Estimation Focus in System Identification: Prefiltering, Noise Models, and Prediction

Lennart Ljung

Department of Electrical Engineering
Linkping University, S-581 83 Linkping, Sweden
WWW: http://www.control.isy.liu.se
Email: ljung@isy.liu.se

Report no.: LiTH-ISY-R-2182
For the IEEE Conference on Decision and Control (CDC) in Phoenix, Dec 1999

Technical reports from the Automatic Control group in Linkping are available by anonymous ftp at the address ftp.control.isy.liu.se. This report is contained in the pdf file 2182.pdf.
Estimation Focus in System Identification: Prefiltering, Noise Models, and Prediction

Lennart Ljung
Department of Electrical Engineering
Linköping University, SE-583 81 Sweden
email: ljung@isy.liu.se

Abstract

In this paper we review some features related to the use of prefiltering data for identification. In addition to the well known interplay between noise models and prefilters we discuss how to find a compromise between the need for a noise model, at the same time as having control of the approximation properties of the model. The statistical paradigm tells us to use so high order models that the bias distribution aspect of the prefilter can be neglected. For real data this may however be infeasible. The discussion is illustrated with both simulated and real data.

1 Introduction

Consider the standard problem of estimating a linear time invariant system

\[ y(t) = G_0(q)u(t) + v(t) \]  

where \( q \) is the shift operator. The number of outputs is \( p \) and the number of inputs \( m \), so the transfer operator \( G \) is \( p \times m \). The system is estimated within the parameterized family

\[ y(t) = G(q, \theta)u(t) \]  

where \( G(q, \theta) \) is an arbitrary parameterization, like the Matrix Fraction Description

\[ G(q, \theta) = A^{-1}(q)B(q) \]  

with \( \theta \) comprising the coefficients of the matrix polynomials \( A \) and \( B \) or the state space description

\[ G(q, \theta) = C(\theta)[qI - A(\theta)]^{-1}B(\theta) \]  

where the state-space matrices are arbitrarily parameterized by \( \theta \).

We estimate the parameters by the simplistic least squares curve-fitting criterion

\[ \hat{\theta}_N = \arg \min_\theta \sum_{i=1}^{N} [y(t) - G(q, \theta)u(t)]^2 \]  

For some added freedom, we allow \( y \) and \( u \) to be pre-filtered through \( L(q) \) (same filter applied to all channels):

\[ y_F(t) = L(q)y(t) \]
\[ u_F(t) = L(q)u(t) \]

\[ \hat{\theta}_N = \arg \min_\theta \sum_{i=1}^{N} [y_F(t) - G(q, \theta)u_F(t)]^2 \]  

The question is how to choose this prefilter \( L \). We shall in this contribution discuss this issue from an “innocent” perspective, assuming the we haven’t heard about noise models and the maximum likelihood method. This will of course lead to a number of déjà-vu’s, but perhaps also to some additional insights.

We shall assume, for convenience, that the input \( u \) has a well defined spectrum, i.e., that we can define

\[ R_u(\tau) = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} u(t)u^T(t-\tau) \]
\[ \Phi_u(\omega) = \int_{-\pi}^{\pi} R_u(\tau)e^{-i\omega\tau}d\tau \]  

Although the discussion of this paper applies to multivariable systems, we will for notational simplicity give the expressions below for single-input-single-output systems.

Related discussion can be found in [4], [3], [6], [2], and [1].

2 Prefiltering Data

From the curve-fitting perspective, what reasons might there be to use the prefilter \( L \)?

1. We may view \( L \) as frequency weighting over the frequency curves.

2. We would like to suppress disturbances. If, for example, the level of \( y \) is drifting, it is customary to differentiate data or to high pass filter it.
The first aspect is reinforced by the interpretations

\[ \hat{\theta}_N \approx \arg \min_{\theta} \int_{-\pi}^{\pi} \left| G(e^{i\omega}, \theta) - \frac{Y_N(\omega)}{U_N(\omega)} \right|^2 \left| L(e^{i\omega}) \right|^2 d\omega \]

\[ \lim_{N \to \infty} \hat{\theta}_N = \theta^* = \arg \min_{\theta} \int_{-\pi}^{\pi} \left| G(e^{i\omega}, \theta) - G_0(e^{i\omega}) \right|^2 \left| \Phi_u(\omega) \right| \left| L(e^{i\omega}) \right|^2 d\omega \]

