HUIYI WANG
ID: 261-016-753
PHYS 512
Oct 28th 2021

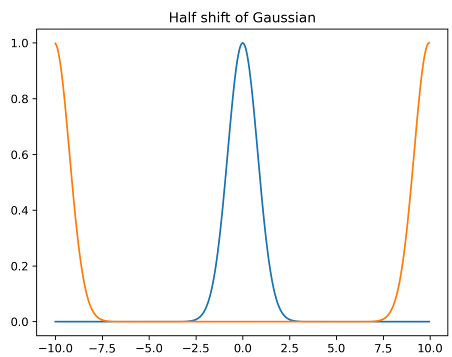
Problem Set 5

1 Shifting by Convolution

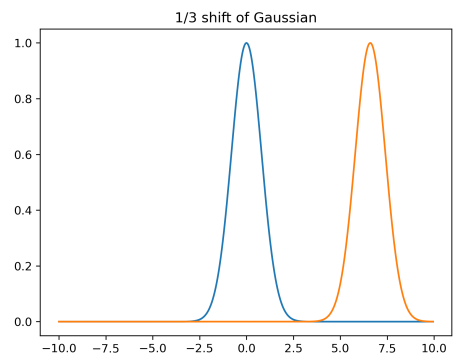
In order to shift a function via convolution, we need to convolve our original array with a delta function. With a shift of x_0, we have the following expression:

$$f(x-x_0) = f(x) * \delta(x = x_0) \tag{1}$$

In Figure 1, we plot the Gaussian function of the array before and after the shift.



(a)



(b)

Figure 1: Shifting the array from 0 to 10 through convolution by half the array length (left) and a third of the array length (right).

2 Correlation of Two Gaussian

By taking the correlation of the same array twice, we have the results in Figure 2.

3 Correlation of Two shifted Gaussian

Here, we use multiple examples to illustrate how much of a shift in Gaussian would affect the outcome of the correlation function.

From Figure 3, we see that, the greater the shift from the original Gaussian function, the more aligned the correlated function is with the original array. Conversely, the less

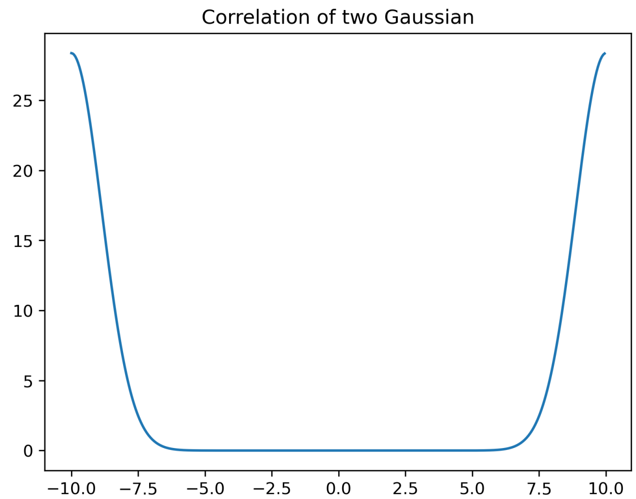


Figure 2: Correlation of Two Gaussian Function

shifted the second Gaussian, the less aligned the correlated function is. This is also evident from Figure 2. If we have two identical array, the correlated function is peaked at half array length away.

4 Wrap-Around Problem of DFT

To avoid the wrap-around problem. If we have two function f and g, each of length N_1 and N_2. We would have to make sure that each array to be padded with zeros till length N that satisfy $\boxed{N \geq N_1 + N_2 - 1}$.[1]. Hence, after transforming the function to our desired length. Of course, We could perform convolution on arrays of different length by padding $N - N_n$ zeros on each function.

In figure 4, we perform a convolution of a Gaussian function with a Sine function of different length.

