Calcul Numeric

Cursul 9

2022

Anca Ignat

Interpolare numerică

Presupunem că despre o funcţie $\boldsymbol{f}$ cunoaştem doar valorile într-un număr finit de puncte. Pornind de la aceste date, dorim să aproximăm funcţia $\boldsymbol{f}$ într-un alt punct.

x	x_0	x_1	x_2	...	x_{n-1}	x_n
f	y_0	y_1	y_2	...	y_{n-1}	y_n

În tabelul de mai sus $f(x_i) = y_i$, $i=0,1,\ldots,n$ şi $x_i \neq x_j$, $i \neq j$.

Dat un punct $x \neq x_i$, $i=0,1,\ldots,n$ dorim să aproximăm $f(x)$ cunoscând cele $(n+1)$ perechi (x_i, y_i), $i=0,\ldots,n$. Punctele x_i se numesc ***noduri de interpolare***.

Polinomul de interpolare Lagrange

Notăm cu Π_n mulțimea polinoamelor de grad cel mult n. Dimensiunea acestui spațiu este $n+1$, baza uzuală fiind dată de polinoamele $1, x, x^2, \ldots, x^n$. Vom considera o altă bază în acest spațiu. Se consideră polinoamele p_i:

$$p_i \in \Pi_n \text{ astfel ca } p_i(x_j) = \begin{cases} 0 & \text{pentru } j \neq i \\ 1 & \text{pentru } j = i \end{cases}, \; j = 0,\ldots,n, \; i = 0,\ldots,n$$

Din relația $\boldsymbol{p_i(x_j)=0}$, $\forall\, \boldsymbol{j \neq i}$ şi faptul că $\boldsymbol{p_i}$ este de grad $\boldsymbol{n}$ rezultă că $\boldsymbol{x_0, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n}$ sunt cele $\boldsymbol{n}$ rădăcini ale polinomului $\boldsymbol{p_i}$.

Avem:

$$\boldsymbol{p_i(x) = c_i(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n),}$$

$$\boldsymbol{c_i \in \mathbb{R},\ i = 0,\ldots,n}$$

Constanta $\boldsymbol{c_i}$ se determină din relaţia $\boldsymbol{p_i(x_i) = 1}$:

$$\boldsymbol{p_i(x_i) = 1 = c_i(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n) \Rightarrow}$$

$$\boldsymbol{c_i = \frac{1}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}}$$

Polinoamele $\boldsymbol{p_i}$ au forma:

$$\boldsymbol{p_i(x) = \frac{(x - x_0)\cdots(x - x_{i-1})(x - x_{i+1})\cdots(x - x_n)}{(x_i - x_0)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)} = \prod_{\substack{j=0\\ j\neq i}}^{n}\left(\frac{x - x_j}{x_i - x_j}\right)}$$

$$\boldsymbol{i = 0,\ldots,n}$$

Propoziţie

Polinoamele $\boldsymbol{p_0, p_1, \ldots, p_n}$ formează o bază în $\boldsymbol{\Pi_n}$.

Demonstraţie: Vom arăta că cele $\boldsymbol{n+1}$ polinoame sunt liniar independente:

$$q(x) = a_0 p_0(x) + a_1 p_1(x) + \cdots + a_n p_n(x) = 0 \text{ , } \forall x$$

$$\Rightarrow a_0 = a_1 = \cdots = a_n = 0$$

Vom face pe rând $x = x_0, x = x_1, \ldots, x = x_n$ în polinomul q:

$$x = x_0 \quad q(x_0) = a_0 p_0(x_0) + a_1 p_1(x_0) + \cdots + a_n p_n(x_0) =$$

$$= a_0 1 + a_1 0 + \cdots + a_n 0 = a_0 = 0 \quad \Rightarrow a_0 = 0$$

$$x = x_1 \quad q(x_1) = 0 \quad \Rightarrow \quad a_1 = 0$$

$$\vdots$$

$$x = x_k \quad q(x_k) = a_0 p_0(x_k) + \cdots + a_k p_k(x_k) + \cdots + a_n p_n(x_k) =$$

$$= a_0 0 + \cdots + a_k 1 + \cdots + a_n 0 = a_k = 0 \Rightarrow a_k = 0$$

$$\vdots$$

$$x = x_n \quad q(x_n) = 0 \quad \Rightarrow \quad a_n = 0$$

Toate constantele a_i sunt nule deci polinoamele $\{p_i; i = 0,\ldots,n\}$ formează o bază în Π_n.

Pentru a aproxima funcţia ***f*** pornind de la tabelul de mai sus, vom construi un polinom $l_n \in \Pi_n$ a.î. să satisfacă ***condiţiile de interpolare***:

$$l_n \in \Pi_n \quad , \quad l_n(x_i) = y_i \quad , \quad \forall i = 0,\ldots,n \tag{1}$$

Odată construit acest polinom, vom aproxima ***f(x)*** prin ***l***$l_n(x)$, $f(x) \approx l_n(x)$

Vom scrie polinomul l_n în raport cu noua bază $\{p_i; i=0,\dots,n\}$, deci există constantele reale a_0, $a_1,\dots,a_n$ astfel ca:

$$l_n(x)=\sum_{i=0}^{n} a_i p_i(x)$$

Constantele a_k se determină astfel:

$$y_k = l_n(x_k) = a_0 p_0(x_k) + \cdots + a_k p_k(x_k) + \cdots + a_n p_n(x_k) =$$

$$= a_0 0 + \cdots + a_k 1 + \cdots + a_n 0 = a_k \;\; \Rightarrow \;\; a_k = y_k$$

Prin urmare un polinom de grad n care îndeplinesc condiţiile de interpolare (1) este:

$$l_n(x) = \sum_{i=0}^{n} y_i p_i(x) = \sum_{i=0}^{n} y_i \frac{(x - x_0)\cdots(x - x_{i-1})(x - x_{i+1})\cdots(x - x_n)}{(x_i - x_0)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)}$$

$$= \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j \neq i}}^{n} \left(\frac{x - x_j}{x_i - x_j}\right) \qquad \textbf{(2)}$$

Polinomul din formula (2) se numeşte ***polinomul de interpolare Lagrange***.

