

## #PARCIAL 3 CORTE\_FISICA COMPUTACIONAL

**Diego Palacios**  
**Johan Mantilla**

### PUNTO 1

Encontrar la transformada inversa de Fourier, tal cual lo hicimos en la última clase, para el número de manchas solares como función del tiempo. Usando las dos frecuencias dominantes. Interprete y analice físicamente el significado de estas dos frecuencias.

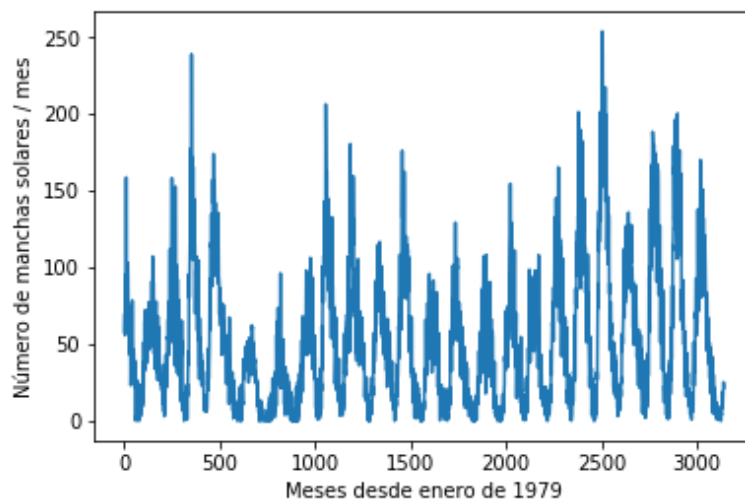
```
from numpy import*  
from numpy import loadtxt, zeros  
from pylab import plot, xlabel, ylabel, show, legend, grid,xlim, ylim, yscale, imshow
```

```
from numpy import zeros  
from cmath import exp, pi  
from numpy import array  
from numpy.fft import rfft
```

```
import matplotlib.pyplot as plt  
import numpy as np  
from numpy import pi
```

```
solman = loadtxt("sunspots.txt", float)
```

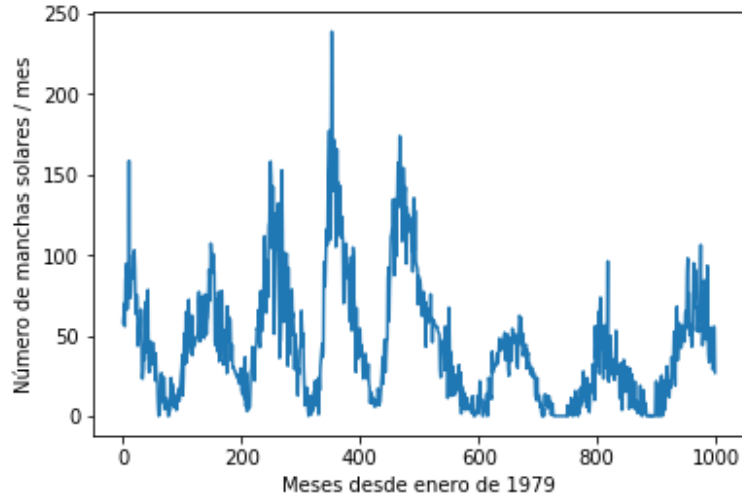
```
plot(solman[:,0], solman[:,1])  
xlabel("Meses desde enero de 1979")  
ylabel("Número de manchas solares / mes")  
show()
```



```

nm = 1000 # Número de meses
plot(solman[0:nm,0], solman[0:nm,1])
xlabel("Meses desde enero de 1979")
ylabel("Número de manchas solares / mes")
show()

```



```

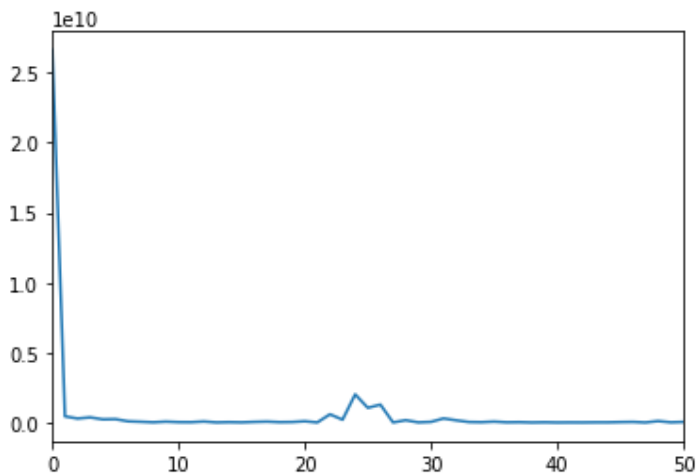
def dft(y):
    N=len(y)
    c=zeros(N//2+1, complex)
    for k in range(N//2+1):
        for n in range(N):
            c[k] += y[n]*exp(-2j*pi*k*n/N)
    return c