Here \( Y_N \) and \( U_N \) are the DFT's and the second result is under the condition that

\[ \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} u(t)v(t-\tau) = 0 \quad \forall \tau \]  

The second aspect can be reinforced in a stochastic setting. Suppose \( v \) in (1) can be described as a stochastic process with spectrum \( \Phi_v(\omega) = \lambda |H_0(e^{i\omega})|^2 \) for some monic, stable, and inversely stable filter \( H_0 \). Then the covariance matrix of \( \hat{\theta}_N \) can be written (cf page 288 in [4]):

\[ \text{Cov}\hat{\theta}_N \approx \frac{\lambda}{N} P \]

\[ P = Q^{-1} W Q^{-1} \]

\[ Q = \frac{1}{2\pi} \int LG^*_0(G^*_0)^L d\omega \]  

\[ W = \frac{1}{2\pi} \int H^*_0 L^* G^*_0(G^*_0)^T L^* L H_0 d\omega \]

What does this tell us about the choice of \( L \)? Well, some simple manipulations show that \( P \) is minimized (in a matrix sense) by the choice

\[ L(e^{i\omega}) = 1/H_0(e^{i\omega}) \]

From a variance point of view this gives a direct advice about the prefilter: It should be the inverse of the noise description

\[ v(t) = H_0(q)e(t) \]

where \( e \) is white noise with variance \( \lambda \).

It is of course well known that this prefilter makes the method (6) equal to the maximum likelihood method for Gaussian disturbances. Also, it is well known that this weighting gives the Markov Estimates (Best linear unbiased estimate, BLUE) in case \( G \) is parameterized as a linear regression.

\section{Prefilters and Noise Models}

The question may now arise, how to find the variance-best filter \( L \) in case \( H_0 \) is unknown. A simple idea would just be to parameterize \( L \) and let it loose in the criterion (6):

\[ \text{minimize} \int_{-\pi}^{\pi} \left| G(e^{i\omega}, \theta) - G_0(e^{i\omega}) \right|^2 \left| \Phi_u(\omega) \right| \left| L(e^{i\omega}) \right|^2 d\omega \]

Under the assumptions (9)-(12) it then follows that (8) can be rewritten

\[ \lim_{N \to \infty} \hat{\theta}_N = \theta^* = \int_{-\pi}^{\pi} \left| G(e^{i\omega}, \theta) - G_0(e^{i\omega}) \right|^2 \left| \Phi_u(\omega) \right| \left| L(e^{i\omega}) \right|^2 d\omega + \lambda \int_{-\pi}^{\pi} \left| L(e^{i\omega}, \theta) \right|^2 \left| H_0(e^{i\omega}) - 1/L(e^{i\omega}, \theta) \right|^2 d\omega \]

This shows that the parameterized \( L \) will be pulled towards \( 1/H_0 \), and if there is some \( \theta \) for which \( G(q, \theta) = G_0(q) \) and \( L(q, \theta) = 1/H_0(q) \) this will be the limit in (14). Again, these are well known results in the traditional stochastic setup.

The two aspects of prefiltering, mentioned in Section 2 puts us in a dilemma: The filter that gives a desired bias-distribution according to (5) is likely to be different from the one that gives the best variance, according to (11). The formal, statistical solution to this dilemma is to increase the flexibility of the parameterized model \( G(q, \theta) \) until it contains the true system \( G_0 \), so that \( \theta^* \) in (8) does not depend on \( L \), and then choose or estimate \( L \) according to (11).

This theoretical solution, however, does not capture the situation in practice. The true system is then so complex that there is always some bias left that is distributed according to \( L \). Even the most ambitious goal would be to minimize the mean square error between the model and the true system, and that balance between variance and bias is struck typically when the bias, squared, is of the same order of magnitude as the variance.

Consequently, even as we are using higher order models, picked, e.g., by Akaike’s order determination criterion, (which heuristically can be viewed as a way to minimize the mean square error) the choice of \( L \) according to (11) could still have an adverse effect on the model, depending on its intended use.

