## Lecture 7:

*Reading: Bendat and Piersol, Ch. 8.5.4, Ch. 5.2.3*

*Recap*

Last time we looked at the Fourier transform, which lets us re-represent data in the time (or space) domain in terms of coefficients of sines and cosines. We looked at notation, and we talked about 3 really key concepts for Fourier transforms:

1. First derivatives in the time domain (or the space domain) can be represented in the frequency (or wavenumber) domain by simple multiplication (e.g. by $i2\pi f$).

2. Convolution, which is what we do when we filter data, can be carried out in the frequency (or wavenumber) domain through simple multiplication. The convolution can be written:

$$x * y(\tau) = \int_{-\infty}^{\infty} x(t)y(\tau - t)\,dt, \tag{1}$$

and its Fourier transform is:

$$X(f)Y(f) = \int_{-\infty}^{\infty} x * ye^{-i2\pi ft}\,dt. \tag{2}$$

3. Parseval's theorem: Total variance in the time domain equals total variance in the frequency domain

$$\int_{-\infty}^{\infty} x^2(t)dt = \int_{-\infty}^{\infty} |X(f)|^2\,df. \tag{3}$$

*More on the formalism of the Fourier transform*

Now we can use this to verify that our Fourier coefficients are consistent. If I have a data set $x(t)$ that can be expressed as

$$x(t) = \sum_{n=-\infty}^{\infty} a_n \exp(i2\pi f_n t) \tag{4}$$

and

$$a_m = \frac{1}{T}\int_{-T/2}^{T/2} x(t)\exp(-i2\pi f_m t)\,dt \tag{5}$$

then let's check that our coefficients work out. We can substitute in $x(t)$ to obtain

$$\begin{aligned} a_m &= \frac{1}{T}\int_{-T/2}^{T/2}\sum_{n=-\infty}^{\infty} a_n \exp(i2\pi f_n t)\exp(-i2\pi f_m t)\,dt & (6)\\ &= \frac{1}{T}\int_{-T/2}^{T/2}\sum_{n=-\infty}^{\infty} a_n \exp(i2\pi(f_n - f_m)t)\,dt & (7)\\ &= \sum_{n=-\infty}^{\infty}\frac{a_n}{T}\int_{-T/2}^{T/2}\sum_{n=-\infty}^{\infty}\exp(i(f_n - f_m)t)\,dt. & (8)\end{aligned}$$

When $n = m$, the integral goes to $T$, and the summed expression becomes $a_n$. When $n \neq m$, we're dealing with orthogonal cosines and sines, and the integral goes to zero. Thus the net result is that

$$a_m = \sum_{n=-\infty}^{\infty} a_n \delta_{nm} \tag{9}$$

$$= a_m \tag{10}$$

where $\delta_{nm}$ is called the Kronecker delta function, with $\delta_{nm} = 1$ if $n = m$ and $\delta_{nm} = 0$ otherwise. (Formally in continuous form the $\delta$ function can be thought of as a distribution, like a pdf, that has shrunk to be infinitely high and infinitesimally narrow, so that the area under the distribution is exactly 1.)

*3. Parseval's theorem: Total variance in the time domain equals total variance in the frequency domain*

Parsval's theorem provides a critical link between total energy in the time domain and total energy in the Fourier transform domain.

*3.1 Parseval's theorem via convolution.* Let's start with the convolution of a data record with itself:

$$y(t) = \int_{-\infty}^{\infty} x(\tau)x(t-\tau)d\tau. \tag{11}$$

What happens if I convolve my data ($x(t)$) with the time reversal of itself ($x(-t)$)?

$$y(t) = \int_{-\infty}^{\infty} x(\tau)x(t+\tau)d\tau. \tag{12}$$

More conventionally we might write:

$$y(\Delta t) = \int_{-\infty}^{\infty} x(t)x(\Delta t + t)dt. \tag{13}$$

So we're looking at the data multiplied by itself for a time lag $\Delta t$. At zero lag, this is the variance, and as we vary $\Delta t$ we're looking at the lagged covariance for different time lags.

What about the Fourier transform? The Fourier transform of a convolution is simply the product of the Fourier transforms of each variable so the Fourier transform of $y(\Delta t)$ shoud be $X^*X$. (We use the complex conjugate of $X$ since we convolved $x$ with its time reversal.) We could inverse transform this back to produce $y(\Delta t)$:

$$y(\Delta t) = \int_{-\infty}^{\infty} X^*X e^{i2\pi f \Delta t}\, df \tag{14}$$

Now, focus on the case when $\Delta t = 0$. This implies that

$$y(0) = \int_{-\infty}^{\infty} x(t)^2 dt = \int_{-\infty}^{\infty} X^*X\, df, \tag{15}$$

which tells us that the total variability in $x$ is equivalent to the total variability in its Fourier transform $X$.

