



UNIVERSIDAD DE SONORA

DIVISIÓN DE CIENCIAS EXACTAS Y NATURALES

DEPARTAMENTO DE FÍSICA

LICENCIATURA EN FÍSICA

FÍSICA COMPUTACIONAL I

---

# Análisis armónico de mareas

---

Martinez López Lizbeth Vanessa

Profesor del curso  
Dr. Carlos Lizárraga Celaya

22 Marzo del 2017

---

## Resumen

En este reporte se habla sobre la transformada de Fourier y el desarrollo de la práctica para llevar a cabo dicha transformada en Python, siendo los datos de la práctica anterior los utilizados para hacer la transformada. Nuevamente trabajamos con los niveles de mar en las costas seleccionadas anteriormente, para así poder identificar modos en la transformada de Fourier y etiquetar el componente de marea que representa dicho modo.

---

## 1. Introducción

En la actualidad, existen distintos métodos para el análisis de mareas, estos permiten conocer mejor el comportamiento de las mareas y así hacer predicciones bastante acertadas sobre dichos fenómenos. Uno de los métodos (o modelos) utilizados para lograr lo ya mencionado, es el análisis armónico usando la transformada de Fourier.

Una transformada de Fourier es una operación matemática que transforma una señal de dominio de tiempo a dominio de frecuencia y viceversa. Una DFT (Transformada de Fourier Discreta - por sus siglas en inglés) es el nombre dado a la transformada de Fourier cuando se aplica a una señal digital (discreta) en vez de una análoga (continua).

El objetivo principal de esta actividad es hacer una transformada discreta de Fourier para transformar los datos de marea y así poder identificar con los modos, las principales componentes armónicas de las mareas en las costas de Atlantic City, New Jersey y Cabo San Lucas, Baja California Sur.

Con ayuda del paquete `fftpack` de SciPy podemos realizar el objetivo de la práctica,

para ello se llevó a cabo el procedimiento que más adelante se menciona.

Para el desarrollo de esta práctica, nuevamente hacemos uso de los archivos de niveles del mar descargados en los sitios web del NOAA y CICESE, para las costas ya mencionadas. Estos archivos contienen los datos de un mes a elegir, en mi caso para Atlantic City se eligió el mes de Enero del presente año y para Cabo San Lucas el mes de Abril del 2016.

## 2. Desarrollo

En la práctica anterior se trató con los archivos descargados, es decir, se hizo limpieza, entre otras cosas si se requerían, por lo que no fue necesario hacer nuevamente la descarga y/o limpieza de archivos. Estos archivos son los que utilizaremos para realizar la transformada de fourier.

### 2.1. Atlantic City, New Jersey

Como ya se mencionó, para los datos de Atlantic City, se utilizaron los recopilados en el mes de Enero del 2017, por lo que la transformada se aplicó a dichos datos. Para iniciar con aplicación de la transformada de Fourier, se inició por importar las bibliotecas necesarias como Pandas, Numpy, Matplotlib, Pylab, etc. Ya con las bibliotecas importadas se hizo la lectura del archivo con los datos, para tener más facilidad, se definieron las columnas con un nombre en específico utilizando lo siguiente `df.columns=['Date', 'Water Level', 'Sigma', 'I', 'L']`; aquí sólo trabajaremos con la columna 'Date' y 'Water Level'.

Con el siguiente código, verifiqué si había presencia de celdas vacías en las columnas:

```
df.apply(lambda x: sum(x.isnull()), axis=0)
```

Al ver que no había renglones donde no hubiera datos, continué con la graficación. Vi que el número de datos en mi archivo es 744 y empleé el siguiente código para realizar la gráfica de transformada de furier:

```
from scipy.fftpack import fft, fftfreq, fftshift
# number of signal points
N = 744
# sample spacing
T = 1
x = df['Date']
y = df['Water Level']
yf = fft(y)
xf = fftfreq(N, T)
xf = fftshift(xf)
yplot = fftshift(yf)
import matplotlib.pyplot as plt
plt.plot(xf, 1.0/N * np.abs(yplot))
plt.xlim(-0.1, 0.1)
fig=plt.gcf()
fig.set_size_inches(7,7)

plt.xlabel("Frecuencia (1/hr)")
plt.ylabel("Amplitud (m)")

plt.text(.04,.08,"O$_1$")
plt.text(.08,.31,"M$_2$")
plt.text(.083,.07,"S$_2$")

plt.grid()
```

```
plt.show()
```

