Notes

Clay S. Turner

Excerpt Mar 3,2009

Interpolation Formulae

Since we are working with polynomials represented by N+1 points, we need to state "how the dots are connected" to be able to find representations of differential and integral operators. Most students are taught that LaGrangian interpolation is fraught with numerical problems and should be avoided. However what the students are not taught is there is a wonderfully stable (from a numerical point of view) form of Lagrangian interpolation. This is known as the Barycentric form and a large reason why it works so well is the near cancellation of numerical errors in its formulation. This formulation is not well known.

Derivation of Barycentric Lagrangian Interpolation Formulae

First we start with the standard Lagrangian interpolation formula.

$$p(x) = \sum_{j=0}^{n} y_j \prod_{\substack{i=0 \\ i \neq j}}^{n} \frac{x - x_i}{x_j - x_i}$$

Next we multiply top and bottom with $x - x_j$ obtaining:

$$p(x) = \sum_{j=0}^{n} y_j \frac{\prod_{i=0}^{n} x - x_i}{\left(x - x_j\right) \prod_{\substack{i=0 \\ i \neq j}}^{n} x_j - x_i}$$

Now we define the following:

$$\ell(x) \equiv \prod_{i=0}^{n} x - x_i$$

$$w_j \equiv \frac{1}{\prod_{\substack{i=0 \\ i \neq j}}^{n} x_j - x_i}$$

After substituting our defined functions into the expression for $p(x)$, we find

$$p(x) = \ell(x) \sum_{j=0}^{n} \frac{w_j}{x - x_j} y_j$$

This is known as the barycentric formula of the first form. Now use this form to interpolate the constant function, $y = 1$ to get:

$$1 = \ell(x) \sum_{j=0}^{n} \frac{w_j}{x - x_j}$$

Finally just divide this into the formula of the first form to obtain the barycentric formula of the second form.

$$p(x) = \frac{\displaystyle\sum_{j=0}^{n} \frac{w_j}{x - x_j} y_j}{\displaystyle\sum_{j=0}^{n} \frac{w_j}{x - x_j}}$$

Chebyshev Nodes

One common mistake made with polynomial representations is the using of equally spaced nodes. Equally spaced nodes will cause the "weights" in the barycentric formula to span a very large range. This can be interpreted as having the interpolation rely much more heavily on some points than on other points. This problem not only occurs with the barycentric formulation, but it also occurs in other formulations as well since it is intrinsic to polynomial interpolation using equally spaced points. Another way of looking at this problem is the equal spaced interpolation requires the near cancellation of very large numbers where the small result is combined with other small numbers. Hence the rounding error comprises a big percentage of the final value.

There is a set of nodes called the "Chebyshev" nodes that makes all but two of the weights have the same magnitude and the other two have magnitudes of just one half of all of the others. So an interpolation based on Chebyshev nodes has essentially all components contributing nearly equally. The result of this is minimizing the loss of precision in numerical calculations.

The N+1 Chebyshev nodes on the interval, $[a,b]$ are calculated as follows:

$$x_j = \frac{1-\cos\left(\frac{j\pi}{N}\right)}{2}\cdot(b-a)+a \text{ where } j\in\{0,1,2,\cdots,N\}$$

In the literature, most users define the Chebyshev nodes on the interval, $[-1,1]$. I added a translation to the formula to create the nodes on a user specified domain. A geometric interpretation for the nodes is they are the "x" coordinates of equi-angular points on a semicircle.

The barycentric formula (form 2) offers a neat simplification when it comes to the weights. Basically it allows for the scaling of all weights by a common factor without affecting the interpolation. So when the Chebyshev nodes are used, the magnitudes of all but two of the weights may be scaled to be just one! The other two become one half. The weights will, however, have alternating signs. But this simplification may be used to speed up the function interpolation by removing unnecessary multiplications.

Resampling

Even though the Chebyshev nodes will be used in the project, there are situations where a function will need to be represented on a different set of nodes. The change of nodes is in effect a resampling of the interpolating polynomial. So if our original function's nodes are given by $\{x_0, x_1, \cdots, x_n\}$ and we wish to represent it on a new set of nodes given by $\{X_0, X_1, \cdots, X_n\}$, then we can actually do this by "connecting the dots" with a continuous polynomial and sampling that polynomial at the new nodes. This is done by constructing a resampling matrix via the barycentric interpolation formula. Don't forget that each set of nodes has its own set of weights. So use the correct set. Our new function's samples are found by simply multiplying the old function's samples by the resampling matrix. Formally our resampling matrix's elements are built as follows:

$$R_{ji} = \frac{w_i}{\alpha_j \cdot (X_j - x_i)} \text{ where } \alpha_j = \sum_{k=0}^{n} \frac{w_k}{X_j - x_k}$$

So our resampling is effected by the following matrix multiply: $Y = R \cdot y$

Or more explicitly:

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} \frac{w_0}{\alpha_0(X_0 - x_0)} & \frac{w_1}{\alpha_0(X_0 - x_1)} & \frac{w_2}{\alpha_0(X_0 - x_2)} & \cdots & \frac{w_n}{\alpha_0(X_0 - x_n)} \\ \frac{w_0}{\alpha_1(X_1 - x_0)} & \frac{w_1}{\alpha_1(X_1 - x_1)} & \frac{w_2}{\alpha_1(X_1 - x_2)} & \cdots & \frac{w_n}{\alpha_1(X_1 - x_n)} \\ \frac{w_0}{\alpha_2(X_2 - x_0)} & \frac{w_1}{\alpha_2(X_2 - x_1)} & \frac{w_2}{\alpha_2(X_2 - x_2)} & \cdots & \frac{w_n}{\alpha_2(X_2 - x_n)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{w_0}{\alpha_n(X_n - x_0)} & \frac{w_1}{\alpha_n(X_n - x_1)} & \frac{w_2}{\alpha_n(X_n - x_2)} & \cdots & \frac{w_n}{\alpha_n(X_n - x_n)} \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$