Lund University
Mathematical Statistics
Centre for Mathematical Sciences

Time Series Analysis
Fall 2017

Computer Exercise 2

Transfer function models and Prediction

In this computer exercise, you will work with input-output relations, as well as prediction in time series models. Firstly, you will be acquainted with time series having an exogenous input, having to analyze the impulse respons of such a system and from it build a suitable model. Secondly, this computer exercise deals with prediction, perhaps the most important application of time series modeling. You will be expected to make predictions of all models introduced in this course, i.e., up to a SARIMAX model.

1 Preparations before the lab

Review chapters 3, 4, and carefully read chapter 6 in the course textbook. Also make sure to read section 4.5, as it deals with transfer function models, as well as this entire computer exercise guide. You should answer the preparatory exercises in the following section with complete solutions, as they will help you to better understand and solve the lab tasks. Before the lab starts these questions will be discussed, and you are expected to be able to show that you have studied them in order to pass the computer exercise.

2 Preparatory exercises

1. Define
   (a) an ARMAX process,
   (b) the cross-covariance function, and
   (c) the transfer function model.
2. Give a brief description of how to select the model orders of the transfer function H(z). Which function is used for the analysis?
3. How do you predict an ARMA(p, q)-process, and what is the prediction error?
4. How do you predict a SARIMA(p, s, q)-process? What is the prediction error?
5. Calculate the 2-step predictor and resulting prediction error for the ARMA-process given by polynomials

$$\begin{aligned} A(z) &= a_0 + a_1 z^{-1} \\ C(z) &= c_0 + c_1 z^{-1} \end{aligned}$$

6. Why is the first entries in $A(z) = 1$?

7. What is the variance of an k-step prediction of an ARMA process? Express your answer using the polynomial $F(z)$.

8. Consider an ARMAX process

$$A(z)y(t) = C(z)e(t) + z^{-d}B(z)u(t)$$

   How do you calculate the prediction polynomials for such a process when $k \leq d$ and when $k > d$? Is there any difference in the calculation if the external input signal is white or if it is possible to model this signal as an ARMA-process.

9. State the 2-step predictor for the ARMAX-process with polynomials given by

$$\begin{aligned} A(z) &= 1 + a_1 z^{-1} \\ C(z) &= c_0 + c_1 z^{-1} \end{aligned}$$

   when

   (a) $B(z) = b_0 z^{-1} + b_1 z^{-2}$

   (b) $B(z) = b_0 z^{-3} + b_1 z^{-4}$

3 Lab Tasks

In this computer exercise, Matlab and the functions that belong to its System Identification Toolbox (SIT) will be used. In addition, some extra functions will also be used in this exercise. Make sure to download the toolkit of functions and data from the course homepage, *as well as* the Matlab files from the course textbook. Use a new empty script in Matlab for inputing all your code. Save this in the same folder as the toolkit or use the `addpath()` command. You can easily divide your script into cells using *cell mode* in Matlaband then run these separately. Simply switch it on and use `%%` on an empty line to divide your code. To run a cell press *Run Section* in the toolbar.

3.1 Modeling of an exogenous input signal

In this and in the next section, you will work with modeling of input-output relations, both using the ARMAX model and the transfer function model frameworks. As modeling of a signal which has an exogenous input (an input which is known, i.e., deterministic) is generally more complex than the common time series models encountered so far in this course, one must take care and proceed with caution. Often very simple models of a low order will suffice, while complex ones will only add variance, detrimental to the precision of predictions.

We start by creating a typical time series with a deterministic input signal, using a slight generalization of the ARMAX model, i.e., the Box-Jenkins model, having the form of

$$y(t) = \frac{C_1(z)}{A_1(z)}e(t) + \frac{B(z)z^{-d}}{A_2(z)}u(t)$$

where y is the output signal, e is a white noise, u is the input signal, and d is the time delay between input and output. Note that if $A_1(z) = A_2(z)$, we have the standard ARMAX model.

Now generate some data following the Box-Jenkins model by inputing

```
n = 500;
A1 = [1 -.65]; A2 = [1 .90 .78];
C  = 1; B  = [0 0 0 0 .4];
e = sqrt(1.5) * randn(n + 100,1);
w =  sqrt(2) * randn(n + 200,1);
A3 = [1 .5]; C3 = [1 -.3 .2];
u = filter(C3,A3,w); u = u(101:end);
y = filter(C,A1,e) + filter(B,A2,u);
u = u(101:end), y = y(101:end)
clear A1, A2, C, B, e, w, A3, C3
```

Thus, the known input u has here is here an ARMA(1,2) process. Now assume that only $y(t)$ and $u(t)$ is known.

