Computational Physics- Assignment 4

Matheus A. S. Pessôa
ID:260889844

November 10, 2019

Generalities about SNR and Linear algebra relations

When dealing with the LIGO data, we wish to create a matched filter algorithm in order to identify gravitational waves. As we have seen before, a key operation to be applied towards experimental data is to make sure we do not obtain spectral leakage problems – this is why we have applied a Blackman Harris window to the data in the first place.

Now, when building the model, we can think in terms of the most general relations possible. First, if our model is assumed good, we can deal it as a matrix operation in a first approximation, to which when we compare the model we're trying to implement to the actual data, we're making a χ^2 computation. In linear algebra language, that means we're minimizing the following relation: $\chi^2 = (d - A(m))^T N^{-1}(d - A(m))$. In our specific case, the best-fit general model is the one that minimizes this quantity. In order to calculate that, we can simply take the first derivative of this function and make it equal to zero. An important note is that in this case we consider our model is not a non-linear model, which means we can consider $A(m) = Am$, a linear model. We get then that this "best model" should respect the following condition:

$$m = (A^T N^{-1} d)(A^T N^{-1} A)^{-1} \tag{1}$$

. For the purposes of this specific problem, m is the *bestsignal* corresponding to the matched filter algorithm we're looking for, N^{-1} corresponds to our noise model and A the template we get from LIGO. What we're trying to do is simply take the values of A and seeking for a possible best-matched signal that will correspond to the signal of a gravitational wave. This means we're looking for the gravitational wave among a huge amount of noise, but if we ever find a best-fit for it in the signal (data d), that will be a measurable quantity and represent an actual feature.

We then follow to the following consideration: we want to get a noise model for the parameters we see. In order to make the whole linear algebra case more useful, we can consider rewriting,

$$N = N^{-1/2} N^{1/2} \tag{2}$$

and rewrite our equation as

$$m = (A^T N^{-1/2} N^{-1/2} d)(A^T N^{-1/2} N^{-1/2} A)\Big)^{-1} \tag{3}$$

$$m = \Big((N^{-1/2} A)^T (N^{-1/2} d)\Big)\Big((N^{-1/2} A)^T (N^{-1/2} A)\Big)^{-1} \tag{4}$$

Knowing that the transpose of the noise matrix is simply the noise again, $N = N^T$, we can proceed to the following step: let us take the expectation value of the product mm^T.

What we're doing here is calculating the covariance matrix of the model, given that it has a defined dimension and a diagonal with elements $x_i x_j$. It can be defined generally as,

$$cov(x_i, x_j) = \Big\langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \Big\rangle. \tag{5}$$

Opening the terms, we get,

$$cov(x_i, x_j) = \Big\langle (x_i - \langle x_i \rangle)(x_i - \langle x_i \rangle)^T \Big\rangle. \tag{6}$$

Making $\mu = \langle x \rangle$,

$$cov(x_i, x_j) = \Big\langle xx^T \Big\rangle - \mu\mu^T, \tag{7}$$

and $\mu\mu^T = 0$ for uncorrelated errors. So, going back to our model, the expected value can be rewritten as,

$$\langle mm^T \rangle = \Big\langle (A^T N^{-1} A)^{-1} (A^T N^{-1} d)(d^T N^{-1} A)(A^T N^{-1} A)^{-1} \Big\rangle \tag{8}$$

Opening the notation, we have,

$$\Big\langle (A^T N^{-1})^{-1} A^T N^{-1} dd^T N^{-1} N^{-1} A (A^T N^{-1} A)^{-1} \Big\rangle. \tag{9}$$

Therefore,

$$\langle mm^T \rangle = \Big\langle (A^T N^{-1} A)^{-1} (A^T N^{-1} A)(A^T N^{-1} A)^{-1} \Big\rangle. \tag{10}$$

Which can then be reduced to simply,

$$\langle mm^T \rangle = \Big\langle (A^T N^{-1} A)^{-1} \Big\rangle. \tag{11}$$

Now, from our linear algebra intuition and from the initial construction in the similarities from this to χ^2, the standard deviation (error) in the model is generally given by the diagonal of the covariance matrix.

