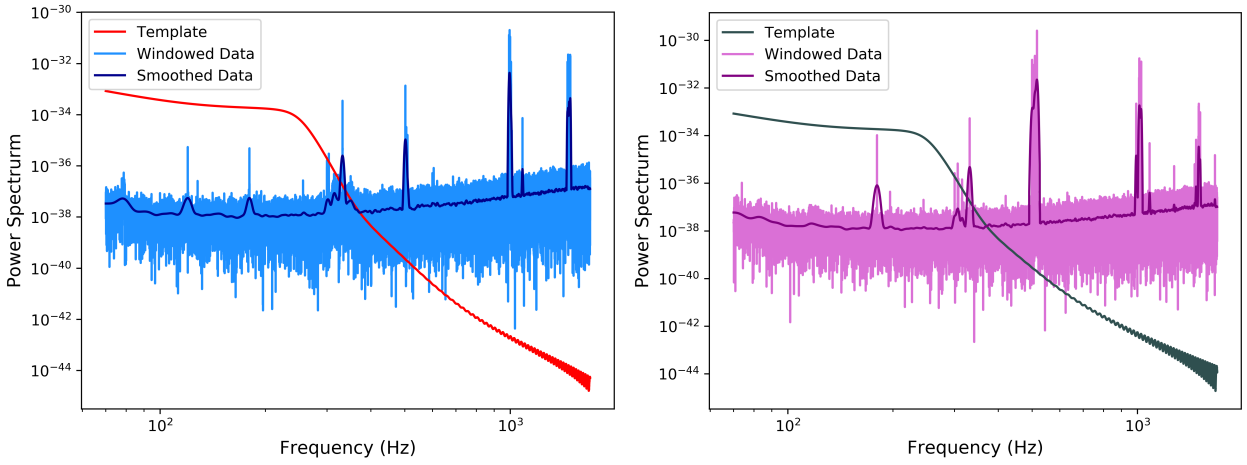
All the analysis for this question is carried out in "problem_1.py".

1. a) Our first aim is to come up with a noise model for the Livingston and Hanford detectors separately. We first calculate the power spectrum of the strain data and take this as an estimate of our noise matrix, N. We can do this assuming the data is uncorrelated, and hence N just becomes a 1D array. The power spectrum is calculated by taking the magnitude squared of the Fourier transform.

However, due to the fact that our data is collected over a discrete period of time, the edges of our data set can be discontinuous which will result in our DFT being slightly inaccurate. We can fix this by multiplying our data by a window function that goes to 0 at the edges. As suggested, a window function with an extended flat period around the center that tapers off to 0 is chosen. For this we use the Tukey window with $\frac{1}{2}$ of the window inside the tapered region.

We can then smooth the windowed power spectrum, by first qualitatively choosing a range of frequencies over which the spectrum most resembles white noise (i.e. fluctuating randomly about a constant amplitude). For simplicity we assume that the range of frequencies are the same for the Livingston and Hanford detectors. We then remove spikes in our data by applying a Gaussian filter, with a standard deviation of 75, which is chosen because it provided the best results compared to the Savitzky-Golay filter, the median filter and so on. Our noise matrix N, is the 1D array that is the power spectrum that results from these manipulations. To make sure that the array has the same length as data/template arrays, we set the values outside the range of frequencies to ∞ so that these regions do not contribute anymore. Calculating this separately for the Hanford and Livingston detectors and taking event "GW150914" as an example (for the entire problem set), the result of our whitening is shown in the plot below. The ideal range of frequencies was found to be [70Hz, 1690Hz].



(a) Hanford Detector.

(b) Livingston Detector.

Figure 1: Plot of the smoothed data, windowed data and the template used to determine N for the data and template over the range [70Hz, 1690Hz].

We can see how the smoothed data has smaller fluctuations and the spikes have been reduced. Furthermore, we see that the Tukey window has transformed this region into an approximately flat region that looks like white noise. The smoothed data line for each detector then becomes our noise matrix with infinities added before and after corresponding to outside the frequency region.

