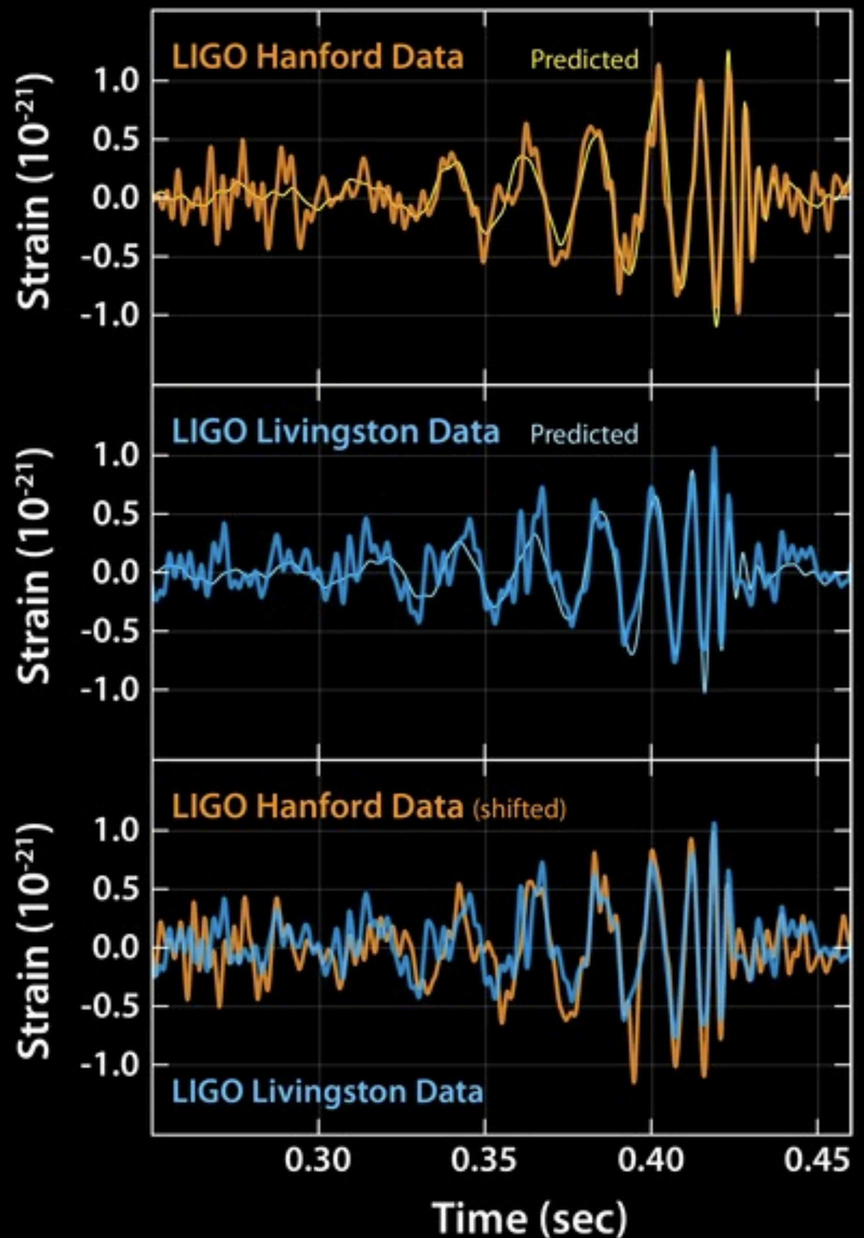
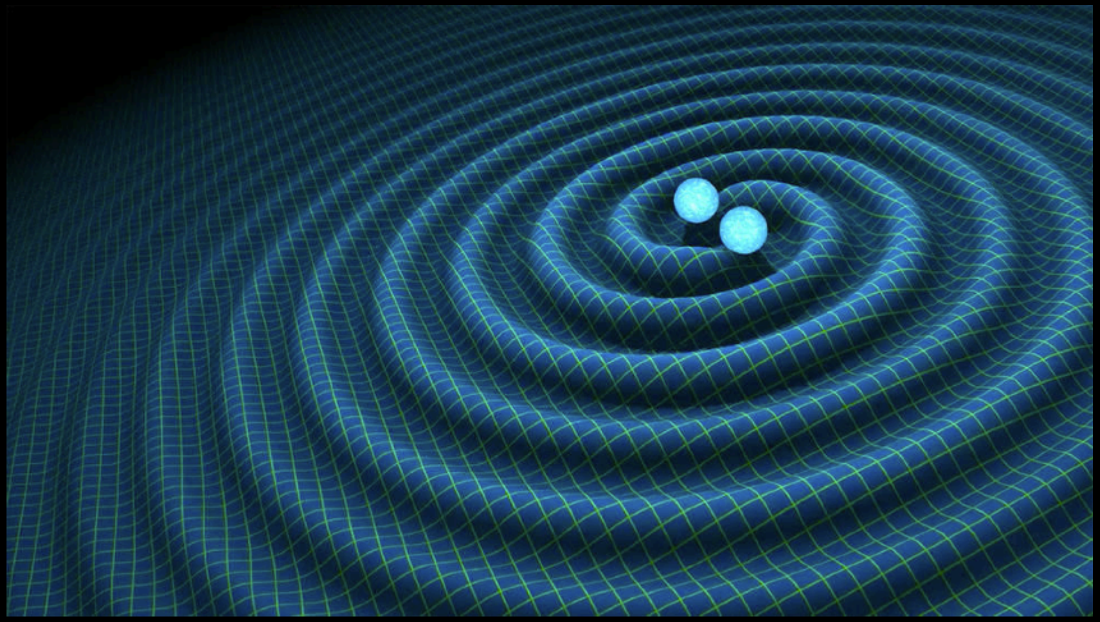
