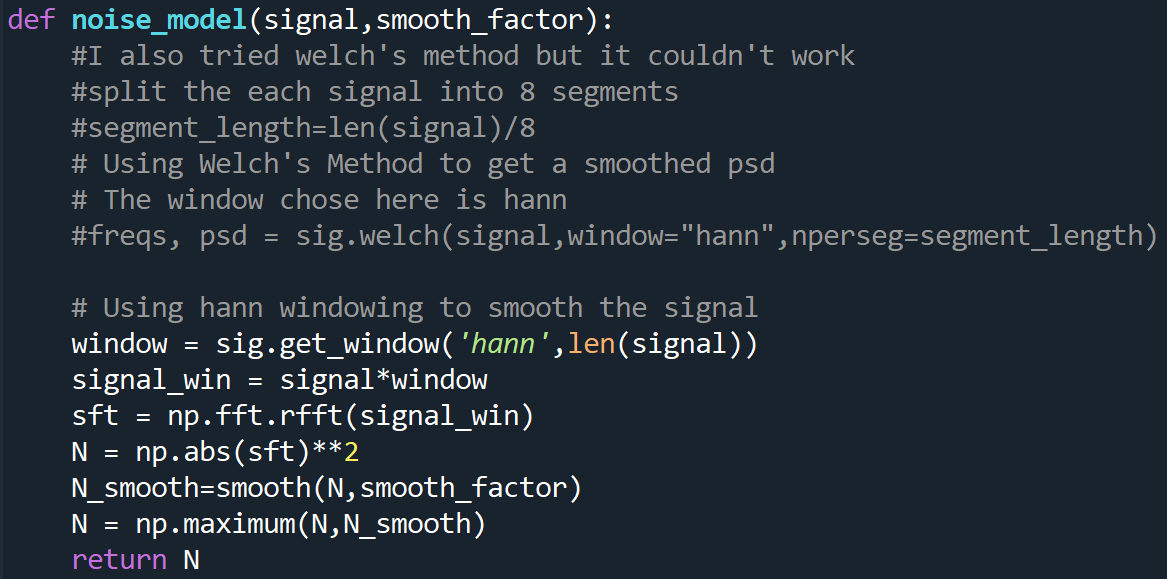
Assignment 5

Yifei Gu

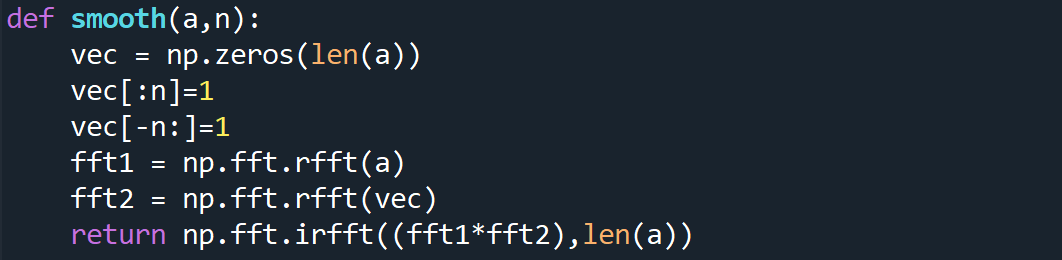
260906596

**Part a)**

To build the noise model, I did windowing the data with **Hann Window** to **smooth the data first**. Because it has a long flat period in the middle. I also tried **Welch’s method** to smooth the data. It **does a better job than Hann Window**, but Welch’s method **reduces the resolution** of the data and I had **difficulties to interpolate** the smoothed data to the equal number of points with the original data. Thus, I chose **Hann Window**.



**The noise** in frequency domain was **taken as the square** of the **strain** in frequency. I also **smoothed the noise model** by convolution with a box car function:



I did a test with box cars in different length. Here is how the noise model with different smooth factors performs:

图表, 直方图

描述已自动生成

Based on this test, **a box car with width (I call it smooth factor in the code) of 5 was chosen** as it **removed sharp peaks but avoid flattening features.** Noise models for Hanford and Livingston are shown below:

