Python 10 assignment
Spectra of non-periodic signals

EE16B014
G.HARSHA KANAKA ESWAR

April 28, 2018

Abstract

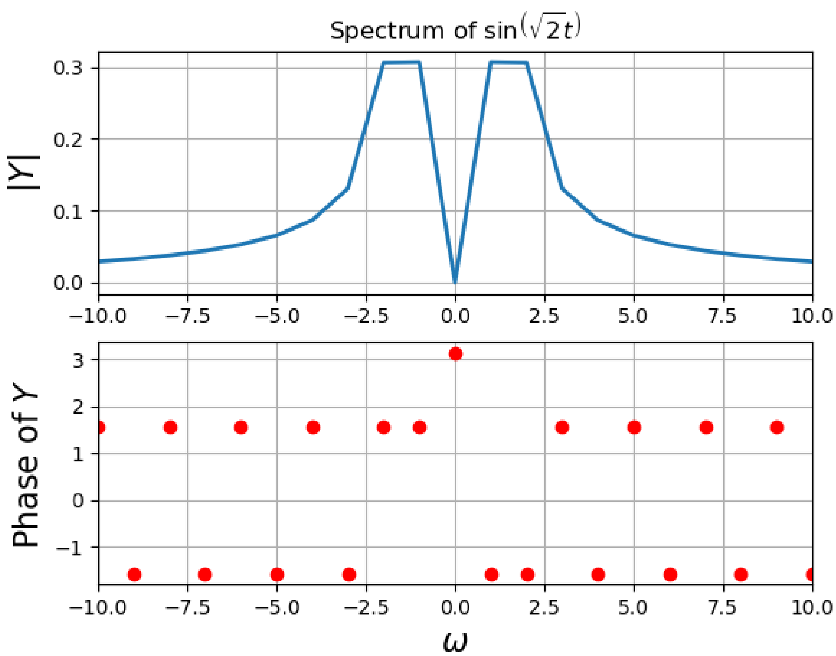
In the previous assignment we have found the spectra of periodic signals, in this report we are going to derive the spectra of non-periodic signals.We will also see the use of Hamming Window,for making the non-periodic signal continuous at periodic interval points.We will also derive the spectra of known signal and verify.

Examples:

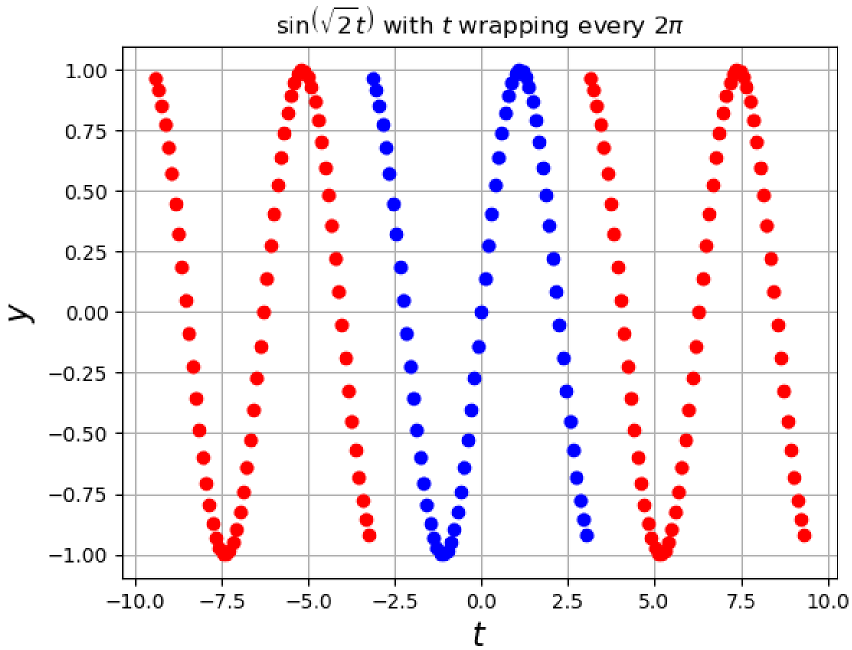
First we will see the general spectra of $sin\left(\sqrt{2t}\right)$,i.e without any window:

```
t=linspace(-pi,pi,65);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
y=sin(sqrt(2)*t)

y[0]=0 # the sample corresponding to -tmax should be set zeroo
y=fftshift(y) # make y start with y(t=0)
Y=fftshift(fft(y))/64.0
w=linspace(-pi*fmax,pi*fmax,65);w=w[:-1]

figure()
subplot(2,1,1)
plot(w,abs(Y),lw=2)
xlim([-10,10])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\sin\left(\sqrt{2}t\right)$")
```