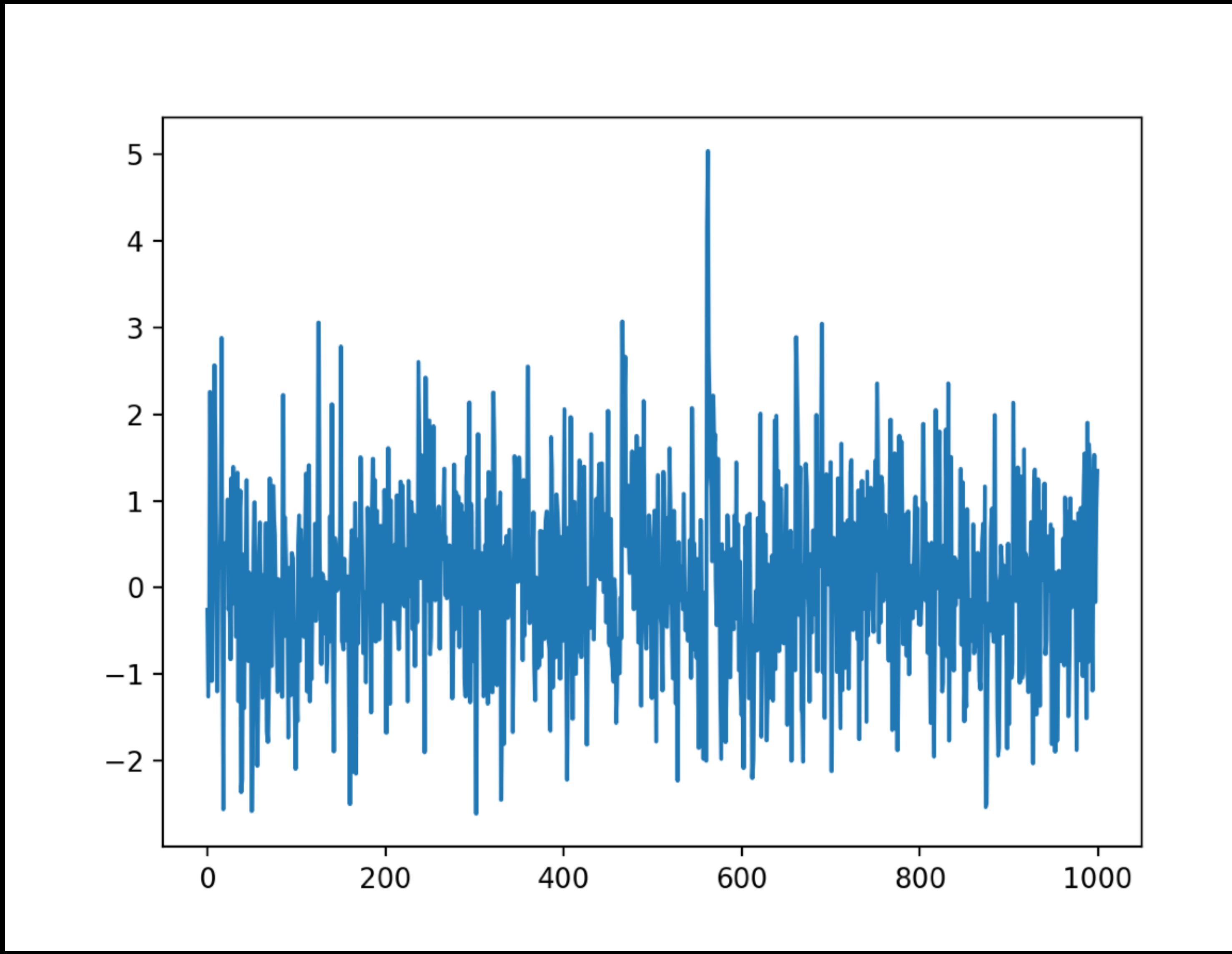
Matched Filters/Ligo