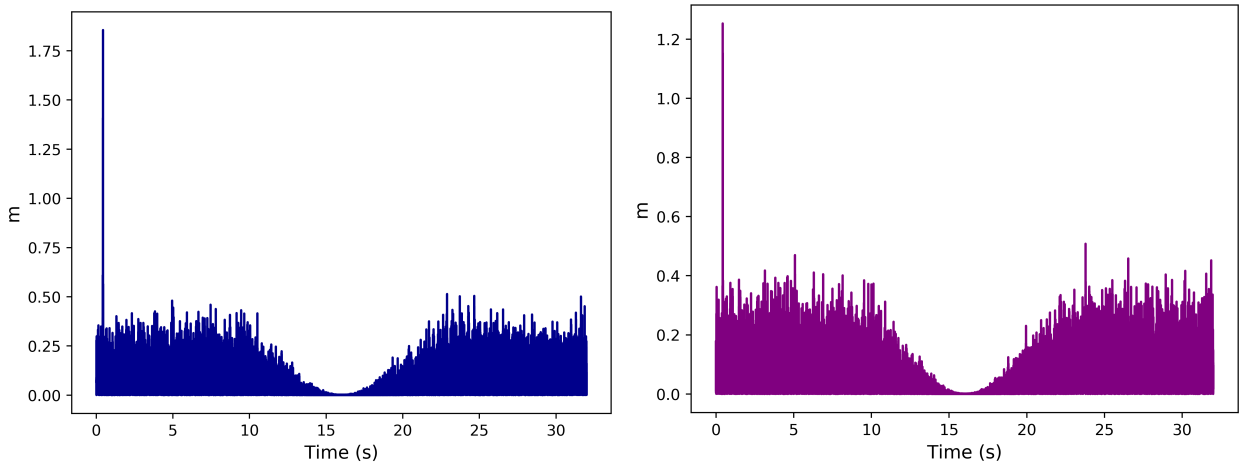
b) Now that we have an estimate for N, we can proceed with the method of matched filters. We know that,

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

where m is our set of parameters, A is our template data and d is our strain data. However, this is not that accurate as our noise matrix may not behave as much like white noise. We can fix this by splitting our noise matrix into factors of $N^{-\frac{1}{2}}$ such that our new noise matrix looks like the white noise matrix, i.e. the identity matrix.

$$m = (A^T N^{-\frac{1}{2}} N^{-\frac{1}{2}} d)(A^T N^{-\frac{1}{2}} N^{-\frac{1}{2}} A)^{-1}. \tag{2}$$

We see that in order for our noise matrix to look like the identity matrix, $A \to N^{-\frac{1}{2}} A$ and $d \to N^{-\frac{1}{2}} d$. Hence, we proceed with the method of matched filters using our new A and d matrices by calculating the correlation function of both of these two terms. We do so by multiplying the Fourier transform (taken using rfft) of $N^{-\frac{1}{2}} d$ with the conjugate of $N^{-\frac{1}{2}} A$. Taking the real inverse transform of the result (using irfft) gives us our m values. We can now plot this against time using the time spacing between samples given to us.

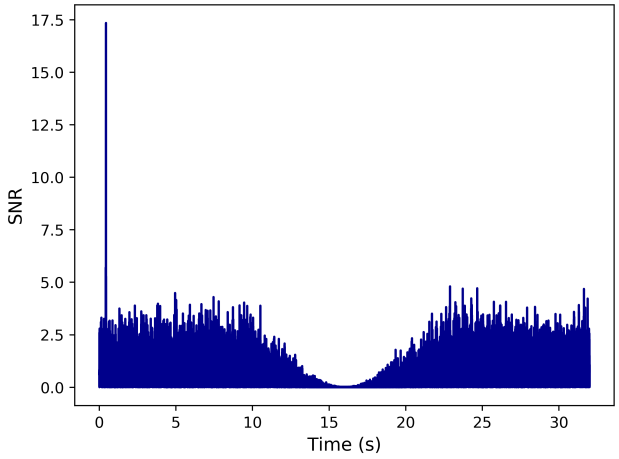


(a) Hanford Detector.

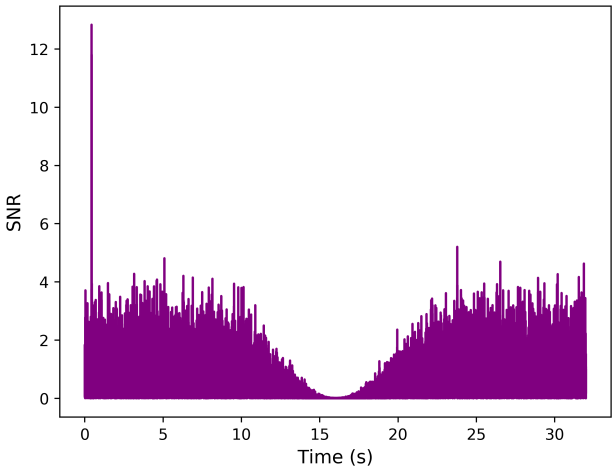
(b) Livingston Detector.

Figure 2: Plot of the matched filter output for both detectors.