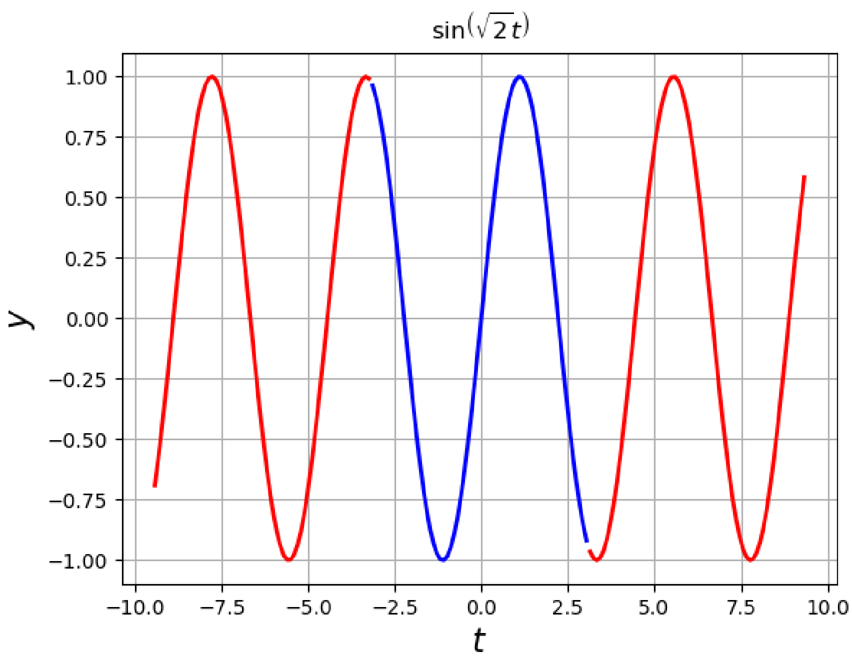
```
grid(True)

subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-10,10])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
savefig("fig10-1.png")
show()
```



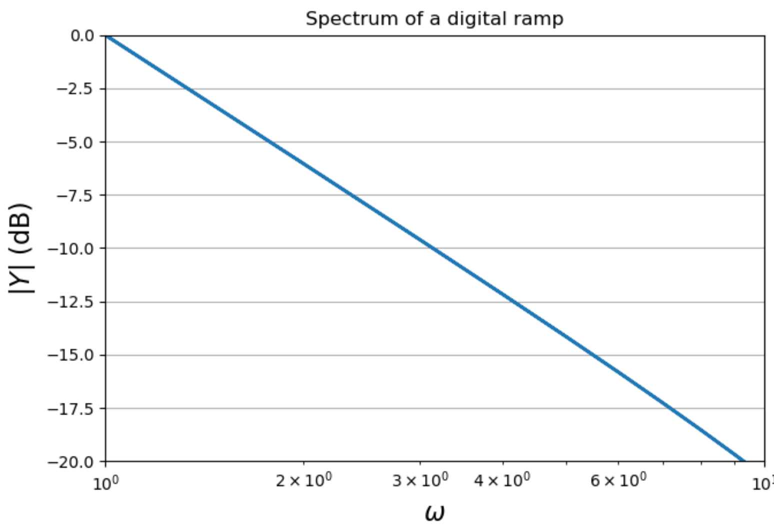
Here we got two peaks whose phase is correct but the decaying of magnitude is very less this is because of discontinuity occurring at repetition of every period ,which is causing a sudden jump:



Where the actual graph should be a continuous as shown below:



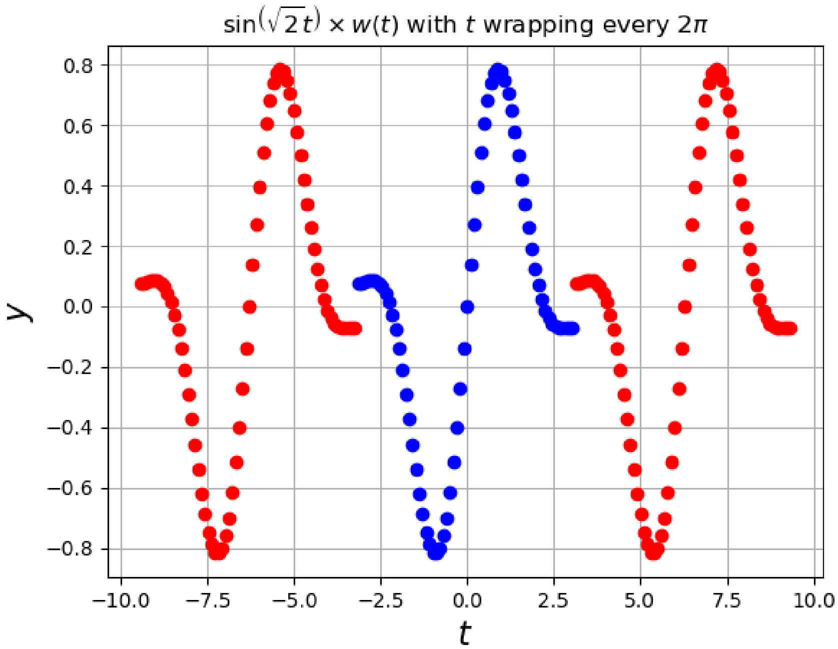
So we understood that there is a discontinuity which is not actually present,So in order to make it continuous at those points we will introduce window,whose Fourier series coefficients are of order $\frac{1}{\omega}$,Lets look at its Fourier coefficients in dB scale:



So now we will multiply the original function with the window to reduce the jump a the discontinues:

```
from pylab import*
t1=linspace(-pi,pi,65);t1=t1[:-1]
t2=linspace(-3*pi,-pi,65);t2=t2[:-1]
t3=linspace(pi,3*pi,65);t3=t3[:-1]
n=arange(64)
wnd=fftshift(0.54+0.46*cos(2*pi*n/63))
y=sin(sqrt(2)*t1)*wnd
figure(3)
plot(t1,y,'bo',lw=2)
plot(t2,y,'ro',lw=2)
plot(t3,y,'ro',lw=2)
ylabel(r"$y$",size=16)
xlabel(r"$t$",size=16)
title(r"$\sin\left(\sqrt{2}t\right)\times w(t)$ with $t$ wrapping ev
grid(True)
savefig("fig10-5.png")
show()
```