Searching for Signals

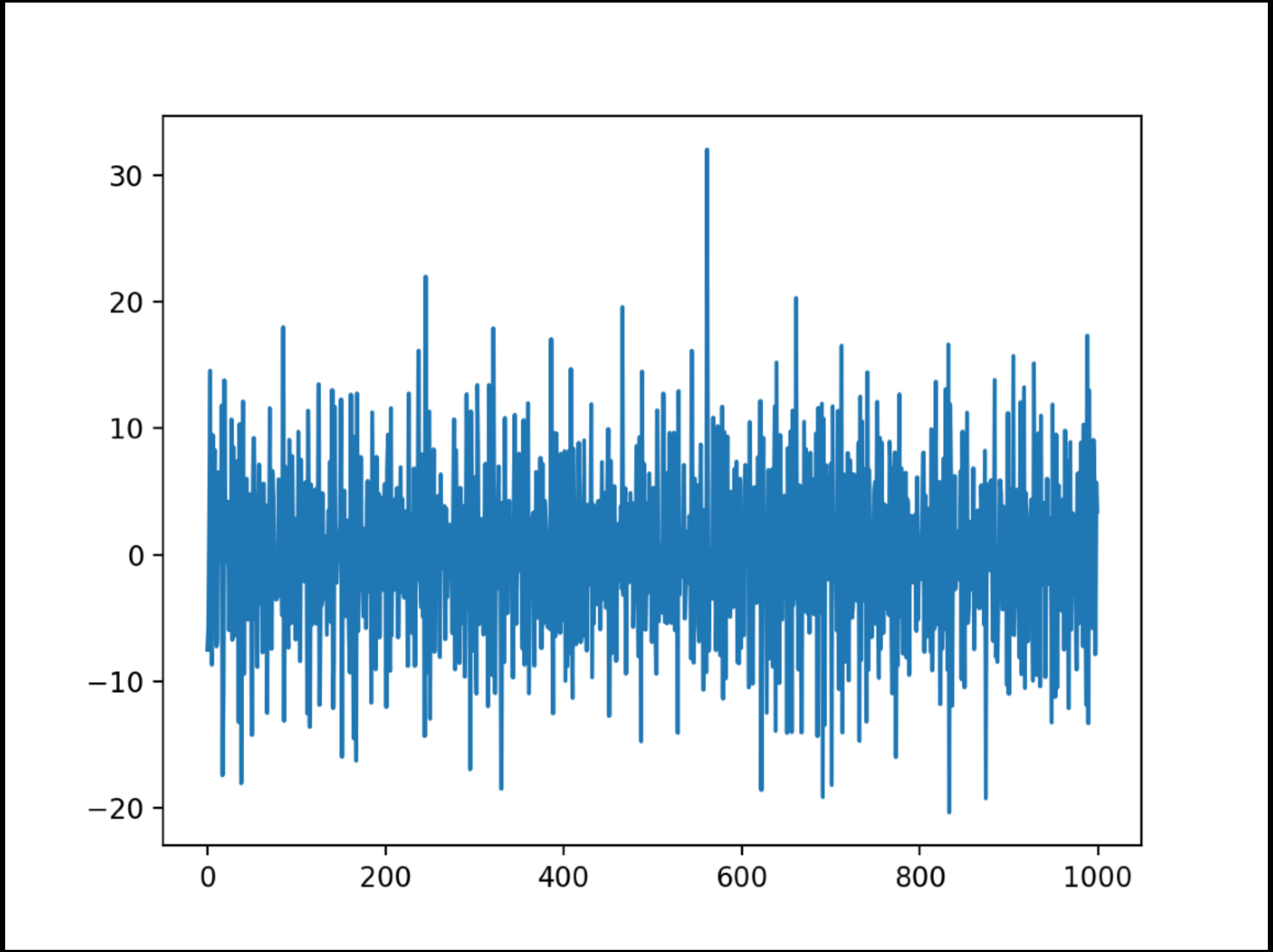
- Simulated data - particle detector.
- Signal decays exponentially when hit by particle.
- Noise is white.
- Where/how energetic were the particles that hit?



Simulated data of exponential-decay detector hit by particles.

Deconvolution

- Simplest attempt is to deconvolve observed data using exponential response.
- How might this work or not work? Why?



Deconvolved version of previous

Matched Filter

- We want to search for a signal in data. We don't know where it will be. How do we find it?

