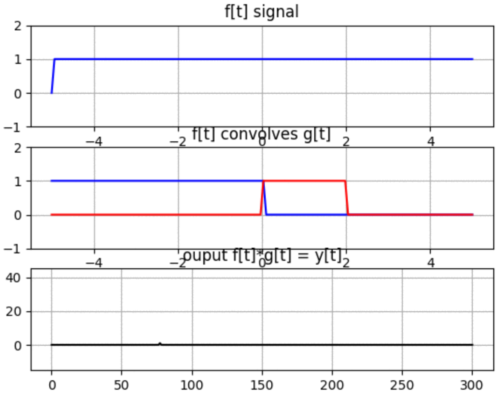
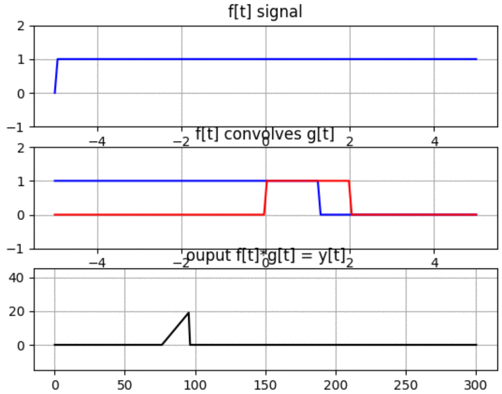
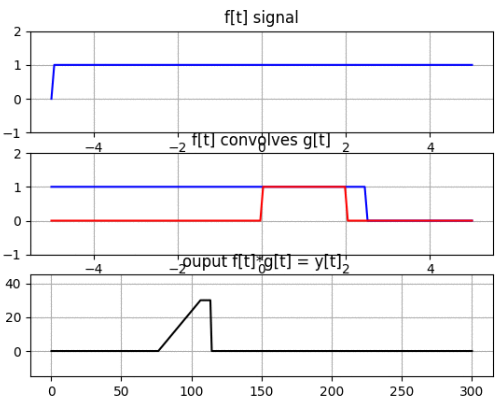
Vishnu Banna
vbanna@purdue.edu
ECE 301 Computational Assignment
Prof. Zubin Jacob

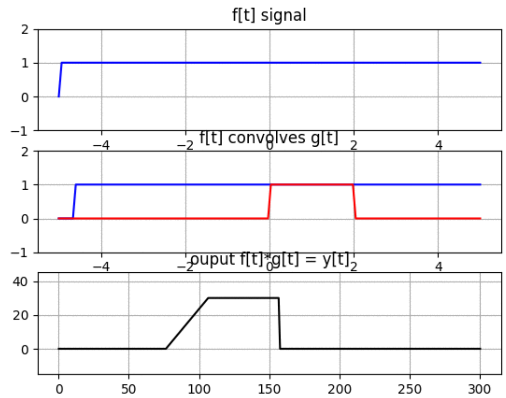
Problem 1::

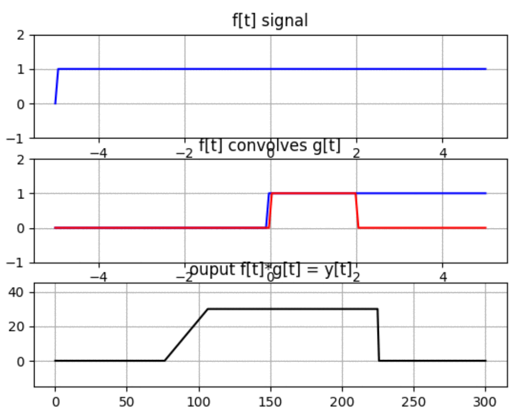
- This problem askes to show the graphical convolution of 2 signals.
- The next slide will show frames from the convolution animation
- Convolution is the point by point combination of 2 arbitrary signals as one signal is passed though the other. Graphical Convolution visually shows how a current or voltage signals react/change as it is input or passed though a circuit.

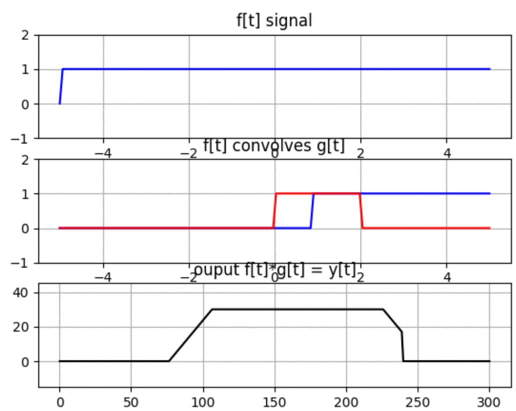

f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]

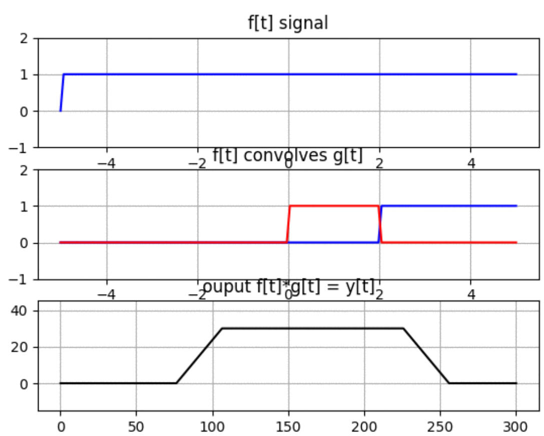

f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]

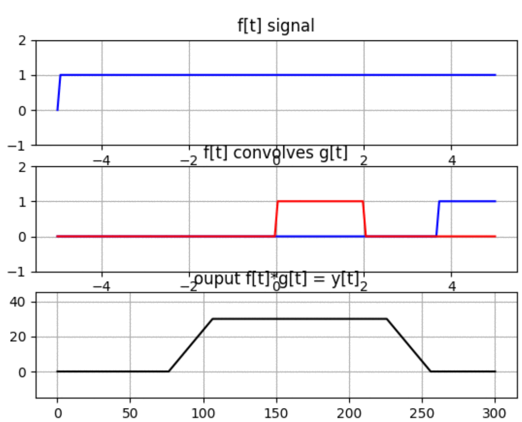

f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]


f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]


f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]


f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]

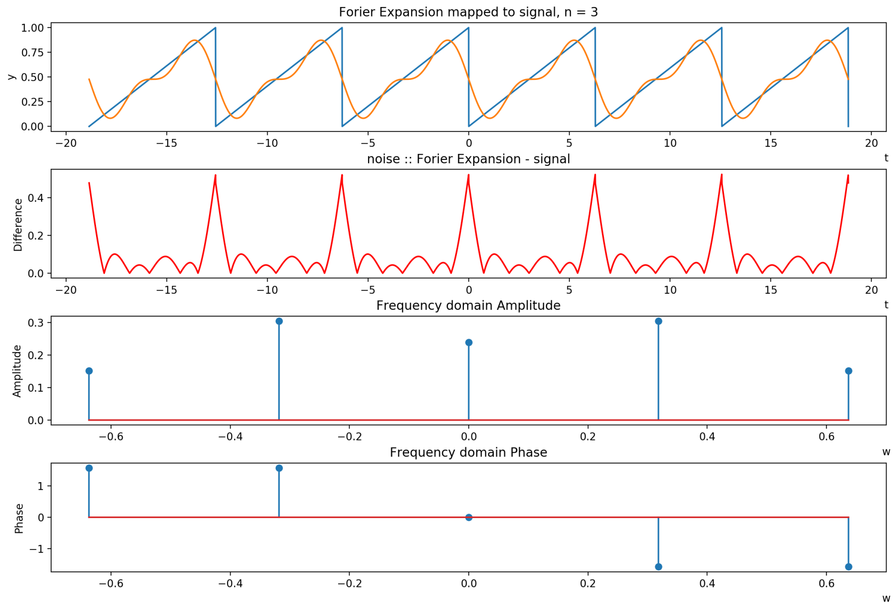

f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]


f[t] signal
f[t] convolves g[t]
ouput f[t]*g[t] = y[t]

Problem 2

- This Problem asks for the Fourier series expansion of a sawtooth function.
- The next few slides will show the expansion for 3, 5, and 10 basis signals
- In all plots the number n refers to the number of basis signals
- $\omega = 1 \; and \; T = 2\pi$ for all samples
- The final page for problem 2 is an expansion with 200 basis signals. You will find analysis for the lack of cosine functions on this page

n = 3

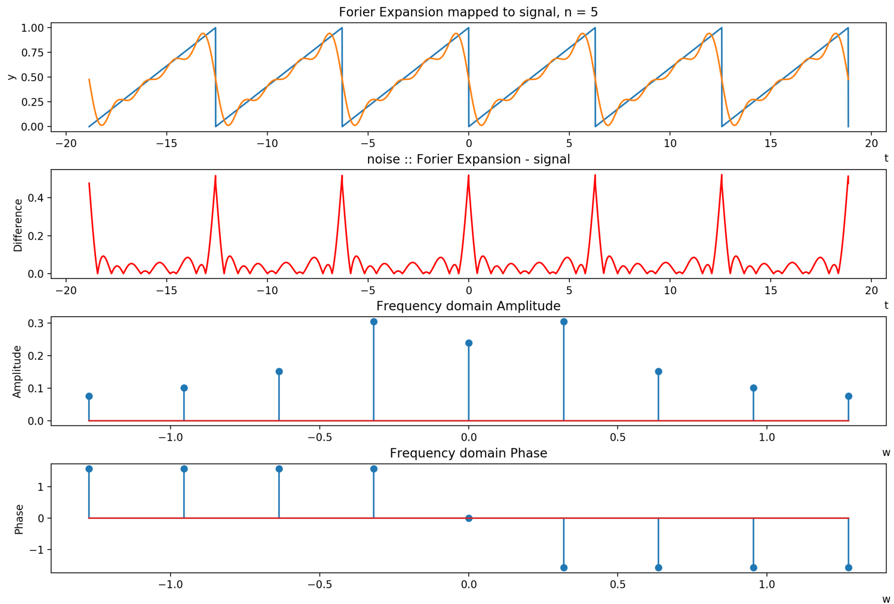


$$0.4773 - 0.3039\sin(1w * t) - 0.152\sin(2w * t)$$

Explanation:

At n = 3 basis signals, we see that the number of basis signals is not enough to accurately fit the expansion to original signal. However, the expansion is able to fit the sawtooth wave to a general sinusoidal signal. We also see a lot of distortion between the input signal and the expansion. Another item I noticed is that the cosine signals at this low basis signal value are not contributing much to the total expansion. It seems that the expansion mostly consists of to sinusoids functioning at 2 different frequencies.

n = 5

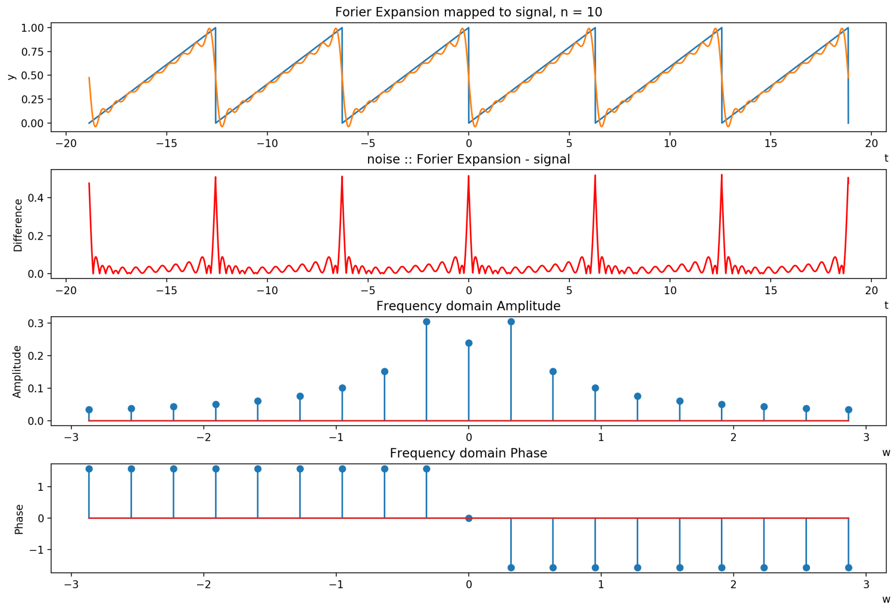


$$0.4773 - 0.3039\sin(1w * t) - 0.152\sin(2w * t) - 0.1013\sin(3w * t) - 0.076\sin(4w * t)$$

Explanation:

At n = 5 basis signals, we see that a small increase in the number of basis signals to utilize leads to an expansion with less distortion and more ability to fit to the original signal. Once again, the cosine terms offer nothing to the signal expansion. Finally, the for this specific case, we can see that as the value of omega(w) increases the amount that each basis term contributes to the expansion decreases. In other words, as |w| increases, the coefficients decrease.

n = 3; w = 2 * pi

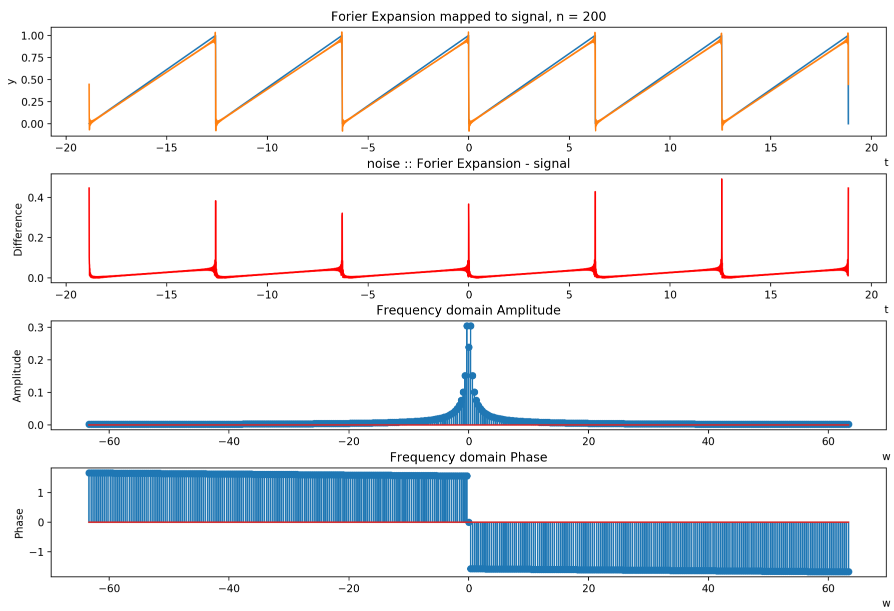


0.4773 - 0.3039sin(w * t) - 0.152sin(2w * t) - 0.1013sin(3w * t) - 0.076sin(4w * t) - 0.0608sin(5w * t) - 0.0507sin(6w * t) - 0.0434sin(7w * t) - 0.038sin(8w * t) - 0.0338sin(9w * t)

Explanation:

At n = 10 basis signals, we see there is very little distortion between the expansion and the original signal. The distortion seems to occur only at locations with very sharp angles. This indicates that lower omega basis signals in the expansion are used to fit the general shape of the original signal. As the value of |w| increases, we see the basis signals are at a higher frequency and are better able to fit to the original signals linear and instantaneous features. Once again, it is clear that the cosine function has little to no affect on the overall expansion of the sawtooth wave.

n =200



$$0.4773 - 0.0002\cos(1w * t) - 0.3039\sin(1w * t) - 0.0002\cos(2w * t) - 0.152\sin(2w * t) - \cdots$$

Explanation:

In order to confirm that there are truly no cosine functions, I choose to plot the expansion for n = 200. The main reason for the lack of cosine terms is due to the nature of the cosine function and the first step of the Fourier expansion. The first step in the Fourier expansion is to solve of a_0 or D_0. This is essentially where the Fourier Expansion accounts for any DC or constant signal offsets along the y axis. In the case of this a_0 is 0.4773. Subtracting a_0 from the saw tooth wave normalizes the function to be around the x axis. It also reveals that the saw tooth wave in its new normalized state is an odd function. There fore, the expansion components must be odd, and sin function is also odd. There fore, only the sin components contribute to the expansion of the sawtooth wave.

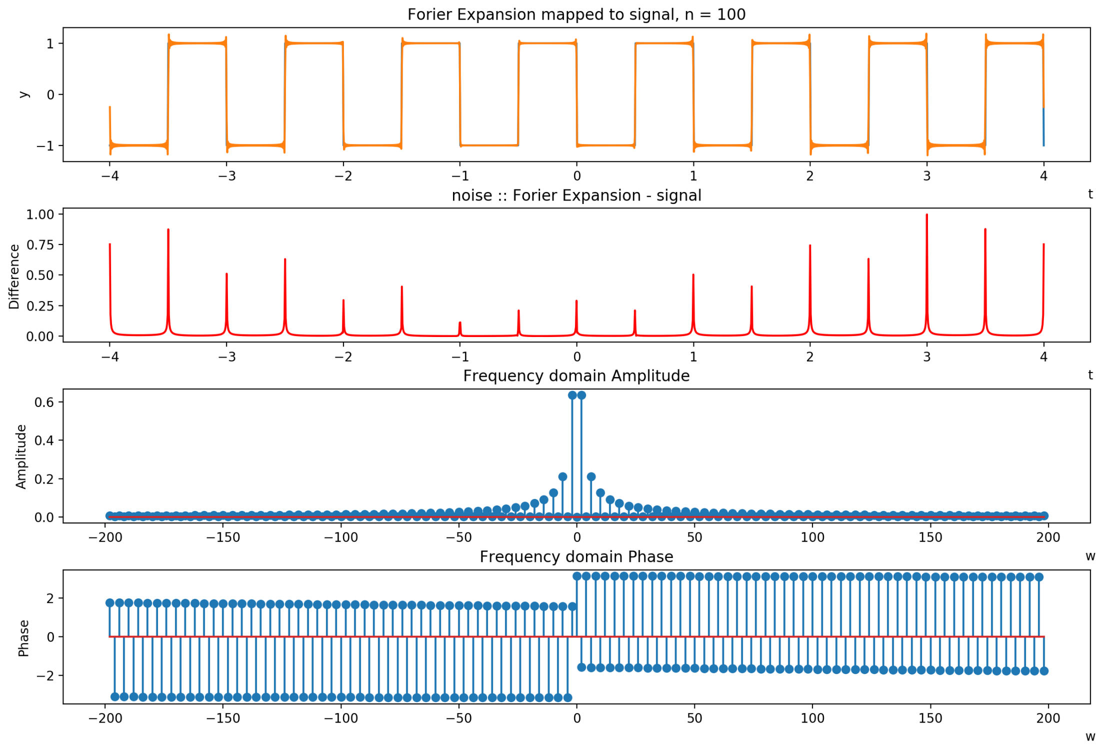
Problem 3

- This problem asks us to pass a square wave thought a low pass filter to see how the output is varied or distorted compared to the input.