Propoziţie

Polinomul $\boldsymbol{l_n}$ dat de formula (2) este unicul polinom de grad $\boldsymbol{n}$ care îndeplineşte condiţiile de interpolare (1).

Demonstraţie: Presupunem că mai există un polinom $q \in \Pi_n$ care îndeplineşte condiţiile (1):

$$q \in \Pi_n \quad , \; q(x_i) = y_i \quad , \quad \forall i = 0,\ldots,n$$

Fie polinomul $p(x)=l_n(x)-q(x) \in \Pi_n$.

$$p(x_k) = l_n(x_k) - q(x_k) = y_k - y_k = 0 \;\; , \forall k = 0,\ldots,n$$

Polinomul p are ca rădăcini toate nodurile de interpolare. Polinomul p este polinom de grad cel mult n şi are $(n+1)$ rădăcini distincte ($x_i \neq x_j, \forall \; i \neq j$). Acest polinom nu poate fi decât polinomul identic nul:

$$p(x) = l_n(x) - q(x) \equiv 0 \quad \forall x\,, \quad l_n(x) = q(x) \quad \forall x$$

Polinomul $\boldsymbol{l_n}$ este unicul care satisface (2).

Fie $\boldsymbol{w_{n+1}}$ polinomul de grand ***(n+1)*** care are ca rădăcini nodurile de interpolare:

$$w_{n+1}(x) = (x - x_0)(x - x_1)\cdots(x - x_n) \in \Pi_{n+1}$$

Fie $\boldsymbol{a = \min\{x_0, x_1,\ldots, x_n\}}$, $\boldsymbol{b = \max\{x_0, x_1,\ldots, x_n\}}$.

Teorema restului (eroarea la interpolarea Lagrange)

Fie $f \in C^{n+1}[a,b]$ şi $\bar{x} \in [a,b],\ \bar{x} \neq x_i\ , \forall i = 0,\ldots,n.$

Atunci există un punct $y \in [a,b],\ y = y(x_0, x_1, \ldots, x_n, \bar{x})$ (punctul y depinde de nodurile de interpolare x_i şi de punctul $\bar{x}$) astfel că eroarea la interpolarea numerică este dată de:

$$f(\bar{x}) - l_n(\bar{x}) = \frac{f^{(n+1)}(y)}{(n+1)!} w_{n+1}(\bar{x}) \qquad (3)$$

Demonstraţie: Considerăm funcţia F:

$$F(x) := f(x) - l_n(x) - c w_{n+1}(x)$$

Constanta reală c este aleasă astfel ca $F(\bar{x}) = 0$ adică:

$$c = \frac{f(\bar{x}) - l_n(\bar{x})}{w_{n+1}(\bar{x})}, \ (x \neq x_i \ \ \forall i) \Rightarrow \ w_{n+1}(\bar{x}) \neq 0 \ ) \qquad (4)$$

Funcţia f fiind de clasă C^{n+1} pe intervalul $[a,b]$ rezultă că şi funcţia F este din $C^{n+1}[a,b]$. Avem:

$$F(x_i) = f(x_i) - l_n(x_i) - cw_{n+1}(x_i) = y_i - y_i - c\,0 = 0 \quad , \forall i = 0,\ldots,n$$

Funcţia F are *(n+2)* zerouri, $x_0, x_1, \ldots, x_n, \bar{x}$. Aplicând succesiv Teorema lui Rolle rezultă că F' are *(n+1)* zerouri, F'' are n zerouri,…, $F^{(n+1)}$ are *1* zero în intervalul *[a,b]*. Vom nota această rădăcină a lui $F^{(n+1)}$ cu y. Punctul y depinde de zerourile iniţiale $x_0, x_1, \ldots, x_n, \bar{x}$ şi:

$$y = y(x_0, x_1, \ldots, x_n, \bar{x}) \in [a,b] \quad \text{a.î.} \quad F^{(n+1)}(y) = 0. \tag{5}$$

Derivata de ordinul *(n+1)* a funcţiei F se calculează astfel:

$$F^{(n+1)}(x) = f^{(n+1)}(x) - l_n^{(n+1)}(x) - c\, w_{n+1}^{(n+1)}(x) =$$

$$= f^{(n+1)}(x) - 0 - c(n+1)! = f^{(n+1)}(x) - c(n+1)!$$

(6)

(derivata de ordin ***(n+1)*** a polinomului de grad ***n*** ***l_n*** este ***0***).
Din relaţiile (4), (5) şi (6) rezultă că:

$$c=\frac{f^{(n+1)}(y)}{(n+1)!}=\frac{f(\overline{x})-l_n(\overline{x})}{w_{n+1}(\overline{x})}\Rightarrow f(\overline{x})-l_n(\overline{x})=\frac{f^{(n+1)}(y)}{(n+1)!}w_{n+1}(\overline{x})$$

Forma Newton a polinomului de interpolare Lagrange

Fie $l_k(x, x_0, x_1,\ldots, x_k, f)$ polinomul de interpolare Lagrange pentru funcţia ***f*** pe sistemul de noduri distincte $\{x_0, x_1,\ldots, x_k\}$.

Propoziţie

Fie $\boldsymbol{l_{k-1}(x, x_0, x_1,\ldots, x_{k-1}, f), l_{k-1}(x, x_1, x_2,\ldots, x_k, f)\in\Pi_{k-1}}$ polinoamele de interpolare Lagrange pentru funcţia $\boldsymbol{f}$ pe sistemele de noduri $\boldsymbol{\{x_0, x_1,\ldots, x_{k-1}\}}$ şi respectiv $\boldsymbol{\{x_1, x_2,\ldots, x_k\}}$. Atunci:

$$l_k(x, x_0, x_1, \ldots, x_k, f) = \\ = \frac{(x-x_0)l_{k-1}(x, x_1, x_2, \ldots, x_k, f) - (x-x_k)l_{k-1}(x, x_0, x_1, \ldots, x_{k-1}, f)}{x_k - x_0} \tag{1}$$

Demonstraţie: Exerciţiu.

