Chapter 7

Multiple Time Series

7.1 Introduction

We now consider the situation where we have a number of time series and wish to explore the relations between them. We first look at the relation between cross-correlation and multivariate autoregressive models and then at the cross-spectral density and coherence.

7.2 Cross-correlation

Given *two* time series x_t and y_t we can delay x_t by T samples and then calculate the *cross-covariance* between the pair of signals. That is

$$\sigma_{xy}(T) = \frac{1}{N-1} \sum_{t=1}^{N} (x_{t-T} - \mu_x)(y_t - \mu_y) \tag{7.1}$$

where μ_x and μ_y are the means of each time series and there are N samples in each. The function $\sigma_{xy}(T)$ is the *cross-covariance* function. The *cross-correlation* is a normalised version

$$r_{xy}(T) = \frac{\sigma_{xy}(T)}{\sqrt{\sigma_{xx}(0)\sigma_{yy}(0)}} \tag{7.2}$$

where we note that $\sigma_{xx}(0) = \sigma_x^2$ and $\sigma_{yy}(0) = \sigma_y^2$ are the variances of each signal. Note that

$$r_{xy}(0) = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{7.3}$$

which is the correlation between the two variables. Therefore unlike the autocorrelation, r_{xy} is not, generally, equal to 1. Figure 7.1 shows two time series and their cross-correlation.

7.2.1 Cross-correlation is asymmetric

First, we re-cap as to why the auto-correlation is a *symmetric* function. The autocovariance, for a zero mean signal, is given by

$$\sigma_{xx}(T) = \frac{1}{N-1} \sum_{t=1}^{N} x_{t-T} x_t \tag{7.4}$$

This can be written in the shorthand notation

$$\sigma_{xx}(T) =< x_{t-T} x_t > \tag{7.5}$$

where the angled brackets denote the average value or *expectation*. Now, for negative lags

$$\sigma_{xx}(-T) =< x_{t+T} x_t > \tag{7.6}$$

Subtracting T from the time index (this will make no difference to the expectation) gives

$$\sigma_{xx}(-T) =< x_t x_{t-T} > \tag{7.7}$$

which is identical to $\sigma_{xx}(T)$, as the ordering of variables makes no difference to the expected value. Hence, the autocorrelation is a symmetric function.

The cross-correlation is a normalised cross-covariance which, assuming zero mean signals, is given by

$$\sigma_{xy}(T) =< x_{t-T} y_t > \tag{7.8}$$

and for negative lags

$$\sigma_{xy}(-T) =< x_{t+T} y_t > \tag{7.9}$$

Subtracting T from the time index now gives

$$\sigma_{xy}(-T) =< x_t y_{t-T} > \tag{7.10}$$

which is different to $\sigma_{xy}(T)$. To see this more clearly we can subtract T once more from the time index to give

$$\sigma_{xy}(-T) =< x_{t-T} y_{t-2T} > \tag{7.11}$$

Hence, the cross-covariance, and therefore the cross-correlation, is an *asymmetric* function.

To summarise: moving signal A right (forward in time) and multiplying with signal B is not the same as moving signal A left and multiplying with signal B; unless signal A equals signal B.

7.2.2 Windowing

When calculating cross-correlations there are fewer data points at larger lags than at shorter lags. The resulting estimates are commensurately less accurate. To take account of this the estimates at long lags can be smoothed using various window operators. See lecture 5.

