APPM 4600 — HOMEWORK # 5

1. In this problem we find the the polynomial $p(x) = c_0 + c_1 x + c_2 x^2 + \cdots + c_n x^n$ that interpolates the data $(x_j, y_j) = (x_j, f(x_j)), j = 0, \ldots, n$.

   (a) Recall that out of all techniques discussed in class, interpolation with the barycentric Lagrange formulas is by far the most stable, and can be as efficient as Newton interpolation. That is, using one of the formulas:

   $$p(x) = \Psi_n(x) \sum_{j=0}^{n} \frac{w_j}{x - x_j} f(x_j).$$

   $$p(x) = \frac{\sum_{j=0}^{n} \frac{w_j}{x-x_j} f(x_j)}{\sum_{j=0}^{n} \frac{w_j}{x-x_j}}, \quad x \neq x_j.$$

   Where

   $$\Psi_n(x) = \prod_{i=0}^{n} (x - x_i), \quad w_j = \frac{1}{\prod_{i=0, i\neq j}^{n} (x_j - x_i)}.$$

   Say you are given $f(x)$, a vector of $n+1$ interpolation points x, and a vector of m target points z. Write down a pseudocode for a python function that uses barycentric Lagrange to evaluate the interpolant at target points, that is, $p(z)$. Briefly explain which of these formulas should be used, and why.

   [(b)] Implement the routine for the algorithm above in python. Then, use it to find the polynomial that interpolates

   $$f(x) = \frac{1}{1 + (16x)^2},$$

   in the points $x_i = -1 + (i-1)h, h = \frac{2}{n}, i = 1, \ldots, n+1$. Plot data points as circles (`plot(x,f,'o')`) and, in the same plot, plot the polynomial and $f(x)$ on a finer grid (still on $x \in [-1, 1]$), say with 1001 points. Observe what happens when you increase n. Try $n = 2, 3, 4, \ldots$ and continue until the maximum value of $p(x)$ is about 100 (should be for $N \sim 15 - 20$). As you can see the polynomial behaves badly near the endpoints of the interval due to Runge's phenomena.

   (c) Change the input interpolation points in your code to Chebyshev nodes

   $$x_j = \cos \frac{(2j+1)\pi}{2(n+1)}, \quad i = 0, \ldots, n,$$

   Perform the same tests as above. Briefly note the differences between the errors for these two kinds of interpolation points.

   (d) Recall that the interpolation error is always of the form $\frac{f^{n+1}(\alpha)}{(n+1)!} \Psi_n(x)$. For the largest n used in the experiments above, superimpose plots for $\log_{10} \Psi_n(x)$ for equispaced and Chebyshev nodes. How does this help explain how Chebyshev nodes avoid the Runge phenomenon?

2. Recall how we built the Lagrange basis for the standard and Hermite interpolation problems. Say we now have the following "mixed" polynomial interpolation problem: We want to find a quadratic polynomial $p(x)$ defined on $[-1, 1]$ such that $p(-1) = y_0, p(1) = y_1, p'(1) = z_1$.

   (a) Find a Lagrange basis for this problem. That is, find quadratics $L_0(x), L_1(x), L_2(x)$ such that they each satisfy one condition equal to 1 and the rest equal to 0.

   (b) Using this Lagrange basis, write down a formula for the interpolant $p(x)$ given data y_0, y_1, z_1.

   (c) Using standard and Hermite interpolation as inspiration, indicate what would be a Newton basis for this problem.

3. We now consider the problem of interpolating the data $(x_j, y_j) = (x_j, f(x_j)), j = 0, \ldots, n$ using a cubic spline.

   (a) Explain the difference between a natural and a periodic cubic spline.

   (b) Consider the derivation in class of an algorithm to obtain the coefficients for a natural cubic spline. Indicate what step would need to be modified if, instead, you wanted to obtain a representation for a periodic cubic spline.

   (c) Use code provided in modules or a scipy routine to interpolate the periodic function $f(x) = sin(9x)$ using periodic cubic splines with $n = 5, 10, 20, 40$ equispaced points in $[0, 1]$. Plot the logarithm of the interpolation error for each one, and briefly indicate what you observe.

4. Consider the following problems using discrete least squares:

   (a) Say we have four data points $(0, 1), (1, 4), (2, 2), (3, 6)$. We want to find the linear polynomial $p(x) = a_1 x + a_0$ that best fits this data in the least squares sense. Write down the least squares optimization problem, and find the system of normal equations $\mathtt{G}a = b$. Solve this system to find the coefficients a_0 and a_1.

   (b) We now want to consider a weighted least squares approach. That is, we want to minimize the quantity:

   $$q_w(a_0, a_1) = \sum_{i=0}^{3} w_i (p(x_i) - y_i)^2$$

   with $w_0 = 1, w_1 = 4, w_2 = 9, w_3 = 6$.
   Let D be a 4×4 diagonal matrix with entries $\mathtt{D}(i, i) = \sqrt{w_i}$. Show that q_w can be written in the form $||\mathtt{D}(\mathtt{M}a - y)||^2$. What are the normal equations for this problem?

   (c) Using the information above, find the coefficients for the weighted least squares problem. Compare plots for both lines against the given data, and explain intuitively what the effect of the weights is on the linear fit.