To model y as a time series of u and e, several steps must be taken beyond regular ARMA modeling. We must first select the appropriate model orders for the polynomials in the model, then proceeding to estimate the parameters of these polynomials.

1. As a first step, we wish to determine the orders of the $B(z)$ and $A_2(z)$ polynomials. Using the transfer function framework, we denote the transfer function from u to y by $H(z) = B(z)z^{-d}/A_2(z)$. In order to estimate the order of the $B(z)$ and $A_2(z)$ polynomials, as well as determining the delay d, we need to form an estimate of the (possibly infinite) impulse response, and from it identify the appropriate models for these polynomials. As noted in the course textbook, if u is a white noise, the (scaled) impulse response can be directly estimated using the cross correlation function (CCF) from u to y. However, if u is not white, we need to perform pre-whitening, i.e., we need to form a model for the input, such that it may be viewed as being driven by a white noise, and then inverse filter both input and output with this model. In order to do so, we form an ARMA model of the input

$$A_3(z)u(t) = C_3(z)u^{pw}(t)$$

and then replace u with u^{pw}, i.e.,

$$y(t) = \frac{C_1(z)}{A_1(z)}e(t) + \frac{B(z)z^{-d}}{A_2(z)}\frac{C_3(z)}{A_3(z)}u^{pw}(t)$$

The pre-whitening step, i.e., multiplying with $A_3(z)/C_3(z)$, yields

$$\underbrace{\frac{A_3(z)}{C_3(z)}y(t)}_{y^{pw}(t)} \quad = \quad \underbrace{\frac{A_3(z)}{C_3(z)}\frac{C_1(z)}{A_1(z)}e(t)}_{v(t)} + \underbrace{\frac{B(z)z^{-d}}{A_2(z)}}_{H(z)}u^{pw}(t)$$

and the preferred transfer function model may thus be expressed as

$$y^{pw}(t) \quad = \quad v(t) + H(z)u^{pw}(t)$$

Note that the pre-whitened $y^{pw}(t)$ is now the output of the transfer function model, having the preferred uncorrelated signal as its input, allowing $H(t)$ to be estimated using the CCF from u^{pw} to y^{pw}.

Task: Use the basic analysis (`acf`, `pacf`, and `normplot`) to create an ARMA model for the input signal u as a function of white noise, u^{pw}.

QUESTION 1 *Which model did you find most suitable for u? Is it reasonably close to the one you used to generate the input?*

We then pre-whiten $y(t)$, i.e., creating $y^{pw}(t)$. Then, we compute the CCF from $u^{pw}(t)$ to $y^{pw}(t)$ by typing

```
M = 40; stem(-M:M, crosscorr(upw,ypw,M));
title('Cross correlation function'), xlabel('Lag')
hold on
plot(-M:M,  2/sqrt(length(w))*ones(1,2*M+1),'--')
plot(-M:M, -2/sqrt(length(w))*ones(1,2*M+1),'--')
hold off
```

As the estimated CCF now yields an estimate of the impulse response, $H(z)$, we can proceed to use this to determine suitable model orders for the delay, and the $B(z)$ and $A_2(z)$ polynomials using Table 4.7 in the textbook. Use `pem` to estimate your model using the transfer function model framework, i.e., by

```
A2      = ... ;
B       = ... ;
Mi      = idpoly([1],[B],[],[],[A2]);
zpw     = iddata(ypw,upw);
Mba2    = pem(zpw,Mi); present(Mba2)
vhat    = resid(Mba2,zpw);
```

where the delay may be added to `B` by adding d zeros in the beginning of the vector. If the model orders are suitable, the CCF from u^{pw} to v should be uncorrelated.

Task: Analyze the CCF u^{pw} to y^{pw} to find the model orders of the transfer function. Calculate the residual $v(t)$ and verify that it is uncorrelated with u^{pw}. Also, analyze the residual using the basic analysis.