Considerăm următoarele probleme de interpolare pentru $\boldsymbol{f}$:

$$\{(x_0, y_0),(x_1, y_1),\ldots,(x_{k-1}, y_{k-1})\} \;\rightarrow\; l_{k-1}(x, x_0, x_1, \ldots, x_{k-1}, f)$$

$$\{(x_0, y_0),(x_1, y_1),\ldots,(x_k, y_k)\} \quad\;\rightarrow\; l_k(x, x_0, x_1, \ldots, x_k, f)$$

Ne interesează să găsim o formulă de trecere rapidă de la polinomul de interpolare pe ***k*** noduri la cel care are un nod în plus. Deoarece polinomul de grad cel mult ***k***:

$$q(x) = l_k(x, x_0, x_1, \ldots, x_k, f) - l_{k-1}(x, x_0, x_1, \ldots, x_{k-1}, f) \in \Pi_k$$

are ca rădăcini punctele $x_0, x_1, \ldots, x_{k-1}$ (***q(***x_i***) = y***$_i$ ***- y***$_i$ ***= 0, i=0,...,k-1***) avem relaţia:

$$l_k(x, x_0, x_1, \ldots, x_k, f) = l_{k-1}(x, x_0, x_1, \ldots, x_{k-1}, f) + A\prod_{j=0}^{k-1}(x - x_j) \quad (2)$$

în care A este dat de relaţia:

$$A = \frac{l_k(x_k, x_0, x_1, \ldots, x_k, f) - l_{k-1}(x_k, x_0, x_1, \ldots, x_{k-1}, f)}{\prod_{j=0}^{k-1}(x_k - x_j)} \quad (3)$$

$$A = \frac{y_k}{\prod_{j=0}^{k-1}(x_k - x_j)} - \frac{\sum_{i=0}^{k-1} y_i \prod_{\substack{j=0 \\ j\neq i}}^{k-1}\left(\frac{x_k - x_j}{x_i - x_j}\right)}{\prod_{j=0}^{k-1}(x_k - x_j)} =$$

$$= \frac{y_k}{\prod_{j=0}^{k-1}(x_k - x_j)} - \sum_{i=0}^{k-1} \frac{y_i}{(x_k - x_i)\prod_{\substack{j=0 \\ j\neq i}}^{k-1}(x_i - x_j)}$$

$$A = \sum_{i=0}^{k} \frac{y_i}{\prod\limits_{\substack{j=0 \\ j\neq i}}^{k} (x_i - x_j)} \qquad (4)$$

$$\left[x_k\right]_f = y_k = f(x_k) \ ,$$

$$\left[x_0, x_1, \ldots, x_k\right]_f = \frac{\left[x_1, x_2, \ldots, x_k\right]_f - \left[x_0, x_1, \ldots, x_{k-1}\right]_f}{x_k - x_0}$$

numită ***diferenţă divizată de ordin k a funcţiei f pe nodurile*** $\left\{x_0, x_1, \ldots, x_k\right\}$.

Propoziție

$$\left[x_0, x_1, \ldots, x_k\right]_f = \sum_{i=0}^{k} \frac{y_i}{\prod\limits_{\substack{j=0 \\ j\neq i}}^{k} (x_i - x_j)} = \sum_{i=0}^{k} \frac{y_i}{\left(w_{n+1}(x_k)\right)'}$$

pentru orice sistem de noduri $\left\{x_0, x_1, \ldots, x_k\right\}$ şi orice $\boldsymbol{k}$.

Demonstraţie: Se face prin inducţie. Pentru $\boldsymbol{k=1}$ avem:

$$\left[x_0, x_1\right]_f = \frac{y_0}{x_0 - x_1} + \frac{y_1}{x_1 - x_0} = \frac{\left[x_1\right]_f - \left[x_0\right]_f}{x_1 - x_0}$$

Presupunem că relaţia (8) este valabilă pentru orice $\boldsymbol{k}$ şi pentru orice sistem de noduri $\left\{x_0, x_1, \ldots, x_k\right\}$. Pentru $\boldsymbol{k+1}$ folosim relaţia de recurenţă şi apoi aplicăm ipoteza inductivă:

$$\left[x_0, x_1, \ldots, x_{k+1}\right]_f = \frac{\left[x_1, x_1, \ldots, x_{k+1}\right]_f - \left[x_0, x_2, \ldots, x_k\right]_f}{x_{k+1} - x_0} =$$

$$= \frac{1}{x_{k+1} - x_0} \left(\sum_{i=1}^{k+1} \frac{y_i}{\prod_{\substack{j=1 \\ j \neq i}}^{k+1} (x_i - x_j)} - \sum_{i=0}^{k} \frac{y_i}{\prod_{\substack{j=0 \\ j \neq i}}^{k} (x_i - x_j)} \right) =$$

$$= \frac{1}{x_{k+1} - x_0} \left\{ - \frac{y_0}{\prod_{\substack{j=0 \\ j \neq 0}}^{k} (x_0 - x_j)} + \frac{y_{k+1}}{\prod_{\substack{j=1 \\ j \neq k+1}}^{k+1} (x_{k+1} - x_j)} + \right.$$

$$\left. + \sum_{i=1}^{k} \left[\frac{y_i}{\prod_{\substack{j=1 \\ j \neq i}}^{k} (x_i - x_j)} \left(\frac{1}{x_i - x_{k+1}} - \frac{1}{x_i - x_0} \right) \right] \right\} =$$

$$= \frac{y_0}{\prod_{\substack{j=0 \\ j\neq 0}}^{k+1}(x_0 - x_j)} + \frac{y_{k+1}}{\prod_{\substack{j=0 \\ j\neq k+1}}^{k+1}(x_{k+1} - x_j)} + \sum_{i=1}^{k} \frac{y_i}{\prod_{\substack{j=0 \\ j\neq i}}^{k+1}(x_i - x_j)} =$$

$$= \sum_{i=0}^{k+1} \frac{y_i}{\prod_{\substack{j=0 \\ j\neq i}}^{k+1}(x_i - x_j)}$$

Inducţia este completă.

