Problem Set 6

Due: 4:00 pm Wednesday, February 24

1/5. Write a program to calculate integrals, i.e. $g(x) = \int_{-1}^{x} f(\xi)\,\mathrm{d}\xi$, using the Chebyshev transform. Show that your code is correct by integrating 1, x^3, and x^6. Then use your program to integrate the functions

(a) $f(x) = xe^{-x^2/2}, \quad -1 \le x \le 1$

(b) $f(x) = \begin{cases} -2x-1 & \text{if } -1 \le x \le 0 \\ 2x-1 & \text{if } \quad 0 \le x \le 1 \end{cases}$.

Take $N = 4$, 8, and 16 and compare to the exact answers.

2/5. In this problem we investigate the convergence behavior of the Chebyshev series for wavelike functions. Define the function of interest

$$f(x) = \sin(M\pi x) \qquad \text{for } -1 \le x \le 1$$

and denote the discrete Chebyshev approximation of its derivative using N points f'_N. Compute the error in the derivatives computed using the discrete Chebyshev transform

$$\epsilon_N = \frac{1}{N}\sum_{j=0}^{N} |f'(x_j) - f'_N(x_j)|^2 \quad ; \qquad x_j = \cos(\pi j/N)$$

for $N = 10, 20, 40, 80, 160, 320, 640$. Repeat for $M = 10, 20, 40, 80$ and plot your results with the error on the y-axis and N/M on the x-axis. Make sure to use different symbols (e.g. triangles, circles, squares, diamonds) for each value of M.

Discuss what you observe. How many polynomials do you need for exponential convergence?

NOTE: There are a few cases to run here and I don't expect it to take much computation time, though if you do run into issues reach out to me and we can discuss speeding up your code or code structure etc. Matlab on my laptop was able to complete this problem in 0.02 seconds (0.6 seconds including plotting) with a 92 line script (including comments and plotting).

3/5. Consider the Bessel series representation of a function $f(x)$

$$f(x) = \sum_{n=1}^{\infty} c_n J_0(j_{0,n}x)$$

and the truncated series

$$f(x) \approx S_N(x) = \sum_{n=1}^{N} c_n J_0(j_{0,n} x)$$

where J_0 is the zeroth order Bessel function of the first kind and $j_{0,n}$ is its nth root. We will use the function

$$f(x) = 1 \qquad \text{for} \;\; 0 \leq x \leq 1.$$

For this particular $f(x)$ the coefficients c_n can be found using the orthogonality property of the Bessel functions to be

$$c_n = \frac{2}{j_{0,n} J_1(j_{0,n})} \tag{1}$$

where J_1 is the first order Bessel function of the first kind.

(a) Derive the coefficients of the Bessel series, c_n, shown in Equation (1). Recall from class that the weight function for the Bessel series is $w(x) = x$ and note that the orthogonality of the Bessel functions is

$$\int_0^1 x J_0(j_{0,m} x) J_0(j_{0,n} x) \mathrm{d}x = \frac{1}{2} \delta_{m,n} \left[J_1(j_{0,n}) \right]^2 .$$

(b) Plot S_N on the domain $0 \leq x \leq 1$ for $N = 10, 20, 40$. Make sure to plot with a sufficient number of grid points to see how the function behaves near the boundaries.

(c) Do you see Gibbs phenomenon at $x = 0$? What about $x = 1$? If so, what is the overshoot?

(d) Compute the error at $x = 0$ and the error in the interior of the domain (e.g. the max error in the region $0.4 \leq x \leq 0.6$) and plot them both vs N, what is the convergence rate at each of these locations?

NOTE:

- Routines for evaluating $J_0(x)$ and $J_1(x)$ should be available in whatever language you are using. In Matlab for example the commmands are `besselj(0,x)` and `besselj(1,x)` respectively.
- A function for computing the zeros of the bessel functions, $j_{o,n}$, may be more difficult to come by (I think there is one in scipy, `scipy.special.jn_zeros`, but I couldn't find one in Matlab). So I have uploaded a Matlab routine that does this, `besselzeroj.m`, and a csv file for those not using matlab and can't find a routine, *besselroots_j0n.csv*.