Chapitre 2

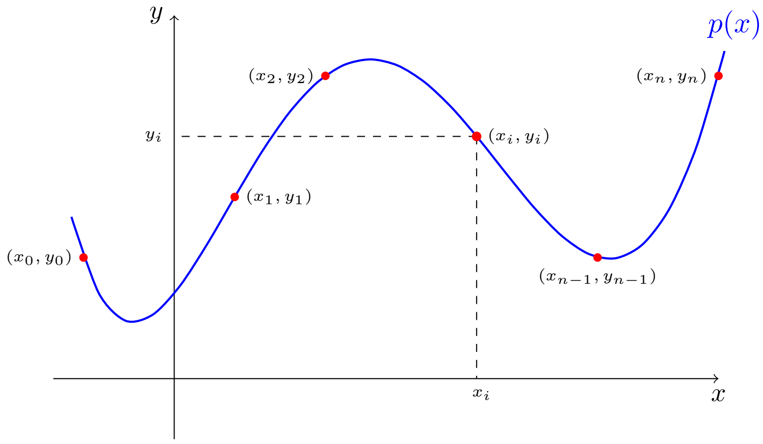
Interpolation polynomiale

Sommaire

2.1 Introduction

Les fonctions les plus faciles à évaluer numériquement sont les fonctions polynômes. Il est donc important de savoir approximer une fonction arbitraire par des polynômes.

L'interpolation polynomiale est l'une des méthodes d'approximation d'une fonction ou d'un ensemble de données par un polynôme. En d'autres termes, étant donné un ensemble de $n+1$ points $\{(x_i, y_i), i = 0, \cdots, n\}$ (obtenu, par exemple, à la suite d'une expérience), on cherche un polynôme p qui passe par tous ces points, c'est-à-dire $p(x_i) = y_i, i = 0, \cdots, n$, et éventuellement vérifie d'autres conditions, de degré si possible le plus bas.



Si un tel polynôme existe, il est appelé polynôme d'interpolation ou polynôme interpolant.

Les abscisses x_i sont appelés noeuds d'interpolation tandis que les couples (x_i, y_i) sont appelés points d'interpolation, ou points de collocation.

2.2 Existence et unicité du polynôme d'interpolation

On se donne un ensemble de $n+1$ points $\{(x_i, y_i), i = 0, \cdots, n\}$. Existe-t-il un polynôme p_n de degré inférieur ou égal à n vérifiant le problème d'interpolation

$$p_n(x_i) = y_i, i = 0, \cdots, n? \tag{2.1}$$

Théorème 7 (Existence et unicite du polynome d'intrpolation)

Si les $x_i, i = 0, \cdots, n$ sont deux à deux distincts, non nécessairement rangés par ordre croissant, alors il existe un unique polynôme p_n de degré inférieur ou égal à n tel que

$$p_n(x_i) = y_i, i = 0, \cdots, n.$$

Démonstration. Une manière de résoudre ce problème est une méthode directe basée sur la résolution d'un système linéaire. Posons

$$p_n(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0.$$

Écrivons explicitement $p_n(x_i) = y_i, i = 0, \cdots, n$.

$$\begin{cases} a_n x_0^n + a_{n-1}x_0^{n-1} + \cdots + a_1 x_0 + a_0 = y_0, \\ a_n x_1^n + a_{n-1}x_1^{n-1} + \cdots + a_1 x_1 + a_0 = y_1, \\ \vdots \\ a_n x_n^n + a_{n-1}x_n^{n-1} + \cdots + a_1 x_n + a_0 = y_n. \end{cases}$$

Puisque les valeurs $x_i, y_i, i = 0, \cdots, n$ sont connues, ces relations forment un système linéaire de $n+1$ équations en les $n+1$ inconnues $a_i, i = 0, \cdots, n$ qu'on peut le mettre sous la forme matricielle

$$\underbrace{\begin{pmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{pmatrix}}_{V} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}. \tag{2.2}$$

La matrice de ce système s'appelle la matrice de Vandermonde. Le déterminant de la matrice V est non nul si les $x_i, i = 0, \cdots, n$ sont distincts deux à deux. Donc on obtient l'existence et l'unicité du polynôme d'interpolation.

En effet, le déterminant $\det V$ de la matrice est clairement un polynôme en $x_0, \cdots, x_n$. De plus ce déterminant s'annule lorsque 2 des nombres x_i, x_j sont égaux (puisqu'il y alors 2 lignes identiques). Par suite

$$\det V = Q \times P \text{ où } P = \prod_{0 \le i < j \le n} (x_j - x_i)$$

et où Q est lui-même un polynôme.

Cependant, le polynôme $\det V$ est homogène, de degré $0 + 1 + \cdots + (n-1) = \frac{n(n-1)}{2}$. Puisqu'il en est de même de P, le polynôme Q est en fait une constante. Enfin, cette constante vaut 1 puisque dans les développements de $\det V$ et de P, le coefficient du monôme $x_n^n x_{n-1}^{n-1} \cdots x_1^1$ a la même valeur non nulle (égale à 1). $\blacksquare$

Ainsi, le problème consistant à chercher le polynôme d'interpolation p_n peut se réduire à résoudre le système linéaire (2.2). Cependant, résoudre un système linéaire de $n+1$ équations à $n+1$ inconnues n'est pas une tache triviale, c'est un calcul lourd en nombre d'opérations.

Cette méthode pour trouver le polynôme p_n n'est donc pas une bonne méthode en pratique. Dans la suite on va étudier une méthode plus astucieuse pour construire le polynôme p_n.

2.3 Interpolation de Lagrange

L'approche de Lagrange pour construire le polynôme d'interpolation est simple, systématique et sans passer par la résolution du système linéaire (2.2).

On se donne $n+1$ points $(x_0, y_0), \cdots, (x_n, y_n)$, avec les x_i distincts deux à deux. Le polynôme d'interpolation sous la forme de Lagrange est une combinaison linéaire

$$L(x) = y_0\ell_0(x) + y_1\ell_1(x) + \cdots + y_n\ell_n(x) = \sum_{i=0}^{n} y_i\ell_i(x)$$

de polynômes de base de Lagrange

$$\ell_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} = \frac{(x-x_0)}{(x_i-x_0)} \cdots \frac{(x-x_{i-1})}{(x_i-x_{i-1})} \frac{(x-x_{i+1})}{(x_i-x_{i+1})} \cdots \frac{(x-x_n)}{(x_i-x_n)},$$

où $i = 0, \cdots, n$. Noter comment, étant donné l'hypothèse initiale qu'il n'y a pas de deux x_j identiques, alors lorsque $i \neq j$, $x_i - x_j \neq 0$, cette expression est toujours bien définie.

Pour tout $j \neq i$, $\ell_i(x)$ inclut le terme $(x - x_j)$ dans le numérateur, donc le produit entier sera nul à $x = x_j$ c'est-à-dire

$$\ell_i(x_j) = 0 \text{ pour tout } j \neq i.$$

D'autre part,

$$\ell_i(x_i) = \prod_{j=0, j\neq i}^{n} \frac{x_i - x_j}{x_i - x_j} = 1.$$

Il s'ensuit que $y_i\ell_i(x_i) = y_i$, donc à chaque point x_i, $L(x_i) = 0 + \cdots + 0 + y_i + 0 + \cdots + 0 = y_i$, ceci montrant que L interpole exactement les points $(x_0, y_0), \cdots, (x_n, y_n)$. Il est donc important de remarquer que le problème d'interpolation (2.1) a une unique solution explicite L.

De plus, les polynômes $\ell_0, \cdots, \ell_n$ sont linéairement indépendants car si l'équation $\sum_{i=0}^{n} y_i\ell_i(x) = 0$ doit être satisfaite pour tout $x \in \mathbb{R}$ alors $\sum_{i=0}^{n} \alpha_i\ell_i(x_j) = 0$ doit être vraie pour tout $j = 0, \cdots, n$ et puisque $\sum_{i=0}^{n} \alpha_i\ell_i(x_j) = \alpha_j$, on conclut que tous les α_j sont nuls. Par conséquent, la famille $\{\ell_0, \cdots, \ell_n\}$ forme une nouvelle base de $\mathcal{P}_n$ l'espace vectoriel des fonctions polynômes sur $\mathbb{R}$ à coefficients réels, de degré inférieur ou égal à n.

De tout ce qui précède, nous avons le théorème suivant.

