6. Interpolation and Curve Fitting.

NUMERICAL ANALYSIS. Prof. Y. Nishidate (323-B, nisidate@u-aizu.ac.jp)
http://web-int.u-aizu.ac.jp/~nisidate/na/

INTERPOLATION

Interpolation is a task of computing values of a tabulated function at points that are not in the table. Polynomials are widely used for interpolation purposes.

Lagrange Polynomials

Interpolation means to estimate a missing function value by taking the weighted average of known function values at neighboring points. Linear interpolation uses a line segment that passes through two points:

$$y = P(x) = y_0 + (y_1 - y_0)\frac{x - x_0}{x_1 - x_0}$$

This linear polynomial can be written in the form of Lagrange polynomial:

$$y = P_1(x) = y_0\frac{x - x_1}{x_0 - x_1} + y_1\frac{x - x_0}{x_1 - x_0}$$

After introduction of Lagrange coefficient polynomials

$$L_{1,0}(x) = \frac{x - x_1}{x_0 - x_1} \qquad L_{1,1}(x) = \frac{x - x_0}{x_1 - x_0}$$

the Lagrange polynomial of the first degree can be written as:

$$P_1(x) = \sum_{k=0}^{1} y_k L_{1,k}(x)$$

The generalization of the above relation is the construction of a polynomial $P_N(x)$ of degree N that passes through the $N+1$ points $(x_0,\ y_0)$, $(x_1,\ y_1)$, $\ldots (x_N,\ y_N)$ and has the form

$$P_N(x) = \sum_{k=0}^{N} y_k L_{N,k}(x)$$

where $L_{N,k}(x)$ is the Lagrange coefficient polynomial:

$$L_{N,k} = \frac{(x - x_0)\cdots(x - x_{k-1})(x - x_{k+1})\cdots(x - x_N)}{(x_k - x_0)\cdots(x_k - x_{k-1})(x_k - x_{k+1})\cdots(x_k - x_N)}$$

Note that the terms $(x - x_k)$ and $(x_k - x_k)$ do not appear in the above relation. Using product notation we can write the Lagrange coefficient polynomial as:

$$L_{N,k} = \left(\prod_{j=0, j\neq k}^{N} (x - x_j)\right) \bigg/ \left(\prod_{j=0, j\neq k}^{N} (x_k - x_j)\right)$$

The Lagrange coefficient polynomial $L_{N,k}(x)$ has the property

$L_{N,k}(x_j) = 1$ when $j = k$ and $L_{N,k}(x_j) = 0$ when $j \neq k$.

This means that the polynomial curve goes through all $(x_j,\ y_j)$.

Newton Polynomials

It is sometimes useful to find several approximating polynomials $P_1(x)$, $P_2(x)$, $\ldots P_N(x)$ and then choose the one which is better. If the Lagrange polynomials are used, there is no constructive relation between $P_{N-1}(x)$ and $P_N(x)$. Each Lagrange polynomial should be constructed individually. A new approach with the name **Newton polynomials** have the following recursive relations:

$$\begin{aligned}
&P_1(x) = a_0 + a_1(x - x_0)\\
&P_2(x) = P_1(x) + a_2(x - x_0)(x - x_1)\\
&P_3(x) = P_2(x) + a_3(x - x_0)(x - x_1)(x - x_2)\\
&\ldots
\end{aligned}$$

The polynomial $P_N(x)$ is obtained from $P_{N-1}(x)$ using the recursive relationship

$$P_N(x) = P_{N-1}(x) + a_N(x - x_0)(x - x_1)\cdots(x - x_{N-1})$$

This means that $P_N(x)$ is an ordinary polynomial of degree N.

In order to find the coefficients a_k for the polynomials $P_1(x), ..., P_N(x)$ let us consider first the polynomial of the first degree $P_1(x)$. In this case

$$P_1(x_0) = f(x_0) \qquad P_1(x_1) = f(x_1)$$

We find that

$$f(x_0) = P_1(x_0) = a_0 + a_1(x_0 - x_0) = a_0$$

The second condition at point x_1 leads to the relation

$$f(x_1) = P_1(x_1) = a_0 + a_1(x_1 - x_0) = f(x_0) + a_1(x_1 - x_0)$$

The coefficient a_1 is equal to

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

Using a_0 and a_1 it is possible to determine a_3 and so on.

