Universidad de Sonora

División de Ciencias Exactas y Naturales

Física computacional

14 de marzo del 2017

Act 6: Análisis Armónico de Mareas

Alumna: Chávez Gutiérrez Yanneth Tzitzin
Profesor: Carlos Lizárraga Celaya.

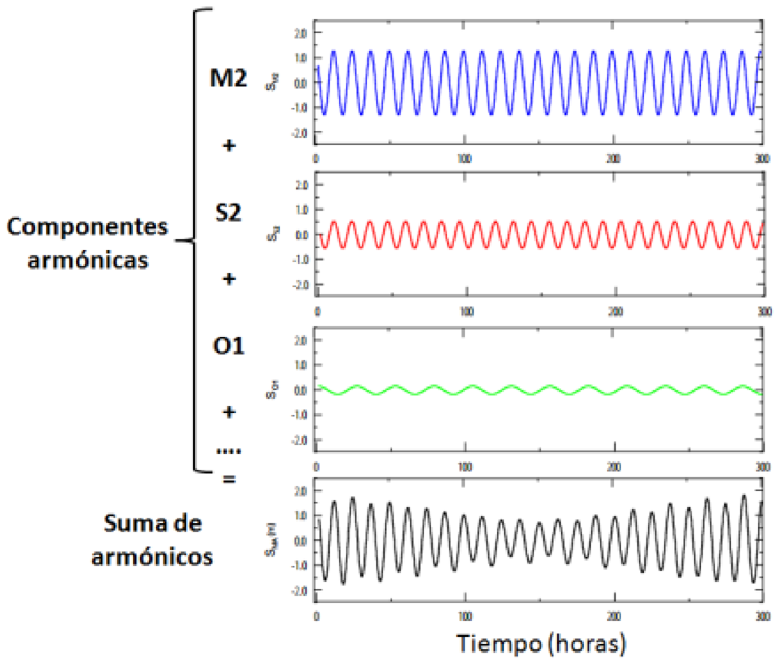
1. Introducción

Este reporte será una continuación del reporte anterior "Mareas y corrientes". Teniendo los datos de los sitios, se tratá de encontrar regularidades ahora en los niveles de la marea.

La teoria de las mareas, es una rama que trata de interpretar y predecir deformaciones en las mareas debido a la atracción gravitacional de los cuerpos celestes, en especial el que más hace notorio los cambios: el de la luna.

Para hacer ese tipo de analisis y predicciones en datos, se aplica lo que llamamos el analisis de Fourier que proporciona un analisis *armónico* el cuál se encarga de representar las funciones o señales como superposición de ondas expresadas o aproximadas por sumas de funciones trigonométricas simples.

En esta práctica se trabajará con un rango de valores donde el nivel del mar se haga cero en varias ocaciones, con ello se definirá un nuevo data frame y a estos datos podremos aplicar la Transformada Discreta de Fourier desarrollada en Python para transformar los datos de la marea. Y se obtendrán las principales componentes de Fourier de las mareas de los datos para cada sitio y las gráficas dadas.



Los datos con los que se seguira trabajando son los siguientes:

NOAA: *Bermuda, St. Georges Island* ***CICESE***: *Ciudad del carmen, Campeche*

2. Resultados

Para realizar esta práctica, se selecciono un rango de valores de los datos, de las ocaciones en las que el nivel del mar se hacia cero. Y se define un nuevo data frame con estos datos. Y se aplica la transformada de Fourier discreta desarrollada en Python para transformar los datos de marea.

CICESE: *Ciudad del carmen, Campeche*

Se utilizaron los datos del mes de enero del 2016. Se importaron los datos a la biblioteca de Python, y se le dio formato de "Date Time" a las columnas del archivo. Con el siguiente código:

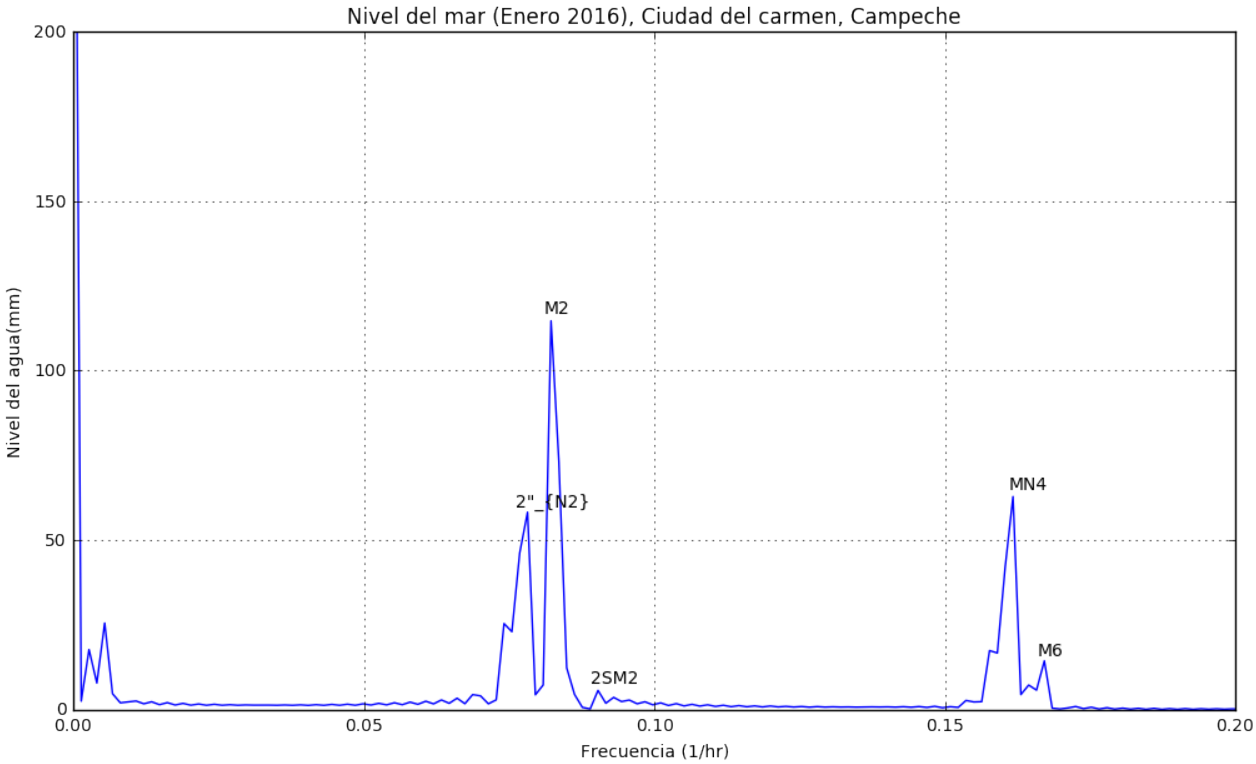
```
from datetime import datetime
df['Date Time']= df.apply(lambda x:datetime.strptime("{0} {1} {2} {3}".
format(x[u'año'],x[u'mes'], x[u'dia'], x[u'hora']), "%Y %m %d %H"),axis=1)
```

El cual indica que se seleccionan las columnas 'año' 'mes' 'dia' y 'hora' para juntas dar el formato necesario para graficar. Y a estos datos se les aplico la transformada de Fourier:

```
import numpy as np
import matplotlib.pyplot as mplt
from scipy.fftpack import rfft
fig = mplt.gcf()
fig.set_size_inches(12, 7)

#Transformada
N=744
T=1.0
x=df_enero[u'hour']
y=df_enero[u'Altura']
yf=rfft(y)
xf=np.linspace(0.0,1.0/(2.0*T),N/2)
import matplotlib.pyplot as plt
plt.plot(xf,2.0/N* np.abs(yf[0:N/2]))
plt.xlim(0,0.2)
plt.ylim(0,200)
mplt.ylabel('Nivel del agua(mm)')
mplt.xlabel('Frecuencia (1/hr)')
mplt.title('Nivel del mar (Enero 2016), Ciudad del carmen, Campeche')
plt.grid()
#plt.text(0.005,28,'M2')
```

```
plt.text(0.076,60,'2"_{N2}')
plt.text(0.081,117,'M2')
plt.text(0.089,8,'2SM2')
plt.text(0.161,65,'MN4')
plt.text(0.166,16,'M6')
plt.show()
```



Para identificar los modos al haber aplicado la transformada de Fourier, se hizo uso del periodo introducido en la transformada. $T = 1$ en este caso. Al calcular el periodo que coorresponde al modo que buscamos, se utiliza la siguiente ecuación que involucra a la frecuencia obtenida.

$$T' = (frecuencia)^{-1}$$

Cuadro 1: Tabla de datos Ciudad del Carmen

Pico (frecuencia)	Periodo	Amplitud	T
0.076	13.1579	60	2"$_{N2}$
0.081	12.3456	117	M_2
0.089	11.2359	8	$2SM_2$
0.161	6.2111	65	MN_4
0.166	6.0241	16	M_6

NOAA: *Bermuda, St. Georges Island*

De la agencia $NOAA$ se descargaron los datos de todo el mes de enero y febrero del 2017. Con estos datos se trabajó.