[1]Following http://fy.chalmers.se/ romeo/RRY025/notes/lec7n4.pdf

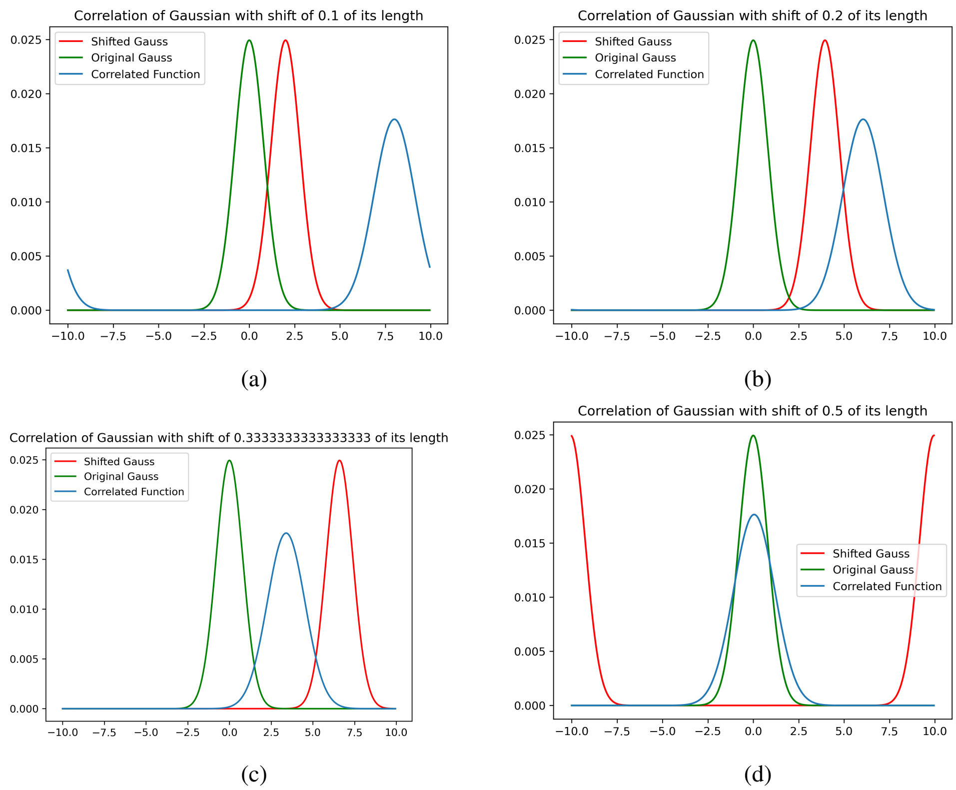


Figure 3: Correlation of two Gaussian functions, one shifted by a fraction of the length of array.

5 DFT of Sine Function

5.1 Derivation of exp

For

$$\sum_{x=0}^{N-1} \exp -2\pi ikx/N = 1 + \exp -2\pi ik/N + \exp -4\pi ik/N + ... + \exp -2\pi ik(N-1)/N$$

$$= 1 + (\exp -2\pi ik/N)^2 + (\exp -2\pi ik/N)^3 + ... + (\exp -2\pi ik/N)^{(N-1)}$$

$$= \sum_{x=0}^{N-1} [\exp -2\pi ik/N]^x \tag{2}$$

From the geometric series:

$$\sum_{x=0}^{N-1} ar^x = \frac{a(1-r^n)}{1-r}$$

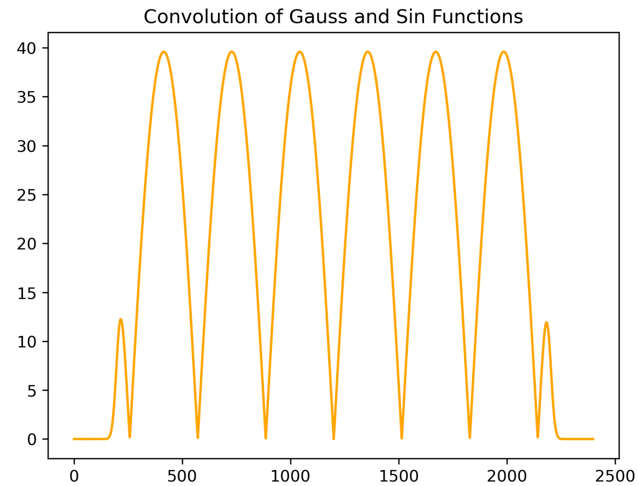


Figure 4: Convolution of a Gaussian function and a Sine function of different length. Here, we avoid the wrap-around problem by altering the length of the two arrays