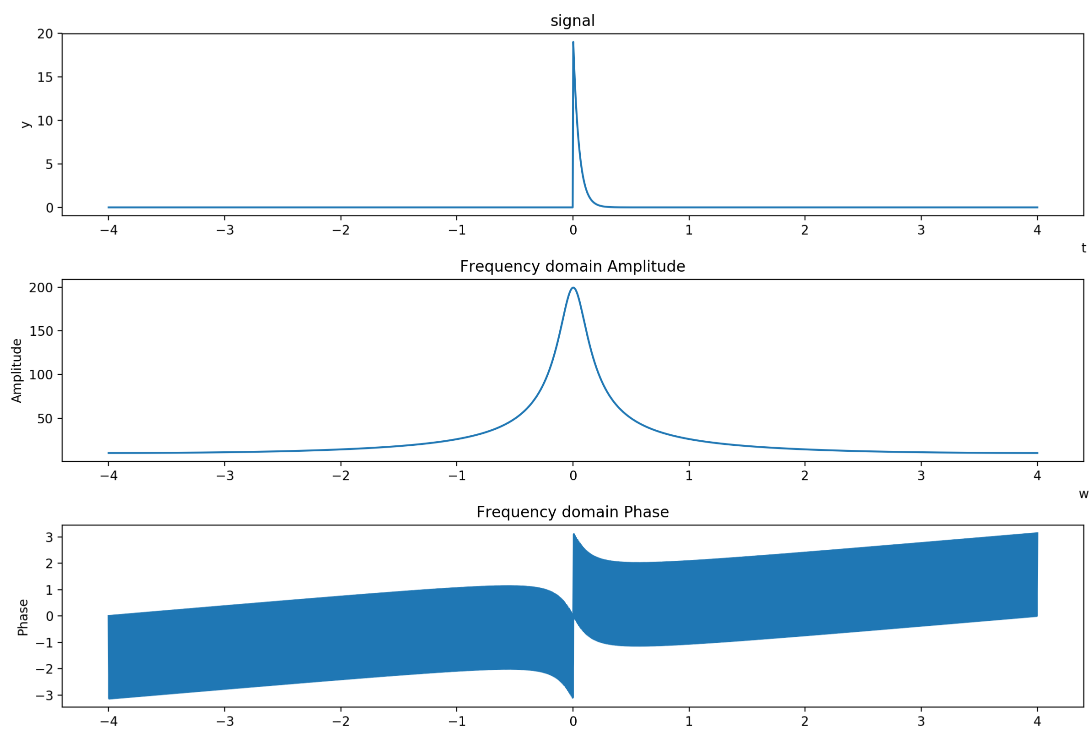
Approach::

- Take the Fourier series of the square wave input.
- Pass the explanation though the lowpass filter
- Analyze the convolved output (Final Page in Problem 3 after all plots)

