Lecture 10:

Reading: Bendat and Piersol, Ch. 5.2.1

Recap

Last time we looked at uncertainties and how to label units on our spectra, and we considered the sinc function and its impact on spectral segmenting. Now we need to look closely at windowing.

Windowing our data

We left off saying that it would be useful to find a window (or "taper") that could minimize the impact of discontinuities in our spectrum, avoiding a rectangular window in favor of something with a moother Fourier transform. What if we chose a triangle window? That will already give us fewer side lobes.

But we can keep going to find a window that looks more like an exponential or a Gaussian. All of these are defined to be centered around 0, for $|t| \leq T$. We looked at a couple, and we'll add more now:

1. Cosine taper:

$$w(t) = \cos^{\alpha}\left(\frac{\pi t}{2T}\right) \tag{1}$$

   with $\alpha = [1, 4]$.

2. Hanning window or "raised cosine" window (developed by von Hann):

$$w(t) = \cos^2\left(\frac{\pi t}{2T}\right) = \frac{1 + \cos(\pi t/T)}{2} = 0.5 + 0.5\cos(\pi t/T) \tag{2}$$

3. Hamming window. This variant of the Hanning window was developed by Hamming.

$$w(t) = 0.54 + 0.46\cos(\pi t/T) \tag{3}$$

   The Hamming window has less energy in the first side lobe but more in the distant side lobes.

Some other options include a Blackman-Harris window or a Kasier-Bessel window, and Harris (1978, Use of windows for harmonic analysis, *Proc. IEEE*) provides detailed discussion of options.

So how do you use a window?

1. First you must demean your data—otherwise, the window will shift energy from the mean into other frequencies. If you're working in segments, you should demean (and detrend) each segment before you do anything further.

2. Second, for a segment with N points, multiply by a window that is N points wide.

3. Since the window attentuates the impact of the edge of each segment, you can use segments that overlap (typically by 50%). This will give you (almost) twice as many segments, so instead of ν degrees some larger number.

4. Now Fourier transform, scale appropriately (e.g. by $\sqrt{8/3}$ for a Hanning window, to account for energy attenuation) and compute amplitudes.

Will the window preserve energy in your system? Not necessarily. You can normalize it appropriately, but windowing can shift the background energy level of your spectrum relative to the spectral peaks, and you'll want to keep track of this.

Exercises

What are the Fourier transforms of the following functions: $x(t) = 1, x(t) = \cos(2\pi ft), x(t) = \sin(2\pi ft), x(t) = \exp(-t^2/2\sigma^2), x(t) = t$?

In our examples, note that the Fourier transforms of single frequency sine and cosine give a single peak. (This uses the Kronecker delta, δ_{nm}.) The Gaussian ($x_f(t)$), has a transform of a Gaussian, though it's a bit distorted in this finite-length log-log domain. Formally if $x_f(t) = e^{-t^2/(2\sigma^2)}$, then $X_f(f) = \sqrt{2\pi}\sigma e^{-2\pi^2 f^2 \sigma^2}$ or with coefficients $x_f(t) = \sqrt{\alpha/\pi}e^{-\alpha t^2}$ corresponds to $X_f(f) = e^{-\pi^2 f^2/(a)}$, which says in essence that the Fourier transform of a Gaussian is still a Gaussian. The normalization for this is dependent on our exact notation for the Fourier transform. And the linear pattern should really be thought of as a repeating sawtooth. It's Fourier transform will is a dramatically red spectrum. What does this mean for the Fourier transform of any long-term trend?

Windowing and overlapping segments

So a quick recap. When we filter, we convolve the filter with our data in the time domain, which is equivalent to multiplying in the frequency domain. When we window, we multiply by a tapered window in the time domain, which is equivalent to convolving in the frequency domain.

How many degrees of freedom do you have for overlapping windows. Not 2ν but close to that. Bendat and Piersol usefully say that overlapping by 50% will recover about 90% of the stability lost due to tapering.

The windowed spectra that we've discussed as a basic approach in class are referred to by a number of names in the literature. Some textbooks refer to windowing as "tapering". And the formal method is sometimes called the Welch method. Percival and Walden (*Spectral Analysis for Physical Applications*, Cambridge University Press, 1993) provide a detailed discussion of this approach under the name "Welch's Overlapping Segment Averaging (WOSA)".