Now lets see how much the jump is reduced:

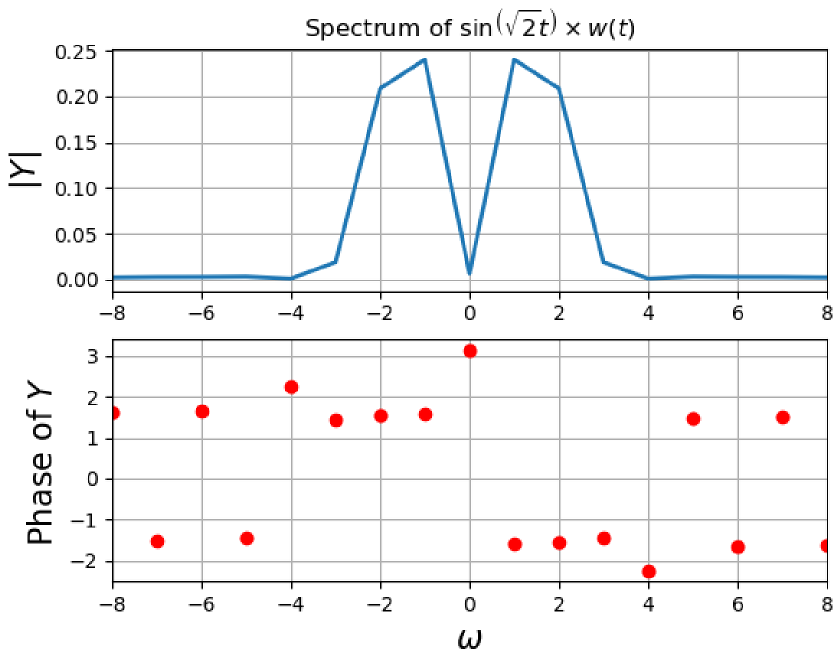


Thus the jump is reduced to maximum extent,so now lets get the fft spectra of the above graph:

```
from pylab import*
t=linspace(-pi,pi,65);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(64)
wnd=fftshift(0.54+0.46*cos(2*pi*n/63))
y=sin(sqrt(2)*t)*wnd
y[0]=0 # the sample corresponding to -tmax should be set zero
y=fftshift(y) # make y start with y(t=0)
Y=fftshift(fft(y))/64.0
w=linspace(-pi*fmax,pi*fmax,65);w=w[:-1]
figure()
subplot(2,1,1)
plot(w,abs(Y),lw=2)
xlim([-8,8])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\sin\left(\sqrt{2}t\right)\times w(t)$")
```

```
grid(True)
subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-8,8])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
savefig("fig10-6.png")
show()
```

The spectra after windowing:

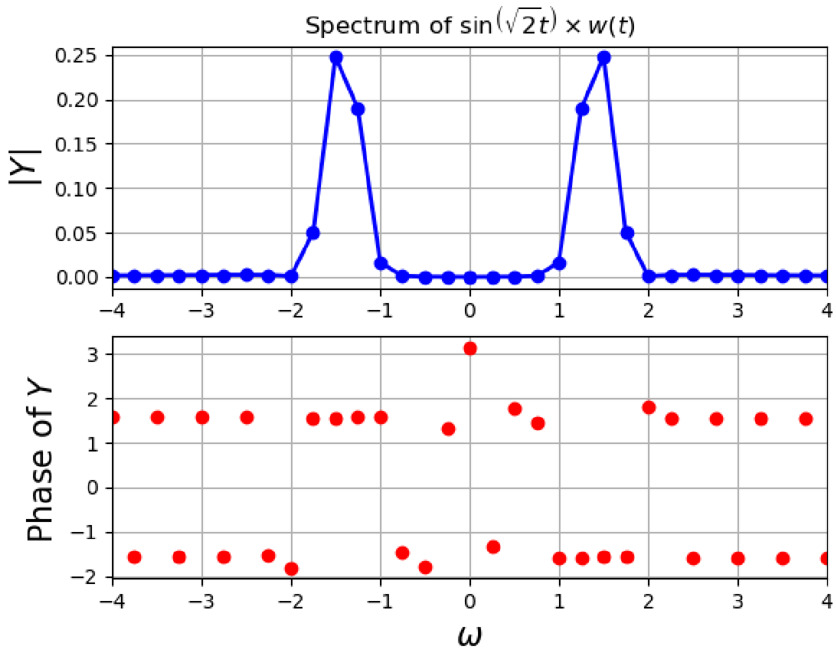


Now the decaying of the magnitude is improved greatly,but still the peak is like width of two samples,so lets increase the no.of samples:

```
from pylab import*
t=linspace(-4*pi,4*pi,257);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(256)
wnd=fftshift(0.54+0.46*cos(2*pi*n/256))
y=sin(sqrt(2)*t)*wnd
```

```
y[0]=0 # the sample corresponding to -tmax should be set zero
y=fftshift(y) # make y start with y(t=0)
Y=fftshift(fft(y))/256.0
w=linspace(-pi*fmax,pi*fmax,257);w=w[:-1]
figure()
subplot(2,1,1)
plot(w,abs(Y),lw=2)
xlim([-4,4])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\sin\left(\sqrt{2}t\right)\times w(t)$")
grid(True)
subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-4,4])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
savefig("fig10-6.png")
show()
```

So lets plot the spectra after increasing no.of samples:



Thus we can see the spectra improved far better than the spectra without windowing