Definition. The divided differences for a function $f(x)$ are defined as follows:

$$\begin{aligned}
&f[x_k] = f(x_k)\\
&f[x_{k-1}, x_k] = \tfrac{f[x_k] - f[x_{k-1}]}{x_k - x_{k-1}}\\
&f[x_{k-2}, x_{k-1}, x_k] = \tfrac{f[x_{k-1}, x_k] - f[x_{k-2}, x_{k-1}]}{x_k - x_{k-2}}\\
&\ldots\\
&f[x_{k-j}, x_{k-j+1}, ..., x_k] = \tfrac{f[x_{k-j+1}, ..., x_k] - f[x_{k-j}, ..., x_{k-1}]}{x_k - x_{k-j}}
\end{aligned}$$

It is possible to show that the Newton polynomial

$$P_N(x) = a_0 + a_1(x - x_0) + ... + a_N(x - x_0)(x - x_1)\cdots(x - x_{N-1})$$

has the coefficients, which are equal to the divided differences:

$$a_k = f[x_0, x_1, ..., x_k] \qquad k = 0,\ 1,\ ...,\ N$$

Example. Construct the divided difference table and Newton polynomial for the function given at six nodes (see first and second columns of the Table below).

x_k	$f[x_k]$	1st dd	2nd dd	3rd dd	4th dd	5th dd
$x_0 = 1$	$\underline{-3}$					
$x_1 = 2$	0	$\underline{3}$				
$x_2 = 3$	15	15	$\underline{6}$			
$x_3 = 4$	48	33	9	$\underline{1}$		
$x_4 = 5$	105	57	12	1	$\underline{0}$	
$x_5 = 6$	192	87	15	1	0	$\underline{0}$

Coefficients of the Newton polynomial are equal to $a_0 = -3$, $a_1 = 3$, $a_2 = 6$, $a_3 = 1$, $a_4 = a_5 = 0$. Thus the given function values are exactly described by the Newton polynomial of the third degree:

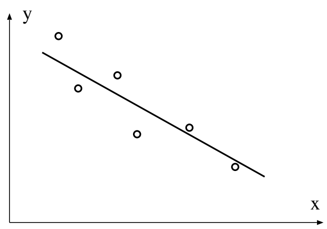
$$P_3(x) = -3 + 3(x - 1) + 6(x - 1)(x - 2) + (x - 1)(x - 2)(x - 3)$$

CURVE FITTING

In science and engineering it is often the case when experiment produces a set of data points. One goal of numerical methods is to determine a formula $y = f(x)$, which results in the most "reasonable" or "best" fit of experimentally measured values of (x_1, y_1), (x_2, y_2), $\ldots$,(x_N, y_N). Usually, a class of allowable formulas is chosen and then coefficients of approximation should be determined.

For example, the plot of the x and y values shown below suggests a straight-line fit, using a function such as

$$y = f(x) = Ax + B$$



How do we find the best linear approximation that goes near the points? Usual approach is to minimize the sum

$$E = \sum_{k=1}^{N} (y_k - f(x_k))^2$$

Here E is the sum of squares of deviations, y_k are experimental (measured) function values, $f(x_k)$ are values of the approximation function at x_k.

Least-squares line. In order to determine coefficients of the approximation $y = f(x) = Ax+B$ it is necessary to minimize the quantity

$$E = \sum_{k=1}^{N} (Ax_k + B - y_k)^2$$

The minimum value of $E(A, B)$ is determined by setting the partial derivatives $\partial E/\partial A$ and $\partial E/\partial B$ equal to zero:

$$\frac{\partial E}{\partial A} = 2\sum_{k=1}^{N} (Ax_k^2 + Bx_k - x_k y_k) = 0$$
$$\frac{\partial E}{\partial B} = 2\sum_{k=1}^{N} (Ax_k + B - y_k) = 0$$

This gives the following equation system for the coefficients A and B

$$A\sum_{k=1}^{N} x_k^2 + B\sum_{k=1}^{N} x_k = \sum_{k=1}^{N} x_k y_k$$
$$A\sum_{k=1}^{N} x_k + BN = \sum_{k=1}^{N} y_k$$

Solution of the equation system provides coefficients of the least-squares line.