Din definiţie se observă că diferenţa divizată $\left[x_0, x_1, \ldots, x_k\right]_f$ nu depinde de ordinea nodurilor $\left\{x_0, x_1, \ldots, x_k\right\}$.

Vom nota în continuare cu ***lk(x)*** polinomul de interpolare Lagrange pe nodurile $\left\{x_0, x_1, \ldots, x_k\right\}$ pentru funcţia ***f***. Avem:

$$
\begin{aligned}
l_n(x) &= l_0(x) + [l_1(x) - l_0(x)] + \cdots + [l_k(x) - l_{k-1}(x)] + \cdots + [l_n(x) - l_{n-1}(x)] = \\
&= y_0 + \left[x_0, x_1\right]_f (x - x_0) + \cdots + \left[x_0, x_1, \ldots, x_k\right]_f (x - x_0)\cdots(x - x_{k-1}) + \cdots \\
&\quad + \left[x_0, x_1, \ldots, x_n\right]_f (x - x_0)\cdots(x - x_{n-1})
\end{aligned}
$$

Am obţinut ***forma Newton*** a polinomului de interpolare Lagrange:

$$l_n(x) = y_0 + [x_0, x_1]_f (x - x_0) + [x_0, x_1, x_2]_f (x - x_0)(x - x_1) + \cdots +$$
$$+ [x_0, x_1, \ldots, x_n]_f (x - x_0) \cdots (x - x_{n-1})$$

Schema lui Aitken de calcul a diferențelor divizate

Ne propunem să calculăm diferențele divizate

$$\left[x_0, x_1\right]_f, \left[x_0, x_1, x_2\right]_f \text{ , ... , } \left[x_0, x_1, \ldots, x_n\right]_f$$

necesare construirii polinomului de interpolare Lagrange în forma Newton. Procedeul folosește definiția recursivă a diferențelor divizate și se desfășoară în ***n*** pași. La pasul ***1*** se calculează numai diferențe divizate de ordinul ***1***:

$$\left[x_0, x_1\right]_f, \left[x_1, x_2\right]_f, \ldots, \left[x_{n-1}, x_n\right]_f .$$

În general, la pasul $\boldsymbol{k}$ se calculează diferențe divizate de ordin $\boldsymbol{k}$:

$$\left[x_0, x_1, \ldots, x_k\right]_f, \left[x_1, x_2, \ldots, x_{k+1}\right]_f, \ldots, \left[x_{n-k}, x_{n-k+1}, \ldots, x_n\right]_f.$$

La pasul $\boldsymbol{n}$ se calculează o singură diferență divizată de ordin $\boldsymbol{n}$ și anume $\left[x_0, x_1, \ldots, x_n\right]_f$.

$$
\begin{array}{lllllll}
 & & \textbf{Pas } 1 & \cdots & \textbf{Pas } k & \cdots & \textbf{Pas } n \\
x_0 & y_0 & & & & & \\
x_1 & y_1 & [x_0, x_1]_f & & & & \\
x_2 & y_2 & [x_1, x_2]_f & & & & \\
\vdots & & & & & & \\
x_k & y_k & [x_{k-1}, x_k]_f & & [x_0, x_1, \ldots, x_k]_f & & \\
\vdots & & & & \vdots & \ddots & \\
x_{n-1} & y_{n-1} & [x_{n-2}, x_{n-1}]_f & & [x_{n-k-1}, \ldots, x_{n-1}]_f & & \\
x_n & y_n & [x_{n-1}, x_n]_f & & [x_{n-k}, \ldots, x_{n-1}]_f & \cdots & [x_0, x_1 \ldots, x_n]_f
\end{array}
$$

Notăm **dd[i,k]=**$\left[x_i, x_{i+1}, \ldots, x_{i+k}\right]_f$ diferenţa divizată de ordin ***k***, pe nodurile consecutive $\left\{x_i, x_{i+1}, \ldots, x_{i+k}\right\}$ ***i=0,…,n-k, k=1,…,n,*** cu ***dd[i,0]=y***$_i$***,*** ***i=0,…,n***. Schema lui Aitken se implementează astfel:

$$dd[i,0] = y_i,\ \ i = 0,\ldots,n;$$

$$\textbf{for } k = 1,\ldots,n$$

$$\quad\textbf{for } i = 0,\ldots,n-k$$

$$\qquad dd[i,k] = \frac{dd[i+1,k-1] - dd[i,k-1]}{x_{i+k} - x_i}$$

Putem face aceleași calcule folosind un singur vector, de exemplu rescriind vectorul $\boldsymbol{y}$ astfel:

$$\textbf{for } k = 1,\ldots,n$$
$$\textbf{for } i = n,\ldots,k$$
$$y_i = \frac{y_i - y_{i-1}}{x_i - x_{i-k}}$$

La finalul acestei secvențe de program, vectorul $\boldsymbol{y}$ va conține elementele:

$$y_0, \left[x_0, x_1\right]_f, \left[x_0, x_1, x_2\right]_f, \ldots, \left[x_0, x_1, \ldots, x_n\right]_f$$
$$(y_k = \left[x_0, x_1, \ldots, x_k\right]_f,\ k=0,\ldots,n).$$