```

```

y = solman[:,1]
c = dft(y)
plot(abs(c)**2)
xlim(0,50)
show()

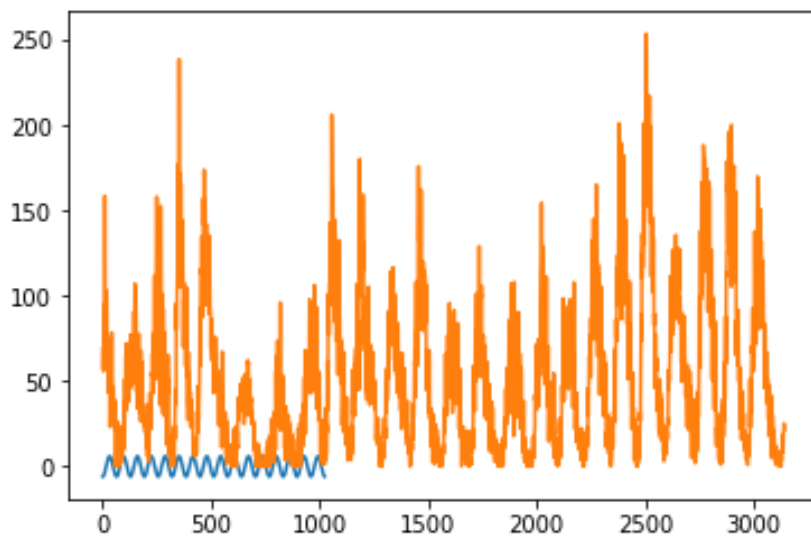
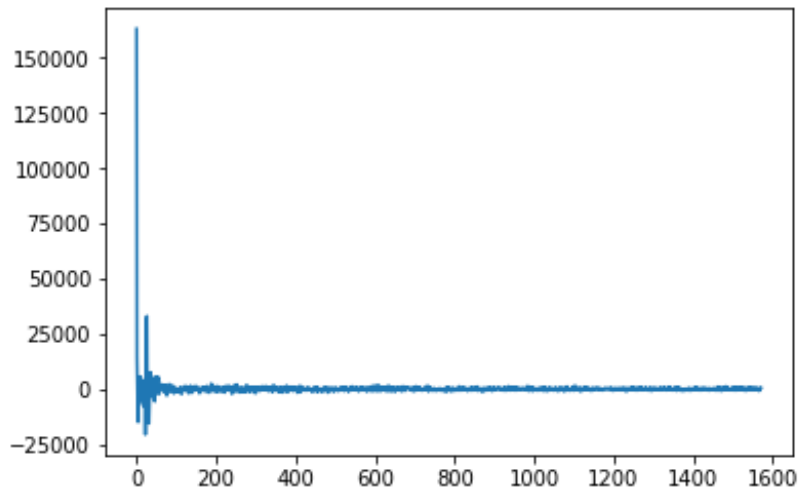
```



```

y = solman[:,1]
c=rfft(y)
plot(c)

```



```

print(len(y))
print(len(y)//16)
N=1024
f=16/N
h=zeros(N,complex)
for i in range(0,1024):
    h[i] = (c[16]/N)*cos(2*pi*f*i) + 2j*sin(2*pi*f*i)

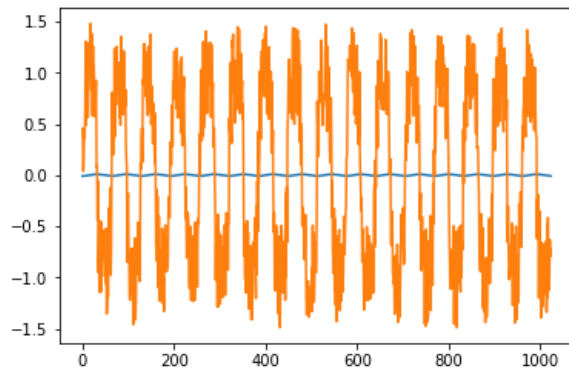
print(h)
plot(h)

```

plot(y)

show()

```
1024
64
[-0.00835281-0.56404197j -0.00831259-0.36529167j -0.00819231-0.16302341j
... -0.00799314-1.12032386j -0.00819231-0.9433847j
-0.00831259-0.75736023j]
```



## INTERPRETACIÓN:

\* Se observa que al transcurrir el tiempo tanto la cantidad de manchas solares como la frecuencia o período del ciclo de actividad solar no es constante.