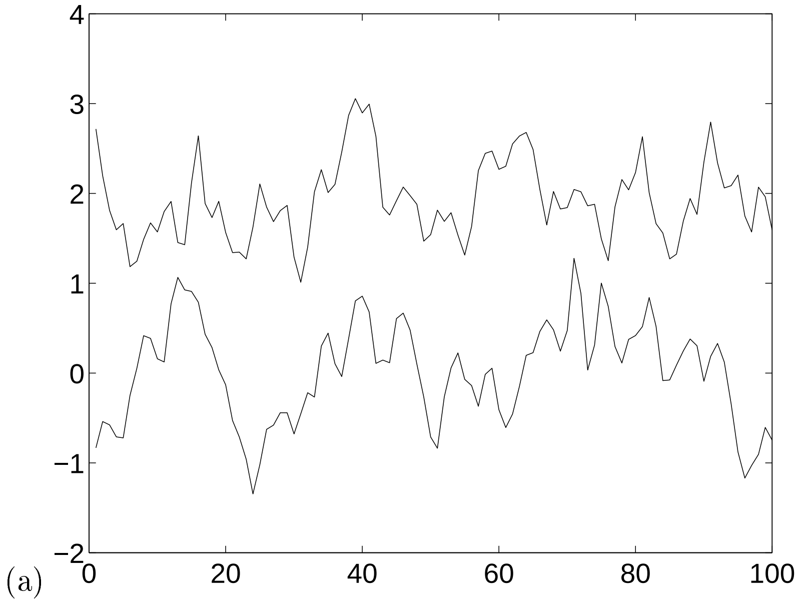

Figure 7.1: *Signals x_t (top) and y_t (bottom).*

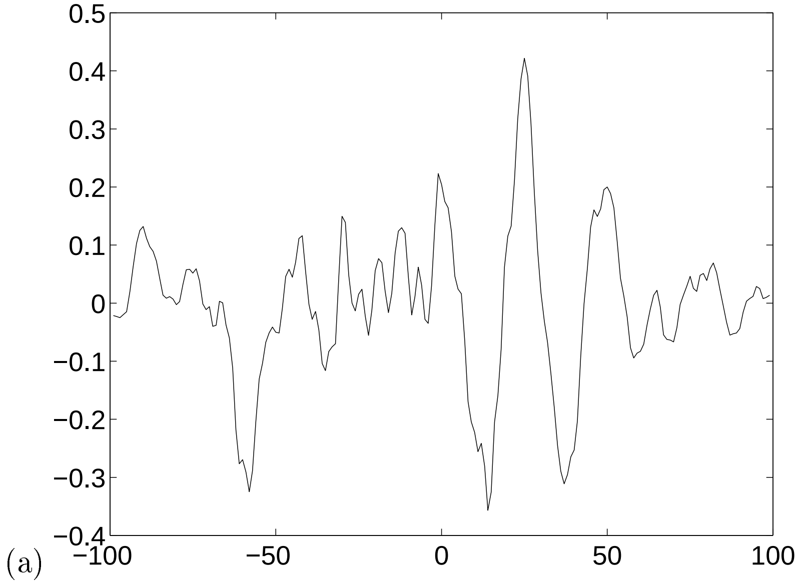

Figure 7.2: *Cross-correlation function $r_{xy}(T)$ for the data in Figure 7.1. A lag of T denotes the top series, x, lagging the bottom series, y. Notice the big positive correlation at a lag of 25. Can you see from Figure 7.1 why this should occur ?*

7.2.3 Time-Delay Estimation

If we suspect that one signal is a, possibly noisy, time-delayed version of another signal then the peak in the cross-correlation will identify the delay. For example, figure 7.1 suggests that the top signal lags the bottom by a delay of 25 samples. Given that the sample rate is 125Hz this corresponds to a delay of 0.2 seconds.

7.3 Multivariate Autoregressive models

A multivariate autoregressive (MAR) model is a linear predictor used for modelling multiple time series. An $MAR(p)$ model predicts the next vector value in a d-dimensional time series, $\boldsymbol{x}_t$ (a *row* vector) as a linear combination of the p previous vector values of the time series

$$\boldsymbol{x}(t) = \sum_{k=1}^{p} \boldsymbol{x}(t-k)\boldsymbol{a}(k) + \boldsymbol{e}_t \tag{7.12}$$

where each $\boldsymbol{a}_k$ is a $d - by - d$ matrix of AR coefficients and $\boldsymbol{e}_t$ is an IID Gaussian noise vector with zero mean and covariance $\boldsymbol{C}$. There are a total of $n_p = p \times d \times d$ AR coefficients and the noise covariance matrix has $d \times d$ elements. If we write the lagged vectors as a single augmented row vector

$$\tilde{\boldsymbol{x}}(t) = [\boldsymbol{x}(t-1), \boldsymbol{x}(t-2), ..., \boldsymbol{x}(t-p)] \tag{7.13}$$

and the AR coefficients as a single augmented matrix

$$\boldsymbol{A} = [\boldsymbol{a}(1), \boldsymbol{a}(2), ..., \boldsymbol{a}(p)]^T \tag{7.14}$$

then we can write the MAR model as

$$\boldsymbol{x}(t) = \tilde{\boldsymbol{x}}(t)\boldsymbol{A} + \boldsymbol{e}(t) \tag{7.15}$$

The above equation shows the model at a single time point t.