It is quite common, especially in connection with simplified models for control, to ignore the variance aspect and choose \( L \) entirely based on the bias distribution effect on (8). This does not give any noise model, which means that model predictors will not work so well.

Another question is how a noise model works when the data also have been prefiltered: It is a common-ad-
vice to apply a bandpass filter to the data before subjecting it to system identification. This will eliminate trends and slow drifts as well as high frequency noise. Experience shows however that if an ARMAX model or a state-space model with adjustable Kalman gain is used, this may lead to “strange” results. To understand this, note that using a noise model $H(q, \theta)$ as well as a prefilter $L$ is the same as using a noise model $H(q, \theta)L^{-1}(q)$ (or a prefilter $L(q)H^{-1}(q, \theta)$). This means that $\theta$ will adjust itself according to (14) so that, in addition to fitting $G$ to $G_0$ it will try and make $H(q, \theta)$ look like $H_0(q)/L(q)$. If $L(q)$ is a high order bandpass filter, this may give a strange noise model and hence also an actual frequency weighting / prefiltering that is far from the desired $L$.

4 A PEM($f$) Approach

A way to handle these conflicting aspects on prefiltering/noise model is to use the following prediction error method with explicit prefiltering, PEM($f$):

Start from the parameterization

$$y(t) = G(q, \theta)u(t) + H(q, \theta, \eta)e(t)$$

(15)

where $\theta$ are all those parameters needed to describe the dynamics from $u$ to $y$, and $\eta$ or those additional ones (if any) that are used to parameterize the noise description. For example, for a state space model $\theta$ would parameterize the $A, B,$ and $C$ matrices as in (4), while the Kalman gain matrix $K$ would be parameterized by $\eta$ as in

$$H(q, \theta, \eta) = I + C(\theta)[qI - A(\theta)]^{-1}K(\eta)$$

(16)

Moreover, for an ARMAX model

$$A(q)y(t) = B(q)u(t) + C(q)e(t)$$

(17)

the $A$ and $B$-polynomials would give $\theta$, while $C$ gives $\eta$.

For a desired $L_2$-weighting function $|L(e^{i\omega})|^2 \Phi_i(\omega)$ in (8) we would then proceed as follows:

$$y_F(t) = L(q)y(t)$$
$$u_F(t) = L(q)u(t)$$

$$\hat{\theta}_N = \arg \min_{\hat{\theta}} \sum_{i=1}^{N} |y_F(t) - G(q, \theta)u_F(t)|^2$$

$$\hat{\eta}_N = \arg \min_{\eta} \sum_{i=1}^{N} |H^{-1}(q, \hat{\theta}_N, \eta)[y(t) - G(q, \hat{\theta}_N)u(t)]|^2$$

(18)

The asymptotic properties of these estimates are easy to derive. If (9) holds, $\hat{\theta}$ will converge to $\theta^*$ given by (8), while

$$\lim_{N \to \infty} \hat{\eta}_N = \eta^* = \int_{-\pi}^{\pi} |H(e^{i\omega}, \theta^*, \eta)|^{-2} |H^1(e^{i\omega}) - H(e^{i\omega}, \theta^*, \eta)|^2 d\omega$$

(19)

where $H^1$ is defined as the monic, stable, and inversely stable factor in

$$\lambda \frac{1}{|H^1(e^{i\omega})|^2} = \Phi_i(\omega) + |G_0(e^{i\omega}) - G(e^{i\omega}, \theta^*)|^2 \Phi_i(\omega)$$

(20)

We see that the method (18) will thus give a noise model that describes the total error spectrum. If the model error is significant, and the model is to be applied to a variety of input spectral properties, it is more robust to minimize the model error part of (20). That is done by replacing the model parameter $\theta$ in the last equation in (18) by the (unfiltered) output error estimate:

$$y_F(t) = L(q)y(t)$$
$$u_F(t) = L(q)u(t)$$

$$\hat{\theta}_N = \arg \min_{\hat{\theta}} \sum_{i=1}^{N} |y_F(t) - G(q, \theta)u_F(t)|^2$$

$$\hat{\eta}_N = \arg \min_{\eta} \sum_{i=1}^{N} |y(t) - G(q, \theta)u(t)|^2$$

$$\hat{\eta}_N = \arg \min_{\eta} \sum_{i=1}^{N} |H^{-1}(q, \hat{\theta}_N, \eta)[y(t) - G(q, \hat{\theta}_N)u(t)]|^2$$

(21)

For the asymptotic variance, in case the true system description is contained in the model parameterization, the result (10) will of course still hold for $\hat{\theta}$, while the variance of $\eta$ will depend on whether $H$ actually depends on $\theta$. If not, $\eta$’s variance is the same as if it was estimated from $v(t) = H(q, \eta)e(t)$, since the effect from $\theta$ on the error spectrum in (20) decays as $1/N$.

