Homework [1]

The Discrete Fourier Transform (DFT) of a periodic array f_j, for $j = 0, 1, \ldots, N-1$ (corresponding to data at equally spaced points, starting at the left end point of the interval of periodicity) is evaluated via the Fast Fourier Transform (FFT) algorithm (N power of 2). Use an FFT package, i.e. an already coded FFT (e.g. scipy.fftpack or numpy.fft).

1. Which of the following expressions define the Fourier coefficients (the DFT) that your fft package (name it) returns for $k = 0, 1, \ldots, N-1$?

$$c_k = \sum_{j=0}^{N-1} f_j e^{-i2\pi kj/N} \tag{1}$$

$$c_k = \sum_{j=1}^{N} f_j e^{-i2\pi kj/N} \tag{2}$$

$$c_k = \sum_{j=0}^{N} f_j e^{i2\pi k(j-1)/N} \tag{3}$$

2. Which of the following expressions define the inverse DFT computed by fft package?

$$f_j = \frac{2}{N}\sum_{k=0}^{N-1} c_k e^{i2\pi kj/N} \quad \text{for } j = 0, 1, \ldots, N-1 \tag{4}$$

$$f_j = \frac{1}{N}\sum_{k=0}^{N-1} c_k e^{i2\pi kj/N} \quad \text{for } j = 0, 1, \ldots, N-1 \tag{5}$$

$$f_j = \frac{1}{N}\sum_{k=1}^{N} c_k e^{-i2\pi kj/N} \quad \text{for } j = 0, 1, \ldots, N-1 \tag{6}$$

3. Prove that if the f_j, for $j = 0, 1, \ldots, N-1$ are real numbers then c_0 is real and $c_{N-k} = \bar{c_k}$, where the bar denotes complex conjugate. Hint: prove that $\omega_N^{N-k} = \bar{\omega}_N^k$.

4. Let $P_N(x)$ be the trigonometric polynomial of lowest order that interpolates the periodic array f_j, $j = 0, 1, \ldots, N-1$ at the equidistributed nodes $x_j = j(2\pi/N)$,

$j = 0, 1, \ldots, N-1$, i.e

$$P_N(x) = \frac{a_0}{2} + \sum_{k=1}^{N/2-1} (a_k \cos kx + b_k \sin kx) + \frac{a_{N/2}}{2} \cos\left(\frac{N}{2}x\right) \tag{7}$$

for $x \in [0, 2\pi]$, where

$$a_k = \frac{2}{N} \sum_{j=0}^{N-1} f_j \cos kx_j \quad \text{for } k = 0, 1, \ldots, N/2, \tag{8}$$

$$b_k = \frac{2}{N} \sum_{j=0}^{N-1} f_j \sin kx_j \quad \text{for } k = 1, \ldots, N/2-1. \tag{9}$$

Write a formula that relates the complex Fourier coefficients computed by your fft package to the real Fourier coefficients, a_k and b_k, that define $P_N(x)$.

5. Using your fft package and Prob. 4, find $P_8(x)$ on $[0, 2\pi]$ for the following periodic array:
   $f_0 = 6.000000000000000$
   $f_1 = 10.242640687119284$
   $f_2 = 2.000000000000000$
   $f_3 = -2.585786437626905$
   $f_4 = 2.000000000000000$
   $f_5 = 1.757359312880716$
   $f_6 = -6.000000000000000$
   $f_7 = -5.414213562373098$

6. Let $f_j = e^{\sin x_j}$, $x_j = j2\pi/N$ for $j = 0, 1, \ldots, N-1$. Take $N = 8$. Using your fft package obtain $P_8(x)$ and find a spectral approximation of the derivative of $e^{\sin x}$ at x_j for $j = 0, 1, \ldots, N-1$ by computing $P_8'(x_j)$. Compute the actual error in the approximation.

7. Note that $P_N'(x)$ is again a trigonometric polynomial of degree $\leq N/2$ and whose coefficients can be computed from those of $P_N(x)$ via the FFT. (a) Write a code to compute a (spectral) approximation to the derivative at $x_j = j(2\pi/N)$ for $j = 0, 1, \ldots, N-1$ for the corresponding periodic array f_j, $j = 0, 1, \ldots, N-1$, using one DFT and one inverse DFT, i.e. in order $N \log_2 N$ operations. (b) Test your code by comparing with your answer of the previous problem. (c) Observe the behavior of the error as N is increased to 16, and 32. **Note**: the contribution to the derivative from the $k = N/2$ node should be zero. Make sure to set the Fourier coefficient for $k = N/2$ *of the derivative* equal to zero.