Computer exercise 3 in Stationary stochastic processes, HT 17.

The purpose of this exercise is to study different techniques for spectral density estimation and some examples using the cross-spectrum and the coherence estimate.

1 Preparations

- Read through the entire computer exercise.
- Study chapter 6.5 and 9 in the course book and familiarize yourself with the MATLAB functions, `fft`, `periodogram`, `pwelch`, `cpsd`, and `mscohere`.
- Answer the exercises in the question dictionary 1.1 below and E-mail your answers to FMSF10@matstat.lu.se at latest by Monday October 9 at 12.00 noon. The answers can be written in English or Swedish by hand (readable) or in any text editor program but the final attached file should be a pdf-file. Use the following in the subject row, "Assignment 3 by tpi07xyz" (one student) or "Assignment 3 by tfy08xyz and nat09xyz" (two students). Larger group sizes than two are not allowed. Note that you might be asked to explain your answers during the exercise.

1.1 Question dictionary

1) Why should you apply zero-padding when you estimate the spectral density using the FFT?

2) Briefly explain how you can improve the periodogram using a data window technique. What are the advantages and disadvantages?

3) What is the covariance function of the stationary process \( x(t) = A \cos(2\pi f_0 t + \phi) \), \( t = 0, \pm 1, \pm 2, \ldots \), where \( A \) and \( \phi \in \text{Rect}(0, 2\pi) \) are independent stochastic variables? If you have just \( n \) available samples of a realization of this process and use the inverse Fourier transform of the periodogram as an estimate of the covariance function, how would the expected value of the estimate look? Sketch the view. Hint: study page 240 and Eq. (9.7).

4) Define an expression for the window length of the time-shifted data window of Welch method using 50% overlap, if the total number of windows are \( K \) and the number of available samples are \( n \). (It is not available in the course book!)

Turn to the next page!
5) The periodogram has large variance. Approximately how large? If you instead apply the Welch method with $K$ windows, and 50% overlap, how large is the reduction of the variance?

6) Define an expression for the cross-spectral density and the squared coherence spectrum between the input and output from a linear filter according to, $Y(t) = \int h(t-u)X(u)du$. Note that there is no disturbing noise. Simplify the expression as far as you can.

2 The periodogram and zero-padding

Please find the additional files and data on the webpage and load these into your computer. Receiving a file of (possibly unknown) data, the first step before more advanced spectral analysis always should be to make a check of the spectral content and possible strong disturbances, e.g., 50 Hz from the electricity current. Some basic algorithm, such as the periodogram can be used for a first view. Load the file unknowndata. The unknown data is contained in data.

Plot the data and check if there seems to be a large mean that differ from zero. If this is the case, the periodogram analysis will not work properly and an estimated mean, should be subtracted before the analysis. Decide if this is the case and in that case adjust data to zero-mean.

Your own periodogram is easily implemented using the Discrete Fourier Transform (DFT) and can be computed in MATLAB, using the command `fft`

```
>> X=fft(x);
```

and

```
>> n=length(x);
>> Rhat=(X.*conj(X))/n;
>> f=[0:n-1]/n;
>> plot(f,Rhat)
```

If you prefer the view of frequencies $-0.5 \leq f < 0.5$, use the command `fftfshift` according to

```
>> plot(f-0.5,fftfshift(Rhat))
```

Q. What do you see? Can you say anything about the signal?
An much better resolution view is given using zero-padding, i.e.,

\[ X = \text{fft}(x, N); \]

where \( N \) specifies the number of frequency values your spectrum estimate is computed for. For fast computation the number of frequency values should be chosen as \( N = 2^I \) where \( I \) is an integer, (e.g., \( N = 512, 1024, 2048, 4096 \ldots \)). Compute and plot the resulting spectrum estimate,

\[ R_{\text{hat}} = \frac{X \ast \text{conj}(X)}{n}; \]
\[ f = \frac{[0:N-1]}{N}; \]
\[ \text{plot}(f, R_{\text{hat}}) \]

Note that the normalizing factor of the spectral estimate should still be the data length and not the FFT-length \( N \).

**Q. How does the use of zero-padding change your spectral estimate?** The unknown signal is a realization of a cosine function process, see preparation exercise 2, (initially with a non-zero-mean). Can you identify the frequency?

