EILCO : Analyse Numérique
Chapitre 1 : Interpolation
H. Sadok

2015-2016

Plan

Bibliographie

- A. Quarteroni, F. Saleri et P. Gervasio, Calcul Scientifique , Springer-Verlag France, Paris, 2010.
- S. Guerre-Delabrière et M. Postel, «Méthodes d'approximation, Equations différentielles, Applications Scilab», Ellipses, Paris, 2004.
- M. Crouzeix et A. L. Mignot, « Analyse Numérique des Equations Différentielles », Masson, Paris, 1983.

Position du Problème

On se donne le tableau de données suivant

i	x_i	y_i
0	x_0	y_0
$\vdots$	$\vdots$	$\vdots$
n	x_n	y_n

Définition

On cherche un polyôme P_n de degré au plus n ($P_n \in \mathcal{P}_n$)tel que

$$P_n(x_i) = y_i, \quad \text{pour } i = 0, \ldots, n.$$

Existence

Soit $P_n \in \mathcal{P}_n$ alors

$$P_n(t) = a_0 + a_1 t + \ldots + a_n t^n$$

et donc $P_n(x_i) = y_i, \quad$ pour $i = 0, \ldots, n$ si et seulement si

$$\left\{ \begin{array}{l} a_0 + a_1 x_0 + \ldots\ldots\ldots\ldots\ldots + a_n x_0^n = y_0 \\ a_0 + a_1 x_1 + \ldots\ldots\ldots\ldots\ldots + a_n x_1^n = y_1 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_0 + a_1 x_n + \ldots\ldots\ldots\ldots\ldots + a_n x_n^n = y_n \end{array} \right.$$

On obtient un système d'équations linéaires dont le vecteur inconnu est $(a_0, \ldots, a_n)$.

Résolution du système linéaire

Le système précédent s'écrit sous forme matricielle sous la forme :

$$\begin{pmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \vdots & \vdots & \dots & \vdots \\ 1 & x_n & \dots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

On obtient un système d'équations linéaires dont la matrice est la matrice de Vandermonde et dont le vecteur inconnu est $(a_0, \dots, a_n)$.

$$P_n(t) = (a_0, a_1, \dots, a_n) \begin{pmatrix} 1 \\ t \\ \vdots \end{pmatrix}$$

Thérème d'existence

Déterminant de Vandermonde

Le déterminant de Vandermonde vérifie :

$$det(V_{n+1}(x_0, \cdots, x_n)) = \begin{vmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \dots & x_n^n \end{vmatrix} = \prod_{0 \leq i < j \leq n} (x_j - x_i).$$

Théorème

Une condition nécéssaire et suffisante pour qu'il existe un et un seul polynome $P_n \in \mathcal{P}_n$ tel que $P_n(x_i) = y_i, \quad$ pour $i = 0, \dots, n$ est que toutes les abscisses soient distinctes.

Preuve de l'unicité

- On suppose que $P_n(x_i) = Q_n(x_i) = y_i, \quad$ pour $i = 0, \dots, n$,
- $P_n - Q_n \in \mathcal{P}_n$,
- $(P_n - Q_n)(x_i) = 0, \quad$ pour $i = 0, \dots, n$,
- $P_n - Q_n$ est un polynme de degré au plus n qui admet $n + 1$ racines.
- conclusion : $P_n \equiv Q_n$.

définition des polynômes de Lagrange

Nous avons obtenu :

$$P_n(t) = (a_0, a_1, \ldots, a_n) \begin{pmatrix} 1 \\ t \\ \vdots \\ t^n \end{pmatrix}$$

avec

$$\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} 1 & x_0 & \ldots & x_0^n \\ 1 & x_1 & \ldots & x_1^n \\ \vdots & \vdots & \ldots & \vdots \\ 1 & x_n & \ldots & x_n^n \end{pmatrix}^{-1} \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

définition des polynômes de Lagrange (suite)

