MCS 471 Project Three: Barycentric Lagrange Interpolation

The goal of this project is to introduce the barycentric formula for polynomial interpolation.

1. The Interpolation Problem

On input are $n+1$ tuples (x_i, f_i), for $i = 0, 1, \ldots, n$, where the interpolating points x_i are mutually distinct: $x_i \neq x_j$, for all $j \neq i$.

The *barycentric formula* for the interpolating polynomial p is

$$p(x) = \frac{\displaystyle\sum_{i=0}^{n} \frac{w_i}{x - x_i} f_i}{\displaystyle\sum_{i=0}^{n} \frac{w_i}{x - x_i}}, \quad w_i = \frac{1}{\displaystyle\prod_{\substack{j = 0 \\ j \neq i}}^{n} \left(x_i - x_j \right)}.$$

To see that p interpolates, that is: $p(x_k) = f_k$, for $k = 0, 1, \ldots, n$, consider the Lagrange form of the interpolating polynomial:

$$\sum_{i=0}^{n} \ell_i(x) f_i, \quad \ell_i(x) = \prod_{\substack{j = 0 \\ j \neq i}}^{n} \left(\frac{x - x_j}{x_i - x_j} \right).$$

We rewrite the i-th Lagrange polynomial as

$$\ell_i(x) = \prod_{\substack{j = 0 \\ j \neq i}}^{n} \left(x - x_j \right) \prod_{\substack{j = 0 \\ j \neq i}}^{n} \left(\frac{1}{x_i - x_j} \right) = \ell(x) \frac{w_i}{x - x_i}, \quad \ell(x) = \prod_{j=0}^{n} \left(x - x_j \right)$$

As the sum of the Lagrange polynomials equals one, we divide the Lagrange form of the interpolating polynomial by the sum of the Lagrange polynomials:

$$\frac{\displaystyle\sum_{i=0}^{n} \ell_i(x) f_i}{1} = \frac{\displaystyle\sum_{i=0}^{n} \ell(x) \frac{w_i}{x - x_i} f_i}{\displaystyle\sum_{i=0}^{n} \ell(x) \frac{w_i}{x - x_i}}.$$

Canceling $\ell(x)$ in both numerator and denominator shows the equivalence between the Lagrange form and the barycentric formula.

Evaluating all weights w_i takes $O(n^2)$ operations. Once the weights w_i are computing, the cost of evaluating p at some value for x is linear in n.

Adding one point x_{n+1} takes $O(n)$ operations to update all weights, an advantage similar to Newton interpolation. An added advantage of the barycentric formula over Newton interpolation is that this addition depends only on the interpolating points, unlike the divided differences which require all data.

2. Chebyshev Points of the Second Kind

Interpolating at the roots of the Chebyshev polynomials gives a uniform error. The $n+1$ Chebyshev points *of the second kind* are

$$x_i = \cos\left(\frac{i\,\pi}{n}\right), \quad \text{for } i = 0, 1, \ldots, n.$$

What are the weights for those interpolation points?

Assignment One. For the Chebyshev points of the second kind, compute the weights for $n = 4, 8, 16$. Do you observe a pattern in the values of the weights?

Hint: compare the weights to a suitable power of 2, divide by n, and take into account the sign changes.

Define a function `cw` that returns as `cw(n, i)` the `i`-th weight for the Chebyshev points of the second kind, for some `i` in $\{0, 1, \ldots, \mathtt{n}\}$.

Verify your definition of `cw` with the `weight` function in the posted notebook, for $n = 4, 8, 16$.

2. The Condition Number

The condition number at x of interpolating f at $n+1$ points is

$$\mathrm{cond}(x, n, f) = \frac{\displaystyle\sum_{i=0}^{n} \left| \frac{w_i f_i}{x - x_i} \right|}{\displaystyle\left| \sum_{i=0}^{n} \frac{w_i f_i}{x - x_i} \right|}, \quad w_i = \frac{1}{\displaystyle\prod_{\substack{j=0 \\ j \neq i}}^{n} \left(x_i - x_j \right)}.$$

Interpolating at equidistant points is ill conditioned. So let us verify the above formula for equidistant points and for Chebyshev points of the second kind.

Assignment Two. Consider $f = 1$. Evaluate $\mathrm{cond}(x, n, f)$ for $n = 4, 8, 16, 32, 64, 128, 256, 512$, at some random point x,

- for $n+1$ equidistant points in the interval $[-1, +1]$, and
- for the $n+1$ Chebyshev points of the second kind.

Run for four different random points.

Observe and compare the growth of the condition numbers.

Is Chebyshev interpolation indeed numerically better?

3. Numerical Stability

The formulas for barycentric Lagrange have $x - x_i$ in the denominator. While testing whether $x = x_i$ is straightforward, what happens when large numbers are appearing when $x \approx x_i$?

Assignment Three. Consider $f = 1$ and $n = 16$. Use Chebyshev points of the second kind and evaluate the interpolating polynomial at $x_8 + 10^{-k}$, for $k = 4, 5, \ldots, 16$.

For each evaluation, report the difference with the exact value 1.

Summarize your observations.

4. The Deadline is Friday 28 October 2022, at 10am

Some important points.

1. You may (not must) work in pairs, with a partner of your own choosing.

   If you work in a pair, then you must declare your partner in an email to `janv@uic.edu` at the latest before 5PM on Friday 21 October.

2. The solution to this project should be in one single Jupyter notebook.

   - The notebook should run as a program.
     All cells should execute correctly in the order in which they appear.
   - For each assignment, use a separate header in the notebook.
   - Use text cells to answer the questions of the assignments.
   - Write complete, grammatically correct sentences to answer the questions.

   Explanations matter just as much as the numerical results and the code.

3. Submit your project before the deadline to gradescope.

   The name of your notebook should be `FirstNameLastName`
   where you replace `FirstName` and `Lastname` by your own first and last name.

   If you work as a pair, submit `FirstName1LastName1FirstName2LastName2`, as one single submission, where `FirstName1 LastName1` and `FirstName2 LastName2` are the names of the pair members, in alphabetic order according to the last name.

4. The first cell in your solution notebook should be a text cell that lists the name of its author(s) in alphabetical order.

5. There is an automatic extension of the deadline till 5PM on the same day. However, late submissions are penalized with ten points off. Submissions after 5PM will not be graded.

If you have questions about the project, you are welcome in the office hours.

References

[1] J.-P. Berrut and L. N. Trefethen: Barycentric Lagrange Interpolation. *SIAM Review*, vol. 46, pages 501-517, 2004.

[2] N. J. Higham: The numerical stability of barycentric Lagrange interpolation. *IMA J. Numer. Anal.* vol. 24, pages 547-556, 2004.

[3] L. N. Trefethen: *Approximation Theory and Approximation Practice. Extended Edition*, SIAM 2020.