Our overarching concern is that we adhere to Parseval's theorem. Since the window default normalization can reduce the variance, this will typically require that we scale up appropriately to conserve energy/variance (e.g. by $\sqrt{8/3}$ for a Hanning window). How do you find this scale factor? One way is to sum over a wide window. For example:

```
N=10000;
[sum(hanning(N).^2)/N 3/8]
[sum(hamming(N).^2/N 0.3974]
```

Will the window preserve energy in your system? Not necessarily. You can normalize it appropriately, but windowing can shift the background energy level of your spectrum relative to the spectral peaks, and you'll want to keep track pay attention to the possibility of biases resulting from the windowing procedure.

How many degrees of freedom do we have?

We've got a full recipe, but how many degrees of freedom do we really have?

We'll try to sort this out today, along with venturing into Monte Carlo simulation.

Once you've created overlapping, windowed segments, then you'll need to figure out how many independent segments you really have. Clearly at a minimum you should have the equivalent

of the number of segments that you would have if you did no overlapping. If you have N data points divided into segments that are $2M$ wide, then the minimum number of segments is $N/(2M)$. But with windowing, the end points of each segment are used less than the middle, making the overlapping segments more independent, so perhaps we have more degrees of freedom. Since the segments overlap by 50%, we probably can't treat them as being independent. Bendat and Piersol usefully say that overlapping by 50% will recover about 90% of the stability lost due to tapering. How much does the use of overlapping segments modulate the degrees of freedom?

If you look this up in the 2nd edition of Emery and Thomson, you find a nice table (their Table 5.6.4), ostensably lifted from Priestley. Priestley's results are nicely discussed by Koopmans (*The Spectral Analysis of Time Series*, Academic Press, 1974). Emery and Thomson describe the table as representing the equivalent degrees of freedom for windowed "block averaged" spectra.

A little further exploration of the literature shows that the values in Emery and Thomson's table are incorrectly labeled and actually represent degrees of freedom for spectra determined by filtering or averaging adjacent frequencies from an initial spectral estimate. If you filter in frequency, you can increase your degrees of freedom just the way you increase degrees of freedom by using multiple segments. And you can refine your filtering to reduce spectral ringing effects by using a carefully constructed filter. While people use the same types of filters for time domain windowing/tapering and for frequency domain filtering, the effect is not identical. (If we wanted an identical effect, we'd need to use the Fourier transform of the time domain window to filter in the frequency domain. You could do that, but it's not what we usually envision when we talk about windowing or tapering, and it's not what Emery and Thomson seem to describe.) What this means is that the tables of "equivalent degrees of freedom" for the frequency domain don't actually work for windowed time domain data.

So at this point we have three options:

1. Abandon windowing and filter in the frequency domain, so that we can take advantage of the correct estimates of degrees of freedom. This is a possibility, and we'll get to it, but it seems like we're unnecessarily giving up on the windowing approach that we've been exploring.

2. Use a brute strength approach—Monte Carlo simulation—to figure out how many degrees of freedom we have.

3. Find an analytic solution to decide how many degrees of freedom we have.

Monte Carlo simulation: How to avoid the traps imposed by standard statistical assumptions (and how to fake your way as a statistician through computational inefficiency rather than clever mathematics)

Most of the time, we estimate spectral error bars using basic statistical assumptions—that data are normally distributed, that we have enough samples for the central limit theorem to apply, that statistics are stationary. These assumptions make our statistical models tractable—we end up with equations we can manipulate, allowing us (or clever statisticians 100 years ago) to derive simple equations that give us rules for how systems should behave. But what happens when those assumptions break down? Or what happens when we have little doubts about the validity of the statistical model. We can always resort to a Monte Carlo process. In Monte Carlo methods, we throw theory on its head and use an empirical approach to generate many realizations of our data set, with noise appropriate to our problem.

As an example, consider the problem of determining the standard error of the mean. When we discussed it in class, we did a quick derivation to show that the standard error of the mean is

$\sigma/\sqrt{N}$, where σ is the standard deviation and N is the effective degrees of freedom. But what if I didn't trust this realization? I could generate a large number of realizations of my data with noise typical of the real data, compute means for each realization, and look at the statistics of those values.