QUESTION 2 *Which delay and which orders of the polynomials $B(z)$ and $A_2(z)$ did you find most suitable? Can you conclude that $v(t)$ is white noise? Should it be?*

2. We have now modeled y as a function of the input u, but have not yet formed a model of the ARMA-process in the Box-Jenkins model, i.e., modeled the polynomials $C_1(z)$ and $A_1(z)$. Therefore, defining the ARMA-part as

$$x(t) = \frac{C_1(z)}{A_1(z)} e(t)$$

we use the estimated polynomials $B(z)$ and $A_2(z)$ and calculate

$$x(t) = y(t) - \frac{\hat{B}(z)z^{-\hat{d}}}{\hat{A}_2(z)} u(t)$$

By filtering out the input-dependent part of the process $y(t)$, we may then estimate determining suitable orders for the polynomials $C_1(z)$ and $A_1(z)$ i the standard ARMA-modeling fashion.

Task: Use the estimates of $B(z)$ and $A_2(z)$ obtained for the pre-whitened data and form $x(t)$. Verify that the CCF from u to x is white. Determine suitable model orders for $A_1(z)$ and $C_1(z)$.

QUESTION 3 *Which orders of the polynomials $A_1(z)$ and $C_1(z)$ did you find most suitable? Was all dependence from $u(t)$ removed in $x(t)$?*

3. Finally, now having determined all the polynomial orders in our model, we estimate all polynomials all together using `pem`.

```
A1 = ... ;
A2 = ... ;
B  = ... ;
C  = ... ;
Mi = idpoly(1,B,C,A1,A2);
z = iddata(y,u);
MboxJ = pem(z,Mi);
present(MboxJ)
ehat = resid(MboxJ,z);
```

Task: Analyze the model residual, verifying that the CCF from u to e, as well as the basic analysis, shows it to be white.

QUESTION 4

- *Can you conclude that the residual is white noise, uncorrelated with the input signal? Are the parameter estimates significantly different from zero?*
- *If not, can you twiddle with the model slightly to improve the residual?*

3.2 Hairdryer data

In this section, we will try to construct a model for a set of measured data. In the file `tork.dat`, you will find 1000 observations from an input-output experiment. These measurements have been obtained from a laboratory process, which essentially is a hair dryer with measuring equipment, i.e., air is propelled by a fan through a pipe. The air is heated at the entrance of the pipe and its temperature is measured at the outlet. The input signal that is applied, stored in the first column of the data set, is the voltage over the heating coil and the output signal. The second column is the temperature of the airflow at the outlet. This physical systems can be reasonably well modeled using a simple linear model of the process. The sampling distance is 0.08 s.

Start by accessing the data material and subtract the mean values, create an `iddata` object, now having both an input and an output, and plot the first 300 points of the object

```
load('tork.dat')
  tork = tork - repmat(mean(tork),length(tork),1);
  y = tork(:,1); u = tork(:,2);
  z = iddata(y,u);
  plot(z(1:300))
```

Task: Model this input-output relation using the Box-Jenkins model introduced above. Repeating the steps in section 3.1, use the basic analysis and the CCF to find suitable model orders. Finally, estimate the model in its entirety and plot the `acf`, `pacf`, `normplot`, and CCF from u to e.

QUESTION 5

- *How long is the delay from u to y in seconds?*
- *Can you conclude that the residual is white noise, uncorrelated with the input signal? Are the parameter estimates significantly different from zero?*
- *If not, can you twiddle with the model slightly to improve the residual?*

3.3 Prediction of ARMA-processes

In this section, we examine how to predict future values of a process, using temperature measurements from the Swedish city Svedala. The temperature data is sampled every hour during a period in April and May 1994, with its (estimated) mean value subtracted (11.35 ° C). Load the measurements with the command `load svedala`. Suitable model parameters for the data set are

```
  A = [ 1  -1.79  0.84  ];
  C = [ 1  -0.18  -0.11  ];
```

To make a k-step prediction, $\widehat{Y}(t+k \mid t)$, one needs to solve the equation

$$C(z)\widehat{Y}(t+k \mid t) = G_k(z)Y(t)$$

This can be done using the filter command

```
  yhat_k = filter( Gk, C, y );
```

where G_k is obtained from the equation

$$C(z) = A(z)F_k(z) + z^{-k}G_k(z).$$

To solve this equation, one may use the convolution and deconvolution functions, writing

```
[Fk,Gk] = deconv( conv( [1,zeros(1,k-1)], CS ), AS );
```

although it is important to note that to do so, the polynomials `CS` and `AS` need to have the same length to ensure that the solution is the one expected. The user-defined function `equalLength` can be used to ensure this

```
[CS,AS] = equalLength( C, A );
```

where `C` and `A` are the original ARMA polynomials. The prediction error are

$$Y(t+k) - \widehat{Y}(t+k \mid t) = F_k(z)e(t+k),$$

where it is important to note that the prediction error will be an MA$(k-1)$-process with the generating polynomial

$$F_k(z) = 1 + f_1 z^{-1} + \cdots + f_{k-1} z^{-(k-1)}.$$

Note that if $k = 1$, then $F_1(z) = 1$, and the prediction error thus allows for an estimate of the noise variance.