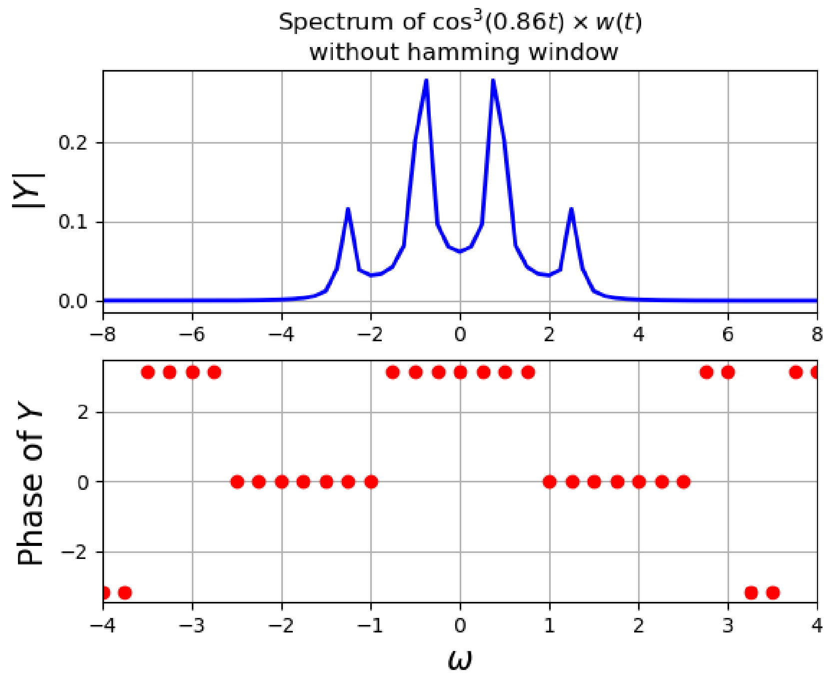
Spectra of $cos^3(\omega_0 t)$ with value of $\omega = 0.86$:

Now lets plot the spectra of $cos^3(\omega_0 t)$ with and without the hamming window and see the differences:

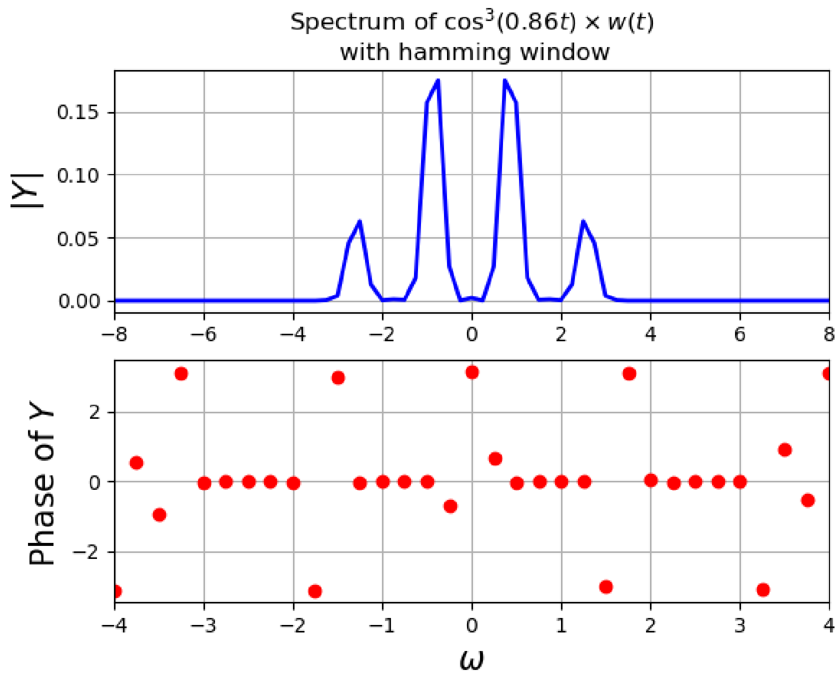
```
from pylab import*
t=linspace(-4*pi,4*pi,257);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(256)
wnd=fftshift(0.54+0.46*cos(2*pi*n/255))
y=((cos(0.86*t))**3)# y=sin(1.25*t)

y[0]=0
y1=fftshift(y)
Y1=fftshift(fft(y))/256.0

y=y*wnd
```

```
y=fftshift(y)
Y=fftshift(fft(y))/256.0
w=linspace(-pi*fmax,pi*fmax,257);w=w[:-1]
figure()

subplot(2,1,1)
plot(w,abs(Y),'b',lw=2)
xlim([-8,8])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\cos^3\left(0.86t\right)\times w(t)$""\n with h
grid(True)
subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-4,4])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)

figure()
subplot(2,1,1)
plot(w,abs(Y1),'b',lw=2)
xlim([-8,8])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\cos^3\left(0.86t\right)\times w(t)$""\n withou
grid(True)
subplot(2,1,2)
plot(w,angle(Y1),'ro',lw=2)
xlim([-4,4])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
savefig("fig10-7.png")
show()
```

