4.12. NUMERICAL DIFFERENTIATION

Suppose on the interval $[a,b]$, there is a discrete set of mismatched points x_i, $i=0,...,n$, $x_0=a$, $x_n=b$ (interpolation nodes), at which the values of a certain function $y(x)$, generating the table $y_i=y(x_i)$, $i=0,...,n$, are known. It is required to determine the value $y^{(l)}(x_j)$ of the derivative of the function of order l at one of these points x_j, $0\le j\le n$. The problem of numerical differentiation arises when finding the derivatives of either a table function or an analytic function, the direct differentiation of which is difficult for some reason. An important application of numerical differentiation is the difference approximation of derivatives, which is widely used in the numerical solution of ordinary differential equations and partial differential equations.

Definition. Let's recall that the *derivative* $y'(x)$ of the function of one variable $y(x)$ at point x is the limit of the ratio of the function increment to the argument increment, when the argument increment tends to zero:

$$y'(x)=\lim_{h\to 0}\frac{y(x+h)-y(x)}{h}.$$

Then, for the table $y_i=y(x_i)$, $i=0,...,n$ in the first approximation, the first derivative of the function $y(x)$ at point x_j, $0\le j\le n$ can be defined as follows:

$$y'(x_j)=\frac{y_{j+1}-y_j}{x_{j+1}-x_j}.$$

Similarly, at point x_{j+1}

$$y'(x_{j+1})=\frac{y_{j+2}-y_{j+1}}{x_{j+2}-x_{j+1}}.$$

According to the same definition, for the second derivative, it is true that

$$y''(x)=\lim_{h\to 0}\frac{y'(x+h)-y'(x)}{h}.$$

Then, for the table $y_i=y(x_i)$, $i=0,...,n$ in the first approximation, the second derivative of the function $y(x)$ at point x_j, $0\le j\le n$ can be defined as follows:

$$y''(x_j) = \frac{y'(x_{j+1}) - y'(x_j)}{x_{j+1} - x_j} = \frac{\dfrac{y_{j+2} - y_{j+1}}{x_{j+2} - x_{j+1}} - \dfrac{y_{j+1} - y_j}{x_{j+1} - x_j}}{x_{j+1} - x_j} .$$

Using such a procedure of differentiation by definition, it is also possible to derive formulas for derivatives of a higher order.

However, a more accurate and efficient approach for numerical differentiation is the use of the expansion of the function in Taylor series. Expanding the function $y(x)$ in Taylor series by powers of *h*, where *h* is small enough, we get:

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2} y''(x) + O(h^3) ,$$

$$y(x-h) = y(x) - hy'(x) + \frac{h^2}{2} y''(x) + O(h^3) .$$

Definition. The value of the order of *h* in the main term of its error is called the *order of accuracy* of the numerical differentiation formula.

For the table $y_i = y(x_i)$, $i = 0,...,n$, Taylor's formulas at point x_j will take the following form:

$$y_{j+1} = y_j + (x_{j+1} - x_j) y'(x_j) + \frac{(x_{j+1} - x_j)^2}{2} y''(x_j) ,$$

$$y_{j-1} = y_j - (x_j - x_{j-1}) y'(x_j) + \frac{(x_j - x_{j-1})^2}{2} y''(x_j) .$$

Then, dropping the square terms and expressing $y'(x_j)$ one by one from each expansion in a series, we obtain for the table $y_i = y(x_i)$, $i = 0,...,n$ two formulas for the first derivatives of the first order of accuracy at point x_j (by two points):

$$y'(x_j) = \frac{y_{j+1} - y_j}{x_{j+1} - x_j} , \; y'(x_j) = \frac{y_j - y_{j-1}}{x_j - x_{j-1}} .$$

Next, solving both formulas with respect to $y'(x_j)$ and $y''(x_j)$, we obtain for the table $y_i = y(x_i)$, $i = 0,...,n$ expressions for the first derivative of the second order of accuracy and the second derivative of the first order of accuracy at point x_j (by three points):

$$y'(x_j)=\frac{(y_{j+1}-y_j)(x_j-x_{j-1})^2-(y_{j-1}-y_j)(x_{j+1}-x_j)^2}{(x_j-x_{j-1})(x_{j+1}-x_j)(x_{j+1}-x_{j-1})},$$

$$y''(x_j)=\frac{2(y_{j+1}-y_j)(x_j-x_{j-1})+2(y_{j-1}-y_j)(x_{j+1}-x_j)}{(x_j-x_{j-1})(x_{j+1}-x_j)(x_{j+1}-x_{j-1})}.$$