We can rewrite Equation (2) as:

$$\sum_{x=0}^{N-1} (\exp -2\pi ik/N)^x = \frac{1-\exp(-2\pi ik)}{1-\exp(-2\pi ik/N)} \tag{3}$$

5.2

When $k \to 0$, $\exp -2\pi ik/N = 1$. Hence

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

When k is not a multiple of n, $\exp -2\pi ik/N = 0$. Hence.

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

5.3 Non-integer Sine Wave

Here, we are interested in plotting a sin wave, where k is not an integer:

$$f(x) = \sin \frac{2\pi ikx}{N} = \frac{e^{i2\pi kx/N} - e^{-i2\pi kx/N}}{2i} \tag{4}$$

The Fourier Transform of Equation (4) gives:

$$F(k) = \sum_{x=0}^{N-1} \frac{e^{i2\pi kx/N} - e^{-i2\pi kx/N}}{2i} \exp\left(-2\pi ik'x/N\right) \tag{5}$$

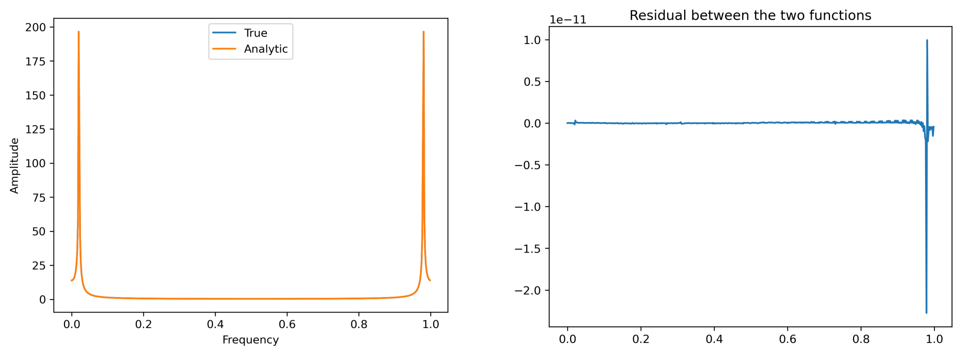
Plugging in Equation (3), we have our analytic solution:

$$F(k) = \frac{1-\exp\left(-2\pi i(k'-k)\right)}{1-\exp\left(-2\pi i(k'-k)/N\right)} - \frac{1-\exp\left(-2\pi i(k'+k)\right)}{1-\exp\left(-2\pi i(k'+k)/N\right)} / (2i) \tag{6}$$

Figure 5 compares the DFT of Equation (4) with our analytic solution (Equation (6)), with its residual on the right. Picking $k = 10.4$ and $N = 512$, we have

The residue between the functions are: 1.107571029710517e-12, with N = 512.

We see that the closer to the right end, the higher the residual. In addition, if we decrease N, the residual would become better $\sim 10^{-15}$.



(a)

(b)

Figure 5: Comparing the DFT of a non-integer sin function of python package and analytic solution. The residue of the two curves on shown on the left.

As we can see, Figure (5a) does not resemble an exact delta function solution around the edges. This is the phenomenon of spectral leakage.

5.4 Window Function

To eliminate the spectral leakage problem, we multiply Equation (4) with a window function:

$$f_{win} = 0.5 - 0.5\cos(2\pi x/N) \tag{7}$$

Hence we would perform a DFT of the new function:

$$f(x) = \sin\frac{2\pi i k x}{N}(0.5 - 0.5\cos(2\pi x/N)) \tag{8}$$

In Figure 7, the new DFT of the non-integer sin function now behave almost exactly like a delta function.