5 Subspace Methods: (s) and (p)

So-called subspace methods estimate the state space matrices $A, B, C, D$ and $K$ in an innovations representation

$$x(t + 1) = Ax(t) + Bu(t) + Ke(t)$$
$$y(t) = Cx(t) + Du(t) + e(t)$$

(22)

by least-squares type projections. See, e.g., [7],[8], and [4], Section 10.6 for details. In short, first $A$ and $C$ are estimated from the extended observability matrix, and then $B$ and $D$ are found from the linear regression

$$y(t) = [C^*(qI - A^*)^{-1}B + D]u(t) + v(t)$$

(23)
where $C^*$ and $A^*$ are fixed to the earlier obtained estimates. The noise term $v$ is $v = [C^*(qI - A^*)^{-1}K + I]e$. To estimate $B$ and $D$ in (23) we can apply the same type of prefiltering as in (2), (6), where the prefilter will affect both the bias distribution and the variance, as discussed in Section 2. The statistically optimal filter is one that whitens $v$ (i.e. (11)). If an estimate $K^*$ of the Kalman gain is available (as usually is the case from the first projection steps in subspace methods), this is easily accomplished by

$$y(t) = C(qI - A^* + K^*(C^*))^{-1}[Bu(t) + K^*y(t)] + Du(t) + e(t)$$

(24)

which still is a linear regression in $B$ and $D$. Although this is the most natural solution from a statistical perspective, it may not be the best choice (as we will see in Section 7) if the intended model application is not prediction.

If $B$ and $D$ are estimated from (23) we say that the method has a simulation focus, while (24) corresponds to a prediction focus. Clearly any other prefilter can be applied to (23) just as discussed in the previous section.

6 A Simulated Example

To illustrate the concepts we consider the system

$$y(t) = \frac{q^{-1} + 0.5q^{-2}}{1 - 2.2q^{-1} + 2.42q^{-2} - 1.87q^{-3} + 0.7225q^{-4}}u(t) + \frac{1 - 0.5Q^{-1}}{1 - 1.5q^{-1} + 0.7q^{-2}}e(t)$$

(25)

with $e$ being white noise with variance 1. First a data set is collected for a white noise binary ($\pm 1$) input. The modeling focus of interest is in this case the frequency band from 1 rad/s to the Nyquist frequency 3.141 rad/sec. Seven different models are estimated from these data:

1. $m4$: A fourth order Box-Jenkins model, containing the true system.
2. $mbj$: A second order Box-Jenkins model.
3. $mbj1$: A second order Box-Jenkins model estimated from the observed data filtered through a 5th order Butterworth high-pass filter with the indicated pass band.
4. $mbj2$: A second order Box-Jenkins model estimated by the method (18) using a focus filter equal to the above one.
5. $mbj3$: Same as above, but using the method (21).
6. $mo$: A second order Output-Error model.
7. $moef$: A second order Output-Error model using the filtered data as a second order Output-Error model obtained using either of the methods (18) or (21).

A separate validation data set was generated for a high-frequency input, having most of its power above 1 rad/sec. The different models were evaluated on this data set regarding their simulation and prediction abilities. The results are shown in Figures 1 and 2.