If we assume that all nodes x_i, $i=0,...,n$ are equally spaced and introduce the notation $h=x_i-x_{i-1}$, $i=1,...,n$, then these formulas will take the following form:

$$y'(x_j)=\frac{y_{j+1}-y_j}{h},\ y'(x_j)=\frac{y_j-y_{j-1}}{h},$$

$$y'(x_j)=\frac{y_{j+1}-y_{j-1}}{2h},\ y''(x_j)=\frac{y_{j+1}-2y_j+y_{j-1}}{h^2}.$$

These formulas are called non-difference formulas of numerical differentiation. In the theory of difference schemes, the first derivative at point x_j calculated using y_{j+1} is called a one-sided right (or right-hand) derivative or a forward derivative; the first derivative calculated using y_{j-1} – a one-sided left (or left-hand) derivative or backward derivative; and the first derivative calculated using both y_{j+1} and y_{j-1} is called the central or symmetric derivative. The second derivative at the boundary points x_{j-1} and x_{j+1} is approximated with the first order of accuracy, and in the central point x_j – with the second one.

Continuing the procedure of expanding the function $y(x)$ in Taylor series to large degrees at different points of the table $y_i=y(x_i)$, $i=0,...,n$ and solving the arising system of linear algebraic equations, it is also possible to derive formulas for derivatives of a higher order.

An alternative way to derive the necessary formulas for numerical differentiation is to use polynomial interpolation.

The original function $y_i=y(x_i)$, $i=0,...,n$ on the selected interval $[x_j,x_{j+k}]$ is replaced by a certain approximating, easily calculated function $\varphi(x,a)$, $y(x)=\varphi(x,a)+R(x)$, where $R(x)$ is the remainder of the approximation, a is the vector of parameters, and it is assumed that $y^{(l)}(x)\approx\varphi^{(l)}(x,a)$. The most studied and widespread case is when an interpolation polynomial of degree $m\le k$

$\varphi(x,a) = P_m(x) = \sum_{j=0}^{m} a_j x^j$ is taken as the approximating function $\varphi(x,a)$ and the derivative of the required order $l \le m$ is determined by differentiating this polynomial. Herewith, it is usually convenient to use an interpolation polynomial in the Newtonian form to derive formulas for numerical differentiation, since it contains divided differences that are analogous to derivatives of the corresponding orders. Obviously, this approach is also applicable, when it is necessary to find the derivative at a certain point within a selected interval $[x_j, x_{j+k}]$ that does not coincide with any of the nodes.

It should be noted that the numerical differentiation procedure is incorrect in the sense that the proximity of the original function $y(x)$ and the approximating function $\varphi(x,a)$ does not guarantee the proximity of their derivatives.

Obviously, the minimum number of nodes for obtaining the *l*-th derivative is equal to $l+1$ and $k = m = l$, since further differentiation of the polynomial leads to the derivative being equal to zero. Herewith, the accuracy of calculating the derivative at a given degree *m* of the interpolation polynomial decreases as far as the order of the derivative *l* increases. When solving practical problems, as a rule, we have to use approximations of the first and second derivatives. The derivatives of higher orders have to be approximated significantly more rarely.

To demonstrate the approach to numerical differentiation using polynomial interpolation, let's present the derivation of the corresponding formulas for numerical differentiation using two, three, and four points for each interpolation node.

In the first approximation, i.e. when an interpolation polynomial of the first degree is used for the approximation, the table function can be approximated by straight-line segments. Let's take a table function at two nodes $y_0 = y(x_0)$, $y_1 = y(x_1)$ and construct a Newton interpolation polynomial:

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

After differentiation of the polynomial, we get:

$$y'(x) \approx \varphi'(x) = \frac{y_1 - y_0}{x_1 - x_0}.$$

If we introduce notation $h = x_1 - x_0$, then this formula will take the following form:

$$y'(x) \approx \varphi'(x) = \frac{y_1 - y_0}{h}.$$

In the second approximation, an interpolation polynomial of the second degree is used to approximate the table function. Let's take a table function at three nodes $y_0 = y(x_0)$, $y_1 = y(x_1)$, $y_2 = y(x_2)$ and construct a Newton interpolation polynomial:

$$y(x) \approx \varphi(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x - x_0)(x - x_1).$$