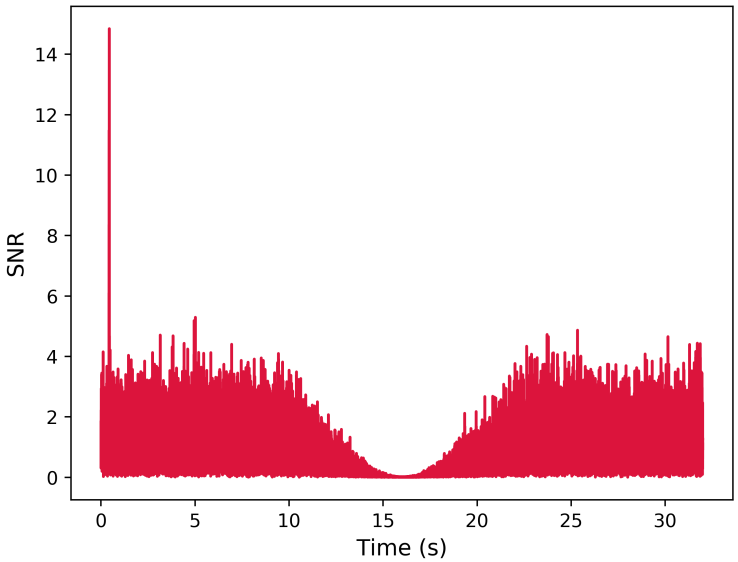
c) We can determine a noise from each event using a similar method as we did for a least squares fit by taking our covariance matrix to be $Cov = A^T N^{-1} N^{-\frac{1}{2}} A$. Since we are working with 1D arrays, we can write this as $Cov = A^2 N^{-1}$ which we can get from what we calculated previously by squaring $N^{-\frac{1}{2}} A$. Finally, we can take the average of this result over our entire template and then square root it to get an estimation for the noise. The signal to noise ratio (SNR) can then be determined by dividing our matched filter output m by this noise σ_m. Furthermore, we can calculate the SNR for the combined events by summing m for both events together and then calculating σ_m and the SNR as before. All the plots can be seen below. σ_m was determined to be 0.107, 0.0976 and 0.145 for the Hanford, Livingston and combined detectors.



(a) Hanford Detector with $\sigma_m \approx 0.107$.



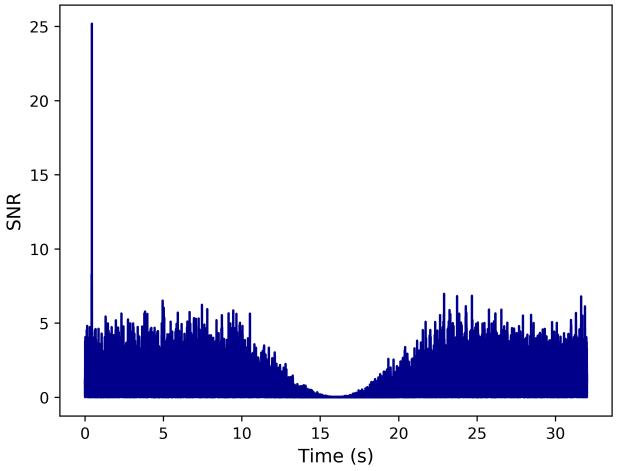
(b) Livingston Detector with $\sigma_m \approx 0.0976$.



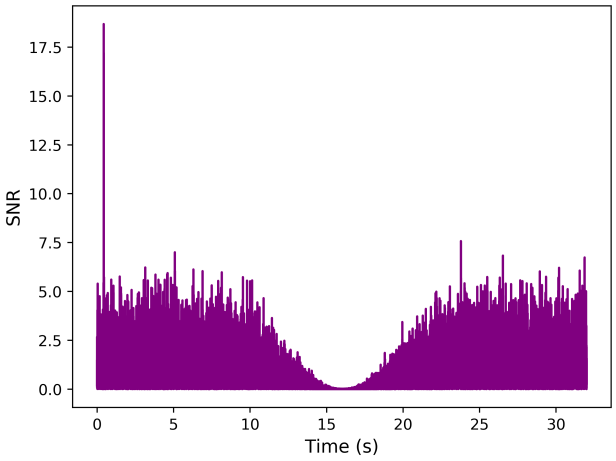
(c) Combined events with $\sigma_m \approx 0.145$.

Figure 3: Plot of the SNR for the individual detectors and the combined events produced from the matched filter output with σ_m calculated from our noise model.