Si analizamos por separado las dos mitades en las que dividimos la serie original, muestra que las dos mitades pueden no coincidir en amplitud, frecuencia y fase a lo largo del periodo analizado de duración de esta señal. Por lo tanto, ninguno de los ciclos tiene la misma duración o el mismo valor máximo.

\* Se puede inferir que no todos los ciclos tienen la misma duración ni tampoco la intensidad de la energía radiada es la misma.

\* Cuando se analiza dicha serie con la FFT y se calcula su espectro de potencia se pueden apreciar dos picos predominantes en el espectro de potencias.

\* Se puede concluir que tenemos básicamente dos picos principales (frecuencias predominantes) en el espectro de manchas solares: uno que se produce en un intervalo de tiempo pequeño y un segundo correspondiente con el intervalo de tiempo mayor al primero durante el cual cambia la cantidad de manchas solares.

## Punto 2

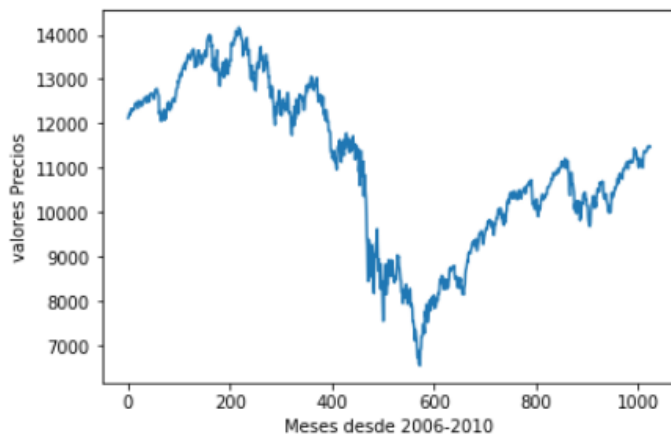
Hacer el ejercicio 7.4 del libro: Mark Newman, Computational Physics (2013, University of Michigan). interprete y analice físicamente.

En los recursos en línea encontrará un archivo llamado dow.txt que contiene el valor de cierre diario para cada día hábil desde fines de 2006 hasta finales de 2010 del dow jones Industrial average, que es una medida de los precios de average en el mercado de valores de EE. UU.

Escriba un programa para hacer lo siguiente:

A) lea los datos de dow.txt y grábelos en un gráfico

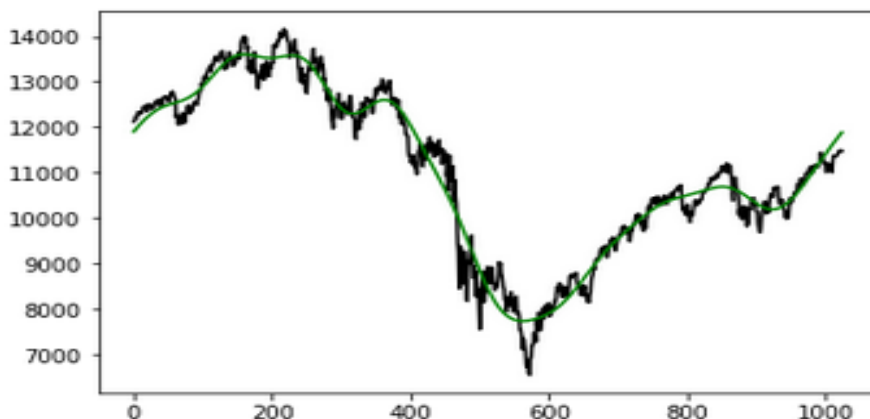
```
dow = loadtxt("dow.txt", float)
plot(dow)
xlabel("Meses desde 2006-2010")
ylabel("valores del Precio")
show()
```