Interpolare Newton pe noduri echidistante

Pp. că nodurile de interpolare sunt echidistante:

$$x_i = x_0 + i\,h \quad , \quad i = 0,1,\ldots,n$$

În relația de mai sus fie se dă h distanța între 2 noduri succesive, fie se precizează primul și ultimul nod, x_0 și x_n iar h se calculează:

$$h = \frac{(x_n - x_0)}{n}\,.$$

$$\left[x_i, x_{i+1}\right]_f = \frac{f(x_{i+1}) - f(x_i)}{(x_{i+1} - x_i)} = \frac{y_{i+1} - y_i}{h}$$

Se introduce noțiunea de **diferență finită de ordinul 1:**

$$\Delta f(x) = f(x+h) - f(x)$$

Pornind de la această definiție se pot introduce și diferențe finite de ordin superior:

$$\Delta^2 f(x) = \Delta(\Delta f(x)) = \Delta(f(x+h) - f(x)) =$$

$$= f(x+2h) - 2f(x+h) + f(x)$$

și în general se pot introduce recursiv **diferențele finite de ordin** ***k***:

$$\Delta^k f(x) = \Delta(\Delta^{k-1} f(x)) = \Delta^{k-1} f(x+h) - \Delta^{k-1} f(x).$$

Prin inducție după ***k*,** se poate deduce formula de calcul a diferențelor finite de ordin ***k*** folosind doar valorile funcției ***f***:

$$\Delta^k f(x) = \sum_{i=0}^{k} (-1)^{k-i} C_k^i \; f(x + i\,h)$$
.

Observație: Dacă funcția ***f*** este polinom de grad ***m*** atunci $\Delta f(x)$ este polinom de grad ***m-1***, $\Delta^2 f(x)$ este polinom de grad ***m-2***, ş.a.m.d. Prin urmare:

$$\Delta^k f(x) \equiv 0 \text{ , pentru } k > m \text{ , } f - \text{polinom de grad } m.$$

Legătura între diferențele divizate şi cele finite:

$$\left[x_i, x_{i+1}\right]_f = \frac{f(x_{i+1}) - f(x_i)}{(x_{i+1} - x_i)} = \frac{\Delta f(x_i)}{h}$$

$$\left[x_i, x_{i+1}, x_{i+2}\right]_f = \frac{\left[x_{i+1}, x_{i+2}\right]_f - \left[x_i, x_{i+1}\right]_f}{(x_{i+2} - x_i)} = \frac{\Delta^2 f(x_i)}{2h^2}$$

Prin inducție se poate arăta următoarea legătură între diferențele divizate de ordin ***k*** şi cele finite:

$$\left[x_i, x_{i+1}, \ldots, x_{i+k}\right]_f = \frac{\Delta^k f(x_i)}{k!h^k}.$$

Polinoame de interpolare pe noduri echidistante

$$l_n(x) = y_0 + \left[x_0, x_1\right]_f (x - x_0) + \left[x_0, x_1, x_2\right]_f (x - x_0)(x - x_1) + \cdots +$$

$$+ \left[x_0, x_1, ..., x_k\right]_f (x - x_0)(x - x_1)\cdots(x - x_{k-1}) + \cdots +$$

$$+ \left[x_0, x_1, ..., x_n\right]_f (x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

Consideră că punctul de interpolare este de forma:

$$\bar{x} = x_0 + t\,h$$

și înlocuim diferențele divizate cu diferențe finite în forma Newton a polinomului de interpolare:

$$(\bar{x} - x_0)\cdots(\bar{x} - x_{k-1}) = (x_0 + t\,h - x_0)\cdots(x_0 + t\,h - x_0 - (k-1)h) =$$

$$= h^k t(t-1)\cdots(t-k+1)$$

$$l_n(\overline{x}) = l_n(x_0 + th) = y_0 + \Delta f(x_0)t + \Delta^2 f(x_0)\frac{t(t-1)}{2} + \cdots +$$

$$+ \Delta^k f(x_0)\frac{t(t-1)\cdots(t-k+1)}{k!} + \cdots +$$

$$+ \Delta^n f(x_0)\frac{t(t-1)\cdots(t-n+1)}{n!}$$

Această relație poartă numele de **formula lui Newton progresivă pe noduri echidistante.**

Considerăm polinomul de interpolare Lagrange pe nodurile în ordine inversă $\{x_n, x_{n-1}, ..., x_0\}$:

$$l_n(x) = y_n + \left[x_n, x_{n-1}\right]_f (x - x_n) + \left[x_n, x_{n-1}, x_{n-2}\right]_f (x - x_n)(x - x_{n-1}) +$$
$$+ \cdots + \left[x_n, x_{n-1}, ..., x_0\right]_f (x - x_n)(x - x_{n-1}) \cdots (x - x_0)$$

Dacă punctul de interpolare este de forma:

$$\overline{x} = x_n + th$$

analog ca mai sus obține **formula lui Newton regresivă pe noduri echidistante:**

$$
\begin{aligned}
l_n(\overline{x}) = l_n(x_n + th) = \; & y_n + \Delta f(x_{n-1})t + \Delta^2 f(x_{n-2})\frac{t(t+1)}{2} + \cdots + \\
& + \Delta^k f(x_{n-k})\frac{t(t+1)\cdots(t+k-1)}{k!} + \cdots + \\
& + \Delta^n f(x_0)\frac{t(t+1)\cdots(t+n-1)}{n!}
\end{aligned}
$$

Funcții spline

Fie nodurile:

$$x_i \in [a,b],\ i = 0,1,\ldots,n\,,$$

cu

$$a = x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n = b$$

Se consideră funcția continuă polinomială pe porțiuni:

$$S(x) = P_i(x) \ \ \textbf{pentru} \ \ x \in [x_i, x_{i+1}] \quad \forall\, i = 0,\ldots,n-1$$

$$S(x)=\begin{cases} P_0(x), & x\in[x_0,x_1], \\ P_1(x), & x\in[x_1,x_2], \\ P_2(x), & x\in[x_2,x_3], \\ \vdots & \\ P_{n-2}(x), & x\in[x_{n-2},x_{n-1}], \\ P_{n-1}(x), & x\in[x_{n-1},x_n]. \end{cases}$$

$P_i(x)$, $i=0,...,n$ sunt polinoame. O asemenea funcție poartă numele de ***funcție spline***.