- Best fit amplitude for 1-D template A is $A^TN^{-1}d/A^TN^{-1}A$

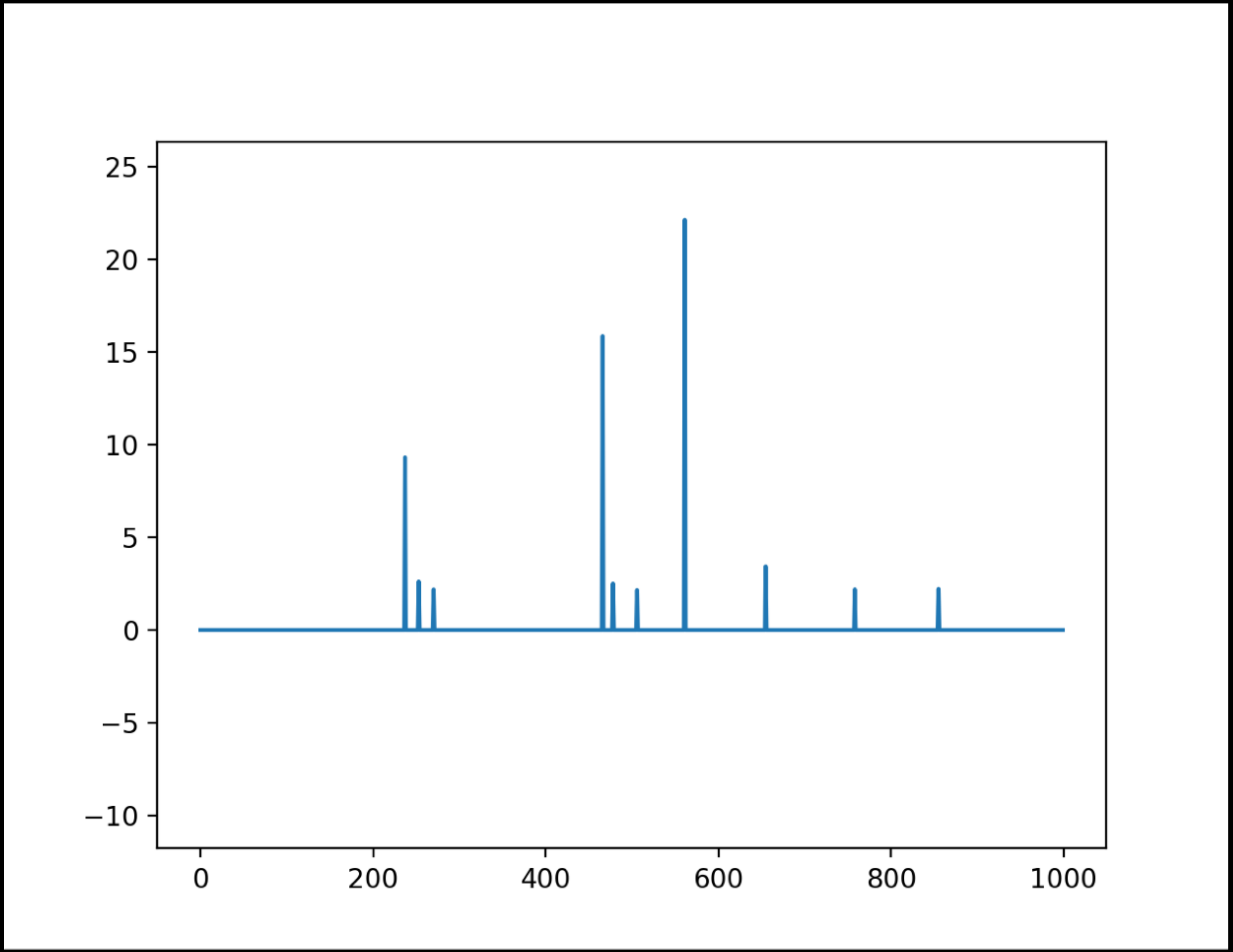
- We can search many possible locations of template with matched filter, replacing top by correlation of A with $N^{-1}d$ (or $N^{-1}A$ with d) if noise is stationary

- Alternatively, could take correlation of $N^{-1/2}A$ with $N^{-1/2}d$. What would the noise in $N^{-1/2}d$ look like?

  NB - relevant operation is cross-correlation, not convolution

MF Output

- Output of MF, vs. true input signal.
- In this case, SNR of MF is twice that of deconvolution.
- MF *multiplies* by noise, deconv. *divides*. MF stays stable in presence of zeros in noise.