The inverse FFT can be used to compute the covariance function estimate, and we plot the values for small \( \tau, \tau = 0, \ldots, 15 \), as

\[ r_{\text{hat}} = \text{ifft}(R_{\text{hat}}); \]
\[ \text{plot}([0:15], r_{\text{hat}}(1:16)) \]

**Q. Can you explain the shape of the estimate from theory?** (Compare with preparation exercise 3)
3 Investigation of different spectrum estimation techniques

Different non-parametric spectrum methods should be compared in their ability to estimate realizations of an ARMA-process. Investigate the ARMA(4,2)-process,

```matlab
>> modell.A=[1 -2.39 3.35 -2.34 0.96];
>> modell.C=[1 0 1];
```

by importing the model to armagui.

**Q. Where are the poles and zeros? Sketch the spectral density in linear and logarithmic scales and note the frequencies where the peaks and zeros are located.**

Simulate a 500 sample realization yourself using

```matlab
>> e = randn(500,1);
>> x = filter(modell.C, modell.A, e);
```

Estimate the spectral density using the MATLAB periodogram function, with zero-padding up to fft-length 4096,

```matlab
>> periodogram(x,[],4096);
```

which depicts in dB-scale. Note the frequency-scale normalized between 0 and 1, but this is just MATLAB normalization which actually should be interpreted as 0 to 0.5! Compare with the true spectral density and the frequency locations of peaks and zeros from armagui. Run the three commands above again to plot the periodogram of a new simulated realization. Note the large variation in the spectrum estimate from realization to realization!

Now compute a spectrum estimate using a Hanning windowed, (modified) periodogram,

```matlab
>> periodogram(x,hanning(500),4096);
```
Q. How does the periodogram and the Hanning windowed periodogram differ? Sketch the estimates and explain.

To visualize in linear/logarithmic scale and to receive the computed spectrum estimates as output from the periodogram function use

```matlab
>> [Rhat,f]=periodogram(x,[],4096,1);
>> plot(f,Rhat) % Linear scale
>> semilogy(f,Rhat) % Logarithmic scale
```

where the last 1 after the fft-length 4096 is the sampling frequency. Specifying this will give the output frequency axis f between 0 and 0.5, which is convenient. See the MATLAB documentation for further details.

Now use the Welch method with K=10 time-shifted Hanning windows and 50% overlap. The length L of the time-shifted window needs to be specified. Use your result from preparation exercise 5.

```matlab
>> pwelch(x,hanning(L),[],4096);
```

Simulate another realization (or a few other) of your ARMA-process, see the start of section 3, and apply the Welch method for the spectrum estimate.

Q. What are the differences using the Welch estimator compared to the periodogram? Sketch and explain.

A simple performance analysis of methods can be made using just a white noise sequence, i.e., e of your ARMA-process. As the spectrum estimate of the white noise sequence should be constant we can compare the variance of the spectrum estimates of a single realization, e.g., from the periodogram and the Welch method,
>> Rhate=periodogram(e,[],4096);
>> Rhatew=pwelch(e,hanning(L),[],4096);

where we compute the variances of $R_{hate}$ and $R_{hatew}$ and the ratio of them, i.e.,

>> var(Rhate)/var(Rhatew)

Repeat this a few times for a new simulation of the white noise sequence.

**Q.** What is the relation between the two variances? Is this in concordance with theory?

If you are interested, also use the Thomson multitaper estimator with 10 multitapers and compare the resulting spectrum estimate to the result of the Welch estimator.

>> pmtm(x,(10-1)/2, [],4096);

4 Identification of stationary Gaussian processes

4.1 Spectral estimates

Load the data of the file, threeprocessdata. The sequences $y_1$, $y_2$, and $y_3$ are realizations of length $N$ from three different zero-mean processes for which the spectral densities are seen in the figure below. We identify $A$ as an AR(5)-process, $B$ as an ARMA(2,4)-process and $C$ as some low-frequency process (an MA(128), white Gaussian noise filtered through a linear-phase FIR-filter, $\text{fir1}$ in MATLAB). The task is now to combine the correct spectral density with the corresponding sequence, $y_1$, $y_2$, or $y_3$, (one from each process), using all tools you have learned. As additional information you also have the white Gaussian noise sequences, $x_1$, $x_2$, and $x_3$, that have been used as input sequences of the filters, from which the outputs are the sequences $y_1$, $y_2$, and $y_3$, respectively.

