Lecture 13: Frequency/Wavenumber Spectra

Recap

We've looked at multiple strategies for computing spectra in the frequency domain or by extension in the wavenumber domain. We've considered uncertainties, resolution and multiple methods. Now, we'll wrap up some details on spectra computed from the autocovariance, and we'll also consider what happens when we want to consider sinusoidal patterns of variability in both time and space.

Spectra from the autocovariance

Last time we looked at the autocovariance of white noise:

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} x(t)x(\tau + t)dt = \begin{cases} 0 & \text{for } \tau \neq 0 \\ 1 & \text{for } \tau = 0 \end{cases} \tag{1}$$

If we generate red noise, as you did for the problem set, then the autocovariance will have more structure:

```
% define red data with autoregressive process
a=randn(10000,1);
b(1)=a(1);
for i=2:length(a)
  b(i)=b(i-1)+a(i);
end

BB=xcorr(b,b)/max(xcorr(b,b));
BB_unbiased=xcorr(b,b,'unbiased')/max(xcorr(b,b,'unbiased'));
plot(-9999:9999,[BB' BB_unbiased'],'LineWidth',3)
```

Biased vs unbiased estimators.

Notice that we could consider both the 'biased' and the "unbiased" estimator. There are arguments for either choice. The difference depends on how we normalize our discrete autocovariance. In the unbiased case, we're computing

$$R(\tau)_{unbiased} = \frac{1}{N-m} \sum_{n=1}^{N-m} x(t_n)x(t_{n+m}). \tag{2}$$

In the biased case, we change how we normalize:

$$R(\tau)_{biased} = \frac{1}{N} \sum_{n=1}^{N} x(t_n)x(t_{n+m}), \tag{3}$$

which means that as the number of values we consider becomes smaller, we constrain the magnitude of the autocovariance by continuing to divide by N. Emery and Thomson note that the biased estimator acts like a triangle window. For spectra, you probably want to stick to the unbiased estimator, and then choose your own window carefully.

When we use the autocovariance to compute spectra, we'll want to omit the poorly sampled edges of the spectrum. There are a number of ways to do this. We have to decide what fraction of the autocovariance to use. For example I could take half, or a quarter. What difference does it make? In essence, the fraction that I use determines the amount of averaging that I do, and therefore defines the number of degrees of freedom. In the simplest form, we use a boxcar window to extract values:

```
N=length(b);  % number of data points
M=N/4;  % half width of points to use
fBB=fft(BB(N-M+1:N+M));  % since the fft assumes the record to
              % be circular, remove one point at the end

loglog(0:M-1,abs(fBB(1:M)))
```

The degrees of freedom are determined by the width of the window. If I have N data points, and I use M of them in the Fourier transform, then I'm implicitly averaging N/M independent samples, which gives me N/M degrees of freedom. I can compute an error bar based on this in exactly the way that we did earlier for the periodogram approach:

```
hold on
nu=N/M;
err_high=nu/chi2inv(.05/2,nu);
err_low=nu/chi2inv(1-.05/2,nu);

loglog([10 10],[err_high err_low],'LineWidth',3)
```

Now you might decide that a boxcar windowed view of the autocovariance is likely to have unfortunate characteristics in the Fourier domain, in which case you could use a different window instead: a triangle window (also called a Bartlett window), or a Hanning window, or a Hamming window. For example:

```
fBB_t=fft(BB(N-M:N+M).*triang(2*M+1)');
fBB_h=fft(BB(N-M:N+M).*hanning(2*M+1)');

loglog(0:M-1,abs(fBB_t(1:M)))
loglog(0:M-1,abs(fBB_h(1:M)))
```

In this case we adjust the error bars to account for the windowing of the autocovariance. As it turns out, the table (5.5 in Thomson and Emery, 2014) that was so misleading for overlapping segmented data is exactly what we need here:

Window type	Equivalent degrees of freedom (ν)
Truncated peridogram (boxcar)	N/M
Bartlett (triangle)	$3N/M$
Daniell (sinc)	$2N/M$
Parzen	$3.708614N/M$
Hanning	$8/3N/M$
Hamming	$2.5164N/M$

This works well when M is small compared with N and when the autocovariance is comparatively narrow. In other words, for white noise, this converges nicely; for a broad red noise peak, the autocovariance tends to have negative lobes, making the choice of M difficult and the results not amenable to interpretation. The challenge in dealing with the truncation of the autocovariance is perhaps the reason that although this approach is often presented in equations (in idealized cases with infinite data), it is less frequently implemented for real applications.

Frequency-Wavenumber examples.

First consider some example frequency-wavenumber spectra. What frequencies and wavenumbers are resolved? What is plotted? What is the Nyquist frequency and Nyquist wavenumber? Is the full frequency-wavenumber space represented?

As a reminder, frequency represents cycles per unit time, and wavenumber represents cycles per unit distance.

We use frequency-wavenumber spectra as a means to track propagating sinusoidal patterns. If it has a characteristic wavelength and frequency, we might suppress that variability if we didn't think about the full structure of the propagating wave. Westward propagating Rossby waves and eastward propagating Kelvin waves have characteristic frequencies and characteristic wavenumbers.