AMATH 582: HOMEWORK 1

VINSENSIUS

Applied Mathematics Department, University of Washington, Seattle, WA
vxvinsen@uw.edu

Abstract. A submarine pressure data was given to be cleaned and filtered. The resulting data is the trajectory of the submarine over 24 hours. The data has frequency signature at indices (26,17,55) and (39,49,10). The data is filtered by gaussian filter with $\sigma = 15$.

1. Introduction and Overview

Signal processing is very important in everyday life because not every data that is received in experiment or measurement be perfect. They always come with noise that may come from electrical equipment or noise that comes from outside in the form of background noise. Thus, it is important to clean the data from the noise in order to extract data. One of the way to clean the data is by applying filter and using fourier transform to work in frequency domain so that one can extract important frequency information about the data.

The data that we are given is the submarine pressure signal data. We are tasked to find the trajectory of the submarine over last 24 hour based on the pressure data.

2. Theoretical Background

This section will give an overview of the theoritical bakcground of the subject [1]. The Fourier transform is defined as:

$$F(k) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \exp(ikx) f(x)dx = \widehat{f(x)} \tag{1}$$

Its inverse is defined as

$$f(x) = \widehat{\widehat{f(x)}} = \frac{1}{2\pi}\int_{-\infty}^{\infty} \exp(-ikx) F(k)dk \tag{2}$$

The transform maps a function into trigonometric with frequency number k on frequency domain. The transform assumes the function to be periodic on certain interval, like $[0, 2\pi]$ or $[a, b]$. The periodic behaviour requirement of the function comes from the basis function, trigonometric.

However, numerically, the mapping will not happen on a infinite domain, instead it will be on discrete number of frequencies. Firstly, the integral form of the fourier transform can be rewritten:

$$F(k) = \sum_{k=-\infty}^{\infty} c_k exp(ikx) \tag{3}$$

where $c_k = \frac{1}{2\pi}\int_0^{2\pi} \exp(ikx) f(x)dx$. The series needs to be truncate in order to be implemented numerically. We can define a uniform grid from $[0, 2L]$ in N grid on which the transformation is

Date: January 29, 2022.

applied and the resulting equation is

$$F(k) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} c_k exp(\frac{ikx}{L}) \tag{4}$$

with $c_k = \frac{1}{2L}\int_0^{2L} \exp(\frac{ikx}{L})f(x)dx$. So, we can write $\widehat{f(x)}$ as :

$$\widehat{f(x)} = [C_{-\frac{N}{2}}, C_{-\frac{N}{2}+1}, ..., C_{N-1}] \tag{5}$$

Where the variable C represents the discrete fourier transform (DFT) coefficient values on each function evaluated on the time domain grid.

The algorithm that will be implemented is called fast fourier transform (FFT). It was developed in 1960's to calculate DFT in an efficient and accurate way. The way that FFT computes is the following. Consider function $f : [0, 2\pi] \to \mathbb{R}$. Let $f_n = f(x_n)$ for n=0,1,2,...,N-1. Then FFT reforms the coefficient as

$$\tilde{c_k} = \sum_{N}^{N-1} f_n(exp(\frac{2\pi ink}{N})) \tag{6}$$

In short, the algorithm only looks at the function value f_n and N, the number of points. The FFT algorithm definition above is for Python Numpy package and Matlab. The convention for the k value for these two are centered around point k= $\frac{N}{2}$ instead of $k = 0$ so, one needs use the built-in function to shift around the position such that the frequency number k is centered around $k - 0$.

Given a noisy signal in time $t \in [0, \tau]$, we can define a window function (filter) such that the noisy part of the signal can be attenuated and data can be extracted for it. In the time domain, this is known as convolution. The transformation of convolution to frequency domain is knowm as Gabor's Transform and is given as:

$$G[f](\tau, k) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \exp(ikt)f(t)g(t-\tau)dt = f(t)\widehat{g(t-}\tau) \tag{7}$$

So, we can define g to be periodic and symmetric to make it easy. Furthermore, we can find g that the transform is the same as the original function such as gaussian filter. The most important thing about the filter function is the width of the function. If the width is too large, the filtered signal is not much different from the original signal. If the width is too small, we will only get limited data because the filter eliminates too much data from the signal. Thus, it is important to choose the appropriate width such that the clean data can be extracted as much as possible. For this problem, we will mainly use gaussian filter defined as:

$$g(t) = \frac{1}{\sqrt{\pi\sigma^2}} exp(-\frac{t^2}{2\sigma^2}) \tag{8}$$

where σ is the width that needs to be determined through trial and error.

3. Algorithm Implementation and Development

Python packages that are used:

- numpy [2]
- matplotlib [3]

The idea is to take FFT of the data and average of the absolute value of 3D array data. This will get rid of the white noise that is available in data. After that, the signature frequency can be found. There are 2 signature frequencies since the mapping is periodic. After that we can make a gausian filter that is centered around the two frequency location. Then apply gaussian filter by multiplying by the FFT of the signal for each time step. The gaussian filter does not need to be FFT because its FFT is still gaussian. After that, take the inverse FFT of the filtered signal at each time step.

Then we can locate the trajectory of the submarine by finding the index of the highest pressure location at each time step. From the index then, the path of submarine can be mapped based on the index of the pressure data point because the cubic grid is known.

4. Computational Results

This section will discuss the results and figures. Figure 1 shows the location of the frequency signature of the signal. The index location of the frequency signature is at (39,49,10) and (26,17,55).