Curve Fitting

Data linearization method. Suppose that we are given the points (x_1, y_1), (x_2, y_2), $\ldots$,(x_N, y_N) and we want to fit an exponential curve of the form

$$y = Ce^{Ax}$$

The first step is to take the logarithm of both sides:

$$\ln y = Ax + \ln C$$

Then we introduce the change of variables:

$$Y = \ln y, \qquad X = x, \qquad B = \ln C$$

This results in a linear relation between the new variables X and Y:

$$Y = AX + B$$

Now we perform linear least-square fit for the coefficients A and B as shown above. After A and B have been found, the parameter C is computed $C = e^B$.

The technique of data linearization has been used in science for a long time for different curve fitting. Once the formula for curve fitting is chosen, a suitable transformation of the variables must be found so that a linear relation is obtained. For example, the function $y = D/(x + C)$ is transformed into a linear problem $Y = AX + B$ by using the change of variables (and constants) $X = xy$, $Y = y$, $C = -1/A$ and $D = -B/A$.

Linear least squares. The linear list-squares problem is stated as follows. Suppose that N data points (x_1, y_1), (x_2, y_2), $\ldots$,(x_N, y_N) and a set of M linear independent functions $f_1(x)$, $\ldots$, $f_M(x)$ are given. We want to find coefficients c_1,..., c_M so that the function $f(x)$ given by the linear combination

$$f(x) = \sum_{j=1}^{M} c_j f_j(x)$$

will minimize the sum of squares of the errors

$$E(c_1, ..., c_M) = \sum_{k=1}^{N} (f(x_k) - y_k)^2 = \sum_{k=1}^{N} (\sum_{j=1}^{M} c_j f_j(x_k) - y_k)^2$$

For E to be minimized, it is necessary that each partial derivative be zero ($\partial E/\partial c_i = 0$, $i = 1, 2, ..., M$). This results in the system of equations:

$$\sum_{k=1}^{N} \left[\sum_{j=1}^{M} c_j f_j(x_k) - y_k \right] f_i(x_k) = 0, \quad i = 1, 2, ..., M$$

After interchange of the order of summation we obtain the system of M linear equations with M unknowns coefficients c_j

$$\sum_{j=1}^{M} \left[\sum_{k=1}^{N} f_i(x_k) f_j(x_k) \right] c_j = \sum_{k=1}^{N} f_i(x_k) y_k \ , \quad i = 1, 2, ..., M$$

Polynomial fitting. Let us apply the above linear least-squares method to polynomial fitting. In this case we can select the functions as polynomial terms:

$$f_j(x) = x^j, \quad j = 0, \ 1, \ 2, ..., M$$

Then the function $f(x)$ will be a polynomial of the degree M.

$$f(x) = c_0 + c_1 x + c_2 x^2 + ... + c_M x^M$$

Using general equation for linear least squares we arrive at the following equation system for the coefficients of a polynomial, which provides best fit to data (x_1, y_1), (x_2, y_2), $\ldots$,(x_N, y_N):

$$c_M \sum_{k=1}^{N} x_k^{2M} + c_{M-1} \sum_{k=1}^{N} x_k^{2M-1} + ... + c_0 \sum_{k=1}^{N} x_k^M = \sum_{k=1}^{N} y_k x_k^M$$
$$c_M \sum_{k=1}^{N} x_k^{2M-1} + c_{M-1} \sum_{k=1}^{N} x_k^{2M-2} + ... + c_0 \sum_{k=1}^{N} x_k^{M-1} = \sum_{k=1}^{N} y_k x_k^{M-1}$$
$$...$$
$$c_M \sum_{k=1}^{N} x_k^M + c_{M-1} \sum_{k=1}^{N} x_k^{M-1} + ... + c_0 N = \sum_{k=1}^{N} y_k$$