After differentiation of the polynomial, we get:

$$y'(x) \approx \varphi'(x) = \frac{y_1 - y_0}{x_1 - x_0} + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}\big((x - x_1) + (x - x_0)\big),$$

$$y''(x) \approx \varphi''(x) = 2\frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}.$$

If we assume that three nodes are equally spaced and introduce notation $h = x_2 - x_1 = x_1 - x_0$, then these formulas will take the following form:

$$y'(x) \approx \varphi'(x) = \frac{y_1 - y_0}{h} + \frac{y_2 - 2y_1 + y_0}{2h^2}\big((x - x_1) + (x - x_0)\big),$$

$$y''(x) \approx \varphi''(x) = \frac{y_2 - 2y_1 + y_0}{h^2}.$$

Herewith, at points x_0, x_1, x_2:

$$y'(x_0) \approx \varphi'(x_0) = \frac{-y_2 + 4y_1 - 3y_0}{2h},$$

$$y'(x_1) \approx \varphi'(x_1) = \frac{y_2 - y_0}{2h},$$

$$y'(x_2) \approx \varphi'(x_2) = \frac{3y_2 - 4y_1 + y_0}{2h}.$$

In the third approximation, an interpolation polynomial of the third degree is used to approximate the table function. Let's take a table function at four nodes $y_0 = y(x_0)$, $y_1 = y(x_1)$, $y_2 = y(x_2)$, $y_3 = y(x_3)$ and construct a Newton interpolation polynomial:

$$y(x) \approx \varphi(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}(x - x_0)(x - x_1) +$$

$$+ \frac{\frac{\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}}{x_3 - x_0}(x - x_0)(x - x_1)(x - x_2).$$

After differentiation of the polynomial, we get:

$$y'(x) \approx \varphi'(x) = \frac{y_1 - y_0}{x_1 - x_0} + \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}\big((x - x_1) + (x - x_0)\big) +$$

$$+ \frac{\frac{\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}}{x_3 - x_0}\big((x - x_1)(x - x_2) + (x - x_0)(x - x_2) + (x - x_0)(x - x_1)\big),$$

$$y''(x) \approx \varphi''(x) = 2\frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} +$$

$$+2\frac{\frac{\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}}{x_3 - x_0}\big((x - x_0) + (x - x_1) + (x - x_2)\big).$$

$$y'''(x) \approx \varphi'''(x) = \frac{6}{x_3 - x_0} \cdot \left(\frac{\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1} - \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}\right).$$

If we assume that four nodes are equally spaced and introduce notation $h = x_3 - x_2 = x_2 - x_1 = x_1 - x_0$, then these formulas will take the following form:

$$y'(x) \approx \varphi'(x) = \frac{y_1 - y_0}{h} + \frac{y_2 - 2y_1 + y_0}{2h^2}\big((x - x_1) + (x - x_0)\big) +$$

$$+ \frac{y_3 - 3y_2 + 3y_1 - y_0}{6h^3}\big((x - x_1)(x - x_2) + (x - x_0)(x - x_2) + (x - x_0)(x - x_1)\big),$$

$$y''(x) \approx \varphi''(x) = \frac{y_2 - 2y_1 + y_0}{h^2} + \frac{y_3 - 3y_2 + 3y_1 - y_0}{3h^3}\big((x - x_0) + (x - x_1) + (x - x_2)\big),$$

$$y'''(x) \approx \varphi'''(x) = \frac{y_3 - 3y_2 + 3y_1 - y_0}{h^3}.$$

Herewith, at points x_0, x_1, x_2, x_3:

$$y'(x_0) \approx \varphi'(x_0) = \frac{2y_3 - 9y_2 + 18y_1 - 11y_0}{6h}, \quad y''(x_0) \approx \varphi''(x_0) = \frac{-y_3 + 4y_2 - 5y_1 + 2y_0}{h^2},$$

$$y'(x_1) \approx \varphi'(x_1) = \frac{-y_3 + 6y_2 - 3y_1 - 2y_0}{6h}, \quad y''(x_1) \approx \varphi''(x_1) = \frac{y_2 - 2y_1 + y_0}{h^2},$$

$$y'(x_2) \approx \varphi'(x_2) = \frac{2y_3 + 3y_2 - 6y_1 + y_0}{6h}, \quad y''(x_2) \approx \varphi''(x_2) = \frac{y_3 - 2y_2 + y_1}{h^2},$$

$$y'(x_3) \approx \varphi'(x_3) = \frac{11y_3 - 18y_2 + 9y_1 - 2y_0}{6h}, \quad y''(x_3) \approx \varphi''(x_3) = \frac{2y_3 - 5y_2 + 4y_1 - y_0}{h^2}.$$

It is easy to see that the obtained formulas have obvious mutual symmetry regarding the direction of nodes numbering and, moreover, the sum of the coefficients in the numerator of each formula is always equal to zero.