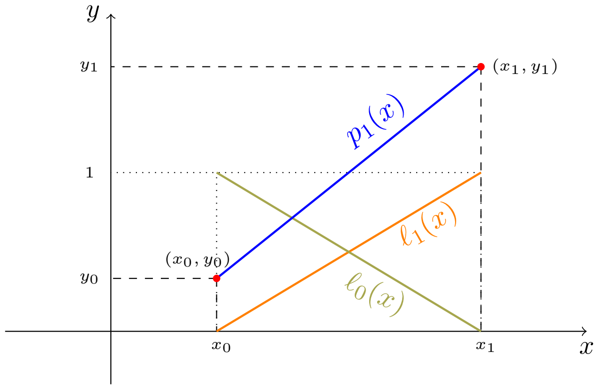
Théorème 8 (Interpolation de Lagrange)

Étant donné $n+1$ points $(x_0, y_0), \cdots, (x_n, y_n)$. Si les x_i sont distincts deux à deux, alors il existe un unique polynôme p_n de degré inférieur ou égal à n tel que $p_n(x_i) = y_i, i = 0, \cdots, n$ qu'on peut écrire sous la forme dite de Lagrange

$$p_n(x) = \sum_{i=0}^{n} y_i \ell_i(x) \quad \text{où} \quad \ell_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}.$$

Exemple 9

Pour $n = 1$ le polynôme de Lagrange s'écrit $\quad p_1(x) = y_0 \dfrac{x - x_1}{x_0 - x_1} + y_1 \dfrac{x - x_0}{x_1 - x_0}.$

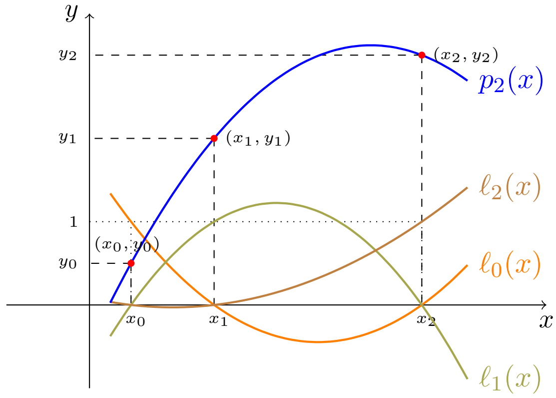


C'est l'équation de la droite qui passe par les points (x_0, y_0) et (x_1, y_1). ∎

Exemple 10

Pour $n = 2$ le polynôme de Lagrange s'écrit

$$p_2(x) = y_0 \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1 \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_2 \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}.$$



C'est l'équation de parabole qui passe par les points $(x_0, y_0), (x_1, y_1)$ et (x_2, y_2). ∎

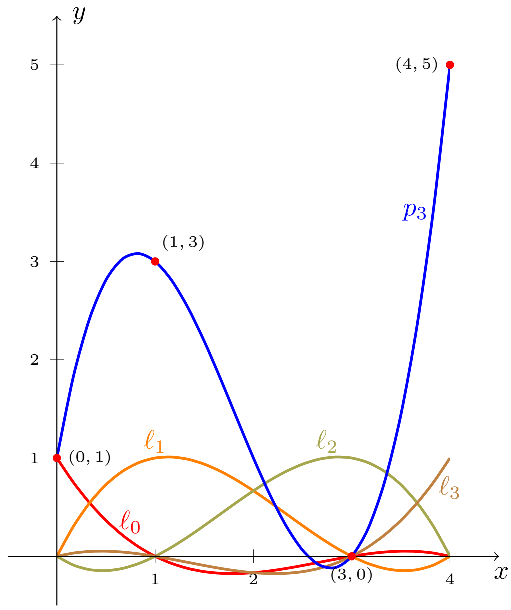
Exemple 11

Déterminons le polynôme d'interpolation p_3 qui interpole les points (x_i, y_i) suivantes :

$$(0,1), (1,3), (3,0) \text{ et } (4,5).$$

Le polynôme de Lagrange s'écrit sous la forme

$$p_3(x) = \sum_{i=0}^{3} y_i \ell_i(x) = \ell_0(x) + 3\ell_1(x) + 5\ell_3(x),$$



où

$$\begin{aligned}
\ell_0(x) &= \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} \\
&= \frac{-1}{12}(x-1)(x-3)(x-4), \\
\ell_1(x) &= \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} \\
&= \frac{1}{6}x(x-3)(x-4), \\
\ell_3(x) &= \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\
&= \frac{1}{12}x(x-1)(x-3).
\end{aligned}$$

Donc

$$\begin{aligned}
p_3(x) &= \tfrac{-1}{12}(x-1)(x-3)(x-4) + \tfrac{1}{2}x(x-3)(x-4) + \tfrac{5}{12}x(x-1)(x-3) \\
&= \tfrac{5}{6}x^3 - \tfrac{9}{2}x^2 + \tfrac{17}{3}x + 1.
\end{aligned}$$

■

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction continue et soit $x_0, x_1, \cdots, x_n$, $(n+1)$ points distincts deux à deux. Interpoler la fonction f aux points $x_i, i = 0, \cdots, n$ signifie chercher un polynôme p_n de degré $\leq n$ tel que

$$p_n(x_i) = f(x_i) \text{ pour } i = 0, \cdots, n.$$

La solution de ce problème est donc donnée par

$$p_n(x) = \sum_{i=0}^{n} f(x_i)\,\ell_i(x) \quad \text{où} \quad \ell_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j},$$

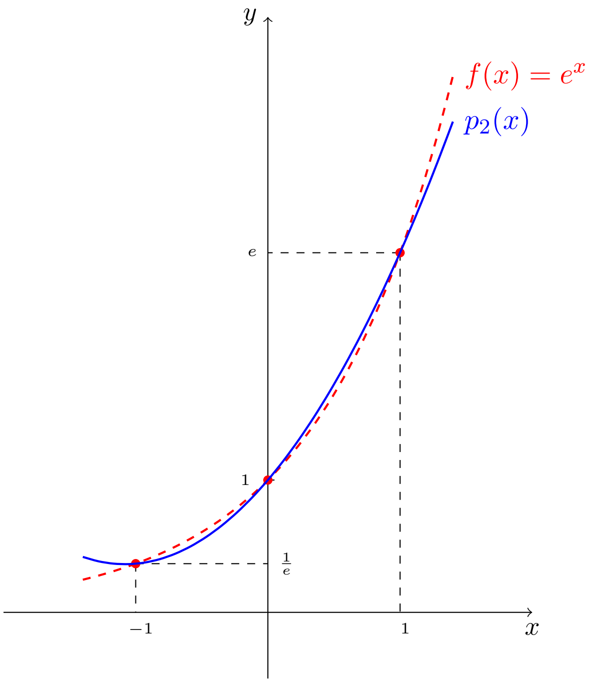
et le polynôme p_n est appelé interpolant de f de degré n aux points $x_0, x_1, \cdots, x_n$.

Exemple 12

Soit $f : \mathbb{R} \to \mathbb{R}$ la fonction définie par $f(x) = e^x$.

Cherchons l'interpolant de f aux points $-1, 0, 1$. On a

$$\begin{aligned}
p_2(x) &= f(x_0)\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f(x_1)\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + f(x_2)\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\
&= e^{-1}\frac{x(x-1)}{2} + \frac{(x+1)(x-1)}{-1} + e\frac{(x+1)x}{2} \\
&= \left(\frac{e}{2} - 1 + \frac{1}{2e}\right)x^2 + \left(\frac{e}{2} - \frac{1}{2e}\right)x + 1
\end{aligned}$$



∎

Remarque 9

En pratique, on utilise l'interpolation polynomiale avec des polynômes de degré n assez grand ou l'interpolation polynomiale par morceaux.

La méthode d'interpolation de Lagrange présente un inconvénient majeur, elle n'est pas récursive. En effet, si l'on souhaite passer d'un polynôme p_n de degré n à un polynôme p_{n+1} de degré $(n+1)$, en ajoutant un point de collocation, on doit reprendre pratiquement tout le processus à zéro. Il est intéressant donc de mettre p_n sous une forme récurrente qui permet de compléter les valeurs déjà obtenues sans refaire tous les calculs. On arrive donc au polynôme d'interpolation de Newton.

Algorithme 2 : Polynôme d'interpolation par la méthode de Lagrange

Données : 1. Les $(n+1)$ points $\{(x_i, y_i), i = 0, \cdots, n\}$.

2. Un point x en lequel à évaluer le polynôme d'interpolation p_n.

```
début
    n := degré du polynôme p_n
    pour i ← 0 à n faire
        si x = x_i                          /* Vérifier si x = x_i, i = 0, 1, ··· , n. */
        alors
            p = y_i
            écrire p                        /* La valeur du polynôme de Lagrange en x. */
            stop
    pour i ← 0 à n faire
        d_i = 1
        pour j ← 0 à n faire
            si i ≠ j alors
                d_i = d_i * (x_i - x_j)     /* Calcul des dénominateurs de ℓ_i. */
    num = 1
    pour i ← 0 à n faire
        num = num * (x - x_i)               /* Calcul de ∏_{i=0}^{n} (x - x_i). */
    p = 0
    pour i ← 0 à n faire
        q = num/(x - x_i)                   /* Évaluation de p_n(x). */
        p = p + y_i * q/d_i
    écrire p.                               /* La valeur du polynôme de Lagrange en x. */
```

Résultat : La valeur du polynôme de Lagrange en x.

Code Matlab 2 : Polynôme d'interpolation de Lagrange

```matlab
function p = polynome_lagrange(xi, fxi, x)
% Polynôme d'interpolation de Lagrange
% Entrée:
%   1)xi:  Un vecteur ligne contenant les abscisses xi.
%   2)fxi: Un vecteur ligne contenant les valeurs f(xi)
%          de la fonction à interpoler pour les xi correspondants.
%   3)x :  Un point en lequel à évaluer le polynôme d'interpolation.
% Sortie:
%   1) p : La valeur du polynôme de Lagrange en x.
% Exemple d'appel:
%  >> xi=[-1 0 1 2 3]
%  >> fxi=[-2 -1 0 3 2]
%  >> p = polynome_lagrange(xi, fxi, 2.5)
n = length(xi);
    for i = 1 : n
        if x==xi(i)  %Vérifier si x est égal à l'un des xi.
            p=fxi(i)
            return
        end
    end
d = zeros(1,n);
    for i=1:n
      d(i)=1;
       for j=1:n
         if i≠j
          d(i) = d(i)*(xi(i)-xi(j));  %Calcul des dénominateurs de L_i.
         end
       end
    end
num=1;
   for i=1:n
    num = num*(x - xi(i));   %Calcul du produit de (x-x_i).
   end
p=0;
  for i=1:n
    q=num/(x-xi(i));
    p = p+fxi(i)*q/d(i);   %Évaluation de p_n(x).
  end
```

2.4 Interpolation de Newton

L'interpolation de Newton est une méthode ne diffère de l'interpolation lagrangienne que par la façon dont le polynôme est calculé, le polynôme d'interpolation qui en résulte est le même. Pour cette raison on parle aussi plutôt de la forme de Newton du polynôme de Lagrange.

Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction continue et soit $x_0, x_1, \cdots, x_n$, $(n+1)$ points distincts deux à deux. Posons $y_i = f(x_i), i = 0, \cdots, n$.

L'interpolation de Newton est un procédé itératif qui permet de calculer le polynôme d'interpolation de Lagrange p_n de degré n basé sur $(n+1)$ points $\{(x_i, y_i), i = 0, \cdots, n\}$ à partir du polynôme d'interpolation p_{n-1} de degré $n-1$ basé sur n points $\{(x_i, y_i), i = 0, \cdots, n-1\}$, en posant

$$p_n(x) = p_{n-1}(x) + C_n(x).$$

Ce procédé permet de construire le polynôme d'interpolation sous la forme de Newton comme une combinaison linéaire

$$N(x) = a_0 N_0(x) + a_1 N_1(x) + \cdots + a_n N_n(x) = \sum_{i=0}^{n} a_i N_i(x)$$

de polynômes de base de Newton

$$N_i(x) = (x - x_0)(x - x_1) \cdots (x - x_{i-1}) = \prod_{0 \leq j < i} (x - x_j) \quad \text{pour } i = 0, \cdots, n.$$

Il est facile de vérifier que la famille $\{N_i, i = 0, \cdots, n\}$ forme une base de $\mathcal{P}_n$ l'espace des polynômes de degré inférieur ou égal à n. Le problème du calcul du polynôme d'interpolation p_n est alors ramené au calcul des coefficients $\{a_i, i = 0, \cdots, n\}$ tels que $p_n(x) = \sum\limits_{i=0}^{n} a_i N_i(x)$.

Noter que si on calcule les $n+1$ coefficients $\{a_i, i = 0, \cdots, n\}$ et après on ajoute un point d'interpolation, il n'y a plus à calculer que le coefficient a_{n+1} car la nouvelle base est déduite de l'autre base en ajoutant le polynôme N_{n+1}.

Pour calculer les coefficients $\{a_i, i = 0, \cdots, n\}$ on doit donc s'assurer que

$$p_n(x_i) = f(x_i), \quad i = 0, \cdots, n.$$

Le polynôme d'interpolation dans la base de Newton évalué en x_0 donne

$$p_n(x_0) = \sum_{i=0}^{n} a_i N_i(x_0) = a_0 = f(x_0).$$

Le premier coefficient est donc

$$a_0 = f(x_0).$$

On doit ensuite s'assurer que $p_n(x_1) = f(x_1)$, c'est-à-dire

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

ce qui permet d'isoler a_1 pour obtenir

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

Le troisième coefficient a_2 est à son tour déterminé par

$$p_n(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f(x_2).$$

En isolant a_2, on obtient

$$\begin{aligned}
a_2 &= \frac{f(x_2) - a_0 - a_1(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{f(x_2) - f(x_0) - \frac{f(x_1)-f(x_0)}{x_1-x_0}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{1}{x_2 - x_0}\left(\frac{f(x_2) - f(x_1)}{x_2 - x_1} + \frac{f(x_1) - f(x_0)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0}\left(\frac{x_2 - x_0}{x_2 - x_1}\right)\right) \\
&= \frac{\dfrac{f(x_2) - f(x_1)}{x_2 - x_1} - \dfrac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_0}.
\end{aligned}$$

Avant de calculer les autres coefficients on va introduire la notion de différences divisées.

2.4.1 Différences divisées

Définition 10 (Différences divisées)

On définit les différences divisées d'ordres successifs $0, 1, 2, \cdots, n$ de la fonction f aux points $x_i, i = 0, \cdots, n$ par les relations de récurrences suivantes.

Ordre 0 : $f[x_i] = f(x_i)$.

Ordre 1 : $f[x_i, x_{i+1}] = \dfrac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$.

Ordre 2 : $f[x_i, x_{i+1}, x_{i+2}] = \dfrac{f[x_{i+1}, x_{i+2}] - f[x_i, x_{i+1}]}{x_{i+2} - x_i}$.

$\vdots$

Ordre k : $f[x_i, x_{i+1}, \cdots, x_{i+k}] = \dfrac{f[x_{i+1}, \cdots, x_{i+k}] - f[x_i, \cdots, x_{i+k-1}]}{x_{i+k} - x_i}$, $2 \leq k \leq n$.

Pour expliciter le processus récursif, les différences divisées peuvent être calculées en les disposant de la manière suivante dans un tableau.

Table de différences divisées

x_i	$f(x_i)$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$	$\cdots$	$f[x_0, x_1, \cdots, x_n]$
x_0	$\boxed{f(x_0)}$					
x_1	$f(x_1)$	$\boxed{f[x_0, x_1]}$				
x_2	$f(x_2)$	$f[x_1, x_2]$	$\boxed{f[x_0, x_1, x_2]}$			
x_3	$f(x_3)$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$\boxed{f[x_0, x_1, x_2, x_3]}$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	
x_n	$f(x_n)$	$f[x_{n-1}, x_n]$	$f[x_{n-2}, x_{n-1}, x_n]$	$f[x_{n-3}, x_{n-2}, x_{n-1}, x_n]$	$\cdots$	$\boxed{f[x_0, x_1, \cdots, x_n]}$

La construction de cette table est simple.

Pour obtenir par exemple $f[x_0, x_1, x_2]$, il suffit de soustraire les 2 termes adjacents $f[x_1, x_2] - f[x_0, x_1]$ et de diviser le résultat par $(x_2 - x_0)$.

De même, pour obtenir $f[x_0, x_1, x_2, x_3]$, on soustrait $f[x_0, x_1, x_2]$ de $f[x_1, x_2, x_3]$ et on divise le résultat par $(x_3 - x_0)$.

Suivant cette notation, on a

$$
\begin{aligned}
a_0 &= f[x_0], \\
a_1 &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_0, x_1], \\
a_2 &= \frac{\dfrac{f(x_2) - f(x_1)}{x_2 - x_1} - \dfrac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_0} = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = f[x_0, x_1, x_2],
\end{aligned}
$$

et les polynômes d'interpolation p_1 et p_2 sous la forme de Newton sont

$$
\begin{aligned}
p_1(x) &= f[x_0] + f[x_0, x_1](x - x_0), \\
p_2(x) &= f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1).
\end{aligned}
$$

Nous remarquons que la forme de Newton utilise la diagonale principale de la table de différences divisées ci-dessus. Dans le cas général, on obtient la formule d'interpolation de Newton.

2.4.2 Formule d'interpolation de Newton

Théorème 11 (Formule de Newton) Le polynôme d'interpolation sous la forme de Newton passant par les points $\{(x_i, f(x_i)), i = 0, \cdots, n\}$ peut s'écrire

$$p_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) \\ + \cdots + f[x_0, x_1, \cdots, x_n](x - x_0)(x - x_1)\cdots(x - x_{n-1}),$$

ou bien

$$p_n(x) = \sum_{i=0}^{n} f[x_0, x_1, \cdots, x_i] \prod_{j=0}^{i-1} (x - x_j), \tag{2.3}$$

ou encore sous la forme récursive

$$\begin{cases} p_0(x) = f(x_0) \\ p_n(x) = p_{n-1}(x) + f[x_0, x_1, \cdots, x_n] \prod\limits_{j=0}^{n-1} (x - x_j), \quad n \geq 1. \end{cases}$$

Démonstration. On démontre le résultat par induction.

Pour $n = 0, 1, 2$ le résultat correspond aux calculs ci-dessus.

Supposons que la formule de Newton soit vraie au rang $n-1$. Il s'agit de montrer qu'il est également vrai pour les polynômes de degré n. Pour ce faire, on introduit les polynômes $p_{n-1}(x)$ et $q_{n-1}(x)$ de degré $(n-1)$ et passant respectivement par les points $\{(x_i, f(x_i)), i = 0, \cdots, n-1\}$ et $\{(x_i, f(x_i)), i = 1, \cdots, n\}$. On pose

$$a_i = f[x_0, x_1, \cdots, x_i], \quad i = 0, \cdots, n-1,$$

$$b_i = f[x_1, x_2, \cdots, x_{i+1}], \quad i = 1, \cdots, n-1.$$

Les a_i et les b_i sont les différences divisées relatives aux n premiers et aux n derniers points, respectivement. Suivant la définition des différences divisées, on observe que

$$a_n = f[x_0, x_1, \cdots, x_n] = \frac{f[x_1, \cdots, x_n] - f[x_0, \cdots, x_{n-1}]}{x_n - x_0},$$

c'est-à-dire

$$a_n = \frac{b_{n-1} - a_{n-1}}{x_n - x_0}. \tag{2.4}$$

L'hypothèse d'induction permet d'affirmer que

$$p_{n-1}(x) = p_{n-2}(x) + a_{n-1}(x - x_0)(x - x_1)\cdots(x - x_{n-2}),$$

$$q_{n-1}(x) = q_{n-2}(x) + b_{n-1}(x - x_1)(x - x_2)\cdots(x - x_{n-1}).$$

Par soustraction on obtient

$$\begin{aligned} q_{n-1}(x) - p_{n-1}(x) = \;& q_{n-2}(x) - p_{n-2}(x) \\ & + \Big(b_{n-1}(x - x_{n-1}) - a_{n-1}(x - x_0)\Big)(x - x_1) \cdots (x - x_{n-2}) \end{aligned} \tag{2.5}$$