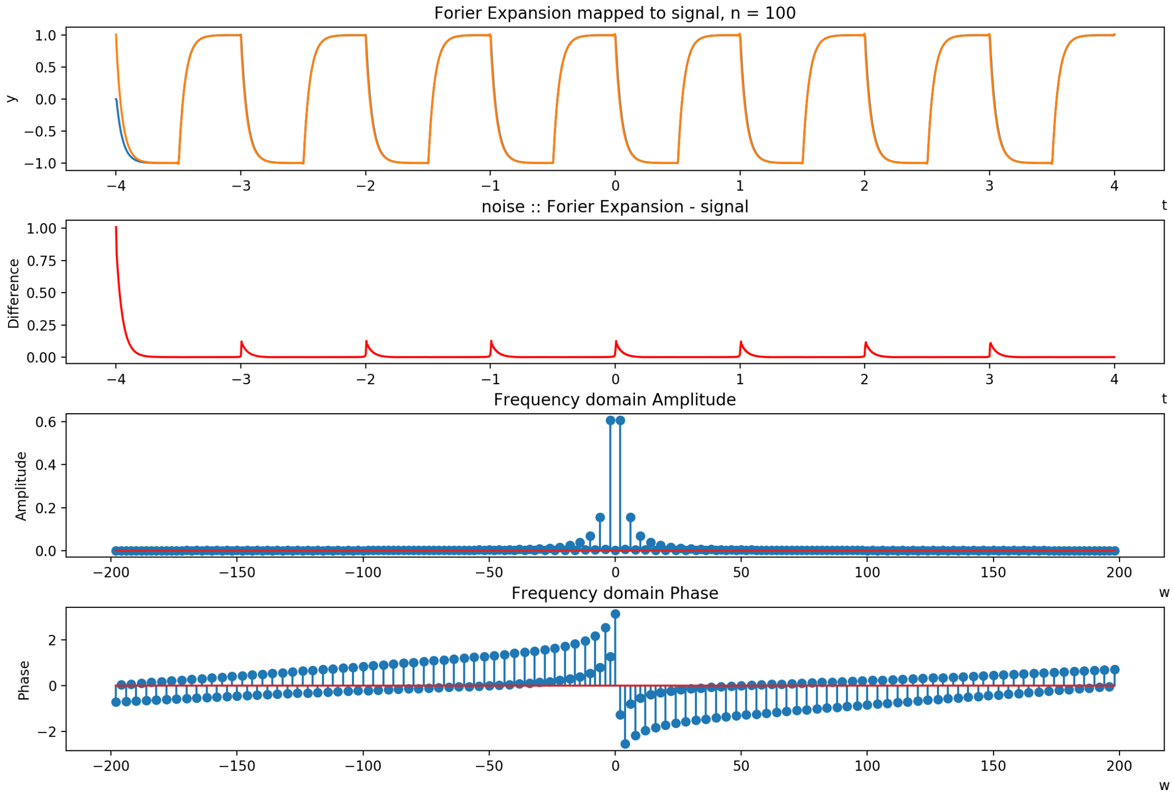
Fourier expansion of input signal



Fourier Transforms of filter

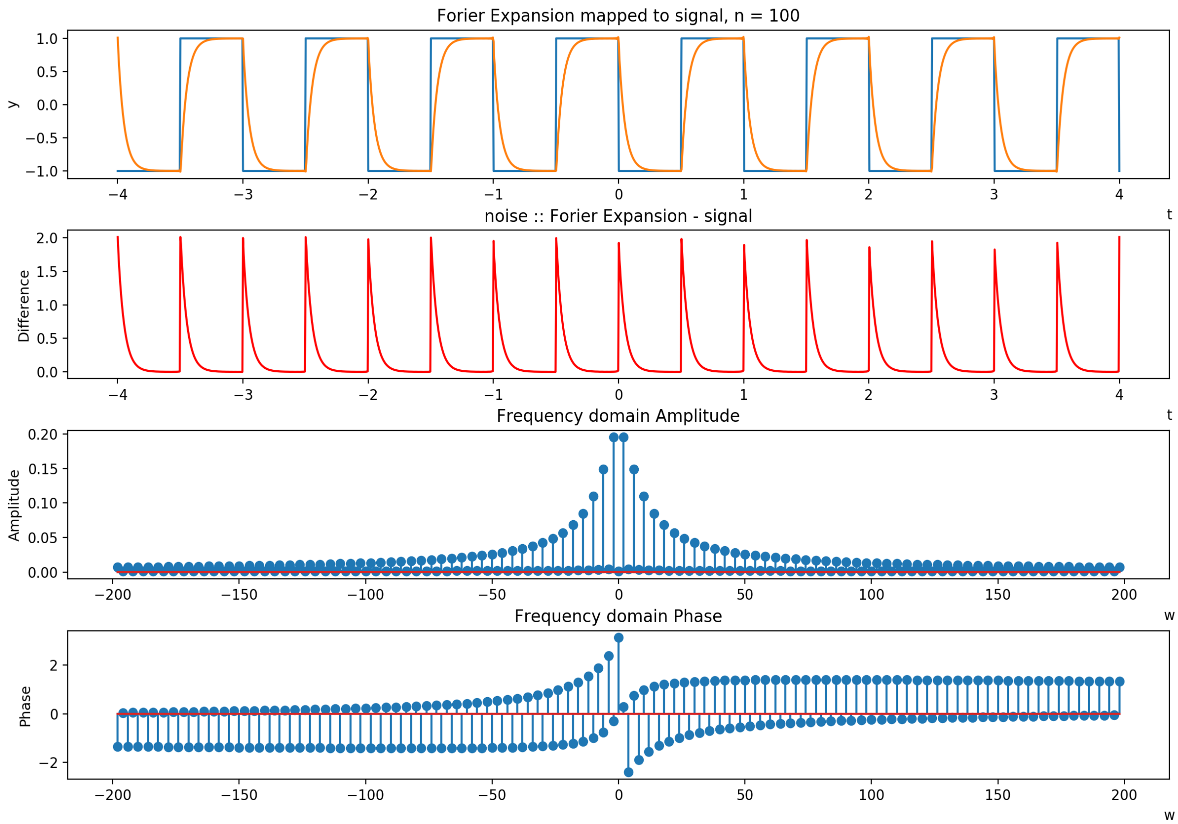


Convolution of input



Analysis on next page

Filtered output - input



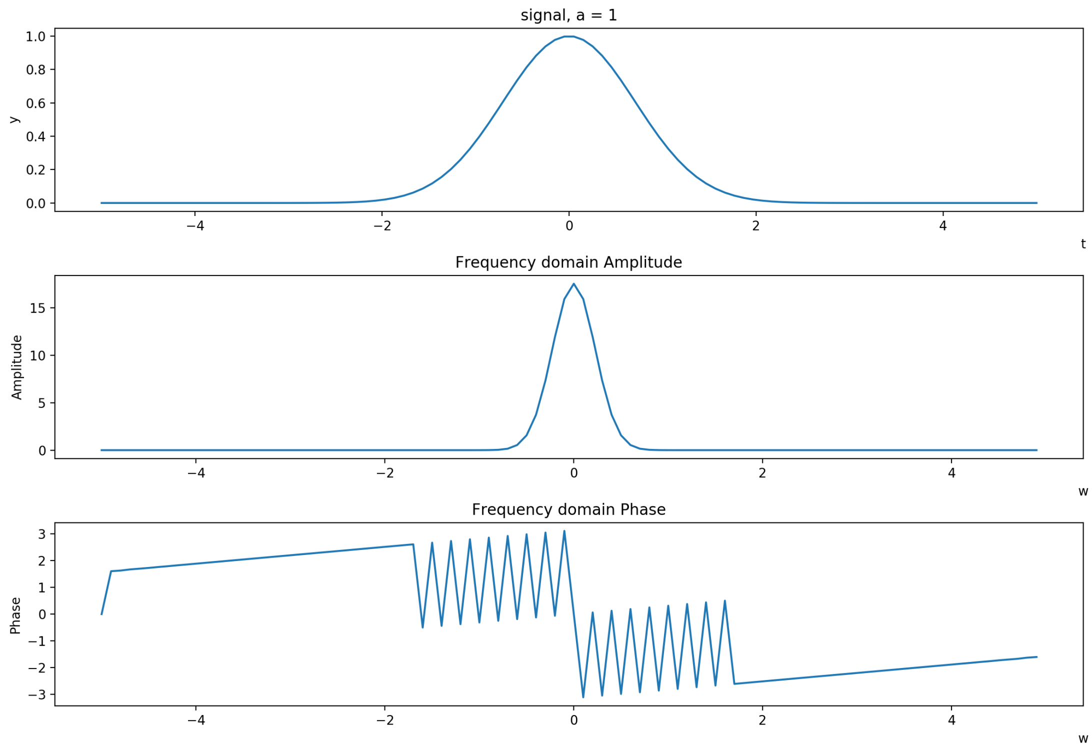
Analysis:

Based on the the Fourier transform of the filter, the low pass filter has a bell curve shape, and the phase of the filter seems to increase as the frequency increases. Given that convolution in the frequency domain is just the multiplication of 2 signals, the filter forms a mask in which certain frequencies near the bell curve are kept while other frequencies further from the 0 or the center are removed. The graph above is a graph over laying the input and the filtered output. Each square pulse is filtered and smoothed such that higher absolute value frequencies are removed. This is evident from the fact that the the left tip and the right base of the square pulse are filtered the same way but with the opposite direction. This is due to filters greater than the cutoff being filtered on one side, and frequencies less than the negative cutoff being filtered on the other side. The bottom 2 subplots are the difference between the frequency domains of the input and the convolved output. Since the difference is not 0 for the amplitude spectrum, the filter seems to have decreased the frequency amplitude of the of the input signal. The phase has also been modulated to gradually change as the frequency increases.

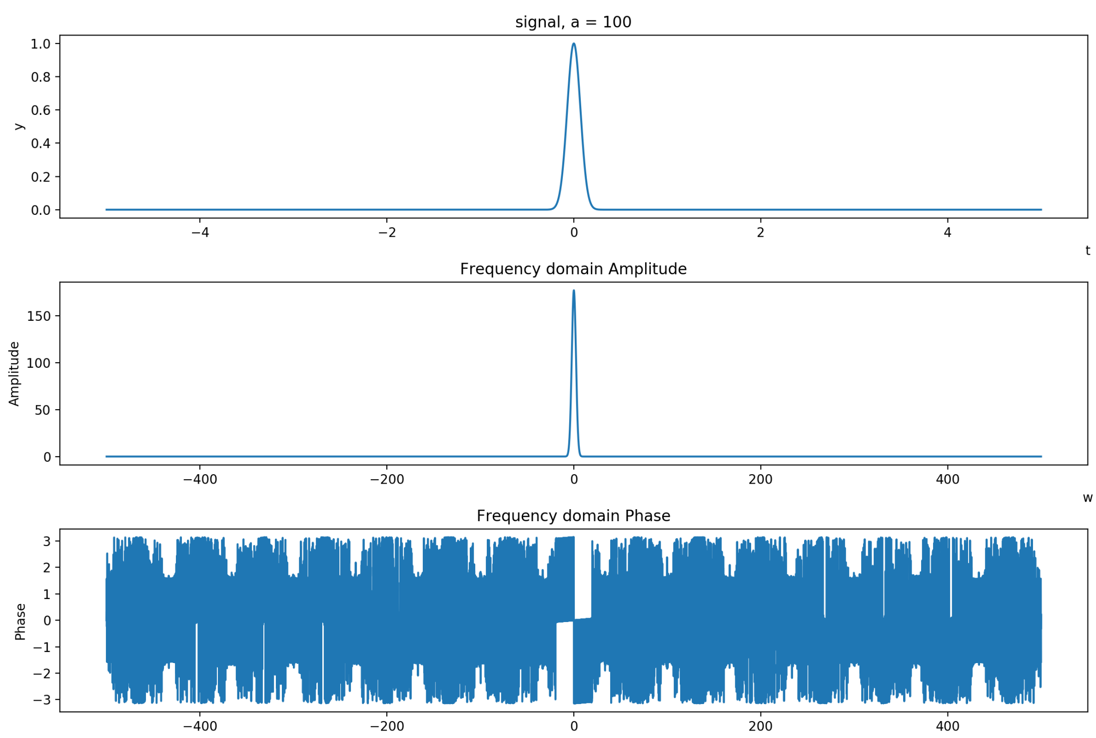
Problem 4

- This problem askes us to analyze a gaussian pulse as it relates to the time and frequency domain.

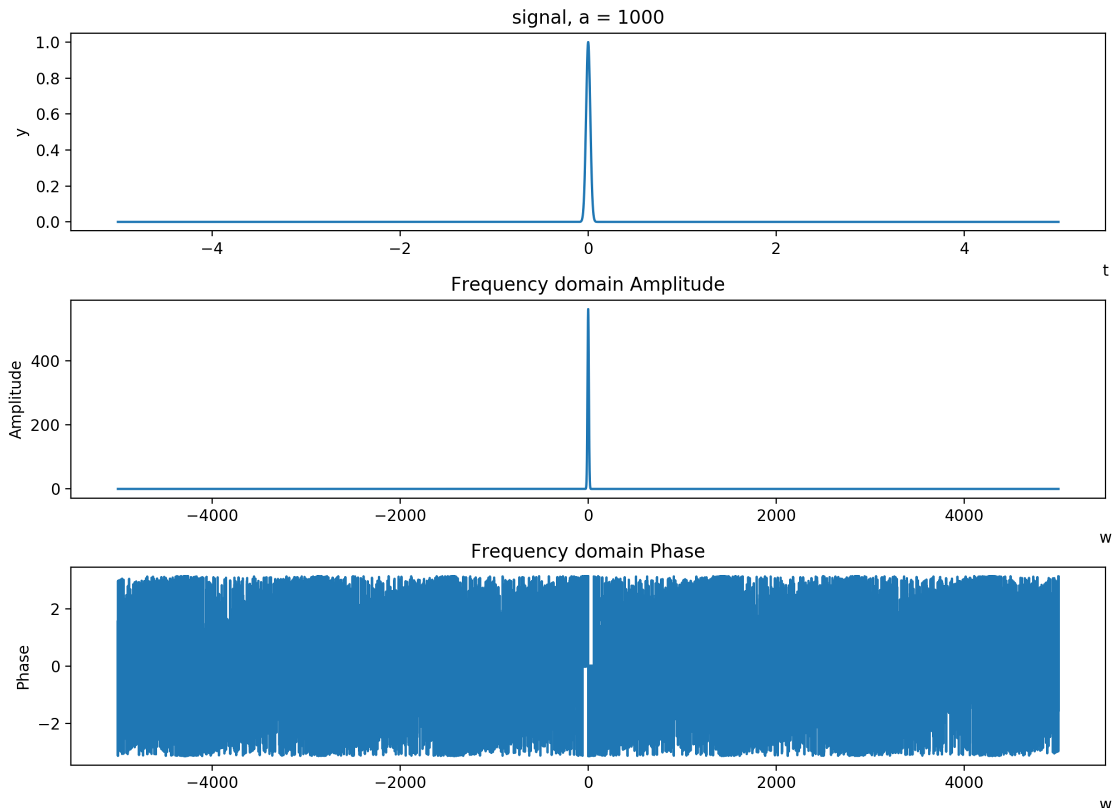
Gaussian a = 1



Gaussian a = 100



Gaussian, a = 1000



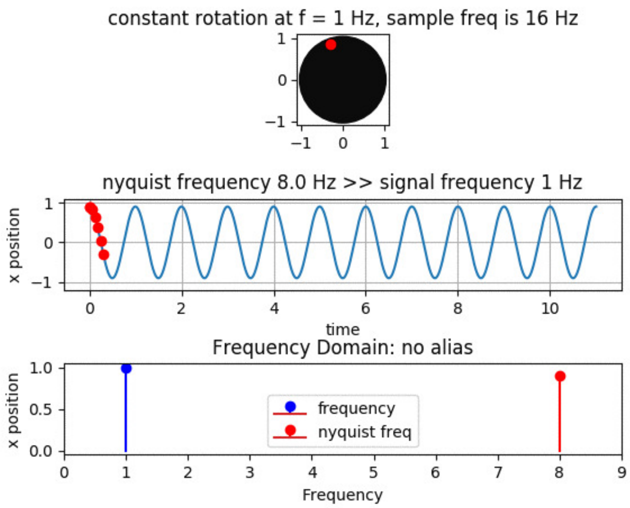
Analysis:

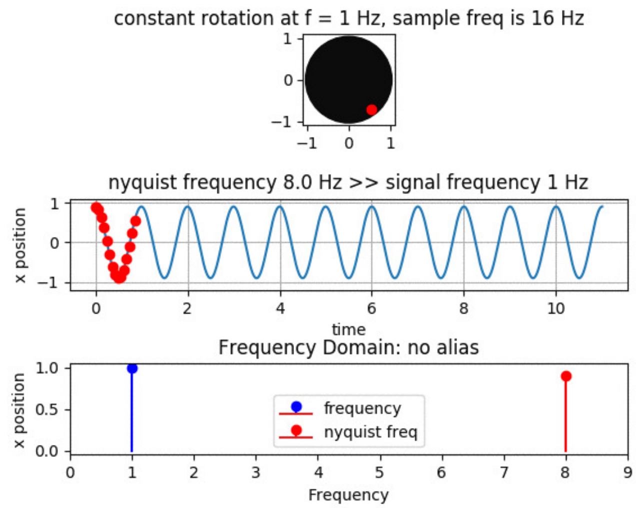
The First and most clear observation is that the Fourier transform of the a gaussian pulse is also a gaussian pulse, but the signal width is smaller. In general, the value of a increases, the pulse width decreases, the time domain and the frequency domain. However, the frequency domain is compressed much more quickly. Another interesting factor is how the phase is related to the value of a, and the gaussian signal in general. The phase of the signal doesn’t seem to follow a pattern, it seems to be random. However, the phase on the negative and positive sides of the omega axis seem to be related to each other.

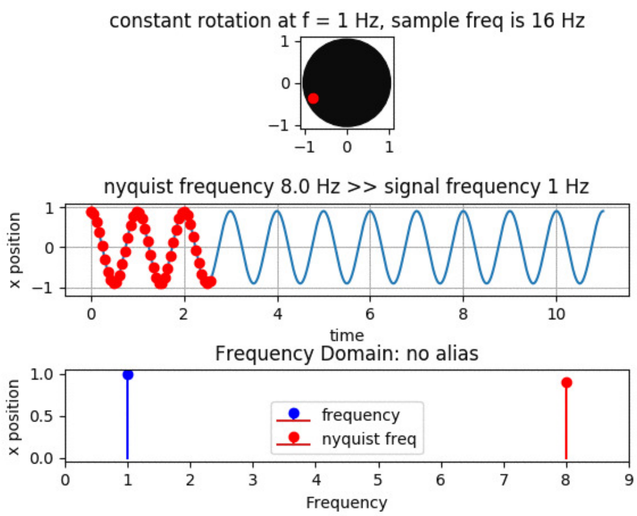
Problem 5

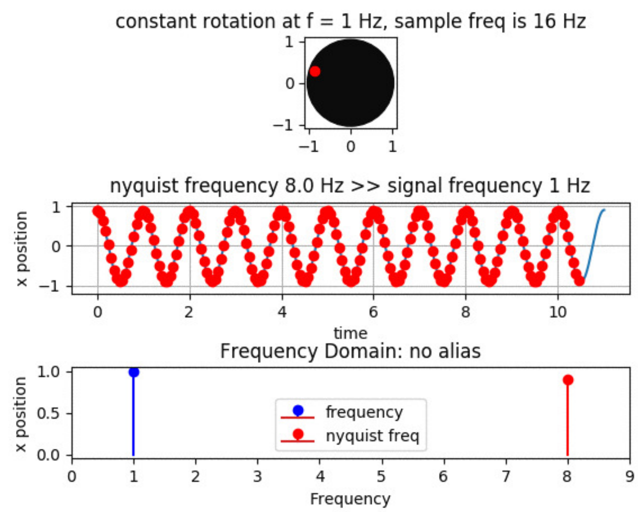
- This problem askes us to analyze the the wagon wheel effect in order to see how the sampling rate affects how a signal is recorded or perceived.
- The following pages will show various screen shots of the animation, and some analysis regarding the output of the oscillation animation under various conditions.

Case 1: Nyquist Frequency >> Signal Frequency





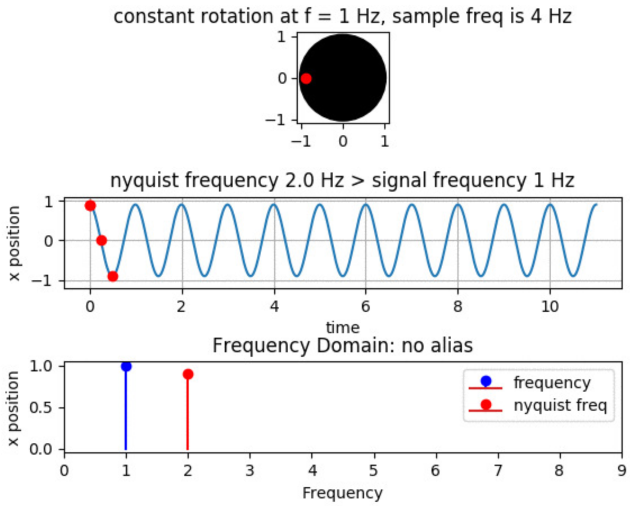


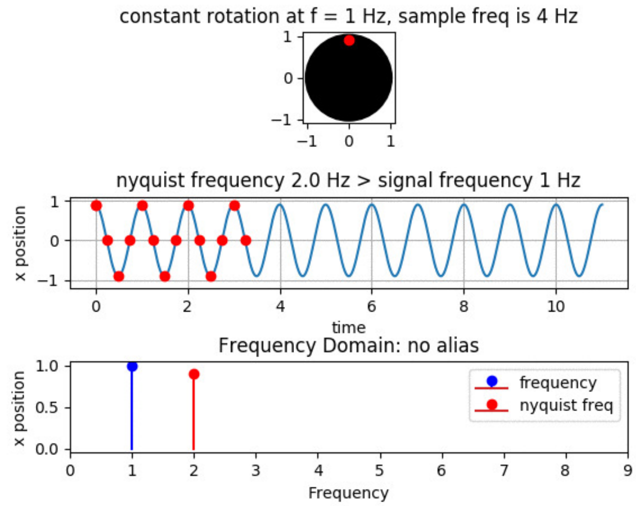


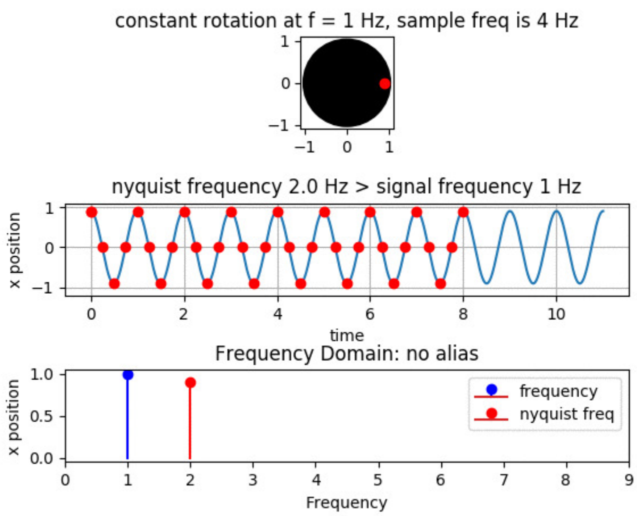
Analysis:

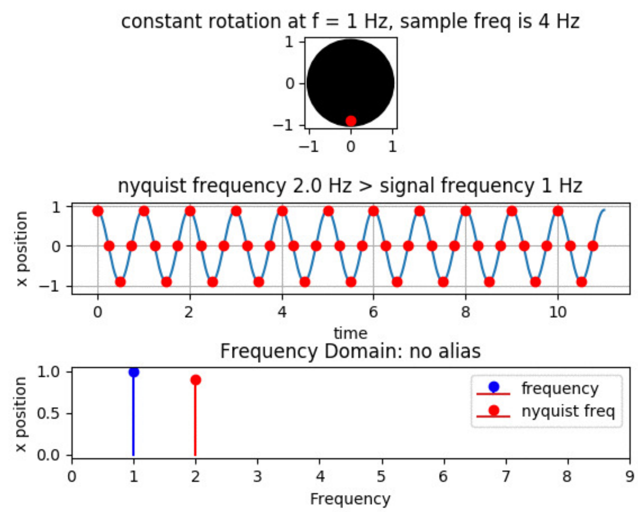
It is clear that at a high sampling rate, we are able to more clearly model the rotation of the dot or the input signal. Though the animation is not viewable due to the pdf formatting, it is clear from the mapping of the rotation of the dot to the cosine wave in plot number 2, that the time distance between each sample is very smaller. Even with out the cosine wave in the background, the wave path followed by the signal is clearly shown to be either a sine or cosine wave. The output signal is generally clear at a very high sample rate