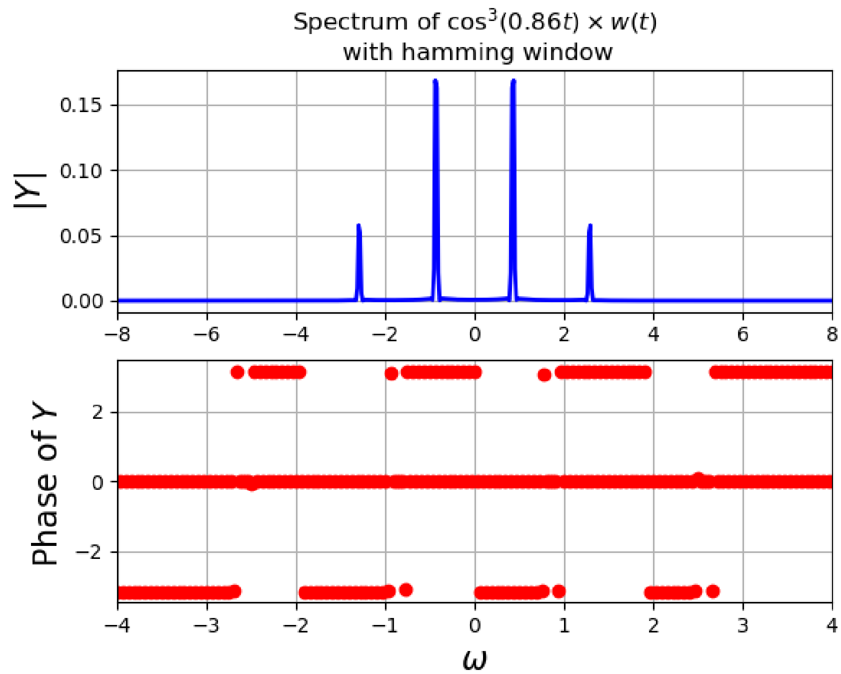
Now lets see the spectra of $cos^3(\omega_0 t)$without hamming window:



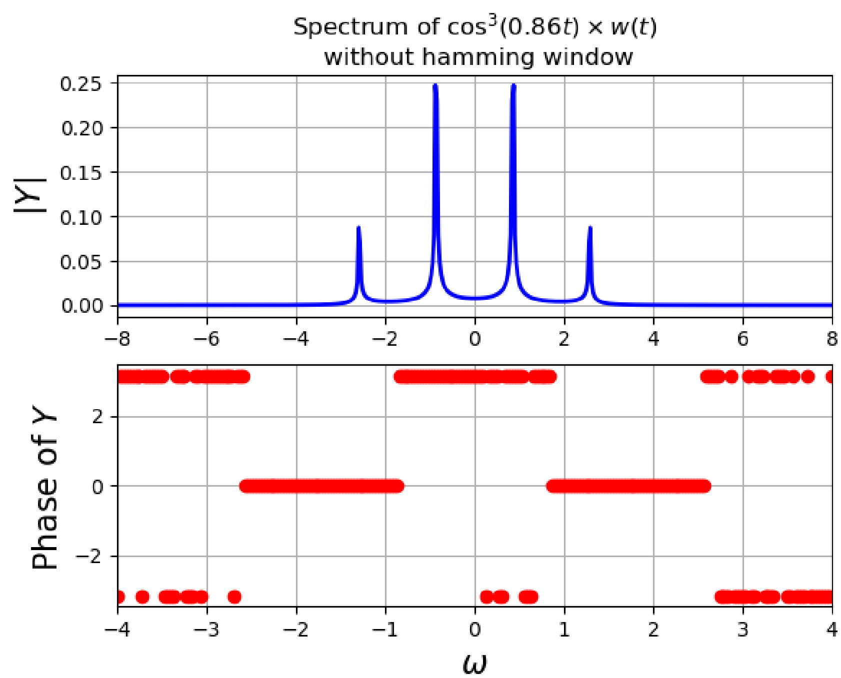
We can see that there is still no zero values existing after the peaks has occurred,Now lets see the spectra with Hamming Window:



Here the values after peaks is going to zero,Lets see the same spectra with Hamming Window and more no.of samples:



Without Window more no.of samples:



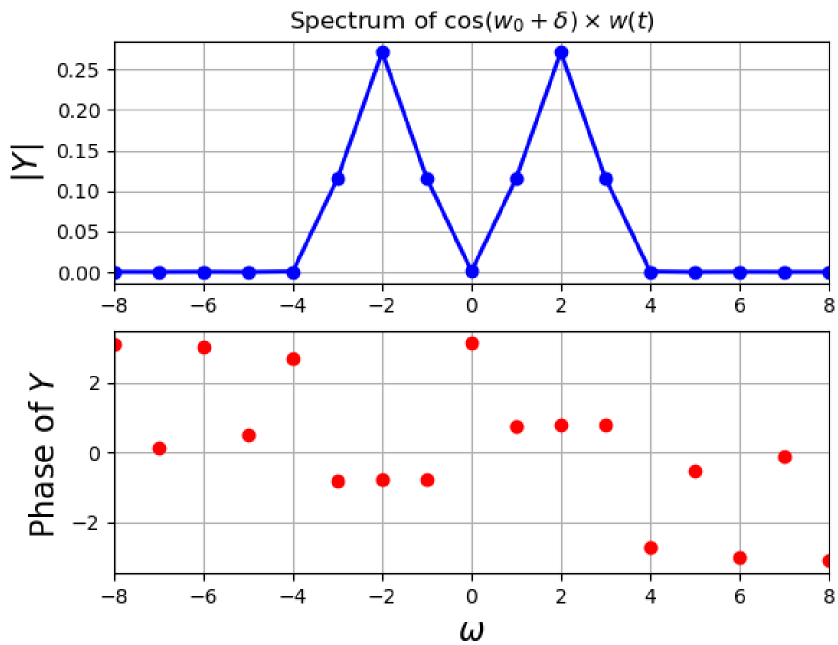
By increasing samples the spectra's has improved,But we can say that the spectra with Hamming window is more accurate.