On remarque qu'on a

$$p_n(x) = p_{n-1}(x) + \frac{x - x_0}{x_n - x_0}\left(q_{n-1}(x) - p_{n-1}(x)\right). \tag{2.6}$$

En effet, suivant la définition des polynômes p_{n-1} et q_{n-1}, on a

$$p_n(x_0) = p_{n-1}(x_0) + 0 = f(x_0)$$

et à l'autre extrémité on a

$$p_n(x_n) = p_{n-1}(x_n) + q_{n-1}(x_n) - p_{n-1}(x_n) = q_{n-1}(x_n) = f(x_n).$$

Aux points intermédiaires correspondant au $i = 1, \cdots, n-1$, on a

$$\begin{aligned} p_n(x_i) &= p_{n-1}(x_i) + \frac{x_i - x_0}{x_n - x_0}\left(q_{n-1}(x_i) - p_{n-1}(x_i)\right) \\ &= f(x_i) + \frac{x_i - x_0}{x_n - x_0}\left(f(x_i) - f(x_i)\right) = f(x_i). \end{aligned}$$

Ce qui montre que p_n interpole les points $\{(x_i, f(x_i)), i = 0, \cdots, n\}$. Donc d'après l'unicité du polynôme d'interpolation, et comme le polynôme p_n est de degré au plus n, on a bien le résultat

$$p_n(x) - p_{n-1}(x) = \frac{x - x_0}{x_n - x_0}\left(q_{n-1}(x) - p_{n-1}(x)\right). \tag{2.7}$$

Ceci permet de trouver les racines de $p_n(x) - p_{n-1}(x)$, qui sont $x_0, x_1, \cdots, x_{n-1}$. Ce polynôme s'écrit donc

$$p_n(x) - p_{n-1}(x) = \alpha_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}), \alpha_n \in \mathbb{R}. \tag{2.8}$$

Selon l'équation (2.6) et (2.5), le coefficient de la puissance x^n de p_n est

$$a_n = \frac{b_{n-1} - a_{n-1}}{x_n - x_0}.$$

En comparant avec (2.8), on trouve que

$$p_n(x) = p_{n-1}(x) + a_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}),$$

où

$$a_n = \frac{b_{n-1} - a_{n-1}}{x_n - x_0} = f[x_0, x_1, \cdots, x_n].$$

Ce qui établit le résultat par récurrence sur n. ■

Exemple 13

Déterminons p_3 le polynôme d'interpolation de la fonction f qui passe par les points $(-1,-2)$, $(0,-1)$, $(1,0)$ et $(2,3)$. La table de différences divisées pour ces points est

Table de différences divisées

x_i	$f(x_i)$	$f[x_i,x_{i+1}]$	$f[x_i,x_{i+1},x_{i+2}]$	$f[x_i,x_{i+1},x_{i+2},x_{i+3}]$
-1	$\boxed{-2}$			
0	-1	$\boxed{1}$		
1	0	1	$\boxed{0}$	
2	3	3	1	$\boxed{\frac{1}{3}}$

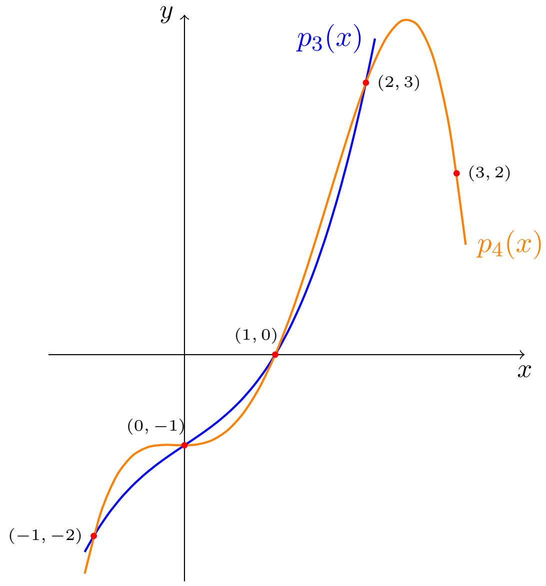
Suivant la formule de Newton, le polynôme interpolant les points $(-1,-2)$, $(0,-1)$, $(1,0)$ et $(2,3)$ est

$$p_3(x) = -2 + 1\times(x+1) + 0\times(x+1)(x-0) + \frac{1}{3}(x+1)(x-0)(x-1) = \frac{1}{3}x^3 + \frac{2}{3}x - 1.$$

Si l'on souhaite ajouter un point d'interpolation et calculer un polynôme de degré 4, il n'est pas nécessaire de tout recommencer. Par exemple, si l'on veut inclure le point $(3,2)$, on peut compléter la table de différences divisées déjà utilisée.

Table de différences divisées

x_i	$f(x_i)$	$DD1$	$DD2$	$DD3$	$DD4$
-1	$\boxed{-2}$				
0	-1	$\boxed{1}$			
1	0	1	$\boxed{0}$		
2	3	3	1	$\boxed{\frac{1}{3}}$	
3	2	-1	-2	-1	$\boxed{-\frac{1}{3}}$



Ce polynôme de degré 4 est alors

$$\begin{aligned} p_4(x) &= p_3(x) - \frac{1}{3}(x+1)(x-0)(x-1)(x-2) \\ &= -\frac{1}{3}x^4 + x^3 + \frac{1}{3}x^2 - 1. \end{aligned}$$

qui est tout simplement le polynôme de degré 3 déjà calculé auquel on a ajouté une correction de degré 4. ▪

Algorithme 3 : Polynôme d'interpolation sous la forme de Newton

Données : 1. Les $(n+1)$ points $\{(x_i, f(x_i)), i = 0, \cdots, n\}$.

2. Un point x en lequel à évaluer le polynôme d'interpolation p_n.

début

  $n :=$ degré du polynôme p_n

  pour $i \leftarrow 0$ **à** n **faire**

    $d_{i0} := f(x_i)$ `/* La colonne 0 ième de la matrice de */` `/* différences divisées` d. `*/`

  pour $i \leftarrow 1$ **à** n **faire**

    pour $j \leftarrow 1$ **à** i **faire**

      $d_{ij} = \dfrac{d_{i,j-1} - d_{i-1,j-1}}{x_i - x_{i-j+1}}$ `/* La table des différences divisées. */`

  écrire $d_{00}, d_{11}, \cdots, d_{nn}$ `/* Noter que` $d_{ii} = f[x_0, x_1, \cdots, x_i]$. `*/`

  définir $p = d_{nn}$ **pour** $i \leftarrow n-1$ **à** 0 **faire**

    $p = p * (x - x_i) + d_{ii}$ `/*` $p_n(x)$ `en utilisant le polynôme de Newton. */`

  écrire p. `/* La valeur du polynôme de Newton en` x. `*/`

Résultat : 1. La matrice contenant la table de différences divisées.

2. Les coefficients du polynôme de Newton.

3. La valeur du polynôme de Newton en x.

Code Matlab 3 : Polynôme d'interpolation en utilisant la méthode des différences divisées

```matlab
function [d, a, p] = polynome_newton(xi, fxi, x)
% Polynôme d'interpolation de Newton
% Entrée:
%   1)xi:  Un vecteur ligne contenant les abscisses xi.
%   2)fxi: Un vecteur ligne contenant les valeurs f(xi)
%          de la fonction à interpoler pour les xi correspondants.
%   3)x :  Un point en lequel à évaluer le polynôme d'interpolation.
% Sortie:
%   1) d : Matrice contenant la table de différences divisées.
%   2) a : Vecteur contenant les coefficients du polynôme de Newton.
%   3) p : La valeur du polynôme de Newton en x.
% Exemple d'appel:
%  >> xi=[-1 0 1 2 3]
```

```
% >> fxi=[-2 -1 0 3 2]
% >> [d, a, p] = polynome_newton(xi, fxi, 2.5)
n = length(xi);
d = zeros(n,n);
a = zeros(1,n);
d(:,1) = fxi;
a(1) = d(1,1);
  for i=2:n
       for j=2:i
         d(i,j) = (d(i,j-1)-d(i-1,j-1))/(xi(i)-xi(i-j+1));
       end
       a(i) = d(i,i);
  end
p=a(n);
  for i=n-1:-1:1
    p = p*(x - xi(i))+a(i);
  end
```

2.4.3 Cas où les points d'interpolation sont équidistants

Il arrive parfois que la fonction à interpoler soit donnée en des points équidistants x_i dans l'intervalle $[a,b]$ avec le pas $h=\frac{b-a}{n}$ qui est appelé le pas d'interpolation où

$$x_i = a + ih = a + i\frac{b-a}{n}, \quad i = 0, 1, \cdots, n.$$

Différences finies progressives

On définit la différence finie progressive d'ordre 1 par

$$\Delta f\left(x_i\right) = f\left(x_{i+1}\right) - f\left(x_i\right), \quad i = 0, 1, \cdots, n-1,$$

et d'ordre $k = 2, \cdots, n$ par la formule de récurrence

$$\Delta^k f\left(x_i\right) = \Delta\left(\Delta^{k-1} f\right)\left(x_i\right) = \Delta^{k-1} f\left(x_{i+1}\right) - \Delta^{k-1} f\left(x_i\right), \quad i = 0, 1, \cdots, n-k,$$

où l'on convient que $\Delta^0 f\left(x_i\right) = f\left(x_i\right), \quad i = 0, 1, \cdots, n.$

Les coefficients $\Delta^k f\left(x_i\right)$ se calculent suivant le processus décrit comme dans la table de différences divisées, voir page 21.

