signal makes blind system-identification problems ill posed. Without further information on the input sequence or the structure of the system, it is impossible to retrieve a unique description of the system (Tong et al., 1991). To circumvent the nonuniqueness some assumptions on the input signal are needed (see, for instance, Ohlsson et al., 2014 and Ahmed, Recht, and Romberg, 2014). Recently, Bayesian methods for semiblind deconvolution have been proposed (see Pillonetto and Bell, 2007, and references therein).

Chapter 2

Regression in Reproducing Kernel Hilbert Spaces and Gaussian Processes

In this chapter, we lay the mathematical ground for the thesis. We study the concept of Reproducing Kernel Hilbert Space (RKHS) and its relation to Gaussian Processes. Then, we show how inference—in the sense of estimation of functions—can be recast into a regression problem and we arrive at the concept of *Gaussian regression*.

2.1 Introduction

The reproducing kernel K, also called *positive-definite kernel*, is a function of two variables defined on an abstract set E

$$K : E \times E \longmapsto \mathbb{R},$$
$$K(x_1, x_2) \longmapsto y,$$

that has the property

$$\sum_{i,j=1}^{n} K(x_i, x_j) a_i a_j \geq 0, \qquad \text{for any } x_i \in E,\ a_i \in \mathbb{R}.$$

Starting from this simple definition, a complex and fascinating theory has been developed where the kernels induce classes of functions.

The theory of reproducing kernels is the result of the confluence of different mathematical approaches. The theory was formulated in its first complete form by Aronszajn in the 1940s (Aronszajn, 1950). Before arriving at the general formulation by Aronszajn, different authors had analyzed what we today know are kernels in specific contexts (for instance in integral equations, partial differential equations, and harmonic analysis).

Two major trends can be identified: in the first, the focus is on application of the kernel K itself, possibly using the class of functions F it induces as tools for the

analysis; in the second, the focus is on a class of functions F, and the corresponding kernel K is used as a tool to study the functions in the class.

The first trend stems form the theory of integral equations as introduced by Hilbert and further developed by Mercer (Mercer, 1909). Mercer studied real-valued kernels; however, his theory can be extended to complex-valued kernels. Moore considered complex-valued kernels on abstract sets, calling them *positive Hermitian matrices* (E. H. Moore, 1916), and discovered the property that links kernel functions to Hilbert spaces. Moore proved that to every Hermitian matrix K there corresponds a class of functions forming a Hilbert space $\mathcal{H}$ in which the kernel has the *reproducing property*[1]

$$f(x) = \big\langle f(\cdot), K(\cdot\,, x)\big\rangle_{\mathcal{H}}\,.$$

This result was extended by Aronszajn into what is known as the *Moore-Aronszajn theorem*, which explains the link between RKHS and Gaussian-process regression.

The second trend was initiated by Zaremba (Zaremba, 1908). He was the first to introduce a reproducing kernel corresponding to a well defined class of functions. In subsequent years, many authors have developed kernels corresponding to particular classes of functions, such as harmonic functions and analytic functions. These kernels were used to solve boundary value problems for elliptic partial differential equations.

In 1943, Aronszajn unified the previous approaches into what is currently referred to as *theory of reproducing kernels*. The whole theory is built around the reproducing property. Aronszajn started from the definition of kernel as a function in a Hilbert space that has the reproducing property. He then proceeded unifying the two trends by noticing the equivalence of the concepts of positive hermitian matrix and of reproducing kernel.

The link to estimation and learning can be traced back to the 1960s and to the works of Schoenberg in the field of spline estimation (Schoenberg, 1969). He considered the problem of finding the function f in the Sobolev space W^m that solves the variational problem

$$\begin{aligned} \underset{f\in W^m}{\text{minimize}} \quad & \int_a^b \left(f^{(m)}(x)\right)^2 dx \\ \text{subject to} \quad & f(x_i) = y_i,\ i = 1 \dots n. \end{aligned}$$

He found that the solution is a piecewise polynomial curve, polynomial in every interval $[x_i,\, x_{i+1}]$, with $2m-2$ continuous derivatives at the knots. He called this solution the *polynomial spline*[2]. The spline is, in words, the curve of minimum generalized curvature (if $m = 2$ we recover the usual notion of curvature) that interpolates the points $(x_i,\, y_i)$.

[1] $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ is the inner product in the Hilbert space $\mathcal{H}$, see Section 2.2

[2] a *spline*, or in modern terms a *flexible curve*, is a tool used for drawing curves. It consists in a long flexible rod of wood, plastic or metal. To draw a curve, certain points are fixed by using pins or weights (referred to as *ducks*). Between the ducks the spline will assume the configuration of minimum bending energy. It was a very common tool is shipbuilding, where it was used to design ship hulls.

In statistics, the main problem is usually not interpolation but smoothing, as the measurements are often corrupted by noise. In this case, we can consider the smoothing problem

$$\underset{f \in W^m}{\text{minimize}} \quad \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int_a^b \left(f^{(m)}(x)\right)^2 dx, \tag{2.1}$$

which represents the search for the function f (in the same Sobolev space) that has the best tradeoff (controlled by the parameter λ) between adherence to data and smoothness. Schoenberg found that the solution to this problem is again a natural polynomial spline, this time not passing exactly through the noisy points (x_i, y_i).

The generalizations of this problem is the core of the work by Wahba and Kimeldorf (Kimeldorf and Wahba, 1970; Wahba, 1990). They consider observational functionals that are generic bounded linear functionals, and smoothness functionals that are mth order differential operators satisfying certain conditions. Solving these variational problems had a very large impact on applications, especially as more computational power became available. In the words of Wahba: "Today it is hard to open an issue of the *Journal of the American Statistical Association,* the *Annals of Statistics,* or the *Journal of the Royal Statistical Society* without finding the word 'spline' somewhere" (Wahba, 1990, page xiv).

The variational problems as expressed in the splines literature can be seen as optimization problems in suitable RKHS, and much of the theory of splines can be derived from the reproducing property of the kernel associated to them.

In parallel to the spline interpretation, Parzen used RKHS to unify the problems of detection and prediction of signals in noise, effectively creating the link between stochastic processes and functions in RKHS (Parzen, 1970). This link opened the whole field of Gaussian regression, where optimization in functional spaces is replaced by probabilistic computations. Parzen showed how regression analysis, in the form of least-squares estimation or minimum-variance unbiased linear estimation, can be recast into a search for a function in a suitable RKHS where the reproducing kernel is the covariance function of the noise process. Parzen's main result is the equivalence theorem of time series and functions in RKHS. It states that if a time series has a covariance function $K(s,t)$, then $K(\cdot\,,\cdot)$ is the kernel of a RKHS. Furthermore, the function $K(\cdot\,,t)$ is a representation of the time series, in the sense that there is congruence transformation between the times series and the RKHS.

Using this theorem, we can do estimation and regression on time series by applying Hilbert space tools, effectively recovering least squares and minimum-variance unbiased linear estimates from the *projection theorem* in Hilbert spaces.

2.2 Norm, inner product and Hilbert space

When we talk about a Hilbert space we are essentially talking about a particular vector space. A vector space (over a field F) is a set $\mathcal{V}$ of objects, called *vectors*, together with an operation of *addition* (which associates with any two vectors v_1

and v_2 in $\mathcal{V}$ a vector $v_1 + v_2 \in \mathcal{V}$) and an operation of *scalar multiplication* (which associates with any vector $v \in \mathcal{V}$ and any scalar $\lambda \in F$ a vector λv in $\mathcal{V}$). For the space to qualify as a vector space, these operations need to satisfy certain axioms (commutative and associative laws for vector addition, existence of additive null element, distributive and associative laws for scalar multiplication, existence of identity element for scalar multiplication; for a complete treatment, see Luenberger, 1997, Chapter 2).

In order to define topological concepts such as openness, closure and convergence, we need a measure of distance in the vector space. This measure is a real-valued function that maps each $x \in \mathcal{V}$ into a real number $\|v\|_{\mathcal{V}}$, called the *norm* of v. The norm satisfies the axioms of homogeneity, point separation and the triangle inequality (see Luenberger, 1997 for details). A vector space with a norm is called a *normed vector space*.

A sequence $\{x_n\}$ in a normed vector space is a *Cauchy sequence* if $\|x_n - x_m\| \to 0$ when $n, m \to \infty$. All convergent sequences are Cauchy; however, in general, not all Cauchy sequences converge. A normed vector space in which all Cauchy sequences converge to elements in the vector space itself is a *Banach space*.

Together with the concept of distance, another very useful geometrical concept is that of *angle* between vectors. Particularly important is the concept of orthogonality. To define these, we need to introduce the *inner product* between two vectors. In a vector space $\mathcal{V}$, the inner product is a function defined on $\mathcal{V} \times \mathcal{V}$. It associates to each pair of vectors v_1 and v_2 in $\mathcal{V}$ the scalar $\langle v_1, v_2 \rangle_{\mathcal{V}}$. The inner product is conjugate symmetric, linear in the first argument, and positive definite[3]. A vector space with an inner product is a *pre–Hilbert space*. Pre–Hilbert spaces are normed spaces where the norm is induced by the inner product, in the sense that if $v \in \mathcal{V}$, $\|v\|_{\mathcal{V}}^2 = \langle v, v \rangle_{\mathcal{V}}$. A complete pre-Hilbert space is a *Hilbert space*. What makes Hilbert spaces very useful is that the geometric intuition we have from ordinary Euclidean space can, with some care[4], be used to treat very abstract objects; particularly important is the concept of *orthogonality*: two vectors v_1 and v_2 in a Hilbert space $\mathcal{H}$ are said to be orthogonal if[5]

$$\left\langle v_1, v_2 \right\rangle = 0.$$

As in all vector spaces, Hilbert spaces contain sets of linearly independent vectors. A set of vectors is linearly independent if no vector in the set can be written as a linear combination of the others. The n vectors v_i are linearly independent if the relationship

$$\sum_{i=1}^{n} \alpha_i v_i = 0$$

[3] $\langle v, v \rangle \geq 0$ for all v, and $\langle v, v \rangle = 0$ if and only if $v = 0$.

[4] Infinite dimensional spaces can be tricky: for instance, in Hilbert spaces we can create sequences of elements in a subspace that converge to elements outside of the subspace. This is not possible in Euclidean space.

[5] We drop the explicit reference to the Hilbert space $\mathcal{H}$ in the inner product notation, when no confusion is possible.

is only satisfied when all α_i are equal to 0. Given a set of vectors, we can construct all the vectors that are linear combinations. The resulting set of vectors is a *subspace*:

$$\mathcal{M} = \text{lin}\{v_1, \ldots, v_n\}.$$

If the vectors $v_1, \ldots, v_n$ are linearly independent, they form a *basis* of the subspace. A family of vectors e_i in $\mathcal{H}$ is called an *orthogonal system* if all the vectors are orthogonal (if, in addition, $\|e_i\| = 1$, they form an *orthonormal system*).

Orthonormal systems are very important because they extend the concept of coordinates: if $e_1, \ldots, e_n$ is an orthonormal system, then for any vector $v \in \text{lin}\{e_1, \ldots, e_n\}$ we have

$$v = \sum_{i=1}^{n} \langle v, e_i \rangle e_i.$$

The coefficients $\langle v, e_i \rangle$ of this expansion are called the *Fourier coefficients*.

Given a subspace $\mathcal{M}$ and a vector x in $\mathcal{H}$, we can define the *linear variety* $V = x + \mathcal{M}$ as the set of vectors obtained by translating the subspace $\mathcal{M}$ to the point x:

$$V = \{v \in \mathcal{H} \text{ such that } v = x + m \text{ for some } m \in \mathcal{M}\}.$$

One of the key results in Hilbert spaces is the *projection theorem*. It is the generalization of the intuitive result that the shortest path from a point to a plane is the line perpendicular to the plane.

Theorem 2.2.1 (Projection theorem)**.** Let $\mathcal{H}$ be a Hilbert space, and $\mathcal{M} \subset \mathcal{H}$ a finite dimensional subspace. For any point $x \in \mathcal{H}$ there is an unique $m_0 \in \mathcal{M}$ such that $\|x - m_0\| < \|x - m\|$ for all $m \in \mathcal{M}$. m_0 is the unique solution of $\min_{m \in \mathcal{M}} \|x - m\|$ if and only if $x - m_0$ is orthogonal to $\mathcal{M}$.

Corollary 2.2.2. If $e_1, \ldots, e_n$ is an orthonormal system in $\mathcal{M}$, then

$$m_0 = \sum_{i=1}^{n} \langle x, e_i \rangle e_i.$$

The projection theorem tells us that the straight line is the shortest path from a point to a plane (in fact to any convex set); in addition, it tells us that among all points on the plane, there is one that is closest to any given point in the space. The corollary provides a convenient way to calculate the closest point.

When looking into approximation and estimation problems, we will use a variation of the projection theorem, called the *dual formulation of the projection theorem*, that deals with linear varieties and minimum norms.

Theorem 2.2.3 (Dual formulation of the projection theorem)**.** Let $\mathcal{H}$ be a Hilbert space and $\mathcal{M} \subset \mathcal{H}$ a finite dimensional subspace. Let x be a fixed point in $\mathcal{H}$ and let $V = x + \mathcal{M}$ be a linear variety. Then, there is a unique $x_0 \in V$ of minimum norm, and x_0 is orthogonal to $\mathcal{M}$.

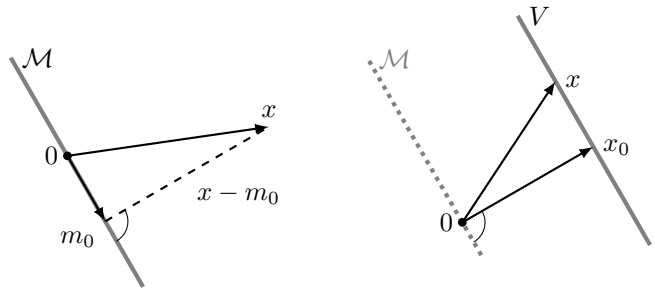

Figure 2.1: Two-dimensional representation of the projection theorem (left) and of its dual formulation (right)

In Figure 2.1, we see the projection theorem and its dual formulation in the two-dimensional Euclidean space. The projection theorem will be very important when we look at the problems of interpolation and approximation. Before doing that, we introduce a particular family of Hilbert spaces called *reproducing Kernel Hilbert spaces*.

2.3 Reproducing Kernel Hilbert Spaces

To arrive at the concept of reproducing kernel Hilbert spaces (RKHS), we will follow a constructive approach. In this way, we will see how the introduction of the reproducing kernel $K(\cdot\,,\cdot)$ can be seen as a natural consequence of considerations of practical usefulness.

Consider the following problem: *given n point observations of an unknown function, reconstruct the underlying function.* We call this the *interpolation problem*. As we can readily see, the general interpolation problem is ill-posed, because there are infinitely many functions passing through a finite number of points. To solve the problem, we need to restrict the candidate functions in some way.

We want to reconstruct an unknown function f from a domain X to a codomain Y. In other words, we want to construct another function $\hat{f}$ that is, in some way, similar to f. The abstract concept of similarity we refer to is naturally translated into the concept of *closeness* in space. Two functions are similar if they are geometrically close to one another. This leads us to consider functions that belong to some Hilbert space $\mathcal{H}$, where the concept of distance is well defined:

$$f, \hat{f} \in \mathcal{H}.$$

If we are looking for an unknown function by observing some evaluations of it, then the evaluations need to give sufficient information to discriminate the function. This means that, if two functions are similar, so should their values be; in other words, similar functions have similar values. We introduce the *evaluation functionals*

e_x defined as

$$\begin{aligned} e_x : \mathcal{H} &\longmapsto Y, \qquad x \in X \\ f &\longmapsto f(x), \end{aligned} \tag{2.2}$$

which associate to each function f in $\mathcal{H}$ the value $f(x)$ of the function f at x. If close functions have close values, then the evaluation functional maps points close in $\mathcal{H}$ into points close in Y; in other words, *the evaluation functionals are continuous.* This is really the only ingredient needed for a RKHS. The evaluation functionals are linear functionals on $\mathcal{H}$, as

$$e_x(f_1 + f_2) = f_1(x) + f_2(x) = e_x f_1 + e_x f_2.$$

We now use the *Riesz representation theorem*, which states that all continuous linear functionals operating on an $f \in \mathcal{H}$ can be written as an inner product between f and a unique element $k \in \mathcal{H}$, called the *representer* of the functional. So, each evaluation functional e_x is represented, in $\mathcal{H}$, by a function $k_x(\cdot)$ such that

$$e_x f = \langle f, k_x \rangle.$$

Because $k_x(\cdot)$ is in $\mathcal{H}$ for every $x \in X$ and is a function of X, we can define the unique function

$$\begin{aligned} K : X \times X &\longmapsto Y \\ K(x_1, x_2) &\longmapsto k_{x_2}(x_1). \end{aligned}$$

We call this function the *reproducing kernel* of the Hilbert space $\mathcal{H}$. From this definition, it is immediate to verify the *reproducing property*

$$f(x) = \big\langle f(\cdot), K(\cdot\,, x) \big\rangle. \tag{2.3}$$

Hilbert spaces that have a reproducing kernel are RKHS. From the arguments above, we see that the continuity of the evaluation functionals e_x is equivalent to the existence of a reproducing kernel. Furthermore, all RKHS have continuous evaluation functionals.

From the practical need to estimate a function from its values, we have decided that the space to look for candidates is a Hilbert space with continuous evaluation functionals and concluded that the natural space in which to look for candidates is a RKHS. We have not yet discussed how to choose the specific space, nor how the properties of the space define the candidate functions; for now, we just suppose that we have a RKHS to work with.

2.3.1 The interpolation problem

Let us return to the interpolation problem as we sketched it in the beginning of the previous section. We are looking for a function f from X to Y, in a RKHS $\mathcal{H}$, that

interpolates a given set of N observations (x_i, y_i). This problem is ill posed: in an infinite dimensional function space, there is an infinite amount of functions that pass through a finite set of points.

Among all the possible functions, it makes sense to choose the smallest one (in the sense of minimum norm). In other words, we are looking to solve the optimization problem

$$\begin{aligned} &\underset{f\in\mathcal{H}}{\text{minimize}} && \|f\|^2, \\ &\text{subject to} && f(x_i) = y_i, \quad i = 1, \dots, N. \end{aligned}$$

Using the reproducing property (2.3), we can rewrite this as

$$\begin{aligned} &\underset{f\in\mathcal{H}}{\text{minimize}} && \big\langle f, f \big\rangle, \\ &\text{subject to} && \big\langle f(\cdot), K(\cdot\,, x_i) \big\rangle = y_i, \quad i = 1, \dots, N. \end{aligned} \tag{2.4}$$

The constraints in (2.4) trace out a closed hyperplane (a linear variety) in the space $\mathcal{H}$. To see this, call $\mathcal{M} = \operatorname{lin}\{K(\cdot\,, x_i)\}$ the linear span of the *kernel slices* $K(\cdot\,, x_i)$. Consider the functions f such that

$$\big\langle f(\cdot), K(\cdot\,, x_i) \big\rangle = 0.$$

These functions f belong to a space that is orthogonal to the space spanned by the $K(\cdot\,, x_i)$. If we call this space $\mathcal{M}^\perp$, the constraints define a translation of $\mathcal{M}^\perp$—that is, a linear variety. We are thus looking for the minimum norm vector that belongs to a linear variety. This is an application of the dual formulation of the projection theorem (Theorem 2.2.3); we can directly say that the solution $\hat{f}$ is orthogonal to the closed subspace $\mathcal{M}^\perp$, so it is in $\mathcal{M}$, and the solution $\hat{f}$ is a combination of the kernel slices:

$$\hat{f}(\cdot) = \sum_{j=1}^{n} \alpha_j K(\cdot\,, x_j). \tag{2.5}$$

The coefficients α_i are such that the constraints are satisfied—that is, $\hat{f}$ is on the linear variety described by the constraints in (2.4). Plugging $\hat{f}$ into the constraints, we obtain

$$\sum_{j=1}^{n} \alpha_j K(x_i, x_j) = y_i, \qquad i = 1, \dots, n, \tag{2.6}$$

and the solution of this linear system of equations gives the α_i that define the solution to the interpolation problem. Collecting the α_i and y_i into vectors, and $K(x_i, x_j)$ into a matrix, we write

$$\alpha = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad K = \begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_n) \\ \vdots & & \vdots \\ K(x_n, x_1) & \cdots & K(x_n, x_n) \end{bmatrix}.$$

Supposing, for simplicity, that the matrix K is invertible, we can write the solution as $\alpha = K^{-1}y$.

What makes working with RKHS very convenient is that, once we have chosen a function family (with a reproducing kernel $K(\cdot\,,\cdot)$) with desired properties, we can forget everything about Hilbert spaces and projection theorems and jump directly to (2.6) and (2.5) to find the minimum norm function in the chosen family that passes trough the points (x_i, y_i).

2.3.2 A smoothing problem

Consider now the case in which our observations of the unknown function are corrupted by measurement noise. In this case, we do not want our estimated function to interpolate the measurements, as it would also interpolate the noise (see Figure 2.2).

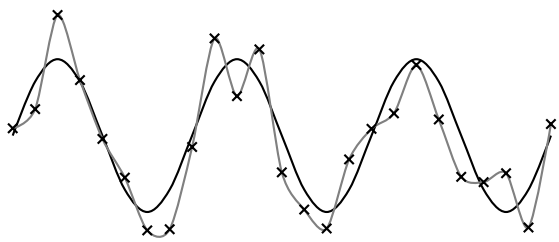

Figure 2.2: A smoothing problem. The interpolating solution (gray) is *overfitting* as it is also describing noise. The smoothing solution (black) filters away the noise, giving a more reasonable estimate.

We need to find a way to tradeoff the fit to the data with the smoothness. We can search for the candidate function in a RKHS where the norm encodes some notion of smoothness, so that the smaller the norm the smoother the function. Using this space, we can see the smoothing problem as the problem of finding function $\hat{f}$ that solves the optimization problem

$$\underset{f\in\mathcal{H}}{\text{minimize}} \quad \sum_{i=1}^{n} \big(f(x_i) - y_i\big)^2 + \gamma\|f\|_{\mathcal{H}}^2.$$

With a reasoning similar to the one we made in the interpolation case, we can say that the solution of the smoothing problem is a function $\hat{f}$ that is a linear combination of kernel slices

$$\hat{f}(\cdot) = \sum_{j=1}^{n} \alpha_j K(\cdot\,, x_j),$$

where the coefficients α_i are the new unknowns. To find their values, we plug this expression into the optimization problem. Using the reproducing property of the

kernel, we find

$$\underset{\alpha_1,\ldots,\alpha}{\text{minimize}} \quad \sum_{i=1}^{n}\left(\sum_{j=1}^{n}\alpha_i K(x_i,x_j)-y_i\right)^2+\gamma\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j K(x_i,x_j).$$

Collecting α_i, y_i, and $K(x_i,x_j)$ as we did in the interpolation case, we can write this as the quadratic problem

$$\underset{\alpha}{\text{minimize}}\|\alpha^T K-y\|^2+\gamma\alpha^T K\alpha$$

and find the solution in closed form:

$$\alpha=(K+\gamma I)^{-1}y.$$

In this expression, we see in which way the parameter γ represents a tradeoff between goodness of fit and smoothness. If we set $\gamma=0$, we recover the interpolating solution, which does not enforce smoothness. If we set $\gamma\to\infty$, we obtain the smoothest solution possible: $\hat{f}\equiv 0$.

These simple calculations show again that, once we have fixed a RKHS and we have some data, we can jump directly to the solution. Note that, in the derivation of the smoothed solution, we have not used any model for the noise and we have assumed that we observe the values of the unknown function directly. However, often we have probabilistic measurement models where we know the statistics of the noise. In the next section, we will see how the RKHS framework can be given a probabilistic interpretation and how we can use this interpretation in estimation problems.

2.4 Kernels and Gaussian regression

To make the shift into the probabilistic framework of *Gaussian processes*, we will start from the definition of positive function:

Definition 2.4.1. A real function K on $F\times F$ is a *positive function* if

$$\sum_{i=1}^{n}\sum_{j=1}^{n}a_i a_j K(x_i,x_j)\geq 0.$$

This definition is important because all reproducing kernels are positive functions, and all positive functions are reproducing kernels; this result is commonly referred to as the *Moore-Aronszajn theorem*:

Theorem 2.4.2 (Aronszajn, 1950)**.** Let K be a positive function on $F\times F$. Then there exists one RKHS $\mathcal{H}$ with K as reproducing kernel. If we denote by $\mathcal{H}_0$ the

subspace spanned by $K(\cdot\,, x)$ with $x \in F$, then $\mathcal{H}_0$ is dense in $\mathcal{H}$ and all Cauchy sequences in $\mathcal{H}_0$ converge pointwise to functions in $\mathcal{H}$, with inner product

$$\big\langle f, g \big\rangle = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j K(y_j, x_i),$$

where $f = \sum_{i=1}^{n} \alpha_i K(\cdot\,, x_i)$ and $g = \sum_{j=1}^{m} \beta_j K(\cdot\,, y_j)$.

Covariance functions are positive functions too. If we are given a stochastic process $X(t)$—that is, a family of random variables indexed by t in an index set T (when t is time, stochastic processes are sometimes called *time series*)—we can define the *mean function*

$$m(t) = \mathbf{E}\left\{X(t)\right\},$$

the *second moment function*

$$R(t, s) = \mathbf{E}\left\{X(t)X(s)\right\},$$

and the *covariance function*

$$\begin{aligned} K(t, s) &= \mathbf{E}\left\{\big(X(t) - m(t)\big)\big(X(s) - m(s)\big)\right\} \\ &= R(t, s) - m(t)m(s). \end{aligned}$$

Covariance functions are positive functions (and covariance matrices are positive semidefinite). This means that all covariance functions are kernel functions and, vice versa, all kernel functions are covariance functions.

There is a close link between the functions in the RKHS with kernel K and the stochastic process with covariance function K. This link is explained in terms of *representations*.

Definition 2.4.3 (Representation of a stochastic process)**.** A family of vectors $\{f(t), t \in T\}$ in a Hilbert space $\mathcal{H}$ is a representation of the second order stochastic process $X(t)$ with second moment function $R(\cdot, \cdot)$ if, for every s and t,

$$\big\langle f(t), f(s) \big\rangle = R(t, s).$$

There are many representation theorems that link stochastic processes and Hilbert spaces. Here, we will use *Parzen's theorem*:

Theorem 2.4.4 (Parzen, 1963)**.** If $X(t)$ is a zero mean stochastic process with second moment $R(s, t) = \mathbf{E}\left\{X(s)X(t)\right\}$, then $R(\cdot, \cdot)$ is the reproducing kernel of a RKHS, and the family of functions $\{R(\cdot\,, t),\, t \in T\}$ is a representation for $X(t)$.

Traditionally, representation theorems were used to construct Hilbert spaces corresponding to stochastic processes. For instance, Wiener used (unknowingly) representation theorems to link filtering and prediction problems of stochastic processes to the solution of the Wiener-Hopf integral equations (DeSantis, Saeks,

and Tung, 1978). In the following, we take the converse path: we will create the stochastic equivalents of the interpolation and smoothing problems in the previous section and show that these variational problems can be solved with probabilistic methods.

This is possible using the concept of *congruence* between Hilbert spaces. Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two Hilbert spaces. Let $\{v_t^1, t \in T\}$ be a family of vectors that span $\mathcal{H}_1$ and let $\{v_t^2, t \in T\}$ be a family of vectors that span $\mathcal{H}_2$; then $\mathcal{H}_1$ and $\mathcal{H}_2$ are *congruent* if, for every s and t in T,

$$\left\langle v_t^1, v_s^1 \right\rangle_{\mathcal{H}_1} = \left\langle v_t^2, v_s^2 \right\rangle_{\mathcal{H}_2}.$$

The simplest stochastic process with a given covariance function is the *Gaussian process*:

Definition 2.4.5 (Gaussian process)**.** A stochastic process $X(t)$ is a Gaussian process if, for any $t_1, \ldots, t_n$ in T, the random variables $X(t_1), \ldots, X(t_n)$ are jointly Gaussian.

To any stochastic process $\{X(t), t \in T\}$ (in particular, to a Gaussian process) we can associate a Hilbert space. Consider the space L of finite combinations of random variables in the process—that is, the space of random variables Y that can be written as

$$Y = \sum_{j=1}^{n} \alpha_j X(t_j)$$

for some $n \in \mathbb{N}$, $\alpha_j \in \mathbb{R}$, and $t_j \in T$. This space can be equipped with an inner product, but it is not necessarily complete (it is a pre–Hilbert space). Let $\mathcal{L}$ be the closure of L in the space of second order random variables. We can define the space spanned by the stochastic process as follows.

Definition 2.4.6 (Hilbert space spanned by a stochastic process)**.** The space $\mathcal{L}$ is the space spanned by the stochastic process $X(t)$.

The space spanned by a stochastic process is the set of random variables that can be constructed with linear operations, including limits, on the random variables in the process. For $Y, Z \in \mathcal{L}$, the inner product in $\mathcal{L}$ is defined as

$$\left\langle Y, Z \right\rangle_{\mathcal{L}} = \mathbf{E}\left\{Y, Z\right\}.$$

So, if we have a RKHS we can construct a representation of it with a stochastic process $X(t)$ (and in particular a Gaussian process). This is very convenient because it allows us to pose a functional-approximation problem, jump over to a probabilistic setting to do the computations, and then jump back to the functional setting to get the solution. This is possible using Loève's theorem.

Theorem 2.4.7 (Loève, 1978)**.** The space $\mathcal{L}$ spanned by the stochastic process $X(t)$ with second moment function $R(\cdot, \cdot)$ is congruent to the RKHS $\mathcal{H}$ with reproducing kernel $R(\cdot, \cdot)$.

Because they are congruent, the inner products in the two spaces are the same, in the sense that

$$\big\langle X(t), X(s)\big\rangle_{\mathcal{L}} = \big\langle R(\cdot\,, t), R(\cdot\,, s)\big\rangle_{\mathcal{H}};$$

furthermore, there is a *congruence function* ϕ from $\mathcal{L}$ to $\mathcal{H}$ that maps any vector in $\mathcal{L}$, with coordinates a_i, into a vector in $\mathcal{H}$ with the same coordinates:

$$\phi\left(\sum_{i=1}^{n} a_i X(t_i)\right) = \sum_{i=1}^{n} a_i R(\cdot\,, t_i). \tag{2.7}$$

2.4.1 The interpolation problem (again)

In the interpolation problem (see Section 2.3.1), we were looking in the RKHS with kernel K for the function f, of minimum norm, that interpolates a given set of N observations (x_i, y_i). Using Parzen's theorem (Theorem 2.4.4), we know that there is a stochastic process $F(x)$ that represents the functions in the Hilbert space, and that this stochastic process has covariance function $K(\cdot\,,\cdot)$. Using Loève's theorem (Theorem 2.4.7), we can say that any function f in the RKHS is linked by congruence to a random variable F in the space $\mathcal{L}$ spanned by the stochastic process $F(x)$, with covariance function $K(x_1, x_2)$, and that

$$\big\langle f, f\big\rangle_{\mathcal{H}} = \big\langle F, F\big\rangle_{\mathcal{L}} = \mathbf{E}\left\{FF\right\} = \mathbf{cov}\left\{F\right\}.$$

So, the objective of minimizing the norm of f can be translated into the objective of finding a random variable of minimum variance. As for the constraints, we use the congruence (2.7) to say that

$$\big\langle f, K(\cdot\,, x_i)\big\rangle_{\mathcal{H}} = \mathbf{E}\left\{\phi^{-1}(f), F(x_i)\right\},$$

and that

$$\phi^{-1}(f) = \phi^{-1}\left(\sum_{i=1}^{n} \alpha_i K(\cdot\,, x_i)\right) = \sum_{i=1}^{n} \alpha_i F(x_i).$$

The interpolating solution at any point x is then given by

$$f(x) = \big\langle K(\cdot\,, x), f\big\rangle_{\mathcal{H}} = \big\langle F(x), \phi^{-1}(f)\big\rangle_{\mathcal{L}} = \mathbf{E}\left\{F(x), \sum_{i=1}^{n} \alpha_i F(x_i)\right\},$$

under the constraint that

$$\sum_{i=1}^{n} \alpha_i \mathbf{E}\left\{F(x_i), F(x_i)\right\} = y_i.$$

We recover the solution at any point x as the linear minimum-variance estimate:

$$\hat{F}(x) = \Sigma_{xy}\Sigma_{yy}^{-1} y,$$

where $[\Sigma_{xy}]_i = \mathbf{E}\{F(x)F(x_i)\}$ and $[\Sigma_{yy}]_{i,j} = \mathbf{E}\{F(x_i)F(x_j)\}$. This framework is very interesting to work with because, if we use a Gaussian process for $F(x)$, the linear minimum-variance estimate is also the mean of the posterior distribution of $F(x)$ given $F(x_1), \ldots, F(x_n)$. This consideration brings us directly to Gaussian regression.

2.5 Gaussian regression

The problem of regression is to reconstruct a linear relationship between a given set of linearly independent vectors $u_i \in \mathbb{R}^N$ (called *regressors*) and another vector $y \in \mathbb{R}^N$ (the *data*):

$$\sum_{i=1}^{n} m_i u_i = y.$$

The coefficients m_i are the unknowns that we are looking for[6]. By collecting the unknowns into a vector, we can write the regression model

$$\Phi m = y, \tag{2.8}$$

where Φ is the full column-rank matrix

$$\Phi = \begin{bmatrix} u_1 & \cdots & u_n \end{bmatrix}.$$

The system of linear equations in (2.8) can either have one single solution or no solution at all, according to the Rouché–Capelli theorem: if y is in the column span of Φ, then there is a unique solution $\hat{m}$ (see Kailath, Sayed, and Hassibi, 2000, Section 2.A).