So let's put this to work. Suppose I'm computing the mean of $N = 500$ data points. With one sample, I can compute the mean μ and standard deviation σ, and standard error $\sigma/\sqrt{500}$. But I might wonder if μ is really representative. So I can generate an ensemble of fake data, perhaps 100 data sets based on adding Gaussian white noise (or non-Gaussian white noise) to the real data. Each of these data sets will have a mean μ_i and a standard deviation σ_i. And I can look at the standard deviation of all of the μ_i values. I can also look at the pdf of my μ_i's and other higher order statistics. For example:

```
A=randn(500,100);
mu=mean(A);
sigma_A=std(A);
std_A=sigma_A/sqrt(500);
[std(mu) mean(std_A)] % compare standard deviation of means
          %  vs standard error
```

Now we could expand on our example and ask, what if our noise were non-Gaussian or gappy or had other problems, and we could adjust our Monte Carlo process appropriately. And importantly, we can use this approach to test windowing strategies, by generating fake data with a white spectrum (or a known red spectrum), windowing, segmenting, and evaluating the statistics of the solution—the standard deviation in log space—relative to the formal error bar.

A more analytic approach to degrees of freedom for overlapping segments

If the Monte Carlo approach seems too ad hoc, we can try something else. Percival and Walden provide a full derivation of the actual degrees of freedom for overlapping segments. They define the following terms:

$$\begin{aligned} N &= \text{total length of record} \\ N_B &= \text{number of blocks} \\ N_S &= \text{segment length or block size} \\ n &= \text{shift factor or number of points of overlap between segments} \\ h &= \text{window, normalized so that } h^2 \text{ sums to 1.} \end{aligned}$$

They point out that the covariance between adjacent segments determines the adjustment to the degrees of freedom, and this depends on h.

In this framework, the variance of the spectral estimate is:

$$\text{var}\left\{\hat{S}^{(WOSA)}(f)\right\} = \frac{1}{N_B}\sum_{j=0}^{N_B-1}\left(\text{var}\left\{\hat{S}_{jn+1}(f)\right\} + \frac{2}{N_B}\sum_{j<k}\text{cov}\left\{\hat{S}_{jn+1}(f), \hat{S}_{kn+1}(f)\right\}\right), \quad (4)$$

where j and k are indices for separate but overlapping segments. The variance of the jth spectrum should converge to the canonical spectrum:

$$\text{var}\left\{\hat{S}_{jn+1}(f)\right\} \approx S^2(f). \quad (5)$$

The covariance depends on the overlap of the tapers or windows:

$$\text{cov}\left\{\hat{S}_{jn+1}(f), \hat{S}_{kn+1}(f)\right\} \approx S^2(f)\left|\sum_{t=1}^{N_S} h_t h_{t+|k-j|n}\right|^2, \tag{6}$$

with h_t defined to be zero when t is out of range (i.e. $t > N_S$). This means that:

$$\begin{aligned}\text{var}\left\{\hat{S}^{(WOSA)}(f)\right\} &\approx S^2(f)\left(1 + \frac{2}{N_B}\sum_{j<k}\left|\sum_{t=1}^{N_S} h_t h_{t+|k-j|n}\right|^2\right) &(7)\\ &= S^2(f)\left(1 + 2\sum_{m=1}^{N_B-1}\left(1-\frac{m}{N_B}\right)\left|\sum_{t=1}^{N_S} h_t h_{t+mn}\right|^2\right). &(8)\end{aligned}$$

Here's a spoiler. Either analytically, or by plugging in normalized discrete window values h_t, we can compute the adjustments to our effective degrees of freedom shown in Table **??**. You'll see that these values provide a fairly effective match to the values that you obtain from Monte Carlo simulation.

Window type	Equivalent degrees of freedom (ν)
Boxcar	4/3
Triangle	16/9
Hanning	36/19 $\approx$ 1.90
Hamming	$\sim$1.80

Table 1: Effective number of degrees of freedom relative to the total number of segments, using 50% overlap. (With no overlap, the equivalent degrees of freedom would be double the number of segments.)

To be continued....