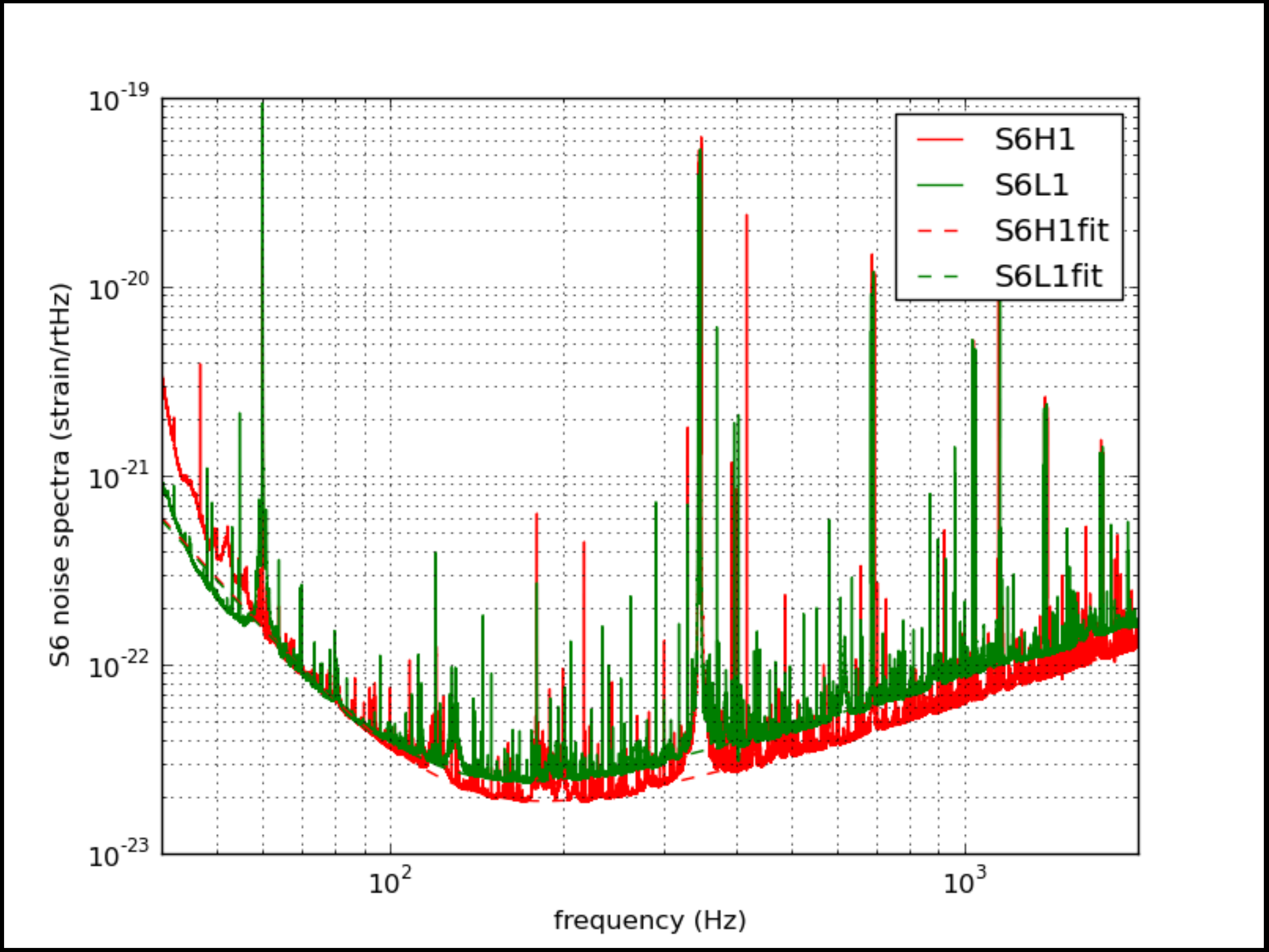
LIGO Data

- https://www.gw-openscience.org/tutorials/
(NB - I think they switched to github: https://github.com/losc-tutorial/LOSC_Event_tutorial)

- Download: “file with data” will get you everything

- simple_read_ligo.py will read for you (once you have h5py installed and working)

First, what should we see for noise?



Power Spectrum Description

- Modes are uncorrelated in Fourier space
- SNR^2/mode is set by (template FT)^2/noise PS
- Noise PS is just FT of correlation function

Fourier Interpretation

- Noise model has same total variance independent of correlation length.
- Looking at FT, long length packs noise power into many long wavelengths. Template has more power on high-frequency scales (good SNR)
- Short length spreads out power over many many modes, dropping average noise power. Template well above noise on large scales (good SNR).
- Intermediate packs all its noise into same scales as template. Never have good SNR.

When your noise looks like your signal,
you're going to have a bad day…

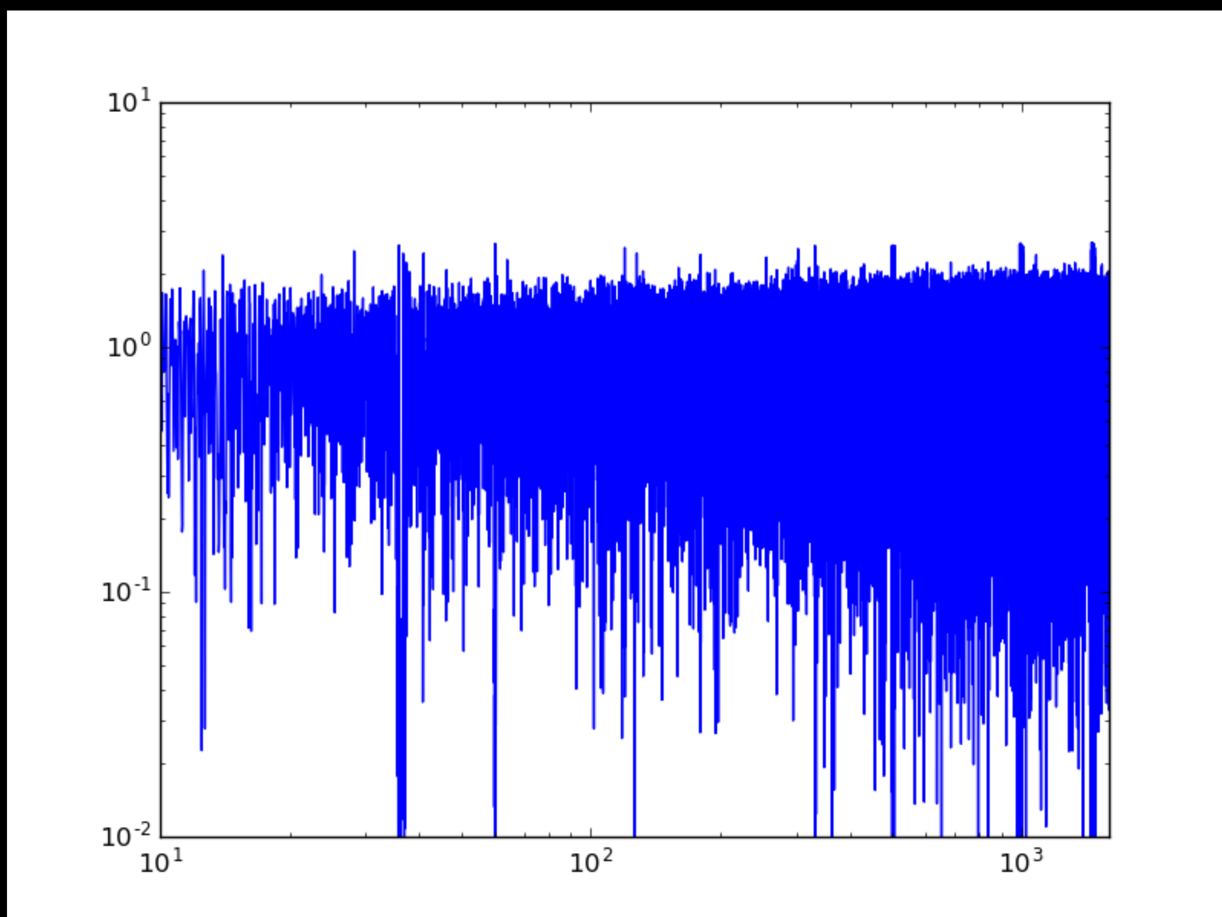
How Should We Estimate Noise?