Table de différences finies progressives

x_i	$f(x_i)$	$\Delta f(x_i)$	$\Delta^2 f(x_i)$	$\Delta^3 f(x_i)$	$\cdots$ $\Delta^n f(x_i)$
x_0	$\boxed{f(x_0)}$				
x_1	$f(x_1)$	$\boxed{\Delta f(x_0)}$			
x_2	$f(x_2)$	$\Delta f(x_1)$	$\boxed{\Delta^2 f(x_0)}$		
x_3	$f(x_3)$	$\Delta f(x_2)$	$\Delta^2 f(x_1)$	$\boxed{\Delta^3 f(x_0)}$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$
x_n	$f(x_n)$	$\Delta f(x_{n-1})$	$\Delta^2 f(x_{n-2})$	$\Delta^3 f(x_{n-3})$	$\cdots$ $\boxed{\Delta^n f(x_0)}$

Proposition 12 (Relation avec différences divisées)
Les différences divisées sont données par

$$f[x_i, x_{i+1}, \cdots, x_{i+k}] = \frac{\Delta^k f(x_i)}{k! h^k}, k = 1, \cdots, n, \quad i = 0, \cdots, n-k.$$

Démonstration. On montre cette relation par récurrence sur k. Pour $k = 1$, on a

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

Donc la relation est vérifiée pour $k = 1$. Maintenant, supposons que la relation est vérifiée à l'ordre k. On a

$$\begin{aligned} f[x_i, x_{i+1}, \cdots, x_{i+k+1}] &= \frac{f[x_{i+1}, \cdots, x_{i+k+1}] - f[x_i, \cdots, x_{i+k}]}{x_{i+k+1} - x_i} = \frac{\Delta^k f(x_{i+1}) - \Delta^k f(x_i)}{(k+1) h} \\ &= \frac{1}{(k+1) h} \frac{\Delta^{k+1} f(x_i)}{k! h^k} = \frac{\Delta^{k+1} f(x_i)}{(k+1)! h^{k+1}}. \end{aligned}$$

Ce qui montre qu'elle l'est encore vérifiée à l'ordre $k + 1$. ■

Formule de Newton progressive

D'après la formule (2.3), le polynôme d'interpolation sous la forme de Newton devient alors, ce qu'on appelle la formule de Newton progressive

$$p_n(x) = f(x_0) + \sum_{k=1}^{n} \frac{\Delta^k f(x_0)}{k! h^k} \prod_{j=0}^{k-1} (x - x_j).$$

En écrivant $x = x_0 + sh, \ s \in [0, n]$ et en utilisant la notation

$$\binom{s}{k} = \frac{s(s-1)\cdots(s-k+1)}{k!} \text{ si } k > 0 \text{ et } \binom{s}{0} = 1, \tag{2.9}$$

on obtient

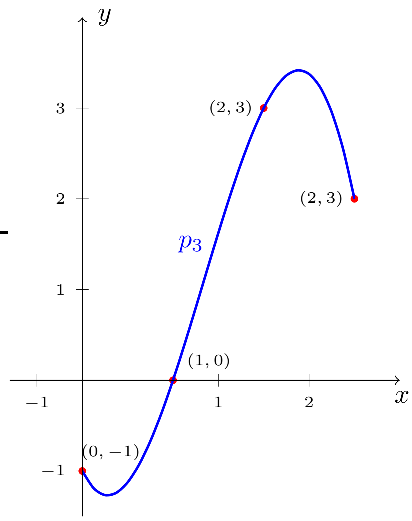
$$p_n(x) = p_n(x_0 + sh) = \sum_{k=0}^{n} \binom{s}{k} \Delta^k f(x_0).$$

Exemple 14

Déterminons la formule de Newton progressive associé à la fonction f passant par les points $(0,-1)$, $(1,0)$, $(2,3)$ et $(3,2)$. On a $n=3, x_0=0$ et $h=1$ donc $x_i = 0+i = i, i = 0,1,2,3$.
La table de différences finies progressives pour ces points est

Table de différences finies progressives

x_i	$f(x_i)$	$\Delta f(x_i)$	$\Delta^2 f(x_i)$	$\Delta^3 f(x_i)$
0	$\boxed{-1}$			
1	0	$\boxed{\Delta f(x_0) = 1}$		
2	3	$\Delta f(x_1) = 3$	$\boxed{\Delta^2 f(x_0) = 2}$	
3	2	$\Delta f(x_2) = -1$	$\Delta^2 f(x_1) = -4$	$\boxed{\Delta^3 f(x_0) = -6}$



Alors la formule de Newton progressive est donnée par

$$\begin{aligned} p_3(x) &= f(x_0) + \frac{\Delta f(x_0)}{h}(x-x_0) + \frac{\Delta^2 f(x_0)}{2h^2}(x-x_0)(x-x_1) \\ &\quad + \frac{\Delta^3 f(x_0)}{6h^3}(x-x_0)(x-x_1)(x-x_2) \\ &= -1 + x + x(x-1) + -x(x-1)(x-2) \\ &= -x^3 + 4x^2 - 2x - 1. \qquad \blacksquare \end{aligned}$$

Différences finies régressives

On définit la différence finie régressives d'ordre 1 par

$$\nabla f(x_i) = f(x_i) - f(x_{i-1}), \quad i = 1, \cdots, n,$$

et d'ordre $k = 2, \cdots, n$ par la formule de récurrence

$$\nabla^k f(x_i) = \nabla\left(\nabla^{k-1} f\right)(x_i) = \nabla^{k-1} f(x_i) - \nabla^{k-1} f(x_{i-1}), \quad i = k, \cdots, n,$$

où l'on convient que $\nabla^0 f(x_i) = f(x_i), \quad i = 0, 1, \cdots, n.$
Comme dans le cas progressive, les coefficients $\nabla^k f(x_i)$ se calculent suivant le processus décrit comme dans la table de différences divisées, voir page 21.

Table de différences finies régressives

x_i	$f(x_i)$	$\nabla f(x_i)$	$\nabla^2 f(x_i)$	$\nabla^3 f(x_i)$	$\cdots$	$\nabla^n f(x_i)$
x_0	$f(x_0)$					
x_1	$f(x_1)$	$\nabla f(x_1)$				
x_2	$f(x_2)$	$\nabla f(x_2)$	$\nabla^2 f(x_2)$			
x_3	$f(x_3)$	$\nabla f(x_3)$	$\nabla^2 f(x_3)$	$\nabla^3 f(x_3)$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	
x_n	$\boxed{f(x_n)}$	$\boxed{\nabla f(x_n)}$	$\boxed{\nabla^2 f(x_n)}$	$\boxed{\nabla^3 f(x_n)}$	$\cdots$	$\boxed{\nabla^n f(x_n)}$

Proposition 13 (Relation avec différences divisées)

Les différences divisées sont données par

$$f[x_i, x_{i+1}, \cdots, x_{i+k}] = \frac{\nabla^k f(x_{i+k})}{k!h^k}, k = 1, \cdots, n, \quad i = 0, \cdots, n-k.$$

Démonstration. On montre cette relation par récurrence sur k. Pour $k = 1$, on a

$$f[x_i, x_{i+1}] = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} = \frac{\nabla f(x_{i+1})}{h}.$$

Donc la relation est vérifiée pour $k = 1$. Maintenant, supposons que la relation est vérifiée à l'ordre k. On a

$$\begin{aligned} f[x_i, x_{i+1}, \cdots, x_{i+k+1}] &= \frac{f[x_{i+1}, \cdots, x_{i+k+1}] - f[x_i, \cdots, x_{i+k}]}{x_{i+k+1} - x_i} = \frac{\nabla^k f(x_{i+k+1}) - \nabla^k f(x_{i+k})}{(k+1)h} \\ &= \frac{1}{(k+1)h} \frac{\nabla^{k+1} f(x_{i+k+1})}{k!h^k} = \frac{\nabla^{k+1} f(x_{i+k+1})}{(k+1)!h^{k+1}}. \end{aligned}$$

Ce qui montre qu'elle l'est encore vérifiée à l'ordre $k+1$. ■

Formule de Newton régressive

Formule de Newton régressive est donnée par

$$p_n(x) = f(x_n) + \sum_{k=1}^{n} \frac{\nabla^k f(x_n)}{k!h^k} \prod_{j=0}^{k-1} (x - x_{n-j}).$$

En écrivant $x = x_n - th, \ t \in [0, n]$ et en utilisant la notation

$$\binom{t}{k} = \frac{t(t-1)\cdots(t-k+1)}{k!} \text{ si } k > 0 \text{ et } \binom{t}{0} = 1,$$

on obtient

$$p_n(x) = p_n(x_n - th) = \sum_{k=0}^{n} \binom{t}{k} \nabla^k f(x_n).$$

Exemple 15

On reprend l'exemple 14. Déterminons le polynôme d'interpolation sous la forme de Newton régressive de la fonction f passant par les points $(0,-1)$, $(1,0)$, $(2,3)$ et $(3,2)$. On a $n = 3, x_0 = 0$ et $h = 1$ donc $x_i = i, i = 0,1,2,3$.