*3.2 Parseval's theorem from the definition of the Fourier transform.* Maybe a clearer way to understand Parseval's theorem is to think about the product of two variables, $x_1$ and $x_2$. We can rewrite the product, substituting the inverse Fourier transform of the Fourier transform of $x_2(t)$:

$$x_1(t)x_2(t) = x_1(t) \int_{-\infty}^{\infty} X_2(f) e^{i2\pi f t}\, df \tag{16}$$

so we can integrate this in time:

$$\int_{-\infty}^{\infty} x_1(t)x_2(t)dt = \int_{-\infty}^{\infty} \left[ x_1(t) \int_{-\infty}^{\infty} X_2(f) e^{i2\pi ft}\, df \right] dt \tag{17}$$

$$= \int_{-\infty}^{\infty} X_2(f) \left[ \int_{-\infty}^{\infty} x_1(t) e^{i2\pi ft} dt \right] df \tag{18}$$

$$= \int_{-\infty}^{\infty} {X_1}^*(f) X_2(f)\, df. \tag{19}$$

(My edition of Bendat and Piersol has a typo in this derivation, which appears just prior to equation 5.83, and this has caused no end of confusion.) Here we use the complex conjugate of the Fourier transform of $x_1$, because we computed the Fourier transform with $e^{+i2\pi ft}$ instead of the standard $e^{-i2\pi ft}$.

Put succinctly, if $x_1 = x_2$:

$$\int_{-\infty}^{\infty} x^2(t)dt = \int_{-\infty}^{\infty} |X(f)|^2\, df \tag{20}$$

This is Parseval's relationship.

It's worth noting that if we worked with $\sigma = 2\pi f$ rather than $f$, we'd have to normalize by $2\pi$:

$$\int_{-\infty}^{\infty} x^2(t)dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(\sigma)|^2\, d\sigma \tag{21}$$

In thinking about the time domain vs the frequency domain, one thing to keep in mind is the distinction between integrating over all time (on the left in the above equation) and integrating over all space (on the right). This implies that we're going to need to keep track of our frequency information carefully. In essence the Fourier coefficients in $X$ (e.g. $|a_m|^2$) do not have the same units as the time domain values in $x^2$, because $x$ is integrated in time and $|a_m|$ is integrated in frequency. If the total integral of $x^2$ is equal to the total integral of $|X|^2$, then we're going to need to adjust by factors of $\delta f$, and this will influence how we label our axes.

*Red, white, and blue spectra*

Now let's look at a few spectra. We use words associated with light to talk about spectra. Red colors have long wavelengths (e.g. infrared), while blues and purples have short wavelengths (e.g. ultraviolet). If a spectrum is dominated by low frequencies or long wavelengths, we refer to it as "red". If it is dominated by short wavelengths or high frequencies, is is "blue". If it has nearly the same energy levels at all frequencies or wavelengths, then it is "white", like the white broad-spectrum lights that we use for electric lighting.

*Computing spectra*

We've covered a lot of territory—we can Fourier transform with an fft, we know some of the properties of the Fourier transform, we've looked at Parseval's theorem. Now we need to stop beating around the bush and produce some spectra of our own.

How do we take our data and produce a meaningful measure of the power per unit frequency? Here's a basic approach:

1. First, we know we're going to need to Fourier transform our data, and plot the squared amplitudes. We'll only need to analyze the first $N/2 + 1$ of the Fourier coefficients, and

we'll look at the amplitudes of these values. (The second half of the Fourier coefficients are complex conjugates of the first half of the record and correspond to negative frequencies.) The frequencies corresponding to the first $N/2+1$ coefficients will run from 0 cycles per $N$ points to $N/2$ cycles per $N$ points. So a first step is to compute:

```
a=fft(data)
N=length(data);
amp=abs(a(1:N/2+1).^2; % for even N
amp=abs(a(1:(N+1)/2).^2; % for odd N
```

2. Second, since we're only taking half the record, we've thrown out half the energy in the original data (except at frequency 0), so we'll need to put that back in.

   ```
   amp(2:end-1)=2*amp(2:end-1); % for even N
   amp(2:end)=2*amp(2:end); % for odd N
   ```

3. Having gotten this far, we'd better check that Parseval's theorem is working for us. We'll need to see that the energy in the initial record matches the energy in the Fourier transform. If we don't worry about units, then in Matlab we just need to divide by $N$ to make our normalization work.

   ```
   amp=amp/N;
   ```

   If we want to make this make sense for the temporal and spatial sampling of our observations we might need to scale our results to reflect real time and frequency units. (Fortunately multiplicative scalings won't influence the shape of our spectra, so we can always delay sorting out the details.)

4. We can plot what we have. Typically we use a loglog axis (or at least semilogy). We'll need to compute our frequencies thoughtfully. So for example if we have measurements every 361 seconds, then we might want to convert to cycles per hour by scaling the frequency appropriately.

   ```
   scale = 1/361 * 3600; % to convert from 1 cycle/2 pts to 0.5 cycles
                          % per 361 s to 0.5 cycles * 3600/361 /hr
   frequency=(0:N/2)/N * scale; % for N even
   frequency=(0:N/2-1)/N * scale; % for N odd
   loglog(frequency,amp)
   ```

5. Now, we have a problem in that our results are way too noisy. We'll have a hard time distinguishing signal from peak. So clearly we're going to need more realizations. To do this, one common practice is chop our data into multiple segments. As a first step, we can just cut the data into $M$ segments of $N/M$ points each. For example, here's a brute strenght approach:

   ```
   N=length(data); M=6; p=N/M;
   data=reshape(data,N/M,M); % this gives us an array with N/M points
             % per column and M columns
   b=fft(data);  % this computes the fft for each column
   amp_b=abs(b(1:p/2+1,:)).^2; amp_b(2:p/2,:)=2*amp_b(2:p/2,:)/p;
   loglog(frequency,mean(amp_b,2))
   ```

Here you can debate whether you should be plotting the sum (to conserve energy) or the average (to represent a mean spectrum for a data set of length $p$. In most cases you'll want the average.