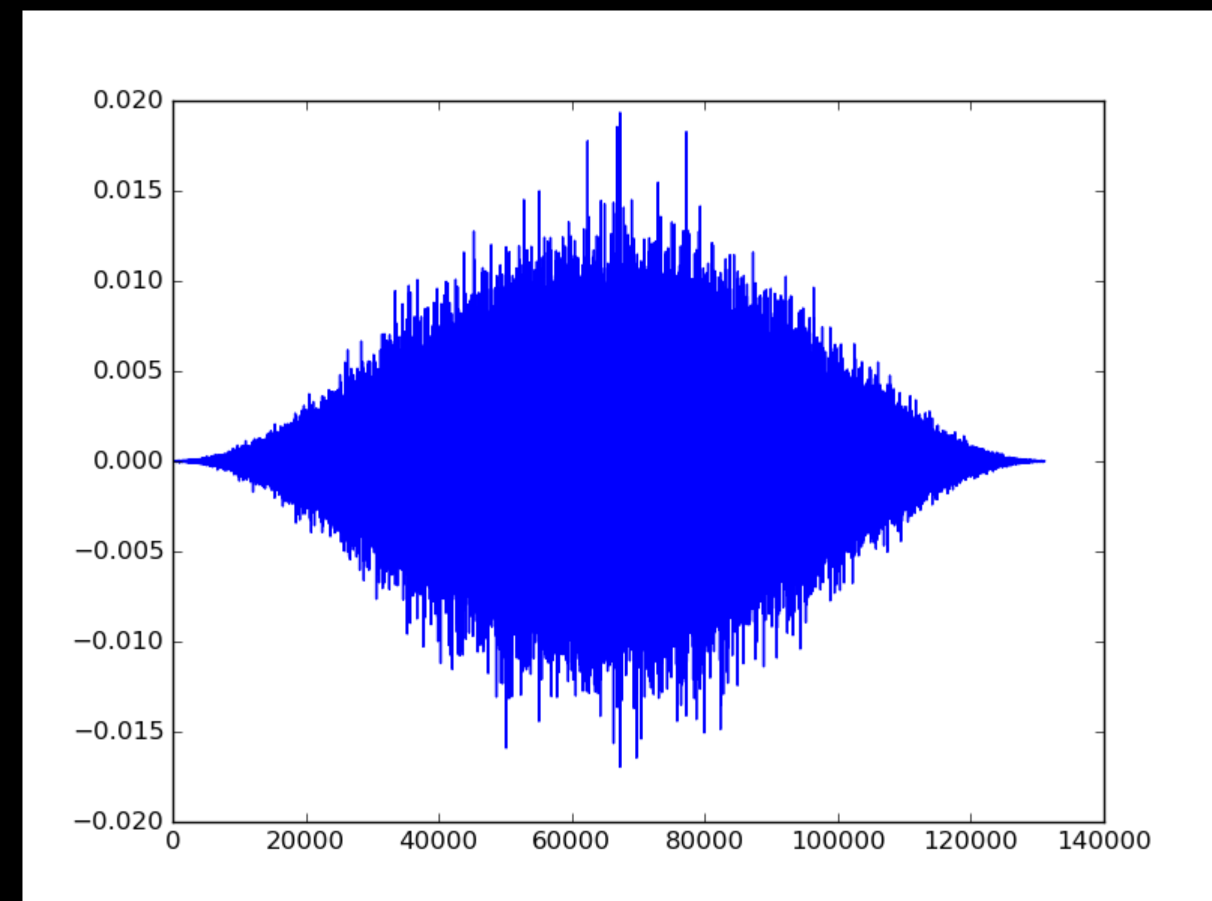
- Windowing key to avoiding FFT ringing
- smooths out spectral features
- Noise large per mode in FT, so we have to average
- What are your thoughts on averaging?

Smoothing PS

- Take $|FT|^2$, which is an estimate
- Smooth by convolving with an extended function.
- Thoughts on the function?