The equation for the model over all time steps can be written in terms of the embedding matrix, $\tilde{\boldsymbol{M}}$, whose tth row is $\tilde{\boldsymbol{x}}(t)$, the error matrix $\boldsymbol{E}$ having rows $\boldsymbol{e}(t+p+1)$ and the target matrix $\boldsymbol{X}$ having rows $\boldsymbol{x}(t+p+1)$. This gives

$$\boldsymbol{X} = \tilde{\boldsymbol{M}}\boldsymbol{A} + \boldsymbol{E} \tag{7.16}$$

which is now in the standard form of a multivariate linear regression problem. The AR coefficients can therefore be calculated from

$$\hat{\boldsymbol{A}} = \left(\tilde{\boldsymbol{M}}^T \tilde{\boldsymbol{M}}\right)^{-1} \tilde{\boldsymbol{M}}^T \boldsymbol{X} \tag{7.17}$$

and the AR predictions are then given by

$$\hat{\boldsymbol{x}}(t) = \tilde{\boldsymbol{x}}(t)\hat{\boldsymbol{A}} \tag{7.18}$$

The predicion errors are

$$\boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t) \tag{7.19}$$

and the noise covariance matrix is estimated as

$$\boldsymbol{C} = \frac{1}{N - n_p} \boldsymbol{e}^T(t)\boldsymbol{e}(t) \tag{7.20}$$

The denominator $N - n_p$ arises because n_p degrees of freedom have been used up to calculate the AR coefficients (and we want the estimates of covariance to be unbiased).

7.3.1 Model order selection

Given that an MAR model can be expressed as a multivariate linear regression problem all the usual model order selection criteria can be employed such as stepwise forwards and backwards selection. Other criteria also exist. Neumaier and Schneider [42] and Lutkepohl [34] investigate a number of methods including the Final Prediction Error

$$FPE(p) = \log \sigma^2 + \log \frac{N + n_p}{N - n_p} \tag{7.21}$$

where

$$\sigma^2 = \frac{1}{N} \left[det((N - n_p)\boldsymbol{C})\right]^{1/d} \tag{7.22}$$

but they prefer the Minimum Description Length (MDL) criterion[1]

$$MDL(p) = \frac{N}{2} \log \sigma^2 + \frac{n_p}{2} \log N \tag{7.23}$$

7.3.2 Example

Given two time series and a MAR(3) model, for example, the MAR predictions are

$$\hat{\boldsymbol{x}}(t) = \tilde{\boldsymbol{x}}(t)\boldsymbol{A} \tag{7.24}$$

$$\hat{\boldsymbol{x}}(t) = [\boldsymbol{x}(t-1), \boldsymbol{x}(t-2), \boldsymbol{x}(t-3)] \begin{bmatrix} \boldsymbol{a}(1) \\ \boldsymbol{a}(2) \\ \boldsymbol{a}(3) \end{bmatrix}$$

$$\begin{bmatrix} \hat{x}_1(t) & \hat{x}_2(t) \end{bmatrix} = \begin{bmatrix} x_1(t-1) x_2(t-1) x_1(t-2) x_2(t-2) x_1(t-3) x_2(t-3) \end{bmatrix} \tag{7.25}$$

$$\begin{bmatrix} \hat{a}_{11}(1) & \hat{a}_{12}(1) \\ \hat{a}_{21}(1) & \hat{a}_{22}(1) \\ \hat{a}_{11}(2) & \hat{a}_{12}(2) \\ \hat{a}_{21}(2) & \hat{a}_{22}(2) \\ \hat{a}_{11}(3) & \hat{a}_{12}(3) \\ \hat{a}_{21}(3) & \hat{a}_{22}(3) \end{bmatrix}$$

[1] The MDL criterion is identical to the negative value of the Bayesian Information Criterion (BIC) ie. $MDL(p) = -BIC(p)$, and Neumaier and Schneider refer to this measure as BIC.

