Neville's algorithm

From Wikipedia, the free encyclopedia

In mathematics, **Neville's algorithm** is an algorithm used for polynomial interpolation that was derived by the mathematician Eric Harold Neville. Given $n + 1$ points, there is a unique polynomial of degree $\leq n$ which goes through the given points. Neville's algorithm evaluates this polynomial.

Neville's algorithm is based on the Newton form of the interpolating polynomial and the recursion relation for the divided differences. It is similar to Aitken's algorithm (named after Alexander Aitken), which is nowadays not used.

Contents [hide]

The algorithm [edit]

Given a set of $n+1$ data points (x_i, y_i) where no two x_i are the same, the interpolating polynomial is the polynomial p of degree at most n with the property

$p(x_i) = y_i$ for all $i = 0,\ldots,n$

This polynomial exists and it is unique. Neville's algorithm evaluates the polynomial at some point x.

Let $p_{i,j}$ denote the polynomial of degree $j - i$ which goes through the points (x_k, y_k) for $k = i, i + 1, \ldots, j$. The $p_{i,j}$ satisfy the recurrence relation

$$\begin{aligned} p_{i,i}(x) &= y_i, & 0 \leq i \leq n, \\ p_{i,j}(x) &= \frac{(x_j - x)p_{i,j-1}(x) + (x - x_i)p_{i+1,j}(x)}{x_j - x_i}, & 0 \leq i < j \leq n. \end{aligned}$$

This recurrence can calculate $p_{0,n}(x)$, which is the value being sought. This is Neville's algorithm.

For instance, for $n = 4$, one can use the recurrence to fill the triangular tableau below from the left to the right.

$$\begin{array}{lllll}
p_{0,0}(x) = y_0 & & & & \\
 & p_{0,1}(x) & & & \\
p_{1,1}(x) = y_1 & & p_{0,2}(x) & & \\
 & p_{1,2}(x) & & p_{0,3}(x) & \\
p_{2,2}(x) = y_2 & & p_{1,3}(x) & & \boxed{p_{0,4}(x)} \\
 & p_{2,3}(x) & & p_{1,4}(x) & \\
p_{3,3}(x) = y_3 & & p_{2,4}(x) & & \\
 & p_{3,4}(x) & & & \\
p_{4,4}(x) = y_4 & & & &
\end{array}$$

This process yields $p_{0,4}(x)$, the value of the polynomial going through the $n + 1$ data points (x_i, y_i) at the point x.

This algorithm needs $O(n^2)$ floating point operations.

The derivative of the polynomial can be obtained in the same manner, i.e:

$$\begin{aligned} p'_{i,i}(x) &= 0, & 0 \leq i \leq n, \\ p'_{i,j}(x) &= \frac{(x_j - x)p'_{i,j-1}(x) - p_{i,j-1}(x) + (x - x_i)p'_{i+1,j}(x) + p_{i+1,j}(x)}{x_j - x_i}, & 0 \leq i < j \leq n. \end{aligned}$$

Application to numerical differentiation [edit]

Lyness and Moler showed in 1966 that using undetermined coefficients for the polynomials in Neville's algorithm, one can compute the Maclaurin expansion of the final interpolating polynomial, which yields numerical approximations for the derivatives of the function at the origin. While "this process requires more arithmetic operations than is required in finite difference methods", "the choice of points for function evaluation is not restricted in any way". They also show that their method can be applied directly to the solution of linear systems of the Vandermonde type.

References [edit]

- Press, William; Saul Teukolsky; William Vetterling; Brian Flannery (1992). "§3.1 Polynomial Interpolation and Extrapolation (encrypted)" (PDF). *Numerical Recipes in C. The Art of Scientific Computing* (2nd edition ed.). Cambridge University Press. doi:10.2277/0521431085. ISBN 978-0-521-43108-8. (link is bad)
- J. N. Lyness and C.B. Moler, Van Der Monde Systems and Numerical Differentiation, Numerische Mathematik 8 (1966) 458-464 (doi: 10.1007/BF02166671)

External links [edit]

- Weisstein, Eric W., "Neville's Algorithm", *MathWorld*.
- Module for Neville Interpolation by John H. Mathews
- Java Code by Behzad Torkian

Categories: Polynomials | Interpolation