Example. Using a table of values of the function $y(x)$ (values y_i calculated at points x_i, $i = 0,...,n$), calculate numerically the first and second derivatives of the function at point x^*. Calculate the value of the interpolation error of the first and second derivatives at point x^*.

$$y = x\ln(x),\ n = 4,\ i = 0,...,4,\ x_i = 0.1, 0.5, 0.9, 1.3, 1.7,\ x^* = 0.9.$$

Let's note that all nodes x_i, $i = 0,...,n$ are equally spaced and let's introduce the notation $h = x_i - x_{i-1} = 0.4$, $i = 1,...,n$.

To begin with, let's calculate for subsequent comparison the exact value of the first and second derivatives of the function $y(x)$ at point $x^* = 0.9$:

$$y' = \ln(x) + 1,\ y'(0.9) = \ln(0.9) + 1 = 0.8946.$$

$$y'' = \frac{1}{x},\ y''(0.9) = \frac{1}{0.9} = 1.1111.$$

Then, let's calculate numerically the first and second derivatives using the non-difference formulas and the interval $[x_1, x_3]$, as the point, at which it is required to find the value of the derivatives, coincides with x_2.

Right-hand derivative:

$$y'(0.9) = \frac{y_3 - y_2}{h} = \frac{0.3411 - (-0.0948)}{0.4} = 1.0897 .$$

Left-hand derivative:

$$y'(0.9) = \frac{y_2 - y_1}{h} = \frac{(-0.0948) - (-0.3466)}{0.4} = 0.6294 .$$

Central derivative:

$$y'(0.9) = \frac{y_3 - y_1}{2h} = \frac{0.3411 - (-0.3466)}{0.4} = 0.8596 .$$

Let's note that the result of calculating the central derivative coincides with the half-sum of the left-hand and right-hand derivatives.

Second derivative:

$$y''(0.9) = \frac{y_3 - 2y_2 + y_1}{h^2} = \frac{0.3411 - 2\cdot(-0.0948) + (-0.3466)}{0.4^2} = 1.1509 .$$

So, the absolute error of interpolation is:

$$\Delta(y'(x^*)) = \left|0.8596 - 0.8946)\right| = 0.0351.$$

$$\Delta(y''(x^*)) = \left|1.1509 - 1.1111)\right| = 0.0398 .$$

Now, for a more accurate calculation of the derivatives, let's use polynomial interpolation on the entire interval $[x_0, x_4]$. Let's construct a Newton polynomial of the 4th degree on the whole table of values: $N_4(x) = \sum_{j=0}^{4} \Delta_{0\ldots j}^{j} \prod_{i=0}^{j-1} (x - x_i)$.

$$N_4(x) = \Delta_0^0 + \Delta_{01}^1 (x - x_0) + \Delta_{012}^2 (x - x_0)(x - x_1) + \Delta_{0123}^3 (x - x_0)(x - x_1)(x - x_2) +$$

$$+\Delta_{01234}^4 (x - x_0)(x - x_1)(x - x_2)(x - x_3) .$$

After substituting the values of the coordinates of points (x_i, y_i) and calculating the coefficients (all necessary divided differences), the desired Newton polynomial can be written as follows:

$$N_4(x) = -0.2303 - 0.2908\cdot(x - 0.1) + 1.1502\cdot(x - 0.1)(x - 0.5) -$$

$$-0.4789\cdot(x - 0.1)(x - 0.5)(x - 0.9) + 0.2032\cdot(x - 0.1)(x - 0.5)(x - 0.9)(x - 1.3) .$$

Let's differentiate the polynomial $N_4(x)$ and calculate the value $N_4'(x^*)$ of the derivative of the interpolation polynomial at point $x^* = 0.9$:

$$N_4'(x) = -0.2908 + 1.1502\cdot(x - 0.5) + 1.1502\cdot(x - 0.1) -$$

$-0.4789\cdot(x-0.5)(x-0.9)-0.4789\cdot(x-0.1)(x-0.9)-0.4789\cdot(x-0.1)(x-0.5)+$

$+0.2032\cdot(x-0.5)(x-0.9)(x-1.3)+0.2032\cdot(x-0.1)(x-0.9)(x-1.3)+$

$+0.2032\cdot(x-0.1)(x-0.5)(x-1.3)+0.2032\cdot(x-0.1)(x-0.5)(x-0.9)$,

$N_4'(0.9)=0.9102$.

