The remaining variance of the total lightning signal is just one minus the \( \text{varExplained} \) calculated here. Now, to remove the influence of the diurnal cycle from your lightning signal, just subtract the diurnal fit from the total lightning signal:

\[
\text{IDL> residual} = \text{lightning} - \text{trend}
\]

By definition, the residual has no influence from the diurnal cycle, and you can now compare it with the hurricane to assess the correlation between the two. Plot the residual along with the hurricane time series to check their similarity, then quantify this similarity with regression:

\[
\text{IDL> newfit} = \text{poly_fit(hurricane, residual, 1)} \\
\text{IDL> newVarExplained} = (\text{StdDev(hurricane)} \times \text{fit}[1] / \text{StdDev(residual)})^2
\]

So now the fraction of the variance of the residual lightning signal explained by the hurricane is \( \text{newVarExplained} \), and the fraction of the total variance of the total (i.e., not the residual) lightning signal explained by the hurricane is \( (1 - \text{varExplained}) \times \text{newVarExplained} \).

Of course, there may be other forcings (apart from the diurnal cycle and the hurricane) that are affecting the lightning. It’s also a good idea to regress the hurricane against the diurnal cycle and see how much variance of the hurricane is accounted for by the diurnal cycle and/or compare both the residual hurricane and residual lightning signals against each other. That is, remove the diurnal cycle from both the hurricane and the lightning before you compare them to each other. Frankly, you’ll probably want to do all of the above in order to get a better idea of which forcings are most important and dominant in both signals.

### 3 Fourier Analysis

Fourier analysis is an extremely powerful technique and has a huge range of applications. There are entire textbooks devoted to it. I’ll address only the very small part of it that applies most directly to the regression techniques discussed above. This application is often called harmonic analysis. You basically think of your signal as being a summation of many, many harmonic functions, each with a different temporal or spatial scale. The most familiar form is a sound or electromagnetic wave that is composed of many different frequencies which add up as sinusoids to give the full signal. The Fourier transform gives you coefficients which quantify how much each of the component harmonics contribute to the full signal.

For regression applications, I find it most instructive to think of harmonic analysis as simply the regression of your signal against each of the harmonics, and the Fourier coefficients as the regression and/or correlation coefficients. These coefficients then give you the variance explained by each frequency.

Say you have a temporal signal \( y \) with \( N \) samples, and you wish to regress it against the set of harmonic functions \( X \) which have the form:

\[
\cos(2\pi k \frac{t}{T}), \ k = 1, 2, 3, \ldots
\]

and

\[
\sin(2\pi k \frac{t}{T}), \ k = 1, 2, 3, \ldots
\]
Your signal may be expressed as the composition:

\[ y = a_0 + \sum_{k=1}^{N/2} A_k \cos\left(2\pi k t / T\right) + \sum_{k=1}^{N/2} B_k \sin\left(2\pi k t / T\right) \]  

(16)

where \( N \) is the number of samples, \( t \) is the time step between samples, \( T \) is the total time interval, and \( A_k, B_k \) are the regression coefficients for each harmonic. Since the average of sines and cosines over the interval \([0, T]\) is zero, \( \overline{y} = a_0 \).

For \( k = 1 \), the function completes \( 2\pi \) radians over the period \([0, T]\). This is the lowest resolvable frequency. For \( k = N/2 \), the function completes \( \pi \) radians in one time step. This is the highest resolvable frequency, also known as the **Nyquist frequency**. Each of the components of \( X \) have variance of \( \sigma_k^2 = 0 \), except for \( A_{N/2} \) and \( B_{N/2} \) which have variance of 1 and 0, respectively. Think of the sine wave at \( k = N/2 \): since it completes \( \pi \) radians in one time step, it has the same value at \( t \) as at \( t+dt \), hence it’s variance is zero. The cosine wave for \( k = N/2 \) has equal magnitude but opposite sign from \( t \) to \( t+dt \), so it’s variance is 1. I’ll avoid getting into more detail here.

The **power spectrum** is the plot of components versus frequency, i.e., defining:

\[ C_k^2 = \frac{A_k^2}{2} + \frac{B_k^2}{2}, \quad k = 1 \ldots N/2 - 1 \]

\[ C_{N/2}^2 = A_{N/2}^2 \]

the power spectrum is a plot of \( C_k^2 \) versus \( k \).

The important thing is that this analysis also gives you the fraction of variance explained by each component:

\[ r^2(y, k) = \frac{C_k^2}{2\sigma_y^2} \]

(17)

where \( \sigma_y^2 \) is the total variance of \( y \).

### 3.1 Fourier analysis using IDL

Rather than write your own harmonic analysis code, you should use the canned FFT routines that most programming languages provide. FFT stands for Fast Fourier Transform. Look at the IDL documentation for the FFT function for more information. In a nutshell, you compute the power spectrum of your \( N \) element input data like this:

IDL> `spectrum = FFT(input_data)`

This returns a complex array with \( N \) elements. The first element (spectrum[0]) is the mean, i.e., \( a_0 \) in Equation 16. The remaining elements, up to index \( N/2 \) contain the power in the positive frequencies in ascending order. Past \( N/2 \), the elements contain the negative frequencies in descending order. In practice, I convert the spectrum to power using only the positive frequencies:

IDL> `power = (ABS(spectrum[0:N/2]))^2`
And, since I typically deal with FFTs when I’m doing regression analysis, I convert this power to variance explained at each frequency. Note that I do not multiply the Nyquist by 2:

IDL> power[0:N/2-1] = 2.0*power[0:N/2-1]
IDL> bandwidth = 2.0*!DPI/N
IDL> variance = power/(bandwidth*Total(power))

Note that the sum of this variance (when multiplied by the bandwidth) should equal one, i.e., all of the variance should be explained, i.e., doing this:

IDL> print,Total(variance)*bandwidth

should give an answer of one.

The tricky part is determining what frequencies this power spectrum or variance spectrum corresponds to. All you have to remember is that the elements of the power spectrum array correspond to the following frequencies:

- spectrum[0] = the mean
- spectrum[1] = one cycle per total time range, T
- spectrum[2] = two cycles per total time range, T
- spectrum[N/2] = Nyquist frequency, N/2 cycles per total time range, T

Basically, all you have to do is construct a frequency array given the knowledge of the total span of your input sample and the physical units involved. Say your input data has N elements that spans a total time T. The power or variance spectrum will have N/2 + 1 elements with frequencies you can define by:

IDL> freq = FINDGEN(N/2)/T

And the units of freq are cycles per T. If the units of T are seconds, then the units of freq are Hz.