La table de différences finies régressives pour ces points est

Table de différences finies régressives

x_i	$f(x_i)$	$\nabla f(x_i)$	$\nabla^2 f(x_i)$	$\nabla^3 f(x_i)$
0	-1			
1	0	$\nabla f(x_1) = 1$		
2	3	$\nabla f(x_2) = 3$	$\nabla^2 f(x_2) = 2$	
3	$\boxed{2}$	$\boxed{\nabla f(x_3) = -1}$	$\boxed{\nabla^2 f(x_3) = -4}$	$\boxed{\nabla^3 f(x_3) = -6}$

Alors la formule de Newton régressive est donnée par

$$\begin{aligned} p_3(x) &= f(x_3) + \frac{\nabla f(x_3)}{h}(x - x_3) + \frac{\nabla^2 f(x_3)}{2h^2}(x - x_3)(x - x_2) \\ &\quad + \frac{\nabla^3 f(x_3)}{6h^3}(x - x_3)(x - x_2)(x - x_1) \\ &= 2 - (x-3) - 2(x-3)(x-2) - (x-3)(x-2)(x-1) \\ &= -x^3 + 4x^2 - 2x - 1. \end{aligned}$$

∎

2.5 Erreur d'interpolation

L'interpolation polynomiale permet d'approximer une fonction donnée $f : [a,b] \to \mathbb{R}$ par un polynôme p_n qui interpole les valeurs de f aux points $x_0, x_1, \cdots, x_n \in [a,b]$. Cette approximation entraîne une erreur d'interpolation en tout point $x \neq x_i, i = 0,1,\cdots,n$,

$$E_n(x) = f(x) - p_n(x).$$

2.5.1 Formule d'erreur d'interpolation

Un avantage de la forme d'interpolation de Newton est qu'elle fournit également une formule pour l'erreur d'interpolation

$$f(x) - p_n(x) = f[x_0, x_1, \cdots, x_n, x](x - x_0)(x - x_1)\cdots(x - x_n). \tag{2.10}$$

En effet, si p_{n+1} est le polynôme qui interpole f aux points $x_0, x_1, \cdots, x_n, x \in [a, b]$, alors d'après le théorème 11,

$$f(x) = p_{n+1}(x) = p_n(x) + f[x_0, x_1, \cdots, x_n, x](x - x_0)(x - x_1)\cdots(x - x_n).$$

Cette formule d'erreur peut être reformulée lorsque la fonction f est $(n+1)$ fois continûment dérivable sur l'intervalle $[a, b]$, comme le montre le théorème suivant.

Théorème 14 (Erreur d'interpolation)

Soient $x_0, x_1, \cdots, x_n \in [a, b], n+1$ points distincts et soit f une fonction continue sur $[a, b]$ et $(n+1)$ fois continûment dérivable sur $]a, b[$, où $a = \min\limits_{0 \le i \le n} x_i$ et $b = \max\limits_{0 \le i \le n} x_i$. Alors pour tout $x \in [a, b]$, il existe un point $\xi_x \in]a, b[$ tel que

$$E_n(x) = f(x) - p_n(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n).$$

De plus si $f^{(n+1)}$ est continue sur $[a, b]$, alors

$$|E_n(x)| \le \frac{M_{n+1}}{(n+1)!}|(x - x_0)(x - x_1)\cdots(x - x_n)| \text{ où } M_{n+1} = \max_{x \in [a,b]}\left|f^{(n+1)}(x)\right|.$$

Pour démontrer ce théorème on a besoin du lemme suivant, qui découle du théorème de Rolle.

Lemme 1

Soit g une fonction continue sur $[a, b]$ et m fois dérivable sur $]a, b[$. On suppose qu'il existe $m+1$ points $c_0 < c_1 < \cdots < c_m$ de $[a, b]$ tels que $g(c_i) = 0$. Alors il existe $\xi \in]c_0, c_m[$, tel que $g^{(m)}(\xi) = 0$.

Démonstration du lemme. Le lemme se démontre par récurrence sur m. Pour $m = 1$, c'est le théorème de Rolle. Supposons le lemme démontré pour $m - 1$. Le théorème de Rolle donne des points $\alpha_0 \in]c_0, c_1[, \cdots, \alpha_{m-1} \in]c_{m-1}, c_m[$ tels que $g'(\alpha_i) = 0$. Par hypothèse de récurrence, il existe donc $\xi \in]\alpha_0, \alpha_{m-1}[\subset]c_0, c_m[$ tel que $(g')^{(m-1)}(\xi) = g^{(m)}(\xi) = 0$. ■

Démonstration du théorème. Si $x = x_i$, on a $\prod_{i=0}^{n}(x - x_i) = 0$, donc tout point ξ_x convient. Supposons maintenant x distinct des points x_i. Considérons la fonction g définie sur $[a, b]$ par

$$g(t) = f(t) - p_n(t) - f[x_0, x_1, \cdots, x_n, x] \prod_{i=0}^{n} (t - x_i).$$

La fonction g est $\mathcal{C}^{n+1}([a, b])$ et s'annule au moins en les $(n + 2)$ points $t = x_0, x_1, \cdots, x_n, x \in [a, b]$ donc d'après le lemme précédent il existe $\xi_x \in]a, b[$ tel que $g^{(n+1)}(\xi_x) = 0$. Or $p_n^{(n+1)}(t) = 0$ et $\frac{d^{n+1}}{t^{n+1}} \prod_{i=0}^{n}(t - x_i) = (n + 1)!$, on a par conséquent

$$f[x_0, x_1, \cdots, x_n, x] = \frac{f^{(n+1)}(\xi_x)}{(n + 1)!}.$$

Donc d'après la formule (2.10), il existe $\xi_x \in]a, b[$ tel que

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

De plus si $f^{(n+1)}$ est continue sur $[a, b]$, qui est un compact, alors $f^{(n+1)}$ est bornée et atteint ses bornes et donc $\left|f^{(n+1)}(\xi_x)\right| \leq \max_{x\in[a,b]} \left|f^{(n+1)}(x)\right|$. ■

Remarque 15

- L'erreur d'interpolation est nulle aux points de collocation, c'est-à-dire $E_n(x_i) = 0, i = 0, 1, \cdots, n$.
- L'erreur est composée de deux termes, le terme $\frac{\prod_{i=0}^{n}(x-x_i)}{(n+1)!}$ qui dépend du choix des points x_i et le terme $\max_{x\in[a,b]} \left|f^{(n+1)}(x)\right|$ lié à la régularité de f.

Exemple 16

Soit $f(x) = e^x$ à interpoler aux points $0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$ et 1. Calculons l'erreur d'approximation en $x = \frac{1}{3}$.

Cinq points d'interpolation donnent $n = 4$. Donc $f^{(n+1)}(x) = f^{(5)}(x) = e^x$. Il existe alors une valeur $\xi \in]0, 1[$ telle que

$$\begin{aligned} e^{\frac{1}{3}} - p_4\left(\tfrac{1}{3}\right) &= \frac{e^{\xi}}{5!}\left(\tfrac{1}{3} - 0\right)\left(\tfrac{1}{3} - \tfrac{1}{4}\right)\left(\tfrac{1}{3} - \tfrac{1}{2}\right)\left(\tfrac{1}{3} - \tfrac{3}{4}\right)\left(\tfrac{1}{3} - 1\right) = \frac{e^{\xi}}{5!}\frac{320}{12^5} \\ &\leq \frac{320}{12^5} \max_{\xi\in[0,1]} \frac{\left|e^{\xi}\right|}{5!} = \frac{320}{12^5}\frac{e^1}{5!} = 0.0000291. \end{aligned}$$

■

2.5.2 Erreur pour le cas des points d'interpolation équidistants

Lorsque les points d'interpolation sont équidistants, une majoration de $|E_n(x)|$ un peu plus fine conduit au résultat suivant.

Corollaire 16 (Points d'interpolation équidistants)

Soient $x_0, x_1, \cdots, x_n \in [a,b], n+1$ points équidistants (ou équirépartis) avec $x_i = a + ih, i = 0, 1, \cdots, n$ où $h = \frac{b-a}{n}$. On suppose que $f \in \mathcal{C}^{n+1}([a,b])$. L'erreur d'interpolation pour tout $x \in [a,b]$ est estimée par

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

Démonstration. Le théorème 14 nous donne déjà que

$$|E_n(x)| \leq \frac{1}{(n+1)!} \max_{x\in[a,b]} \left| \prod_{i=0}^{n} (x - x_i) \right| \max_{x\in[a,b]} \left| f^{(n+1)}(x) \right|. \tag{2.11}$$

Il reste à estimer $\max_{x\in[a,b]} \left| \prod_{i=0}^{n} (x - x_i) \right|$. Soit $x \in [a,b]$ avec $x \neq x_i, i = 0, 1, \cdots, n$. On a $x \in I_k =]x_{k-1}, x_k[$ pour un certain $k \in \{1, \cdots, n\}$. Comme les points x_i sont équidistants, nous avons $x_k = a + kh$. On obtient alors

$$\max_{x\in I_k} |(x - x_{k-1})(x - x_k)| = \frac{h^2}{4},$$

de plus on peut estimer $|(x - x_{k-2})|$ par $2h$, $|(x - x_{k-3})|$ par $3h$ etc. D'où