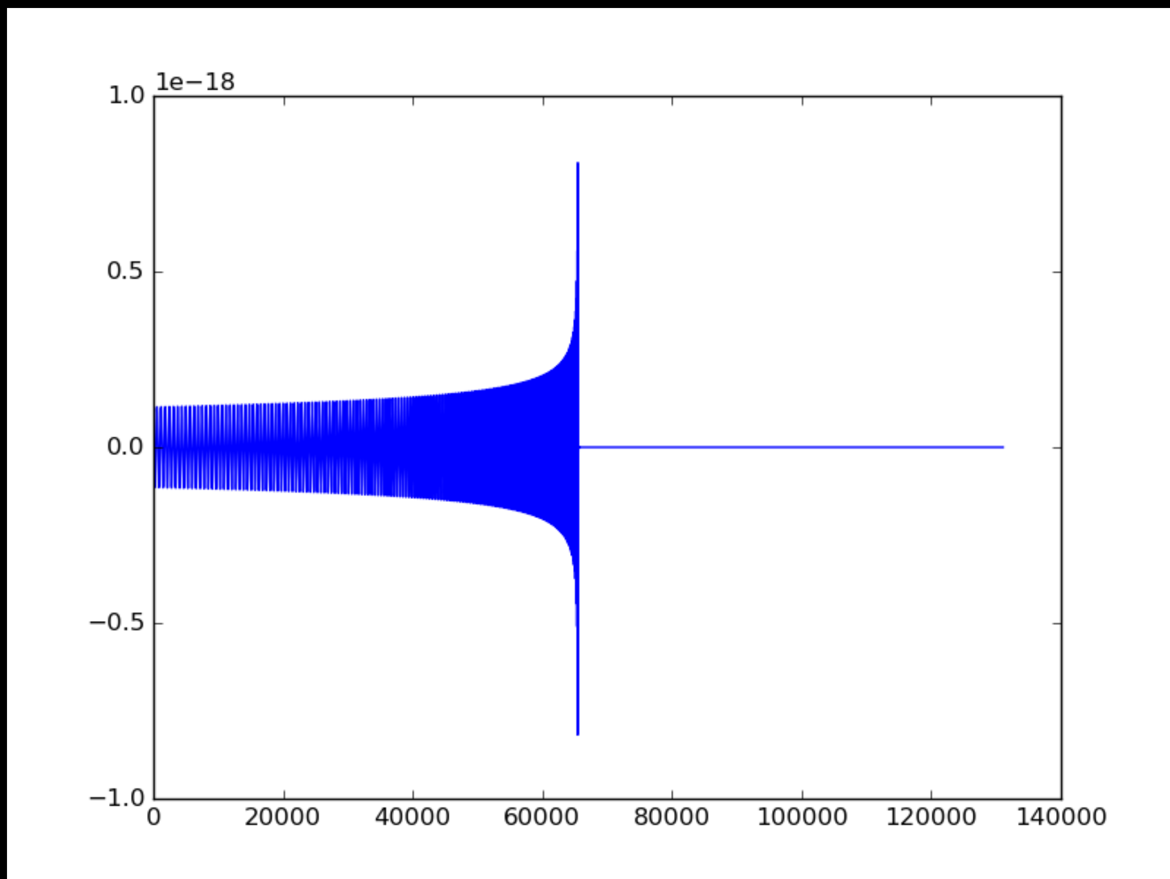
Pre-Whitened Data from Smoothed PS

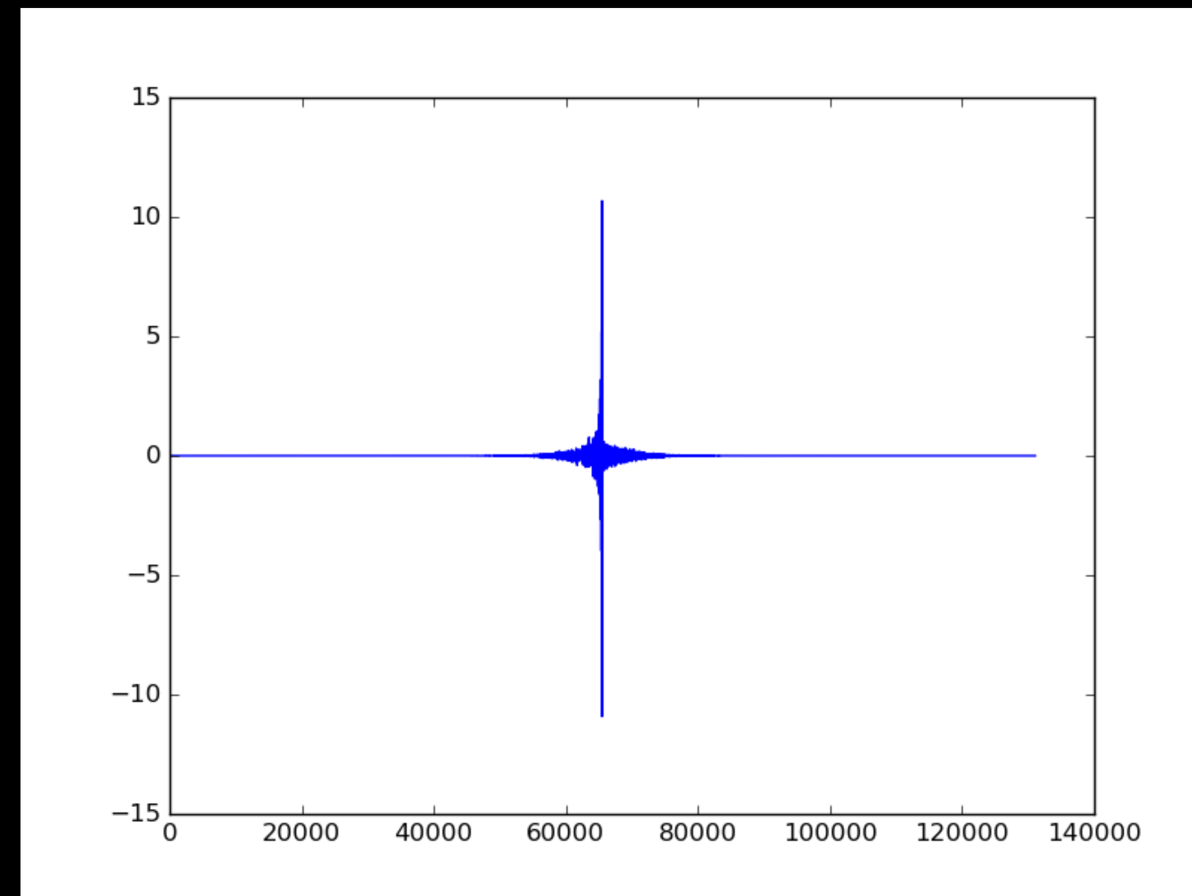




Left: Whitened FT of data. Looks not crazy. What are little nubbins sticking up?
Right: whitened data. Window shape is pretty obvious.