When (2.8) represents the relationship between a model Φ and actual measurements y, there will in general be no solution, because the noise will move y out of the span of Φ. In this case, the model becomes

$$y = \Phi m + \varepsilon, \tag{2.9}$$

where ε is the unknown *error*. We can find the *least-squares solution* $\hat{m}$—that is, the solution that minimizes the squared norm of the error—as

$$\hat{m} = \arg\min_m \|y - \Phi m\|^2 = \Phi^\dagger y, \tag{2.10}$$

[6] If the regressors are linearly dependent, we select a linearly independent subset of them and reduce the problem: suppose $u_n = \sum_{j=1}^{n-1} \alpha_j u_j$ and $u_1, \ldots u_{n-1}$ are linearly independent, then

$$\sum_{i=1}^{n} m_i u_i = \sum_{i=1}^{n-1} (m_i + m_n \alpha_i) u_i = \sum_{i=1}^{n-1} \tilde{m}_i u_i$$

and we reformulate the problem in the new unknowns $\tilde{m}_i$

where $\Phi^\dagger = (\Phi^T\Phi)^{-1}\Phi^T$ is the pseudoinverse of Φ (see Kailath, Sayed, and Hassibi, 2000, Lemma 2.2.2).

Often, we have a model for the errors—for instance, if the errors are due to measurement noise with a certain distribution $p_\varepsilon(\cdot)$. Using the model, we can set up a *maximum-likelihood* problem, where we see the unknowns m as parameters of the distribution of the data and find the parameters that give the highest likelihood to the observed data:

$$\hat{m} = \arg\max_m p_y(y; m) = \arg\max_m p_\varepsilon(y - \Phi m). \tag{2.11}$$

In this equation, $p_y(\cdot\,; m)$ is the distribution of y coming from the model (2.9) with ε distributed according to $p_\varepsilon(\cdot)$. If the noise is white and Gaussian (with known variance σ^2), we recover the least-squares solution (2.10):

$$\hat{m} = \arg\max_m p(y; m) = \arg\max_m \log p(y; m) = \arg\min_m \|y - \Phi m\|^2.$$

If the distribution of the noise contains some other parameters θ to be estimated, we can add these to the maximum-likelihood problem, finding

$$\hat{m}, \hat{\theta} = \arg\max_{m,\theta} p_y(y; m, \theta) = \arg\max p_\varepsilon(y - \Phi m; \theta).$$

Suppose now that m is not a vector of general coefficients, but it is a vector of evaluations of an unknown function $m(t)$ at certain design points (sometimes called *input locations*) $t_1, \ldots, t_n$ in a set T. As we saw in the interpolation and smoothing problems, it is impossible to reconstruct it from point observations if we do not specify anything about the function $m(\cdot)$. However, if we know something about the function —it is smooth, it is exponentially decaying, it is periodic— then we can search for $m(\cdot)$ in the set of functions that share that characteristic. We encode the desired properties of the function in a kernel $K(\cdot\,,\cdot)$ and we look for the function of minimum norm in $\mathcal{H}$. What we get is a slightly modified version of the interpolation problem, where we do not have direct observations of $m(\cdot)$, but observations through the linear model (2.9):

$$\hat{m}(\cdot) = \arg\min_{m(\cdot)\in\mathcal{H}} \|y - \Phi m\|^2 + \gamma\|m(\cdot)\|^2_{\mathcal{H}}, \tag{2.12}$$

where the elements of the vector $m \in R^n$ are $m_i = m(t_i)$. In view of Parzen's and Loève's theorems (Theorem 2.4.4 and Theorem 2.4.7), it is not surprising that this variational problem can be given a probabilistic interpretation.

We construct a Gaussian process[7] $M(t)$ with covariance function $K(\cdot\,,\cdot)$. This process is linked to the functions in $\mathcal{H}$ through the congruence

$$\phi\left(\sum_{i=1}^n a_i M(t_i)\right) = \sum_{i=1}^n a_i K(\cdot\,, t_i).$$

[7]We consider zero mean Gaussian processes to keep the formulas simple, everything remains valid considering nonzero-mean processes.

Consequently, the vector m is a vector of Gaussian random variables, $m_i = M(t_i)$, with joint distribution

$$p(m) = \mathcal{N}(m\,; 0, K), \qquad [K]_{i,j} = K(t_i, t_j). \tag{2.13}$$

Looking at the model (2.9), we see that (assuming that m and ε are independent) y and m have a joint Gaussian distribution:

$$p\left(\begin{bmatrix} y \\ m \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} y \\ m \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \Phi K \Phi^T + \sigma^2 I & \Phi K \\ K\Phi^T & K \end{bmatrix}\right),$$

and the posterior distribution of m given the observations of y is given by (see Theorem A.2)

$$p(m|y) = \mathcal{N}(m\,; Cy, P),$$

where

$$P = \left(\frac{\Phi^T\Phi}{\sigma^2} + K^{-1}\right)^{-1}, \qquad C = \frac{P\Phi^T}{\sigma^2}.$$

The minimum-variance estimate is given by

$$\hat{m} = \mathbf{E}\{m|y\} = Cy. \tag{2.14}$$

This expression gives the estimates of the process $M(t)$ at the input locations $t_1, \ldots, t_n$. For any other time t, we use interpolation[8]

$$\hat{M}(t) = \mathbf{E}\{M(t)|y\} = K_t K^{-1} \hat{m},$$

where

$$K_t = \mathbf{E}\left\{M(t)\, m^T\right\} = \begin{bmatrix} K(t,t_1) & \cdots & K(t,t_n) \end{bmatrix}.$$

If we construct the vector $\alpha = K^{-1}\hat{m}$, we see that

$$\begin{aligned} \hat{M}(t) &= \sum_{i=1}^{n} \mathbf{E}\{M(t)M(t_i)\}\,\alpha_i = \left\langle M(t), \sum_{i=1}^{n} \alpha_i M(t_i)\right\rangle_{\mathcal{L}} \\ &= \left\langle \phi(M(t)), \sum_{i=1}^{n} \alpha_i \phi(M(t_i))\right\rangle_{\mathcal{H}} = \left\langle K(\cdot\,,t), \sum_{i=1}^{n} \alpha_i K(\cdot\,,t_i)\right\rangle_{\mathcal{H}}. \end{aligned}$$

In the last equality, we recognize the reproducing property of the kernel and we can say that the estimate of the function $m(\cdot)$, in $\mathcal{H}$, is given by

$$\hat{m}(\cdot) = \sum_{i=1}^{n} \alpha_i K(\cdot\,, t_i).$$

[8] This comes from the conditional independence of $M(t)$ and y, given $M(t_1), \ldots, M(t_n)$.

To further strengthen the link between the Gaussian-regression formulation and the variational problem (2.12), we can use a Bayesian view. If we look at the measurement model (2.9), we can construct the likelihood of the data. This likelihood includes the samples of the Gaussian process m_t at the input locations collected in the vector m. This vector has a Gaussian distribution. If we consider the distribution of m as a *prior distribution*, we can find the maximum-a-posteriori estimate of m as

$$\hat{m} = \arg\max_m \log p(m|y) = \arg\min_m -\log p(y\,;m)p(m).$$

Writing out the last term, we have

$$-\log p(y\,;m)p(m) = \frac{1}{2\sigma^2}\|y - \Phi m\|^2 + \frac{1}{2}m^T K^{-1} m.$$

The solution to this problem is given by

$$\hat{m} = \left(\Phi^T\Phi + \sigma^2 K^{-1}\right)^{-1}\Phi^T y,$$

which is exactly the solution found in (2.14).

With this last observation, we have closed the circle. The procedure is schematically represented in Figure 2.3.

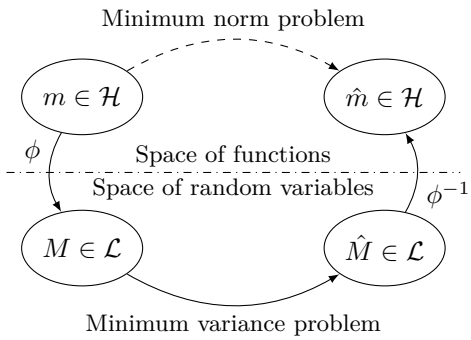

Figure 2.3: We start from the problem of estimating a function m. We construct an equivalent problem in a space of random variables. Solving this equivalent problem, we find a solution to the initial problem.

We started from the general problems of interpolation and smoothing as functional estimation problems—that is, estimate a function $m \in \mathcal{H}$. We showed that the natural space in which to look for solutions is a RKHS with a suitable kernel; then, using Loève's theorem, we constructed a Gaussian process that spans a space $\mathcal{L}$ that is congruent to the chosen RKHS. In this space, we showed that the estimation problem can be seen as the minimum-variance estimation problem of a random variable M. From the estimated random variable, we traced the congruence back into the RKHS to get the solution to the functional estimation problem we started with. What makes this approach very interesting is that it enables the use of many

techniques (and some interesting interpretations) from theory of probability; such as empirical Bayes, the Gibbs sampler, and marginalization (see Chapter 3).

In the following, we will almost exclusively use the probabilistic interpretation. We will only see Gaussian-process regression as a Bayesian estimation problem. However, it is important to keep the link to RKHS in the back pocket, ready for use should need arise. In the next chapter, we introduce the statistical tools we will use to perform the modeling and the estimation of the uncertain-input system.

Chapter 3

Bayes, Empirical Bayes and the EM-method

When we use *Bayesian techniques*, or we say that we believe in *Bayesian approaches*, what we are really saying is that we adhere to the interpretation of probability that refers to the reverend Thomas Bayes. In the Bayesian interpretation, probability is the measure of a *degree of belief*.

In the frequentist (or *operational*) definition, probability is the limit of a ratio of occurrences of an event E over the total number of observations. For example, consider the event

$$E_i = \{\text{the result of the } i\text{th coin toss is heads}\}.$$

and the indicator function

$$I(E) = \begin{cases} 1 & \text{if } E \text{ is true} \\ 0 & \text{otherwise} \end{cases}.$$

If we toss N coins in sequence, we can define the *empirical probability* of observing heads as

$$p_N = \frac{1}{N} \sum_{i=1}^{N} I(E_i),$$

and the *probability* p of each event E_i is defined as the limit

$$p = \lim_{N \to \infty} p_N.$$

What Bayesians argue is that this operational definition fails when we try to define probabilities of various events; for instance, "p is the probability that it will rain tomorrow" or "p is the probability that it rained on the 23rd of March 1382 (supposed of course you do not know the answer to this question)". What makes the operational approach fail is that the events we are referring to cannot be repeated infinite times to observe the average number of occurrences of the event: tomorrow will happen only once, and there is only one 23rd of March 1382.

Bayesians define the probability as a *degree of belief* (see Jeffreys, 1983, Section 1.2). When we say that there is 60% probability of downpour in the afternoon, we are saying that it is almost two times as likely to rain than it is to be sunny[1]. The rules of probability remain the same, and the difference is in general just a matter of interpretation. However, when we move over into statistics and estimation, the differences become more profound.

In *frequentist statistics* the main tool is maximum likelihood. After postulating a model for reality, in the form of a distribution $p(\cdot\,;\theta)$ with some unknown parameters θ, the frequentist statistician looks for the parameters $\hat{\theta}$ that best explain the data y she has collected[2]:

$$\hat{\theta} = \arg\max_{\theta} p(y\,;\theta).$$

The function $p(y\,;\theta)$, called the *likelihood*, is the probability distribution predicted by the model with parameters θ, where the value of the independent variable has been replaced with the data. Among the many properties of the maximum likelihood estimates, the most interesting one is its *consistency*. That maximum likelihood is consistent means that, if we collect N samples of data y_N generated from the *true* distribution $p(\cdot\,;\theta_0)$ and we construct the maximum likelihood estimate $\hat{\theta}_N$ from those data, then

$$\hat{\theta}_N \xrightarrow{p} \theta_0, \quad \text{as} \quad N \longrightarrow \infty,$$

where p indicates convergence in probability. Another very interesting property is the *asymptotic normality* of the estimates. It means that the estimates, as we increase the available data, become normally distributed around the true value θ_0, with a covariance matrix that is the inverse of the *Fisher information* matrix:

$$\big[\mathcal{I}(\theta_0)\big]_{i,j} = -\,\mathbf{E}\left\{\frac{\partial^2 \log p(y\,;\theta)}{\partial\theta_i \partial\theta_j}\right\}\Bigg|_{\theta=\theta_0}.$$

To do these computations, the frequentist statistician needs to believe that there is a *true model*, in the form of true parameters θ_0, and sets up his methods to recover a point estimate of this true value. For a review of Maximum likelihood inference, see Hastie et al. (2009, Section 8.2.2).

The Bayesian statistician does not look for the estimate of the value of the parameters, but looks for a *distribution of possible values*. According to the Bayesian paradigm, the truth, in the sense of the vector of parameters θ, is a random variable. What we are looking for is not a point estimate of the truth based on the data y, but the distribution of the plausible values of θ after having observed the data y.

[1] There is a very quaint rebuttal to the Bayesian interpretation: if probability is only a measure of belief, why should my belief be the same as the weatherman's? For an interesting teaser, see Freedman and Stark *What is the chance of an earthquake?,* Technical report, Dept. of Statistics Univ. of California

[2] In the following, the term *model* will be used interchangeably for the data-generating mechanism, the distribution of the data and the parameters θ of the distribution

To do Bayesian inference, we start from the definition of a *prior*–that is, a distribution $p(\theta)$ of the values of θ. The prior encodes our initial state of belief regarding the values of θ. If θ is the probability of obtaining heads in the coin toss, a reasonable prior belief is that $\theta = 0.5$, if we do not have strong reasons to suppose that the coin is biased[3]. Together with a prior, we need to define a data-generating mechanism, in the form of a *likelihood*[4] $p(y|\theta)$. The likelihood describes the probability of observing different data for different values of θ. In the coin toss example, if we have performed N tosses and observed n heads, the data is $y = \{N, n\}$ and the likelihood is the binomial distribution:

$$p(y\,|\theta) = \binom{N}{n} \theta^n (1-\theta)^{N-n}.$$

When we have likelihood and prior, we combine them using Bayes' rule to find the *posterior*:

$$p(\theta|y) = \frac{p(y\,|\theta)p(\theta)}{p(y)}, \tag{3.1}$$

where we have defined the *marginal distribution* (sometimes called *evidence* or *partition function*)

$$p(y) = \int p(y\,|\theta)p(\theta)\mathrm{d}\theta.$$

Distribution (3.1) describes our belief about the values of θ as modified by the information given by the observation of y.

After this brief introduction to Bayesian estimation, we will address the following question: what if we do not have a single prior, but our prior belief depends on the value of some other parameters that we need to estimate?

3.1 Hierarchical models, Full Bayes and Empirical Bayes

In the cases when the prior depends on other parameters, $p(\theta\,|\rho)$, a fully Bayesian approach would dictate the introduction of another prior $p(\rho)$ for the parameters of the prior. As the result of a Bayesian estimation is the posterior distribution, this approach would give us an estimate of these parameters as

$$p(\rho\,|y) = \frac{p(y\,|\rho)p(\rho)}{p(y)},$$

[3] "The biased coin is the unicorn of probability theory—everybody has heard of it, but it has never been spotted in the flesh [...] In fact, the biased coin does not exist, at least as far as flipping goes", Gelman and Nolan (2002)

[4] The frequentist likelihood and the Bayesian likelihood have the same form, however in the Bayesian framework the likelihood is really the conditional distribution of the data given the parameters, $p(y|\theta)$.

where

$$p(y\,|\rho) = \int p(y\,|\theta)p(\theta\,|\rho)\mathrm{d}\theta, \quad \text{and} \quad p(y) = \int p(y\,|\theta)p(\theta\,|\rho)p(\rho)\mathrm{d}\theta\mathrm{d}\rho.$$

As an estimate for θ, we would have

$$p(\theta|y) = \frac{p(y\,|\theta)p(\theta)}{p(y)}, \tag{3.2}$$

where

$$p(\theta) = \int p(\theta\,|\rho)p(\rho)\mathrm{d}\rho.$$

What happens now if the new parameters also have a prior that depends on some other parameters? We would have to define another prior for these parameters. What results from this procedure is a *hierarchical model*, where each layer is conditionally dependent from the layers above (Bernardo and Smith, 1994, Section 5.6.4).

When is this hierarchy of parameters going to end? If we follow the *full Bayes* approach, we would proceed as long as we have reasonable priors, building a hierarchy of dependent random variables. Often the hierarchy is stopped by introducing *non-informative priors*, or *diffuse priors* (see, e.g., Murphy, 2012, Section 5.4 or Bishop, 2006, Section 2.4.3).

Another possible approach is the *empirical Bayes* approach. In empirical Bayes we can stop the hierarchy at any point, saying that the remaining parameters are *hyperparameters*.

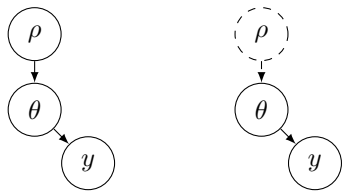

Figure 3.1: Bayesian network of a simple hierarchical model. Nodes are variables, and edges represent dependence. Shows the difference between a full Bayesian model (left) and an empirical Bayesian model (right).

Figure 3.1 shows the *Bayesian network* representation of the two hierarchical models. Each node is a variable, solid circles represent random variables, dashed circles are parameters. The arrows represent conditional dependences (Bayesian networks are also called *graphical models*, see Bishop, 2006, Chapter 8).

The empirical Bayes approach allows us to approximate the true posterior distribution (3.1), the one obtained after integrating out the hyperparameters, with the posterior for a fixed value $\hat{\rho}$ of the hyperparameters:

$$p(\theta|y) \approx p(\theta|y\,|\hat{\rho}) = \frac{p(y\,|\theta)p(\theta\,|\hat{\rho})}{p(y\,|\hat{\rho})}.$$

The value $\hat{\rho}$ is fixed at the *marginal-likelihood* estimate:

$$\hat{\rho} = \arg\max_{\rho} p(y\,|\rho).$$

This estimate is sometimes called *type-2 maximum likelihood* for its resemblance with the frequentist approach of maximizing the likelihood of the data given the parameters. If the prior assumption is true—that is, θ is a latent variable with distribution $p(\theta\,;\rho)$—then $\hat{\rho}$ is indeed the maximum-likelihood estimate of the hyperparameters.

As we have seen, empirical Bayes is a way of estimating the shape of the prior distribution, in the sense of the hyperparameters, from the available data y. Because what we are doing is, in essence, maximum likelihood, we can use methods for maximum likelihood to estimate the hyperparameters. To this end, one attractive computational method is the expectation-maximization method.

3.2 The expectation-maximization method

The *expectation-maximization method* (or *EM method*) is a family of methods to find maximum-likelihood estimates in problems with latent variables (or *unobserved* variables, *nuisance parameters*, *missing data*). Consider the maximum-likelihood problem

$$\hat{\rho} = \arg\max_{\rho}\ L_c(y, \theta\,;\rho), \tag{3.3}$$

where $L_c(y, \theta\,;\rho) = \log p(y, \theta\,;\rho)$ is the joint log likelihood of y and θ; ρ are the parameters to estimate. Suppose now that we only have data (in the form of observations) of y, and no observation of θ. In this case θ are latent variables, and the maximum-likelihood approach would be to integrate out the latent variables and maximize the resulting likelihood:

$$\hat{\rho} = \arg\max_{\rho}\ \log p(y\,;\rho) = \arg\max_{\rho}\ \log \int p(y, \theta\,;\rho)\mathrm{d}\theta. \tag{3.4}$$

There are problems, as we will see in the next chapter, where the complete problem (3.3) has a trivial solution, and the marginal problem (3.4) is very difficult to solve. In these cases, the EM method comes to aid.

The EM method is an iterative procedure that solves maximum-likelihood problems with latent variables. It consists of two steps. In the first, we create a lower bound on the marginal likelihood $l(y\,;\rho) = \log p(y\,;\rho)$ based on an estimate of the parameters and the latent variables. In the second, we maximize the lower bound, driving the marginal likelihood upwards as well.

The lower bound on $l(y\,;\rho)$ is created in the *expectation step* (the E in EM):

$$\textbf{E-step} \quad Q(\rho, \hat{\rho}^{(k)}) = \mathbf{E}\left\{L_c(y, \theta\,;\rho)\right\},$$

where the expectation is taken with respect to the posterior distribution $p(\theta|y\,;\hat{\rho}^{(k)})$ of the latent variables, for a fixed value $\hat{\rho}^{(k)}$ of the parameter ρ.

In the *maximization step* (the M in EM), we find an update of the parameter ρ by maximizing the function $Q(\rho, \hat{\rho}^{(k)})$:

$$\textbf{M-step} \quad \hat{\rho}^{(k+1)} = \arg\max_{\rho} Q(\rho, \hat{\rho}^{(k)}).$$

With the updated parameters, we can calculate the lower bound in the E-step again.

The iterations are then repeated, providing a sequence of better and better estimates of ρ, called an *EM sequence*.

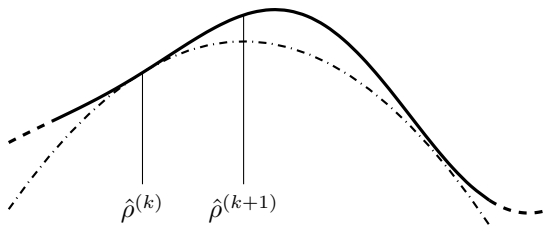

Figure 3.2: Schematic representation of one iteration of the EM method; the lower bound (dashdotted) is maximimized to find an updated estimate of higher likelihood (black).

Figure 3.2 shows a schematic representation of one iteration of the EM method. The black curve is the likelihood. We have available one estimate $\hat{\rho}^{(k)}$ and we use it to construct the lower bound $Q(\rho\,;\hat{\rho}^{(k)})$, shown in the dashdotted line. The maximum of $Q(\rho\,;\hat{\rho}^{(k)})$ corresponds to a parameter value $\hat{\rho}^{(k+1)}$ of higher likelihood.

To understand why each iteration of the EM method gives parameter values of higher likelihood, we start from the observation that

$$p(y\,;\rho) = \frac{p(y,\theta\,;\rho)}{p(\theta|y\,;\rho)}.$$

Therefore, we can decompose the marginal likelihood as

$$l(y\,;\rho) = L_c(y,\theta\,;\rho) - \log p(\theta|y\,;\rho).$$

Taking the expectation with respect to $p(\theta|y\,;\hat{\rho}^{(k)})$, we can define

$$l(y\,;\rho) = Q(\rho\,;\hat{\rho}^{(k)}) + H(\rho\,;\hat{\rho}^{(k)}), \tag{3.5}$$

where

$$Q(\rho\,;\hat{\rho}^{(k)}) = \mathbf{E}\left\{L_c(y,\theta\,;\rho)\right\}, \quad H(\rho\,;\hat{\rho}^{(k)}) = -\mathbf{E}\left\{\log p(\theta|y\,;\rho)\right\}.$$

From its definition, we see that the function $H(\rho\,;\hat{\rho}^{(k)})$ has a minimum in $\hat{\rho}^{(k)}$:

$$H(\rho\,;\hat{\rho}^{(k)}) - H(\hat{\rho}^{(k)}\,;\hat{\rho}^{(k)}) = \mathbf{E}\left\{\log\left[\frac{p(\theta|y\,;\hat{\rho}^{(k)})}{p(\theta|y\,;\rho)}\right]\right\} \geq 0,$$

because, from *Jensen's inequality*, we have

$$\mathbf{E}\left\{\log\left[\frac{p(\theta|y\,;\hat{\rho}^{(k)})}{p(\theta|y\,;\rho)}\right]\right\} \geq -\log\left[\mathbf{E}\left\{\frac{p(\theta|y\,;\rho)}{p(\theta|y\,;\hat{\rho}^{(k)})}\right\}\right] = -\log\left[\int p(\theta|y\,;\rho)\mathrm{d}\theta\right] = 0,$$

with equality only when $\rho = \hat{\rho}^{(k)}$.

Returning to the marginal likelihood (3.5), we see that

$$\begin{aligned} l(y\,;\rho) - l(y\,;\hat{\rho}^{(k)}) &= Q(\rho\,;\hat{\rho}^{(k)}) - Q(\hat{\rho}^{(k)}\,;\hat{\rho}^{(k)}) + H(\rho\,;\hat{\rho}^{(k)}) - H(\hat{\rho}^{(k)}\,;\hat{\rho}^{(k)}) \\ &> Q(\rho\,;\hat{\rho}^{(k)}) - Q(\hat{\rho}^{(k)}\,;\hat{\rho}^{(k)}), \end{aligned} \tag{3.6}$$

for $\rho \neq \hat{\rho}^{(k)}$. Choosing the updated parameters $\hat{\rho}^{(k+1)}$ as the maximum of $Q(\rho\,;\hat{\rho}^{(k)})$, we have that, as long as $\hat{\rho}^{(k+1)} \neq \hat{\rho}^{(k)}$, the likelihood is increased at each iteration:

$$l(y\,;\hat{\rho}^{(k+1)}) > l(y\,;\hat{\rho}^{(k)}).$$

We now consider two parallel aspects. First, if the likelihood is upper bounded, then the values of the likelihood will converge. Second, if the EM sequence $\hat{\rho}^{(k)}$ converges to some $\rho^\star$, then $\rho^\star$ is a stationary point of the likelihood.

To establish the first result, it is enough to notice that, from (3.6), the likelihood is nondecreasing. If the likelihood is upper bounded, $l(y\,;\hat{\rho}^{(k)})$ cannot diverge, then there is some value $l^\star$ such that it converges:

$$l(y\,;\hat{\rho}^{(k)}) \longrightarrow l^\star \quad \text{when} \quad k \longrightarrow \infty.$$

For the second result, we notice that the likelihood is increased at every iteration, as long as $\hat{\rho}^{(k)}$ is not a maximum of $Q(\rho\,;\hat{\rho}^{(k)})$. On the contrary, if $\hat{\rho}^{(k)}$ is a maximum of $Q(\rho\,;\hat{\rho}^{(k)})$, then differentiating (3.5) we obtain

$$\left.\frac{\partial\, l(y\,;\rho)}{\partial \rho}\right|_{\rho=\hat{\rho}^{(k)}} = \left.\frac{\partial\, Q(\rho\,;\hat{\rho}^{(k)})}{\partial \rho}\right|_{\rho=\hat{\rho}^{(k)}} = 0,$$

where we have used the fact that $H(\rho\,;\hat{\rho}^{(k)})$ has a minimum in $\hat{\rho}^{(k)}$. This means that, if maximizing $Q(\rho\,;\hat{\rho}^{(k)})$ gives $\rho = \hat{\rho}^{(k)}$, then $\hat{\rho}^{(k)}$ is a stationary point of the marginal likelihood. If the EM sequence converges, it has to converge to a stationary point of the likelihood.

We have established that the likelihood sequence $l(y;\hat{\rho}^{(k)})$ always converges; however, this does not guarantee the convergence of the EM sequence. On the other hand, the convergence of the EM sequence to a stationary point of the likelihood does not guarantee convergence to a local maximum. There are examples of EM sequences that converge to saddle points, or even local minimizers, of the likelihood (see Section 3.6 in McLachlan and Krishnan, 2007 for examples). The only thing we have established is that *if the EM sequence converges, then it converges to a stationary point of the likelihood*. Before solving the convergence issue, let us recapitulate what we have found. First, any EM sequence $\hat{\rho}^{(k)}$—that is, a sequence

of parameter values generated by the EM iterations—increases the likelihood $l(y\,;\rho)$. If the likelihood is upper bounded, then $l(y\,;\hat{\rho}^{(k)})$ converges to some value $l^\star$.

The main convergence result can be found in Wu (1983):

Theorem 3.2.1 (Wu, 1983)**.** Let $\hat{\rho}^{(k)}$ be an EM sequence and suppose that $Q(\rho_1\,;\rho_2)$ is continuous in ρ_1 and ρ_2. Then, $\hat{\rho}^{(k)}$ converges to a stationary point of $l(y\,;\rho)$.

To establish whether the stationary point is a local maximum, we can use the following theorem.

Theorem 3.2.2 (Wu, 1983)**.** Let $\hat{\rho}^{(k)}$ be an EM sequence and suppose that $Q(\rho_1\,;\rho_2)$ is continuous in ρ_1 and ρ_2. Furthermore, suppose that

$$\sup_{\rho} Q(\rho\,;\rho^0) > Q(\rho^0\,;\rho^0),$$

for all stationary points ρ^0 of $l(y\,;\rho)$ that are not local maxima. Then, $l^\star$ is a local maximum of $l(y\,;\rho)$.

This condition may be difficult to verify, but it is the only available result on the global convergence of EM to a local maximum. However, in most practical cases the EM method will nearly always converge to a local maximum, except for very bad luck in the choice of the initialization or local pathologies in the likelihood function (McLachlan and Krishnan, 2007, Section 1.7).

The EM method is a very convenient way to deal with maximum-likelihood problems with latent variables where direct optimization of the marginal likelihood $l(y\,;\rho)$ is involved. However, it is built around the mathematical operations of expectation and maximization. What if either one, or even both, of these steps are complicated? In these cases, we can rely on a vast amount of generalizations of the EM method. In the following we will see some of these generalizations, which will be used in the rest of the work. We will first see two ways to simplify the M-step, then see to two ways to simplify the E step.

3.2.1 Generalized Expectation Maximization

Going back to (3.6), we see that to increase the likelihood we do not need to maximize $Q(\rho\,;\hat{\rho}^{(k)})$. In fact, it is enough to choose a value $\hat{\rho}^{(k+1)}$ such that

$$Q(\hat{\rho}^{(k+1)}\,;\hat{\rho}^{(k)}) > Q(\hat{\rho}^{(k)}\,;\hat{\rho}^{(k)}),$$

to ensure that $l(y\,;\rho^{(k+1)}) \geq l(y\,;\rho^{(k+1)})$.

Sequences generated like this are called *generalized EM sequences*, or GEM sequences. Every EM sequence is also a GEM sequence. This generalization was proposed directly by Dempster, Laird and Rubin in their seminal paper (Dempster, Laird, and Rubin, 1977) for the cases when the M-step is not easy to perform.

To show the convergence, we report a result by Wu. It establishes convergence of GEM sequences to a stationary point of the likelihood under mild conditions.

Theorem 3.2.3 (Wu, 1983)**.** Let $\hat{\rho}^{(k)}$ be a GEM sequence. Suppose that $Q(\rho_1\,;\rho_2)$ is continuous in ρ_1 and ρ_2 and that $l(y\,;\hat{\rho}^{(k+1)}) > l(y\,;\hat{\rho}^{(k)})$. Then, $\hat{\rho}^{(k)}$ converges to a stationary point of $l(y\,;\rho)$.

The following alternative formulation allows us to state the convergence to stationary points without needing the strict increase in the likelihood value at each iteration.

Theorem 3.2.4 (Wu, 1983)**.** Let $\hat{\rho}^{(k)}$ be a GEM sequence, and suppose that $\partial Q(\rho_1\,;\rho_2)/\partial\rho_1$ is continuous in ρ_1 and ρ_2, and that

$$\partial Q(\rho\,;\hat{\rho}^{(k)})/\partial\rho\,\big|_{\rho=\hat{\rho}^{(k+1)}} = 0.$$

Then, $\hat{\rho}^{(k)}$ converges to a stationary point $\rho^\star$ of the likelihood if either it is the unique point such that $l(y\,;\rho^\star) = l^\star$, or $\|\hat{\rho}^{(k+1)} - \hat{\rho}^{(k)}\| \to 0$ as $k \to \infty$.

3.2.2 Expectation-Conditional Maximization

The Expectation Conditional-Maximization algorithm (ECM) is a particular type of GEM sequence. It was introduced in Meng and Rubin (1993). In ECM, the M-step is replaced by a sequence of conditional maximization steps. In each step, $Q(\rho\,;\hat{\rho}^{(k)})$ is maximized subject to some constraint.

If we define a set of S constraint functions $\varphi_s(\rho)$ for $s = 1,\ldots,S$, the ECM method performs S constrained maximizations to find $\hat{\rho}^{k+s/S}$ according to

$$\begin{aligned} &\underset{\rho}{\text{minimize}} && Q(\rho\,;\hat{\rho}^{(k)}) \\ &\text{subject to} && \varphi_s(\rho) = \varphi_s(\hat{\rho}^{(k+(s-1)/S)}). \end{aligned}$$

For the method to converge as nicely as the EM method, we need that the constraint functions $\varphi_s(\rho)$ allow the method to explore the whole parameter space. In other words, the sequence of maximizations must result in a procedure that is free to explore any direction. This is referred to as the *space filling property*. We will assume that the constraint functions are differentiable, and that the gradients $\nabla\varphi_s(\rho)$ are full rank. If we consider the space spanned by the gradient of $\varphi_s(\rho)$,

$$J_s(\rho) = \{\nabla\varphi_s(\rho)v \,:\, v \in \mathbb{R}^{d_s}\},$$

where d_s is the dimensionality of $\varphi_s(\rho)$, then the space filling property is equivalent to say that

$$\bigcap_{s=1}^{S} J_s(\rho) = \{0\}.$$

The space filling property allows us to easily assess the convergence of ECM.