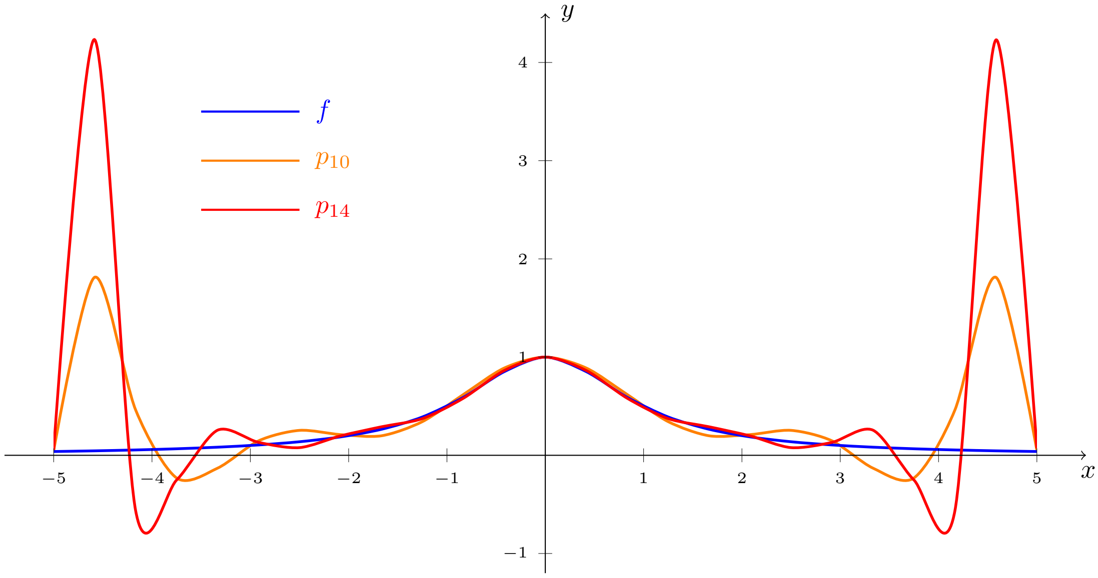
$$\max_{x\in[a,b]} \left| \prod_{i=0}^{n} (x - x_i) \right| \leq \left(\frac{h^2}{4} \right) \left(h^{n-1} n! \right) = \frac{n!}{4} h^{n+1}. \tag{2.12}$$

En substituant (2.12) dans (2.11) on obtient $|E_n(x)| \leq \frac{h^{n+1}}{4(n+1)!} \max_{x\in[a,b]} \left| f^{(n+1)}(x) \right|$. ■

Il est clair que $\frac{h^{n+1}}{4(n+1)} \to 0$ quand $n \to +\infty$ et cette convergence vers 0 est même très rapide. On serait donc tenté de penser que l'erreur tend vers 0 quand $n \to +\infty$. Mais attention car la $(n+1)^{ième}$ dérivée de f dépend de n et en fait peut croître très rapidement avec n et pour certaines fonctions, p_n ne converge pas vers f lorsque n tend vers $+\infty$. L'exemple suivant illustre une telle situation de non-convergence.

Exemple 17 (Phénomène de Runge)

On considère la fonction définie par $f(x) = \dfrac{1}{1+x^2}$ sur l'intervalle $[-5, 5]$. On note p_n le polynôme d'interpolation de f aux $(n+1)$ points équidistants dans l'intervalle $[-5, 5]$. On observe alors, comme le montre la figure ci-dessous, quand n augmente, des problèmes aux extrémités de l'intervalle. En fait $\left|f^{(n)}(5)\right|$ devient rapidement grand avec n.



■

2.6 Interpolation de Tchebychev

Contrairement aux interpolations précédentes dans lesquelles l'utilisateur peut choisir sa subdivision, l'interpolation de Tchebychev impose une subdivision $x_0, x_1, \cdots, x_n$ de l'intervalle $[a, b]$ en des points appelés points de Tchebychev. L'interpolation utilise les polynômes de Tchebychev dont les racines sont connus explicitement. L'interpolation de Tchebychev est encore appelée interpolation de Lagrange aux points de Tchebychev, car il s'agit d'une interpolation de Lagrange réalisée en des points particuliers.

2.6.1 Choix des points d'interpolation de Tchebychev

On se demande si, indépendamment de la fonction f, on peut réduire l'erreur en jouant sur le choix des $(n+1)$ points d'interpolation. On va donc s'intéresser aux choix des x_i qui rendent minimale la quantité

$$\max_{x\in[a,b]} \left|f(x) - p_n(x)\right|.$$

D'après le théorème 14, pour obtenir la meilleure estimation possible de l'erreur, il faut choisir les $(n+1)$ points d'interpolation $x_0, x_1, \cdots, x_n$ de manière à minimiser

$$\max_{x\in[a,b]} \left|\prod_{i=0}^{n}(x - x_i)\right|.$$

Théorème 17 (Points d'interpolation de Tchebytchev) Soit $n \in \mathbb{N}^*$ et soit $f : [a, b] \to \mathbb{R}$ une fonction de classe $\mathcal{C}^{n+1}$. La meilleure borne supérieure pour l'erreur d'interpolation de f que nous pouvons atteindre en faisant varier le choix des x_i est

$$|E_n(x)| = |f(x) - p_n(x)| \leq \frac{1}{2^n (n+1)!} \max_{x \in [a,b]} \left| f^{(n+1)}(x) \right|,$$

qui s'obtient en choisissant les points d'interpolation

$$x_i = \frac{b+a}{2} + \frac{b-a}{2} \cos\left(\frac{2i+1}{2n+2} \pi \right), \qquad i = 0, 1, \cdots, n,$$

qui sont appelés les points d'interpolation de Tchebychev.

Exemple 18

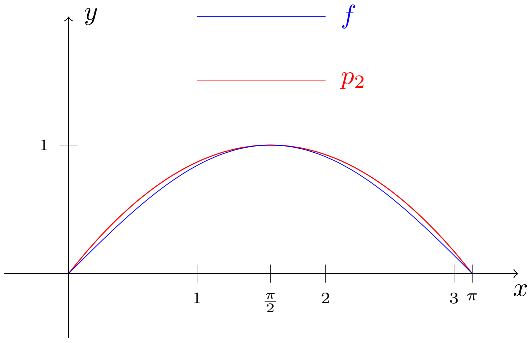
Soit la fonction f définie par $f(x) = \sin x$ à interpoler avec un polynôme de degré 2 sur l'intervalle $[0, \pi]$ en utilisant les trois points $(0, 0), \left(\frac{\pi}{2}, 1\right), (\pi, 0)$ et puis en utilisant les points d'interpolation de Tchebychev.

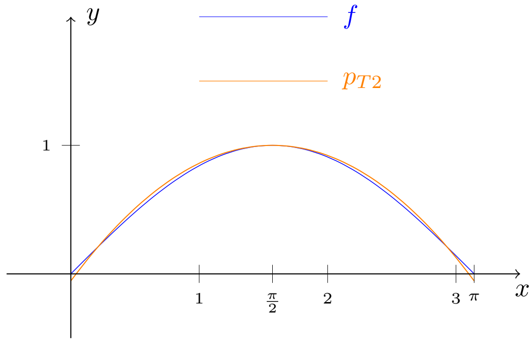
Les abscisses de Tchebychev sont

$$x_0 = \frac{\pi}{2} + \frac{\pi}{2} \cos \frac{\pi}{6} = 2.931146, x_1 = \frac{\pi}{2} + \frac{\pi}{2} \cos \frac{\pi}{2} = 1.570796, x_2 = \frac{\pi}{2} + \frac{\pi}{2} \cos \frac{5\pi}{6} = 0.210447.$$

En utilisant les points $0, \frac{\pi}{2}, \pi$: $p_2(x) = \frac{-4}{\pi^2} x^2 + \frac{4}{\pi} x \simeq -0.405285x^2 + 1.273239x$.

En utilisant les points de Tchebychev : $p_{T2}(x) \simeq -0.427496x^2 + 1.343018x - 0.0548042$.





■

La démonstration du théorème 17 fait appel à l'étude des polynômes de Tchebychev.

2.6.2 Polynômes de Tchebychev

Il existe plusieurs possibilités pour définir la famille des polynômes de Tchebychev. La plus simple est par la relation de récurrence suivante.

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \; n = 1, 2, 3, \cdots \text{ avec } T_0(x) = 1 \text{ et } T_1(x) = x.$$

Par récurrence, T_n est un polynôme de degré n et le coefficient du terme en x^n de T_n est 2^{n-1}.
À l'aide de l'identité trigonométrique $\cos(a+b) + \cos(a-b) = 2\cos a \cos b$, on démontre par récurrence que pour tout $x \in [-1, 1]$,

$$T_n(x) = \cos(n \operatorname{Arccos} x), \quad n = 1, 2, 3, \cdots,$$

ce qui peut servir de définition alternative des polynômes T_n, vus comme fonctions polynomiales définies sur l'intervalle $[-1, 1]$ même si elles ne montrent pas clairement les polynômes.
En effet, l'équation $T_n(x) = \cos(n \operatorname{Arccos} x)$ est vérifiée pour $n = 0$ et pour $n = 1$.
Pour tout $n \geq 1$, on suppose qu'elle est vérifiée aux ordres $n-1$ et n. On a

$$\begin{aligned} T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) &= 2x\cos(n \operatorname{Arccos} x) - \cos((n-1)\operatorname{Arccos} x) \\ &= 2\cos(\operatorname{Arccos} x)\cos(n \operatorname{Arccos} x) - \cos((n-1)\operatorname{Arccos} x) \\ &= \cos((n+1)\operatorname{Arccos} x). \end{aligned}$$

On en déduire qu'elle l'est encore à l'ordre $n+1$.
On observe immédiatement les propriétés suivantes

$$\begin{aligned} &|T_n(x)| \leq 1, \quad x \in [-1, 1]. \\ &T_n\left(\cos\left(\tfrac{i\pi}{n}\right)\right) = (-1)^i, \quad i = 0, 1, \cdots, n. \\ &T_n\left(\cos\left(\tfrac{2i-1}{2n}\pi\right)\right) = 0, \qquad i = 1, \cdots, n. \end{aligned}$$

Notation 2
On note $\mathbb{P}_n([a, b])$ l'ensemble des polynômes de degré n admettant n racines distinctes dans l'intervalle $[a, b]$ et dont le coefficient du monôme de degré n est égal à 1 (unitaire).