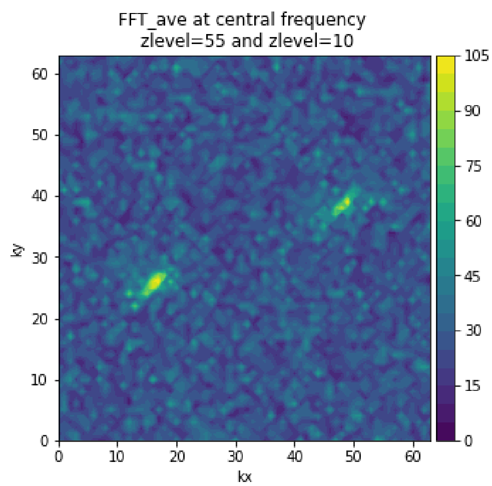


Figure 1. Figure shows the signature frequency from the average of FFT signal in 3D array. The plot is done by adding the slice at z=10 and z=55.

Figure 2 shows plots of the original average fft signal at one of the frequency signature, the gaussian filter , the resulting filtered average fft at one of the frequency signal and the original fft at one time and z-slice. The gaussian filter uses $\sigma = 1.5$. The gaussian filter only uses (39,49,10) as its center frequency instead of both center frequency.

Figure 3 shows trajectory of submarine from different angles.

5. Summary and Conclusions

From the submarine signal data, we found frequency signature at indices (39,49,10) and (26,17,55). After applying gaussian filter, we found the trajectory of the submarine. The gaussian filter works nicely in filtered most of the noise in the signal. In the future, we can apply a more sophisticated filter such as wavelet to the data and we can compare accuracy or efficiency of the filter.

Acknowledgements

The author is thankful to Prof. Hosseni for helpful discussion on the theoretical background of the problem. The author is thankful to teaching assistant (TA) in providing notes for the problem. Lastly, the author is grateful for the discord community of amath 482/582 in providing guidance and help to do the problem and other programming issue.

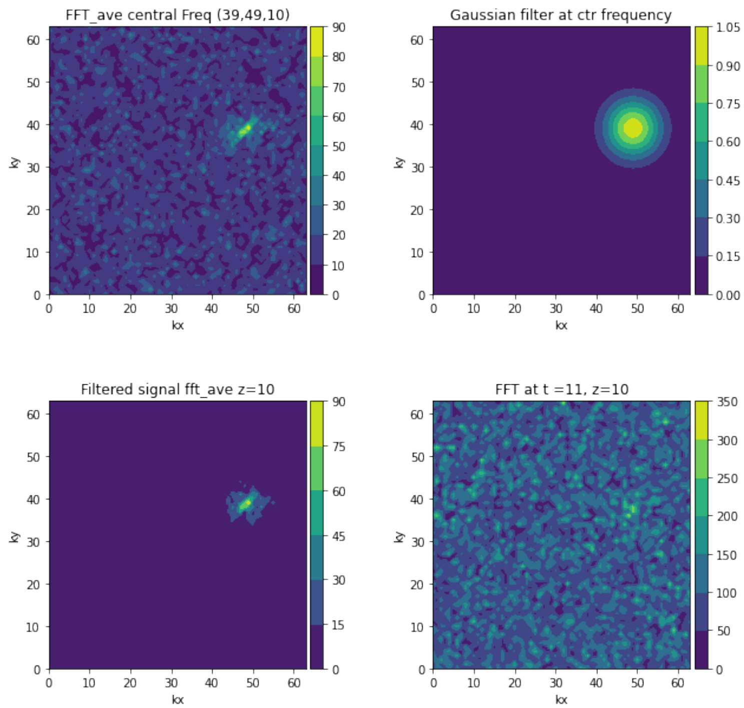
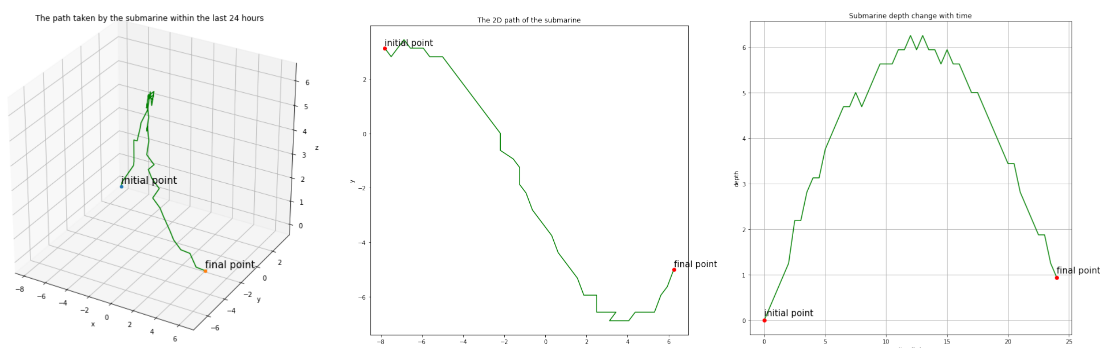


FIGURE 2. Different FFT plot and gaussian filter plot



(A) 3D path of the submarine path

(B) 2D path of submarine from top view

(C) The depth change of submarine over 24 hour period

FIGURE 3

REFERENCES

[1] S. Brunton and N. Kutz. *Data Driven Science Engineering*. databook.uw.edu, 2017.

[2] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020.

[3] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.