Plot the sequences, $y_1$, $y_2$, and $y_3$, and calculate the covariance functions using $\text{covf}$. When the frequency contents of realizations are very similar, visualization of data and covariance estimates are not the best tools. Estimate the spectral densities using the periodogram (with some zero-padding), for the three process realizations. Also use the Welch method, e.g., with $K = 10$ time-shifted Hanning windows and study the resulting spectrum. If you have time, change the window length (and thereby also the number of windows) and study the results.
Figure 1: The spectral densities of the three processes.

**Q. Compare your periodograms and Welch spectra and the true spectral densities of Figure 1. Identify which sequence that belongs to each of the spectral densities, A, B and C. Sketch and explain.**

4.2 Coherence spectrum

You have possibly a good guess of what process each of the realizations comes from, although maybe you are not definitely sure. However, as we also have access to the input sequences, \( x_1 \), \( x_2 \), and \( x_3 \), (white zero-mean Gaussian noise), we could use the cross-spectrum and coherence spectrum to further investigate the realizations. If you want to be sure that no mistakes have been made in the data description and labeling, verify that the input sequences actually are white noise sequences using the Welch method.

We choose the sequences the \( y_1 \) and \( y_3 \), and would also like to use information from the input sequences \( x_1 \) and \( x_3 \). Talking with the colleague that sent you the data file, you get indications that the input sequences \( x_1 \) and \( x_3 \) might have been mixed up and that actually \( x_1 \) is \( x_3 \) and \( x_3 \) is \( x_1 \). You can easily check this using the coherence spectra
between the input sequences (and the shifted input sequences) and output sequences of respective process using the Welch estimator, with 10 windows and overlap 50 %, i.e.

\[
\text{>> mscohere}(x1,y1,\text{hanning}(L),[],4096);
\]

\[
\text{>> mscohere}(x3,y1,\text{hanning}(L),[],4096);
\]

**Q. What is the coherence spectral density according to theory, (see preparation exercise 6)? Relating to this knowledge, which of your coherence estimates seems to be correct? Were the names of the input sequences shifted?**

**4.3 Cross-spectrum and amplitude function estimate**

Make sure that you have the correct names of your input sequences. Use the Welch method to compute the cross-spectrum estimate of the correctly data pairs

\[
\text{>> Rxy=cpsd}(x1\text{renamed},y1,\text{hanning}(L),[],4096);
\]

and the spectrum estimate of the input noise sequence,

\[
\text{>> Rxx=pwelch}(x1\text{renamed},\text{hanning}(L),[],4096);
\]

and plot an estimate of the amplitude function of the respective filter, see preparation exercise 6. You can verify if the estimate is correct depicting the absolute value of \( H \), using the MATLAB function \( H=\text{tfestimate}(x1\text{renamed},y1,\text{hanning}(L),[],4096) \);

For two of the sequences, the estimated amplitude functions should be in concordance with the theoretical ones presented in the Figure 1.

If you have time you can estimate the amplitude function of sequence pair number 2 even better. Use the frequency function estimate from a Hanning windowed periodogram to compute the impulse response of the filter,

\[
\text{>> H=tfestimate}(x2,y2,\text{hanning}(500),[],4096,\text{’twosided’});
\]

\[
\text{>> h=ifft}(H);
\]

Compare with the impulse responses of sequence pairs 1 and 3. The impulse response from sequence pair number 2 does have its maximum at a high value of \( t = 64 \). This means that \( y2 \) is 64 samples delayed compared to \( x2 \). This is usual not a problem in estimation of frequency function if the delay is small and the total number of samples
reasonably large. However, for short sequences, and when the Welch method is applied with many short windows resulting in short sequences, the two data sequences used for the estimate of the cross-spectrum are no longer correlated and accordingly, the cross-spectrum will not show the actual dependence.

A good advice is to be careful and estimate the frequency function and corresponding impulse response and the time delay, correct for the time delay, and then compute the frequency function. Cut out the correct corresponding parts of $x_2$ and $y_2$, corrected for the time delay and compute the amplitude function using just these pieces of data. The result should now be in concordance with subplot C of Figure 1!