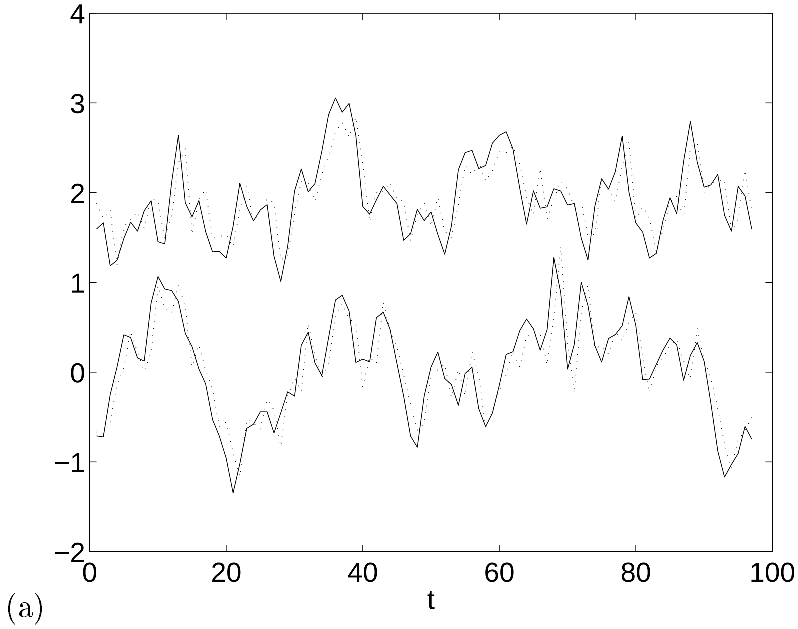

Figure 7.3: *Signals $x_1(t)$ (top) and $x_2(t)$ (bottom) and predictions from MAR(3) model.*

Applying an MAR(3) model to our data set gave the following estimates for the AR coefficients, $\boldsymbol{a}_p$, and noise covariance $\boldsymbol{C}$, which were estimated from equations 7.17 and 7.20

$$\boldsymbol{a}_1 = \begin{bmatrix} -1.2813 & -0.2394 \\ -0.0018 & -1.0816 \end{bmatrix}$$

$$\boldsymbol{a}_2 = \begin{bmatrix} 0.7453 & 0.2822 \\ -0.0974 & 0.6044 \end{bmatrix}$$

$$\boldsymbol{a}_3 = \begin{bmatrix} -0.3259 & -0.0576 \\ -0.0764 & -0.2699 \end{bmatrix}$$

$$\boldsymbol{C} = \begin{bmatrix} 0.0714 & 0.0054 \\ 0.0054 & 0.0798 \end{bmatrix}$$

7.4 Cross Spectral Density

Just as the Power Spectral Density (PSD) is the Fourier transform of the auto-covariance function we may define the Cross Spectral Density (CSD) as the Fourier transform of the cross-covariance function

$$P_{12}(w) = \sum_{n=-\infty}^{\infty} \sigma_{x_1 x_2}(n) \exp(-iwn) \tag{7.26}$$

Note that if $x_1 = x_2$, the CSD reduces to the PSD. Now, the cross-covariance of a signal is given by

$$\sigma_{x_1 x_2}(n) = \sum_{l=-\infty}^{\infty} x_1(l) x_2(l-n) \tag{7.27}$$

Substituting this into the earlier expression gives

$$P_{12}(w) = \sum_{n=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} x_1(l) x_2(l-n) \exp(-iwn) \tag{7.28}$$

By noting that

$$\exp(-iwn) = \exp(-iwl) \exp(iwk) \tag{7.29}$$

where $k = l - n$ we can see that the CSD splits into the product of two integrals

$$P_{12}(w) = X_1(w) X_2(-w) \tag{7.30}$$

where

$$\begin{aligned} X_1(w) &= \sum_{l=-\infty}^{\infty} x_1(l) \exp(-iwl) \\ X_2(-w) &= \sum_{k=-\infty}^{\infty} x_2(k) \exp(+iwk) \end{aligned} \tag{7.31}$$

For real signals $X_2^*(w) = X_2(-w)$ where $*$ denotes the complex conjugate. Hence, the cross spectral density is given by

$$P_{12}(w) = X_1(w) X_2^*(w) \tag{7.32}$$

This means that the CSD can be evaluated in one of two ways (i) by first estimating the cross-covariance and Fourier transforming or (ii) by taking the Fourier transforms of each signal and multiplying (after taking the conjugate of one of them). A number of algorithms exist which enhance the spectral estimation ability of each method. These algorithms are basically extensions of the algorithms for PSD estimation, for example, for type (i) methods we can perform Blackman-Tukey windowing of the cross-covariance function and for type (ii) methods we can employ Welch's algorithm for averaging modified periodograms before multiplying the transforms. See Carter [8] for more details.

