



UNIVERSIDAD DE SONORA

DIVISIÓN DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE FÍSICA

RECONSTRUYENDO LA SEÑAL

SOFÍA GONZÁLEZ MONTOYA

PROFESOR: CARLOS LIZÁRRAGA CELAYA

28 DE ABRIL DE 2017

1. RESUMEN

En la actividad anterior pudimos descomponer una señal de las amplitudes de las mareas, en un intervalo de tiempo dado, en este caso de dos meses. Con la transformada discreta de Fourier pudimos extraer los modos naturales de la marea en los sitios de estudio y obtener las amplitudes de las frecuencias naturales de la marea. Con estos resultados reconstruimos la señal de la marea con los valores de la transformada de Fourier.

2. INTRODUCCIÓN

En esta práctica se pretende reconstruir las gráficas obtenidas en la actividad 5 del curso, las cuales son obtenidas de los datos descargados de NOAA y CICESE de las locaciones escogidas, esto con ayuda de la actividad 6 en la cual descomponemos las ondas de la marea en sus principales modos y los identificamos.

Para esto hacemos uso de la transformada discreta de Fourier, y los modos encontrados. Esta actividad es realizada para confirmar si el análisis armónico realizado es el correcto, y ver el error relativo entre la gráfica proveniente de los sitios oficiales y la realizada en clase.

3. RECONSTRUYENDO LA SEÑAL

3.1. PROCEDIMIENTO

Con ayuda de la actividad anterior, habiendo encontrado los modos principales con sus respectivas amplitudes y periodos, podemos iniciar con la práctica 7 y aplicar la transformada de Fourier discreta.

Para el sitio de La Paz, BCS se requirió juntar las columnas de año, mes, día y hora en una sola, de la siguiente manera:

```
from datetime import datetime
df['date']= df.apply(lambda x:datetime.strptime("{0} {1} {2} {3}".
format(x[u'anio'],x[u'mes'], x[u'dia'], x[u'hora(utc)']), "%Y %m %d %H"),axis=1)
```

Para comenzar con la reconstrucción de la altura tenemos el siguiente código

```
z = np.arange(0.0, 1440.0, 1.0)
df['T'] = pd.Series(z, index =None)
```

Para la transformada discreta de Fourier, que fue lo calculado en la actividad 6

```
N = 1440
T = 1.0
```

```

#aplicacion de la transformada de Fourier
y = df["altura(mm)"]
yf = fft(y)
xf = fftfreq(N, T)
xf = fftshift(xf)
yplot = fftshift(yf)

graf = plt.plot(xf, 2.0/N *abs(yplot))
plt.xlim(-0.01,0.1)
plt.grid(True)

plt.xlabel('Frecuencia [1/hora]')
plt.ylabel('Amplitud [m]')
plt.title('La Paz, BCS')

fig = plt.gcf()
fig.set_size_inches(9, 5)
plt.show()

```

Y para encontrar las amplitudes de los modos encontrados de igual manera en la actividad 6 para el caso de La Paz:

```

a=2*np.absolute(yf)/N
print(np.where(a[:,] > 100))

```

```
b=a[a[:,] > 100]
```

```
b
```

Con esto, sólo definimos las amplitudes, frecuencias y periodos de cada modo con el siguiente código para La Paz

```
#Amplitudes notorias, fueron ubicadas en la actividad 6
```

```
A0_s = np.absolute(yf[int(0),]/N)
```

```
A3_s= 2.0*np.absolute(yf[int(56),]/N)
```

```
A6_s = 2.0*np.absolute(yf[int(60),]/N)
```

```
A9_s = 2.0*np.absolute(yf[int(116),]/N)
```

```
A10_s = 2.0*np.absolute(yf[int(120),]/N)
```

```
#PERIODOS
```

```
f_A3s = xf[int(720 +56),]
```

```
f_A6s = xf[int(720 +60),]
```

```
f_A9s = xf[int(720 +116),]
```

```
f_A10s = xf[int(720 +120),]
```

```
#FASES
```

```
qA0s = np.angle(yf[int(0),])
```

```
qA3s = np.angle(yf[int(56),])
```

```
qA6s = np.angle(yf[int(60),])
```

```
qA9s = np.angle(yf[int(116),])
```

```
qA10s = np.angle(yf[int(120),])
```

Después crear la gráfica con estas mismas amplitudes y periodos

```
y= df['altura(mm)']/1000
```

```
w= 2.0*np.pi
```

```
a=0
```

```
def f(t):
```

```
    return A0_s + (A3_s*np.cos(w*f_A3s*t+qA3s) + A6_s*np.cos(w*f_A6s*t+qA6s)
    + A9_s*np.cos(w*f_A9s*t+qA9s) + A10_s*np.cos(w*f_A10s*t+qA10s))
```

```
plt.plot(df['date'], df[u'altura(mm)'], 'b-', label="Altura")
```

```
plt.plot(df['date'], f(df['T']), 'g-', label='Altura reconstruida')
```

```
plt.xlim(pd.Timestamp("2016-01-01 0:00"), pd.Timestamp('2016-01-31 23:00'))
```

```
plt.ylabel('Altura de marea [m]', fontsize = 13)
```

```
plt.xlabel('Día', fontsize = 13)
```

```
plt.title('Reconstrucción de señal en La Paz, BCS durante Enero y Febrero 2017', f
```

```
plt.grid(True)
```

```
fig = plt.gcf()
```

```
fig.set_size_inches(12, 7)
```