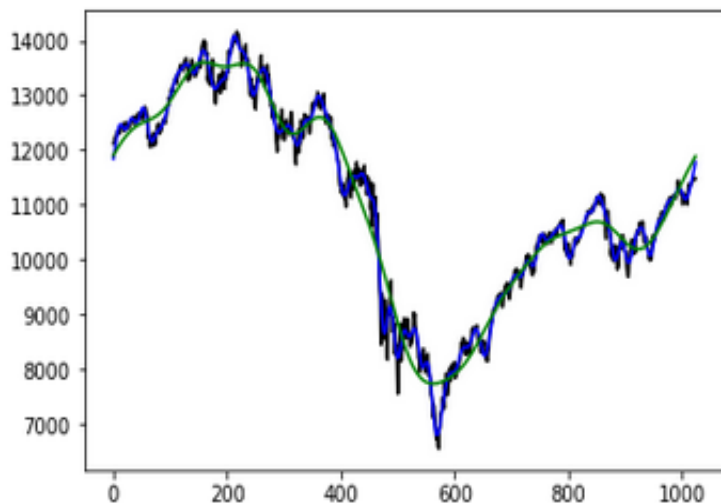
B) Calcule los coeficientes de la transformada discreta de Fourier de los datos utilizando la función `fft` de `numpy.fft`, que produce una matriz de números complejos!  $(1 / 2N) + 1$ .

C) Ahora ponga todos menos el primer 10% de los elementos de esta matriz a cero. (es decir, establezca el último 90% en cero pero mantenga los valores del primer 10%)

```
17
18 plot(dow, 'k')
19 #plot(dow_first_10_p, 'b')
20 plot(dow_first_2_p, 'g')
21 show()
```



D) Calcule la transformada inversa de Fourier de la matriz resultante, ceros y todo, utilizando la función 'irfft', y gráfiquela en el mismo gráfico que los datos originales. Es posible que deba variar los colores de las dos curvas para asegurarse de que ambas aparezcan en el gráfico. Comenten lo que ven. ¿Qué sucede cuando configura los coeficientes de Fourier a cero?

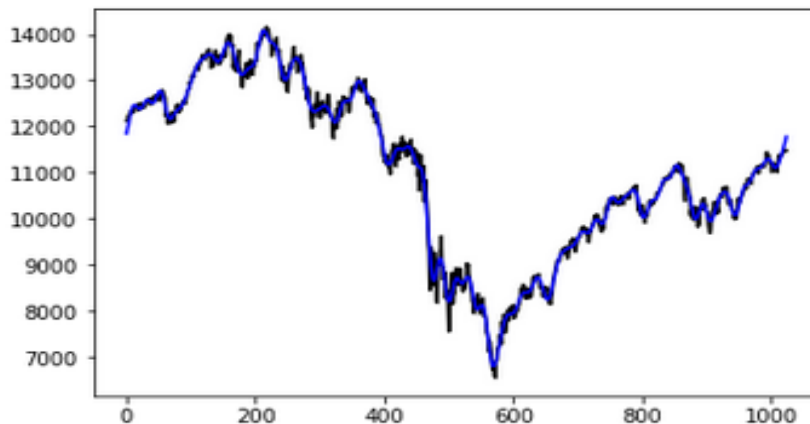


Rta:

si al colocar la función irfft si cambia los colores de las curvas y si aparecen ambas en el grafico también se observa que la grafica del 2% no tiene unos picos tan representativos como los de la grafica del 10%

E) Modifique su programa para que establezca todos menos el primer 2% de los coeficientes a cero y ejecútelo nuevamente.

```
17  
18 plot(dow, 'k')  
19 plot(dow_first_10_p, 'b')  
20 #plot(dow_first_2_p, 'g')  
21 show()
```



```
from scipy import loadtxt, zeros, copy, floor
from scipy.fftpack import rfft, irfft
from pylab import plot, show, xlabel, ylabel
```

```
dow = loadtxt("dow.txt", float)
# plot(dow)
# show()
```

```
dow_fourier = rfft(dow)
N = len(dow_fourier)
first_10_percent = zeros(N, float)
first_10_percent[0 : int(N / 10)] = copy(dow_fourier[0 : int(N / 10)])
dow_first_10_p = irfft(first_10_percent)
first_2_percent = zeros(N, float)
first_2_percent[0 : int(N / 50)] = copy(dow_fourier[0 : int(N / 50)])
dow_first_2_p = irfft(first_2_percent)
```

```
plot(dow, 'k')
plot(dow_first_10_p, 'b')
plot(dow_first_2_p, 'g')
show()
```