The CSD is complex

The CSD is complex because the cross-covariance is asymmetric (the PSD is real because the auto-covariance is symmetric; in this special case the Fourier transorm reduces to a cosine transform).

7.4.1 More than two time series

The frequency domain characteristics of a multivariate time-series may be summarised by the power spectral density *matrix* (Marple, 1987[39]; page 387). For d time series

$$\boldsymbol{P}(f) = \begin{pmatrix} P_{11}(f) & P_{12}(f) & \cdots & P_{1d}(f) \\ P_{12}(f) & P_{22}(f) & \cdots & P_{2d}(f) \\ \multicolumn{4}{c}{\dotfill} \\ P_{1d}(f) & P_{2d}(f) & \cdots & P_{dd}(f) \end{pmatrix} \tag{7.33}$$

where the diagonal elements contain the spectra of individual channels and the off-diagonal elements contain the cross-spectra. The matrix is called a *Hermitian matrix* because the elements are complex numbers.

7.4.2 Coherence and Phase

The *complex coherence function* is given by (Marple 1987; p. 390)

$$r_{ij}(f) = \frac{P_{ij}(f)}{\sqrt{P_{ii}(f)}\sqrt{P_{jj}(f)}} \tag{7.34}$$

The coherence, or *mean squared coherence* (MSC), between two channels is given by

$$r_{ij}(f) =\mid r_{ij}(f) \mid^2 \tag{7.35}$$

The phase spectrum, between two channels is given by

$$\theta_{ij}(f) = tan^{-1}\left[\frac{Im(r_{ij}(f))}{Re(r_{ij}(f))}\right] \tag{7.36}$$

The MSC measures the linear correlation between two time series at each frequency and is directly analagous to the squared correlation coefficient in linear regression. As such the MSC is intimately related to *linear filtering*, where one signal is viewed as a filtered version of the other. This can be interpreted as a linear regression at each frequency. The optimal regression coefficient, or linear filter, is given by

$$H(f) = \frac{P_{xy}(f)}{P_{xx}(f)} \tag{7.37}$$

This is analagous to the expression for the regression coefficient $a = \sigma_{xy}/\sigma_{xx}$ (see first lecture). The MSC is related to the optimal filter as follows

$$r^2_{xy}(f) = |H(f)|^2 \frac{P_{xx}(f)}{P_{yy}(f)} \tag{7.38}$$

which is analagous to the equivalent expression in linear regression $r^2 = a^2(\sigma_{xx}/\sigma_{yy})$.

At a given frequency, if the phase of one signal is *fixed* relative to the other, then the signals can have a high coherence at that frequency. This holds even if one signal is entirely out of phase with the other (note that this is different from adding up signals which are out of phase; the signals cancel out. We are talking about the coherence *between* the signals).

At a given frequency, if the phase of one signal changes relative to the other then the signals will not be coherent at that frequency. The time over which the phase relationship is constant is known as the *coherence time*. See [46], for an example.

7.4.3 Welch's method for estimating coherence

Algorithms based on Welch's method (such as the cohere function in the matlab system identification toolbox) are widely used [8] [55]. The signal is split up into a number of segments, N, each of length T and the segments may be overlapping. The complex coherence estimate is then given as

$$\hat{r}_{ij}(f) = \frac{\sum_{n=1}^{N} X_i^n(f)(X_j^n(f))^*}{\sqrt{\sum_{n=1}^{N} X_i^n(f)^2}\sqrt{\sum_{n=1}^{N} X_j^n(f)^2}} \tag{7.39}$$

where n sums over the data segments. This equation is exactly the same form as for estimating correlation coefficients (see chapter 1). Note that if we have only $N = 1$ data segment then the estimate of coherence will be 1 regardless of what the true value is (this would be like regression with a single data point). Therefore, we need a number of segments.

Note that this only applies to Welch-type algorithms which compute the CSD from a product of Fourier transforms. We can trade-off good spectral resolution (requiring large T) with low-variance estimates of coherence (requiring large N and therefore small T). To an extent, by increasing the overlap between segments (and therefore the amount of computation, ie. number of FFTs computed) we can have the best of both worlds.