Theorem 3.2.5 (Meng and Rubin, 1993)**.** Suppose that all the conditional maximizations in an ECM are unique. Then, all the limit points of the ECM sequence $\hat{\rho}^{(k)}$ are stationary points of $l(y\,;\rho)$ if the constraints are space filling at $\hat{\rho}^{(k)}$.

3.2.3 Monte Carlo Expectation Maximization

This method was introduced in Wei and Tanner (1990) for the cases when the expectation step is not available in closed form. In these cases we can replace the mathematical expectation with Monte Carlo integration—that is, we replace the operations of expectation with weighted sums,

$$\mathbf{E}\left\{\psi(x)\right\} \approx \frac{1}{M}\sum_{i=1}^{M}\psi(x_i),$$

where the expectation is taken with respect to some $p(x)$, and each of the samples x_i is drawn from the same $p(x)$. Monte Carlo Expectation can be seen as an unbiased approximation of the true expectation:

$$\mathbf{E}\left\{\frac{1}{M}\sum_{i=1}^{M}\psi(x_i)\right\} = \mathbf{E}\left\{\psi(x)\right\}.$$

The variance of the estimator is inversely proportional to the number M of samples used:

$$\mathbf{cov}\left\{\frac{1}{M}\sum_{i=1}^{M}\psi(x_i)\right\} = \frac{1}{M}\mathbf{cov}\left\{\psi(x)\right\}.$$

This strengthens the intuitive idea that the larger the M the better the approximation; in many practical applications, a number of samples between 10 and 20 will suffice (see Bishop, 2006, Chapter 11).

In the case of the MCEM, the E-step is replaced by a Monte Carlo Expectation step. Given a estimate $\hat{\rho}^{(k)}$ of the parameter vector, we draw M independent samples θ_i from the posterior density $p(\theta|y\,;\hat{\rho}^{(k)})$. Using these samples, we can approximate the E-step with the sample average

$$\hat{Q}(\rho,\hat{\rho}^{(k)}) = \frac{1}{M}\sum_{i=1}^{M}L_c(y,\theta_i\,;\rho).$$

If M is large enough, this sample average will be a good approximation of the true $Q(\rho\,;\hat{\rho}^{(k)})$.

It is very difficult to assess the convergence properties of MCEM. What seems to be a commonly accepted point is that the number of samples has to increase with the iterations. In this way, MCEM becomes like a simulated annealing method. We will not deal with the technical details too much and refer the interested reader to Neath (2013) for the convergence results.

3.2.4 Stochastic Expectation Maximization

Stochastic EM is the one-sample version of MCEM. In this method, the E step is replaced with a *simulation step*. In the simulation step, one single simulation $\theta^{(k)}$ of

the missing data is created as a realization from the posterior $p(\theta|y\,;\hat{\rho}^{(k)})$. In the subsequent M-step, the complete likelihood is maximized, with the missing data replaced with the simulated value; in this way, we find the updated parameters:

$$\hat{\rho}^{(k+1)} = \arg\max_{\rho} L_c(y, \theta^{(k)}\,;\hat{\rho}^{(k)}).$$

The sequence of estimates $\hat{\rho}^{(k)}$ generated by Stochastic EM will not converge to any value; however, their stationary distribution will be around the maximum-likelihood estimate. As was the case for MCEM, it is very difficult to ascertain the convergence of Stochastic EM. The interested reader is referred to Nielsen (2000) for a survey over the method.

Chapter 4

Input Uncertainties

We study the problem of identifying an output-error model of the type

$$y_t = \sum_{k=1}^{+\infty} g_k w_{t-k} + \varepsilon_t,$$

where g_t is the impulse response of a strictly causal and asymptotically stable transfer function (the *linear system*) that represents the dynamics of the system. The *output measurements* y_t are measurements of the output of the linear system corrupted by the measurement noise process ε_t. The noise process is zero-mean white noise with unknown variance σ_y^2.

Typically, in system identification, the input to the linear system is perfectly known (Ljung, 1999, Section 2.1). In this work, we consider the case where we do not have full access to the *input* signal w_t, but only to limited information about it. This is the *uncertain-input system* framework (see Figure 4.1).

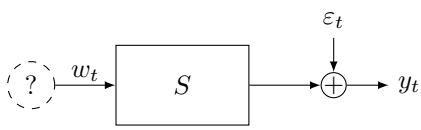

Figure 4.1: System with uncertain inputs: a linear system S is fed an input w_t of unknown nature.

Having limited information about the input means, for instance, that we have noisy measurements—effectively recovering the errors-in-variables framework—or a model generating the input. The *input measurements* v_t are measurements of the input corrupted by a zero-mean white noise process η_t with unknown variance σ_v^2. In Section 4.2, we will make clear what the *input model* is; for now, we just consider it as a kind of prior information about the input process.

We can represent the model, with the measurement signals, with the block schematic in Figure 4.2. From the figure, we see that uncertain-input systems are

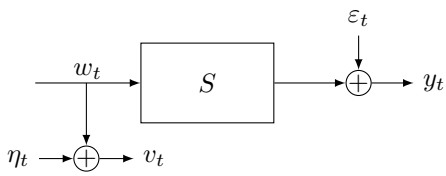

Figure 4.2: System with uncertain inputs: y_t and v_t are measurements, corrupted by the noise processes ε_t and η_t.

generalizations of errors-in-variables systems. The generalization involves the amount of information available about the input signal w_t.

Summing up, the uncertain-input system is a system represented by a mathematical description of the type

$$\begin{cases} y_t = \sum_{k=1}^{\infty} g_k w_{t-k} + \varepsilon_t \\ v_t = w_t + \eta_t \end{cases}, \tag{4.1}$$

where

- g_k is the impulse response of an exponentially stable, strictly causal transfer function,
- ε_t is zero-mean white noise with variance σ_y^2,
- η_t is zero-mean white noise with variance σ_v^2,
- ε_t and η_t are independent,
- w_t is a partially known signal.

This framework includes, as a special case, the errors-in-variables identification problem (see Section 6.6). We can recover all the classical errors-in-variables formulations by using different models for the signal w_t. Among the possible choices we can, assume knowledge about the ratio of the noise variances (Fernando and Nicholson, 1985; Söderström, 2010; Risuleo, Bottegal, and Hjalmarsson, 2016b), model the input as filtered white noise (Söderström, 2007; Pintelon and Schoukens, 2007), model the input as a periodic signal (Schoukens, Pintelon, Vandersteen, et al., 1997), model the input as the result of a known external reference signal (Forssell and Ljung, 2000; Pintelon, Schoukens, et al., 2010). In addition, by setting σ_v^2 to infinity, the framework can be used to model Hammerstein systems with nonparametric and parametric models of the static nonlinearity (see Section 6.3), as well as blind system identification problems (see Section 6.5).

In the following sections, we will introduce the models for the linear system and for the input w_t, as well as for the measurement errors. The models will depend on some parameters to be estimated from data. To simplify the expressions, we drop

the time dependency from the equations, and use the vectorized forms for all time series: if $\{x_t\}_{t=a}^{b}$ is a set of variables, the vector x is the $b-a+1$ dimensional vector whose ith component is x_{a+i-1}:

$$x = \begin{bmatrix} x_a \\ x_{a+1} \\ \vdots \\ x_{b-1} \\ x_b \end{bmatrix}. \tag{4.2}$$

4.1 Modeling the linear system

The linear system in the uncertain-input model is a linear time-invariant system S characterized by a strictly causala and asymptotically stable rational transfer function. From this assumption, it follows that the sequence of impulse response parameters $\{g_k\}_{k=1}^{+\infty}$ is such that

$$g_k = 0 \qquad \text{for all } k \leq 0, \qquad (\textbf{Causality})$$
$$\|g\|_1 = \sum_{k=1}^{+\infty} |g_k| < \infty. \qquad (\textbf{BIBO Stability})$$

To model the linear system, we will use a Gaussian regression approach. Consider the system of equations (4.1). The first observation we make is that, due to the BIBO stability assumption, the impulse response is decaying to zero. For this reason, we can truncate the impulse response after a number of samples n sufficiently large and model the system S as a (possibly long) finite impulse response:

$$g_k = 0 \qquad \text{for all } k > n. \qquad (\textbf{FIR})$$

Using the Gaussian-process framework, we can postulate a RKHS for the impulse response g_t in the form of a Gaussian-process prior with a suitable mean function $\mu_g(\cdot\,;\rho)$ and covariance function $K_g(\cdot\,,\cdot\,;\rho)$. The hyperparamers ρ parameterize the Gaussian process. The samples of the impulse response have a joint Gaussian distribution:

$$\begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix} ; \begin{bmatrix} \mu_g(1\,;\rho) \\ \vdots \\ \mu_g(n\,;\rho) \end{bmatrix} , \begin{bmatrix} K_g(1,1\,;\rho) & \cdots & K_g(1,n\,;\rho) \\ \vdots & & \vdots \\ K_g(n,1\,;\rho) & \cdots & K_g(n,n\,;\rho) \end{bmatrix} \right).$$

With a slight abuse of notation, we define the mean vector $\mu_g(\rho)$ and the covariance matrix $K_g(\rho)$ as

$$\big[\mu_g(\rho)\big]_i = \mathbf{E}\,\{g_i\} = \mu_g(i\,;\rho), \qquad \big[K_g(\rho)\big]_{i,j} = \mathbf{cov}\,\{g_i, g_j\} = K_g(i\,,j\,;\rho).$$

This allows us to write the vectorized expression for the prior:

$$p(g\,;\rho) = \mathcal{N}(g\,;\mu_g(\rho), K_g(\rho)), \tag{4.3}$$

where g is the vectorization (4.2) of g_t. We have not discussed the choice of the mean and covariance functions, as this is a modeling choice that depends on the specific application. In Chapter 6, we will see examples of how to choose the prior distribution for some applications.

4.2 Modeling the input

To model the input process w_t, we use a Gaussian-regression approach. We postulate a RKHS that encodes our information about the input.

Consider the system of equations (4.1). We use a model for the input w_t in the form of a mean function $\mu_w(\cdot\,;\theta)$ and a covariance function $K_w(\cdot\,,\cdot\,;\theta)$. With this choice, the vectorization w of the input samples w_t has a multivariate Gaussian prior distribution given by

$$p(w\,;\theta) = \mathcal{N}(w\,;\mu_w(\theta), K_w(\theta)), \tag{4.4}$$

where, with some abuse of notation, we have defined the mean vector $\mu_w(\theta)$ and the covariance matrix $K_w(\theta)$ as

$$\big[\mu_w(\theta)\big]_i = \mathbf{E}\,\{w_i\} = \mu_w(i\,;\theta), \qquad \big[K_w(\theta)\big]_{i,j} = \mathbf{cov}\,\{w_i, w_j\} = K_w(i\,,j\,;\theta).$$

Similarly to what we did for the impulse response, we do not give any specific expression for the model of the input, and we refer the reader to Chapter 6 for examples.

4.3 Modeling the measurements

Suppose we have N successive samples $y_1, \ldots, y_N$ of the output, collected in a vector y. From (4.1), after truncation of the impulse response, we can write a vectorized model for the output y as

$$y = \mathbf{T}_{N\times n}(w)g + \varepsilon,$$

where g is the vector of impulse-response samples and where we have collected the output measurements in the vector y and the noise samples in the vector ε. The matrix $\mathbf{T}_{N\times n}(w)$ is the N by n Toeplitz matrix of the input:

$$\mathbf{T}_{N\times n}(w) = \begin{bmatrix} w_0 & w_{-1} & \cdots & w_{-n+1} \\ w_1 & w_0 & \cdots & w_{-n+2} \\ w_2 & w_1 & \cdots & w_{-n+3} \\ \vdots & \vdots & & \vdots \\ w_{N-1} & w_{N-2} & \cdots & w_{N-n+2} \\ w_N & w_{N-1} & \cdots & w_{N-n+1} \end{bmatrix},$$

where we assume that $w_0, \ldots, w_{-n+1}$ are all equal to zero (in Section 6.8, we will relax this assumption and show how to estimate these initial conditions).

We assume that the output measurement noise is white and Gaussian, with variance σ_y^2:

$$p(\varepsilon\,;\sigma_y^2) = \mathcal{N}(\varepsilon\,;0,\sigma_y^2 I_N).$$

Therefore, given the input and the impulse response, we can write the distribution of the output measurements as

$$p(y|g,w\,;\sigma_y^2) = \mathcal{N}(y\,;Wg,\sigma_y^2 I_N), \tag{4.5}$$

where $W = \mathbf{T}_{N\times n}(w)$. From (4.1), we can write a vectorized measurement model for the input measurements of the form

$$v = w + \eta.$$

We assume that the input measurement noise is white and Gaussian, with variance σ_v^2:

$$p(\eta\,;\sigma_v^2) = \mathcal{N}(\eta\,;0,\sigma_v^2 I_N).$$

Therefore, given the input, we can write the distribution of the input measurements as

$$p(v|w\,;\sigma_v^2) = \mathcal{N}(v\,;w,\sigma_y^2). \tag{4.6}$$

Because the noise vectors ε and η are independent, we can write the joint distribution of the measurements as the product of (4.5) and (4.6):

$$p(y,v|g,w\,;\sigma_y^2,\sigma_v^2) = p(y|g,w;\sigma_y^2)p(v|g,w;\sigma_v^2). \tag{4.7}$$

To make the model more general, we will allow the noise variances to take the value $+\infty$ when the corresponding measurements are unavailable or completely useless.

4.4 The uncertain-input model

In the previous sections, we have constructed probabilistic models of the various components that define the uncertain-input system. When we assemble them together, we obtain what we call the *uncertain-input model*.

The uncertain-input model is a probabilistic model that describes the relationship between N samples of y_t and N samples of v_t, according to

$$\begin{cases} y = Wg + \varepsilon \\ v = w + \eta \\ g \sim \mathcal{N}(g\,;\mu_g(\rho),K_g(\rho)) \\ w \sim \mathcal{N}(w\,;\mu_w(\theta),K_w(\theta)) \\ \varepsilon \sim \mathcal{N}(\varepsilon\,;0,\sigma_y^2 I_N) \\ \eta \sim \mathcal{N}(\eta\,;0,\sigma_v^2 I_N) \end{cases}. \tag{4.8}$$

This uncertain-input model depends on two sets of hyperparameters, as well as on the noise variances, if these are unknown. We call ρ the *system hyperparameters* and θ the *input hyperparameters*. In the next chapter, we will show how we can use this model to solve the uncertain-input system identification problem—that is, to identify the impulse response g and the input w. To perform this identification, we will use an empirical Bayes approach and find the hyperparameters that maximize the marginal likelihood of the data. Before going into the identification, we will look at the probabilistic relationships in the model and how these can be exploited to make computations easier.

4.5 Probabilistic relationships in the uncertain-input model

The estimation technique we propose in the next chapter builds on the probabilistic relationships between the stochastic variables that compose the uncertain-input model. Figure 4.3 shows the Bayesian network of the uncertain-input model.

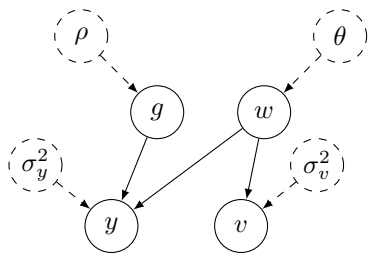

Figure 4.3: Bayesian network of the uncertain-input model. Solid circles represent stochastic variables; dashed circles represent hyperparameters.

The Bayesian network allows us to find the conditional independence structure in the model. We see that the joint distribution of all the parameters can be factorized as follows:

$$p(y, v, g, w\,;\rho, \theta, \sigma^2) = p(y|g, w\,;\sigma_y^2)p(v|w\,;\sigma_v^2)p(g\,;\rho)g(w\,;\theta), \tag{4.9}$$

where $\sigma^2 = \{\sigma_y^2, \sigma_v^2\}$.

To carry out the computations in the next chapter, it is convenient to work with Gaussian distributions. Therefore, we look for Gaussian distributions in the structure. All the factor distributions in (4.9) are Gaussian[1]:

$$\begin{aligned}
p(g\,;\rho) &= \mathcal{N}(g\,;\mu_g, K_g), && \text{(Equation (4.3))}\\
p(w\,;\theta) &= \mathcal{N}(w\,;\mu_w, K_w), && \text{(Equation (4.4))}\\
p(y|g, w\,;\sigma_y^2) &= \mathcal{N}(y\,;Wg, \sigma_y^2 I_N), && \text{(Equation (4.5))}\\
p(v|w\,;\sigma_v^2) &= \mathcal{N}(v\,;w, \sigma_y^2 I_N). && \text{(Equation (4.6))}
\end{aligned}$$

[1]We drop the dependency on the hyperparameters in the mean functions and covariance matrices, when no confusion is possible.

We can also find other Gaussian distributions in the structure. Multiplying $p(g\,;\rho)$ and $p(y|g,w\,;\sigma_y^2)$ together, we find that, conditioned on the input w, the output measurements y and the impulse response g are jointly Gaussian with distribution

$$p(y,g|w;\rho,\sigma_y^2) = \mathcal{N}\left(\begin{bmatrix} y \\ g \end{bmatrix}\,;\begin{bmatrix} W\mu_g \\ \mu_g \end{bmatrix},\begin{bmatrix} WK_gW^T+\sigma_y^2 I_N & WK_g \\ K_gW^T & K_g \end{bmatrix}\right).$$

From this joint distribution, we find that the conditional distribution of the impulse response (given the output measurements y and the input w) is Gaussian according to

$$p(g|y,w;\rho,\sigma_y^2) = \mathcal{N}\left(g\,;m_g,P_g\right), \tag{4.10}$$

with mean and covariance given by (see Theorem A.2)

$$P_g = \left(\frac{W^TW}{\sigma_y^2} + {K_g}^{-1}\right)^{-1}, \tag{4.11}$$

$$m_g = P_gW^T\left(\frac{W^T}{\sigma_y^2}y - K_g^{-1}\mu_g\right). \tag{4.12}$$

In addition, we find the marginal distribution

$$p(y|w\,;\rho,\sigma_y^2) = \mathcal{N}(y\,;W\mu_g, WK_gW^T+\sigma_y^2 I_N).$$

Using the relationship

$$\mathbf{T}_{N\times n}(w)g = \mathbf{T}_{N\times N}(g)w,$$

and defining $G = \mathbf{T}_{N\times N}(g)$, we can multiply together $p(y|g,w\,;\sigma_y^2)$, $p(w\,;\theta)$ and $p(v|w\,;\sigma_v^2)$ and find that, conditioned on the impulse response g, the output measurements y, the input measurements v, and the input w are jointly Gaussian:

$$p(y,v,w|g;\theta,\sigma^2) = \mathcal{N}\left(\begin{bmatrix} y \\ v \\ w \end{bmatrix};\begin{bmatrix} G\mu_w \\ \mu_w \\ \mu_w \end{bmatrix},\begin{bmatrix} GK_wG^T+\sigma_y^2 I_N & GK_w & GK_w \\ K_wG^T & K_w+\sigma_v^2 I_N & K_w \\ K_wG^T & K_w & K_w \end{bmatrix}\right).$$

From this joint distribution, we find that the conditional distribution of the inputs w (given the output measurements y, the input measurements v, and the impulse response g) is Gaussian according to

$$p(w|y,v,g;\theta,\sigma^2) = \mathcal{N}\left(w\,;m_w,P_w\right), \tag{4.13}$$

where (see Theorem A.1)

$$m_w = \mu_w + \begin{bmatrix} K_wG^T & K_w \end{bmatrix}\begin{bmatrix} GK_wG^T+\sigma_y^2 I_N & GK_w \\ K_wG^T & K_w+\sigma_v^2 I_N \end{bmatrix}^{-1}\begin{bmatrix} y-G\mu_w \\ v-\mu_w \end{bmatrix}, \tag{4.14}$$

$$P_w = K_w - \begin{bmatrix} K_wG^T & K_w \end{bmatrix}\begin{bmatrix} GK_wG^T+\sigma_y^2 I_N & GK_w \\ K_wG^T & K_w+\sigma_v^2 I_N \end{bmatrix}^{-1}\begin{bmatrix} GK_w \\ K_w \end{bmatrix}. \tag{4.15}$$

In addition, we find the marginal distributions

$$p(y|g\,;\theta,\sigma_y^2) = \mathcal{N}(y\,;G\mu_w, GK_wG^T + \sigma_y^2 I_N), \tag{4.16}$$

$$p(v\,;\theta,\sigma_v^2) = \mathcal{N}(v\,;\mu_w, K_w + \sigma_v^2 I_N). \tag{4.17}$$

In the next section, we will use these distributions to solve the estimation problem. As we will see, the Gaussian distributions in the structure will allow us to find an effective solution.

Chapter 5

Identification of uncertain-input models

In the previous chapter, we defined the *uncertain-input model*

$$\begin{cases} y = Wg + \varepsilon \\ v = w + \eta \\ g \sim \mathcal{N}(g\,;\mu_g(\rho), K_g(\rho)) \\ w \sim \mathcal{N}(w\,;\mu_w(\theta), K_w(\theta)) \\ \varepsilon \sim \mathcal{N}(\varepsilon\,;0, \sigma_y^2 I_N) \\ \eta \sim \mathcal{N}(\eta\,;0, \sigma_v^2) \end{cases}. \tag{5.1}$$

The model depends on some hyperparameters that need to be estimated, namely the input hyperparameters θ, the system parameters ρ, and the noise variances $\sigma^2 = \{\sigma_y^2, \sigma_v^2\}$.

In this chapter, we will see how to solve the system-identification problem of uncertain-input systems using the proposed model. The identification problem we are looking at is defined as follows.

Problem 5.1 (System identification)**.** Given a set of N noisy measurements of input and output $\{v_i, y_i\}_{i=1}^N$, generated from an uncertain-input system, estimate the noiseless input w and the system impulse response g.

To solve this problem, we will use an empirical Bayes approach.

5.1 Empirical Bayes in system identification

Suppose for a moment that we are addressing the problem of identifying the finite impulse response of a linear system of the type

$$y = Ug + \varepsilon,$$

where $U = \mathbf{T}_{N\times n}(u)$, u is the known input and ε is Gaussian measurement noise with unknown variance σ^2. For this system, we can define the Gaussian-process

model

$$p(g\,;\rho) = \mathcal{N}\big(g\,;\mu(\rho), K(\rho)\big).$$

Because the noise ε is Gaussian, we can write the joint distribution of g and the measurements y,

$$p(g, y\,;\rho) = \mathcal{N}\left(\begin{bmatrix} y \\ g \end{bmatrix}\,;\begin{bmatrix} U\mu(\rho) \\ \mu(\rho) \end{bmatrix},\begin{bmatrix} UK(\rho)U^T + \sigma^2 I & UK(\rho) \\ K(\rho)U^T & K(\rho) \end{bmatrix}\right), \tag{5.2}$$

and we can find the posterior mean of the impulse response g given the measurements y. This is the minimum mean-square-error estimate of g, in the Bayesian sense (see Anderson and J. B. Moore, 2012, Section 2.3). However, this estimate depends on the hyperparameters ρ. As we saw in Section 3.1, we can approximate the posterior mean estimator using an empirical Bayes approach: we estimate the hyperparameters maximizing the marginal likelihood of the measurements,

$$\hat{\rho},\, \hat{\sigma}^2 = \arg\max_{\rho,\sigma} \log\, p(y\,;\rho,\sigma^2),$$

where the marginal likelihood is

$$\log\, p(y\,;\rho,\sigma^2) = -\frac{1}{2}\big(y-\mu_y\big)^T\Sigma_y^{-1}\big(y-\mu_y\big) - \frac{1}{2}\log\det\Sigma_y,$$

and where

$$\mu_y = U\mu(\rho), \qquad \Sigma_y = UK(\rho)U^T + \sigma^2 I.$$

Once we have found the hyperparameters (and the noise variance), we can plug them into the joint distribution (5.2) and use Theorem A.2 to find the estimated posterior mean

$$\hat{g} = \mathbf{E}\left\{g|y\right\} = \hat{P}_g\left(\frac{U^T}{\hat{\sigma}^2}y + K(\hat{\rho})^{-1}\mu(\hat{\rho})\right),$$

where the expectation is taken with respect to the posterior distribution with hyperparameters set to $\hat{\rho}$. The estimated posterior covariance $\hat{P}_g$ is given by

$$\hat{P}_g = \left(\frac{U^TU}{\hat{\sigma}^2} + K(\hat{\rho})^{-1}\right)^{-1}.$$

The empirical Bayes method relies on the estimation of the parameters of the prior distribution from data (Bishop, 2006, Section 3.5). This estimated prior is then used to calculate the posterior distribution and eventual point estimates of interest.

5.2 Empirical Bayes estimation of uncertain-input systems

In the Bayesian framework we have set up, the minimum mean-square-error estimates of the impulse response and of the input can be found as their respective means

conditioned on the data:

$$g^* = \mathbf{E}\{g|y,v\},$$
$$w^* = \mathbf{E}\{w|y,v\},$$

where the expectations are taken with respect to the posterior distributions (4.10) and (4.13) after marginalization of w and g, respectively:

$$\begin{aligned} g^* &= \int g\, p(g|y,v\,;\rho,\theta,\sigma^2)\mathrm{d}g, \\ w^* &= \int w\, p(w|y,v\,;\rho,\theta,\sigma^2)\mathrm{d}w, \end{aligned} \tag{5.3}$$

where $\sigma^2 = \{\sigma_y^2, \sigma_v^2\}$. Notice that neither posterior distribution in (5.3) is Gaussian, so we have to use sampling methods and Monte Carlo integration (Bishop, 2006, Section 11). In addition, the posterior distributions are not available in closed form; this further complicates computations. However, we can find the posterior distributions as marginalizations of the same joint posterior distribution, that is

$$\begin{aligned} g^* &= \int g\, p(g,w|y,v\,;\rho,\theta,\sigma^2)\mathrm{d}w\,\mathrm{d}g, \\ w^* &= \int w\, p(g,w|y,v\,;\rho,\theta,\sigma^2)\mathrm{d}g\,\mathrm{d}w. \end{aligned} \tag{5.4}$$

Similarly to (5.3), these expressions involve the non–Gaussian joint posterior distribution of g and w given the data. However, both integrals are with respect to the same distribution, so we only need one sampling method to solve both problems. In addition, there is an effective way to generate samples from this joint posterior: we can use the *Gibbs sampler* (see Section 5.2.1).

The estimators (5.4) depend on the specific choice of the hyperparameters ρ, θ, and σ^2. Using the empirical Bayes approach, we estimate the posterior distribution in (5.4) by replacing the hyperparameters with their estimates $\hat{\rho}$, $\hat{\theta}$, and $\hat{\sigma}^2$. The hyperparameter estimates are found by maximizing the marginal distribution of the data:

$$\hat{\rho}, \hat{\theta}, \hat{\sigma}^2 = \arg\max_{\rho,\theta,\sigma^2} p(y,v\,;\rho,\theta,\sigma^2). \tag{5.5}$$

The noise variances σ^2 deserve a bit of a special treatment. In this most general problem they can be unidentifiable–that is, it may be impossible to estimate their values from data. This depends on the shape of the prior distributions of w and g. We will suppose that all maximizations are carried out over sets where the parameters are identifiable. See the next chapter for examples of the problem with nonidentifiability (Sections 6.3, 6.5–6.7).

With the specified hyperparameters, we can set the estimates of the impulse response and of the input as the means of the estimated posterior densities given

the data:

$$\begin{aligned}\hat{g} &= \int g\, p(g, w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2)\mathrm{d}w\,\mathrm{d}g,\\ \hat{w} &= \int w\, p(g, w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2)\mathrm{d}g\,\mathrm{d}w.\end{aligned} \tag{5.6}$$

As we already said, these integrals do not have a closed form solution. However, if we are able to sample from the posterior distribution, we can make a *particle approximation*, replacing the integration with a sum of a large number M of samples:

$$\begin{aligned}\hat{g} &= \mathbf{E}\,\{g|y, v\} \approx \frac{1}{M}\sum_{j=1}^{M} \bar{g}^{(j)},\\ \hat{w} &= \mathbf{E}\,\{w|y, v\} \approx \frac{1}{M}\sum_{j=1}^{M} \bar{w}^{(j)},\end{aligned} \tag{5.7}$$

where the samples $\bar{g}^{(j)}$, $\bar{w}^{(j)}$, are drawn, using the Gibbs sampler, from the posterior distribution $p(g, w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2)$.

5.2.1 Gibbs sampling from the joint posterior

To use the Monte Carlo approximation (5.7) we need a way to draw from the joint posterior $p(g, w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2)$. This distribution is not Gaussian and drawing from it directly is impossible. However, we can use the observations in Section 4.5 to set up an iterative method to draw the samples.

Consider the joint distribution

$$p(g, w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2),$$

we notice that

- $g|w, y$ is Gaussian (Equation (4.10)),
- $w|g, y, v$ is Gaussian (Equation (4.13)),

so it is easy to draw from these distributions. Therefore, it is very convenient to set up a *Gibbs sampler* to draw from the joint posterior distribution. Because it samples from an empirical Bayes approximation of the posterior distribution, this kind of Gibbs sampler is called *empirical Bayes Gibbs sampler* (Casella, 2001).

The general Gibbs sampler was introduced in S. Geman and D. Geman (1984) as a way to sample from a joint distribution of random variables by sampling from the conditional distributions of each variable (for an introduction to sampling methods see Bishop, 2006, Chapter 11, for the Gibbs sampler see Bishop, 2006, Section 11.3 or Murphy, 2012, Section 24.2). In turns, we select each variable in the joint distribution and draw a sample of it conditioned on its *Markov blanket*, that is the set

of its parents, its children and its coparents in the Bayesian network (Murphy, 2012, Section 24.2.1; see also Gilks, Richardson, and Spiegelhalter, 1996, Section 2.4.2). In Figure 5.1 we see the Markov blankets of g and of w for the uncertain-input system.

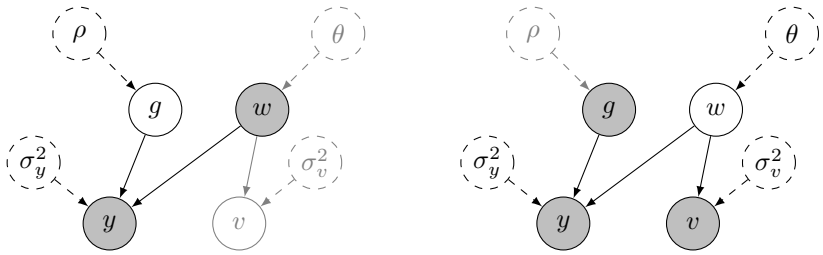

Figure 5.1: Markov blankets of g (left) and of w (right) considered in the Gibbs sampler. Shaded circles represent conditioning variables.

If we denote by $g^{(k)}$ and $w^{(k)}$ the samples at the kth iteration of the method, we draw the next sample of the chain from the conditional distributions according to

$$\begin{aligned} g^{(k+1)} &\sim p(g|y, w^{(k)}; \hat{\rho}, \hat{\sigma}_y^2), \\ w^{(k+1)} &\sim p(w|y, v, g^{(k+1)}; \hat{\theta}, \hat{\sigma}^2). \end{aligned} \tag{5.8}$$

The above iterations create samples that form a Markov chain whose stationary distribution is the desired joint posterior distribution (Bishop, 2006, Section 11.3). Together, the chains of samples $g^{(k)}$ and $w^{(k)}$ converge to the desired joint posterior distribution. In isolation, each chain of samples converges to each posterior marginal distribution in (5.6).

To implement the Gibbs Sampling Monte Carlo method (5.7), we run the Gibbs sampler (5.8) for a sufficiently large number of iterations B. This is the *burn-in* period; during these iterations the Markov chain is mixing and is far from the stationary distribution. After the burn-in period, we can collect samples from the chain and use them, confident that the samples will come from the joint posterior distribution. In practice, it is very difficult to establish the length of the burn-in period. Various heuristics have been proposed to diagnose whether the Markov chain has converged to the stationary distribution (see Murphy, 2012, Section 24.4.3). Another difficulty is that, because it is a Markov chain, successive samples from (5.8) will be correlated. If this is a nuisance, the samples can be collected from the chain only each m iterations, as doing so will reduce the correlation (this technique is called *thinning*, see Murphy, 2012, Section 24.4.4). The Gibbs sampling algorithm is presented in Algorithm 1.

5.2.2 Marginal likelihood estimation with EM

The estimator (5.6) is an empirical Bayes approximation of the true posterior mean, where we have plugged in the estimated hyperparameters. To estimate the

Algorithm 1 Sample M samples from $p(g, w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma})$ saving every mth sample from Gibbs sampling with burn-in length B

procedure POSTERIORGS
  input $y, v, \hat{\rho}, \hat{\theta}, \hat{\sigma}^2$
  parameters M, B, m
  output $\{\bar{g}^{(k)}\}_{k=1}^{M}, \{\bar{w}^{(k)}\}_{k=1}^{M}$
  $k \leftarrow 0$
  $j \leftarrow 0$
  draw $\bar{g} \sim p(g|y, w = v; \hat{\rho}, \hat{\sigma}^2)$
  draw $\bar{w} \sim p(w|y, v, g; \hat{\theta}, \hat{\sigma}^2)$
  while $k < M$ **do**
    draw $\bar{g} \sim p(g|y, \bar{w}; \hat{\rho}, \hat{\sigma}^2)$
    draw $\bar{w} \sim p(w|y, v, \bar{g}; \hat{\theta}, \hat{\sigma}^2)$
    if $j > B \quad \& \quad \mod(j, m) = 1$ **then**
      $k \leftarrow k + 1$
      $\bar{g}^{(k)} \leftarrow \bar{g}$
      $\bar{w}^{(k)} \leftarrow \bar{w}$
    $j \leftarrow j + 1$

hyperparameters we use the maximum marginal-likelihood criterion (5.5). This maximization is impossible to solve directly, so again we use an iterative method. Furthermore, the marginal distribution does not admit a closed-form expression.