Funcții spline liniare continue

Definiție

Funcția ***S(x)*** definită mai sus se numește ***funcție spline liniară continuă*** dacă polinoamele $P_i(x), i = 0, \dots, n-1$ sunt polinoame de gradul ***1*** și ***S(x)*** $\in$ ***C[a,b]***, adică:

$$\lim_{\substack{x \to x_i \\ x < x_i}} S(x) = \lim_{\substack{x \to x_i \\ x > x_i}} S(x), \; i = 1, \dots, n-1.$$

Fie funcția $f : [a,b] \to \mathbb{R}$ pentru care se cunosc valorile:

$$y_i = f(x_i), i = 0, \dots, n.$$

Funcția spline liniară de interpolare S pentru funcţia f îndeplinește condițiile de interpolare:

$$S(x_i) = y_i \ , i = 0,\dots,n.$$

Ținând seamă că polinoamele $P_i(x)$ sunt polinoame de gradul *1* și $S(x)$ este continuă vom avea condițiile:

$$\begin{cases} P_i(x_i) = y_i \ , \\ P_i(x_{i+1}) = y_{i+1}, \qquad i = 0,\dots,n-1, \\ P_i(x) - \text{polinom de gradul } 1. \end{cases}$$

Din aceste condiții rezultă:

$$P_i(x) = \frac{x - x_i}{x_{i+1} - x_i} y_{i+1} + \frac{x_{i+1} - x}{x_{i+1} - x_i} y_i \ , i = 0,\dots,n-1$$

$$S(x_k) = P_{k-1}(x_k) = P_k(x_k) = y_k \ , k = 1,\ldots,n-1,$$
$$S(x_0) = P_0(x_0) = y_0 \ , S(x_n) = P_{n-1}(x_n) = y_n \ .$$

Funcții spline cubice de clasă C^2

Se consideră sistemul de noduri distincte din intervalul ***[a,b]***:

$$\Delta = \{a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b\}$$

Funcția ***S(x)*** asociată divizării ***Δ*** care îndeplinește condițiile :

$$S(x) \in C^2[a,b],$$

polinoamele $P_i(x)$ **au gradul** 3 , $i = 0,\ldots,n-1,$

se numește ***funcție spline cubică***.

Dată fiind o funcție $\boldsymbol{f} :[\boldsymbol{a},\boldsymbol{b}] \to \mathbb{R}$ cu valorile:

$$\boldsymbol{y_i = f\left(x_i\right), i = 0,\dots,n},$$

se consideră funcția spline cubică ***S(x)*** de interpolare ce satisface

$$\boldsymbol{S(x_i) = y_i , i = 0,\dots,n.}$$

Pentru determinarea funcției spline cubice de interpolare observăm că polinoamele:

$$\boldsymbol{P_i(x) = \alpha_i x^3 + \beta_i x^2 + \gamma_i x + \delta_i , x \in [x_i , x_{i+1}], i = 0,\dots,n-1,}$$

implică determinarea a celor ***4n*** necunoscute $\boldsymbol{\{\alpha_i,\beta_i,\gamma_i,\delta_i ; i = 0,\dots,n-1\}}$ pentru care se impun:

$$\begin{cases} \boldsymbol{n+1} \textbf{ condiţii din relaţiile de interpolare } \boldsymbol{S(x_i) = y_i \;, i = 0,\ldots,n,} \\ \boldsymbol{3(n-1)} \textbf{ condiţii de continuitate pentru } \boldsymbol{S(x), S^{'}(x)} \textbf{ şi } \boldsymbol{S^{''}(x)} \\ \qquad\qquad \textbf{în nodurile } \boldsymbol{x_i \;, i = 1,\ldots,n-1\,,} \end{cases}$$

în total ***4n-2*** condiții.

Se pot avea în vedere pentru adăugarea a două condiții suplimentarea următoarele abordări :

- fixarea pantelor în extremitățile intervalului ***[a,b].*** Se presupune că funcția ***f*** este derivabilă şi se cunosc valorile ***f′(a), f′(b)***. Se impun condițiile:
  $$S'(x_0) = P_0'(x_0) = f'(a)\,,\; S'(x_n) = P_{n-1}'(x_n) = f'(b);$$
- periodicitatea primelor două derivate:

$$f'(a) = f'(b) \quad (S'(x_0) = P_0'(x_0) = P_{n-1}'(x_n) = S'(x_n)),$$

$$f''(a) = f''(b) \quad (S''(x_0) = P_0''(x_0) = P_{n-1}''(x_n) = S''(x_n));$$

- anularea derivatei secunde în capetele intervalului:

$$f''(a) = f''(b) = 0$$

$$(S''\left(x_0\right) = P_0''\left(x_0\right) = 0, S''\left(x_n\right) = P_{n-1}''\left(x_n\right) = 0).$$

Funcțiile spline care îndeplinesc aceste condiții se numesc ***funcții spline cubice normale***.

- derivata de ordinul al treilea a funcției S este continuă în punctele x_1 și x_{n-1} . Aceasta înseamnă că polinoamele P_0, P_1 respectiv P_{n-2}, P_{n-1} coincid. Acest tip de funcție spline se numește „*not – a – knot*” și este utilizat în MATLAB.

Ne vom ocupa în continuare de determinarea funcţiei spline de interpolare în cazul în care cunoaştem prima derivată a funcţiei $\boldsymbol{f}$ în capetele intervalului de interpolare**:** $\boldsymbol{f'(a),\ \ f'(b)}$.