Finding ω_0, δ of $cos(\omega_0 + \delta)$ using the given samples:

Now we will have 128 samples of $cos(\omega_0 + \delta)$ the from which we calculate the values of ω_0, δ ,From the spectra created from the given samples.

```
from pylab import*
t=linspace(-pi,pi,129);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(128)
wnd=fftshift(0.54+0.46*cos(2*pi*n/127))
w0=2
d=pi/4
y=cos(w0*t+d)
y=y*wnd
y[0]=0
y=fftshift(y)
Y=fftshift(fft(y))/128.0
w=linspace(-pi*fmax,pi*fmax,129);w=w[:-1]
figure()
subplot(2,1,1)
plot(w,abs(Y),'b',w,abs(Y),'bo',lw=2)
xlim([-8,8])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\cos^3(w_0+\delta)\times w(t)$")
grid(True)
subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-8,8])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
savefig("fig10-7.png")
show()
```

Here we have take the samples of function whose $\omega_0 = 2, \delta = \frac{\pi}{4}$,Now lets plot the spectra from the samples and see whether we can get those values from spectra:

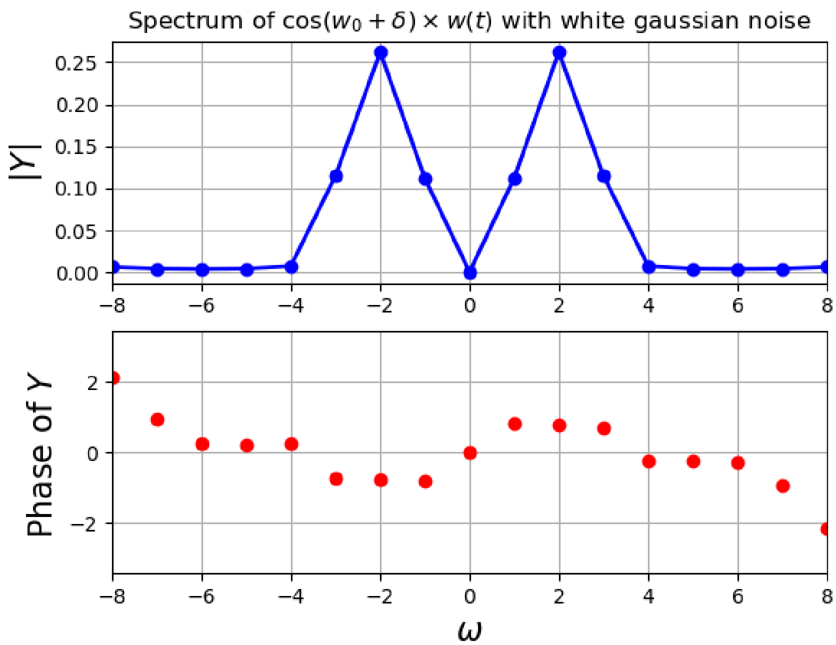


We can get the value of .ω_0 by seeing the point of peak occurring and the value of δ can be found out by looking at the phase at the peak lying in positive x-axis.(which give the value of δ along with sign)

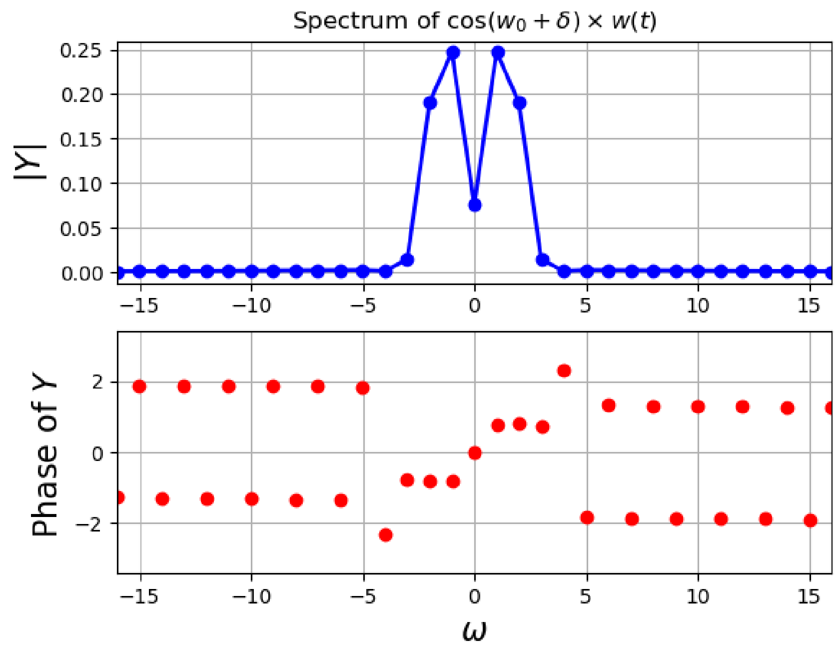
Now lets add some "white Gaussian noise" and see the spectra and get the values of ω_0, δ.

```
y=cos(w0*t+d)+0.1*randn(128)
y=y*wnd
```