7.4.4 MAR models

Just as the PSD can be calculated from AR coefficients so the PSD's and CSD's can be calculated from MAR coefficients. First we compute

$$\boldsymbol{A}(f) = \boldsymbol{I} + \sum_{k}^{p} \boldsymbol{a}_k \exp(-ik2\pi fT) \tag{7.40}$$

where $\boldsymbol{I}$ is the identity matrix, f is the frequency of interest and T is the sampling period. $\boldsymbol{A}(f)$ will be complex. This is analagous to the denominator in the equivalent AR expression $(1 + \sum_{k=1}^{p} a_k \exp(-ik2\pi ft))$. Then we calculate the PSD matrix as follows (Marple 1987 [39]; page 408)

$$\boldsymbol{P}_{MAR}(f) = T\left[\boldsymbol{A}(f)\right]^{-1} \boldsymbol{C}\left[\boldsymbol{A}(f)\right]^{-H} \tag{7.41}$$

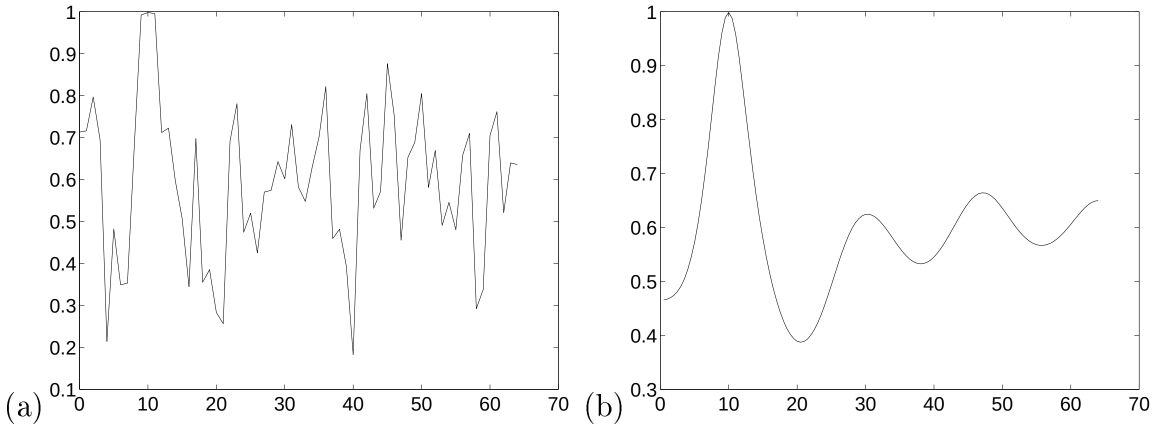

Figure 7.4: *Coherence estimates from (a) Welch's periodogram method and (b) Multivariate Autoregressive model.*

where $\boldsymbol{C}$ is the residual covariance matrix and H denotes the Hermitian transpose. This is formed by taking the complex conjugate of each matrix element and then applying the usual transpose operator.

Just as $\boldsymbol{A}^{-T}$ denotes the transpose of the inverse so $\boldsymbol{A}^{-H}$ denotes the Hermitian transpose of the inverse. Once the PSD matrix has been calculated, we can calculate the coherences of interest using equation 7.35.

7.5 Example

To illustrate the estimation of coherence we generated two signals. The first, x, being a 10Hz sine wave with additive Gaussian noise of standard deviation 0.3 and the second y being equal to the first but with more additive noise of the same standard deviation. Five seconds of data were generated at a sample rate of 128Hz. We then calculated the coherence using (a) Welch's modified periodogram method with $N = 128$ samples per segment and a 50% overlap between segments and smoothing via a Hanning window and (b) an MAR(8) model. Ideally, we should see a coherence near to 1 at 10Hz and zero elsewhere. However, the coherence is highly non-zero at other frequencies. This is because due to the noise component of the signal there is power (and some cross-power) at all frequencies. As coherence is a ratio of cross-power to power it will have a high variance unless the number of data samples is large.

You should therefore be careful when interpreting coherence values. Preferably you should perform a significance test, either based on an assumption of Gaussian signals [8] or using a Monte-Carlo method [38]. See also the text by Bloomfield [4].

7.6 Partial Coherence

There is a direct analogy to partial correlation. Given a target signal y and other signals $x_1, x_2, ..., x_m$ we can calculate the 'error' at a given frequency after including $k = 1..m$ variables $E_m(f)$. The partial coherence is

$$k_m(f) = \frac{E_{m-1}(f) - E_m(f)}{E_{m-1}(f)} \tag{7.42}$$

See Carter [8] for more details.