Case 2: Nyquist Frequency > Signal Frequency





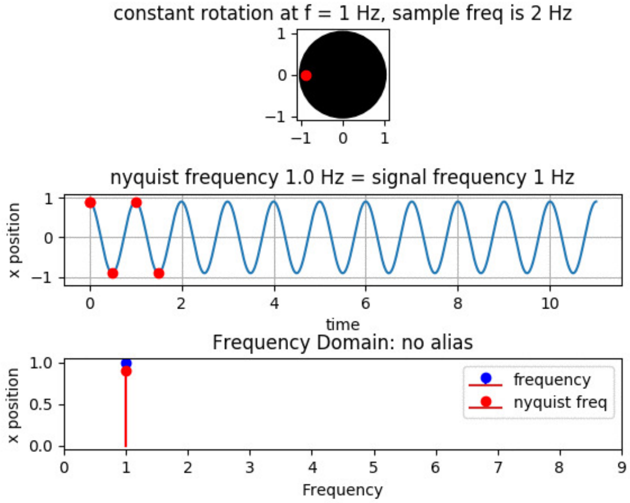


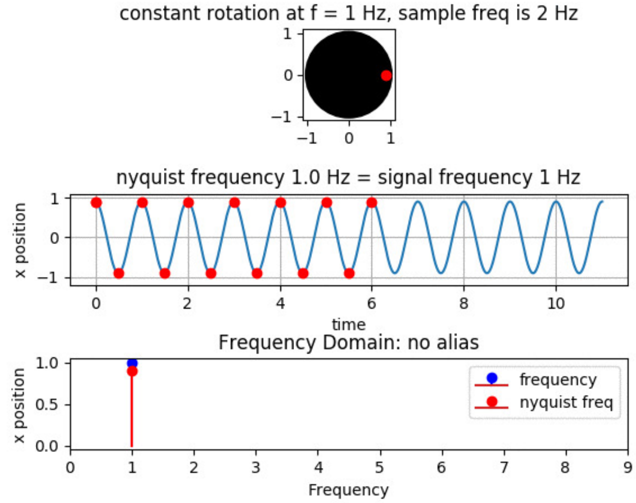


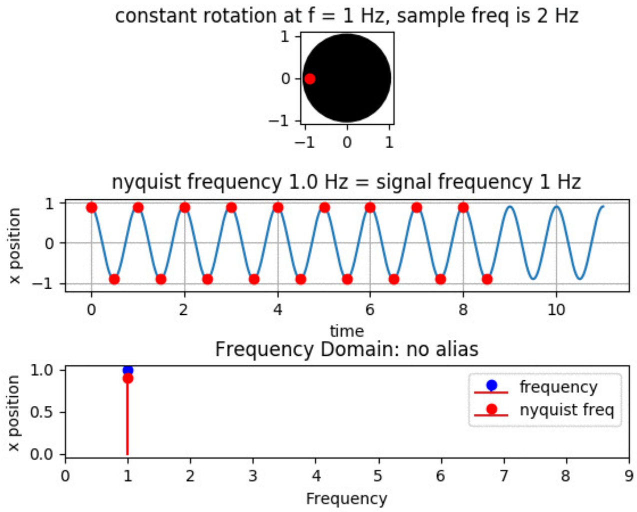
Analysis:

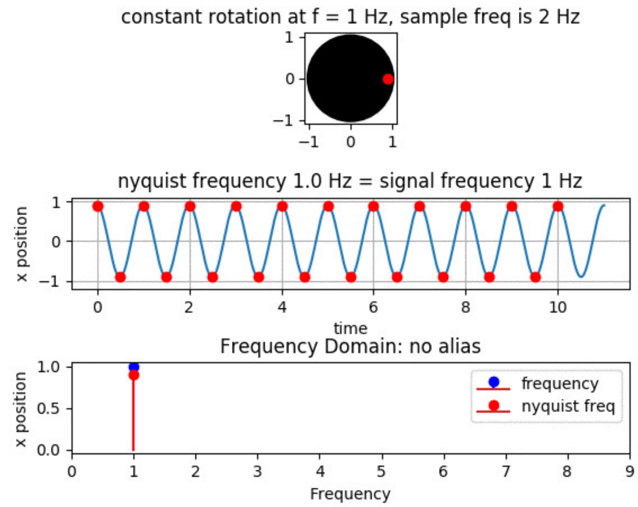
Compared to the previous case where the sampling frequency is far larger than the signal frequency, this case shows a lower sampling frequency that is closer to the signal frequency. In this case, the dot starts to jump around the circle a lot more, and spacing between the dots in plot 2 are increased. Another thing to notice is that the signal is still rotating counter-clockwise the same as it was in the previous case. Also due to the low sampling rate, based on plot 2, the samples seem to generally map the signal being produced by the rotation. However, in this case, it is unclear without path followed in the background the exact nature of the signal being sampled.

Case 3: Nyquist Frequency = Signal Frequency





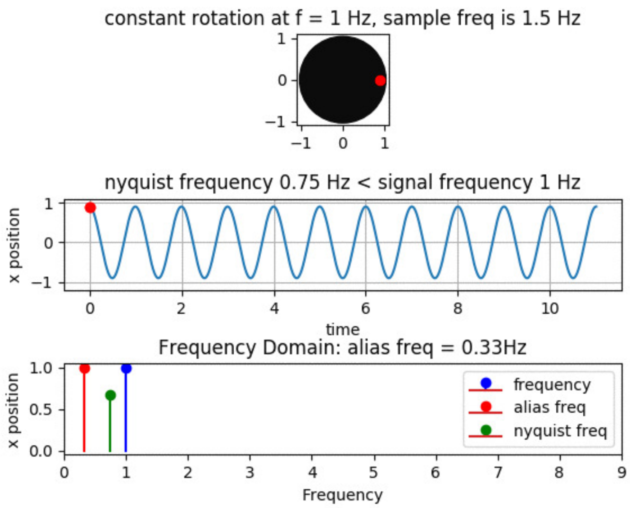


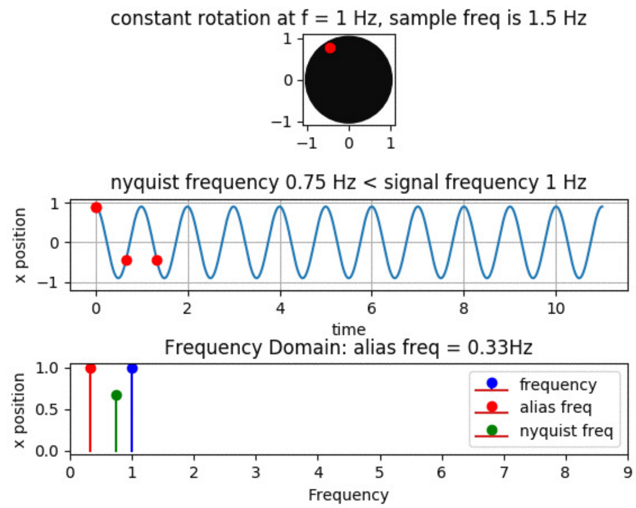


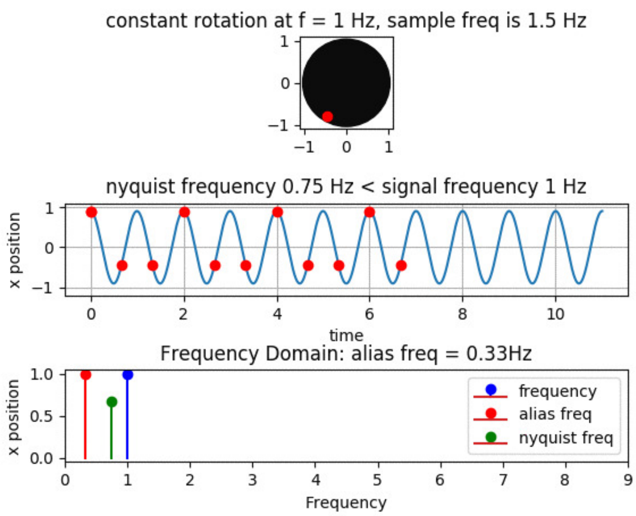
Analysis:

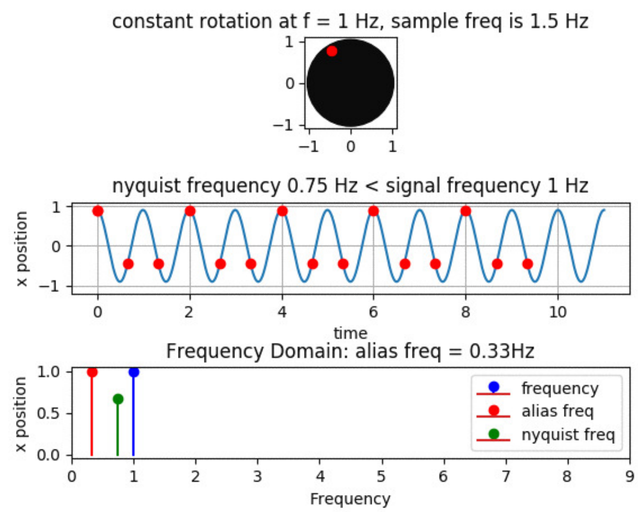
Case 3 considers the what happens when the Nyquist frequency is the same as the signal frequency. In this case we see that when the frequencies are equal, samples are taken 2 times per period. Therefore, the dot no longer rotates around the circle, but jumps from one side to the other along a single axis. Based on the samples, there is no indication of whether the dot is rotating or not. Based on plot 2, at this sampling rate, the samples only represent the peaks of the rotating dot signal.