![Figure 1: The measured output and the simulated outputs from each of the seven models. (In this colorless reproduction, the curves cannot be distinguished, but that is of no importance.) The box gives the fit for the different models, described as the percentage of the output variance that has been reproduced by the model. That is $\sum (y(t) - \hat{y}(t))^2 / \sum y(t)^2$ expressed in percent.](image-url)

We see from Figure 1 that the theoretically best fit, obtained by the true order model $m4$ is about 28%. The standard second order approximation $mbj$ achieves only 6% fit, while the PEM(f)-models that have focused on the high frequency behavior come close to the optimal fit, i.e., more than 24%. The theoretical best prediction performance is about 70% as seen in Figure 2. The PEM(f) models achieve almost 90% while the conventional model for the prefiltered data is totally useless for prediction. This is the “strange” noise models that we alluded to in Section 3. It may easily happen that this noise model (which for $mbj1$ is estimated together with the dynamics model) also destroys the simulation behavior. For a higher signal to noise ratio it may also happen that $mbj2$ based on (18) is useless. Generally speaking, models based on (21) tend to have a more robust behavior, but it may still happen that they have...
4. pem A standard prediction error method.

All methods are implemented and run in the System Identification Toolbox [5], Version 5. The auxiliary orders for the subspace methods were selected as \[ 8 \ 6 \ 6 \] (prediction horizon and past outputs and inputs respectively) and MOESP weighting was used.

The four models obtained were evaluated by prediction and simulation on the validation data. The results are shown in Figures 3 and 4.

![Graph](image)

**Figure 3:** The three curves show the three measured outputs of the validation data set, along with the four model's simulated outputs. The model `sspred` turned out to be unstable, so its \( A \)-matrix was first stabilized by simply mirroring the eigenvalues in the unit circle. For more explanations, see Figure 1.

To discuss the results let us first point to a number of cautions: When dealing with real and rather complex industrial data there are always some non-stationary effects: Time variation in the system, in the noise characteristics, and/or in the input properties. This means that the evaluation over estimation/validation data sets may be random and show effects that are caused by quite irrelevant factors. The outcome of plots like Figures 3 and 4 may thus depend on how the splits have been made, what model orders have been chosen, etc.

Having said this, we may point to some features of the plots:

1. The subspace models based on a simulation focus are clearly better at simulation than those based on a prediction focus. Similarly the prediction based models are clearly better at prediction.

---

7 A Data Set from Process Industry

Consider the glass furnace data (set 96-002) from the DAISY identification data base (http://www.esat.kuleuven.ac.be/sista/daisy/). We refer to this web-site for info about these data. We select all three inputs and outputs numbers 3, 4 and 5. The delays from the inputs are estimated as \[ 4 \ 2 \ 6 \]. The first 600 data points are used for estimation, and the remaining 640 ones for validation. A 7th order model is picked by the default order selection process in the employed subspace method (`n4sid` in [5]). This also, in this case, turns out to be the choice of a standard model order procedure based on prediction methods and an MDL-criterion.

For this setup we show how four different methods perform. The methods are:

1. **sssim** A subspace method with simulation focus according to Section 5.
2. **sspred** A subspace methods with prediction focus according to Section 5.
3. **pem** A prediction error method with simulation focus, according to Section 4. That is (18) with \( L = 1 \).

---

**Figure 2:** The measured output and the one-step-ahead predicted output from each of the models. The interpretation is the same as in Figure 1. A negative fit means that the variance of the difference between the measured and model output is larger than the output variance itself.

a worse prediction performance than `moef`, which does not use any noise model.
A General lesson to be learned is that for real-world
methods to focus on prediction of measurement.

We have also pointed to the choice of the space in the

nose model in this case. To estimate it separately

beneath the cost of toeing in the scenario with a

relevance, better interpretations are to dispense with a

set of time, you should gain better data and

applicability and noise models. The important prediction

We have discussed and illustrated various aspects on

8 Conclusions

main on validation data

on estimation data is in a linear fit for the fitter-

support as to simulation

1. The above two comments apply also to the PEM

2. When the above may seem trivial, note that un-

3. The same two comments apply also to the PEM

4. The models (to be based on expert input) are


References

settles.

The predictions this paper, also in a prediction after

issue of control on the desired feedback domain.

Here, then, it is important that these models be

disabled over the factors with noise models-

data, here will always be a remaining bias that can

mismatches. This is to ensure that the model is:
The model, then, it is important that these models be

disabled over the factors with noise models-

data, here will always be a remaining bias that can

Figure 4: The above curves show the forces measured on-

Figure 4: The above curves show the forces measured on-

Figure 4: The above curves show the forces measured on-

Figure 4: The above curves show the forces measured on-