EE2703: Assignment 8

Sagar, EE20B115

April 16, 2022

1 Aim

In this assignment we aim to analyse DFTs using Python's *numpy.fft* toolbox.

2 Procedure and Observations

2.1 Spectrum of $sin(5t)$

2.1.1 Code snippet:

```
x = linspace(0,2*pi,129)
x = x[:-1]

w = linspace(-64,64,129)
w=w[:-1]

y = sin(5*x)

Y = fftshift(fft(y))/128
```

2.1.2 Plots:

We plot the phase and magnitude of the DFT of $sin(5t)$:

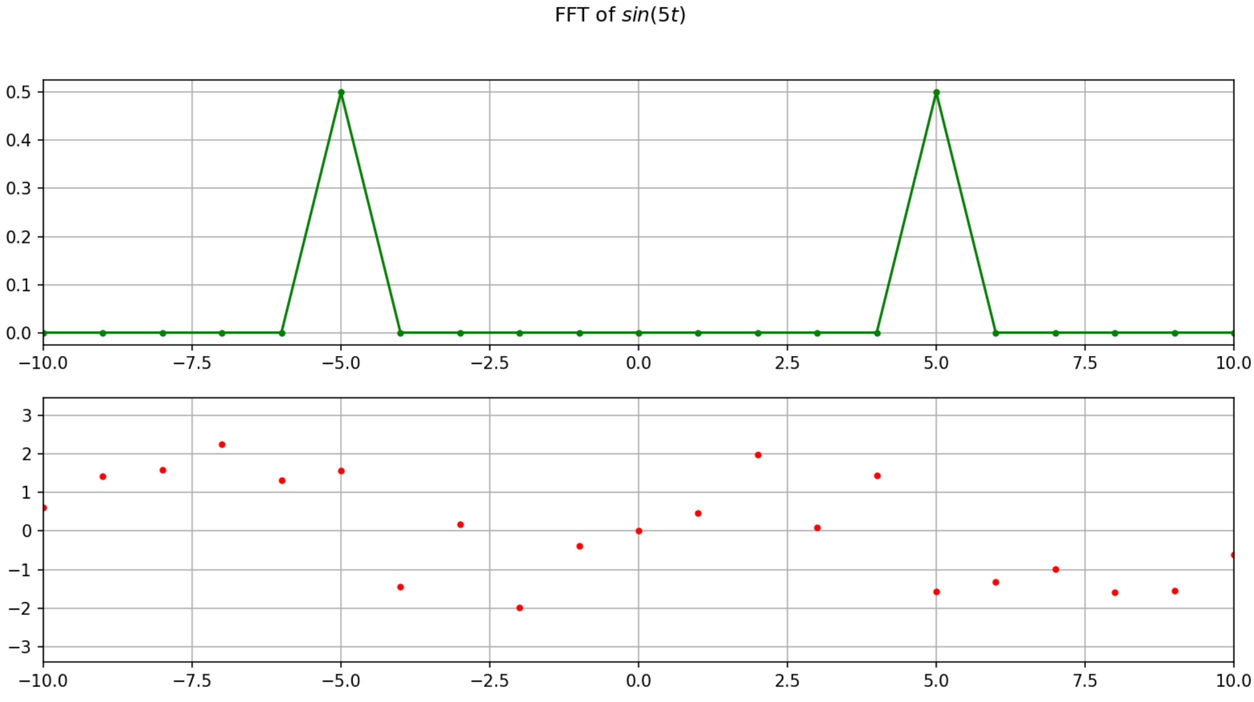

Figure 1: Spectrum of $sin(5t)$

2.1.3 Observations:

This is expected, because:

$$sin(5t) = \frac{1}{2j}(e^{5jt} - e^{-5jt}) \tag{1}$$

So, the frequencies present in the DFT of $sin(5t)$ are $\omega = \pm 5\ rad/sec$, and the phase associated with them is $\phi = \pm\frac{\pi}{2}\ rad/sec$ respectively. This is exactly what is shown in the above plot.