Case 4: Nyquist Frequency < Signal Frequency





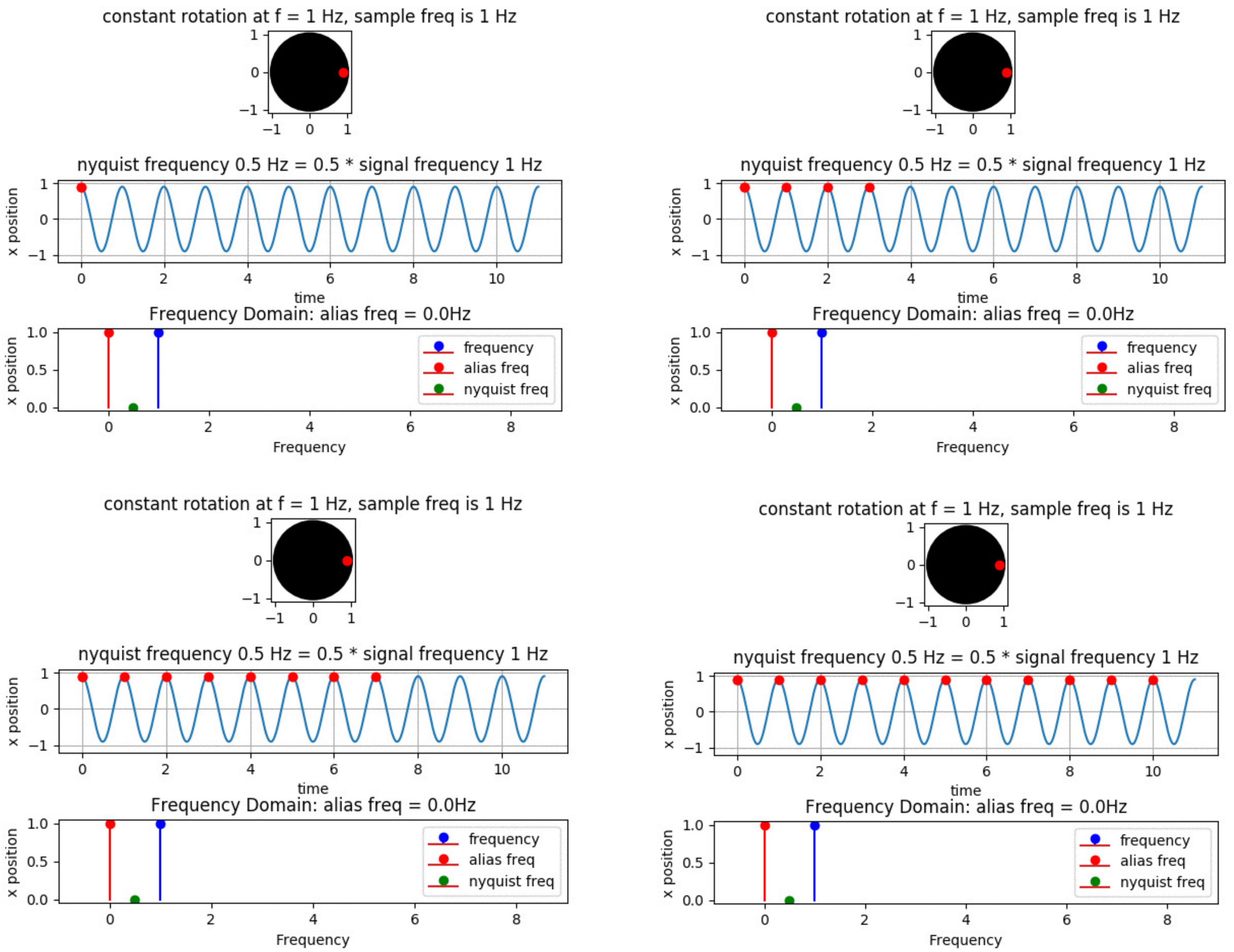




Analysis:

This is the first case where we see the dot definitively rotating in the opposite or clockwise rotation. It in this case, we see that as the dot traces its path, and like the previous case it still makes 2 samples per period. But, the distance between each sample is greater, and the sample location in each period seems to be the mirror of the sample locations in the previous period. Based on this case and the pervious case, it is clear that the Nyquist frequency equaling the sampling frequency is a limiting point at which all Nyquist frequencies above the sampling frequency sample the signal fast enough to maintain the integrity of the input signal. The Nyquist frequencies below the signal frequency are too slow and as a result you see a large distortion between the output signal and the input signal. This case also shows the first instance of an alias signal, or a signal that can be generated by under sampling the input signal in order to generate a new signal at a different frequency. The alias frequency indicates the rate at which the dot is rotating in reverse, and the rate at which the sampled signal is oscillating.

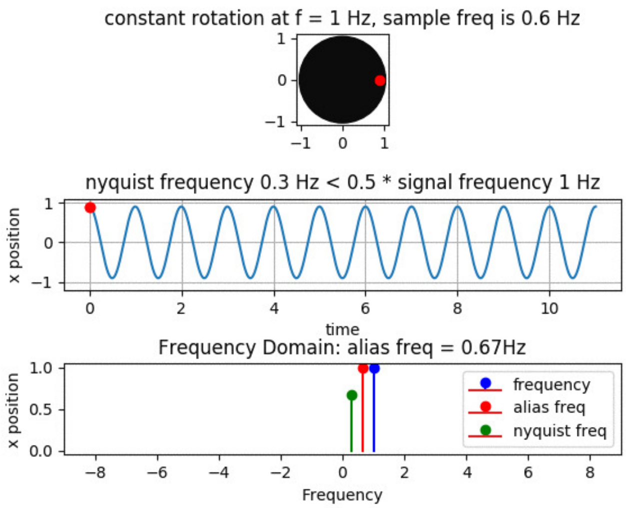
Case 5: Nyquist Frequency = 0.5 * Signal Frequency

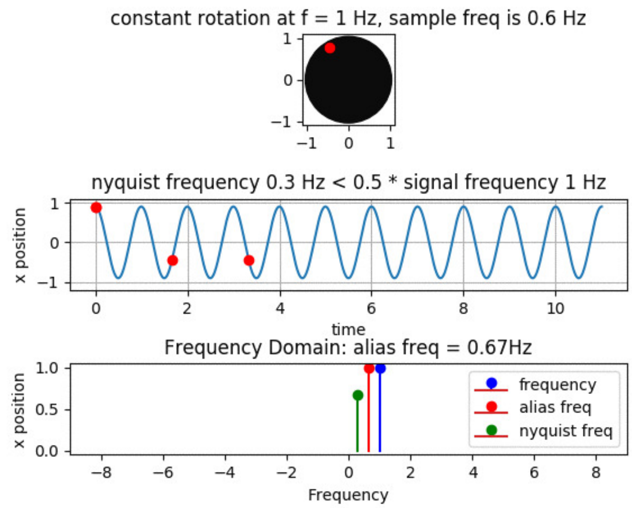


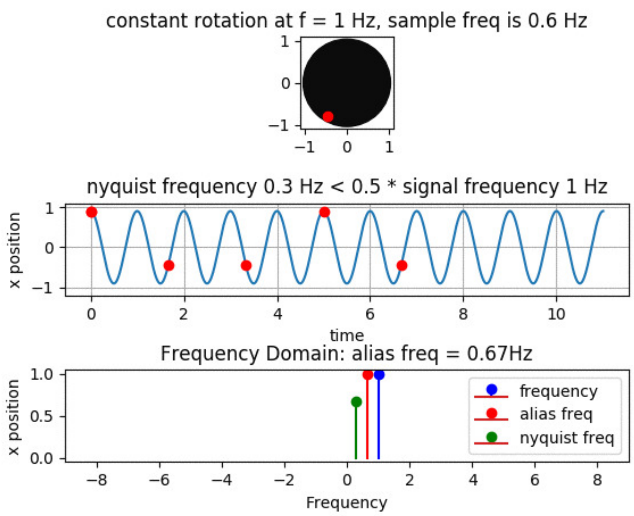
Analysis:

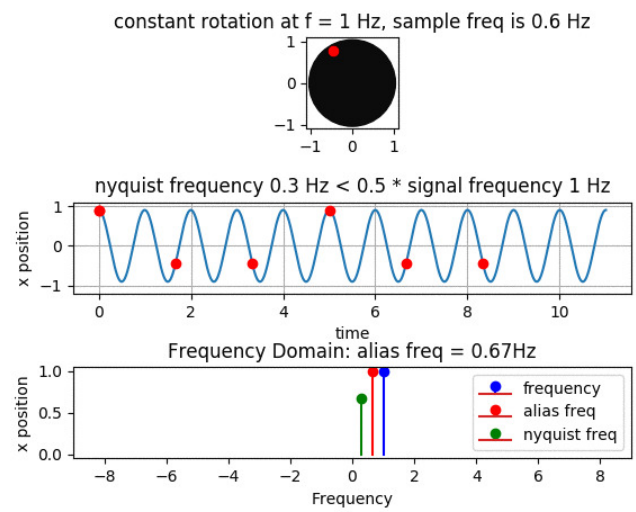
This case shows that when the Nyquist frequency is the same as the signal frequency, only one sample is taken per period. As a result the dot does not move. Another thing to notice is that frequency of the alias signal reflects this trend in that it is 0 Hz, and therefore is not oscillating. Based on this result I also tested a Nyquist frequency of 0.25 and 0.125. Both of these frequencies have the same feature where the samples lead to a linear signal where the dot does not move. This leads me to believe that all frequencies that follow the format sample rate $= \frac{signal\ frequency}{2^k}$ (where k is a whole integer) will sample the signal in such a way that the alias signals frequency is 0 Hz.