Recapitulând, vom avea următoarele condiţii :

$$\begin{cases} \boldsymbol{P_i(x_i)=y_i\ ,\ i=0,\ldots,n-1,\ P_{n-1}(x_n)=y_n\ -\ \textbf{interpolare}\ ,} \\ \boldsymbol{P_{i-1}(x_i)=P_i(x_i)\ ,\ i=1,\ldots,n-1,-\ \textbf{continuitatea funcţiei}\ S\ ,} \\ \boldsymbol{P'_{i-1}(x_i)=P'_i(x_i)\ ,\ i=1,\ldots,n-1,-\ \textbf{continuitatea primei derivatei,}} \\ \boldsymbol{P''_{i-1}(x_i)=P''_i(x_i)\ ,\ i=1,\ldots,n-1,-\ \textbf{continuitatea derivatei secunde,}} \\ \boldsymbol{P'_0(x_0)=f'(a),\ \ P'_{n-1}(x_n)=f'(b).} \end{cases}$$

Vom nota:

$$S''(x_i) = a_i \ , i = 0,\ldots n.$$

Ţinând seama de faptul că funcţia ***S″∈C[a,b]*** este o funcţie liniară pe fiecare din intervalele $[x_i, x_{i+1}]$ rezultă că:

$$S''(x) = \frac{x - x_i}{h_i} a_{i+1} + \frac{x_{i+1} - x}{h_i} a_i \ , x \in [x_i, x_{i+1}], \quad \forall i = 0,\ldots,n-1$$

$$h_i = x_{i+1} - x_i \quad , \ i = 0,\ldots,n-1$$

iar din

$$S'(x) = \int S''(x)dx\,, \qquad S(x) = \int S'(x)dx$$

rezultă:

$$S'(x) = \frac{\left(x - x_i\right)^2}{2h_i} a_{i+1} - \frac{\left(x_{i+1} - x\right)^2}{2h_i} a_i + b_i\,,$$

$$x \in [x_i, x_{i+1}],\, b_i \in \mathbb{R}\,, i = 0, \ldots n-1,$$

$$S(x) = \frac{\left(x - x_i\right)^3}{6h_i} a_{i+1} + \frac{\left(x_{i+1} - x\right)^3}{6h_i} a_i + b_i x + c_i\,,$$

$$x \in [x_i, x_{i+1}],\, b_i, c_i \in \mathbb{R}\,, i = 0, \ldots n-1,$$

$$P_i(x)=\frac{\left(x-x_i\right)^3}{6h_i}a_{i+1}+\frac{\left(x_{i+1}-x\right)^3}{6h_i}a_i+b_i x+c_i\ ,$$

$$b_i, c_i \in \mathbb{R}\ , i=0,\ldots n-1,$$

Vom calcula funcţia spline pentru cazul:

$$S'(a)=P_0'(x_0)=f'(a),$$
$$S'(b)=P_{n-1}'(x_n)=f'(b).$$

Impunând condiţiile de interpolare şi de continuitate vom obţine:

$$P_i(x_i) = \frac{h_i^2}{6} a_i + b_i x_i + c_i = y_i \ ,$$

$$P_i(x_{i+1}) = \frac{h_i^2}{6} a_{i+1} + b_i x_{i+1} + c_i = y_{i+1} \ , \ \ i = 0,\ldots n-1.$$

Din aceste relaţii calculăm b_i şi c_i în funcţie de a_i , a_{i+1}, y_i , y_{i+1}:

$$b_i = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{6}\left(a_{i+1} - a_i\right),$$

$$c_i = \frac{x_{i+1} y_i - x_i y_{i+1}}{h_i} - \frac{h_i}{6}\left(x_{i+1} a_i - x_i a_{i+1}\right), i = 0,\ldots n-1.$$

Avem:

$$P_0'(x)=\frac{\left(x-x_0\right)^2}{2h_0}a_1-\frac{\left(x_1-x\right)^2}{2h_0}a_0+b_0$$

$$P_{n-1}'(x)=\frac{\left(x-x_{n-1}\right)^2}{2h_{n-1}}a_n-\frac{\left(x_n-x\right)^2}{2h_{n-1}}a_{n-1}+b_{n-1}$$

Din condiţia $S'(a)=P_0'(x_0)=f'(a)$ avem

$$P_0'(x_0)=-\frac{h_0}{2}a_0+b_0=-\frac{2h_0}{6}a_0-\frac{h_0}{6}a_1+\frac{y_1-y_0}{h_0}=f'(a)$$

$$2h_0 a_0 + h_0 a_1 = 6\left(\frac{y_1 - y_0}{h_0} - f'(a)\right) \qquad (1)$$

Din condiţia $S'(b) = P'_{n-1}(x_n) = f'(b)$ avem

$$P'_{n-1}(x_n) = \frac{h_{n-1}}{2}a_n + b_{n-1} = \frac{h_{n-1}}{6}a_{n-1} + \frac{h_{n-1}}{6}a_n + \frac{y_n - y_{n-1}}{h_{n-1}} = f'(b)$$

$$h_{n-1} a_{n-1} + 2h_{n-1} a_n = 6\left(f'(b) - \frac{y_n - y_{n-1}}{h_{n-1}}\right) \qquad (2)$$

Din condiţia de continuitate a primei derivate a funcţiei spline cubice $P'_{i-1}(x_i) = P'_i\left(x_i\right)$, $i = 1,...,n-1$, ţinând seama de:

$$P'_{i-1}(x) = \frac{\left(x - x_{i-1}\right)^2}{2h_{i-1}} a_i - \frac{\left(x_i - x\right)^2}{2h_{i-1}} a_{i-1} + b_{i-1},$$

$$P'_i(x) = \frac{\left(x - x_i\right)^2}{2h_i} a_{i+1} - \frac{\left(x_{i+1} - x\right)^2}{2h_i} a_i + b_i,$$

rezultă, utilizând formulele pentru b_{i-1} şi b_i deduse mai sus:

$$P'_{i-1}(x_i) = \frac{h_{i-1}}{2} a_i + \frac{y_i - y_{i-1}}{h_{i-1}} - \frac{h_{i-1}}{6}\left(a_i - a_{i-1}\right) =$$

$$P'_i(x_i) = -\frac{h_i}{2} a_i + \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{6}\left(a_{i+1} - a_i\right)$$

sau

$$h_{i-1}a_{i-1} + 2(h_{i-1} + h_i)\, a_i + h_i\, a_{i+1} = 6\left(\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}}\right),\ \ i = 1,\ldots,n-1.$$

