

Problem Set 6 for Fourier transforms/matched filters. Due Friday November 11 at 11:59 PM

1) Write a function that will shift an array by an arbitrary amount using a convolution (yes, I know there are easier ways to do this). The function should take 2 arguments - an array, and an amount by which to shift the array. Plot a gaussian that started in the centre of the array shifted by half the array length.

2) a) The correlation function $f \star g$ is $\int f(x)g(x+y)dx$. Through a similar proof, one can show $f \star g = ift(dft(f) * conj(dft(g)))$. Write a routine to take the correlation function of two arrays. Plot the correlation function of a Gaussian with itself.

b) using these results, write a routine to take the correlation function of a Gaussian (shifted by an arbitrary amount) with itself. How does the correlation function depend on the shift? Does this surprise you?

3) The circulant (wrap-around) nature of the dft can sometimes be problematic. Write a routine to take an FFT-based convolution of two arrays without any danger of wrapping around. You may wish to add zeros to the end of the input arrays.

4) DFTs work very nicely out of the box when there are an integer number of periods of a wave in the region analyzed. Sadly, when we are dealing with real data, we usually are forced to analyze a finite chunk of data, and there will in general be no particular relation between the frequencies in the data and the interval we're analyzing. We'll look at the effects of this a bit now.

a) Show that:

$$\sum_{x=0}^{N-1} \exp(-2\pi i k x / N) = \frac{1 - \exp(-2\pi i k)}{1 - \exp(-2\pi i k / N)}$$

It may help to recognize that the sum can be re-written $\sum \alpha^x$ where $\alpha = \exp(-2\pi i k / N)$ so we can treat it as the sum of a geometric series.

b) Show that this approaches N as k approaches zero, and is zero for any integer k that is not a multiple of N .

c) We can use this to analytically write down the DFT of a non-integer sine wave. Pick a non-integer value of k and plot your analytic estimate of the DFT. Show that the FFT agrees (to within machine precision) with your analytic estimate. Normally, we think of the Fourier transform of a pure

sine wave to be a delta function. Are we close to that? This phenomenon is usually known as *spectral leakage*.

d) A common tool to get around this is the use of *window functions*. The leakage essentially comes from the fact that we have a sharp jump at the edge of the interval. If we multiply our input data by a function that goes to zero at the edges, this cancels out the jump, and so prevents the leakage from the jumps at the edges. Of course, since we have multiplied by the window in real space, we have convolved by it in Fourier space. One simple window we could use is $0.5 - 0.5 \cos(2\pi x/N)$ (there are many, many choices). Show that when we multiply by this window, the spectral leakage for a non-integer period sine wave drops dramatically.

e) Show that the Fourier transform of the window is $[N/2 \ N/4 \ 0 \ \dots \ 0 \ N/4]$ (either numerically or analytically). Use this to show that you can get the windowed Fourier transform by appropriate combinations of each point in the unwindowed Fourier transform and its immediate neighbors (you may need to be careful with signs here, since if you work through the math, some of the transforms need to be inverse FFTs). The choice of suitable windows is as much art as science (and depends on the details of what you're most concerned about), but I hope this gives at flavor of what's going on and will be useful for matched filter results.

5) Matched Filter of LIGO data

We are going to find gravitational waves! Key will be getting LIGO data from github:

https://github.com/losc-tutorial/LOSC_Event_tutorial

While they include code to do much of this, please don't use it (although you may look at it for inspiration) and instead write your own. You can look at/use `simple_read_ligo.py` that I have posted for concise code to read the hdf5 files. Feel free to have your code loop over the events and print the answer to each part for that event. In order to make our life easy, in case we have to re-run your code (which we should not have to do), please also have a variable at the top of your code that sets the directory where you have unzipped the data. LIGO has two detectors (in Livingston, Louisiana, and Hanford, Washington) and GW events need to be seen by both detectors to be considered real. Note that my `read_template` function returns the templates for both the plus and cross polarizations, but for simplicity you can just pick one of them to work with.

a) Come up with a noise model for the Livingston and Hanford detectors separately. Describe in comments how you go about doing this. Please

mention something about how you smooth the power spectrum and how you deal with lines (if at all). Please also explain how you window the data (you may want to use a window that has an extended flat period near the center to avoid tapering the data/template where the signal is not small).

b) Use that noise model to search the four sets of events using a matched filter. The mapping between data and templates can be found in the file `BBH_events_v3.json`, included in the zipfile.

c) Estimate a noise for each event, and from the output of the matched filter, give a signal-to-noise ratio for each event, both from the individual detectors, and from the combined Livingston + Hanford events.

d) Compare the signal-to-noise you get from the scatter in the matched filter to the analytic signal-to-noise you expect from your noise model. How close are they? If they disagree, can you explain why?

e) From the template and noise model, find the frequency from each event where half the weight comes from above that frequency and half below.

f) How well can you localize the time of arrival (the horizontal shift of your matched filter). The positions of gravitational wave events are inferred by comparing their arrival times at different detectors. What is the typical positional uncertainty you might expect given that the detectors are a few thousand km apart?