The Gibbs sampler (5.8) provides a method to sample from the posterior distribution of g and w, for some fixed value of the hyperparameters, given the data. Can we use this to find the maximum marginal likelihood? The answer is yes: we can use the *EM method* (see Chapter 3).

Consider the maximum marginal-likelihood problem (5.5):

$$\hat{\rho}, \hat{\theta}, \hat{\sigma}^2 = \arg\max_{\rho, \theta, \sigma^2} p(y, v\,; \rho, \theta, \sigma^2).$$

If we reintroduce the random variables g and w,

$$\hat{\rho}, \hat{\theta}, \hat{\sigma}^2 = \arg\max_{\rho, \theta, \sigma^2} \int p(y, v, w, g\,; \rho, \theta, \sigma^2) \mathrm{d}g \mathrm{d}w,$$

we can see the maximum marginal-likelihood problem as a *maximum-likelihood problem with latent variables*, the latent variables being g and w. Using (4.9), we can write the complete log likelihood

$$\begin{aligned} L_c(y, v, w, g\,; \rho, \theta, \sigma^2) &= \log p(y, v, w, g\,; \rho, \theta, \sigma^2) \\ &= \log p(y, v|w, g\,; \rho, \theta, \sigma^2) + \log p(w, g\,; \rho, \theta, \sigma^2) \\ &= \log p(y|w, g\,; \sigma_y^2) + p(v|w\,; \sigma_v^2) + \log p(g\,; \rho) + \log p(w\,; \theta), \end{aligned}$$

where

$$\begin{aligned}
\log p(y|w,g\,;\sigma_y^2) &= -\frac{1}{2\sigma_y^2}\|y - Wg\|^2 - \frac{N}{2}\log\sigma_y^2,\\
\log p(v|w\,;\sigma_v^2) &= -\frac{1}{2\sigma_v^2}\|v - w\|^2 - \frac{N}{2}\log\sigma_v^2,\\
\log p(g\,;\rho) &= -\frac{1}{2}\big(g-\mu_g(\rho)\big)^T K_g(\rho)^{-1}\big(g-\mu_g(\rho)\big) - \frac{1}{2}\log\det K_g(\rho),\\
\log p(w\,;\theta) &= -\frac{1}{2}\big(w-\mu_w(\theta)\big)^T K_w(\theta)^{-1}\big(w-\mu_w(\theta)\big) - \frac{1}{2}\log\det K_w(\theta).
\end{aligned} \tag{5.9}$$

To define the Q function—that is,the lower bound of the marginal likelihood—we need to take the expectation of the complete likelihood with respect to the posterior distribution of the latent variables for a fixed value of the hyperparameters:

$$Q(\rho,\theta,\sigma\,;\hat\rho^{(k)},\hat\theta^{(k)},\hat\sigma^{(k)}) = \int L_c(y,v,w,g;\rho,\theta,\sigma^2)p(g,w|y,v\,;\hat\rho^{(k)},\hat\theta^{(k)},\hat\sigma^{2\,(k)})\mathrm{d}g\mathrm{d}w.$$

In this expression we see why having the Gibbs sampling method for the joint posterior is very convenient: we can approximate this integral, and the function Q, with the sum

$$\hat Q(\rho,\theta,\sigma\,;\hat\rho,\hat\theta,\hat\sigma) \approx \frac{1}{M}\sum_{j=1}^{M} L_c(y,v,\bar w^{(j)},\bar g^{(j)};\rho,\theta,\sigma^2),$$

where $\{\bar g^{(j)}\}_{j=1}^M$ and $\{\bar w^{(j)}\}_{j=1}^M$ are samples from the posterior, drawn using the Gibbs sampler (Algorithm 1), with the hyperparameters fixed at $\hat\rho^{(k)}$, $\hat\theta^{(k)}$, and $\hat\sigma^{2\,(k)}$.

Thanks to the structure in (5.9), we can go quite far in the M step. We want to maximize $\hat Q$ with respect to all the hyperparameters and with respect to the noise variances. We can split the maximization in four independent problems, because each term in (5.9) depends on only one of the variables,

$$\hat\sigma^{y\,(k+1)} = \arg\max_\rho \frac{1}{M}\sum_{j=1}^{M}\log p(y|\bar w^{(j)},\bar g^{(j)}\,;\sigma_y^2), \tag{5.10}$$

$$\hat\sigma_v^{2\,(k+1)} = \arg\max_{\sigma_v^2} \frac{1}{M}\sum_{j=1}^{M}\log p(v|\bar w^{(j)}\,;\sigma_v^2), \tag{5.11}$$

$$\hat\rho^{(k+1)} = \arg\max_\rho \frac{1}{M}\sum_{j=1}^{M}\log p(\bar g^{(j)}\,;\rho), \tag{5.12}$$

$$\hat\theta^{(k+1)} = \arg\max_\theta \frac{1}{M}\sum_{j=1}^{M}\log p(\bar w^{(j)}\,;\theta). \tag{5.13}$$

Notice that all these distributions are Gaussian (Equations (4.3)–(4.6)). The problems (5.10) and (5.11) have closed form solutions:

$$\hat{\sigma}_y^{(k+1)} = \frac{1}{N\,M}\sum_{j=1}^{M} \|y - \hat{W}^{(j)}\hat{g}^{(j)}\|^2,$$

$$\hat{\sigma}_v^{(k+1)} = \frac{1}{N\,M}\sum_{j=1}^{M} \|v - \hat{w}^{(j)}\|^2.$$

The problems (5.12) and (5.13), typically, do not have closed form solutions. We can write the maximizations explicitly as

$$\hat{\rho}^{(k+1)} = \arg\min_{\rho} \frac{1}{M}\sum_{j=1}^{M} \big(\bar{g}^{(j)} - \mu_g(\rho)\big)^T K_g(\rho)^{-1}\big(\bar{g}^{(j)} - \mu_g(\rho)\big) + \log\det K_g(\rho),$$

$$\hat{\theta}^{(k+1)} = \arg\min_{\theta} \frac{1}{M}\sum_{j=1}^{M} \big(\bar{w}^{(j)} - \mu_w(\theta)\big)^T K_w(\theta)^{-1}\big(\bar{w}^{(j)} - \mu_w(\theta)\big) + \log\det K_w(\theta).$$

With the updated hyperparameters, we can run Algorithm 1 again to get new samples from the posterior. With these new samples, we can update $\hat{Q}$ and proceed with the iteration. We then iterate until some convergence criteria are met. Typically, the EM method is stopped once the increase in the likelihood drops below a certain threshold, or when the relative change in the parameters is below a certain threshold (McLachlan and Krishnan, 2007).

The whole Monte Carlo expectation-maximization method for the general uncertain-input system-identification problem is given in Algorithm 2.

In the next section, we will see how this method can be specialized when dealing with particular classes of uncertain-input models.

In the next chapter we will present applications of this method to Hammerstein systems (see Section 6.3) and to cascaded systems (see Section 6.4).

5.3 Special Classes

We now introduce four special classes of uncertain-input models that we have encountered in applications. These special classes arise from the general uncertain-input model by introducing additional assumptions.

5.3.1 Parametric input model, parametric system model

If we reduce the variance of the stochastic models of the input and of the system to zero, we can imagine the prior densities becoming infinitely peaked Dirac delta distributions. In this case, all variability in the models is removed, and we are left

Algorithm 2 Estimate the impulse response g and the input w from the data y and v using Monte Carlo EM with Gibbs sampling from the posterior.

input $y,\ v$
parameters $\mu_g(\rho),\ K_g(\rho),\ \mu_w(\rho),\ K_w(\rho)$
output $\hat{g},\ \hat{w}$
initialize $\hat{\rho},\ \hat{\theta},\ \hat{\sigma}^2$
while not converged **do**
  $\{\bar{g}^{(j)}\}_{j=1}^{M}, \{\bar{w}^{(j)}\}_{j=1}^{M} \leftarrow$ **call** PosteriorGS with $y, v, \hat{\rho}, \hat{\theta}, \hat{\sigma}^2$
  $\hat{\sigma}_y^2 \leftarrow \frac{1}{MN} \sum_{j=1}^{M} \|y - \hat{W}^{(j)} \hat{g}^{(j)}\|^2$
  $\hat{\sigma}_v^2 \leftarrow \frac{1}{MN} \sum_{j=1}^{M} \|v - \hat{w}^{(j)}\|^2$
  $\hat{\rho} \leftarrow \arg\max_\rho \frac{1}{M} \sum_{j=1}^{M} \log \mathcal{N}(\bar{g}^{(j)}\,; \mu_g(\rho), K_g(\rho))$
  $\hat{\theta} \leftarrow \arg\max_\theta \frac{1}{M} \sum_{j=1}^{M} \log \mathcal{N}(\bar{w}^{(j)}\,; \mu_w(\theta), K_w(\theta))$
$\{\bar{g}^{(j)}\}_{j=1}^{M}, \{\bar{w}^{(j)}\}_{j=1}^{M} \leftarrow$ **call** PosteriorGS with $y, v, \hat{\rho}, \hat{\theta}, \hat{\sigma}^2$
$\hat{g} \leftarrow \frac{1}{M} \sum_{j=1}^{M} \bar{g}^{(j)}$
$\hat{w} \leftarrow \frac{1}{M} \sum_{j=1}^{M} \bar{w}^{(j)}$

with classical parametric models. Symbolically, the operation we are making is

$$\begin{aligned} p(g\,;\rho) &\to \delta(g - \mu_g(\rho)), \qquad \text{as} \qquad K(\rho) \to 0, \\ p(w\,;\theta) &\to \delta(w - \mu_w(\theta)), \qquad \text{as} \qquad K(\theta) \to 0. \end{aligned}$$

If we look at the posterior-mean estimates of g and w, we see that the posteriors collapse into impulsive distributions centered around the prior means. First, we notice that

$$\begin{aligned} p(y, v\,;\rho, \theta, \sigma^2) &= \int p(y, v|g, w\,;\sigma^2) p(g\,;\rho) p(w\,;\theta) \mathrm{d}g \mathrm{d}w \\ &= \int p(y, v|g, w\,;\sigma^2) \delta\big(g - \mu_g(\theta)\big) \delta\big(w - \mu_w(\theta)\big) \mathrm{d}g \mathrm{d}w \\ &= p(y, v|\mu_g(\rho), \mu_w(\theta)\,;\sigma^2). \end{aligned}$$

The marginal likelihood of the data is equal to the distribution of the data conditioned on the events $g = \mu_g(\theta)$ and $w = \mu_w(\theta)$. This distribution is Gaussian and available in closed form:

$$\begin{aligned} &\log p(y, v|\mu_g(\rho), \mu_w(\theta)\,;\sigma^2) = \log p(y|\mu_g(\rho), \mu_w(\theta)\,;\sigma_y^2) + \log p(v|\mu_w(\theta)\,;\sigma_v^2) \\ &= -\frac{1}{2\sigma_y^2} \big\| y - \mathbf{T}_{N\times n}\big(\mu_w(\theta)\big) \mu_g(\rho) \big\|^2 - \frac{N}{2} \log \sigma_y^2 - \frac{1}{2\sigma_v^2} \|v - \mu_w(\theta)\|^2 - \frac{N}{2} \log \sigma_v^2. \end{aligned}$$

Maximizing this distribution gives the hyperparameter estimates $\hat{\rho}$, $\hat{\theta}$ and $\hat{\sigma}^2$.

Algorithm 3 Estimate the impulse response g and the input w from the data y and v using parametric models for the input and the linear system

1: **input** $y,\ v$
2: **parameters** $\mu_g(\rho),\ \mu_w(\rho)$
3: **output** $\hat{g},\ \hat{w}$
4: **initialize** $\hat{\rho},\ \hat{\theta},\ \hat{\sigma}^2$
5: $\hat{\rho}, \hat{\theta}, \hat{\sigma}^2 \leftarrow \arg\max_{\rho,\theta,\sigma^2} \log p(y, v\,|\mu_g(\rho), \mu_w(\theta); \sigma^2)$
6: $\hat{g} \leftarrow \mu_g(\hat{\rho})$
7: $\hat{w} \leftarrow \mu_w(\hat{\theta})$

When we turn to look at the posteriors, we see that, after plugging in the hyperparameter estimates,

$$\begin{aligned} p(w|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2) &= \frac{1}{p(y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2)} \int p(y, v|g, w\,; \hat{\rho}, \hat{\sigma}^2) p(g\,; \hat{\rho}) p(w\,; \hat{\theta}) \mathrm{d}g \\ &= \frac{\delta\big(w - \mu(\hat{\theta})\big)}{p(y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2)} \int p(y, v|g, w\,; \hat{\rho}, \hat{\sigma}^2) \delta\big(g - \mu_g(\hat{\rho})\big) \mathrm{d}g \\ &= \frac{p(y, v|\mu_g(\hat{\rho}), w\,; \hat{\sigma}^2) \delta\big(w - \mu_w(\hat{\theta})\big)}{p(y, v|\mu_g(\hat{\rho}), \mu_w(\hat{\theta})\,; \hat{\sigma}^2)} = \delta\big(w - \mu(\hat{\theta})\big) \end{aligned}$$

and, with exactly the same reasoning,

$$p(g|y, v\,; \hat{\rho}, \hat{\theta}, \hat{\sigma}^2) = \delta\big(g - \mu_g(\hat{\theta})\big).$$

From these considerations it follows that the posterior-mean estimators degenerate into the prior means, with the parameters tuned by maximum (marginal) likelihood. In other words, we have recovered the classical maximum-likelihood estimation method where we use the parametric models $\mu_g(\rho)$ and $\mu_w(\theta)$:

$$\begin{aligned} \hat{g} &= \int g \delta\big(g - \mu_g(\hat{\theta})\big) \mathrm{d}g = \mu_g(\hat{\rho}), \\ \hat{w} &= \int w \delta\big(g - \mu_g(\hat{\theta})\big) \mathrm{d}w = \mu_w(\hat{\theta}), \\ \hat{\rho}, \hat{\theta}, \hat{\sigma}^2 &= \arg\max_{\rho,\theta,\sigma^2} \log p(y, v\,|\mu_g(\rho), \mu_w(\theta); \sigma^2). \end{aligned}$$

The procedure is described in Algorithm 3.

The method to identify this class of models contains, among others, the prediction-error method for LTI models (Section 6.1), and maximum likelihood for errors in variables models (Section 6.6). Parametric identification of Hammerstein systems (see Ljung, Singh, et al., 2009) can also be considered under this case.

5.3.2 Parametric input model, Gaussian system model

If we reduce the variance of the input model to zero, the distribution $p(w\,;\theta)$ becomes peaked and tends to a Dirac delta distribution centered around the mean function

$$p(w\,;\theta) \to \delta(w - \mu_w(\theta)), \qquad \text{as} \qquad K(\theta) \to 0.$$

Following the collapse of the prior distribution, the marginal distribution of the data can be replaced with the distribution of the data conditioned on $w = \mu_w(\theta)$:

$$p(y, v\,;\rho, \theta, \sigma^2) = \int p(y, v|w\,;\rho, \sigma^2)p(w\,;\theta)\mathrm{d}w = p(y, v|\mu_w(\theta)\,;\rho, \sigma^2);$$

consequently, the posterior distribution reduces to the prior:

$$\begin{aligned} p(w|y, v\,;\rho, \theta, \sigma^2) &= \frac{p(y, v|w\,;\rho, \sigma^2)p(w\,;\theta)}{p(y, v\,;\rho, \theta, \sigma^2)} = \frac{p(y, v|w\,;\rho, \sigma^2)\delta\big(w - \mu_w(\theta)\big)}{p(y, v|\mu_w(\theta)\,;\rho, \sigma^2)} \\ &= \delta\big(w - \mu_w(\theta)\big), \end{aligned}$$

and the posterior mean becomes the prior mean

$$w^*(\theta) = \int w\, p(w|y, v\,;\rho, \theta, \sigma^2)\mathrm{d}w = \int w\, \delta\big(w - \mu_w(\theta)\big)\mathrm{d}w = \mu_w(\theta).$$

As for the posterior mean of g (as a function of the hyperparameters), we get that

$$\begin{aligned} g^*(\rho, \theta, \sigma^2) &= \int g\, p(g|y, v\,;\rho, \theta, \sigma^2)\mathrm{d}g \\ &= \int g\, p(g|y, w\,;\rho, \sigma^2)p(w|y, v\,;\rho, \theta, \sigma^2)\mathrm{d}g\mathrm{d}w \\ &= \int g\, p(g|y, \mu_w(\theta)\,;\rho, \sigma^2)\mathrm{d}g. \end{aligned}$$

This is very convenient, because we can calculate the posterior mean of g using the posterior density of g given y and w, which is Gaussian, and replace w with the mean value $\mu_w(\theta)$. In other words, we can use the posterior mean (4.12) replacing w with its prior mean $\mu_w(\theta)$. Therefore,

$$g^*(\rho, \theta, \sigma^2) = \left(\frac{W(\theta)^T W(\theta)}{\sigma_y^2} + K_g(\rho)^{-1}\right)^{-1} \left(\frac{W(\theta)^T}{\sigma_y^2} y + K_g(\rho)^{-1}\mu_g(\rho)\right),$$

where $W(\theta) = \mathbf{T}_{N\times n}(\mu(\theta))$. Also the marginal likelihood is Gaussian, because it corresponds to the distribution of the data conditioned on $w = \mu_w(\theta)$:

$$p(y, v|\mu_w(\theta)\,;\rho, \sigma^2) = \mathcal{N}\left(\begin{bmatrix} y \\ v \end{bmatrix}; \begin{bmatrix} W(\theta)\mu_g \\ \mu_g \end{bmatrix}, \begin{bmatrix} W(\theta)K_g W(\theta)^T + \sigma_y^2 I_N & W(\theta)K_g \\ K_g W(\theta)^T & K_g + \sigma_v^2 I_N \end{bmatrix}\right).$$

The maximization of this marginal likelihood can be simplified using the EM method. In this case, the latent variables that we introduce are the samples of the impulse response g. We define the complete likelihood

$$\begin{aligned} L_c(y, v, g\,;\rho, \theta, \sigma^2) &= \log p(y, v, g|\mu_w(\theta)\,;\rho, \sigma^2) \\ &= \log p(y|g, \mu_w(\theta)\,;\sigma_y^2) + \log p(v|\mu_w(\theta)\,;\sigma_v^2) + \log p(g\,;\rho), \end{aligned}$$

where

$$\begin{aligned} \log p(y|g, \mu_w(\theta)\,;\sigma_y^2) &= -\frac{1}{2\sigma_y^2}\|y - W(\theta)g\|^2 - \frac{N}{2}\log\sigma_y^2, \\ \log(v|\mu_w(\theta)\,;\sigma_v^2) &= -\frac{1}{2\sigma_v^2}\|v - \mu_w(\theta)\|^2 - \frac{N}{2}\log\sigma_v^2, \\ \log p(g\,;\rho) &= -\frac{1}{2}\big(g - \mu_g(\rho)\big)^T K_g(\rho)^{-1}\big(g - \mu_g(\rho)\big) - \frac{1}{2}\log\det K_g(\rho). \end{aligned}$$

We can compute the expectation of the complete likelihood with respect to the posterior distribution of g in closed form, bypassing the need for Monte Carlo integration. At any iteration of the EM method, we have

$$p\big(g|y, \mu_w(\hat{\theta}^{(k)})\,;\hat{\rho}^{(k)}, \hat{\sigma}^{2\,(k)}\big) = \mathcal{N}(g\,;m_g^{(k)}, P_g^{(k)}), \tag{5.14}$$

where (Equations (4.11) and (4.12))

$$\begin{aligned} P_g^{(k)} &= \left(\frac{\hat{W}^{(k)\,T}\hat{W}^{(k)}}{\hat{\sigma}_y^{2\,(k)}} + K_g(\hat{\rho}^{(k)})^{-1}\right)^{-1}, \\ m_g^{(k)} &= P_g\left(\frac{\hat{W}^{(k)\,T}}{\sigma_y^2}y + K_g(\hat{\rho}^{(k)})^{-1}\mu_g(\hat{\rho}^{(k)})\right), \end{aligned}$$

and $\hat{W}^{(k)} = W(\hat{\theta}^{(k)})$. With this distribution, we can calculate

$$Q(\rho, \theta, \sigma^2\,;\hat{\rho}^{(k)}, \hat{\theta}^{(k)}, \hat{\sigma}^{2\,(k)}) = \mathbf{E}\left\{L_c(y, v, g\,;\rho, \theta, \sigma^2)\right\},$$

where the expectation is taken with respect to (5.14). Thanks to the decomposition of the complete likelihood, the function Q is the sum of the following terms:

$$\mathbf{E}\left\{\log p(y|g, \mu_w(\theta)\,;\sigma_y^2)\right\} = \mathbf{E}\left\{-\frac{1}{2\sigma_y^2}\|y - W(\theta)g\|^2 - \frac{N}{2}\log\sigma_y^2\right\}$$

$$= -\frac{1}{2\sigma_y^2}\mathrm{Trace}\left\{yy^T - 2yW(\theta)m_g^{(k)} + W(\theta)\big(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T}\big)W(\theta)^T\right\} - \frac{N}{2}\log(\sigma_y^2),$$

$$\mathbf{E}\left\{\log(v|\mu_w(\theta)\,;\sigma_v^2)\right\} = -\frac{1}{2\sigma_v^2}\|v - \mu_w(\theta)\|^2 - \frac{N}{2}\log\sigma_v^2,$$

$$\mathbf{E}\left\{\log p(g\,;\rho)\right\} = -\frac{1}{2}\big(g - \mu_g(\rho)\big)^T K_g(\rho)^{-1}\big(g - \mu_g(\rho)\big) - \frac{1}{2}\log\det K_g(\rho)$$

$$= -\frac{1}{2}\mathrm{Trace}\left\{K_g(\rho)^{-1}\big(P_g^{(k)} + (m_g^{(k)} - \mu_g(\rho))(m_g^{(k)} - \mu_g(\rho))^T\big)\right\} - \frac{1}{2}\log\det K_g(\rho).$$

Algorithm 4 Estimate the impulse response g and the input w from the data y and v using a parametric model for the input

input $y,\ v$
parameters $\mu_g(\rho),\ K_g(\rho),\ \mu_w(\rho)$
output $\hat{g},\ \hat{w}$
initialize $\hat{\rho},\ \hat{\theta},\ \hat{\sigma}^2$
while not converged **do**
    $\hat{\sigma}_y^2 \leftarrow \frac{1}{N}\mathbf{E}\left\{\|y - W(\hat{\theta})g\|^2\right\}$
    $\hat{\sigma}_y^2 \leftarrow \frac{1}{N}\|v - \mu_w(\hat{\theta})\|^2$
    $\hat{\theta} \leftarrow \arg\max_\theta \mathbf{E}\left\{\log p(y|g, \mu_w(\theta)\,;\hat{\sigma}_y^2)\right\} + \log p(v|\mu_w(\theta)\,;\hat{\sigma}_v^2)$
    $\hat{\rho} \leftarrow \arg\max_\rho \mathbf{E}\left\{\log p(g\,;\rho)\right\}$
$\hat{g} \leftarrow g^*(\hat{\rho}, \hat{\theta}, \hat{\sigma}^2)$
$\hat{w} \leftarrow \mu_w(\hat{\theta})$

To solve the maximization, we can use the expectation–conditional-maximization method. We can make a first conditional maximization of Q with respect to σ^2, with the other parameters constrained to their previous values:

$$\hat{\sigma}_y^{2\,(k+1)} = \frac{1}{N}\mathrm{Trace}\left\{yy^T - 2yW(\hat{\theta}^{(k)})m_g^{(k)} + W(\hat{\theta}^{(k)})\left(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T}\right)W(\hat{\theta}^{(k)})^T\right\},$$
$$\hat{\sigma}_v^{2\,(k+1)} = \frac{1}{N}\|v - \mu_w(\hat{\theta}^{(k)})\|^2.$$

In the second conditional maximization step, we maximize Q with respect to all the other parameters, with σ^2 fixed at the values found in the first conditional maximization step:

$$\begin{aligned}\hat{\rho}^{(k+1)} &= \arg\max_\rho \mathbf{E}\left\{\log p(g\,;\rho)\right\},\\ \hat{\theta}^{(k+1)} &= \mathbf{E}\left\{\log p(y|g, \mu_w(\theta)\,;\hat{\sigma}_y^{2\,(k+1)})\right\} + \mathbf{E}\left\{\log(v|\mu_w(\theta)\,;\hat{\sigma}_v^{2\,(k+1)})\right\}.\end{aligned} \tag{5.15}$$

The constraints used are space filling, because the gradients are linearly independent. This means that if the maximizations are unique, the method will converge to a stationary point of the likelihood (see Theorem 3.2.5). With the estimated hyperparameters $\hat{\rho}$, $\hat{\theta}$, and $\hat{\sigma}^2$, we can find the estimates of the impulse response and of the input as

$$\hat{g} = g^*(\hat{\rho}, \hat{\theta}, \hat{\sigma}^2),$$
$$\hat{w} = \mu_w(\hat{\theta}).$$

The whole procedure is presented in Algorithm 4.

The case when the input model is linearly parameterized is particularly interesting;

$$\mu_w(\theta) = H\theta,$$

where θ is an unknown p dimensional vector of parameters and H is a known N by p matrix. In this case, we can write the update of the input-model parameters θ in closed form. Notice that, if we define $G = \mathbf{T}_{N\times N}(g)$, we can use the equivalence

$$W(\theta)g = \mathbf{T}_{N\times n}(\mu_w(\theta))g = \mathbf{T}_{N\times N}(g)\mu_w(\theta) = G\mu_w(\theta)$$

to write

$$\mathbf{E}\left\{\log p(y|g,\mu_w(\theta)\,;\sigma_y^2)\right\} = \mathbf{E}\left\{-\frac{1}{2\sigma_y^2}\|y - GH\theta\|^2 - \frac{N}{2}\log\sigma_y^2\right\} =$$
$$= -\frac{1}{2\sigma_y^2}\left(y^T y - 2y^T M_g^{(k)} H\theta + \theta^T H^T A^{(k)} H\theta\right) - \frac{N}{2}\log(\sigma_y^2),$$

where we have defined $M_g^{(k)} = \mathbf{E}\{G\} = \mathbf{T}_{N\times N}(\mathbf{E}\{g\}) = \mathbf{T}_{N\times N}(m_g^{(k)})$ and $A^{(k)} = \mathbf{E}\left\{G^T G\right\}$. This expression, together with

$$\mathbf{E}\left\{\log(v|\mu_w(\theta)\,;\sigma_v^2)\right\} = -\frac{1}{2\sigma_v^2}\|v - H\theta\|^2 - \frac{N}{2}\log\sigma_v^2,$$

tells us that the cost function of the second conditional step (5.15) is quadratic in θ, and that

$$\hat{\theta}^{(k+1)} = \left(\frac{1}{\hat{\sigma}_y^{2\,(k)}} H^T A^{(k)} H + \frac{1}{\hat{\sigma}_v^{2\,(k)}} H^T H\right)^{-1} \left(\frac{1}{\hat{\sigma}_y^{2\,(k)}} H^T M_g^{(k)\,T} y + \frac{1}{\hat{\sigma}_v^{2\,(k)}} H^T v\right).$$

This method has many applications: we apply it to Hammerstein models with parametric input nonlinearity (see Section 6.3), to blind system identification (see Section 6.5), to the estimation of initial conditions (see Section 6.8) and to the estimation of models with missing data (see Section 6.7).

5.3.3 Gaussian input model, parametric system model

This class is the dual of the one in the previous section, and the method works exactly the same, modulo swapping places of every g and w, and of every ρ and θ.

In this case the posterior mean of the system impulse response reduces to the prior mean, and the posterior mean of the input is available in closed form.

We reintroduce in the marginal likelihood w as a latent variable and we take the expectations with respect to the distribution

$$p(w|y, v\,;\hat{\rho}^{(k)}, \hat{\theta}^{(k)}, \hat{\sigma}^{2\,(k)}) = \mathcal{N}(w\,;\hat{m}_w^{(k)}, \hat{P}_w^{(k)}),$$

where

$$\hat{m}_w^{(k)} = \hat{\mu}_w + \begin{bmatrix}\hat{K}_w\hat{G}^T & \hat{K}_w\end{bmatrix}\begin{bmatrix}\hat{G}\hat{K}_w\hat{G}^T + \hat{\sigma}_y^{2\,(k)} I_N & \hat{G}\hat{K}_w \\ \hat{K}_w\hat{G}^T & \hat{K}_w + \hat{\sigma}_v^{2\,(k)} I_N\end{bmatrix}^{-1}\begin{bmatrix}y - \hat{G}\hat{\mu}_w \\ v - \hat{\mu}_w\end{bmatrix},$$

$$\hat{P}_w^{(k)} = \hat{K}_w - \begin{bmatrix}\hat{K}_w\hat{G}^T & \hat{K}_w\end{bmatrix}\begin{bmatrix}\hat{G}\hat{K}_w\hat{G}^T + \hat{\sigma}_y^{2\,(k)} I_N & \hat{G}\hat{K}_w \\ \hat{K}_w\hat{G}^T & \hat{K}_w + \hat{\sigma}_v^{2\,(k)} I_N\end{bmatrix}^{-1}\begin{bmatrix}\hat{G}\hat{K}_w \\ \hat{K}_w\end{bmatrix},$$

Algorithm 5 Estimate the impulse response g and the input w from the data y and v using a parametric model for the system

input $y,\ v$
parameters $\mu_g(\rho),\ K_g(\rho),\ \mu_w(\rho)$
output $\hat{g},\ \hat{w}$
initialize $\hat{\rho},\ \hat{\theta},\ \hat{\sigma}^2$
while not converged **do**
    $\hat{\sigma}_y^2 \leftarrow \frac{1}{N}\mathbf{E}\left\{\|y - \mathbf{T}_{N\times N}(\mu_g(\hat{\rho}))w\|^2\right\}$
    $\hat{\sigma}_y^2 \leftarrow \frac{1}{N}\mathbf{E}\left\{\|v - w\|^2\right\}$
    $\hat{\rho} \leftarrow \arg\min_\rho \mathbf{E}\left\{\|y - \mathbf{T}_{N\times N}(w)\mu_g(\rho)\right\}\|^2$
    $\hat{\theta} \leftarrow \arg\max_\theta \mathbf{E}\left\{\log p(w\,;\theta)\right\}$
$\hat{g} \leftarrow \mu_g(\hat{\rho})$
$\hat{\mu}_w \leftarrow \mu_w(\hat{\theta}), \qquad \hat{K}_w \leftarrow K_w(\hat{\theta}), \qquad \hat{G} \leftarrow \mathbf{T}_{N\times N}(\mu_g(\hat{\theta}))$
$\hat{w} \leftarrow \hat{\mu}_w + \begin{bmatrix}\hat{K}_w\hat{G}^T & \hat{K}_w\end{bmatrix}\begin{bmatrix}\hat{G}\hat{K}_w\hat{G}^T + \hat{\sigma}_y^2 I_N & \hat{G}\hat{K}_w \\ \hat{K}_w\hat{G}^T & \hat{K}_w + \hat{\sigma}_v^2\end{bmatrix}^{-1}\begin{bmatrix}y - \hat{G}\hat{\mu}_w \\ v - \hat{\mu}_w\end{bmatrix}$

and where

$$\hat{\mu}_w = \mu_w(\hat{\theta}^{(k)}), \quad \hat{K}_w = K_w(\hat{\theta}^{(k)}), \quad \hat{G} = \mathbf{T}_{N\times N}(\mu_g(\hat{\theta}^{(k)})).$$

We can do the same kind of consideration we did in the previous section to arrive to the expectation–conditional-maximization algorithm presented in Algorithm 5.

5.3.4 Estimated input, Gaussian system model

In this class of methods, similarly to the general case, we use a Gaussian description for the input and for the system. However, in this case we suppose that he have some reliable way to estimate the input.

If have a reliable estimate $\hat{w}$ of w—in the sense that our confidence in the estimate is very high—we can suppose that the posterior distribution of w given the data is sharply peaked around this estimate

$$p(w|y, v\,;\rho, \theta, \sigma) \approx \delta(w - \hat{w}).$$

With this in mind, we can calculate the posterior mean of g without using the Gibbs sampler, in a similar way to what we did for the parametric input case in

Algorithm 6 Estimate the impulse response g from the data y and v using an estimated input $\hat{w}$.