Esto me permitió graficar el nivel de manchas en el eje y y la frecuencia en el eje x. En el código N es el número de datos y T el 'periodo' para estos datos. Se obtuvo como resultado la gráfica de la transformada, para poder etiquetar los modos con las componentes armónicas se realizó lo siguiente:

Para poder obtener las componentes armónicas, fue necesario calcular el período de las frecuencias de los modos que más sobresalían en la transformada. En esta práctica se hicieron cálculos aproximados de los períodos, porque para las frecuencias consideré valores aproximados que podía ver a simple vista. Al obtener los períodos, pude identificar los modos, etiquetándolos con el respectivo componente.

## 2.2. Cabo San Lucas, Baja California Sur

Para esta costa, se utilizó el archivo con los datos de Abril del 2016. El procedimiento a seguir fue el mismo que el ya mencionado en la sección anterior, siendo pequeños detalles los que difieren.

En este caso no fue necesario definir las columnas, pues con el archivo que contaba ya tenían los nombres definidos. Aquí contamos con columna de Año, Mes, Día y Hora, por lo que fue necesario juntar estas columnas para crear una con formato de fecha, el código empleado fue el siguiente:

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

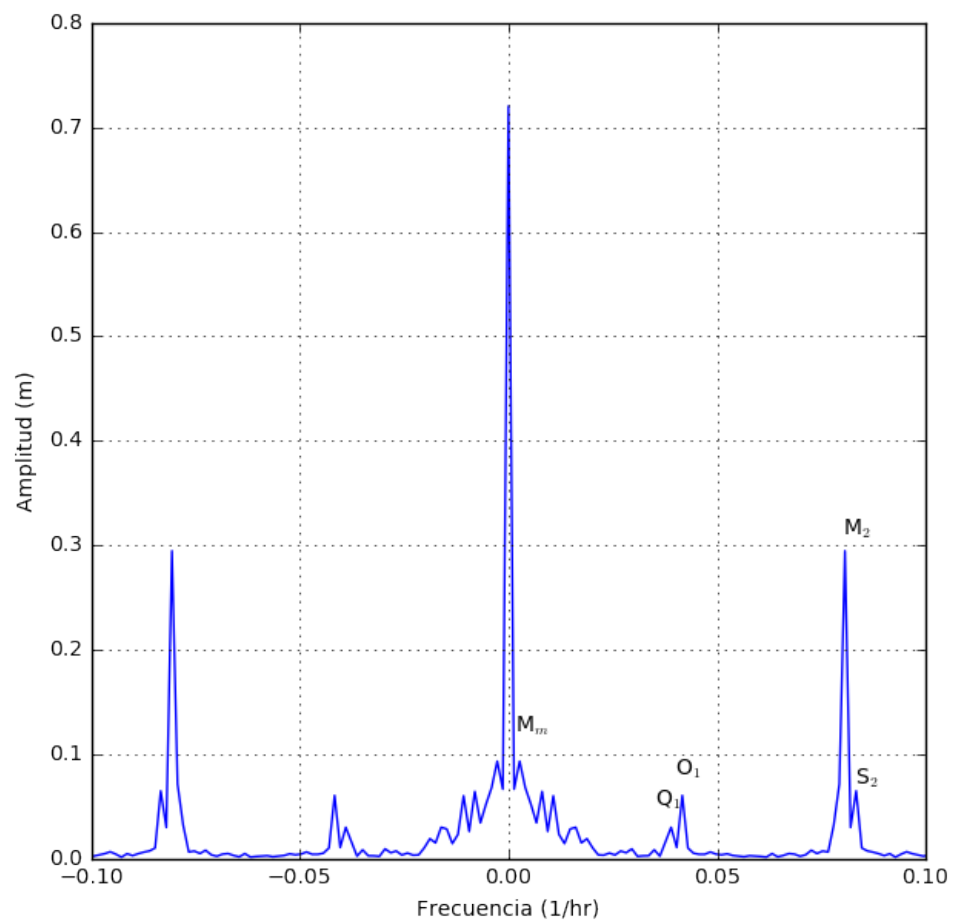
Ya con ésto realizado, el procedimiento siguió de la misma manera que lo dicho para Atlantic City, siendo el mismo código y método para el etiquetado de modos, el

utilizado aquí.