Case 6: Nyquist Frequency < 0.5 * Signal Frequency









Analysis:

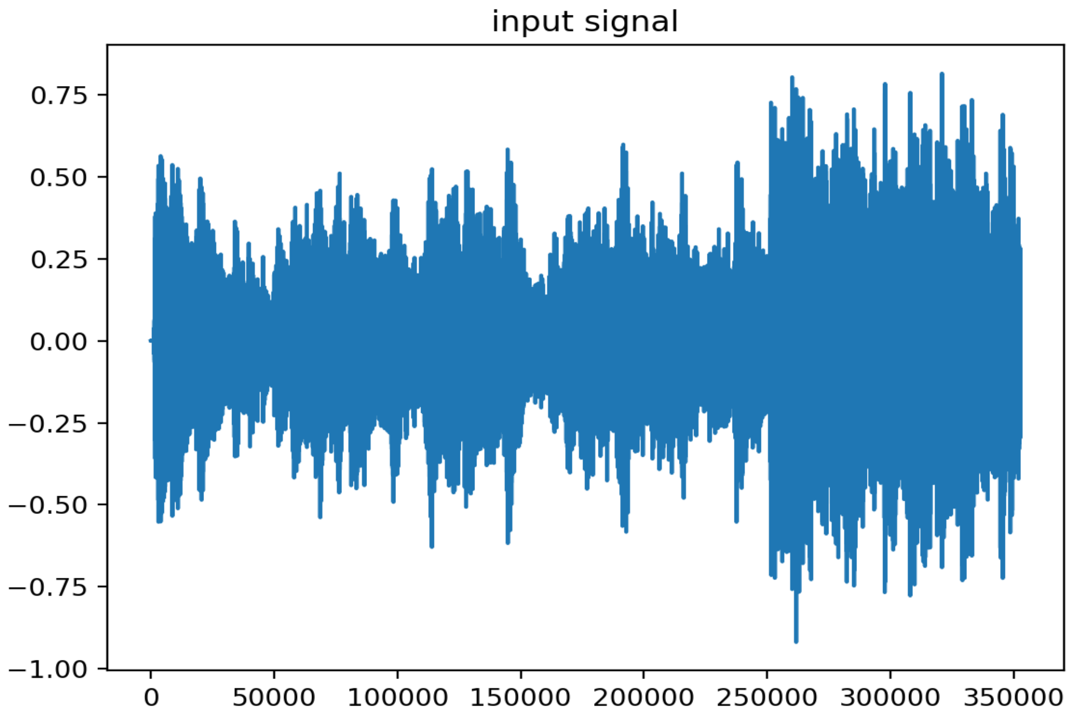
In this case, we see once again the dot is rotating, and the dot is rotating in the clockwise direction. This confirms that all Nyquist Frequencies below the Nyquist Frequency equaling the signal frequency result in the sampled signal rotating in the opposite direction of the input signal. In other words, if a signal is under sampled, the recorded or sampled signal will be a new signal that oscillates at a different frequency than the input signal. Another interesting factor is that below 0.5 * the signal frequency, samples are no longer recorded at once per period.

Problem 6

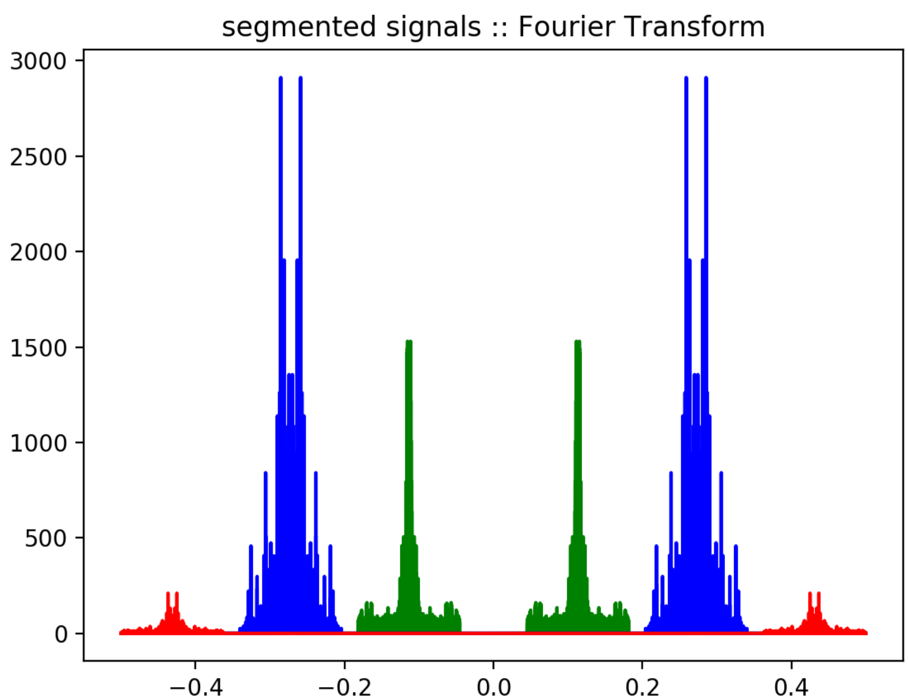
- This problem askes us to use tools like filtering, and Fourier transforms to filter a signal and extract the songs that are stored within the spectrum.
- The next few pages will go over the process I followed to deconstruct the signal.

Process

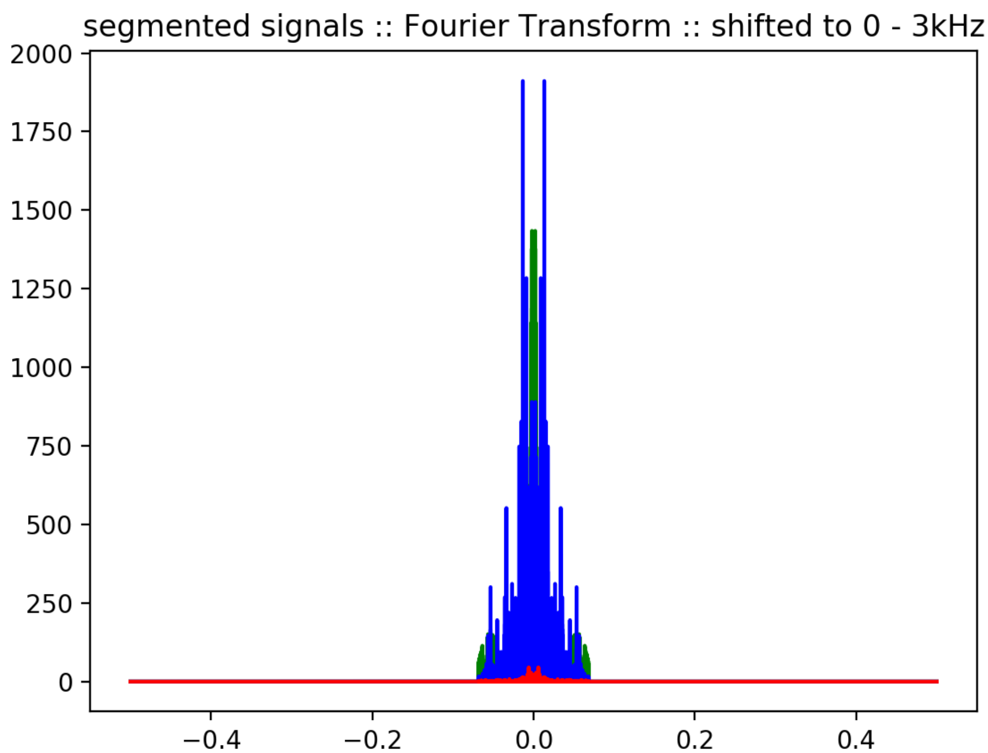
1. The first step in my process was to load the data and read the signal.



2. Take the Fourier transform and use a band pass filter to separate the songs in each frequency range. Song 1 is in red, Song 2 is in blue, Song 3 is in green



3. Shift the signal in the frequency domain such that the signal is centered at at 0hz
   - I tried 2 different methods to demodulate the signals. One was to multiply each shifted signal by the conjugate of the signal that shifted it, the other was to use a complex exponential to shift the Fourier transforms correctly
   - Both methods seemed to work, but the multiplying by the conjugate signal was more computationally efficient
4. Use a low pass filter to properly segment each song to the frequency range of 0-3kHz. Song 1 is in red, Song 2 is in blue, Song 3 is in green



Analysis:

The song in the central band at 12kHz is The Office Theme song. The other 2 songs are Never Gonna Give You Up and the Jan and Hunter song from the office. The signals are separated by 1kHz because each song has frequency range from 0 to 3kHz. Considering that in the frequency spectrum each signal spans both the the negative and positive frequencies, the signal actually spans from -3kHz to 3kHz resulting in a 6kHz spread around the central frequency. This is one reason why each signals separation from the other ends up being about 1kHz. Separating the signals by 1kHz also allows for proper isolation between each of the amplitude modulated signals and removes the possibility for frequencies and basis signals to overlap in the frequency domain. The bandpass filter is used to separate each song based on their central frequency into 3 separate signals. The low pass filter is used to clean up the shifted signals so that each songs frequency spectra ranges from -3kHz to 3kHz.