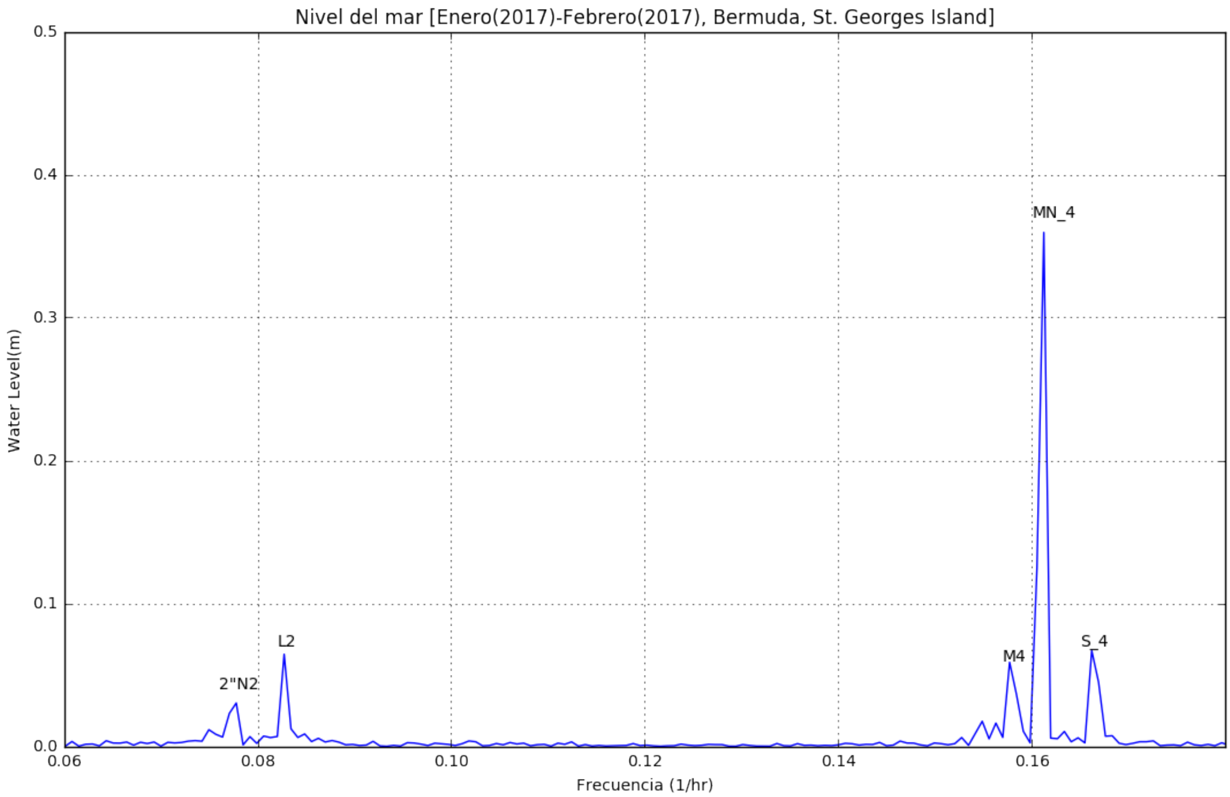
De manera similar a los datos de $CICESE$ se aplico para los datos de $NOAA$ en estados unidos, ahí ya nos proporcionaba los datos con formato *datetime* y no hubo necesidad de aplicar el formato. Solo se aplicó la transformada como a continuación se muestra:

```
import matplotlib.pyplot as plt
import numpy as np
from scipy.fftpack import rfft
fig = plt.gcf()
fig.set_size_inches(13, 8)

N = 1416
T = 1.0
x = df_union[u'Hora']
y = df_union[u'WaterLevel']
yf = rfft(y)
xf = np.linspace(0.0, 1.0/(2.0*T), N/2)
plt.plot(xf, 2.0/N * np.abs(yf[0:N/2]))
plt.xlim(0.06,0.18)
plt.ylim(0,0.5)
mplt.ylabel('Water Level(m)')
mplt.xlabel('Frecuencia (1/hr)')
mplt.title('Nivel del mar [Enero(2017)-Febrero(2017),
Bermuda, St. Georges Island]')
plt.grid()

#Identificando picos significativos
plt.text(0.076,0.04,'2"N2')
```

```
#13.16
plt.text(0.082,0.07,'L2')
#12.1951
plt.text(0.157,0.06,'M4')
#6.3694
plt.text(0.160,0.37,'MN_4')
#6.25
plt.text(0.165,0.07,'S_4')
#6.06
#Mostrar gráfico
plt.show()
```



Para encontrar los modos, se hizo de la misma forma, y los resultados que se obtuvieron fueron los siguientes:

Cuadro 2: Tabla de datos Bermudas

Pico (frecuencia)	Periodo	Amplitud	T
0.076	13.16	0.04	2"$N2$
0.082	12.1951	0.07	L_2
0.157	6.3694	0.06	M_4
0.160	6.25	0.37	MN_4
0.165	6.06	0.07	S_4

3. Referencias

1. Análisis de Fourier
   $https://en.wikipedia.org/wiki/Fourier_analysis/$
2. Análisis armónico
   $https://en.wikipedia.org/wiki/Harmonic_analysis$
3. Actividad Computacional
   http://computacional1.pbworks.com/w/page/116353369/
   Actividad %206 %20(2017-1)
4. Show specific columns
   http://stackoverflow.com/questions/11285613/selecting-columns