Numerical Analysis

Math 370 — MWF 2:30-3:25
Spring 1997 — 127 Fowler

(02.28.97) Newton Polynomials and Divided Differences

Previously we have shown that it is a relatively trivial task to write down the Lagrange polynomials for a set of nodes $(x_0, f_0), (x_1, f_1), \ldots, (x_n, f_n)$

The only problem is that these polynomials are rather annoying to evaluate because they have expressions like: $(X - x_0)(X - x_1) \ldots (X - x_{i-1})(X - x_{i+1}) \ldots (X - x_n)$ which can easily overflow or underflow if n gets large.

One solution of this is to use the **Newton interpolating polynomials** instead. (Yes, this is the same Sir Isaac Newton who invented calculus, Newton's Method, Laws of Gravitation, *et cetera.*)

The general form of the Newton interpolating polynomial of degree n is

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots a_n(x - x_0)(x - x_1) \ldots (x - x_{n-1})$$

where a_k for $k = 0$ to n are "appropriate constants". The neat thing about this form of the interpolating polynomial is that computing the appropriate constants a_k is actually quite easy.

In addition Newton polynomials have the proprty that they can be written in **nested form** very easily. For example,

$$P_4(x) = [[[[a_4] * (x - x_3) + a_3] * (x - x_2) + a_2] * (x - x_1) + a_1] * (x - x_0) + a_0$$

Show that the nested form of the Newton Polynomial is exactly the same as "natural" form of the Newton Polynomial. (*Which form uses more arithmetic operations when evaluated?*)

$$P_4(x) = a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+a_3(x-x_0)(x-x_1)(x-x_2)+a_4(x-x_0)(x-x_1)(x-x_2)(x-x_3)$$

Example

For the general Newton Polynomial $P_n(x)$ what is the value of a_0? [To think of this, figure out what x value one would have to evaluate $P(x)$ at to get $P(x) = a_0$]

Using your answer for a_0 find a formula for a_1

Can you surmise a formula for a_2? What about a_k?

Divided Differences

The **zeroth divided difference** of the function $f(x)$ with respect to x_i is denoted by $f[x_i]$ and is given by $f(x_i)$. In other words, $f[x_i]$ is just the function $f(x)$ evaluated at x_i,

$$f[x_i] = f(x_i)$$

The **first divided difference** of f with respect to x_i and x_{i+1} is denoted by $f[x_i, x_{i+1}]$ and defined recursively in terms of zeroth divided differences:

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i}$$

As you may have guessed, in general the **kth divided difference** of f with respect to $x_i, x_{i+1}, \ldots, x_{i+k}$ is given by

$$f[x_i, x_{i+1}, \ldots, x_{i+k}] = \frac{f[x_{i+1}, x_{i+2}, \ldots, x_{i+k}] - f[x_i, x_{i+1}, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$$

The value of a_k, the coefficient of x^k of the Newton Polynomial is exactly the **kth divided difference**. **Example**

Consider the following data for a Bessel's function J(x) evaluated at certain points.

x	1.0	1.3	1.6	1.9	2.2
$J(x)$	0.7651977	0.6200860	0.4554022	0.2818186	0.1103623

Let's use a divided difference table to compute the coefficients of a Newton interpolating polynomial for this function in order to find $J(1.5)$

i	x_i	$f[x_i]$	$f[x_{i-1}, x_i]$	$f[x_{i-2}, \ldots, x_i]$	$f[x_{i-3}, \ldots, x_i]$	$f[x_{i-4}, \ldots, x_i]$

Thus,
$P_4(x) =$
and
$J(1.5) \approx P_4(1.5) =$

NOTE: Mon Classes Are Held In Fowler 201