ECE438 - Laboratory 7 (week 3): Power Spectrum Estimation

October 9, 2008

1 Introduction

In the **first** and **second** weeks of this experiment, we introduced methods of statistically characterizing random processes. The sample autocorrelation and cross correlation are examples of "time domain" characterizations of random signals. In many applications, it can also be useful to get the frequency domain characteristics of a random process. Examples include detection of sinusoidal signals in noise, speech recognition and coding, and range estimation in radar systems.

In this week, we will introduce methods to estimate the *power spectrum* of a random signal given a finite number of observations. We will examine the effectiveness of the *periodogram* for spectrum estimation, and introduce the *spectrogram* to characterize a nonstationary random processes.

Questions or comments concerning this laboratory should be directed to Prof. Charles A. Bouman, School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907; (765) 494-0340; bouman@ecn.purdue.edu

2 Power Spectrum Estimation

In this section, you will estimate the power spectrum of a stationary discrete random process. The power spectrum is defined as

$$S_{xx}(\omega) = \lim_{N\to\infty} E\left[\frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)e^{-j\omega n}\right|^2\right] \tag{1}$$

There are 4 steps for calculating a power spectrum:

1. Select a window of length N and generate a finite sequence $x(0), x(1), \ldots, x(N-1)$.
2. Calculate the DTFT of the windowed sequence $x(n)$, $(n = 0, 1, \ldots, N-1)$, square the magnitude of the DTFT and divide it by the length of the sequence.
3. Take the expectation with respect to x.
4. Let the length of the window go to infinity.

In real applications, we can only approximate the power spectrum. Two methods are introduced in this section. They are the *periodogram* and the *averaged periodogram*.

3 Periodogram

Help on **fft** command
Help on **random** command

The periodogram is a simple and common method for estimating a power spectrum. Given a finite duration discrete random sequence $x(n)$, $(n = 0, 1, \ldots, N-1)$, the periodogram is defined as

$$P_{xx}(\omega) = \frac{1}{N}\left|X(\omega)\right|^2 = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)e^{-j\omega n}\right|^2$$

where $X(\omega)$ is the Discrete Time Fourier Transform (DTFT) of $x(n)$.

The periodogram $P_{xx}(\omega)$ can be computed using the Discrete Fourier Transformation (DFT), which in turn can be efficiently computed by the Fast Fourier Transformation (FFT). If $x(n)$ is of length N, you can compute an N-point DFT.

$$P_{xx}(\omega_k) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)e^{-j\omega_k n}\right|^2 \tag{2}$$

$$\omega_k = \frac{2\pi k}{N}, \qquad k = 0, 1, \ldots, N-1 \tag{3}$$

Using (2) and (3), write a Matlab function called *Pgram* to calculate the periodogram. The syntax for this function should be

```
[P,w] = Pgram(x)
```

where x is a discrete random sequence of length N. The outputs of this command are P, the samples of the periodogram, and w, the corresponding frequencies of the samples. Both P and w should be vectors of length N.

Now, let x be a Gaussian (Normal) random variable with mean 0 and variance 1. Use Matlab function *random* or *randn* to generate 1024 i.i.d. samples of x, and denote them as x_0, x_1, ..., x_{1023}. Then filter the samples of x with the filter which obeys the following difference equation

$$y(n) = 0.9y(n-1) + 0.3x(n) + 0.24x(n-1) \ . \tag{4}$$

Denote the output of the filter as $y_0, y_1, \ldots, y_{1023}$.

Note: The samples of $x(n)$ and $y(n)$ will be used in the following sections. Please save them.

Use your *Pgram* function to estimate the power spectrum of $y(n)$, $P_{yy}(\omega_k)$. Plot $P_{yy}(\omega_k)$ vs. ω_k.

Next, estimate the power spectrum of y, $P_{yy}(\omega_k)$ using 1/4 of the samples of y. Do this only using samples y_0, y_1, ..., y_{255}. Plot $P_{yy}(\omega_k)$ vs. ω_k.

INLAB REPORT:

1. Hand in your labeled plots and your *Pgram* code.
2. Compare the two plots. The first plot uses 4 times as many samples as the second one. Is the first one a better estimation than the second one? Does the first give you a smoother estimation?
3. Judging from the results, when the number of samples of a discrete random variable becomes larger, will the estimated power spectrum be smoother?

4 Averaged Periodogram

The periodogram is a simple method, but it does not yield very good results. To get a better estimate of the power spectrum, we will introduce Bartlett's method, also known as the *averaged periodogram*. This method has three steps. Suppose we have a length-N sequence $x(n)$.