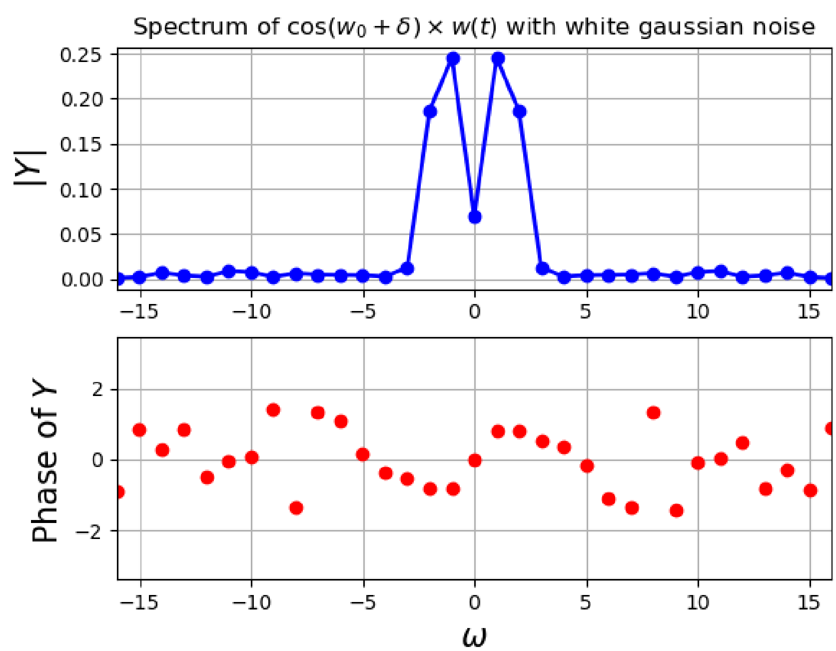
Lets see the spectra:



For integer values of ω_0 the spectra is looking good.
Lets take a look at the graphs when $\omega_0 = 1.34$ non-interger:
Without “white Gaussian noise”:



With “white Gaussian noise”:



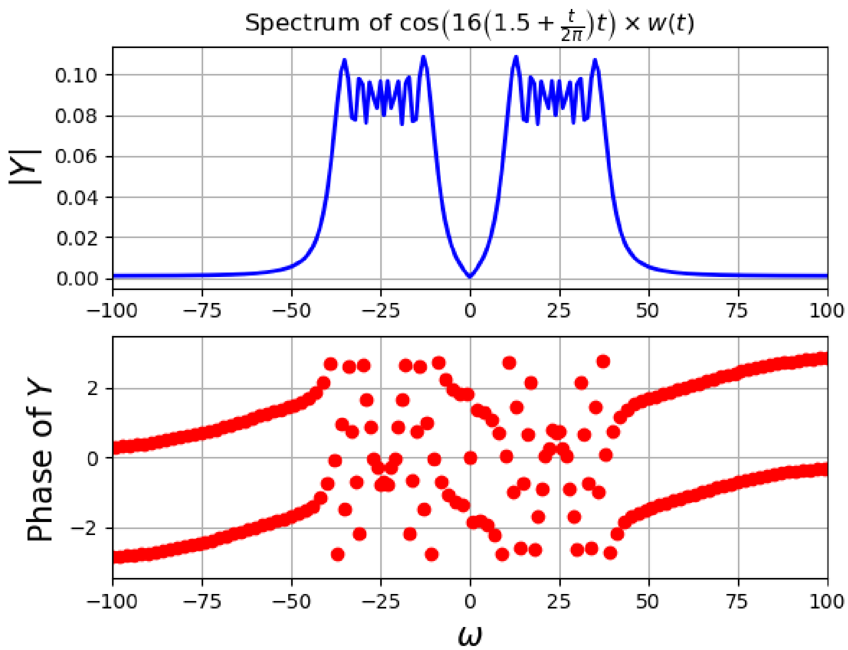
Here also we can see the peaks ,From which we can get the values of ω_0, δ but some slight distortion,unlike in case of integer values of ω_0 more accurate.(Because of the no of samples are fixed to 128)

DFT of $cos\left(16\left(1.5+\frac{t}{2\pi}\right)t\right)$:

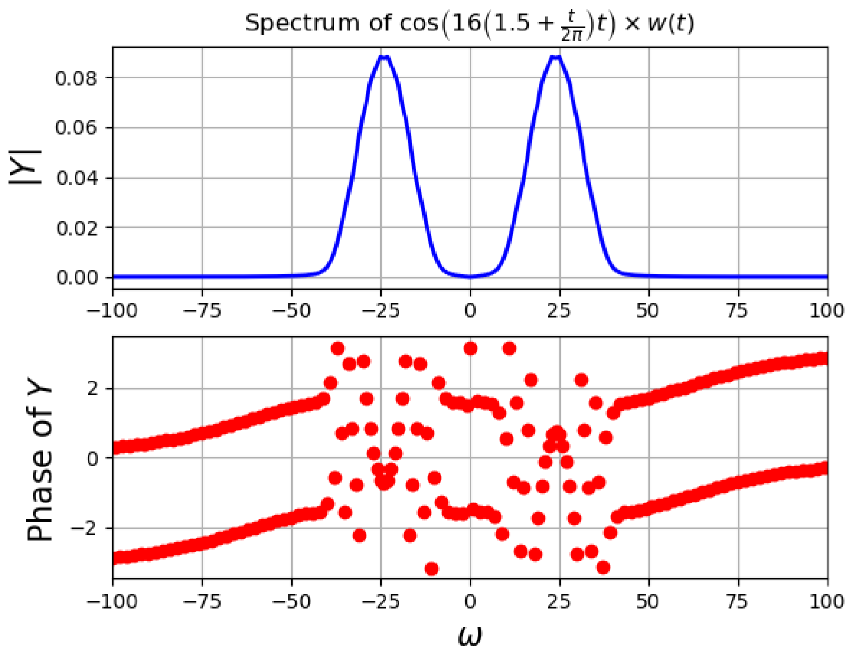
This function is also known as the chirped signal,whose frequency continuously changes from 16 to 32 rads per second.
Lets get the spectra of this chirped signal with and without window:

```
from pylab import*
t=linspace(-pi,pi,1025);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(1024)
wnd=fftshift(0.54+0.46*cos(2*pi*n/1024))

y=cos(16*(1.5+t/(2*pi))*t)# y=sin(1.25*t)
#y=y*wnd
y[0]=0
y=fftshift(y)
Y=fftshift(fft(y))/1024.0
w=linspace(-pi*fmax,pi*fmax,1025);w=w[:-1]
figure()
subplot(2,1,1)
plot(w,abs(Y),'b',lw=2)
xlim([-100,100])
ylabel(r"$|Y|$",size=16)
title(r"Spectrum of $\cos\left(16\left(1.5+\frac{t}{2\pi}\right)t\ri
grid(True)
subplot(2,1,2)
plot(w,angle(Y),'ro',lw=2)
xlim([-100,100])
ylabel(r"Phase of $Y$",size=16)
xlabel(r"$\omega$",size=16)
grid(True)
savefig("fig10-7.png")
show()
```