QUESTION 6 *What is the estimated variance of the noise?*

In the following questions, examine the `k`-step prediction using $k = 3$ and 26.

QUESTION 7

1. *What is the estimated mean of the prediction error?*
2. *What is the expectation of the prediction error?*

QUESTION 8

1. *Assuming that the above estimated noise variance is the true one, what is the theoretical variance of the prediction error?*
2. *Using the same noise variance, what is the estimated variance of the prediction error?*

QUESTION 9 *What is the theoretical 95% confidence interval of the prediction errors?*

QUESTION 10 *How large percentage of the prediction errors are outside the 95% confidence interval? A useful trick might be to use* `sum(res>c)` *to compute how many elements in* `res` *that are greater then* `c`.

QUESTION 11 *Plot the process and the predictions in the same plot, and in a separate figure, plot the residuals. Check if the sequence of residuals behaves as an MA$(k-1)$-process by e.g. estimating its covariance function using* `covf`. *If it does not, try to explain why.*

3.4 Prediction of ARMAX-processes

When predicting ARMAX-processes, one needs to consider also the external input. We will now make use of an additional temperature measurement done at the airport Sturup. The Swedish Meteorological and Hydrological Institute (SMHI) has made a 3-step predictions of the temperature for Sturup, which may be used as an external signal to our temperature measurements in Svedala. Load the SMHI predictions into Matlab, with `load sturup`, and set the model parameters to be

```
A = [ 1 −1.49 0.57 ];
B = [ 0 0 0 0.28 −0.26 ];
C = [ 1 ];
```

QUESTION 12 *How much is the delay?*

Form the k-step predictor of the temperature at Svedala using $k = 3$ and 26 as

$$C(z)\widehat{Y}(t+k \mid t) = B(z)F_k^y(z)u(t) + G_k^y(z)Y(t),$$

where the F_k^y and G_k^y polynomials are computed as above. Then, solve

$$C(z)\widehat{u}(t+k \mid t) = G_k^u(z)u(t)$$

where $G_k^u(z)$ is found from the equation

$$B(z)F_k^y(z) = C(z)F_k^u(z) + z^{-k}G_k^u(z).$$

To do so, you first need to multiply $B(z)F_k^y(z)$ using `conv`, then use the same method as before. Independently, solve $C(z)\widehat{Y_1}(t+k \mid t) = G_k^y(z)Y(t)$, and then form the prediction

$$\widehat{Y}(t+k \mid t) = \widehat{u}(t+k \mid t) + \widehat{Y_1}(t+k \mid t).$$

QUESTION 13 *Using $k = 3$, what is the variance of the prediction errors? Plot the process, the prediction and the prediction errors. Compare the prediction errors for a process with and without the external signal.*

3.5 Prediction of SARIMA-processes

The temperature measurements from Svedala are very periodical and can therefore be modeled as a SARIMA-process. Load the data and find the suitable period.

QUESTION 14 *What is the period for this data?*

In Matlab, it is easier to write the SARIMA-processes as an ARMA process by multiplying the differentiation polynomial and A. Let

$$A^S(z)Y(t) = (1 - z^{-S})Y(t)$$

which, in Matlab, is the same as the polynomial

```
AS = [ 1 zeros(1, S-1) -1 ];
```

Task: After removing the season, form an appropriate model for the svedala data.

QUESTION 15 *Which model did you find?*

Forming predictions for the SARIMA-model

$$A(z)A^S(z)Y(t+k \mid t) = C(z)e(t+k),$$

may be done seeing it as a non-stable ARMA-model (recall that that polynomial multiplication is computed using `conv`), and performing predictions for such a model.

QUESTION 16 *Compute the estimated prediction error variance for $k = 3$ and 26, and compare them with the variance obtained from the ARMA model. Are they any better?*