Homework 6
Math 151A: Numerical Methods
Due: Wed, February 21

1 Pen and paper

For the following problems, use the polynomial interpolation error bound for equally spaced data that has been presented in class.

1. Consider the task of approximating $\sin(x)$ over the interval $[0, 1]$. If one uses a polynomial interpolant based upon $n+1$ equally spaced data points in $[0, 1]$, $x_i = ih,\ h = \frac{1}{n},\ i = 0, \ldots, n$, how many points are required so that the error bound for the interpolant is less than 1×10^{-6}?

2. To get an idea about the impact that "noise" has on the errors associated with equidistant interpolation, consider the task of interpolating the function $f(x) = 1.0 + \epsilon \sin(1000x)$ over the interval $[0, 1]$. (If ϵ is small, then $\epsilon \sin(1000x)$ is a model for "noise" being added to the function.)

   (a) Give a general bound for $\max x \in [0, 1]|f^{(n+1)}(x)|$.

   (b) Using (a), give a general bound for $\max x \in [0, 1]|p(x) - f(x)|$ where $p(x)$ is the interpolating polynomial obtained by interpolating $f(x)$ at $n+1$ equally spaced points in the interval $[0, 1]$.

   (c) For a value of $\epsilon = .001$, give values for the bound in (b) when $n = 5,\ n = 10$, and $n = 20$.

   (d) Do you expect that high order equidistant interpolation is acceptable for functions that are contaminated by noise?

Submit In Class: The answers to the Pen and Paper problems, 1 and 2, in class on Wednesday, Feb. 21.

2 Programming

One of the skills that a numerical analyst must posses is an ability to translate written descriptions of algorithms into working routines. In this assignment you will get some practice at this, since the first part of the assignment consists of translating two routines Divdif and Interp into code (*routines.pdf*). These routines are for constructing and evaluating the Newton form of an interpolating polynomial. The routines come from "An Introduction to Numerical Analysis" by Atkinson. Warning! Warning! Matlab uses array indices that start with index 1; this means that you have to figure out how to translate loop bounds and array indices from an index range of 0..n to 1..(n+1).

1. The program *PolyTest.m* is a script that uses routines Divdif and Interp that are declared but not implemented in *Divdif.m* and *Interp.m*. Download the two files *Divdif.m* and *Interp.m* and finish the implementation of Divdif and Interp using the text description in *routines.pdf*.

2. Test your implementation by evaluating the error in the interpolation of $f(x) = x^7 - 2x$ over the interval $[0, 1]$ for $n = 5$, $n = 10$ and $n = 20$. Give the values of the error for $n = 5$, $n = 10$, $n = 20$. (Note : *PolyTest.m* already contains code to estimate the error; you just have to copy the output from the screen to another document (or change the code so it writes to a file). The script also plots the function and the interpolant — so you can view the results as they are computed.

3. For problem 2, what should the error be when $n = 10$ and $n = 20$? How do your computed results compare with your observed values?

4. Test the accuracy of polynomial interpolation on the function $f(x) = \sin(x)$ over the interval $[0, 1]$. Give the values of the error for $n = 10$, $n = 20$, $n = 30$.

5. Test the accuracy of the polynomial interpolation on the function $f(x) = \sin(x) + \epsilon \sin(100x)$ with $\epsilon = .001$ over the interval $[0, 1]$. Give the value of the error for $n = 10$, $n = 20$, $n = 30$.

Upload to CCLE: (a) A listing of your implementation of Divdif and Intep. (b) In a plain text file, answers to the questions and a record of the errors for computational problems 2-5.