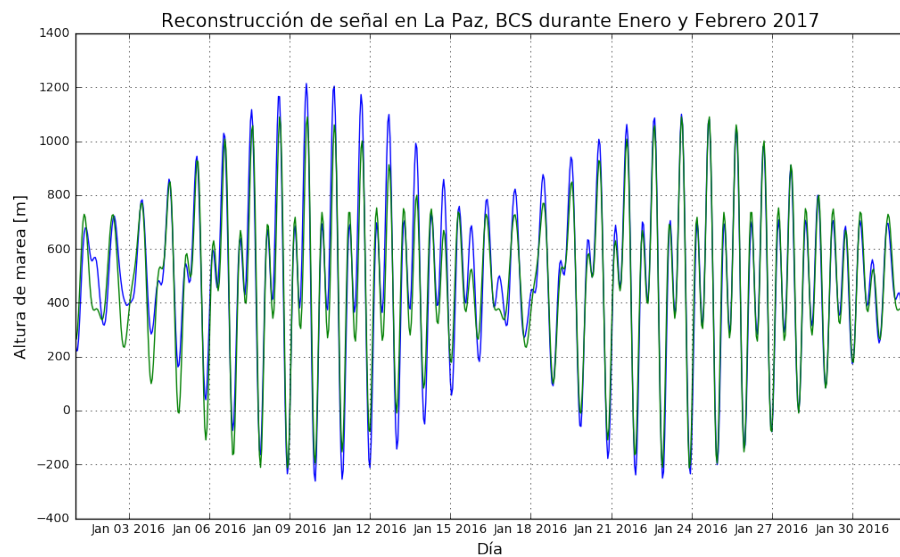
```
plt.show()
```

Para la locación en Estados Unidos, procedemos de la misma manera, cambiando por los datos de Montauk, NY para este caso.

3.2. RESULTADOS

Las gráficas encontradas con ayuda de la transformada de Fourier son:

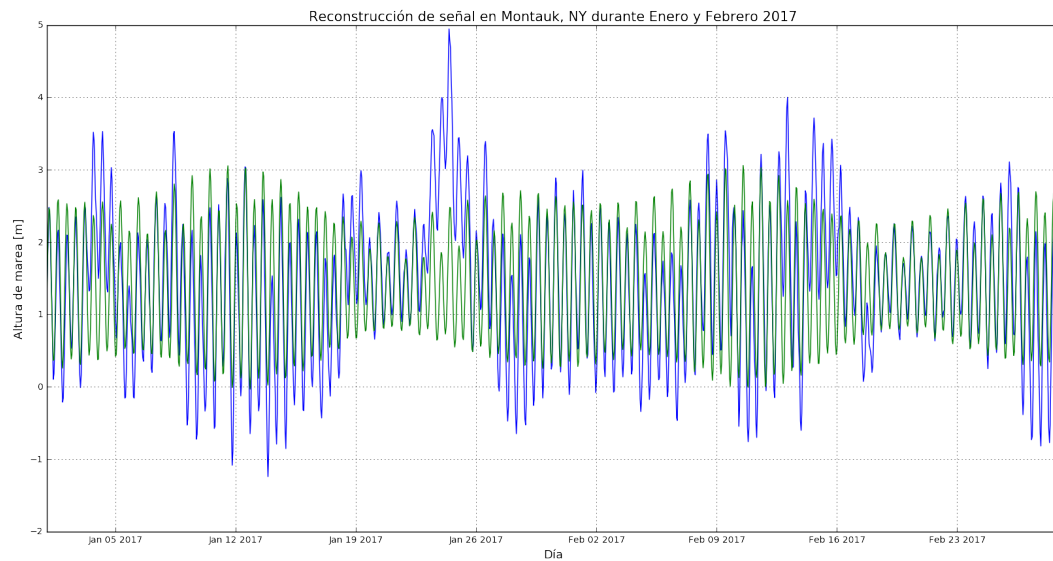
Para La Paz, BCS.



Con un error relativo del 2.23 %, obtenido con el siguiente comando:

```
y0=df['altura(mm)']  
y1=f(df['T'])  
  
#ERROR RELATIVO  
print()  
sum(abs(y0-y1)**2) / sum(abs(y0)**2)
```

Para Montauk, NY.



Con un error relativo de 12.64% obtenido con el mismo comando:

```
y0=df['Water Level']  
y1=f(df['T'])}  
  
#ERROR RELATIVO  
print()  
sum(abs(y0-y1)**2) / sum(abs(y0)**2)
```


4. Referencias

- Theory of Tides. https://en.wikipedia.org/wiki/Theory_of_tides
- Harmonic Analysis. https://en.wikipedia.org/wiki/Harmonic_analysis