1: **input** $y,\ v,\ \hat{w}$
2: **parameters** $\mu_g(\rho),\ K_g(\rho)$
3: **output** $\hat{g}$
4: **initialize** $\hat{\rho},\ \hat{\sigma}^2$
5: **while** not converged **do**
6: $\quad \hat{\sigma}_y^2 \leftarrow \frac{1}{N}\mathbf{E}\left\{\|y - \mathbf{T}_{N\times n}(\hat{w})g\|\right\}$
7: $\quad \hat{\rho} \leftarrow \arg\max_\rho \mathbf{E}\left\{\log p(g\,;\rho)\right\}$
8: $\hat{\sigma}_v^2 \leftarrow \frac{1}{N}\|v - \hat{w}\|^2$
9: $\hat{g} \leftarrow g^*(\hat{\rho}, \hat{\theta}, \hat{\sigma}^2)$

Section 5.3.2:

$$\begin{aligned} g^*(\rho, \hat{w}, \sigma^2) &= \int gp(g, w|y, v\,;\rho, \theta, \sigma^2)\mathrm{d}g\mathrm{d}w \\ &= \int gp(g|y, w\,;\rho, \sigma^2)p(w|y, v\,;\rho, \theta, \sigma^2)\mathrm{d}g\mathrm{d}w \\ &= \int gp(g|y, w\,;\rho, \sigma^2)\delta(w - \hat{w})\mathrm{d}g\mathrm{d}w \\ &= \int gp(g|y, \hat{w}\,;\rho, \sigma^2); \end{aligned}$$

so, we can use the posterior mean estimate (4.12), replacing the input w with the estimated input $\hat{w}$. To calculate the marginal likelihood estimates of the hyperparameters, we observe that this method is essentially equivalent to the parametric input method in Section 5.3.2 with a fixed $\mu_w(\theta) = \hat{w}$. The whole procedure is presented in Algorithm 6.

This method can be used whenever we have some reliable way to estimate the input signal. There are various ways to obtain input estimates; for example, we could have performed a measurement experiment only on the input and collected additional input data that we used to compute the estimate $\hat{w}$.

One possible way to obtain estimates of the input from the available data is to set up a joint maximum-a-posteriori–maximum-likelihood criterion (joint MAP–ML; see Yeredor, 2000) to estimate w and the system hyperparameters:

$$\begin{aligned} \hat{w}, \hat{\rho}, \hat{\sigma}^2 &= \arg\max_{w,\rho} p(y, v, w\,;\rho, \sigma^2) \\ &= \arg\max_{w,\rho} p(y, v|w\,;\rho, \sigma^2)p(w\,;\hat{\theta}), \end{aligned} \tag{5.16}$$

where $\hat{\theta}$ is some fixed value of the input hyperparameters. What makes the joint MAP–ML criterion very interesting is that it allows us to bypass the need for the Gibbs sampler while still allowing for probabilistic models for the input.

To compute the joint MAP–ML estimates (5.16) we can use the ECM method. To this end, we introduce the impulse response samples in the criterion and we compute the complete likelihood

$$L_c(y, v, g, w\,;\rho, \sigma^2) = \log p(y|g, w\,;\sigma_y^2) + \log p(v|w\,;\sigma_v^2) + \log p(w\,;\hat{\theta}) + \log p(g\,;\hat{\rho});$$

this time, ρ, σ^2, and w act as unknown parameters to be estimated. We can then use the posterior distribution of g, for fixed hyperparameters $\hat{\rho}^{(k)}$, and $\hat{\sigma}^{2\,(k)}$, to compute the function Q

$$Q(\rho, w, \sigma^2\,;\hat{\rho}^{(k)}, \hat{w}^{(k)}, \hat{\sigma}^{2\,(k)}) = \mathbf{E}\left\{L_c(y, v, g, w\,;\rho, \sigma^2)\right\},$$

where the expectation is taken with respect to the posterior distribution

$$p(g|y, \hat{w}\,;\hat{\rho}^{(k)}, \hat{\sigma}^{2\,(k)}).$$

Using the decomposition of the complete likelihood, we see that the Q function is the sum of the following terms:

$$\mathbf{E}\left\{\log p(y|g, w\,;\sigma_y^2)\right\} = \mathbf{E}\left\{-\frac{1}{2\sigma_y^2}\|y - Wg\|^2 - \frac{N}{2}\log\sigma_y^2\right\}$$

$$= -\frac{1}{2\sigma_y^2}\mathrm{Trace}\left\{yy^T - 2yWm_g^{(k)} + W\big(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T}\big)W^T\right\} - \frac{N}{2}\log\sigma_y^2,$$

$$\mathbf{E}\left\{\log(v|w\,;\sigma_v^2)\right\} = -\frac{1}{2\sigma_v^2}\|v - w\|^2 - \frac{N}{2}\log\sigma_v^2,$$

$$\mathbf{E}\left\{\log p(g\,;\rho)\right\} = -\frac{1}{2}\big(g - \mu_g(\rho)\big)K_g(\rho)^{-1}\big(g - \mu_g(\rho)\big)^T - \frac{1}{2}\log\det K_g(\rho)$$

$$= -\frac{1}{2}\mathrm{Trace}\left\{K_g(\rho)^{-1}\big(P_g^{(k)} + (m_g^{(k)} - \mu_g(\rho))(m_g^{(k)} - \mu_g(\rho))^T\big)\right\} - \frac{1}{2}\log\det K_g(\rho),$$

$$\mathbf{E}\left\{\log p(w\,;\hat{\theta})\right\} = -\frac{1}{2}(w - \mu_w(\hat{\theta}))^T K_w(\hat{\theta})^{-1}(w - \mu_w(\hat{\theta})) - \frac{1}{2}\log\det K_w(\hat{\theta}),$$

where $W = \mathbf{T}_{N\times n}(w)$, and where $m_g^{(k)}$ and $P_g^{(k)}$ are the posterior mean and covariance at the kth iteration given by (4.12) and (4.11).

In the first conditional-maximization step, we find the updated noise variances, according to

$$\hat{\sigma}_y^{2\,(k+1)} = \frac{1}{N}\mathrm{Trace}\left\{yy^T - 2y\hat{W}^{(k)}m_g^{(k)} + \hat{W}^{(k)}\big(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T}\big)\hat{W}^{(k)\,T}\right\},$$

$$\hat{\sigma}_v^{2\,(k+1)} = \frac{1}{N}\|v - \hat{w}^{(k)}\|^2,$$

where $\hat{W}^{(k)} = \mathbf{T}_{N\times n}(\hat{w}^{(k)})$. In the second conditional maximization step, we find the updated input estimate and system hyperparameters. Notice that the maximization

Algorithm 7 Estimate the impulse response g and from the data y and v using joint MAP–ML estimates of ρ and w

input $y,\ v$
parameters $\mu_g(\rho),\ K_g(\rho),\ \mu_w(\theta),\ K_w(\theta)$
output $\hat{g},\ \hat{w}$
initialize $\hat{\rho},\ \hat{\theta},\ ,\hat{w},\ \hat{\sigma}^2$
while not converged **do**
    $\hat{\sigma}_y^2 \leftarrow \frac{1}{N}\mathbf{E}\left\{\|y - \mathbf{T}_{N\times n}(\hat{w})g\|^2\right\}$
    $\hat{\sigma}_y^2 \leftarrow \frac{1}{N}\|v - \hat{w}\|^2$
    $\hat{w} \leftarrow \arg\max_\theta \mathbf{E}\left\{\log p(y|g, w\,; \hat{\sigma}_y^2)\right\} + \log p(v|w\,; \hat{\sigma}_v^2) + \log p(w\,; \hat{\theta})$
    if model update needed **then**
        update $\hat{\theta}$
    $\hat{\rho} \leftarrow \arg\max_\rho \mathbf{E}\left\{\log p(g\,; \rho)\right\}$
$\hat{g} \leftarrow g^*(\hat{\rho}, \hat{w}, \hat{\sigma}^2)$

splits in two independent problems. Thanks to the quadratic form, the update of w is available in closed form:

$$\hat{w}^{(k+1)} = \left(A^{(k)} + \hat{\gamma}^{(k+1)} K_w(\hat{\theta})^{-1}\right)^{-1} \left(M_g^{(k)\,T} y + \hat{\gamma}^{(k+1)} K_w(\hat{\theta})^{-1} \mu_w(\hat{\theta})\right),$$

where we have defined $M_g^{(k)} = \mathbf{T}_{N\times N}(m_g^{(k)})$ and $A^{(k)} = \mathbf{E}\left\{\mathbf{T}_{N\times N}(g)^T \mathbf{T}_{N\times N}(g)\right\}$, and where $\hat{\gamma}^{(k+1)} = \hat{\sigma}_y^{2\,(k+1)}/\hat{\sigma}_v^{2\,(k+1)}$. The update for ρ is given in the maximization

$$\hat{\rho}^{(k+1)} = \arg\max_\rho \mathbf{E}\left\{\log p(g\,; \rho)\right\}.$$

In this case the constraints are space filling, because the gradients are linearly independent. This means that the iterations will converge to a stationary point of the likelihood if the maximizations at each iteration are unique. The whole procedure is presented in Algorithm 7. To set the model parameter $\hat{\theta}$ we can use, for instance, cross validation. Alternatively we can update the model parameter according to some other criterion, during the ECM iterations.

When using cross validation, we create a grid of possible values of the input-model parameter θ. Then we split the available data in two parts, a training set and a validation set. We estimate a model for each value of θ in the grid using the training-set data, and then we compute the performance of the model using the validation data. The input-model parameter that gives the highest performance is chosen. Then, the model is estimated again using the complete dataset (see Hastie et al., 2009, Section 7.1).

As an alternative to cross validation, we can use the following model selection criterion. Suppose that we have the true impulse response g, then we could use a marginal-likelihood criterion to estimate θ,

$$\theta^\star = \arg\max_\theta p(y, v, g\,; \theta, \sigma^2),$$

where $p(y, v, g\,;\theta, \sigma^2)$ is found by marginalizing over w in the joint likelihood (4.9). This distribution is not Gaussian; however, we see that

$$p(y, v, g\,;\theta, \rho, \sigma^2) = p(y|g\,;\theta, \sigma_y^2)p(v\,;\theta, \sigma_v^2)p(g\,;\rho)$$

and we can maximize the product of the two Gaussian distributions (4.16) and (4.17).

Because we do not have the true value of the impulse response g, to use this criterion in practice we would use the joint MAP–ML criterion to estimate the impulse response[1] $g(\theta)$ for the values of θ in a grid, and then choose the value such that

$$\hat{\theta} = \arg\max_{\theta} p(y|g(\theta)\,;\theta, \sigma_y^2)p(v\,;\theta, \sigma_v^2).$$

Alternatively, we can use this criterion during the EM iterations. At any iteration of the EM method, we can compute an estimate $\hat{g}^{(k)}$ of the impulse response using the current values of the hyperparameters and the input, for some input model $\hat{\theta}^{(k)}$. Then, we can update the input model by using

$$\hat{\theta}^{(k+1)} = \arg\max_{\theta} p(y|\hat{g}^{(k)}\,;\theta, \sigma_y^2)p(v\,;\theta, \sigma_v^2). \tag{5.17}$$

In Section 6.3, we compare these different model selection criteria for the estimation of Hammerstein systems.

[1] with $g(\theta)$, we underline the dependence of the estimate on the value of θ.

Chapter 6

Applications

In the previous chapter, we presented an empirical Bayes method for the identification of a quite general type of model, the uncertain-input model. We also saw how the general structure can be simplified by introducing additional assumptions on the Gaussian processes describing the input or the system. In this chapter, we will see how we can recover some classical applications in system identification as particular types of uncertain-input models.

6.1 Classical PEM

The first application we will consider is the prediction-error method (PEM). PEM has a long history; for a complete treatment, see the classical books Söderström and Stoica (1988), and Ljung (1999).

In the simplest formulation of PEM, we are given a set of data, measurements of the input u_t and the output y_t of some dynamical system. The system is affected by random noise ε_t. Figure 6.1 shows a block diagram of the setup.

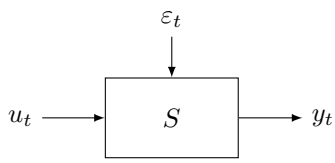

Figure 6.1: The linear system considered in the classical prediction-error method.

The system S is a linear time-invariant dynamical system described, in discrete time, by a difference equation. To easily fit in the framework we are using, we consider the *output-error* transfer-function model,

$$y_t = \frac{B(q\,;\rho)}{F(q\,;\rho)} u_t + \varepsilon_t, \qquad \textbf{(OE)}$$

where $B(q\,;\rho)$ and $F(q\,;\rho)$ are polynomials of known order (parameterized by the vector ρ) in the one step forward operator q,

$$u_t = q\,u_{t-1},$$

and ε_t is Gaussian white noise of unknown variance.

Given this model and the data up to a certain time $t-1$, we can predict the value of the next sample of the output. In other words, we can create a *predictor of the output*:

$$\hat{y}_t(\rho) = \frac{B(q\,;\rho)}{F(q\,;\rho)}u_t. \qquad \textbf{(OE predictor)}$$

We will assume that the model transfer function and the predictor transfer function are strictly causal and asymptotically stable and that the system was at rest before the identification experiment began. The predictor can be expressed through its unknown impulse response $g_t(\rho)$ as

$$\hat{y}_t(\rho) = \sum_{k=1}^{\infty} g_k(\rho)u_{t-k}.$$

Due to the stability assumption, the impulse response decays asymptotically to zero. Therefore, we can truncate the impulse response after a large enough number of samples n. We have arrived at a model of the system of the form

$$\hat{y}_t(\rho) = \sum_{k=1}^{n} g_k(\rho)u_{t-k}.$$

PEM would then find the parameters ρ, describing the locations of the poles and the zeros in the model transfer function, that minimize the criterion

$$J_1(\rho) = \sum_{t=1}^{N} \big(y_t - \hat{y}_t(\rho)\big)^2.$$

Consider now the uncertain-input model (4.1)

$$\begin{cases} y = Wg + \varepsilon \\ v = w + \eta \\ g \sim \mathcal{N}(g\,;\mu_g(\rho), K_g(\rho)) \\ w \sim \mathcal{N}(w\,;\mu_w(\theta), K_w(\theta)) \\ \varepsilon \sim \mathcal{N}(\varepsilon\,;0, \sigma_y^2 I_N) \\ \eta \sim \mathcal{N}(\eta\,;0, \sigma_v^2 I_N) \end{cases}.$$

We make the following assumptions:

$$\begin{array}{lll} K_g(\rho) = 0, & [\mu_g(\rho)]_i = g_i(\rho), & \sigma_v^2 = +\infty, \\ K_w(\theta) = 0, & \mu_w(\theta) = u. & \end{array}$$

Then, we know from Section 5.3.1 that the estimate of the impulse response corresponds to the prior mean evaluated at the estimated parameters:

$$\hat{g}_i = g_i(\hat{\rho}).$$

The parameter estimates $\hat{\rho}$ are found maximizing the marginal likelihood

$$\hat{\rho}, \hat{\theta}, \hat{\sigma}^2 = \arg \max_{\rho,\theta,\sigma^2} \log p(y, v \,|\mu_g(\rho), \mu_w(\theta); \sigma^2).$$

Using our assumptions, we have that

$$p(y, v \,|\mu_g(\rho), \mu_w(\theta); \sigma^2) = -\frac{1}{2\sigma_y^2}\|y - \mathbf{T}_{N\times n}(u)\mu_g(\rho)\|^2 + \frac{N}{2}\log\sigma_y^2,$$

which is a function of ρ and σ_y^2 alone. In particular, we notice that the estimate of ρ is found minimizing the criterion

$$J_2(\rho) = \|y - \mathbf{T}_{N\times n}(u)\mu_g(\rho)\|^2,$$

which is equal to $J_1(\rho)$. This shows that the estimation of uncertain-input systems contains the prediction-error method with a quadratic cost function as a particular case.

6.2 Regularized FIR

In recent years, a lot of research has been made on the use of regularization for system identification. We saw in Chapter 2 how kernels can be used to penalize certain solutions when problems are ill posed. The interested reader can find in Pillonetto, Dinuzzo, et al. (2014) a survey of kernel methods in system identification. There have been many works incorporating kernels into system identification. In Chen, Ohlsson, and Ljung (2012), the authors show how simple kernels can reduce the variance of FIR estimates and increase the robustness of the estimate, especially for short data records.

In Pillonetto and De Nicolao (2010), the authors present a regularization method for estimating predictor models that can be seen as a particular example of the method for the identification of uncertain-input systems. We consider the case where the impulse response g of the system is truncated after n samples, but we do not introduce any parametrization. Instead, we attempt to estimate all the impulse-response samples from data. This is a linear least-squares problem, and the solution is the pseudoinverse of the Toeplitz matrix of the input (see Section 1.2). When the impulse response is long and the data record is short, the linear least-squares estimate may have high variance; this results in a large mean square error. With regularization, we introduce a bias and reduce the variance according to the bias-variance tradeoff principle (see Section 1.4).

This type of regularization can be recovered from the uncertain-input framework by setting

$$\begin{aligned} K_g(\rho) &= K(\rho), & \mu_g(\rho) &= 0, & \sigma_v^2 &= +\infty, \\ K_w(\rho) &= 0, & \mu_w(\theta) &= u, \end{aligned}$$

where the matrix valued function K is the *stable spline* kernel (Pillonetto and De Nicolao, 2010, also *tuned-correlated* kernel in Chen, Ohlsson, and Ljung, 2012)

$$[K(\rho)]_{i,j} = \rho_1 \, \rho_2^{\max(i,j)}, \tag{6.1}$$

where ρ_1 is a positive scaling factor and where ρ_2 is a parameter, between 0 and 1, that is related to the speed of decay of the impulse response.

Using the method in Section 5.3.2, we obtain the marginal likelihood estimator

$$\begin{aligned} \hat{\rho} &= \arg\max -\frac{1}{2} y^T \left(UK(\rho)U^T + \sigma_y^2 I\right)^{-1} y - \frac{1}{2} \log\det\left(UK(\rho)U^T + \sigma_y^2 I\right), \\ \hat{g} &= \left(\frac{U^T U}{\sigma_y^2} + \left(\hat{K}(\hat{\rho})\right)^{-1}\right)^{-1} \frac{U^T}{\sigma_y^2} y. \end{aligned}$$

To solve the marginal-likelihood maximization, we can use Algorithm **??**.

The impulse-response estimate can be seen as the solution of a regularized least-squares problem

$$J_3(g,\rho) = \frac{1}{\sigma_y^2}\|y - Ug\|^2 + g^T K(\rho)^{-1} g;$$

or as a Bayesian posterior estimate with a Gaussian prior,

$$g = \mathbf{E}\left\{g|y\right\}, \qquad g \sim \mathcal{N}\big(g\,; 0, K(\rho)\big),$$

where the hyperparameters are tuned using the marginal-likelihood criterion.

6.3 Hammerstein models

The Hammerstein model is a block-oriented nonlinear model where a static nonlinear function is followed by a linear time-invariant dynamical system (see Figure 6.2).

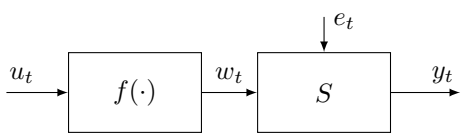

Figure 6.2: Hammerstein model structure.

Hammerstein models can be used in a vast range of applications, and have been well studied in the literature. We consider four possible ways to model Hammerstein

structures. All four can be cast into the uncertain-input framework and can be identified using the techniques shown in Chapter 5. Without further specification of the system or of the input nonlinearity, the Hammerstein system is not identifiable. In fact, given any Hammerstein model, multiplying the input by a factor α and multiplying the system gain by the inverse of α will yield another description with the same input-output behavior (see, for instance, Bai, 1998). This is one instance of the general nonidentifiability of the uncertain-input structure.

To cast the Hammerstein model into the uncertain-input framework, we start by observing that, if we call w_t the transformed input, we can see w_t as an *uncertain input*, where the uncertainty comes from the fact that we do not know the nonlinear function $f(\cdot)$. We can model the input using parametric models or nonparametric models. For the linear system, can either opt for a parametric description, where we introduce some model of the linear system in the form of an output-error or FIR model, or we can choose a nonparametric description in the form of a Gaussian prior—that is, a kernel function.

In all cases, we can use the techniques detailed in the previous chapter to estimate the parameters, and hyperparameters, of the model.

Say that we use a nonparametric description of the nonlinear function $f(\cdot)$. We can then define a Gaussian process with a kernel $K_1(\cdot, \cdot\,; \theta)$ suitable for functional estimation; one common example is the Gaussian kernel (see, for instance, Murphy, 2012, Section 14.2.1)

$$K_1(x, y\,; \theta) = \exp\left[-\frac{1}{\theta}(x - y)^2\right],$$

where the parameter θ (called *kernel width*) is linked to the smoothness of the function.

The vector of samples w, evaluations of $f(\cdot)$ at the input locations u, will be a Gaussian vector with zero mean and covariance matrix $K_1(\theta)$:

$$w \sim \mathcal{N}(w\,; 0, K_1(\theta)), \qquad [K_1(\theta)]_{i,j} = K_1(u_i, u_j\,; \theta).$$

Notice that the covariance matrix has no scaling factor, because of the nonidentifiability of Hammerstein systems.

To describe the linear part of the system, we can use a nonparametric description. To this end, we define a Gaussian process for g with a covariance matrix chosen among the various kernels proposed for linear system estimation. In this example we use the stable spline kernel (6.1); so, we define the matrix valued function

$$[K_2(\rho)]_{i,j} = \rho_2^{\max(i,j)}.$$

With this choice, the impulse-response samples have a joint Gaussian distribution:

$$g \sim \mathcal{N}(g\,; 0, \rho_1 K_2(\rho_2)).$$

We can see the nonparametric Hammerstein problem as an uncertain-input problem with

$$\begin{aligned} K_g(\rho) &= \rho_1 K_2(\rho_2), & \mu_g(\rho) &= 0, & \sigma_v^2 &= +\infty, \\ K_w(\theta) &= K_1(\theta), & \mu_w(\theta) &= 0. \end{aligned}$$

Therefore, we can set up the general method in Algorithm 2 to estimate the posterior means of g and w.

We first estimate the hyperparameters ρ, θ, and σ_y^2. We start form any hyperparameter estimate $\hat{\rho}^{(k)}$, $\hat{\theta}^{(k)}$, and $\hat{\sigma}_y^{2\,(k)}$ and we set up a Gibbs sampler to sample from the joint posterior distribution of g and w with these hyperparameters. Then, we calculate the estimated Q function—that is, the expectation of the complete likelihood—with the samples from the posterior,

$$\begin{gathered} Q(\rho, \theta, \sigma_y^2 \,; \hat{\rho}^{(k)}, \hat{\theta}^{(k)}, \hat{\sigma}_y^{2\,(k)}) = \\ \frac{1}{M} \sum_{j=1}^{M} \log p(y|\bar{g}^{(j)}, \bar{w}^{(j)} \,; \sigma_y^2) + \log p(\bar{w}^{(j)} \,; \theta) + \log p(\bar{g}^{(j)} \,; \rho), \end{gathered}$$

where

$$\begin{aligned} \frac{1}{M} \sum_{j=1}^{M} \log p(y|\bar{g}^{(j)}, \bar{w}^{(j)} \,; \sigma_y^2) &= -\frac{1}{2\sigma_y^2} \left(\frac{1}{M} \sum_{j=1}^{M} \left\| y - \mathbf{T}_{N\times n}(\bar{w}^{(j)}) \bar{g}^{(j)} \right\|^2 \right) - \frac{N}{2} \log \sigma_y^2, \\ \frac{1}{M} \sum_{j=1}^{M} \log p(\bar{g}^{(j)} \,; \rho) &= -\frac{1}{2\rho_1} \mathrm{Trace} \left\{ K_2(\rho_2)^{-1} \hat{S}_g \right\} - \frac{1}{2} \log \det \rho_1 K_2(\rho_2), \\ \frac{1}{M} \sum_{j=1}^{M} \log p(\bar{w}^{(j)} \,; \theta) &= -\frac{1}{2} \mathrm{Trace} \left\{ K_1(\theta)^{-1} \hat{S}_w \right\} - \frac{1}{2} \log \det K_1(\theta), \end{aligned}$$

and where $\hat{S}_g$ and $\hat{S}_w$ are estimates of the posterior second moments of g and w:

$$\begin{aligned} \hat{S}_g &= \frac{1}{M} \sum_{j=1}^{M} \bar{g}^{(j)} \bar{g}^{(j)\,T} \approx \mathbf{E} \left\{ g g^T | y \right\}, \\ \hat{S}_w &= \frac{1}{M} \sum_{j=1}^{M} \bar{w}^{(j)} \bar{w}^{(j)\,T} \approx \mathbf{E} \left\{ w w^T | y \right\}. \end{aligned}$$

With this technique, the marginal-likelihood estimation reduces to an iterative procedure where, at each iteration, we first run a Gibbs sampler to obtain samples from the posterior and then we update the parameters with scalar optimization

problems:

$$\begin{aligned}
\hat{\sigma}_y^{2\,(k+1)} &= \frac{1}{NM}\sum_{j=1}^{M}\left\|y - \mathbf{T}_{N\times n}(\bar{w}^{(j)})\bar{g}^{(j)}\right\|^2,\\
\hat{\rho}_2^{(k+1)} &= \arg\min_{\rho_2} n\log\left(\mathrm{Trace}\left\{K_2(\rho_2)^{-1}\hat{S}_g\right\}\right) + \log\det K_2(\rho_2),\\
\hat{\rho}_1^{(k+1)} &= \frac{1}{n}\mathrm{Trace}\left\{K_2(\hat{\rho}_2^{(k+1)})^{-1}\hat{S}_g\right\},\\
\hat{\theta}^{(k+1)} &= \arg\min_{\theta} \mathrm{Trace}\left\{K_1(\theta)^{-1}\hat{S}_w\right\} + \log\det K_1(\theta).
\end{aligned} \tag{6.2}$$

If we are using a parametric description of the nonlinearity, the computation becomes easier, because we can skip the Gibbs sampling steps. Suppose that we choose a finite parametrization of the nonlinearity in the form of a combination of known basis functions:

$$f(\cdot) = \sum_{i=1}^{p}\theta_i\varphi_i(\cdot).$$

The basis functions $\varphi_i(\cdot)$ may be polynomials, Legendre polynomials, wavelets, trigonometric functions, among others. With this parametrization, we can represent the Hammerstein model as an uncertain-input model with

$$\begin{aligned}
&K_g(\rho) = \rho_1 K_2(\rho_2), && \mu_g(\rho) = 0, && \sigma_v^2 = +\infty,\\
&K_w(\theta) = 0, && \mu_w(\theta) = H\theta, &&
\end{aligned}$$

where $\rho_1 K_2(\rho_2)$ is, again, a linear-system identification kernel (for instance, the stable spline kernel) and H is a matrix of evaluations of the basis functions at the input samples:

$$[H]_{i,j} = \varphi_j(u_i).$$

With this choice, we can use Algorithm 4. We use the EM method to compute the marginal-likelihood estimates of the hyperparameters. We define the complete likelihood by reintroducing the impulse-response samples g. Taking the expectation with respect to the posterior distribution of g given the data, for a fixed value of the hyperparameters, we obtain the function Q. We find the updated hyperparameters maximizing this function:

$$\begin{aligned}
\hat{\rho}_2^{(k+1)} &= \arg\min_{\rho_2}\ n\ \log\mathrm{Trace}\left\{K_2(\rho_2)^{-1}(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T})\right\} + \log\det K_2(\rho_2),\\
\hat{\rho}_1^{(k+1)} &= \frac{1}{n}\mathrm{Trace}\left\{K_2(\hat{\rho}_2^{(k+1)})^{-1}(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T})\right\},\\
\hat{\theta}^{(k+1)} &= \left(H^T A^{(k)} H\right)^{-1} H^T \mathbf{T}_{N\times N}(m_g^{(k)})^T y,
\end{aligned} \tag{6.3}$$

where $m_g^{(k)}$ is the posterior mean of g at the kth iteration and $P_g^{(k)}$ is the posterior variance at the kth iteration (see Theorem A.2), and $A^{(k)}$ is the posterior expected

value of $\mathbf{T}_{N\times N}(g)\mathbf{T}_{N\times N}(g)^T$, given by

$$A^{(k)} = \mathbf{E}\left\{\mathbf{T}_{N\times N}(g)^T\mathbf{T}_{N\times N}(g)\right\} = \mathbf{R}^T\left[\left(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T}\right)\otimes I_N\right]\mathbf{R},$$

where

$$\mathbf{R}^T = \begin{bmatrix} I_N & S^T & S^{2\,T} & \cdots & S^{n-1\,T}\end{bmatrix}, \qquad [S]_{i,j} = \delta_{i,j+1}.$$

This procedure has an interesting link to regularization methods for overparameterized Hammerstein models. In overparameterized Hammerstein models, the parameters of the input nonlinearity and impulse-response samples are collected in a vector that contains the products of all the variables,

$$T_{N\times n}(H\theta)g = \Phi\vartheta,$$

where

$$\Phi = \begin{bmatrix} H & SH & S^2H & \cdots & S^{n-1}H\end{bmatrix}, \qquad \vartheta = g\otimes\theta,$$

and S is the lower shift matrix $[S]_{i,j} = \delta_{i,j+1}$. Then, the overparameterized vector ϑ is identified. Once the overparameterized vector has been identified, we recover the parameters θ and g by means of a reduction step based on the observation that the $n \times p$ matrix

$$\mathcal{R}(\vartheta) = \begin{bmatrix} \vartheta_1 & \cdots & \vartheta_{n(p-1)+1} \\ \vdots & & \vdots \\ \vartheta_n & \cdots & \vartheta_{np}\end{bmatrix}$$

has rank one (for details, see Bai, 1998). In Risuleo, Bottegal, and Hjalmarsson (2015b), we proposed a method for overparameterized Hammerstein system identification where we posed the problem as a Gaussian regression problem and introduced a kernel for ϑ, the *Kronecker overparametrized* kernel, that makes the reduction step superfluous. The method always returns estimates $\hat{\vartheta}$ that give $\operatorname{rank}\mathcal{R}(\hat{\vartheta}) = 1$. Furthermore, the Gaussian regression method for overparameterized Hammerstein models is equivalent to the Hammerstein method in this section, with a linear parametrization of the input nonlinearity (for details, see Risuleo, Bottegal, and Hjalmarsson, 2015b)

Numerical example

In this section, we present a numerical example that shows how the different methods outlined in the previous section perform when confronted with data from Hammerstein systems. We perform four Monte Carlo simulations. In each simulation, we randomly generate transfer-function models by sampling poles and zeroes in the complex plane. The poles are sampled (uniformly in magnitude and phase) in the annulus of radii 0.4 and 0.8. The zeros are sampled in the disk of radius 0.92.

The nonlinear transformation is given by a finite combination of Legendre polynomials, defined as

$$\varphi_j(x) = 2^j \cdot \sum_{k=0}^{j} x^k \binom{j}{k} \binom{\frac{j+k-1}{2}}{j}.$$

The coefficients of the combination are independent and chosen uniformly in $[-1, 1]$. The input is white noise, uniform in $[-1, 1]$.

Figure 6.3 shows the first 5 Legendre polynomials.

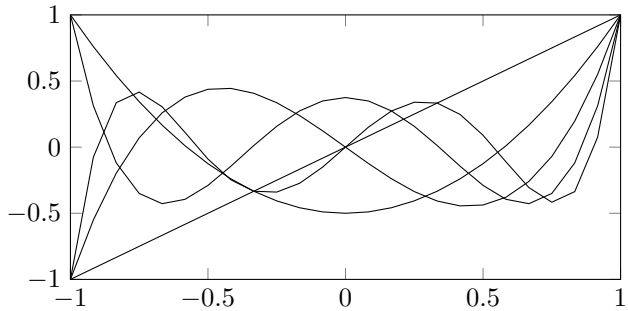

Figure 6.3: Plot of the first 5 Legendre polynomials

We consider four datasets, grouping Hammerstein systems with different orders. In Table 6.1, we present the orders of the linear blocks and of the nonlinear transformations in the datasets. In each dataset, we have generated 200 systems

Dataset	S	$f(\cdot)$
LOLO	$3 \div 5$	$5 \div 10$
HILO	$9 \div 20$	$5 \div 10$
LOHI	$3 \div 5$	$15 \div 20$
HIHI	$9 \div 20$	$15 \div 20$

Table 6.1: Hammerstein systems: orders of the systems used in the simulations

according to the specified rules.

In the simulations, we compare the following estimators:

HS-P This method uses a linearly parameterized model for the input nonlinearity. We construct the matrix H of the evaluations of the basis functions and use (6.3) to update the hyperparameters.

HS-CV This method uses the joint MAP–ML criterion in Algorithm 7 to estimate the nonlinear transformation and the system hyperparameters using (5.16).

The input nonlinearity hyperparameter $\hat{\theta}$ is tuned with cross validation on a grid of 12 values, logarithmically spaced between 0.01 and 10. For each value in the grid we estimate one model using 70% of the available data. Then the models are compared using the remaining 30% of the data. The value $\hat{\theta}$ that gives the highest predictive performance is chosen, and then the model with the best $\hat{\theta}$ is estimated again using the whole dataset.

HS-MS This method uses the joint MAP–ML criterion in Algorithm 7 to estimate the nonlinear transformation and the system hyperparameters. To choose the input-nonlinearity model it uses the following model-selection criterion derived from (5.17): at each iteration of the EM method, it chooses the model according to

$$\hat{\theta}^{(k+1)} = \arg\max_{\theta} = p(y|\hat{g}^{(k)}; \theta, \hat{\sigma}_y^{2\,(k)}). \tag{6.4}$$

HS-MCEM This method uses the general uncertain-input model estimator with Gibbs sampling from the joint posterior. It estimates the model hyperparameters using Monte Carlo EM (Algorithm 2); the explicit expressions for the parameter updates are given in (6.2).