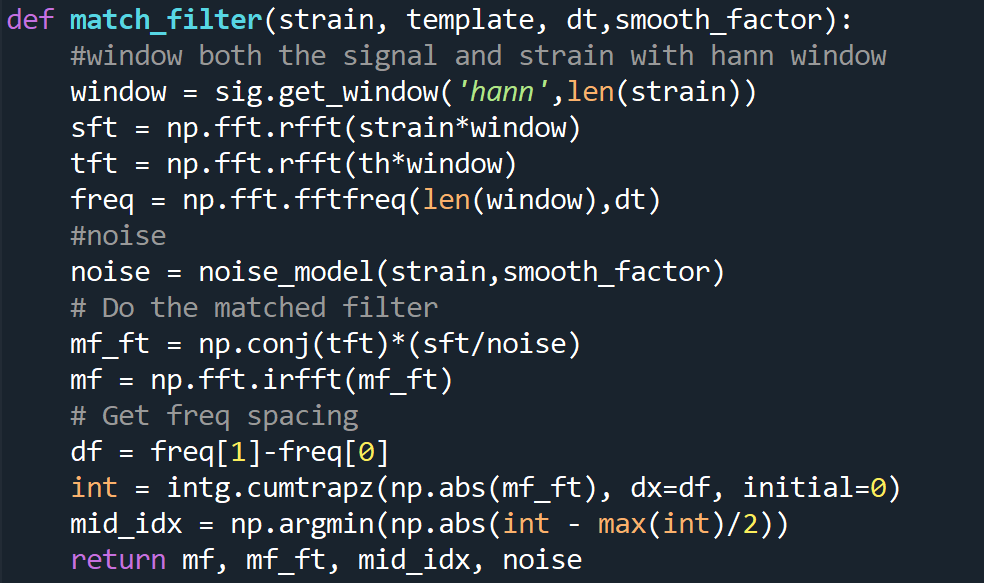
图表

描述已自动生成图表, 直方图

描述已自动生成

**Part b)**

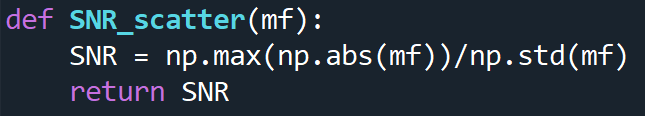
**Since I windowed the data, in the matched filter I windowed the templates as well.**



After loading all the data, I sorted them as which detector each collected from. Searching through all data with corresponding templates, I found a few events. They are attached in the end of the document with all the other questions.

**Part c)**

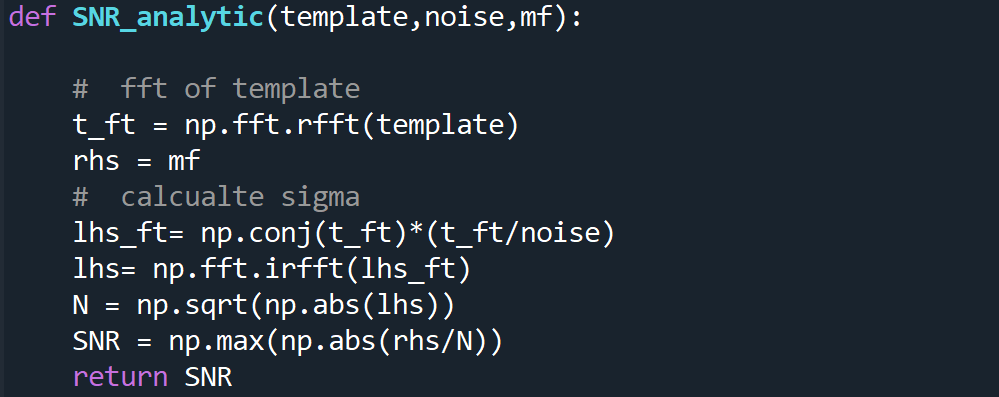
In this question, I defined the noise as the standard deviation of the matched filter. The SNR of scatter was calculated as the ratio between the maximum of the matched filter to the noise.



**Part d)**

The analytic SNR, I calculated the noise in the matched filter first in frequency domain then transform back to time domain and take square root of it:

The SNR was estimated as the maximum of the ratio between matched filter signal and the noise:

Where ‘Matched filter signal’ is the raw output of the matched filter in the time domain. Here is how I built the function for calculating the analytic SNR: 

**Part e)**

Since the output of the match \_lter in the frequency domain represents the weighting of those frequencies, we can determine the frequency above and below which half of the weight lies by taking the cumulative integral of our \_lter, and \_nding the frequency at which the cumulative integral is half the full integral.

To find the frequency from each event where half the weight comes from above that frequency and half below, we need to find the