En transposant la dernière équation on obtient

$$(a_0, a_1, \ldots, a_n) = (y_0, y_1, \ldots, y_n) \begin{pmatrix} 1 & 1 & \ldots & 1 \\ x_0 & x_1 & \ldots & x_n \\ \vdots & \vdots & \ldots & \vdots \\ x_0^n & x_1^n & \ldots & x_n^n \end{pmatrix}^{-1}$$

Et donc

$$P_n(t) = (y_0, y_1, \ldots, y_n) \begin{pmatrix} 1 & 1 & \ldots & 1 \\ x_0 & x_1 & \ldots & x_n \\ \vdots & \vdots & \ldots & \vdots \\ x_0^n & x_1^n & \ldots & x_n^n \end{pmatrix}^{-1} \begin{pmatrix} 1 \\ t \\ \vdots \\ t^n \end{pmatrix}$$

Construction des polynômes de Lagrange (suite)

Le polynôme d'interpolation peut donc s'écrire :

$$P_n(t) = (y_0, y_1, \dots, y_n) \begin{pmatrix} L_0(t) \\ L_1(t) \\ \vdots \\ L_n(t) \end{pmatrix}$$

avec

$$\begin{pmatrix} 1 & 1 & \dots & 1 \\ x_0 & x_1 & \dots & x_n \\ \vdots & \vdots & \dots & \vdots \\ x_0^n & x_1^n & \dots & x_n^n \end{pmatrix} \begin{pmatrix} L_0(t) \\ L_1(t) \\ \vdots \\ L_n(t) \end{pmatrix} = \begin{pmatrix} 1 \\ t \\ \vdots \\ t^n \end{pmatrix}$$

Charactérisation

Propriétés

$$\left\{ \begin{array}{c} L_0(t) + L_1(t) + \ldots + L_n(t) = 1 \\ x_0 L_0(t) + x_1 L_1(t) + \ldots + x_n L_n(t) = t \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_0^n L_0(t) + x_1^n L_1(t) + \ldots + x_n^n L_n(t) = t^n \end{array} \right.$$

Formules explicites pour $i = 0, \ldots, n$

$$L_i(t) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{t - x_j}{x_i - x_j}.$$

$$L_i(x_k) = \delta_{i,k} = \left\{ \begin{array}{ll} 1 & \text{si } i = k \\ 0 & \text{si } i \neq k \end{array} \right.$$

Ecriture du polynôme d'interpolation dans la base de Lagrange

Théorème

$\{L_0, \ldots, L_n\}$ est une base de $\mathcal{P}_n$

Polynôme d'interpolation dans la base de Lagrange

$$P_n(t) = \sum_{i=0}^{n} y_i \, L_i(t) = \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{t - x_j}{x_i - x_j}.$$

Formule Barycentrique de Lagrange

définition

Soit Π le polynome de degré $n+1$ tel que

$$\Pi(t) = (t - x_0)(t - x_1)\cdots(t - x_n).$$

Il est clair que

$$\Pi_i(t) \equiv \frac{\Pi(t) - \Pi(x_i)}{t - x_i} = \prod_{\substack{j=0\\ j\neq i}}^{n} t - x_j,$$

et que $\Pi_i(x_i) = \Pi'(x_i)$. On en déduit que

$$L_i(t) = \frac{\Pi(t)}{(t - x_i)\Pi_i(x_i)}.$$

Formule Barycentrique de Lagrange (suite)

Décomposition en élements simples

$$\frac{1}{\Pi(t)} = \sum_{i=0}^{n} \frac{\lambda_i^{(n)}}{(t - x_i)}.$$

On remarque que $\lambda_i^{(n)} = \dfrac{1}{\Pi_i(x_i)}$, ce qui donne

Formule Barycentrique

$$P_n(t) = \sum_{i=0}^{n} y_i \frac{\Pi(t)}{(t - x_i)\Pi_i(x_i)} = \frac{\sum_{i=0}^{n} y_i \frac{\lambda_i^{(n)}}{t-x_i}}{\sum_{i=0}^{n} \frac{\lambda_i^{(n)}}{t-x_i}}.$$