1. Subdivide $x(n)$ into K nonoverlapping segments of length M. Denote the i^{th} segment as $x_i(n)$.

$$x_i(n) = x(n + iM), \quad i = 0, 1, \ldots, K-1, \quad n = 0, 1, \ldots, M-1$$

2. For each segment $x_i(n)$, compute its periodogram

$$P^{(i)}_{x_i x_i}(\omega_k) = \frac{1}{M}\left|\sum_{n=0}^{M-1} x_i(n)e^{-j\omega_k n}\right|^2$$

$$\omega_k = \frac{2\pi k}{M}$$

where $k = 0, 1, \ldots, M-1$ and $i = 0, 1, \ldots, K-1$.

3. Average the periodograms over all K segments to obtain the averaged periodogram, $P^A_{xx}(k)$,

$$P^A_{xx}(\omega_k) = \frac{1}{K}\sum_{i=0}^{K-1} P^{(i)}_{x_i x_i}(\omega_k)$$

$$\omega_k = \frac{2\pi k}{M}$$

where $k = 0, 1, \ldots, M-1$.

Write a Matlab function called *AvPgram* to calculate the averaged periodogram, using the above steps. The syntax for this function should be

```
[P,w] = AvPgram(x, K)
```

where x is a discrete random sequence of length N and the K is the number of nonoverlapping segments. The outputs of this command are P, the samples of the averaged periodogram, and w, the corresponding frequencies of the samples. Both P and w should be vectors of length M, where $N = KM$. You may use your *Pgram* function.

Hint: The command `A=reshape(x,M,K)` will orient length M segments of the vector x into K columns of the matrix A.

Use your *AvPgram* function to estimate the power spectrum of $y(n)$ which was generated in Section 3. Use all 1024 samples of $y(n)$, and let $K = 16$. Plot P vs. w.

INLAB REPORT:

1. Submit your plot and your *AvPgram* code.
2. Compare the power spectrum that you estimated using the averaged periodogram with the one you calculated in the previous section using the standard periodogram. What differences do you observe? Which do you prefer?

5 Power Spectrum and LTI Systems

Consider a linear time-invariant system with frequency response $H(e^{j\omega})$, where $S_{xx}(\omega)$ is the power spectrum of the input signal, and $S_{yy}(\omega)$ is the power spectrum of the output signal.

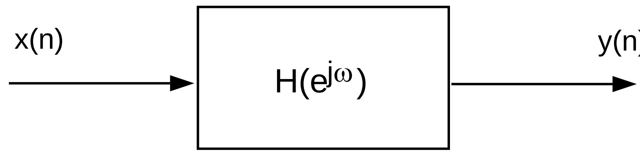

Figure 1: An LTI system

It can be shown that these quantities are related by

$$S_{yy}(\omega) = \mid H(e^{j\omega}) \mid^2 S_{xx}(\omega) \ . \tag{5}$$

In Section 3, the sequence $y(n)$ was generated by filtering an i.i.d. Gaussian (mean=0, variance=1) sequence $x(n)$, using the filter in equation (4). By hand, calculate the power spectrum $S_{xx}(\omega)$ of $x(n)$, the frequency response of the filter, $H(e^{j\omega})$, and the power spectrum $S_{yy}(\omega)$ of $y(n)$. **Hint:** In computing $S_{xx}(\omega)$, use the fact that $|ab|^2 = ab^*$. Plot $S_{yy}(\omega)$ vs. ω, and compare it with the plots from Sections 3 and 4. What do you observe?

INLAB REPORT:

1. Hand in your plot.
2. Submit your analytical calculations for $S_{xx}(\omega)$, $H(e^{j\omega})$, $S_{yy}(\omega)$.
3. Compare the theoretical power spectrum of $S_{yy}(\omega)$ with the power spectrum estimated using the standard periodogram. What can you conclude?
4. Compare the theoretical power spectrum of $S_{yy}(\omega)$ with the power spectrum estimated using the averaged periodogram. What can you conclude?

6 Estimate the Power Spectrum of a Speech Signal

Down load **speech.au** file
How to **load and play audio signals**
Help on **specgram** function

The methods used in the last two sections can only be applied to stationary random processes. However, most signals in nature are not stationary. For a nonstationary random process, one way to analyze it is to subdivide the signal into segments (which may be overlapping) and treat each segment as a stationary process. Then we can calculate the power spectrum of each segment. This yields what we call a *spectrogram*.

While it is debatable whether or not a speech signal is actually random, in many applications it is necessary to model it as being so. In this section, you are going to use the Matlab command *specgram* to calculate the spectrogram of a speech signal. Read the help for the **specgram function** . Find out what the command does, and how to calculate and draw a spectrogram.

Draw the spectrogram of the speech signal in **speech.au** . When using the *specgram* command with no output arguments, the absolute value of the spectrogram will be plotted. Therefore you can use

```
speech=auread('speech.au');
```

to load the speech signal and use

```
specgram(speech);
```

to draw the spectrogram.

INLAB REPORT:

1. Hand in your spectrogram plot.
2. Describe the information that the spectrogram is giving you.