HS-SEM This method uses the general uncertain-input model estimator with Gibbs sampling from the joint posterior. It estimates the model hyperparameters using Stochastic EM (Algorithm 2 with $M = 1$); the explicit expressions for the parameter updates are given in (6.2), where

$$\hat{S}_g = \bar{g}\bar{g}^T, \qquad \hat{S}_w = \bar{w}\bar{w}^T,$$

and $\bar{g}$ and $\bar{w}$ are single samples from the stationary distribution of the Gibbs sampler.

We evaluate the performance using the standard goodness-of-fit score

$$\text{fit}(\hat{a}, a) = 1 - \frac{\|\hat{a} - a\|}{\|a - \bar{a}\|}, \tag{6.5}$$

where a is a true value and $\hat{a}$ is an estimate; $\bar{a}$ is the mean value of a. In Figure 6.4, we see the boxplots of the simulation results for the considered datasets. We compute the fit of the estimated impulse response and of the nonlinear transformation, evaluated on an uniform grid of 300 values.

The numerical values of the median fit of the methods over the simulations are given in Table 6.2 (page 86). As a reference, we have reported the performance of the Matlab routine for the identification of Hammerstein systems (in the column marked NLHW).

From the simulations, we see that the proposed uncertain-input model can be used effectively to estimate Hammerstein systems. Our simulations confirm the intuitive idea that using more information about the input signal yields better results. In this case, the maximum information is used by the parametric method,

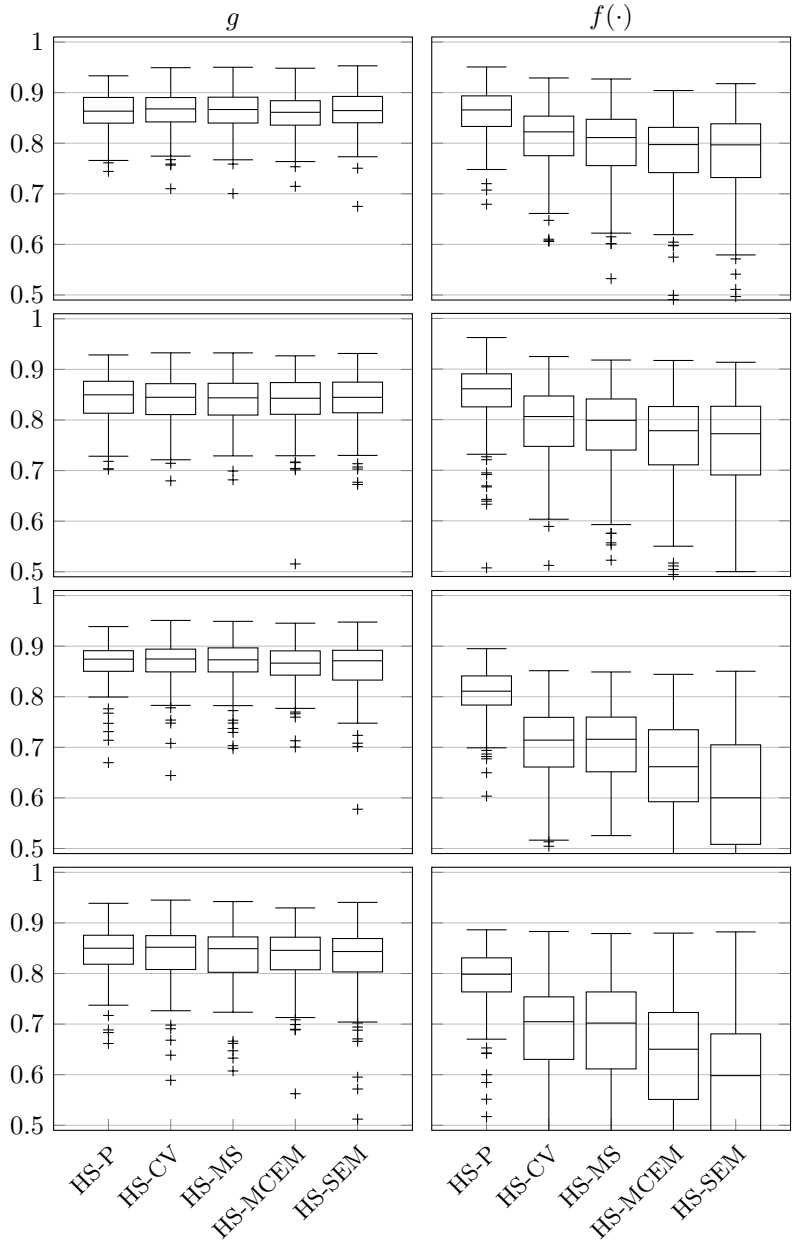

Figure 6.4: Boxplots of the estimation results in the Hammerstein Monte-Carlo experiment. The figure shows the fit of the estimated impulse responses (left column) and the fit of the estimated nonlinearity (right column) for the datasets (from top to bottom: LOLO, HILO, LOHI, HIHI).

		NLHW	HS-P	HS-CV	HS-MS	HS-MCEM	HS-SEM
g	LOLO	84.1	86.4	86.8	86.7	86.1	86.5
	HILO	40.8	85.0	84.6	84.4	84.3	84.5
	LOHI	85.5	87.4	87.5	87.3	86.7	87.2
	HIHI	40.5	85.0	85.2	84.9	84.7	84.4
$f(\cdot)$	LOLO	36.5	86.6	82.2	81.2	79.8	79.7
	HILO	16.3	86.2	80.7	80.0	77.8	77.5
	LOHI	32.6	81.1	71.5	71.6	66.2	60.0
	HIHI	10.3	79.9	70.5	70.2	65.1	60.0

Table 6.2: Median fits (in percent) of the estimated impulse response and nonlinearity in the Hammerstein Monte-Carlo simulations.

which knows the order of the nonlinearity and the shape of the basis functions. The second best performance is achieved by the joint MAP–ML with cross-validation model selection. Our proposed model-selection criterion, Equation (6.4), is also very effective, with a median fit close to the median fit of the cross-validation method. The Gibbs-sampling based methods have very good performance when dealing with the impulse response. The worse performance of the estimates of the nonlinearity (compared to the non sampling-based methods) is probably caused by problems in the convergence of the Markov chains: when sampling the nonlinearity, we are sampling a vector of many samples; because of the curse of dimensionality, this chain will be very slow to converge and many samples are needed to find the Monte Carlo average. Another possible reason could be the correlation between the variables; in the case of correlated variables, the random walk behaviour of the chain will be impaired (see Bishop, 2006, Section 11.3, see also Neal, 1998). The stochastic EM method has a performance that is close to the performance of full Monte Carlo EM method, especially for lower orders of the nonlinearity. This characteristic, paired with the much shorter running time, makes stochastic EM a valid alternative to the full-blown Monte Carlo EM method.

To show the performance of the method on hard nonlinearities, we consider a single experiment. We estimate one Hammerstein system with a linear component with 40 poles and 40 zeros, generated as discussed in the previous example. The input is a randomly generated symmetric saturation. We generate $N = 400$ samples of the output in response to a Gaussian white noise input with unit variance. The output is corrupted by Gaussian noise with variance 10% of the variance of the noiseless output.

We use the stable spline kernel to model the impulse response of the system and

we use the following two-parameter model for the input saturation:

$$[\mu_w(\theta)]_i = \begin{cases} \theta_2, & \text{if } u_i \geq \theta_2 \\ u_i, & \text{if } \theta_1 \leq u_i \leq \theta_2 \\ \theta_1, & \text{if } u_i \leq \theta_1 \end{cases}$$

We use Algorithm 4 to estimate the hyperparameters and the first $n = 50$ samples of the impulse response. The results are shown in Figure 6.5 and Figure 6.6. The figures show the estimated input nonlinearity and impulse response compared with the true ones.

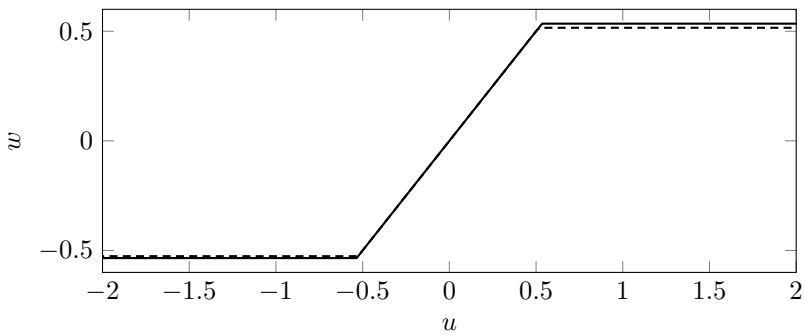

Figure 6.5: The plot shows the true input saturation (solid) and the estimated input saturation (dashed) in the Hammerstein system.

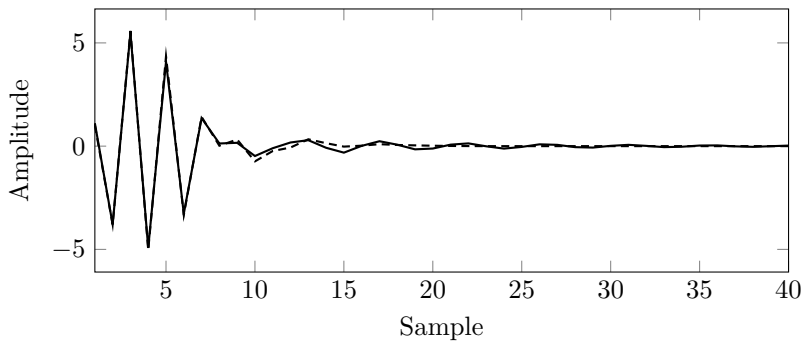

Figure 6.6: The plot shows the true impulse response (solid) and the estimated impulse response (dashed) of the linear component in the Hammerstein system.

This simulation shows that the proposed estimation method works also for hard nonlinearities; however, we need to solve a nonlinear optimization problem at each iteration of the EM method.

6.4 Cascaded linear systems

A *cascaded linear system* is a cascade composition of linear systems in which the output of one linear system becomes the input to the next linear system. We will consider a two-system cascade composed of two linear systems S_1 and S_2. The known input signal u_t is fed to the system S_1. The noiseless output w_t of the first system is measured with a measurement noise η_t; we call this measured signal v_t. The noiseless output of the first system is fed to the system S_2 and the output of the second system is measured with a measurement noise ε_t. The block schematic of the system is shown in Figure 6.7.

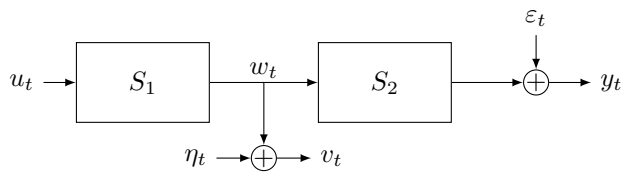

Figure 6.7: Cascaded linear systems.

We describe the linear systems with impulse-response models:

$$w_t = \sum_{k=1}^{\infty} h_j u_{t-k}, \qquad \textbf{(system } S_1\textbf{)}$$

$$y_t = \sum_{k=1}^{\infty} g_j w_{t-k} + \varepsilon_t. \qquad \textbf{(system } S_2\textbf{)}$$

We model the impulse responses with two Gaussian processes with kernel functions $K_1(\cdot,\cdot\,;\theta)$ and $K_2(\cdot,\cdot\,;\rho)$. We choose kernel functions that well represent linear system impulse responses. The vectors of impulse-response samples are Gaussian vectors, with covariance matrices given by the kernel functions;

$$h \sim \mathcal{N}(h\,;0,K_1(\theta)), \qquad g \sim \mathcal{N}(g\,;0,K_2(\rho)),$$

where

$$[K_1(\theta)]_{i,j} = K_1(i,j\,;\theta), \qquad [K_2(\rho)]_{i,j} = K_2(i,j\,;\rho).$$

Assembling the samples w_t and u_t into two vectors, we can write the convolution describing S_1 as the matrix product

$$w = \mathbf{T}_{N\times n}(u)h = Uh. \tag{6.6}$$

This convolution describes a known linear transformation between the Gaussian vector h and the vector w. Because w is a linear transformation of a Gaussian vector, it is Gaussian and

$$\begin{aligned}\mathbf{E}\{w\} &= \mathbf{E}\{Uh\} = 0,\\ \mathbf{cov}\{w\} &= \mathbf{E}\left\{(Uh)(Uh)^T\right\} = U\mathbf{E}\left\{hh^T\right\}U^T = UK_1(\theta)U^T.\end{aligned}$$

Therefore, w is a Gaussian vector with distribution

$$w \sim \mathcal{N}(w,\ 0, UK_1(\theta)U^T).$$

Considering this model for the vector w, we can describe the cascade as an uncertain-input model with

$$\begin{aligned} K_g(\rho) &= K_2(\rho), & \mu_g(\rho) &= 0, \\ K_w(\theta) &= UK_1(\theta)U^T, & \mu_w(\theta) &= 0. \end{aligned}$$

We can estimate the cascaded linear model using Algorithm 2. We estimate the hyperparameters using the marginal likelihood of the data, computed with Monte-Carlo EM. We reintroduce g and w in the marginal likelihood as nuisance parameters and we calculate the Monte Carlo approximated expectation with respect to the nuisance parameters.

At the kth iteration of the method, we set up a Gibbs sampler to draw samples from the joint posterior of w and g. We collect, after a burn-in period, the samples $\{\bar{g}^{(j)}\}_{j=1}^{M}$ and $\{\bar{w}^{(j)}\}_{j=1}^{M}$ and we set

$$\hat{Q}(\rho, \theta, \sigma\,; \hat{\rho}^{(k)}, \hat{\theta}^{(k)}, \hat{\sigma}^{2\,(k)}) = \frac{1}{M}\sum_{j=1}^{M} \log p(y, v, \bar{g}^{(j)}, \bar{w}^{(j)}\,; \rho, \theta, \sigma^2).$$

Then, maximizing $\hat{Q}$, we find the updates of the parameters according to

$$\begin{aligned} \hat{\sigma}_y^{2\,(k+1)} &= \frac{1}{M\,N}\sum_{j=1}^{M} \|y - \mathbf{T}_{N\times n}(\bar{w}^{(j)})\bar{g}^{(j)}\|^2, \\ \hat{\sigma}_v^{2\,(k+1)} &= \frac{1}{M\,N}\sum_{j=1}^{M} \|v - \bar{w}^{(j)}\|^2, \\ \hat{\rho}^{(k+1)} &= \arg\min_{\rho}\ \text{Trace}\left\{K_g(\rho)^{-1}\hat{P}_g^{(k)}\right\} + \log\det K_g(\rho), \\ \hat{\theta}^{(k+1)} &= \arg\min_{\theta}\ \text{Trace}\left\{K_w(\theta)^{-1}\hat{P}_w^{(k)}\right\} + \log\det K_w(\theta), \end{aligned}$$

where $\hat{P}_g^{(k)}$ and $\hat{P}_w^{(k)}$ are Monte Carlo estimates of the posterior second moments of g and w at the kth iteration of the algorithm:

$$\hat{P}_g^{(k)} = \sum_{j=1}^{M} \bar{g}^{(j)}\bar{g}^{(j)\,T}, \qquad \hat{P}_w^{(k)} = \sum_{j=1}^{M} \bar{w}^{(j)}\bar{w}^{(j)\,T}.$$

After convergence of the expectation-maximization iterations, we have estimates of the hyperparameters $\hat{\rho}$, $\hat{\theta}$, and $\hat{\sigma}^2$. With these, we use a Gibbs sampler again

and we calculate the posterior means of g and w, finding the minimum-variance estimates:

$$\hat{g} = \frac{1}{M}\sum_{j=1}^{M} \bar{g}^{(j)}, \qquad \hat{w} = \frac{1}{M}\sum_{j=1}^{M} \bar{w}^{(j)}.$$

The estimate $\hat{g}$ is the estimate of the impulse response of the second linear system in the cascade. We can recover the impulse responses of the first linear system in the cascade with a simple inversion of (6.6):

$$\hat{h} = (U^T U)^{-1} U^T \hat{w}.$$

Numerical example

In this numerical example, we estimate cascaded systems with the structure presented in Figure 6.7, where S_1 and S_2 are represented using stable transfer function models. We perform three Monte Carlo experiments. We generate 200 stable transfer functions by drawing 40 random poles and 40 random zeros, in complex pairs. The pole magnitudes are drawn uniformly between 0.4 and 0.8. The zero magnitudes are drawn uniformly between 0 and 0.92. The phases of the pairs are draw uniformly between 0 and π. For each system, we generate an input signal u consisting of Gaussian white noise of unitary variance. We consider measurements y and v of the output of the cascade and of the signal w, respectively; the measurements are corrupted by Gaussian white noise with variance that depends on the experiment. In *Experiment A*, the noises have a variance that is 10% of the corresponding noiseless signal variance. In *Experiment B*, the noise variances are 20% of the corresponding noiseless signal variance. In *Experiment C* the noise variances are 50% of the corresponding noiseless signal variance. In *Experiment D* the noise variances are 100% of the corresponding noiseless signal variance. In each experiment, we generate $N = 200$ samples of input and output measurements, from zero initial conditions, and we use them to estimate $n = 60$ samples of the impulse responses of the two systems. The impulse responses are modeled using zero-mean Gussian processes with stable spline covariance (6.1).

In the experiments, we compare two different strategies to estimate cascaded systems:

CS-Gibbs This is the cascaded system estimation method presented in this chapter. It estimates the hyperparameters using Algorithm 2 and sets the impulse response estimates to their posterior means given by the Gibbs sampler.

CS-TwoStage This two-stage method estimates h first and then uses it to estimate g. In the first stage, it computes the estimate $\hat{h}$ of the impulse response of the first system using the regularized FIR method, using the input u and the measured output v. Then, it computes an estimate of the input $\hat{w} = \mathbf{T}_{N\times n}(u)\hat{h}$ and uses this to estimate the impulse response $\hat{g}$ using Algorithm 6.

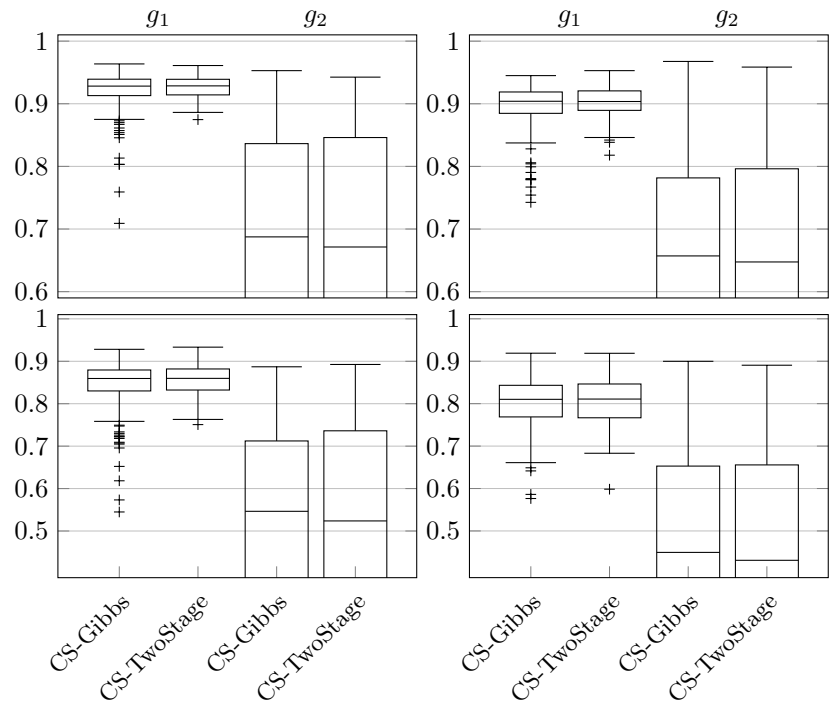

Figure 6.8: Boxplots of the estimation results in the cascaded system identification experiment. In each plot we report the fit of g_1 (left columns) and g_2 (right columns) for the two methods. The four plots refer to different noise variances: 10% (top left), 20% (top right), 50% (bottom left), 100% (bottom right), of the variance of the noiseless input signal.

The results of the simulations are presented in Figure 6.8. In the figure, we see the boxplots of the 200 simulations in the different experiments. We see an overall decrease in performance for increasing values of the noise variance. Furthermore, we see that CS-Gibbs works better than CS-TwoStage. Even though CS-Gibbs uses all the measurements jointly, it performs only marginally better than CS-TwoStage. Figure 6.9 (page 92) shows the mean performance of the methods as a function of the noise variance. As we saw in the Gibbs-sampling based method for Hammerstein systems (HS-MCEM, see Section 6.3), the correlation between the variables might be responsible for the slow convergence of the Markov chain to its stationary distribution (Neal, 1998), which could explain the performance of CS-Gibbs.

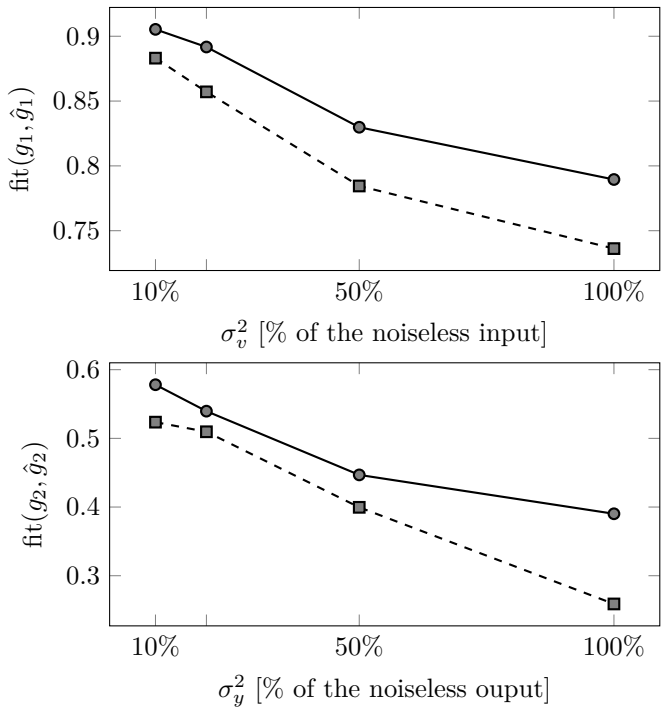

Figure 6.9: Mean fit of the estimated impulse responses as a function of the noise to signal variance ratio. We show the mean fit over the Monte Carlo experiments for CS-Gibbs (solid line), CS-TwoStage (dashed line).

6.5 Blind system identification

Blind system identification can be cast into the framework of uncertain-input models. Consider the case of a linear system S, represented by the truncated impulse response g, that is fed an unknown input (see Figure 6.10).

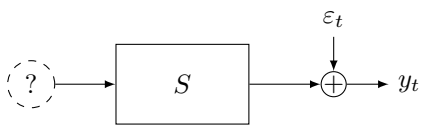

Figure 6.10: A blind identification problem.

The output samples y_t are measured with a measurement noise ε_t. We want to estimate the input and the system using the measurements of the output and prior knowledge about the input and the system S.

This problem is intrinsically ill posed: without further specification of the input sequence or of the system structure, it is impossible to retrieve an unique description

of the input and of the system.

We will suppose that the unknown input sequence w_t belongs to a known set. To be more specific, we suppose that the vector of input samples w belongs to a known p dimensional subspace,

$$w = H\theta, \tag{6.7}$$

where H is an $N \times p$ matrix whose columns span the subspace and θ are the unknown coordinates of w in the subspace.

Many signals of practical interest can be modeled in this way. For instance, we can model piecewise constant signals with known switching instants.

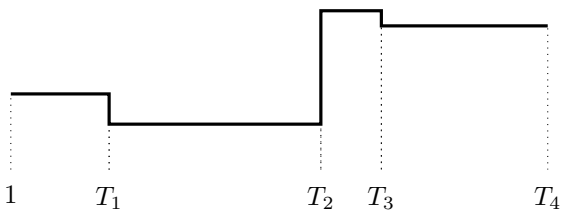

Figure 6.11: A piecewise constant signal as a linear combination of rectangular pulses.

Figure 6.11 represents a piecewise constant signal. It can be seen as the linear combination of 4 rectangular pulses with unknown amplitudes. If we call $T_1, \ldots, T_p$ the known switching instants between 1 and N, we can define the matrix whose jth column is the vectorization of a signal that is one between two subsequent time instants T_{j-1} and T_j (with the convention $T_0 = 1$ and $T_p = N$):

$$[H]_{i,j} = \begin{cases} 1, & \text{if } T_{j-1} \leq i \leq T_j, \\ 0, & \text{otherwise.} \end{cases}$$

We can also model periodic signals with components of known frequency.

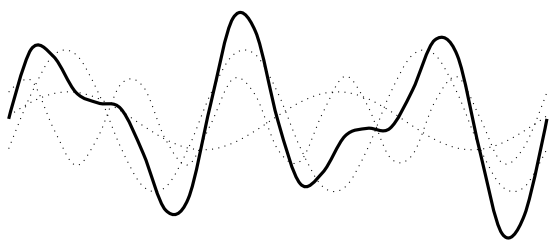

Figure 6.12: A periodic signal as a linear combination of known sinusoids.

Figure 6.12 represents periodic signal. It is the linear combination of three sinusoids of known frequency (and phase) and unknown amplitude. Also in this case, we can create a matrix H such that the unknown input w is in the linear span of

the columns of H. Call $\omega_1, \dots, \omega_p$ the known frequencies and $\phi_1, \dots, \phi_p$ the known phases of the sinusoidal components; then, the matrix H is given by

$$[H]_{i,j} = \sin(\omega_j\, i + \phi_j). \tag{6.8}$$

To model the system, we can either opt for a finite parametrization, such as a transfer function model or a state space model, or for a nonparametric description in the form of a Gaussian process model. Notice that, as in Hammerstein systems, it is in general impossible to recover the gain of the linear system and the amplitude of the input signal.

In the first case, we choose a model $S(\rho)$ where ρ is a vector of parameters. If we denote by $g(\rho)$ the vector of impulse-response samples of $S(\rho)$, we can see this case as the identification of an uncertain-input model with

$$\begin{aligned} K_g(\rho) &= 0, & \mu_g(\rho) &= g(\rho), & \sigma_v^2 &= +\infty, \\ K_w(\theta) &= 0, & \mu_w(\theta) &= H\theta, \end{aligned}$$

and we can use Algorithm 3 to get the maximum likelihood estimates of the model parameters ρ and of the input coefficients θ.

In the second case, we define a covariance matrix $K(\rho)$ for the vector of the impulse-response samples of S. We can see this case as the identification of an uncertain-input model with

$$\begin{aligned} K_g(\rho) &= K(\rho), & \mu_g(\rho) &= 0, & \sigma_v^2 &= +\infty, \\ K_w(\theta) &= 0, & \mu_w(\theta) &= H\theta, \end{aligned}$$

and we can use Algorithm 4 to get the maximum marginal-likelihood estimates of the hyperparameters ρ and of the input coefficients θ. Then, we can set the estimate of the impulse response to $\hat{g}$, the posterior mean of g given the data.

Numerical example

To assess the performance of the method, we perform a set of Monte Carlo simulations. In each simulation, we generate 100 random systems by randomly choosing 20 poles and 20 zeros. The poles are sampled uniformly in magnitude and phase inside the disk of radius 0.92 in the complex plane. The zeros are sampled uniformly in magnitude and phase inside the disk of radius 0.95. The input signals are piecewise-constant signals generated with random and independent levels and switching instants. We consider signals with $p = 10$, 20, 30, 40, 50, 60, switching instants. We estimate the first $n = 50$ samples of the impulse responses of the systems in each experiment. For each system, we generate $N = 200$ samples of the input and of the output. The output measurements are corrupted by zero mean Gaussian noise with variance that is 10% of the noiseless output variance.

We use the standard goodness-of-fit score, Equation (6.5), and we evaluate the median fit of the estimated output $\hat{W}\hat{g}$ to the true noiseless output Wg over the

100 systems (where $W = \mathbf{T}_{N\times n}(w)$ and $\hat{W} = \mathbf{T}_{N\times n}(\hat{w})$). In Figure 6.13, we see the median performance of the proposed blind kernel-based method as a function of the number of input levels to be estimated.

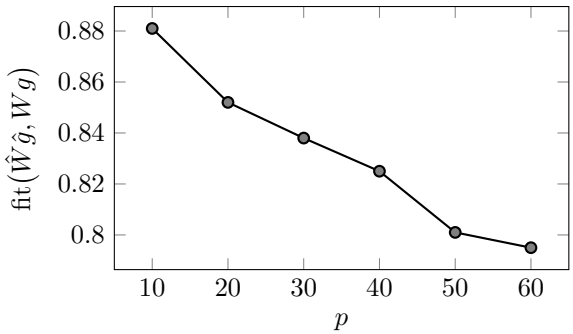

Figure 6.13: The plot shows the median fit of the predicted output over the 100 Monte Carlo experiments for different number of input switchings.

In Figure 6.14, we see the true and the estimated input signal in one Monte Carlo experiment (with $p = 20$); in Figure 6.15 (page 96), we see the true and the estimated impulse responses in the same experiment.

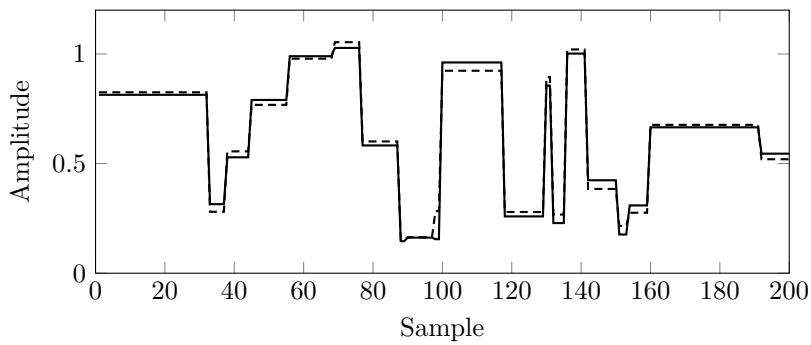

Figure 6.14: The plot shows the true input signal (solid) and the estimated input signal (dashed) in one Monte Carlo experiment.

Our simulations confirm the validity of the proposed uncertain-input method when used to do blind system identification with linearly parametrized input models. By incorporating knowledge about the input signal—in this case, that it is piecewise constant with known inputs—we can estimate the system impulse response and the input signal jointly with good accuracy.

As an application of blind system identification, we consider *occupancy-estimation* problem. Estimating occupancy—that is, the number of occupants in a room—is essential for home automation and for improving the performance of air conditioning systems. In this application (adapted from Risuleo, Molinari, et al., 2015), we use

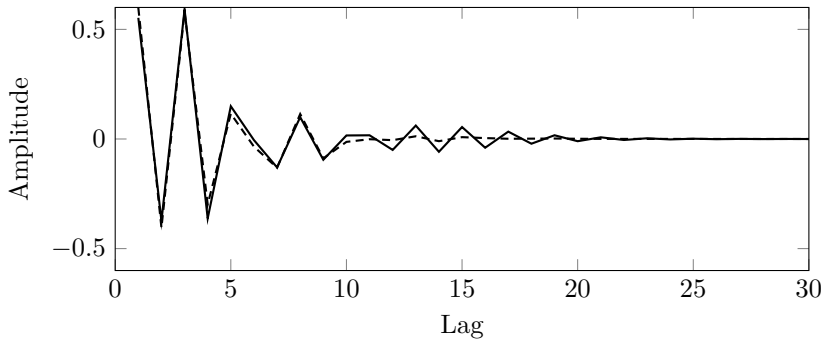

Figure 6.15: The plot shows the true impulse response (solid) and the estimated impulse response (dashed) in one Monte Carlo experiment.

CO_2 measurements from a simulated office environment to estimate the occupants in the room. We suppose that the door to the office is fitted with a sensor that detects when the door is opened. Between two door-opening events, the number of occupants in the room can not change. Therefore, we can model the occupancy over time as a piecewise constant signal with known switching instants—that is, we can use a linear parametrization like (6.7). We suppose that the relationship between the number of occupants in the room and the CO_2 concentration can be modeled as a linear dynamical system (the relationship is in fact nonlinear, see Ebadat, Bottegal, Molinari, et al., 2015). We model the linear system using a zero mean Gaussian process with stable spline covariance.

We use Algorithm 4 to estimate the kernel hyperparameters and the occupancy levels.

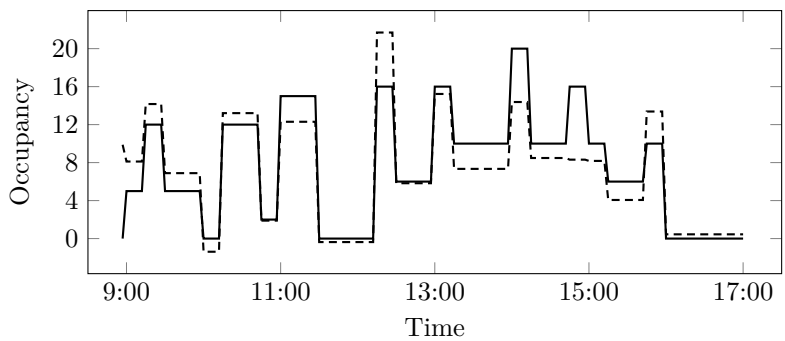

Figure 6.16: The plot shows the true occupancy (solid) and the estimated occupancy (dashed) in the blind CO_2 deconvolution experiment.