2.2 Amplitude Modulation with $(1 + 0.1cos(t))cos(10t)$

2.2.1 Code snippet:

```
x = linspace(0,8*pi,513)
x = x[:-1]

w = linspace(-64,64,513)
w=w[:-1]

y = (1+0.1*cos(x))*cos(10*x)

Y = fftshift(fft(y))/512
```

2.2.2 Plots:

Plotting the DFT using the *numpy.fft* package, we get:

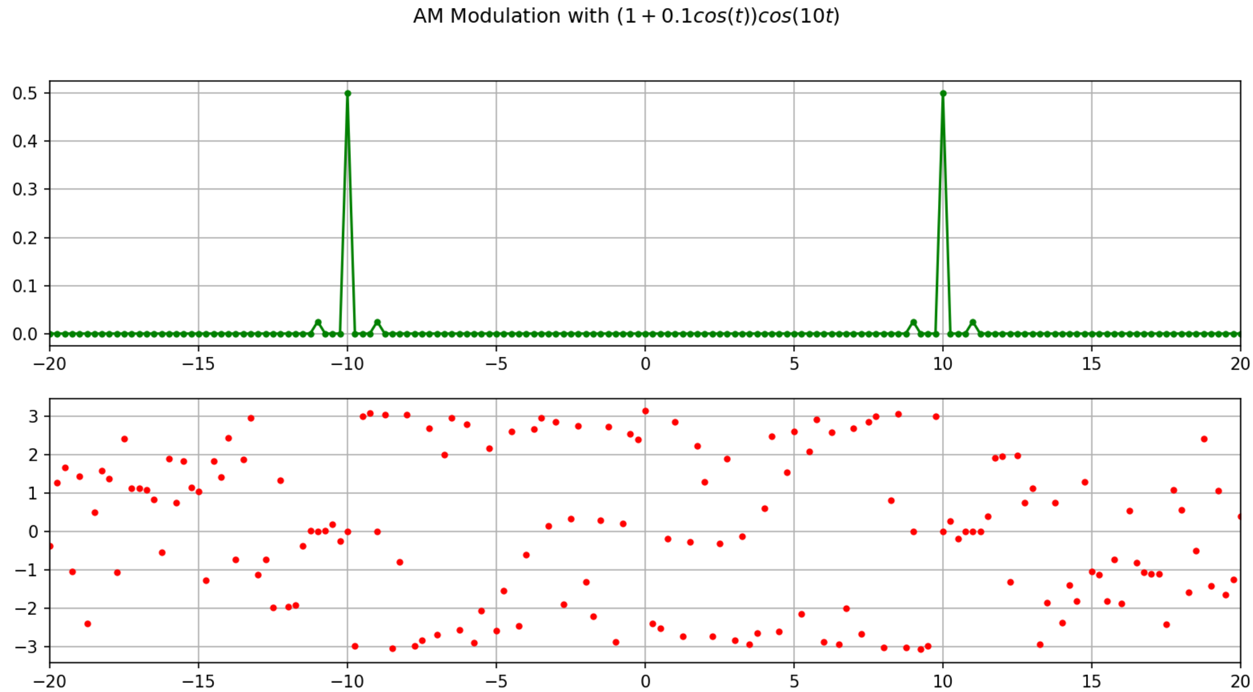

Figure 2: DFT of $(1 + 0.1cos(t))cos(10t)$

2.2.3 Observations:

We have,

$$(1 + 0.1cos(t))cos(10t) = \frac{1}{2}(e^{10jt} + e^{-10jt}) + 0.1 \cdot \frac{1}{2} \cdot \frac{1}{2}(e^{11jt} + e^{-11jt} + e^{9jt} + e^{-9jt}) \tag{2}$$

Writing $(1 + 0.1cos(t))cos(10t)$ in a different form, we observe that the frequencies present in the signal are $\omega = \pm 10\ rad/sec$, $\omega = \pm 11\ rad/sec$ and $\omega = \pm 9\ rad/sec$. Thus we expect the DFT also to have non-zero magnitudes only at these frequencies.

2.3 Spectra of $sin^3(t)$ and $cos^3(t)$

2.3.1 Code snippet:

```
x = linspace(0,2*pi,129)
x = x[:-1]

w = linspace(-64,64,129)
w=w[:-1]

y = (sin(x))**3

Y = fftshift(fft(y))/128

x = linspace(0,2*pi,129)

x = x[:-1]

w = linspace(-64,64,129)
w=w[:-1]
```

```
y = (cos(x))**3
Y = fftshift(fft(y))/128
```

2.3.2 Plots:

DFT Spectrum of $sin^3(t)$:

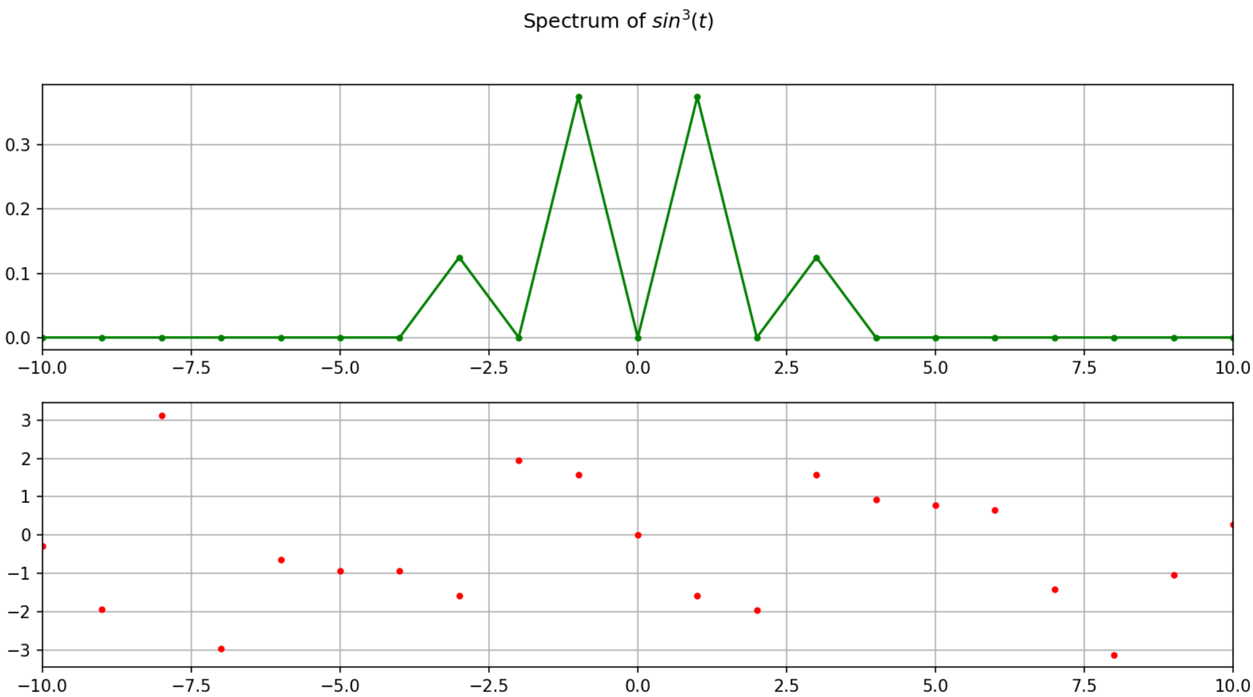

Figure 3: Spectrum of $sin^3(t)$

DFT Spectrum of $cos^3(t)$:

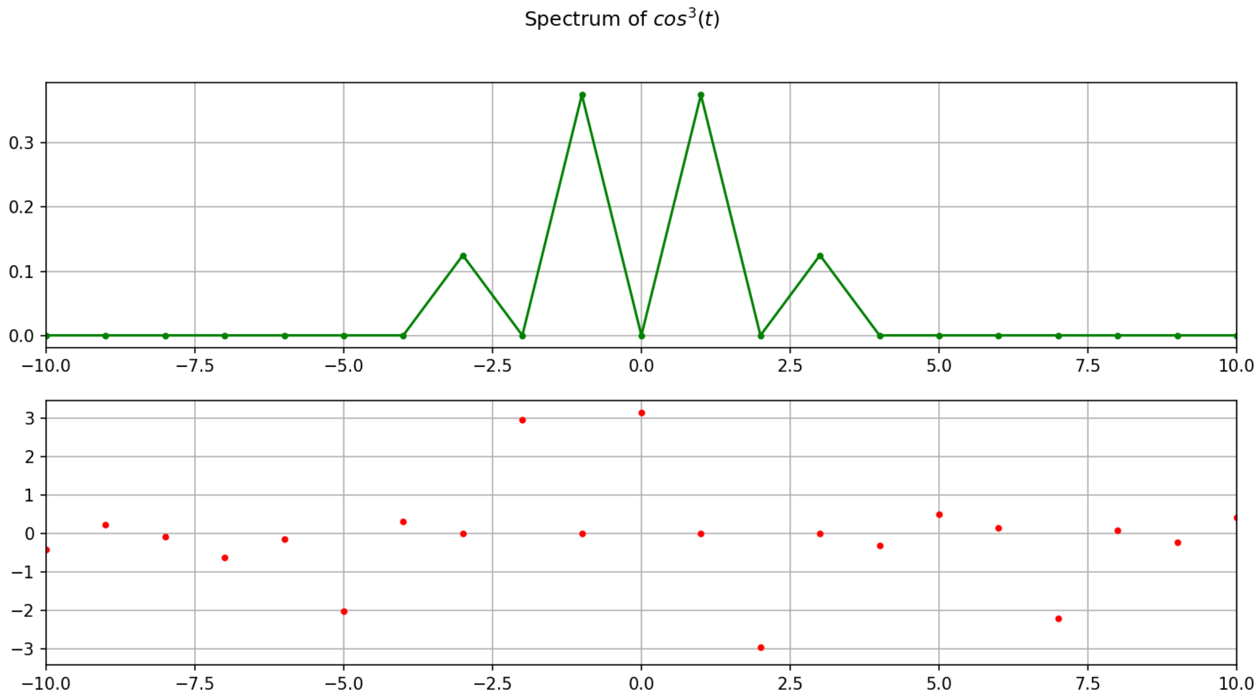

Figure 4: Spectrum of $cos^3(t)$

2.3.3 Observations:

The above 2 figures are expected because:

$$sin^3(t) = \frac{3}{4}sin(t) - \frac{1}{4}sin(3t) \tag{3}$$

$$cos^3(t) = \frac{3}{4}cos(t) + \frac{1}{4}cos(3t) \tag{4}$$

So, we expect peaks $\omega = \pm 1\ rad/sec$ and $\omega = \pm 3\ rad/sec$.

2.4 Frequency Modulation with $cos(20t + 5cos(t))$

2.4.1 Code snippet:

```
x = linspace(0,2*pi,129)
x = x[:-1]

w = linspace(-64,64,129)
w=w[:-1]

y = cos(20*x + 5*cos(x))

Y = fftshift(fft(y))/128

i = where(abs(Y)>1e-3)
subplot(2,1,2)
plot(w[i],angle(Y[i]),'r.')
```

2.4.2 Plots:

The DFT of $cos(20t + 5cos(t))$ can be seen below:

