Spectral Estimation

This lecture describes some applications of adaptive filters and explains the LMS algorithm.
After this lecture you should be able to: (1) define and apply basic terms describing stochastic processes, (2) predict the result of using a DFT as a spectral estimator for a given signal, (3) use the Welch method as a spectral estimator.
This material is covered in sections 10.1 to 10.3 of the text.

Introduction

There are many applications where it's necessary to find the frequecy content of a signal. This process is called spectral estimation. We will consider only "classical" spectral estimation in which the spectrum is defined as the FT of the ACF a signal[1]

In order to get meaningful results it is necessary to understand in detail how the spectral estimate is made. Not understanding the process (e.g. simply taking the DFT of a waveform) will generally lead to misleading or confusing results.

The basis of classical spectral estimation is the DFT and linear systems theory. In this lecture we will apply this theory to estimating the spectrum of random signals.

Stochastic Processes

A stochastic process is one that generates a signal that is a function of time. The set of all possible signals that the source can generate (possibly an infinite number) is called the *ensemble*. The expectation of some quantity (e.g. the mean), also called the ensemble average is the average of that quantity computed over all of the possible waveforms.

A process is *ergodic* if its time averages are the same as its ensemble averages. This is useful since it allows us to compute the ensemble averages by averaging over time.

A process is *stationary* if its statistics do not change with time. Most of the processes we are interested in are stationary.

Exercise: Must a stationary process be ergodic?

[1] There are other definitions based on signal models that are not sums of sinusoids.

Quality of Estimators

We are often interested in estimating a statistic of a random process. For example, we may be interested in estimating the mean, the probability distribution, or the power spectrum. When comparing estimation methods two quantities, the bias and variance are often of interest.

The *bias* is the difference between the expected value of the estimate and the true population statistic. Ideally we want an estimator with zero bias. An estimator is said to be *unbiased* if the bias tends to zero as the number of points used in the estimate grows.

The *variance* is the variance of the distribution of the estimate about the true statistic. Ideally we would like an efficient estimator that provides a low variance for a given number of samples. An estimator is said to be *consistent* if the the variance tends to zero as the number of points used in the estimate grows.

The autocorrelation of a stochastic process is:

$$c_{xx}(m) = E\{x(n)x^*(n+m)\}$$

The autocovariance is

$$c_{xx}(m) = E\{[x(n) - \overline{x}][x(n+m) - \overline{x}]^*\}$$

where $\cdot^*$ denotes complex conjugate and $\overline{x}$ is the expected value of x.

Exercise: How are the autocorrelation and autocovariance related if the mean is zero?

Spectral Estimation with the DFT (FFT)

The spectrum a stochastic process is defined as the FT of the autocovariance function.

We can use this definition to design an estimator for the spectrum of a signal. We do this by computing the ACF of one sequence generated by the random process and computing its DFT. Note that since the sequence has a finite length our ACF estimate will be biased (it will taper to zero at the ends).

Exercise: How is the ACF estimate at a lag m biased?

It is possible to show that computing the spectrum in this way is equivalent to computing the magnitude squared of the DFT of the sequence. This spectral estimate is called the *periodogram*.

The periodogram is an unbiased estimator but it is *not* consistent, i.e. as the number of samples in the periodogram (DFT) increases the mean of the estimate gets closer to the true value but the variability of the estimate increases.

It is important to understand the limitations of the periodogram that result from the fact that the DFT is computed on a *windowed* signal. Recall from previous discussions that the effect of multiplying the time-domain signal with a window is to cause convolution by the FT of the window in the frequency domain. This convolution in the frequency domain has a number of effects including limiting the resolution, "smearing" and "spectral leakage."

Consider the DFT of sampled sinusoid. The spectrum of the sinusoid will be smeared due to the convolution of the FT of the window function (a sinc function for a rectangular window) and FT of the sinusoid (an impulse). Note that if the sinusoid's frequency is a multiple of the inverse of the DFT (window) duration then the zeroes of the convolution will fall at the other DFT sample points and this smearing will be invisible. In other cases the apparent degree of "smearing" or "spectral leakage" will depend on the sinusoid's frequency. This frequency-domain convolution also has an effect on how close two frequency components can be and still be distinguished (the "resolution"). There only thing that can be done to reduce these effects is to increase the duration of the window by using longer sequences.

Improving on the Periodogram

Other than the fact that the resolution is limited by the length of the sequence, the main problem with the periodogram is that it is not a consistent estimator. A number of techniques have been developed to overcome this (i.e. to reduce the variance as the number of points used to make the estimate is increased).

The simplest technique is to divide up the sequence into a smaller number of sequences, compute the periodogram for each one and average the results at each frequency. If the sequences are independent this reduces the variance by a factor of $1/K$ where K is the number of periodograms averaged[2].

Exercise: How will averaging K periodograms affect the resolution of the spectral estimate, assuming the we start with the same number of samples?

Another technique used to reduce the variance of the spectral estimate is to use a tapered window (e.g. triangular or Hamming). These windows have a wider main lobe in the frequency domain and thus cause more spectral smearing. This smearing also causes averaging in the frequency domain and thus reduces the variance. There are many different windows available, but they all trade off reduced resolution (due to more "smearing") for reduced variance (due to more "averaging").

A consequence of windowing with tapered windows is that the samples at the start and end of the sequence do not contribute very much to the spectral estimate. Another way to improve the estimate (reduce the variance) is to extract overlapping sequences from the original sequence and average the periodograms of these overlapping sequences. Since these sequences are no longer completely independent the reduction in variance will not longer be by a factor $1/K$, but K will be larger.

The Welch Method

The standard method ("Welch Method") used to compute the spectrum of random signal (e.g. as used by the Matlab function `specgram()`) combines all of the above techniques (averaging, windowing and overlapping).

The steps are as follows:

- the sample sequence is split into overlapping segments. The amount of overlap is not too critical and will depend on the window used in the

[2] Averaging K independent, identically distributed (IID) random variables reduces the variance by K.

next step. Typically the overlap is 25 to 75 percent.

- each sequence is windowed in the time domain using a window such as the Bartlett (triangular) or Hamming windows. As with the design of FIR filters, the choice of window depends on the relative levels of the signal components that need to be resolved and the required resolution.

- the DFT is computed for each sequence (typically with the FFT) and the square of the magnitude is computed at each frequency.

- the results of the different periodograms are averaged at each frequency to obtain the final spectral estimate

If a “smoother” estimate of the spectrum is required it is possible to append zeros to each DFT. This increases the number of points computed by the DFT but *it does not change the size of the data window* and so the resolution is unaffected. Zero padding is simply a way to interpolate between the points computed by the DFT.