In Figure 6.16, we present the result of the occupancy estimation. The solid line is the true occupancy profile, the dashed line is the estimated signal.

In Figure 6.17, we see the CO_2 profile predicted by the model (dashed) compared

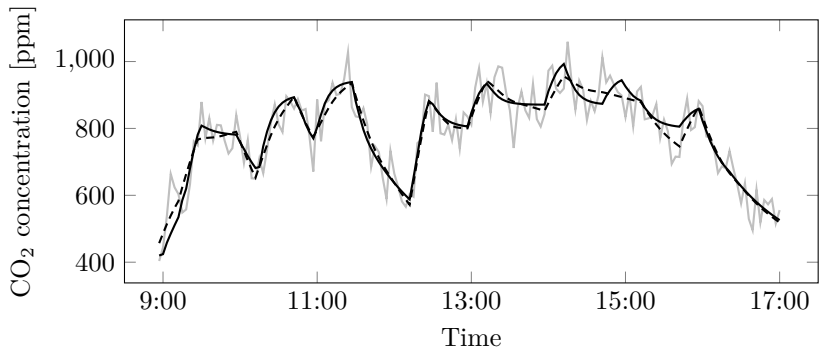

Figure 6.17: The plot show the true CO_2 concentration (solid) and CO_2 concentration estimated by the kernel-based method (dashed). The plot also shows the available measurements (gray).

to the true CO_2 profile (solid). The measurements are in gray. If we compare the result in Figure 6.16 with the result in Figure 6.14, we see that the nonlinear behavior of the room makes the identification harder.

6.6 Errors-in-variables

Errors-in-variables models are natural candidates for the uncertain-input framework. When we talk about errors-in-variables problems in system identification, we are referring to a class of models where a linear time invariant system is subject to an input signal of which only noisy measurements are available.

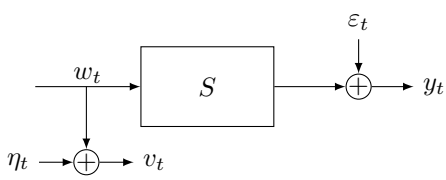

Figure 6.18: An errors-in-variables model.

Figure 6.18 shows a linear errors-in-variables model. The input w_t is unavailable to the experimenter. We have noisy measurements v_t of the input and noisy measurements y_t of the output. The objective is to estimate the system S.

We choose a parametric model of the system, in the form of a transfer function or a state-space model. The model depends on some parameters ρ. If we denote by $g(\rho)$ the vector of impulse-response samples of the model with parameters ρ, then

we can represent the errors-in-variables model as an uncertain-input model with

$$\begin{aligned} K_g(\rho) &= 0, & \mu_g(\rho) &= g(\rho), \\ K_w(\theta) &= 0, & [\mu_w(\theta)]_i &= \theta_i. \end{aligned}$$

Using Algorithm 3, we can find the hyperparameters ρ and θ that maximize the marginal likelihood (to make the problem identifiable, we suppose that we know the noise variance ratio $\gamma = \sigma_y^2/\sigma_v^2$):

$$\begin{aligned} &p(y, v|\mu_g(\rho), \theta\,; \sigma^2) = \\ &\quad - \frac{1}{2\sigma_y^2}\|y - \mathbf{T}_{N\times n}(\mu_g(\rho))\theta\|^2 - \frac{1}{2\sigma_v^2}\|v - \theta\|^2 - \frac{N}{2}\log\sigma_y^2 - \frac{N}{2}\log\sigma_v^2. \end{aligned}$$

We can maximize this expression in closed form with respect to θ, obtaining for each ρ

$$\hat{\theta}(\rho) = \left(\mathbf{T}_{N\times n}(\mu_g(\rho))^T\mathbf{T}_{N\times n}(\mu_g(\rho)) + \gamma I\right)^{-1}\left(\mathbf{T}_{N\times n}(\mu_g(\rho))^T y + \gamma v\right);$$

plugging this back into the marginal likelihood, we obtain a function of ρ alone; this is sometimes called *profile likelihood* or *concentrated likelihood* (see Söderström, 2003).

If we choose a nonparametric model of the system (in the form of a covariance matrix $K(\rho)$ for the vector g of the impulse-response samples), we can see the estimation of errors-in-variables models as the estimation of uncertain-input models with

$$\begin{aligned} K_g(\rho) &= K(\rho), & \mu_g(\rho) &= 0, \\ K_w(\theta) &= 0, & [\mu_w(\theta)]_i &= \theta_i. \end{aligned} \tag{6.9}$$

Errors-in-variables are intrinsically not identifiable, and without the introduction of additional assumptions it is impossible to recover an unique description of the system. We will suppose that the noise variance ratio γ is known; this is enough to remove the indeterminacy (Söderström, 2010).

Numerical example

In this numerical example, we see how uncertain-input estimation works for errors-in-variables models. To evaluate the performance, we perform a set of Monte Carlo simulations. In each simulation, we identify the impulse responses of the first 500 systems from the D1 dataset from Chen, Ljung, et al. (2012). For each system in the dataset, we generate $N = 210$ input and output samples; the input is Gaussian white noise with unit variance. We corrupt the samples with measurement noises. The output measurements are affected by Gaussian white noise with variance equal to 10% of the output noiseless variance. The input measurements are affected by

Gaussian white noise with a variance that changes with the simulations. We consider values of the input noise variance between 0 (no noise) and 1 (same variance as the input), in 0.2 increments.

We compare the following approaches:

KB-EIV This method uses the errors-in-variables formulation. It corresponds to Algorithm 4 with the choice (6.9). It estimates the first $n = 100$ samples of the impulse response using the stable spline kernel.

KB-Naive This method does not account for the input measurement errors and uses the noisy input samples as if they where noiseless. It corresponds to Algorithm 6 with the choice $\hat{w} = v$.

We evaluate the performance using the standard goodness of fit score (6.5).

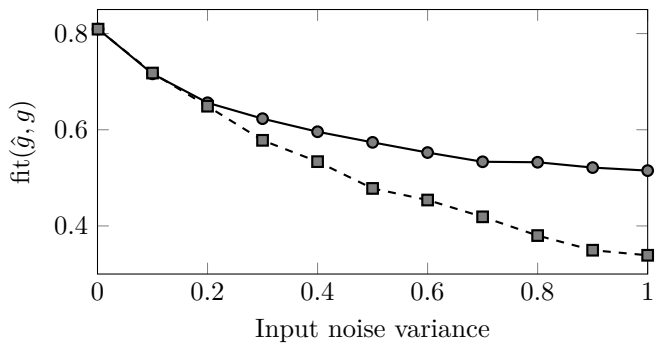

Figure 6.19: Plot of the median fit of the impulse response estimate over 500 MC runs, for increasing values of the input noise variance. We compare the errors-in-variables estimator (solid) with performance of an estimator that does not account for input noise (dashed).

The results are plotted in Figure 6.19. We see the median fit over the 500 Monte Carlo experiments, for different values of the input noise variance. From the simulation, we see that proposed estimator (solid line) is able to counteract the effect of input measurement noise; on the contrary the naive estimator (dashed line) has worse performance for higher noise variances. Notice that this method estimates the input w nonparametrically; furthermore the input in not predictable in any way. Using a finite parametrization of the input (or an input model) would yield better results (Söderström, 2003).

6.7 Missing data

To incorporate missing data into the framework, we need to modify the measurement model. We consider an errors-in-variables model like the one in Figure 6.18. We suppose that, after the measurements, some samples of the input and the output

go missing, so we do not have a full set of data to perform the identification. To model this, we introduce the operators $\mathbb{P}_v$ and $\mathbb{P}_y$, which take the full vectors of measurements and return the vectors of available measurements:

$$\tilde{y} = \mathbb{P}_y y, \qquad \tilde{v} = \mathbb{P}_v v.$$

These operators are right semi-orthogonal matrices,

$$\mathbb{P}_v {\mathbb{P}_v}^T = I_{N_v}, \qquad \mathbb{P}_y {\mathbb{P}_y}^T = I_{N_y},$$

where N_v and N_y are the number of available input and output measurements, respectively.

The measurement model we are considering is given by

$$\begin{aligned} \tilde{y} &= \mathbb{P}_y \left(\mathbf{T}_{N\times n}(w) g + \varepsilon \right), \\ \tilde{v} &= \mathbb{P}_v \left(w + \eta \right). \end{aligned}$$

We need to make a slight modification to the likelihood to account for the missing measurements. Thanks to the right semi-orthogonality of $\mathbb{P}_v$ we have that the input measurement noise at the available samples is white:

$$\begin{aligned} \mathbf{E}\left\{\mathbb{P}_v \eta\right\} &= \mathbb{P}_v \mathbf{E}\left\{\eta\right\} = 0, \\ \mathbf{cov}\left\{\mathbb{P}_v \eta\right\} &= \mathbb{P}_v \mathbf{cov}\left\{\eta\right\} \mathbb{P}_v^T = \sigma_v^2 \mathbb{P}_v \mathbb{P}_v^T = \sigma_v^2 I_{N_v}. \end{aligned}$$

The same holds for the output measurement noise at the available samples.

We use a nonparametric model for the impulse response, in the form of a Gaussian prior with zero mean and covariance matrix $K(\rho)$. Because we are interested in smoothing the input signal, we estimate all the input samples. Similarly to what we did in the errors-in-variables example, we can see this case as an uncertain-input model with

$$\begin{aligned} K_g(\rho) &= K(\rho), & \mu_g(\rho) &= 0, \\ K_w(\theta) &= 0, & \left[\mu_w(\theta)\right]_i &= \theta_i. \end{aligned}$$

We define the complete likelihood,

$$\log p(\tilde{y}|g, \mu_w(\theta)\,; \sigma_y^2) + \log(\tilde{v}\,|\mu_w(\theta)\,; \sigma_v^2) + \log p(g\,; \rho),$$

and we iteratively integrate out g using Algorithm 4. We obtain the iterations

$$\begin{aligned} \hat{\theta}^{(k+1)} &= \left(A^{(k)} + \gamma \mathbb{P}_v^T \mathbb{P}_y \right)^{-1} \left(\mathbf{T}_{N\times N}(m_g^{(k)})^T \mathbb{P}_y^T \tilde{y} + \gamma \mathbb{P}_v^T \tilde{u} \right), \\ \hat{\rho}^{(k+1)} &= \arg\min_{\rho} \operatorname{Trace}\left\{ K(\rho)^{-1} (P_g^{(k)} + m_g^{(k)} m_g^{(k)\,T}) \right\} + \log\det K(\rho), \end{aligned} \tag{6.10}$$

where $\gamma = \sigma_y^2/\sigma_v^2$ and where

$$\begin{aligned} A^{(k)} &= \mathbf{E}\left\{ \mathbf{T}_{N\times N}(g)^T {\mathbb{P}_y}^T \mathbb{P}_y \mathbf{T}_{N\times N}(g) \right\} \\ &= \mathbf{R}^T \left[\left(P_g^{(k)} + m_g^{(k)} m_g^{(k)\,T} \right) \otimes {\mathbb{P}_y}^T \mathbb{P}_y \right] \mathbf{R}, \end{aligned}$$

and

$$\mathbf{R}^T = \begin{bmatrix} I_N & S^T & S^{2\,T} & \cdots & S^{n-1\,T} \end{bmatrix}, \qquad [S]_{i,j} = \delta_{i,j+1}.$$

This example has a structure that is very similar to the errors-in-variables formulation; therefore, it inherits the same nonidentifiability problems. Without specifying, at least, the noise covariance ratio $\gamma = \sigma_y^2/\sigma_v^2$, it is impossible to recover an unique description of the system (Söderström, 2003). In addition, to obtain unique solutions to (6.10), we need that the matrix $A^{(k)} + \gamma \mathbb{P}_v^T \mathbb{P}_y$ is invertible. This is the case as long as the effect of every missing input sample is visible at least once in the output. The complete result, together with the proof, is found in Risuleo, Bottegal, and Hjalmarsson, 2016b; here, we consider a simple example. Consider the noiseless case of an FIR system, given by

$$y_t = u_{t-1} + \frac{1}{2} u_{t-2}, \tag{6.11}$$

excited by an impulse at $t = 1$. If the first input sample and the second and third output samples go missing, all the available samples will be zero, and it is impossible to discriminate whether any excitation occurred in the fist place (see Figure 6.20).

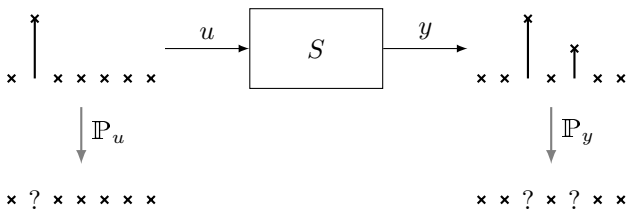

Figure 6.20: A nonidentifiable example; the loss of samples masks all information about the dynamics of the system.

Numerical example

To evaluate the performance of the uncertain-input approach to the identification of systems with missing data, we use Monte Carlo simulations. In each simulation, we identify the impulse responses of the first 500 systems from the D1 dataset from Chen, Ljung, et al. (2012). For each system in the dataset we generate $N = 210$ input and output samples. The input to the system is Gaussian white noise with unit variance. The input is measured with white Gaussian measurement noise with variance 0.1 (10% of the noiseless input variance). The output is measured with Gaussian white noise with variance that is 10% of the output noiseless variance. The objective is to estimate the impulse response of the system, modeled using the stable spline kernel. In addition, we are interested in the problems of input and output smoothing—that is, the estimation of the corresponding noiseless signals.

We perform the identification in three different scenarios. In each scenario, we remove samples, chosen at random, from the dataset. In *Scenario A*, we remove 0% to 50% of the input samples, in 10% increments. In *Scenario B*, we remove 0% to 50% of the output samples, in 10% increments. In *Scenario C* we remove equal fractions of input and output samples, between 0% and 25%, in 5% increments.

We evaluate the performance using the standard goodness of fit score (6.5).

The results of the estimation are shown in Figure 6.21.

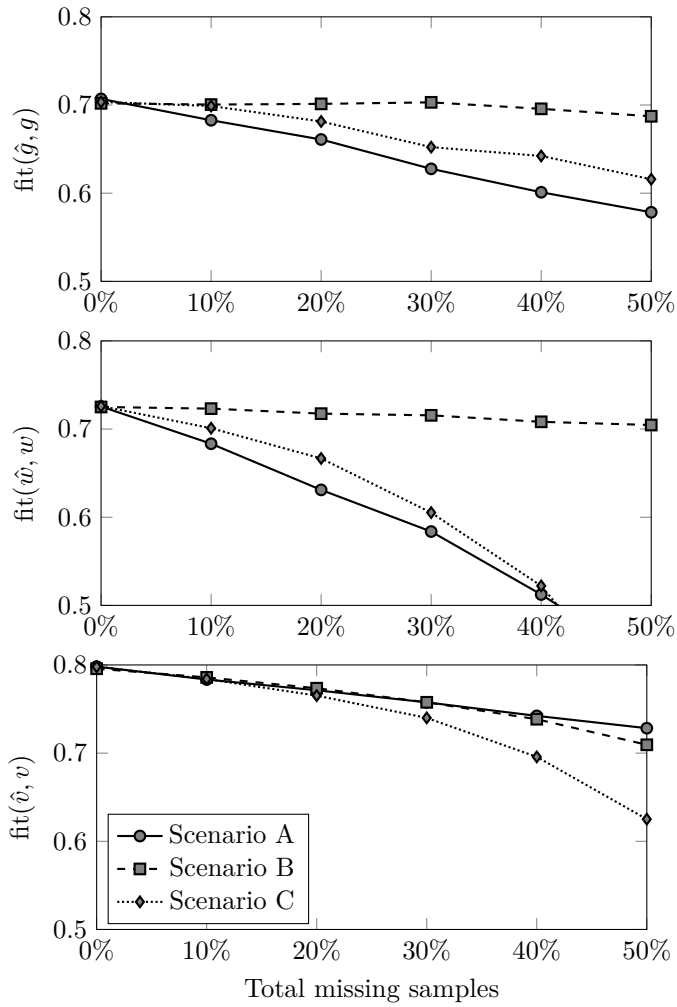

Figure 6.21: Plot of the median fit of the impulse response (top), the smoothed input (middle) and the smoothed output (bottom) over 500 MC runs, for increasing fractions of missing samples; In Scenario A we remove input samples, in Scenario B we remove output samples, in Scenario C we remove input and output samples.

From the simulations, we see that the loss of input samples (solid line) affects the performance severely, whereas the loss of output samples (dashed line) has less impact on the performance. Interestingly, the method is very robust to the loss of output samples, and the estimates remain good even when faced with only 50% of the original dataset.

As we stated, the missing-sample model is identifiable (in the sense that the EM-iteration have unique solutions and converge to a stationary point) if the effect of each missing input sample is visible in at least one available output sample. Our simulations confirm this. All the estimations in Scenario A and Scenario B were solvable. We found problems that where non identifiable, in the sense that the EM-iterations were not unique, only in Scenario C. Table 6.3 shows the number of unsolvable problems for each fraction of missing data.

Unsolvable problems	0	3	5	16	27	40
Total missing samples	0%	10%	20%	30%	40%	50%

Table 6.3: Number of nonidentifiable models in Scenario C (out of 500)

6.8 Estimation of initial conditions

Sometimes, in system identification experiments, we cannot suppose that the system is at rest. For instance, if we want to estimate the model of a process that is running, there will be initial conditions that we do not know. One common approach is to discard the initial samples, truncating the available dataset.

Consider the following case: we are estimating a finite impulse-response model, given by

$$y_t = \sum_{k=1}^{n} g_k u_{t-k} + \varepsilon_t, \qquad t = 1, \dots N,$$

from measurements $y_1, \dots, y_N$ of the output. The first n samples of the output will depend on the input samples $u_0, \dots, u_{-n+1}$ that occurred before we started measuring. If we write the convolution using the Toeplitz matrix (for simplicity we represent the case $n = 3$)

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \hdashline y_4 \\ y_5 \\ \vdots \end{bmatrix} = \begin{bmatrix} u_0 & u_{-1} & u_{-2} \\ u_1 & u_0 & u_{-1} \\ u_2 & u_1 & u_0 \\ \hdashline u_3 & u_2 & u_1 \\ u_4 & u_3 & u_2 \\ & \vdots & \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix}$$

we see that the area marked in gray contains unavailable input samples. If we opt for truncation, we truncate the matrix at the dotted line, and correspondingly discard the first three samples of the output. If the impulse response is long, and we have few data available, truncation may be a too drastic measure.

The idea is to estimate the missing samples of the input. This is in line with the blind identification problem, albeit with a different measurement model.

Let w be the n dimensional vector of initial conditions. We can define the $N \times n$ *initial-conditions operator* (example for $n = 3$)

$$\mathbf{I}_{N\times n}(w) = \begin{bmatrix} w_3 & w_2 & w_1 \\ 0 & w_3 & w_2 \\ 0 & 0 & w_3 \\ 0 & 0 & 0 \\ & \vdots & \\ 0 & 0 & 0 \end{bmatrix};$$

then, the vector of output measurements can be written as

$$y = \Big(\mathbf{T}_{N\times n}(u) + \mathbf{I}_{N\times n}(w)\Big)g + \varepsilon.$$

The initial conditions operator is such that

$$\Big(\mathbf{T}_{N\times n}(u) + \mathbf{I}_{N\times n}(w)\Big)g = \begin{bmatrix} 0_{N\times n} & I_N \end{bmatrix} \Big(\mathbf{T}_{(N+n)\times(N+n)}(g)\Big) \begin{bmatrix} w \\ u \end{bmatrix}.$$

and

$$\mathbf{I}_{N\times n}(w)g = \begin{bmatrix} 0_{N\times n} & I_N \end{bmatrix} \Big(\mathbf{T}_{(N+n)\times n}(g)\Big)w,$$

where w is the vector of unknown initial conditions and u is the vector of available input samples.

For the linear system impulse response g, we use a nonparametric model in the form a Gaussian prior with covariance matrix $K_1(\rho)$ and we estimate all samples of the input for $t < 1$.

We can see this as an uncertain-input model with

$$\begin{aligned} &K_g(\rho) = K_1(\rho), && \mu_g(\rho) = 0, && \sigma_v^2 = +\infty, \\ &K_w(\theta) = 0, && \mu_w(\theta) = \begin{bmatrix} \theta \\ u \end{bmatrix}. \end{aligned} \tag{6.12}$$

We can use Algorithm 4 to get estimates of the hyperparameters ρ and of the initial conditions (in the form of the hyperparameters θ). We introduce the complete likelihood

$$\log p(y|g\,;\theta,\sigma_y^2) + \log p(g\,;\rho)$$

where

$$\log p(y|g\,;\theta,\sigma_y^2) = \frac{1}{2\sigma_y^2}\left\|y - \Big(\mathbf{T}_{N\times n}(u) + \mathbf{I}_{N\times n}(\theta)\Big)g\right\|^2 - \frac{N}{2}\log\sigma_y^2,$$
$$\log p(g\,\rho) = -\frac{1}{2}g^T K_1(\rho)^{-1}g - \log\det K_1(\rho).$$

Taking the expectation with respect to the posterior distribution of g, for a fixed value of the hyperparameters, we obtain the update for the system hyperparameters ρ:

$$\hat{\rho}^{(k+1)} = \arg\min_{\rho}\ \text{Trace}\left\{K_g(\rho)^{-1}\big(P_g^{(k)} + m_g^{(k)}m_g^{(k)\,T}\big)\right\} + \log\det K_g(\rho).$$

The update of the initial conditions estimate is available in closed form

$$\hat{\theta}^{(k+1)} = \Big(A^{(k)}\Big)^{-1}\Big(B^{(k)}y - C^{(k)}u\Big)$$

where

$$A^{(k)} = \mathbf{E}\left\{\mathbf{T}_{(N+n)\times n}(g)^T\begin{bmatrix}0_{n\times n} & 0_{n\times n}\\ 0_{N\times n} & I_N\end{bmatrix}\mathbf{T}_{(N+n)\times N}(g)\right\},$$
$$B^{(k)} = \mathbf{E}\left\{\mathbf{T}_{(N+n)\times n}(g)^T\begin{bmatrix}0_{n\times N}\\ I_N\end{bmatrix}\right\}$$
$$C^{(k)} = \mathbf{E}\left\{\mathbf{T}_{(N+n)\times n}(g)^T\begin{bmatrix}0_{n\times N}\\ I_N\end{bmatrix}\mathbf{T}_{N\times N}(g)\right\}$$

At times, we might have some insight about the nature of the input process; for instance, we may known that is a stationary process with known spectrum. We can use this information to devise a probabilistic model for the unknown initial conditions. If we call $K_w(\cdot,\cdot)$ the known correlation function of the process, we can define the covariance matrix of the input as

$$[K_2]_{i,j} = K_w(i,j), \qquad i,j = -n+1,\ldots,N.$$

We can see the input signal w_t as the composed of two independent parts, one perfectly known part $w_+ = u$ (corresponding to the strictly positive time instants), and one unknown part w_- (corresponding to the negative time instants) that needs to be estimated from data:

$$w = \begin{bmatrix} w_{-n+1} \\ w_{-n+2} \\ \vdots \\ w_0 \\ w_1 \\ w_2 \\ \vdots \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} w^- \\ \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} w_+ \end{matrix} .$$

Correspondingly, we can partition the covariance matrix K_w into four blocks:

$$K_w = \begin{bmatrix} K_- & K_{-+} \\ K_{+-} & K_+ \end{bmatrix}.$$

With this partitioning, we can find the distribution of the initial conditions given the available input samples,

$$p(w_-|w_+) = \mathcal{N}(w_- ; w_{-|+}, K_{-|+}),$$

where, using Theorem A.1, we have

$$\begin{aligned} w_{-|+} &= K_{-+}K_+^{-1}w_+, \\ K_{-|+} &= K_- - K_{-+}K_+^{-1}K_{+-}. \end{aligned}$$

and we can use $w_{-|+}$ as an estimate of the initial conditions based on the input measurements only; alternatively, we can use $p(w_-|w_+)$ as a model (a prior distribution) for the initial conditions in an uncertain input system with

$$\begin{aligned} &K_g(\rho) = K_1(\rho), && \mu_g(\rho) = 0, && \sigma_v^2 = +\infty, \\ &K_w(\theta) = \begin{bmatrix} K_{-|+} & 0 \\ 0 & 0 \end{bmatrix}, && \mu_w(\theta) = \begin{bmatrix} w_{-|+} \\ u \end{bmatrix}. \end{aligned}$$

Numerical example

In the following numerical example, we show how the estimation of the initial conditions influence the accuracy of the estimates for FIR models. We perform five simulations, with increasing measurement sample sizes of $N = 150$, 200, 250, 300, and 400, each one consisting of 200 Monte Carlo experiments. In each simulation, we identify the impulse responses of linear systems of order 40 excited with inputs generated by filtering white noise with unit variance through an 8th order ARMA filter of the form

$$w_t + d_1 w_{t-1} + \cdots + d_p w_{t-p} = c_0 e_t + c_1 e_{t-1} + \cdots + c_p e_{t-p}, \qquad (6.13)$$

where e_t is zero-mean Gaussian white noise with unitary variance. This kind of input signal is representative of a stationary Gaussian process with zero mean and known rational spectrum (see, for instance, Papoulis and Pillai, 2002, Chapter 11). Figure 6.22 shows the block diagram of the linear system with the input-generating system.

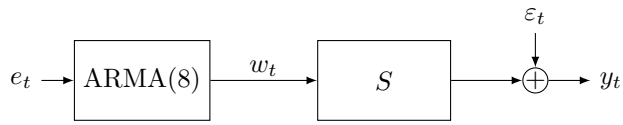

Figure 6.22: The system under study in the initial condition estimation problem; the input to the linear system is an ARMA process.

In each Monte Carlo experiment, we generate a dynamical system by randomly sampling 40 poles and 39 zeros in the complex plane. The poles are sampled within a disk of radius 0.99 and the zeros within a disk of radius 0.95. The impulse response length is $n = 100$ samples. In each experiment, we also generate a random ARMA filter with coefficients chosen randomly such that the poles are constrained within the annulus of radii 0.8 and 0.95 in the complex plane.

New realizations of input and noise are generated in each Monte Carlo experiment. The measurement noise variance is such that the ratio between the variance of the noiseless output of the system and the noise variance is equal to 20. We start collecting measurements after 500 time instants, to be sure that w_t is described by the stationary process (6.13).

The input generating model (6.13) induces a Gaussian distribution on the input signal. If we define D and C as the Toeplitz matrices of the coefficients 0, d_1, $d_2, \ldots$, and c_0, c_1, $c_2, \ldots$,

$$D = \begin{bmatrix} 0 & 0 & & \\ d_1 & 0 & 0 & \\ d_2 & d_1 & 0 & \ddots \\ d_3 & d_2 & d_1 & \ddots \\ & \ddots & \ddots & \ddots \end{bmatrix}, \qquad C = \begin{bmatrix} c_0 & 0 & & \\ c_1 & c_0 & 0 & \\ c_2 & c_1 & c_0 & \ddots \\ c_3 & c_2 & c_1 & \ddots \\ & \ddots & \ddots & \ddots \end{bmatrix},$$

then we can write a probabilistic model for the input samples according to

$$p(w) = \mathcal{N}(w\,; 0, K_w),$$

where the known covariance matrix is given by

$$K_w = (I_{N+n} + D)^{-1} C C^T (I_{N+n} + D)^{-T}.$$

In the simulations, we compare the following six strategies to deal with the unknown initial conditions:

N	150	200	250	300	400
KB-Zero	51.7	54.9	61.2	61.4	63.2
KB-Trunc	42.0	51.0	59.4	61.1	64.0
KB-Est1	54.1	55.8	61.7	63.1	63.5
KB-Est2	54.1	56.0	62.1	63.7	64.2
KB-Est3	55.7	57.1	62.8	64.3	64.5
KB-Oracle	57.3	57.9	63.8	64.9	65.0

Table 6.4: Average impulse-response fit (in percent) for different data record lengths.

KB-Zero This method does not attempt any estimation of the initial conditions. It assumes that the system was at rest before the experiment began and sets the initial conditions to 0.

KB-Trunc This method does not attempt any estimation of the initial conditions. It discards the first n samples of the output, and uses the first n samples of the input to complete the Toeplitz matrix.

KB-Est1 This method estimates all missing samples of the input, without using the probabilistic input model. It corresponds to the uncertain-input model estimate (6.12), where we estimate the whole vector $\theta = w_-$. It uses Algorithm 4.

KB-Est2 This method estimates the missing samples of the input, using only input measurements. It sets the initial conditions to their minimum mean square error estimates $\hat{w} = w_{-|+}$ and uses Algorithm 6 to estimate the impulse response.

KB-Est3 This method estimates all missing samples of the input, using the probabilistic input model $p(w_-|w_+)$. The system hyperparameters ρ are estimated, together with the initial conditions w_-, with the joint MAP–ML criterion in Algorithm 7.

KB-Oracle This method knows the value of the initial conditions and uses Algorithm 6 to estimate the impulse response, with $\hat{w} = w$.

We compare the estimators using the goodness of fit score (6.5) Table 6.4 shows the median fit of the estimated impulse responses over the Monte Carlo experiments. We can see that, especially for short data records, the estimator KB-Trunc has worse performance than the other methods, due to the loss of information caused by the truncation. The estimator KB-Zero performs better; however, it suffers from the wrong assumption that the system was at rest before the experiment was performed. Notice that, given the distributional assumption on the input, KB-Zero sets the unknown initial conditions to their prior mean. From the table, we see that

estimating the initial conditions improves the accuracy of the estimated impulse responses. The comparable performance of KB-Est1 and KB-Est2 shows that the output measurements carry much information about the initial conditions, and that using a model to estimate the initial conditions based on input measurements is a good choice. The better performance of KB-Est2 indicates that model based approaches to initial-condition estimation outperform estimation methods that do not use a model (if a reliable input model is available). The joint estimator KB-Est3 performs better than KB-Est1 and KB-Est2. In fact, KB-Est3 uses both the input model and the output measurements to estimate the initial conditions (Notice that KB-Est1 corresponds to KB-Est3 with $K_{-|+} = \infty$, and that KB-Est2 corresponds to KB-Est3 with $K_{-|+} = 0$). When the available data increases, all the methods perform well and their performance are in a neighborhood of the performance of the oracle.

Chapter 7

Conclusions and future work

In this thesis, we presented a unified framework for systems with uncertain inputs. We proposed the uncertain-input system as a generalization of the errors-in-variables system where additional information about the input signal can be encoded. The uncertain-input system is a linear system subject to an input of which only limited information is available.

To formulate a model for the uncertain-input system, we used the Gaussian regression approach. We modeled the impulse response of the linear system using a Gaussian process. The mean function and the covariance function of the Gaussian process can be used to incorporate any available information about the linear system, such as stability, overall exponential decay, and resonance. Similarly, we modeled the input signal as a Gaussian process. The mean function and the covariance function of the process can similarly be used to encode information about the input, such as smoothness or structure. Using a Bayesian interpretation, we have found the estimates of the input signal and of the system impulse response as the Bayesian minimum mean-square-error estimates—that is, their posterior means given the data.

To compute the posterior means of the unknowns, we used the empirical Bayes formulation, where the quantities parameterizing the prior distributions are estimated maximizing the marginal likelihood of the data. To carry out the marginalization of the data, we proposed an iterative algorithm based on the EM method where we treated the input and the impulse response as latent variables. Depending on the assumptions made on the input and the system models, the standard E-step may not be available in closed form. In this case the E-step is replaced with a Markov Chain Monte Carlo integration scheme based on the Gibbs sampler. After finding the hyperparameters, we used the same Gibbs sampler to calculate the estimates of the system impulse response and of the input.

In this thesis

In Chapter 1, we presented a brief introduction to FIR system identification and to the bias-variance tradeoff. In Chapter 2, we presented the mathematical background of RKHS. We started from the interpolation and smoothing problems and we showed how they are linked to RKHS. Then we presented the Gaussian process regression. In Chapter 3, we gave an introduction to empirical Bayesian inference an to the EM method. In Chapter 4, we presented the uncertain-input model. The input signal and the impulse response of the system are modeled using Gaussian processes. Furthermore, we showed the probabilistic relationships between the random variables that define the uncertain-input model In Chapter 5, we presented an empirical Bayesian algorithm to estimate the hyperparameters in the uncertain-input model. Using the probabilistic structure in the model, we set up a Monte Carlo EM method to estimate the hyperparameters of the model. The Monte Carlo integration in the method uses a Gibbs sampling step to draw from the posterior distribution of the impulse response and the input. Using the same Gibbs sampler, we estimate the impulse response and the system using the empirical Bayes minimum-variance estimate. In Chapter 6, we discussed in what way several published methods (and some novel ones) can be seen as particular cases of identification of uncertain input models. In particular, we discussed PEM, estimation of Hammerstein systems, errors-in-variables, blind system identification, estimation of systems with missing data, and estimation of initial conditions.

Future work

In this thesis, we limited our choice of dynamical models to output-error models. As a future work, we can relax this restriction and allow for more general system models. When studying systems with missing samples, we generalized the measurement model to include, in the observation operation, right semi-orthogonal selection matrices; this can be extended to other linear observation operations. Figure 7.1 shows one possible extension of the model to include linear observation devices.

