261: 7th February

Interpolation

Last week, we saw how to use the conjugate gradient algorithm to solve large matrix equations in a finite number of iterations. In doing so, we brought to a conclusion our study of iterative methods. This week we shall look at interpolation.

Imagine that we have a set of data points, obtained from an experiment. It would be quite natural to plot the points and we would want to add a curve that fitted the points as smoothly as possible. This would enable us to predict values that fell between points.

Interpolation is the process of generating the equation that smoothly joins the data points.

Polynomial interpolation

The simplest starting point in our search for a good interpolation equation is the polynomial $a_0+a_1x+a_2x^2+a_3x^3+...+a_ix^i$. Our approach will be to try to find the values of $a_0, a_1, ..., a_i$ that best fit the data.

We choose the order of the equation, i, depending on how many datapoints there are. If the polynomial is of order i, then there are $i+1$ unknown coefficients, a_0 to a_i. However, if we have n data points, then we only have enough information to fix n of them, so we choose the order of the polynomial so that there are n coefficients.

In short, when we have n data points, we start with the polynomial

$$P(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + ... + a_{n-1}x^{n-1}$$

In choosing the a_j, we have two requirements. The first is that, between data points, the equation must be smooth. The second is that the equation must recreate all the data points.

Lagrange polynomials

One solution is to use a series of Lagrange polynomials, one for each data point. If we assume that our data points are of the form (x_i, f_i), The polynomials have the property that when $x = x_i$, the ith polynomial will be 1, whilst all the others will be 0.

The ith Lagrange polynomial looks like

$$L_i(x) = \frac{(x-x_0)(x-x_1)(x-x_2)...(x-x_{i-1})(x-x_{i+1})...(x-x_n)}{(x_i-x_0)(x_i-x_1)(x_i-x_2)...(x_i-x_{i-1})(x_i-x_{i+1})...(x_i-x_n)}$$

If we construct an interpolation equation using $n+1$ of these, one for each point, we get

$$P(x) = f_0L_0(x) + f_1L_1(x) + f_2L_2(x) + ... + f_nL_n(X)$$

with the property that $P(x_i) = f_i$.

Question 1) For the following data points, work out a_0, a_1 and a_2. (0,5),(1,2),(2,5). The data points are in the format $(x_i,\ f_i)$.

Neville's algorithm

Neville's algorithm is another method for generating the interpolating polynomial. It starts by creating functions that linearly interpolate between each pair of data points and proceeds to create a second set of functions that linearly interpolate between the first set. The algorithm repeateds this process of creating new functions that interpolate between the previous set, until no more interpolation is possible. Once this point has been reached, the equation that we are left with is our desired interpolating polynomial, P(x).

Let's illustrate this with the points $(x_0,\ f_0)$, $(x_1,\ f_1)$, $(x_2,\ f_2)$ and $(x_3,\ f_3)$. Interpolating linearly between f_0 and f_1, gives us the equation

$$P_{01} = \frac{(x - x_0)}{(x_1 - x_0)} f_1 + \frac{(x_1 - x)}{(x_1 - x_0)} f_0$$

which we take to be valid only in the region $x_0 < x < x_1$. Similarly we define P_{12} to be

$$P_{12} = \frac{(x - x_1)}{(x_2 - x_1)} f_2 + \frac{(x_2 - x)}{(x_2 - x_1)} f_1$$

which is valid in the region $x_1 < x < x_2$. We define P_{23} in the same fashion. The next step is to interpolate between P_{01} and P_{12} in the region $x_0 < x < x_2$ and between P_{12} and P_{23} in the region $x_1 < x < x_3$. We do this with the equations

$$P_{012} = \frac{(x - x_0)}{(x_2 - x_0)} P_{12} + \frac{(x_2 - x)}{(x_2 - x_0)} P_{01}$$

$$P_{123} = \frac{(x - x_1)}{(x_3 - x_1)} P_{23} + \frac{(x_3 - x)}{(x_3 - x_1)} P_{12}$$

Finally, we interpolate between these two equations to give us

$$P_{0123} = \frac{(x - x_0)}{(x_3 - x_0)} P_{123} + \frac{(x_3 - x)}{(x_3 - x_0)} P_{012}$$

as our final interpolation equation. P_{0123} is a third order polynomial, which is consistant with the requirement that we have the same number of coefficients as data points.

Question 2) Calculate a_0, a_1, and a_2, for the data points (0, 4), (1,7), (2,6), using Neville's algorithm. As before, the data points are of the form $(x_i,\ f_i)$.