Let's differentiate the polynomial $N_4'(x)$ and calculate the value $N_4''(x^*)$ of the second derivative of the interpolation polynomial at point $x^*=0.9$:

$N_4''(x)=1.1502+1.1502-0.4789\cdot(x-0.9)-0.4789\cdot(x-0.5)-$

$-0.4789\cdot(x-0.9)-0.4789\cdot(x-0.1)-0.4789\cdot(x-0.5)-0.4789\cdot(x-0.1)+$

$+0.2032\cdot(x-0.9)(x-1.3)+0.2032\cdot(x-0.5)(x-1.3)+0.2032\cdot(x-0.5)(x-0.9)+$

$+0.2032\cdot(x-0.9)(x-1.3)+0.2032\cdot(x-0.1)(x-1.3)+0.2032\cdot(x-0.1)(x-0.9)+$

$+0.2032\cdot(x-0.5)(x-1.3)+0.2032\cdot(x-0.1)(x-1.3)+0.2032\cdot(x-0.1)(x-0.5)+$

$+0.2032\cdot(x-0.5)(x-0.9)+0.2032\cdot(x-0.1)(x-0.9)+0.2032\cdot(x-0.1)(x-0.5)$,

$N_4''(0.9)=1.0859$.

So, the absolute error of interpolation is:

$\Delta(y'(x^*))=\left|0.9102-0.8946)\right|=0.0155$.

$\Delta(y''(x^*))=\left|1.0859-1.1111)\right|=0.0252$.

It is easy to see that the use of all points of the table of values of the function $y(x)$ for the numerical calculation of the first and second derivatives allows getting the answer with greater accuracy.

Calculation results are given in table 12.

Table 12

i	x_i	$\Delta_i^0=y_i$	$\Delta^1_{i(i+1)}$	$\Delta^2_{i(i+1)(i+2)}$	$\Delta^3_{i(i+1)(i+2)(i+3)}$	$\Delta^4_{i(i+1)(i+2)(i+3)(i+4)}$
0	0.1	**-0.2303**	**-0.2908**	**1.1502**	**-0.4789**	**0.2032**
1	0.5	-0.3466	0.6294	0.5755	-0.1538	
2	0.9	-0.0948	1.0897	0.3909		
3	1.3	0.3411	1.4025			
4	1.7	0.9021				

4.13. PROGRAM #16

Below is a proposed variant of the program algorithm for calculating the values of the first and second derivatives of a function at a given point using a Newton interpolation polynomial on the entire table of function values.

```
ALGORITHM “Numerical differentiation”
INPUT     n, x[n], y[n], u
OUTPUT    i, j, k, d[n][n], f, g, v, w
BEGIN
    #Table of divided differences
    CYCLE “Rows” FOR i FROM 1 TO n BY 1
        d[i][1]:=y[i]
    CYCLE “Columns” FOR j FROM 2 TO n BY 1
        CYCLE “Rows” FOR i FROM 1 TO n-j+1 BY 1
            d[i][j]:=(d[i][j-1]-d[i+1][j-1])/(x[i]-x[i+j-1])
    #First derivative
    v:=0
    CYCLE “Summands” FOR j FROM 2 TO n BY 1
        f:=0
        CYCLE “Summands” FOR k FROM 1 TO j-1 BY 1
            g:=d[1][j]
            CYCLE “Multipliers” FOR i FROM 1 TO j-1 BY 1
                IF (i≠k) g:=g*(u-x[i])
            f:=f+g
        v:=v+f
    #Second derivative
    w:=0
    CYCLE “Summands” FOR j FROM 3 TO n BY 1
        f:=0
        CYCLE “Summands” FOR k FROM 1 TO (j-1)*(j-2) BY 1
            g:=d[1][j]
```

```
                        CYCLE “Multipliers” FOR i FROM 1 TO j-1 BY 1
                              IF (i≠1+DIV(k-1,j-2))AND
(i≠1+REMAINDER(k+DIV(k-1,j-1),j-1)) g:=g*(u-x[i])
                        f:=f+g
                  w:=w+f
            PRINT v, w
      END
```