Algorihme

- $\lambda_0^{(0)} = 1$
- pour $j = 1, \dots, n$ faire
  - pour $k = 0, \dots, j-1$ faire
    - $\lambda_k^{(j)} = (x_k - x_j)\lambda_k^{(j-1)}$
  - fin du pour
  - $\lambda_j^{(j)} = \prod_{k=0}^{j-1}(x_k - x_j)$
- fin du pour
- pour $j = 0, \dots, n$ faire
  - $\lambda_j^{(j)} = \frac{1}{\lambda_j^{(j)}}$

Ecriture du polynôme d'interpolation dans la base de Newton

Théorème

Posons $N_0(t) = 1$ et $N_i(t) = N_{i-1}(t)(t - x_{i-1}) = \prod_{j=0}^{i-1}(t - x_j)$, alors $\{N_0, \dots, N_n\}$ est une base de $\mathcal{P}_n$

Polynôme d'interpolation dans la base de Newton

$$P_n(t) = \sum_{i=0}^{n} [x_0, \dots, x_i]\, N_i(t).$$

Différences divisées

Définition

On appelle diffrence divisée :

- d'ordre zero la quantitée : $[x_i] \equiv y_i$ et
- d'ordre un la quantitée : $[x_i, x_j] \equiv \frac{y_j - y_i}{x_j - x_i}$
- d'ordre k-1

$$[x_{i_1}, \dots, x_{i_k}] \equiv \frac{[x_{i_2}, \dots, x_{i_k}] - [x_{i_1}, \dots, x_{i_{k-1}}]}{x_{i_k} - x_{i_1}}$$

Propriétés

$$[x_0, \dots, x_k] = \sum_{i=0}^{k} \frac{y_i}{\prod_{j=0}^{k} (x_i - x_j)}$$

Différences Divisées

$$
\begin{array}{cccccc}
x_0 & [x_0] & & & & \\
 & & [x_0, x_1] & & & \\
x_1 & [x_1] & & [x_0, x_1, x_2] & & \\
 & & [x_1, x_2] & & [x_0, x_1, x_2, x_3] & \\
x_2 & [x_2] & & [x_1, x_2, x_3] & & [x_0, x_1, x_2, x_3, x_4] \\
 & & [x_2, x_3] & & [x_1, x_2, x_3, x_4] & \\
x_3 & [x_3] & & [x_2, x_3, x_4] & & \\
 & & [x_3, x_4] & & & \\
x_4 & [x_4] & & & &
\end{array}
$$

Calcul du Polynôme d'interpolation

pour $i = 0, \ldots, 4$,

$$P_i(x_j) = y_j, \text{ pour } j = 0, \ldots, i.$$

$$P_0(x) = [x_0]$$

$$P_1(x) = [x_0] + (x - x_0)[x_0, x_1]$$

$$P_2(x) = [x_0] + (x - x_0)[x_0, x_1] + (x - x_0)(x - x_1)[x_0, x_1, x_2]$$

$$P_3(x) = P_2(x) + (x - x_0)(x - x_1)(x - x_2)[x_0, x_1, x_2, x_3]$$

$$P_4(x) = P_3(x) + (x - x_0)(x - x_1)(x - x_2)(x - x_3)[x_0, x_1, x_2, x_3, x_4]$$

Exemple Numérique : Polynôme de Lagrange

-1	0						
		0					
$-\frac{2}{3}$	0		0				
		0		$\frac{9}{2}$			
$-\frac{1}{3}$	0		$\frac{9}{2}$		$-\frac{27}{2}$		
		3		$-\frac{27}{2}$		$\frac{81}{4}$	
0	1		-9		$\frac{81}{4}$		$-\frac{81}{4}$
		-3		$\frac{27}{2}$		$-\frac{81}{4}$	
$\frac{1}{3}$	0		$\frac{9}{2}$		$-\frac{27}{2}$		
		0		$-\frac{9}{2}$			
$\frac{2}{3}$	0		0				
		0					
1	0						

