Homework 5, Due: Friday, 10/9

This assignment is due on **Friday, October 9**, by 11:59 PM. Your assignment should be well-organized, typed (or neatly written and scanned) and saved as a .pdf for submission on Canvas. You must show all of your work to receive full credit. For problems requiring the use of MATLAB code, please remember to also submit your .m-files on Canvas as a part of your completed assignment. Your code should be appropriately commented to receive full credit.

Problems

1 Consider the following data points

$$f(0) = 1, \qquad f(0.25) = 0.4724, \qquad f(0.5) = 0.2231, \qquad f(0.75) = 0.1054$$

which were generated from the function $f(x) = e^{-3x}$.

(a) (5 points) Use appropriate 1st, 2nd, and 3rd Lagrange interpolating polynomials to approximate $f(0.43)$.

(b) (5 points) Use the remainder formula to find a bound for the error in each case (when $n = 1$, 2, and 3), and compare that bound to the actual absolute error between the true function value and the polynomial approximation.

2 Consider the function

$$f(x) = \frac{1}{1 + 25x^2}$$

The goal of this problem is to explore the results of approximating $f(x)$ with Lagrange interpolating polynomials $P_n(x)$ of increasing degree n.

(a) (5 points) Write a MATLAB code that generates data from $f(x)$ at $N = n + 1$ equispaced nodes $x_0, \dots, x_n$ on the interval $[-1, 1]$ and then uses this data to evaluate $P_n(x)$ at specified x values in $[-1, 1]$. You may make use of the function `langrange.m` posted on Canvas. To obtain N equispaced nodes, you can use the `linspace` function in MATLAB; in particular, `linspace(-1,1,N)`. Evaluate $P_n(x)$ at a dense set of x values in $[-1, 1]$ to get a good sense of what the polynomial looks like.

Plot your results for $N = 5$, 10, 15, and 20, each in a separate figure. In each figure, plot both the true function $f(x)$ and your approximation of $P_n(x)$ at your evaluation points, along with markers showing the data points (x_i, y_i) where $y_i = f(x_i)$ for each $i = 0, \dots, n$. Use different line styles and/or colors, as well as a legend, to clearly distinguish between $f(x)$ and $P_n(x)$.

(b) (5 points) Comment on your results from part (a). What do you notice as N (and n, correspondingly) increases? Compute the ℓ_∞ and ℓ_2 norm of the difference between $f(x)$ and $P_n(x)$ for each N, and comment on your findings.

(c) (5 points) The effects seen in part (a) are due to what's known as Runge's phenomenon, which can occur when interpolating certain functions with polynomials of high degree over a set of equispaced nodes. Why does this phenomenon occur? Consider the derivatives $f'(x)$, $f''(x)$, $f^{(3)}(x)$, ... evaluated at the endpoint $x = 1$. How do the derivatives change, and how does this affect the approximation error?

(d) (5 points) Different methods of selecting nodes have been developed to minimize the effects of Runge's phenomenon. For example, Chebyshev nodes are distributed such that more nodes are spaced closer to the endpoints of the interval. The Chebyshev nodes in the interval $(-1, 1)$ are given by

$$x_i = \cos\left(\frac{2i-1}{2N}\pi\right), \qquad i = 1, \ldots, N.$$

Make a plot comparing $f(x)$ with the nth Lagrange interpolating polynomial computed using Chebyshev nodes with $N = 15$. As in part (a), your plot should include both the true function $f(x)$ and your approximation of $P_n(x)$ at your evaluation points, along with markers showing the data points (x_i, y_i) where $y_i = f(x_i)$ for each $i = 0, \ldots, n$. Use different line styles and/or colors, as well as a legend, to clearly distinguish between $f(x)$ and $P_n(x)$. Discuss your results, and compare the difference in using the Chebyshev nodes as opposed to the equispaced nodes in part (a).

Note: For any of the above problems for which you use MATLAB to help you solve, you must submit your code/.m-files as part of your work. Any code that you submit should be your own. Your code must run in order to receive full credit. If you include any plots, make sure that each has a title, axis labels, and readable font size, and include the final version of your plots as well as the code used to generate them.