5.5 Fourier Transform of Window

Here, by performing a Discrete Fourier Transform of Equation (7), we have:

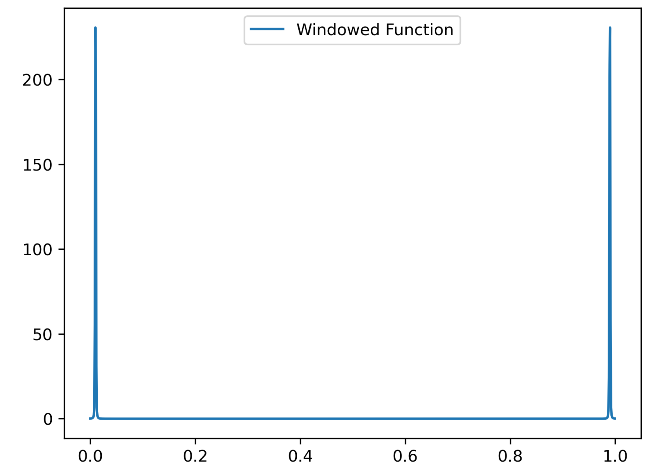


Figure 6: The Discrete Fourier Transform of a non-integer sin function overlapped by a window function.

```
By taking the DFT of the Window function,
we have the first two terms as:[256. -128.]
and the last terms as -128.0.
```

All other terms in the middle appears 0. Since we take $N = 512$ in this case, we have the array distributed as: $[N/2, -N/4, 0, 0, 0, ..., -N/4]$

6 Random Walk

We know that the power spectral of a random walk is proportional to $1/k^2$. We could derive this through the Brownian Noise Model.

$$
\begin{aligned}
P[f](k) =& |F(f)(k)|^2 \\
F[\frac{df}{dx}](k) =& \frac{1}{\sqrt{2\pi}} \int_{\text{inf}}^{\text{inf}} \frac{df}{dx} e^{-iwx} dx \\
=& \frac{1}{\sqrt{2\pi}} f e^{-ikx} - \frac{ik}{\sqrt{2\pi}} \int_{\text{inf}}^{\text{inf}} \frac{df}{dx} e^{-iwx} dx \\
=& -ikF[f](k) \\
P[df/dx](k) =& k^2 P[f](k) \\
P[f](k) =& |F(f)(k)|^2 = C/k^2
\end{aligned}
$$

We take $n = 10000$ random walk data and take the Fourier transform.The window function we use follows Equation (4). In figure 7, we plotted both the windowed and unwindowed function together at the predicted model $y = 1/x^2$

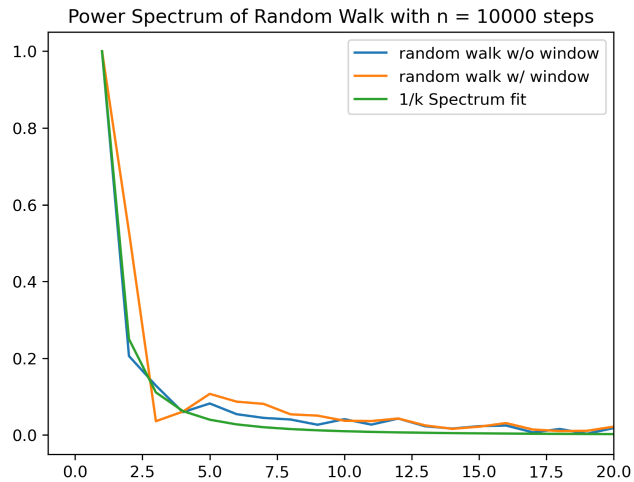


Figure 7: The Fourier Transform of the random walk. This only shows the first 20 data points of the 10000 points that we use. Both the windowed and unwindowed function shows a correlation of $1/x^2$

A $1/x^2$ model gives a $\chi^2 = 1.1024238254604048$ for unwindowed transform, which is desirable. A $1/x^2$ model gives a $\chi^2 = 2.210992341963131$ for windowed transform, which is desirable.

This is because, for a non-zero slope line, the Fourier transform is also to $1/x^2$. Hence, a random walk with a slight slope would behave as we expect without windowing.