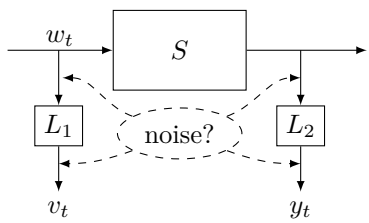

Figure 7.1: Uncertain-input model extension with linear observation devices.

The applicability of the method we have presented relies, in essence, on the property of the Toeplitz operation

$$\mathbf{T}_{N\times n}(w)g = \mathbf{T}_{N\times N}(g)w,$$

which allows us to set up the Gibbs sampling procedure to sample the posterior distribution of the unknowns. We can consider the more general case where the system is described by an operator $S(\cdot, \cdot)$, such that

$$y = S(g, w) + \varepsilon,$$

and $S(g, w)$ is affine in g (for fixed w) and affine in w (for fixed g). Also in this case, we can set up a Gibbs sampling method to sample from the joint posterior of g and w; this extension can be used, for instance, to model ARMAX linear systems.

In this thesis we have not dealt with the general problem of identifiability of the uncertain-input model; we have only seen some applications—such as Hammerstein systems, errors-in-variables systems and systems with missing data—that are non-identifiable. It could be interesting to study the problem of nonidentifiability in the general setting.

We made an interesting observation in the simulations for the Hammerstein models (Section 6.3) and in the simulations for the cascaded models (Section 6.4). In these applications, we saw that the correlation between the variables could explain the suboptimal performance of the Gibbs-sampling based methods. In these applications, the component distributions are Gaussian; this makes it easy to use *ordered overrelaxations* to increase the efficiency of the Gibbs sampler (see Adler, 1981; see also Neal, 1998). In general, a rigorous study of the properties of the MCEM used is in order.

Another future direction of research is in the cascaded system setting. We could imagine a larger cascade, with more linear systems. Another possible extension is networks of systems, or systems with mixed linear and nonlinear blocks (Wiener-Hammerstein, Hammerstein-Wiener, and networks of such models).

Appendix A

Useful mathematics

The following results on the conditional distribution of partitioned Gaussian vectors can be found in many classical books (see Bishop, 2006, Section 2.3.1; see also Kailath, Sayed, and Hassibi, 2000, Section 3.C) and are provided here to make the discussion self-contained.

Theorem A.1. Consider a Gaussian vector x with mean μ and covariance matrix Σ and consider the partition

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \qquad \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}.$$

Then, $x_1|x_2$ is Gaussian with

$$\begin{aligned} \mathbf{E}\left\{x_1|x_2\right\} &= \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2), \\ \mathbf{cov}\left\{x_1|x_2\right\} &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}. \end{aligned}$$

Proof. Define $z = x_1 - \Sigma_{12}\Sigma_{22}^{-1}x_2$. Then, z and x_2 are jointly Gaussian; in addition

$$\begin{aligned} \mathbf{cov}\left\{z, x_2\right\} &= \mathbf{cov}\left\{x_1, x_2\right\} - \Sigma_{12}\Sigma_{22}^{-1}\mathbf{cov}\left\{x_2, x_2\right\} \\ &= \Sigma_{12} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{22} = 0, \end{aligned}$$

so z and x_2 are uncorrelated and, therefore, independent. It follows that

$$\begin{aligned} \mathbf{E}\left\{x_1|x_2\right\} &= \mathbf{E}\left\{z + \Sigma_{12}\Sigma_{22}^{-1}x_2 \middle| x_2\right\} \\ &= \mathbf{E}\left\{z|x_2\right\} + \Sigma_{12}\Sigma_{22}^{-1}\mathbf{E}\left\{x_2|x_2\right\} \\ &= \mathbf{E}\left\{z\right\} + \Sigma_{12}\Sigma_{22}^{-1}x_2 \\ &= \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2). \end{aligned}$$

Then

$$\begin{aligned}&\mathbf{cov}\left\{x_1|x_2\right\} = \mathbf{cov}\left\{z+\Sigma_{12}\Sigma_{22}^{-1}x_2|x_2\right\} = \mathbf{cov}\left\{z|x_2\right\} = \mathbf{cov}\left\{z\right\}\\ &= \mathbf{cov}\left\{x_1\right\}+\Sigma_{12}\Sigma_{22}^{-1}\mathbf{cov}\left\{x_2\right\}\Sigma_{22}^{-1}\Sigma_{21}-\mathbf{cov}\left\{x_1,x_2\right\}\Sigma_{22}^{-1}\Sigma_{21}-\Sigma_{12}\Sigma_{22}^{-1}\mathbf{cov}\left\{x_2,x_1\right\}\\ &= \Sigma_{11}+\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{22}\Sigma_{22}^{-1}\Sigma_{21}-\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}-\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} = \Sigma_{11}-\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}.\end{aligned}$$

□

The previous theorem can be used to find the posterior distribution of the unknowns in the Gaussian regression problem.

Theorem A.2. Consider a Gaussian vector g with mean μ and covariance matrix K, and a Gaussian vector ε, independent of g, with zero mean and covariance matrix $\sigma^2 I$. Consider the vector $y = \Phi g + \varepsilon$ with Φ a known matrix. Then, $g|y$ is Gaussian with

$$\begin{aligned}\mathbf{E}\left\{g|y\right\} &= \left(\frac{\Phi^T\Phi}{\sigma^2}+K^{-1}\right)^{-1}\left(\frac{\Phi^T}{\sigma^2}y+K^{-1}\mu\right),\\ \mathbf{cov}\left\{g|y\right\} &= \left(\frac{\Phi^T\Phi}{\sigma^2}+K^{-1}\right)^{-1}.\end{aligned}$$

Proof. The vectors y and g are jointly Gaussian, with

$$\mathbf{E}\left\{\begin{bmatrix}y\\g\end{bmatrix}\right\} = \begin{bmatrix}\Phi\mu\\\mu\end{bmatrix}, \qquad \mathbf{cov}\left\{\begin{bmatrix}y\\g\end{bmatrix}\right\} = \begin{bmatrix}\Phi K\Phi^T+\sigma^2 I & \Phi K\\ K\Phi^T & K\end{bmatrix}$$

we use Theorem A.1 to say that

$$\begin{aligned}\mathbf{E}\left\{g|y\right\} &= \mu + K\Phi^T\left(\Phi K\Phi^T+\sigma^2 I\right)^{-1}(y-\Phi\mu)\\ &= \mu + \left(K\Phi^T\Phi+\sigma^2 I\right)^{-1}K\Phi^T(y-\Phi\mu)\\ &= \left(K\Phi^T\Phi+\sigma^2 I\right)^{-1}\left[\left(K\Phi^T\Phi+\sigma^2 I\right)\mu + K\Phi^T(y-\Phi\mu)\right]\\ &= \left(K\Phi^T\Phi+\sigma^2 I\right)^{-1}\left[K\Phi^T\Phi\mu+\sigma^2\mu+K\Phi^T y-K\Phi^T\Phi\mu\right]\\ &= \left(\frac{\Phi^T\Phi}{\sigma^2}+K^{-1}\right)^{-1}\left[\frac{\Phi^T}{\sigma^2}y+K^{-1}\mu\right].\end{aligned}$$

where we have used the Searle identities (Searle, 1982, p. 151, see also Bishop, 2006, Appendix C). Similarly, from Theorem A.1 we have

$$\mathbf{cov}\left\{g|y\right\} = K - K\Phi^T\left(\Phi K\Phi^T+\sigma^2 I\right)^{-1}\Phi = \left(\frac{\Phi^T\Phi}{\sigma^2}+K^{-1}\right)^{-1},$$

where the last equality follows from the Sherman-Morrison-Woodbury identity (Horn and Johnson, 2012, Section 0.7.4).

□

Bibliography

Abed-Meraim, K., Qiu, W., and Hua, Y. (1997). "Blind system identification". In: *Proc. IEEE* 85.8, pp. 1310–1322.

Adler, S. L. (1981). "Over-relaxation method for the Monte Carlo evaluation of the partition function for multiquadratic actions". In: *Physical Review D* 23.12, pp. 2901–2904.

Ahmed, A., Recht, B., and Romberg, J. (2014). "Blind Deconvolution Using Convex Programming". In: *IEEE Trans. Inform. Theory* 60.3, pp. 1711–1732.

Anderson, B. D. O. and Moore, J. B. (2012). *Optimal filtering*. Courier Corporation.

Aravkin, A., Burke, J. V., Chiuso, A., and Pillonetto, G. (2012). "On the Estimation of Hyperparameters for Empirical Bayes Estimators: Maximum Marginal Likelihood vs Minimum MSE". In: *Proc. IFAC Symp. System Identification (SYSID)*. Ed. by Kinnaert, M. Elsevier BV.

Aronszajn, N. (1950). "Theory of reproducing kernels". In: *Trans. Amer. Math. Soc.* 68.3, pp. 337–337.

Ayers, G. R. and Dainty, J. C. (1988). "Iterative blind deconvolution method and its applications". In: *Opt. Lett.* 13.7, p. 547.

Bai, E. W. (1998). "An optimal two-stage identification algorithm for Hammerstein–Wiener nonlinear systems". In: *Automatica* 34.3, pp. 333–338.

Bai, E. W., Cai, Z., Dudley-Javorosk, S., and Shields, R. (2009). "Identification of a modified Wiener–Hammerstein system and its application in electrically stimulated paralyzed skeletal muscle modeling". In: *Automatica* 45.3, pp. 736–743.

Bai, E. W. and Fu, M. (2002). "A blind approach to Hammerstein model identification". In: *IEEE Trans. Signal Process.* 50.7, pp. 1610–1619.

Bai, E. W. and Li, D. (2004). "Convergence of the iterative Hammerstein system identification algorithm". In: *IEEE Trans. Autom. Control* 49.11, pp. 1929–1940.

Baumgartner, S. and Rugh, W. (1975). "Complete identification of a class of nonlinear systems from steady-state frequency response". In: *IEEE T. Circuits. Syst.* 22.9, pp. 753–759.

Beghelli, S., Guidorzi, R., and Soverini, U. (1990). "The Frisch scheme in dynamic system identification". In: *Automatica* 26.1, pp. 171–176.

Berger, J. O. (1993). *Statistical Decision Theory and Bayesian Analysis*. Springer. ISBN: 0387960988.

Berlinet, A. and Thomas-Agnan, C. (2011). *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science and Business Media.

Bernardo, J. M. and Smith, A. F. M. (1994). *Bayesian Theory*. John Wiley and Sons. ISBN: 0471924164.

Billings, S. and Fakhouri, S. (1978). "Identification of a class of nonlinear systems using correlation analysis". In: *Proc. Inst. Electr. Eng.* 125.7, pp. 691–697.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Bottegal, G. and Pillonetto, G. (2013). "Regularized spectrum estimation using stable spline kernels". In: *Automatica* 49.11, pp. 3199–3209.

Bottegal, G., Aravkin, A. Y., Hjalmarsson, H., and Pillonetto, G. (2016). "Robust EM kernel-based methods for linear system identification". In: *Automatica* 67, pp. 114–126.

Bottegal, G., Hjalmarsson, H., Aravkin, A. Y., and Pillonetto, G. (2015). "Outlier robust kernel-based system identification using ℓ_1-Laplace techniques". In: *Proc. IEEE Conf. Decis. Control (CDC)*.

Bottegal, G., Risuleo, R. S., and Hjalmarsson, H. (2015). "Blind system identification using kernel-based methods". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 466–471.

Boutayeb, M., Aubry, D., and Darouach, M. (1996). "A robust and recursive identification method for MISO Hammerstein model". In: *Proc. UKACC Int. Conf. Control*. Vol. 1. IET, pp. 234–239.

Boyles, R. A. (1983). "On the Convergence of the EM Algorithm". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 45.1, pp. 47–50. ISSN: 00359246.

Casella, G. (2001). "Empirical Bayes Gibbs sampling". In: *Biostatistics* 2.4, pp. 485–500.

Chen, T., Andersen, M. S., Chiuso, A., Pillonetto, G., and Ljung, L. (2014). "Anomaly detection in homogenous populations: A sparse multiple kernel-based regularization method". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 265–270.

Chen, T., Andersen, M. S., Ljung, L., Chiuso, A., and Pillonetto, G. (2014). "System Identification Via Sparse Multiple Kernel-Based Regularization Using Sequential Convex Optimization Techniques". In: *Int. J. Adapt. Control Signal Process.* 59.11, pp. 2933–2945. ISSN: 0018-9286.

Chen, T. and Ljung, L. (2014). "Constructive State Space Model Induced Kernels for Regularized System Identification". In: *Proc. IFAC World Congr.* Vol. 19. 1, pp. 1047–1052.

Chen, T., Ljung, L., Andersen, M., Chiuso, A., Carli, F., and Pillonetto, G. (2012). "Sparse multiple kernels for impulse response estimation with majorization minimization algorithms". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 1500–1505.

Chen, T., Ohlsson, H., and Ljung, L. (2012). "On the estimation of transfer functions, regularizations and Gaussian processes—Revisited". In: *Automatica* 48.8, pp. 1525–1535.

Chiuso, A., Pillonetto, G., and De Nicolao, G. (2008). "Subspace identification using predictor estimation via Gaussian regression". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 3299–3304.

De Nicolao, G. and Pillonetto, G. (2008). "A new kernel-based approach for system identification". In: *Proc. Amer. Control Conf. (ACC)*, pp. 4510–4516.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). "Maximum likelihood from incomplete data via the EM algorithm". In: *J. R. Stat. Soc. Ser. B. Stat. Methodol.* Pp. 1–38.

— (1980). "Iteratively reweighted least squares for linear regression when errors are normal/independent distributed". In: *Multivariate analysis V*, pp. 35–57.

DeSantis, R. M., Saeks, R., and Tung, L. J. (1978). "Basic optimal estimation and control problems in Hilbert space". In: *Mathematical Systems Theory* 12.1, pp. 175–203.

Dinuzzo, F. (2015). "Kernels for linear time invariant system identification". In: *SIAM J. Control Optim.* 53.5, pp. 3299–3317.

Diversi, R., Guidorzi, R., and Soverini, U. (2007). "Maximum likelihood identification of noisy input–output models". In: *Automatica* 43.3, pp. 464–472.

Ebadat, A., Bottegal, G., Molinari, M., Varagnolo, D., Wahlberg, B., Hjalmarsson, H., and Johansson, K. H. (2015). "Multi-room occupancy estimation through adaptive gray-box models". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 3705–3711.

Ebadat, A., Bottegal, G., Varagnolo, D., Wahlberg, B., Hjalmarsson, H., and Johansson, K. H. (2015). "Blind identification strategies for room occupancy estimation". In: *Proc. European Control Conf.* Pp. 1315–1320.

Falck, T., Suykens, J., Schoukens, J., and De Moor, B. (2010). "Nuclear norm regularization for overparametrized Hammerstein systems". In: *Proc. IEEE Conf. Decis. Control (CDC)*. IEEE, pp. 7202–7207.

Fan, D. and Luo, G. (2010). "Frisch scheme identification for Errors-in-Variables systems". In: *Proc. IEEE Int. Conf. Cogn. Infor. (ICCI)*, pp. 794–799.

Fernando, K. V. and Nicholson, H. (1985). "Identification of linear systems with input and output noise: the Koopmans-Levin method". In: *IEE Proc. D. Control Theory Appl.* 132.1, pp. 30–36. ISSN: 0143-7054.

Forssell, U. and Ljung, L. (2000). "A projection method for closed-loop identification". In: *IEEE Trans. Autom. Control* 45.11, pp. 2101–2106.

Frisch, R. (1934). "Statistical Confluence Analysis by Means of Complete Regression Systems (University Institute of Economics, Oslo, 1934, pp. 5–8)". In: *The Foundations of Econometric Analysis*. Cambridge University Press, pp. 271–273.

Gelman, A. and Nolan, D. (2002). "You Can Load a Die, But You Can't Bias a Coin". In: *The American Statistician* 56.4, pp. 308–311.

Geman, S. and Geman, D. (1984). "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images". In: *IEEE Trans. Pattern Anal. Mach. Intell.* PAMI-6.6, pp. 721–741.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in Practice*. Chapman and Hall London. ISBN: 0412055511.

Giri, F. and Bai, E. W. (2010). *Block-oriented nonlinear system identification*. Vol. 1. Springer.

Goethals, I., Pelckmans, K., Suykens, J., and De Moor, B. (2005). "Subspace identification of Hammerstein systems using least squares support vector machines". In: *IEEE Trans. Autom. Control* 50.10, pp. 1509–1519.

Golub, G. and Pereyra, V. (2003). "Separable nonlinear least squares: the variable projection method and its applications". In: *Inverse Problems* 19.2, R1.

Greblicki, W. (1989). "Non-parametric orthogonal series identification of Hammerstein systems". In: *Int. J. Syst. Sci.* 20.12, pp. 2355–2367.

— (2000). "Continuous-time Hammerstein system identification". In: *IEEE Trans. Autom. Control* 45.6, pp. 1232–1236.

Greblicki, W. and Pawlak, M. (1986). "Identification of discrete Hammerstein systems using kernel regression estimates". In: *IEEE Trans. Autom. Control* 31.1, pp. 74–77.

Gustafsson, F. and Wahlberg, B. (1995). "Blind equalization by direct examination of the input sequences". In: *IEEE Trans. Commun.* 43.7, pp. 2213–2222.

Han, Y. and De Callafon, R. (2011). "Closed-Loop Identification of Hammerstein Systems Using Iterative Instrumental Variables". In: *Proc. IFAC World Cong.* Vol. 18. 1, pp. 13930–13935.

Han, Y. and De Callafon, R. (2012). "Hammerstein system identification using nuclear norm minimization". In: *Automatica* 48.9, pp. 2189–2193.

Hastie, T., Tibshirani, R., Friedman, J., and Franklin, J. (2009). *The elements of statistical learning: data mining, inference and prediction*. Springer-Verlag New York Inc. ISBN: 0387848576.

Horn, R. A. and Johnson, C. R. (2012). *Matrix Analysis*. Cambridge University Press. ISBN: 0521548233.

Horng, S.-R. C. (1986). "Sublinear Convergence of the EM Algorithm". PhD thesis.

Hunter, I. and Korenberg, M. (1986). "The identification of nonlinear biological systems: Wiener and Hammerstein cascade models". In: *Biol. Cybern.* 55.2-3, pp. 135–144.

Isaksson, A. (1993). "Identification of ARX-models subject to missing data". In: *IEEE Trans. Autom. Control* 38.5, pp. 813–819.

James, W. and Stein, C. (1961). "Estimation with quadratic loss". In: *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 1961, pp. 361–379.

Jamshidian, M. and Jennrich, R. I. (1997). "Acceleration of the EM Algorithm by Using Quasi-Newton Methods". In: *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 59.3, pp. 569–587. ISSN: 00359246.

Jeffreys, H. (1983). *Theory of Probability*. Oxford University Press. ISBN: 0198531931.

Kailath, T., Sayed, A. H., and Hassibi, B. (2000). *Linear Estimation*. Prentice Hall. ISBN: 0130224642.

Kimeldorf, G. S. and Wahba, G. (1970). "A Correspondence Between Bayesian Estimation on Stochastic Processes and Smoothing by Splines". In: *Ann. Math. Statist.* 41.2, pp. 495–502.

Lansky, D., Casella, G., McCulloch, C., and Lansky, D. (1992). "Convergence and invariance properties of the EM algorithm". In: *Proc. Stat. Comp. Sec.* Pp. 28–33.

Lansky, D. and Casella, G. (1992). "Improving the EM algorithm". In: *Computing Science and Statistics*. Springer, pp. 420–424.

Liu, C. (1998). "Parameter expansion to accelerate EM: the PX-EM algorithm". In: *Biometrika* 85.4, pp. 755–770.

Liu, C. and Rubin, D. B. (1994). "The ECME algorithm: A simple extension of EM and ECM with faster monotone convergence". In: *Biometrika* 81.4, pp. 633–648.

Liu, Y. and Bai, E. W. (2007). "Iterative identification of Hammerstein systems". In: *Automatica* 43.2, pp. 346–354.

Liu, Z., Hansson, A., and Vandenberghe, L. (2013). "Nuclear norm system identification with missing inputs and outputs". In: *Syst. Control Lett.* 62.8, pp. 605–612.

Ljung, L. (1999). *System Identification, Theory for the User*. Prentice Hall.

Ljung, L. and Glad, T. (1994). *Modeling of Dynamic Systems*. Prentice Hall. ISBN: 0135970970.

Ljung, L., Singh, R., Zhang, Q., Lindskog, P., and Iouditski, A. (2009). "Developments in The MathWorks System Identification Toolbox". In: *IFAC Proceedings Volumes* 42.10, pp. 522–527.

Loève, M. (1978). *Probability Theory II*. Springer New York.

Luenberger, D. G. (1997). *Optimization by vector space methods*. John Wiley and Sons.

Maritz, J. and Lwin, T. (1989). *Empirical bayes methods*. Chapman and Hall London.

Markovsky, I. and Usevich, K. (2013). "Structured Low-Rank Approximation with Missing Data". In: *SIAM J. Matrix Anal. & Appl.* 34.2, pp. 814–830.

McCombie, D. B., Reisner, A. T., and Asada, H. H. (2005). "Laguerre-Model Blind System Identification: Cardiovascular Dynamics Estimated From Multiple Peripheral Circulatory Signals". In: *IEEE Trans. Biomed. Eng.* 52.11, pp. 1889–1901.

McLachlan, G. and Krishnan, T. (2007). *The EM algorithm and extensions*. Vol. 382. John Wiley and Sons.

Meng, X. L. and Rubin, D. B. (1993). "Maximum likelihood estimation via the ECM algorithm: A general framework". In: *Biometrika* 80.2, pp. 267–278.

Mercer, J. (1909). "Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations". In: *Philos. Trans. R. Soc. London, Ser. A* 209.441-458, pp. 415–446.

Moore, E. H. (1916). "On properly positive Hermitian matrices". In: *Bull. Amer. Math. Soc.* 23.59, pp. 66–67.

Moulines, E., Duhamel, P., Cardoso, J.-F., and Mayrargue, S. (1995). "Subspace methods for the blind identification of multichannel FIR filters". In: *IEEE Trans. Signal Process.* 43.2, pp. 516–525.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. The MIT Press. ISBN: 9780262018029.

Mzyk, G. (2007). "Generalized kernel regression estimate for the identification of Hammerstein systems". In: *Int. J. Appl. Math. Comput. Sci.* 17.2, pp. 189–197.

Nakajima, N. (1993). "Blind deconvolution using the maximum likelihood estimation and the iterative algorithm". In: *Opt. Commun.* 100.1-4, pp. 59–66.

Neal, R. M. (1998). "Suppressing Random Walks in Markov Chain Monte Carlo Using Ordered Overrelaxation". In: *Learning in Graphical Models*. Springer Science and Business Media, pp. 205–228.

Neath, R. C. (2013). "On Convergence Properties of the Monte Carlo EM Algorithm". In: *Advances in Modern Statistical Theory and Applications: A Festschrift in honor of Morris L. Eaton*. Institute of Mathematical Statistics, pp. 43–62.

Nielsen, S. F. (2000). "The Stochastic EM Algorithm: Estimation and Asymptotic Results". In: *Bernoulli* 6.3, p. 457.

Ning, L., Georgiou, T. T., Tannenbaum, A., and Boyd, S. P. (2015). "Linear Models Based on Noisy Data and the Frisch Scheme". In: *SIAM Rev.* 57.2, pp. 167–197.

Ohlsson, H., Ratliff, L. J., Dong, R., and Sastry, S. S. (2014). "Blind Identification Via Lifting". In: *Proc. IFAC World Cong.*

Papoulis, A. S. and Pillai, U. (2002). *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill Education Ltd. ISBN: 0071226613.

Parzen, E. (1963). "Probability density functionals and reproducing kernel Hilbert spaces". In: *Proc. Symp. Time Series Anal.* Pp. 155–169.

— (1970). "Statistical inference on time series by RKHS methods". In: *Proc. Bienn. Semin. Canad. Math. Congr.*

Pettitt, A. (1985). "Re-weighted least squares estimation with censored and grouped data: An application of the EM algorithm". In: *J. R. Stat. Soc. Ser. B. Stat. Methodol.* Pp. 253–260.

Phillips, R. F. (2002). "Least absolute deviations estimation via the EM algorithm". In: *Stat. Comput.* 12.3, pp. 281–285.

Pillonetto, G. and Bell, B. M. (2007). "Bayes and empirical Bayes semi-blind deconvolution using eigenfunctions of a prior covariance". In: *Automatica* 43.10, pp. 1698–1712.

Pillonetto, G. and Chiuso, A. (2009a). "A Bayesian learning approach to linear system identification with missing data". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 4698–4703.

— (2014). "Tuning complexity in kernel-based linear system identification: The robustness of the marginal likelihood estimator". In: *Proc. European Control Conf. (ECC)*, pp. 2386–2391.

Pillonetto, G., Chiuso, A., and De Nicolao, G. (2010). "Regularized estimation of sums of exponentials in spaces generated by stable spline kernels". In: *Proc. Amer. Control Conf. (ACC)*, pp. 498–503.

Pillonetto, G. and De Nicolao, G. (2011). "Kernel selection in linear system identification Part I: A Gaussian process perspective". In: *Proc. IEEE Conf. Decis. Control - European Control Conf. (CDC-ECC)*, pp. 4318–4325.

Pillonetto, G. and Chiuso, A. (2009b). "Gaussian processes for Wiener-Hammerstein system identification". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 15. 1, pp. 838–843.

— (2015). "Tuning complexity in regularized kernel-based regression and linear system identification: The robustness of the marginal likelihood estimator". In: *Automatica* 58, pp. 106–117.

Pillonetto, G., Chiuso, A., and De Nicolao, G. (2011). "Prediction error identification of linear systems: a nonparametric Gaussian regression approach". In: *Automatica* 47.2, pp. 291–305.

Pillonetto, G. and De Nicolao, G. (2010). "A new kernel-based approach for linear system identification". In: *Automatica* 46.1, pp. 81–93.

Pillonetto, G., Dinuzzo, F., Chen, T., Nicolao, G. D., and Ljung, L. (2014). "Kernel methods in system identification, machine learning and function estimation: A survey". In: *Automatica* 50.3, pp. 657–682.

Pillonetto, G., Quang, M. H., and Chiuso, A. (2011). "A new kernel-based approach for nonlinear system identification". In: *IEEE Trans. Autom. Control* 56.12, pp. 2825–2840.

Pintelon, R. and Schoukens, J. (1999). "Identification of continuous-time systems with missing data". In: *IEEE Trans. Instrum. Meas.* 48.3, pp. 736–740.

— (2000). "Frequency domain system identification with missing data". In: *IEEE Trans. Autom. Control* 45.2, pp. 364–369.

— (2007). "Frequency domain maximum likelihood estimation of linear dynamic errors-in-variables models". In: *Automatica* 43.4, pp. 621–630.

Pintelon, R., Schoukens, J., Vandersteen, G., and Barbé, K. (2010). "Estimation of nonparametric noise and FRF models for multivariable systems—Part II: Extensions, applications". In: *Mech. Syst. Sig. Process.* 24.3, pp. 596–616.

Pintelon, R. and Schoukens, J. (2012). *System identification: a frequency domain approach*. John Wiley and Sons.

Rangan, S., Wolodkin, G., and Poolla, K. (1995). "New results for Hammerstein system identification". In: *Proc. IEEE Conf. Decis. Control (CDC)*. Vol. 1. IEEE, pp. 697–702.

Risuleo, R. S., Molinari, M., Bottegal, G., Hjalmarsson, H., and Johansson, K. H. (2015). "A benchmark for data-based office modeling: challenges related to CO2 dynamics". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 1256–1261.

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2015a). "A kernel-based approach to Hammerstein system identication". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 1011–1016.

— (2015b). "A new kernel-based approach to overparameterized Hammerstein system identification". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 115–120.

— (2015c). "On the estimation of initial conditions in kernel-based system identification". In: *Proc. IEEE Conf. Decis. Control (CDC)*, pp. 1120–1125.

— (2016a). "A nonparametric kernel-based approach to Hammerstein system identification". (in preparation).

Risuleo, R. S., Bottegal, G., and Hjalmarsson, H. (2016b). *Kernel-based system identification from noisy and incomplete input-output data. arXiv:1605.03733.*

Schoenberg, I. (1969). "Cardinal interpolation and spline functions". In: *J. Approx. Theory* 2.2, pp. 167–206.

Schölkopf, B. and Smola, A. J. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond.* MIT press.

Schoukens, J., Dobrowiecki, T., and Pintelon, R. (1998). "Parametric and nonparametric identification of linear systems in the presence of nonlinear distortions-a frequency domain approach". In: *IEEE Trans. Autom. Control* 43.2, pp. 176–190.

Schoukens, J., Pintelon, R., Dobrowiecki, T., and Rolain, Y. (2005). "Identification of linear systems with nonlinear distortions". In: *Automatica* 41.3, pp. 491–504.

Schoukens, J., Pintelon, R., Vandersteen, G., and Guillaume, P. (1997). "Frequency-domain system identification using non-parametric noise models estimated from a small number of data sets". In: *Automatica* 33.6, pp. 1073–1086.

Searle, S. R. (1982). *Matrix Algebra Useful for Statistics.* John Wiley and Sons. ISBN: 0470009616.

Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P., Hjalmarsson, H., and Juditsky, A. (1995). "Nonlinear black-box modeling in system identification: a unified overview". In: *Automatica* 31.12, pp. 1691–1724.

Söderström, T. (2003). "Why are errors-in-variables problems often tricky?" In: *Proc. European Control Conf. (ECC)*, pp. 802–807.

— (2010). "System identification for the errors-in-variables problem". In: *Proc. UKACC Int. Conf. Control*, pp. 1–14.

Söderström, T. (1981). "Identification of stochastic linear systems in presence of input noise". In: *Automatica* 17.5, pp. 713–725.

— (2007). "Errors-in-variables methods in system identification". In: *Automatica* 43.6, pp. 939–958.

Söderström, T., Hong, M., Schoukens, J., and Pintelon, R. (2010). "Accuracy analysis of time domain maximum likelihood method and sample maximum likelihood method for errors-in-variables and output error identification". In: *Automatica* 46.4, pp. 721–727.

Söderström, T., Soverini, U., and Mahata, K. (2002). "Perspectives on errors-in-variables estimation for dynamic systems". In: *Signal Process.* 82.8, pp. 1139–1154.

Söderström, T. and Stoica, P. (1988). *System identification.* Prentice-Hall, Inc.

Tikhonov, A. and Arsenin, V. Y. (1977). "Solutions of ill-posed problems". In: *WH Winston, Washington, DC* 330.

Tong, L., Liu, R.-W., Soon, V. C., and Huang, Y.-F. (1991). "Indeterminacy and identifiability of blind identification". In: *IEEE Trans. Circuits Syst.* 38.5, pp. 499–509.

Vanbeylen, L., Pintelon, R., and Schoukens, J. (2009). "Blind maximum-likelihood identification of Wiener systems". In: *IEEE Trans. Signal Process.* 57.8, pp. 3017–3029.

Verhaegen, M. and Westwick, D. (1996). "Identifying MIMO Hammerstein systems in the context of subspace model identification methods". In: *Int. J. Control* 63.2, pp. 331–349.

Wahba, G. (1990). *Spline models for observational data*. Vol. 59. SIAM. ISBN: 0898712440.

Wallin, R. and Hansson, A. (2014). "Maximum likelihood estimation of linear SISO models subject to missing output data and missing input data". In: *Int. J. Control*, pp. 1–11.

Wallin, R., Isaksson, A., and Ljung, L. (2000). "An iterative method for identification of ARX models from incomplete data". In: *Proc. IEEE Conf. Decis. Control (CDC)*.

Wei, G. C. G. and Tanner, M. A. (1990). "A Monte Carlo Implementation of the EM Algorithm and the Poor Man's Data Augmentation Algorithms". In: *Journal of the American Statistical Association* 85.411, pp. 699–704.

Westwick, D. and Kearney, R. (2001). "Separable least squares identification of nonlinear Hammerstein models: Application to stretch reflex dynamics". In: *Ann. Biomed. Eng.* 29.8, pp. 707–718.

Williams, C. and Rasmussen, C. (2006). *Gaussian processes for machine learning*.

Wills, A., Schön, T., Ljung, L., and Ninness, B. (2013). "Identification of Hammerstein–Wiener models". In: *Automatica* 49.1, pp. 70–81.

Wu, C. F. J. (1983). "On the Convergence Properties of the EM Algorithm". In: *Ann. Statist.* 11.1, pp. 95–103.

Yeredor, A. (2000). "The joint MAP-ML criterion and its relation to ML and to extended least-squares". In: *IEEE Trans. Signal Process.* 48.12, pp. 3484–3492.

Zaremba, S. (1908). "Sur l'intégration de l'équation biharmonique". In: *Bull. Acad. Sci. Cracovie*, pp. 1–29.

Zhang, E. and Pintelon, R. (2015). "Errors-in-variables identification of dynamic systems in general cases". In: *Proc. IFAC Symp. System Identification (SYSID)*. Vol. 48. 28, pp. 309–313.

Zhang, E., Pintelon, R., and Schoukens, J. (2013). "Errors-in-variables identification of dynamic systems excited by arbitrary non-white input". In: *Automatica* 49.10, pp. 3032–3041.

Zheng, W.-X. and Feng, C.-B. (1989). "Unbiased parameter estimation of linear systems in the presence of input and output noise". In: *Int. J. Adapt. Control Signal Process.* 3.3, pp. 231–251.