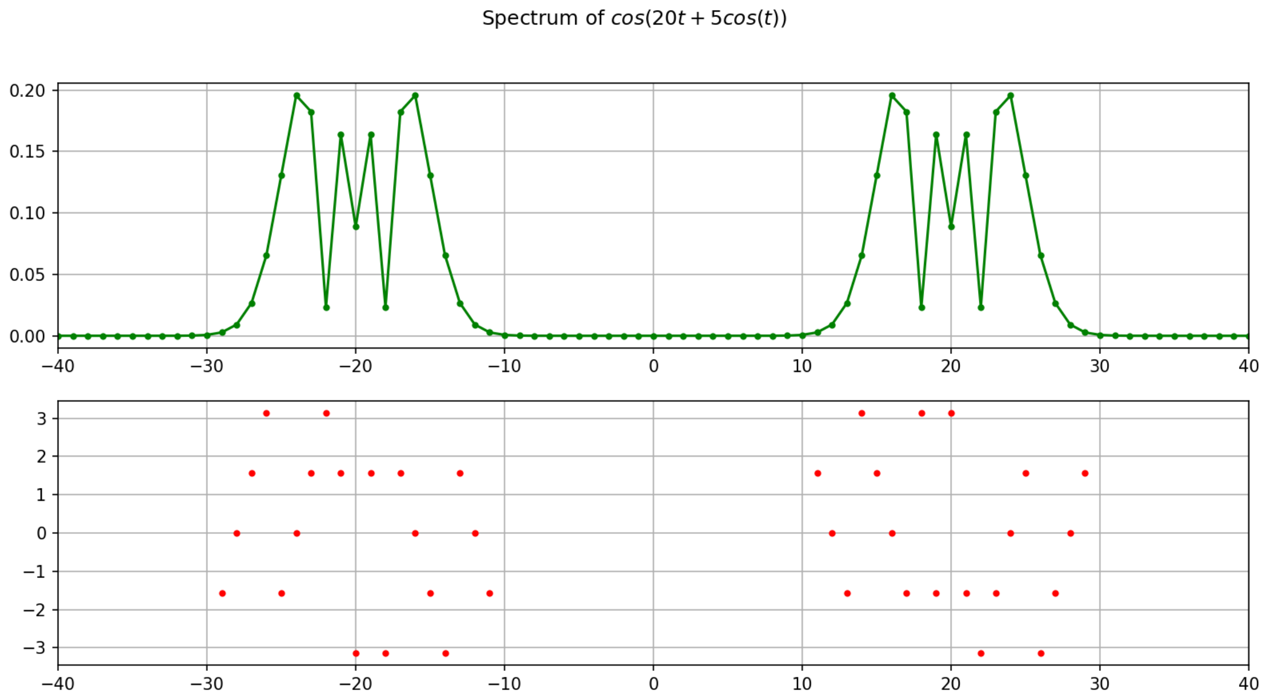

Figure 5: DFT of $cos(20t + 5cos(t))$

2.4.3 Observations:

When we compare this result with that of the Amplitude Modulation, we see that there are more side bands, and some of them have even higher energy than $\omega = \pm 20\ rad/sec$.

2.5 DFT of a Gaussian

2.5.1 Code snippet:

```
t =  np.linspace(-8*pi, 8*pi, 1025)
t = t[:-1]
xTrueGaussian = np.exp(-(t**2)/2)
Y = fftshift(fft(ifftshift(xTrueGaussian)))*8/1024.0

YMag = np.abs(Y)
YPhase = np.angle(Y)
absentFreqs = np.where(YMag < 1e-3)
YPhase[absentFreqs] = 0
w = np.linspace(-40, 40, 1025)
w = w[:-1]

trueY = np.exp(-(w**2)/2)/np.sqrt(2*pi)
trueYMag = np.abs(trueY)
trueYPhase = np.angle(trueY)

meanError = np.mean(np.abs(trueY - Y))
```

2.5.2 Plots:

The DFT of a gaussian is also a gaussian, as shown below:

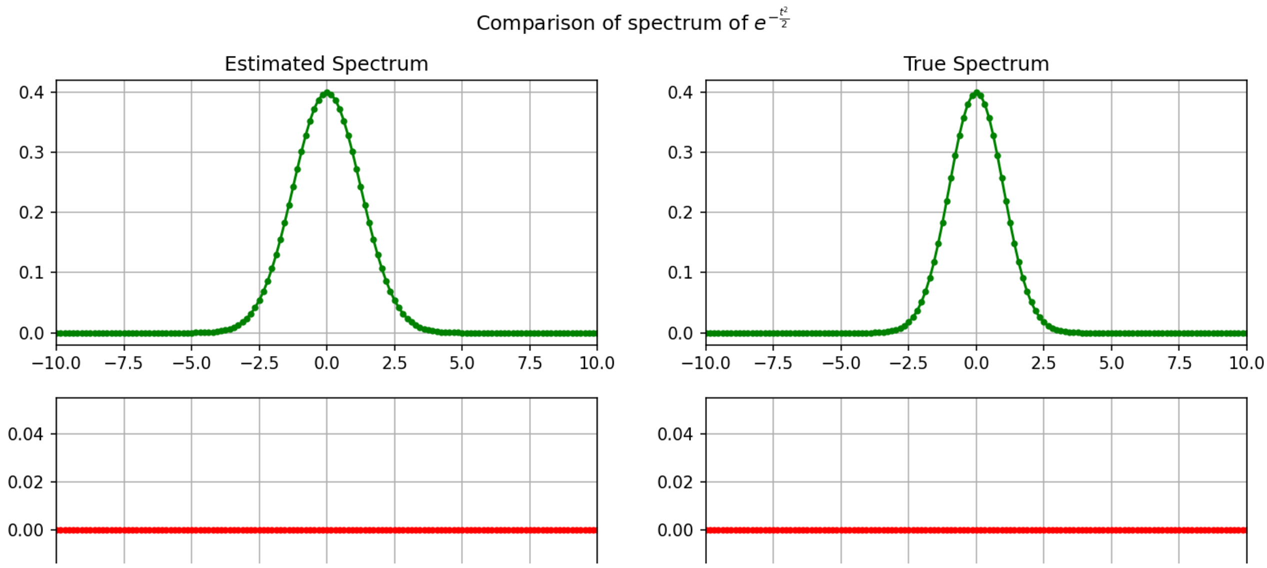

Figure 6: Gaussian Spectrum

2.5.3 Observations:

Magnitude of Mean error between computed and actual values of the Gaussian is: 0.004687500000000002 for a window from [-8π, 8π] and taking 512 points in that interval.

3 Conclusions

1. We have analysed the DFT's of various signals using the *numpy.fft* package.
2. We have used the *numpy.fft.fftshift()* and *numpy.fft.ifftshift()* methods to fix distortions in the phase response.