(3)

Sistemul liniar format din ecuaţiile (1), (3), (2) cu necunoscutele $\{\boldsymbol{a}_0, \boldsymbol{a}_1, \cdots, \boldsymbol{a}_n\}$ are forma:

$$\boldsymbol{Ha} = \boldsymbol{f} \text{, cu } \boldsymbol{H} \in \mathbb{R}^{(n+1)\times(n+1)}, \ \boldsymbol{f} \in \mathbb{R}^{n+1}$$

$$2h_0 a_0 + h_0 a_1 = 6\left(\frac{y_1 - y_0}{h_0} - f'(a) \right)$$

$$h_{i-1} a_{i-1} + 2(h_{i-1} + h_i) a_i + h_i a_{i+1} = 6\left(\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}} \right)$$

$$i = 1, \ldots, n-1$$

$$h_{n-1} a_{n-1} + 2h_{n-1} a_n = 6\left(f'(b) - \frac{y_n - y_{n-1}}{h_{n-1}} \right)$$

$$
\boldsymbol{H} = \begin{bmatrix}
2h_0 & h_0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
h_0 & 2(h_0+h_1) & h_1 & 0 & \cdots & 0 & 0 & 0 \\
0 & h_1 & 2(h_1+h_2) & h_2 & \cdots & 0 & 0 & 0 \\
\vdots & & & & & & & \\
0 & 0 & 0 & 0 & \cdots & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\
0 & 0 & 0 & 0 & \cdots & 0 & h_{n-1} & 2h_{n-1}
\end{bmatrix}
$$

$$\boldsymbol{f} = \begin{bmatrix} 6\left(\dfrac{y_1 - y_0}{h_0} - f'(a)\right) \\ 6\left(\dfrac{y_{i+1} - y_i}{h_i} - \dfrac{y_i - y_{i-1}}{h_{i-1}}\right) \; i = 1,\ldots,n-1 \\ 6\left(f'(b) - \dfrac{y_n - y_{n-1}}{h_{n-1}}\right) \end{bmatrix}$$

Matricea ***H*** are diagonala dominantă atât pe linii cât şi pe coloane, este simetrică şi pozitiv definită prin urmare putem utiliza metoda Gauss-Seidel sau o metodă de relaxare pentru rezolvarea sistemului ***Ha=f***.

Interpolare în sensul celor mai mici pătrate

x	x_0	x_1	x_2	...	x_{n-1}	x_n
f	y_0	y_1	y_2	...	y_{n-1}	y_n

$$f(x_i) = y_i \;, \; i=0,\ldots,n$$

$$f(x) \approx S_f(x; a_0, a_1, \ldots, a_m)$$

$$S_f(x; a_0, a_1, \ldots, a_m) = a_m x^m + a_{m-1}x^{m-1} + \ldots + a_1 x + a_0$$

Coeficienții $a_0, a_1, \ldots, a_m$ se găsesc rezolvând problema de minimizare în sensul celor mai mici pătrate:

$$\min\left\{ \sum_{r=0}^{n}\Big(S_f(x_r;a_0,a_1,...,a_m)-y_r \Big)^2 ;\ a_0,a_1,...,a_m \in \mathbb{R} \right\} \quad \text{(LSP)}$$

$$g:\mathbb{R}^{m+1}\to\mathbb{R}_+ \ ,$$

$$g(a_0,a_1,...,a_m)=\sum_{r=0}^{n}\Big(S_f(x_r;a_0,a_1,...,a_m)-y_r \Big)^2$$

$$g(a_0,a_1,...,a_m)=\sum_{r=0}^{n}\Big(a_m x_r^m+\cdots+a_k x_r^k+\cdots+a_1 x_r+a_0-y_r \Big)^2$$

$$\frac{\partial g}{\partial a_k}(a_0,a_1,...,a_m)=2\sum_{r=0}^{n}\Big(a_m x_r^m+\cdots+a_k x_r^k+\cdots+a_1 x_r+a_0-y_r \Big)x_r^k$$

Soluția problemei de minimizare a problemei **(LSP)** este obținută rezolvând sistemul de ecuații liniare, de dimensiune ***(m+1)***:

$$\frac{\partial g}{\partial a_k}(a_0, a_1, ..., a_m) = 0 \ , \quad k = 0, 1, ..., m$$

$$\sum_{r=0}^{n}\left(a_m x_r^m + \cdots + a_k x_r^k + \cdots + a_1 x_r + a_0\right) x_r^k = \sum_{r=0}^{n} y_r x_r^k \ , \ k = 0, ..., m$$

$$a_0 \sum_{r=0}^{n} x_r^k + a_1 \sum_{r=0}^{n} x_r^{k+1} + \cdots + a_{m-1} \sum_{r=0}^{n} x_r^{k+m-1} + a_m \sum_{r=0}^{n} x_r^{k+m} = \sum_{r=0}^{n} y_r x_r^k \ ,$$

$$k = 0, ..., m$$

Constantele $\{ a_0, a_1, ..., a_m \}$ sunt soluția sistemului liniar:

$$Ba = z \ ,$$

$$B \in \mathbb{R}^{(m+1)\times(m+1)} \ , \ B = (b_{kj})_{k,j=0}^{m} \ , \ z \in \mathbb{R}^{(m+1)} \ z = (z_k)_{k=0}^{m}$$

$$b_{kj} = \sum_{r=0}^{n} x_r^{k+j} \quad , \quad z_k = \sum_{r=0}^{n} y_r x_r^k \ , \qquad k, j = 0, ..., m$$