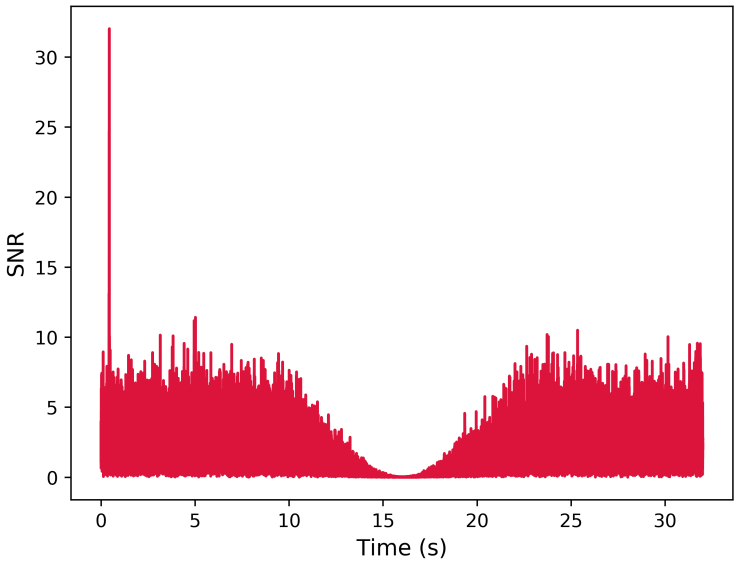
d) We can calculate the scatter in the matched filter output as the sample standard deviation of the respective event. The SNR can then be calculated as before. The plots for this have been shown below:



(a) Hanford Detector with a scatter of 0.0737.



(b) Livingston Detector with a scatter of 0.0671.



(c) Combined events with a scatter of 0.111.

Figure 4: Plot of the SNR for the individual detectors and the combined events produced from the matched filter output with a scatter calculated as the standard deviation of m.

We can now compare the value of the scatter to σ_m for the Hanford, Livingston and combined events. This is similar to comparing the SNR as the maximum value of m for each detector is the same using both methods. There is an absolute discrepancy of 0.0333, 0.0306 and 0.0339 or a percentage discrepancy of 31.1%, 31,3% and 23.4%, respectively, where the SNR calculated using scatter is larger. This large discrepancy is likely caused by issues in our approximations for the noise model. For example, the frequency range is decided somewhat arbitrarily leaving room for improvement and a different window or different standard deviation for our Gaussian filter could result in a lower discrepancy.

e) Using our noise model for the template, we can calculate the frequency at which half the weight comes from above and half below by taking a cumulative sum using np.cumsum. We calculate the cumulative sum of the whitened Fourier transform of our templates and find the closest argument to the midway point. The output of this can be seen in Figure 5. For the Hanford and Livingston detectors, the mid-point frequency was calculated to be approximately 142.72Hz and 143.22Hz, respectively.

f) We can estimate the time of arrival as the peak of the SNR calculated from our noise model (i.e. from Figure 3). For the Hanford and Livingston detectors, the time of arrival was determined to be approximately 0.440s and 0.432s, respectively. Since gravitational waves travel at the speed of light, c, we can calculate the expected time delay between both detectors from the fact that both detectors are 3,002km apart (from the LIGO website). Therefore, the expected time delay is $\frac{3{,}002\times 1000}{c} \approx 0.0100$s, whereas the calculated delay between both detectors is 0.00879s. From this, we see that our experimental delay is within the theoretically expected delay and hence, our results are accurate.

**The outputs for each part can be seen in the combined image below:*

```
[Running] python -u "/Users/jehandastoor/Phys512/assignment5/problem_1.py"
SNR Hanford Sigma_m_H = 0.10693214442528241
SNR Livingston Sigma_m_L = 0.09763920429011694
SNR Combined Sigma_m = 0.14474084119813466
Scatter SNR Hanford std_H = 0.07365186028578238 Discrepancy_H = -0.03328028413950003 % Discrep = -31.122806260332915
Scatter SNR Livingston std_L = 0.0670804949265164 Discrepancy_L = -0.030558709363600542 % Discrep = -31.297581320717192
Scatter SNR Combined std_L = 0.11080258397089532 Discrepancy_L = -0.03393825722723934 % Discrep = -23.447602588395565
Mid frequency Hanford = 142.71875
Mid frequency Livingston = 143.21875
Arrival Time H = 0.4404296875 Arrival Time L = 0.431640625
Theoretical Uncert = 0.010013594137848525 Calculated Uncert = 0.0087890625

[Done] exited with code=0 in 33.718 seconds
```

Figure 5: Python output for "problem_1.py".