Without Window:



We are now seeing many peaks are occurring side by side at a frequency,Now lets see what happens if we apply window:



They all merged into single peak of amplitude=sum of amplitudes of the different peaks,at that frequency.

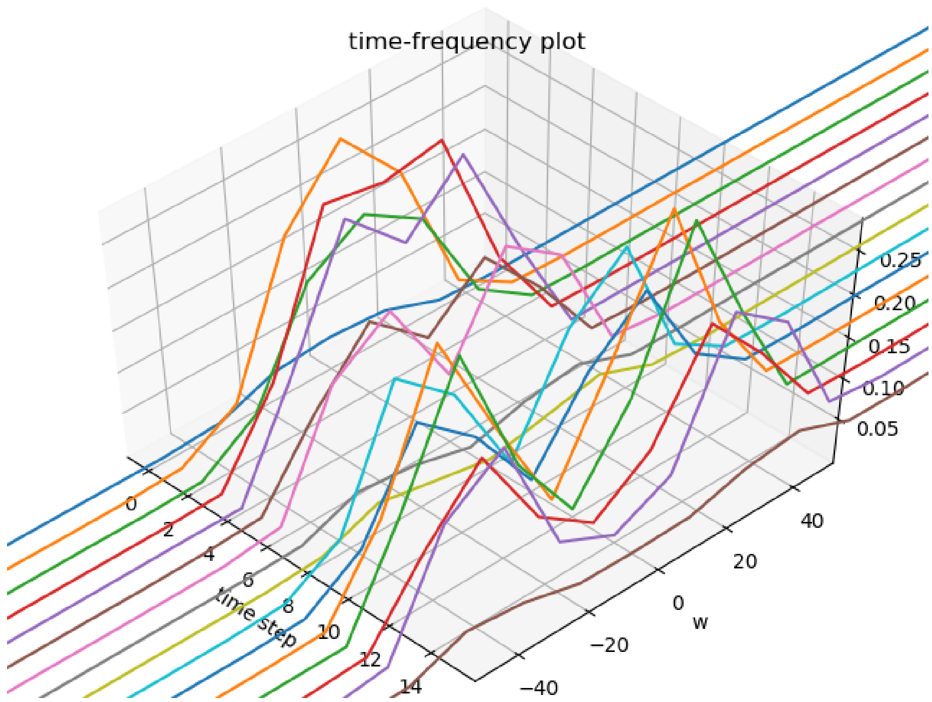
3D surface plot of different samples of $cos\left(16\left(1.5+\frac{t}{2\pi}\right)t\right)$:

Lets see how the surface plot looks like,when the 1024 samples of the function are taken into an array whose columns contain 64 samples ,filled one after the other by the total samples of 1024 forming 16 columns,And then we will draw the surface 3D plot.

```
from pylab import*
from scipy import *
import mpl_toolkits.mplot3d.axes3d as p3
t=linspace(-pi,pi,1025);t=t[:-1]
dt=t[1]-t[0];fmax=1/dt
n=arange(64)
wnd=fftshift(0.54+0.46*cos(2*pi*n/63))
A=zeros((64,16))
t2=arange(16)
```

```
y=cos(16*(1.5+t/(2*pi))*t)
print(y)
#y=y*wnd
#A[:,t1]=y[(t1-1)*64:t1*64]
for t1 in t2:
        A[:,t1]=fftshift(fft(fftshift(y[t1*64:(t1+1)*64]*wnd)))/64.0
print(A)
w=linspace(-pi*fmax,pi*fmax,65);w=w[:-1]
x=arange(0,16,1)
y=arange(64)
X,Y=meshgrid(x,y)

fig1=figure()
ax=p3.Axes3D(fig1)
title("time-frequency plot")

for i in range(16):
        ax.plot3D(X[:,i],w,abs(A[:,i]))
ylim(-50,50)
xlabel("time step")
ylabel("w")
show()
```

Lets see the surface 3D plot:



From this graph we can clearly observe the different peaks at different timesteps,which are nothing but the peaks occurred in the spectra of $cos\left(16\left(1.5+\frac{t}{2\pi}\right)t\right)$ without application of window.[i.e by application of window all peaks are merging to give rise to a new single peak]