Pre-whitened template

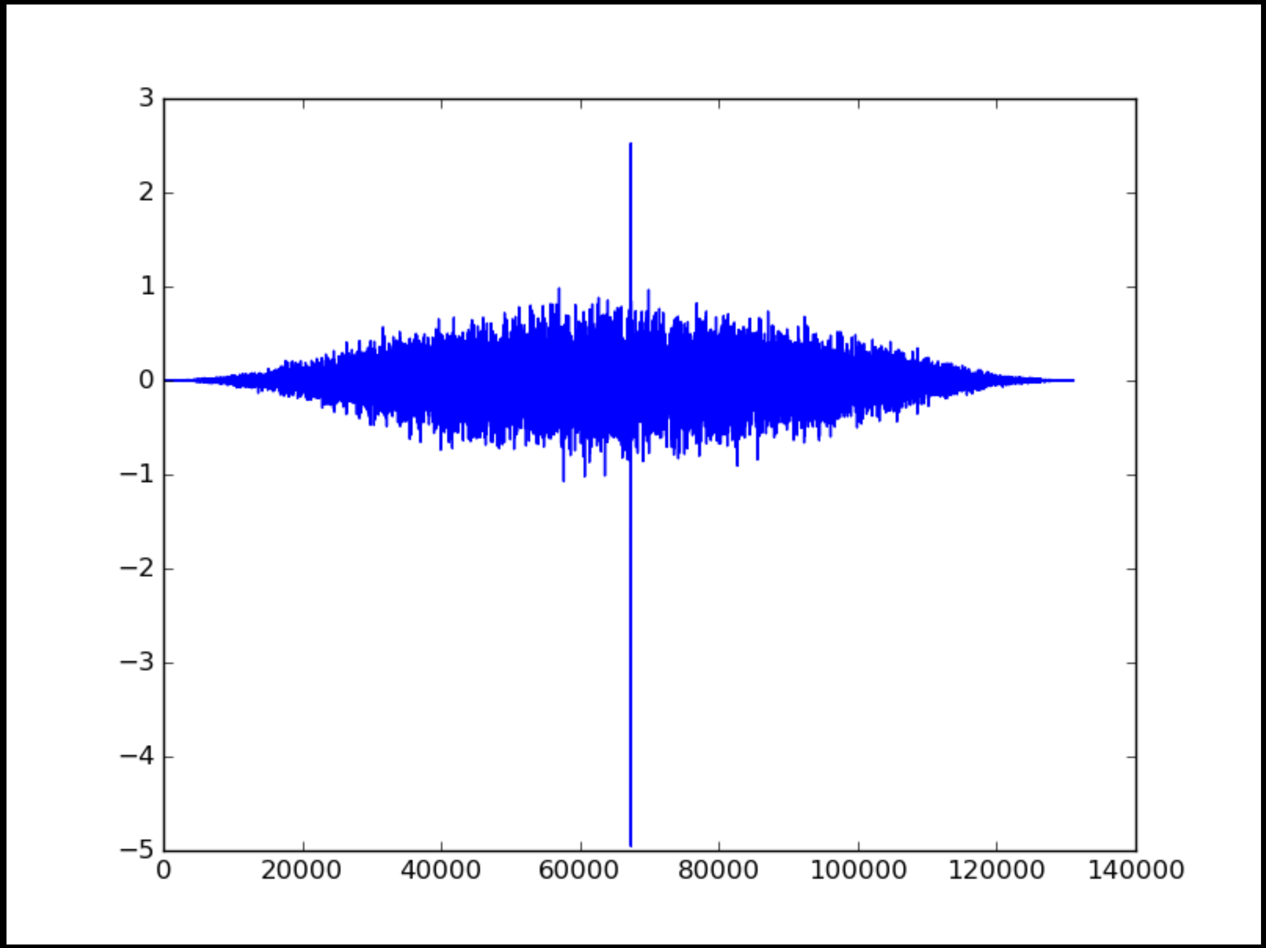




- For this event, template is not small at start of data. Will this be a problem?

- Can look at pre-whitened version of template to get an idea.

Can Use for MF now



FFT Shift of matched filter output. We found a GW!

Averaging PS

- Break PS up into small chunks so we have many
- Take the FT of each chunk
- Add the FT^{2}s together.
- How do we apply this (short) PS to original data?
- Qualitatively, how do we relate this PS estimate to smoothed one?

More Windowing

- Usual windows taper every sample.
- Reduces power in a way we probably aren’t happy with
- How could we modify window to make this less of an issue?
- Let’s try this on data…

Normalizations

- Properly normalizing noise can require care. I usually check with white noise.
- N^{-1} for white noise with $\sigma=1$ should be identity.
- Variance of FT is sum over data = $\sigma^2 N_{data}$. In Fourier space, we want $N(k)=Var(F(k))/N_{data}$. Not just $Var(F(k))$
- Window function: $Var(F)$ for white = $\sum(\sigma^2 W^2)$.
- If you want the real-space variance to be correct where $W\sim1$, you'll need to use $\sum(W^2)$ as your normalization.