Le Polynôme de Lagrange

Les points d'interpolations sont
$x_0 = -1, x_1 = -\frac{2}{3}, x_2 = -\frac{1}{3}, x_3 = 0, x_4 = \frac{1}{3}, x_5 = \frac{2}{3}, x_6 = 1$

$$P(x) = \frac{1}{4}(4 - 49x^2 + 126x^4 - 81x^6)$$

$P(x) = \frac{9}{2}(\frac{1}{3}+x)(\frac{2}{3}+x)(1+x) - \frac{27}{2}x(\frac{1}{3}+x)(\frac{2}{3}+x)(1+x) + \frac{81}{4}(-\frac{1}{3}+x)x(\frac{1}{3}+x)(\frac{2}{3}+x)(1+x) - \frac{81}{4}(-\frac{2}{3}+x)(-\frac{1}{3}+x)x(\frac{1}{3}+x)(\frac{2}{3}+x)(1+x).$

Exemple

Considérons le tableau suivant :

i	x_i	y_i
0	x_0	x_0^{n+1}
$\vdots$	$\vdots$	$\vdots$
n	x_n	x_n^{n+1}

Le polynôme d'interpolation peut être donné explicitement :

$$P_n(x) = x^{n+1} - \prod_{i=0}^{n}(x - x_i).$$

Expression de l'erreur

Théorème

Soient f une fonction de classe $\mathcal{C}^{n+1}[a,b]$ avec $a = \min_k x_k$ et $b = \max_k x_k$ et P_n le polynôme interpolant f en $n+1$ points $x_0, \cdots, x_n$ appartenant à $[a,b]$ et $y_i = f(x_i)$. Alors $\exists \xi_t \in [\min(t, \min_k x_k), \max(t, \max_k x_k)]$ tel que :

$$E_n(t) = f(t) - p_n(t) = \frac{f^{(n+1)}(\xi_t)}{(n+1)!} \prod_{i=0}^{n} (t - x_i)$$

Théorème de Cauchy

$$E_n(t) = f(t) - p_n(t) = [x_0, \ldots, x_n, t] \prod_{i=0}^{n} (t - x_i)$$

Majoration de l'erreur : Cas d'abscisses équidistantes

Théorème

Soientt f une fonction de classe $\mathcal{C}^{n+1}[a,b]$ avec $a = \min_k x_k = x_0$ et $b = \max_k x_k = x_n$ et P_n le polynôme interpolant f en $n+1$ **abscices équidistantes** $x_0, \cdots, x_n$ appartenant à $[a,b]$ et $y_i = f(x_i)$. Alors

$$\max_{x \in [a,b]} |f(t) - p_n(t)| \leq \frac{1}{4(n+1)} \left(\frac{b-a}{n} \right)^{n+1} \max_{t \in [a,b]} |f^{(n+1)}(t)|$$

Utilisation de Matlab ou Octave

Fonctions Matlab/Octave

En Matlab/Octave on peut calculer les polynômes d' interpolation en utilisant les commandes **polyfit et polyval**.

1. p = polyfit(x,y,n) calcule les coefficients du polynÃ´me de degrÃ© n qui interpole les valeurs y aux points x.
2. px = polyval(p,t) calcule les valeurs px dâun polynÃ´me de degrÃ© n, dont les $n+1$ coefficients sont memorisÃ©s dans le vecteur p, au point t, c'est-Ã -dire :

$$px = p_1 t^n + \ldots + p_n t + p_{n+1}.$$

Installation d' Octave

Le logiciel Octave est accessible pour ;

1. Windows
2. Linux
3. Mac OS

site Internet

http://sourceforge.net/project/showfiles.php?group id=2888