$$\langle mm^T \rangle = \Big\langle (A^T N^{-1} A)^{-1} \Big\rangle \rightarrow \langle mm^T \rangle = \Big\langle (A^T N^{-1/2})(A N^{-1/2})^{-1} \Big\rangle. \tag{12}$$

The diagonal of the covariance matrix corresponds to the errors. We can make then,

$$\sigma_m^2 = \langle mm^T \rangle \rightarrow \boxed{\sigma_m = \Big((N^{-1/2} A)^T (N^{-1/2} A) \Big)^{1/2}} \tag{13}$$

The signal-to-noise ratio (SNR) can be considered as an approximate quality of the signal when compared to the total error we have in the measurements. Now, it is time to make the complete analogy to our LIGO scenario. Here, our best-model m takes into account the parameter

A, corresponds to the template generated by the collaboration that we're trying to match with experimental data d filled with noise N. The SNR is equivalently given as,

$$SNR = \frac{m}{\sigma_m}. \tag{14}$$

In terms of the quantities we have defined above, we get,

$$SNR = \frac{\Big((N^{-1/2}A)^T(N^{-1/2}d)\Big)\Big((N^{-1/2}A)^T(N^{-1/2}A)\Big)^{-1}}{\Big((N^{-1/2}A)^T(N^{-1/2}A)\Big)^{1/2}} \tag{15}$$

Translated to non-linear-algebra quantities, we then get,

$$\boxed{SNR = \frac{d}{\sqrt{N}}}, \tag{16}$$

in time domain. Now, another piece of information will tell us how we can define these more accurately. In theory, this would work fine, however we have trouble when identifying specific peaks in time domain, since the experimental data is filled with noise. One way to obtain a good result then for our model is going to Fourier space, where we do not have correlated errors. The matched-filter method, thus, consists in convolving the template we have, A, with the data signal d. In Fourier space, convolution is translated as a simple multiplication. Even if our function has a quite complicated correlation structure, as long as that structure is time invariant, the individual Fourier modes that make up a realization of that function are independent. So, it becomes much easier to work with stationary functions by taking their Fourier transforms.

Now let us go back to the usual form for a second,

$$m = \frac{A^T N^{-1} d}{A^T N^{-1} A}. \tag{17}$$

If we define $b \equiv N^{-1}d$, and let m to be equal to the top part of this relation; We have something like,

$$m = \sum A(t)b(t) \rightarrow m = \sum A(t-\tau)b(t), \tag{18}$$

where we want to treat the quantities in function of time, and $\ddot{-}\tau$ represents the action of sliding across time domain (convolving) in frequency domain) when we take the Fourier transform.

$$\mathscr{F}(m) = \mathscr{F}(A(t-\tau))\mathscr{F}(b(t)) \tag{19}$$

Notice that this is not a real convolution.The way we write convolution is usually,$\sum A(\tau - t)b(t)$. So how can we write the convolution considering that these coefficients are inversed? We visit the definition of the Fourier transform,

$$\mathscr{F}(f) = \sum f(x)e^{\frac{-2\pi ikx}{N}} \tag{20}$$

We can make the inversion $x' = -x$, and take the Fourier transform of this function now called f'

$$\mathscr{F}(f') = \sum f(x)e^{\frac{2\pi ikx}{N}} = \mathscr{F}(f^*), \tag{21}$$

where we see that the convolution for this case is given in fact by the Fourier transform of complex conjugate of the function f instead of just the function itself. So this tells us we need to take the complex conjugate of the template if we are willing to convolve it with the data.

The matched filter method can then be described mathematically as,

$$MF = \mathscr{F}\Big((N^{-1/2}A)^*(N^{-1/2}d))\Big), \tag{22}$$

where $\mathscr{F}$ is the operator that acts upon both terms taking the Fourier Transform. So in Fourier ($\mathscr{F}$) space, we're using our template A to fit the data d, both being influenced by the way we treat the noise. In this case, if they match, we expect to see a spike in the frequency domain. Thinking about the real-case scenario, we are applying same operations to smooth the noise (that will be mentioned afterwards), and we are left with only template A and data d to be actually considered, since we are applying the same operations for N that multiply these two quantities. By "eliminating" the noise importance for both template and data, we can consider: if there is a real gravitational wave event corresponding to the prepared template, when we convolve it with the real data, we will see a spike where they match. To see the behavior in time domain, it is enough to take the inverse Fourier transform.