### 3. Resultados

#### 3.1. Gráficas

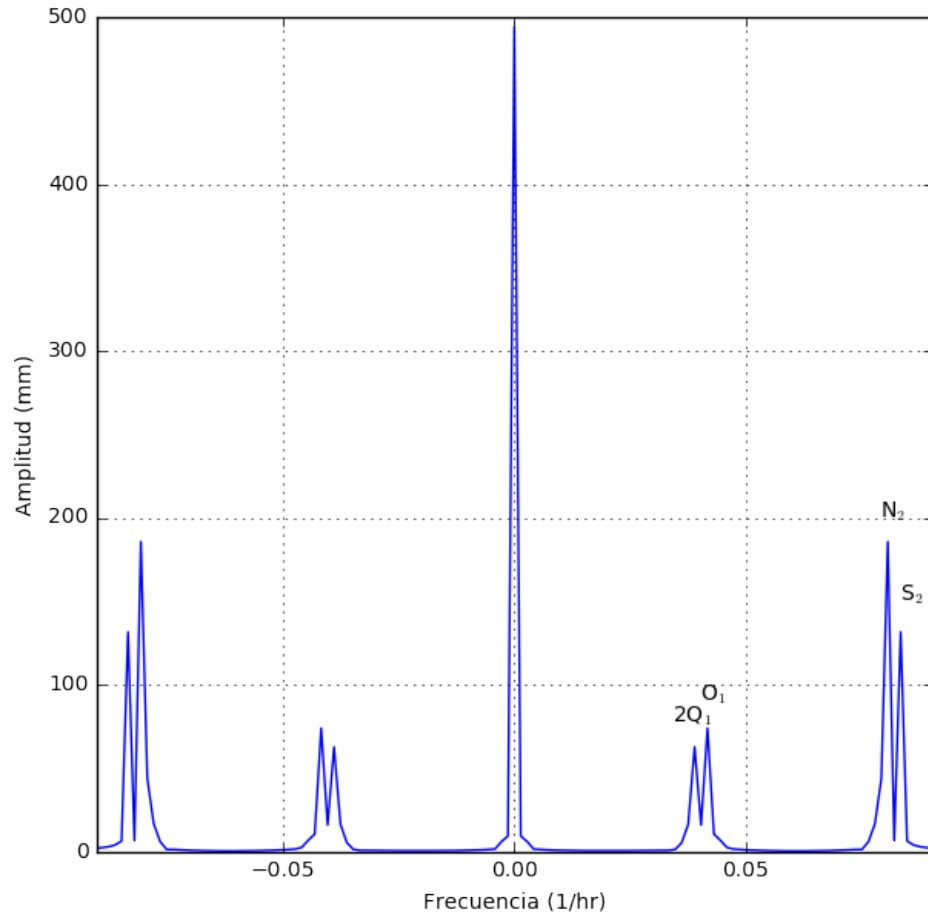
Transformada de Fourier para datos de Enero 2017, Atlantic City



Podemos observar que las principales componentes de las mareas son:

Lunar mensual, Lunar elíptica diurna más grande, Lunar diurnal, Principal lunar semidiurna y Principal solar semidiurnal.

### Transformada de Fourier para datos de Abril 2016, Cabo San Lucas



Podemos observar que las principales componentes de las mareas son:

Elíptica diurnal más grande, Lunar diurnal, Lunar elíptica semidiurnal más grande y Principal solar semidiurnal.

## 4. Conclusión

Con esta práctica pudimos observar que la transformada de Fourier es una herramienta matemática que ayuda bastante en el análisis de señales periódicas. En este caso se hizo el análisis para las mareas y fue fácil identificar las principales componentes armónicas de las mareas en cada costa. Además, la información que proporciona la transformada de Fourier, permite predecir el comportamiento de las mareas.

## 5. Referencias

- [1] *[http : //www.ni.com/support/esa/cvi/analysis/analy3.htm](http://www.ni.com/support/esa/cvi/analysis/analy3.htm)*
- [2] *[https : //docs.scipy.org/doc/scipy - 0,18,1/reference/tutorial/fftpack.html](https://docs.scipy.org/doc/scipy-0.18.1/reference/tutorial/fftpack.html)*
- [3] *[https : //tidesandcurrents.noaa.gov/stations.html?type = Water+Levels %29](https://tidesandcurrents.noaa.gov/stations.html?type=Water+Levels%29)*
- [4] *[http : //predmar.cicese.mx/](http://predmar.cicese.mx/)*