Théorème 18
Si $p \in \mathbb{P}_n([-1, 1])$ alors

$$\max_{x \in [-1,1]} |p(x)| \geq 2^{1-n}.$$

Démonstration. Supposons que $p \in \mathbb{P}_n([-1,1])$ et que $|p(x)| < 2^{1-n}$ pour tout $x \in [-1,1]$. Posons $Q_n(x) = 2^{1-n}T_n(x)$ et $z_i = \cos\left(\frac{i\pi}{n}\right), i = 0, 1, \cdots, n$. Alors par construction Q_n est un polynôme de $\mathbb{P}_n([-1,1])$ et nous aurons

$$(-1)^i Q_n(z_i) = (-1)^i 2^{1-n} T_n\left(\cos\left(\frac{i\pi}{n}\right)\right) = (-1)^i 2^{1-n} (-1)^i = 2^{1-n}.$$

Puisque p et Q_n sont tous les deux unitaires, leur différence $Q_n - p$ sera un polynôme de degré $\leq n-1$. D'autre part

$$(-1)^i p(z_i) \leq |p(z_i)| < 2^{1-n} = (-1)^i Q_n(z_i), \quad i = 0, 1, \cdots, n.$$

Par conséquent,

$$(-1)^i \Big(Q_n(z_i) - p(z_i)\Big) > 0, \quad i = 0, 1, \cdots, n.$$

Ainsi, la fonction $Q_n - p$ doit changer le signe au moins $n+1$ fois sur l'intervalle $[-1,1]$. Mais ceci n'est pas possible puisque $Q_n - p$ est un polynôme de degré au plus $n-1$. Par conséquent, nous avons une contradiction si $|p(x)| < 2^{1-n}$ pour tout $x \in [-1,1]$. ■

Lemme 3

Si Q_n est le polynôme unitaire défini sur $[-1,1]$ par

$$Q_n(x) = 2^{1-n}T_n(x),$$

alors la valeur maximale de $|Q_n(x)|$ sur l'intervalle $[-1,1]$ est 2^{1-n}.

Démonstration. La démonstration est facile puisque sur l'intervalle $[-1,1]$,

$$Q_n(x) = 2^{1-n}T_n(x) = 2^{1-n}\cos(n \operatorname{Arccos} x)$$

et $|\cos\theta| \leq \cos(0) = 1$ pour tout θ. ■

Lemme 4

Soit

$$x_i = \cos\left(\frac{2i+1}{2n+2}\pi\right), \qquad i = 0, 1, \cdots, n.$$

Alors

$$\prod_{i=0}^{n}(x - x_i) = Q_{n+1}(x) \equiv 2^{-n}T_{n+1}(x).$$

Démonstration. Cela découle du théorème fondamental de l'algèbre.

Par construction, chacun des x_i est une racine distincte du polynôme unitaire Q_{n+1}, qui est de degré $n+1$. Le théorème fondamental de l'algèbre affirme que Q_{n+1} doit donc se factoriser comme $Q_{n+1}(x) = (x-x_0)(x-x_1)\cdots(x-x_n)$. ■

Maintenant, on a le théorème suivant.

Théorème 19

Soit f une fonction dans $\mathcal{C}^{n+1}([-1,1])$. Si les points x_i de l'intervalle $[-1,1]$ sont choisis comme racines du polynôme de Tchebychev T_{n+1},

$$x_i = \cos\left(\frac{2i+1}{2n+2}\pi\right), \qquad i = 0, 1, \cdots, n,$$

alors le terme d'erreur pour l'interpolation polynomiale utilisant les points x_i est

$$|E_n(x)| = |f(x) - p_n(x)| \leq \frac{1}{2^n (n+1)!} \max_{x\in[-1,1]} \left|f^{(n+1)}(x)\right|.$$

De plus, c'est la meilleure borne supérieure que nous pouvons atteindre en faisant varier le choix des x_i.

Démonstration. La démonstration est un résultat immédiat du théorème 18 et des lemmes 3, 4 et du fait que $|E_n(x)| \leq \frac{1}{(n+1)!} |(x-x_0)(x-x_1)\cdots(x-x_n)| \max_{x\in[-1,1]} \left|f^{(n+1)}(x)\right|$. ■

Démonstration du théorème 17. Si $[a,b] = [-1,1]$, nous sommes dans le cas du théorème 19. Si l'intervalle $[a,b] \neq [-1,1]$, pour trouver un ensemble optimal de $n+1$ points x_i dans l'intervalle $[a,b]$, nous redimensionnons simplement les $n+1$ racines de T_{n+1} en points dans $[a,b]$. Plus précisément, soit s l'application linéaire (bijective) qui associe un point $x \in [-1,1]$ à un point $s(x) \in [a,b]$, tel que $s(-1) = a$ et $s(1) = b$. Cela détermine s en fait de manière unique en fonction de x,

$$s = \frac{b+a}{2} + \frac{b-a}{2}x.$$

L'ensemble optimal de $n+1$ points x_i pour interpoler la fonction f sur l'intervalle $[a,b]$ sera alors l'image sous s des $n+1$ racines de T_{n+1} :

$$x_i = \frac{b+a}{2} + \frac{b-a}{2}\cos\left(\frac{2i+1}{2n+2}\pi\right), \qquad i = 0, 1, \cdots, n.$$

■

2.7 Interpolation d'Hermite

L'interpolation d'Hermite est une extension de l'interpolation de Lagrange, qui consiste, pour une fonction donnée f dérivable et un nombre fini de points donnés $x_i, i = 0, \cdots, n$, à construire un polynôme p en faisant coïncider non seulement les fonctions f et p aux points x_i, mais aussi leurs dérivées aux points x_i. Cette méthode d'interpolation permet d'éviter les phénomènes de Runge dans l'interpolation numérique ou, plus simplement, de manipuler des polynômes ayant des propriétés proches de celles de la fonction interpolée.

Soit f une fonction de classe $\mathcal{C}^1([a,b])$. Supposons que nous connaissons la fonction f et f' en $n+1$ points, on cherche un polynôme p qui interpole ces $n+1$ points. On peut écrire au plus $n+1$ équations passant par ces $n+1$ points et vérifiant $p(x_i) = f(x_i)$ et $n+1$ équations vérifiant $p'(x_i) = f'(x_i)$, il y a donc lieu de déterminer $2n+2$ coefficients, on doit alors rechercher un polynôme de degré au plus égal $2n+1$.

Définition 20 (Polynôme d'interpolation d'Hermite)

On appelle polynôme d'interpolation d'Hermite de f aux points deux à deux distincts $x_i, i = 0, 1, \cdots, n$, un polynôme p de degré au plus $2n+1$ satisfaisant

$$p(x_i) = f(x_i) \quad \text{et} \quad p'(x_i) = f'(x_i) \text{ pour } i = 0, 1, \cdots, n. \tag{2.13}$$

Une méthode pour construire de tels polynômes est de prendre le carré des polynômes de Lagrange

$$q_i(x) = \ell_i^2(x) = \prod_{j=0, j\neq i}^{n} \left(\frac{x - x_j}{x_i - x_j}\right)^2,$$

qui sont de degré $2n$ et qui vérifient

$$q_i(x_i) = 1, \quad q_i'(x_i) = \sum_{j=0, j\neq i}^{n} \frac{2}{x_i - x_j} \text{ et pour tout } j \neq i, \quad q_i(x_j) = q_i'(x_j) = 0.$$

Un polynôme p de la forme

$$p(x) = \sum_{i=0}^{n} q_i(x)\, p_i(x)$$

satisfait donc les $2n+2$ conditions si et seulement si les polynômes p_i vérifient

$$f(x_i) = p_i(x_i) \qquad \text{et} \qquad f'(x_i) = p_i'(x_i) + q_i'(x_i)\, p_i(x_i),$$

ce qui équivaut à

$$p_i(x_i) = f(x_i) \qquad \text{et} \qquad p_i'(x_i) = f'(x_i) - q_i'(x_i)\, f(x_i).$$

Puisque le polynôme p est de degré au plus égal à $2n+1$, donc la solution la plus simple est de choisir les p_i des polynômes de degré 1 sous la forme

$$p_i(x) = f(x_i) + (x - x_i)\Big(f'(x_i) - q_i'(x_i)\, f(x_i)\Big).$$

Dans ce cas le polynôme p s'écrit sous la forme

$$p_{2n+1}(x) = \sum_{i=0}^{n} q_i(x)\Big(1 - (x - x_i)\, q_i'(x_i)\Big) f(x_i) + \sum_{i=0}^{n} q_i(x)(x - x_i) f'(x_i).$$

On a alors le théorème suivant.

Théorème 21 (Existence et unicité du polynôme d'interpolation d'Hermite)

Le polynôme p_{2n+1} donné sous la forme

$$\begin{aligned}
p_{2n+1}(x) &= \sum_{i=0}^{n} H_i(x) f(x_i) + \sum_{i=0}^{n} K_i(x) f'(x_i),\\
H_i(x) &= \Big(1 - 2(x - x_i)\, \ell_i'(x_i)\Big) \ell_i^2(x),\\
K_i(x) &= (x - x_i)\, \ell_i^2(x),\\
\ell_i(x) &= \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}, \quad \ell_i'(x_i) = \sum_{j=0, j\neq i}^{n} \frac{1}{x_i - x_j},
\end{aligned}$$

est l'unique polynôme d'interpolation d'Hermite de f aux points deux à deux distincts $x_i, i = 0, 1, \cdots, n$.

Démonstration. Il reste à montrer l'unicité puisque le polynôme p_{2n+1} satisfait aux conditions (2.13). Supposons qu'il existe deux polynômes p et q de degré au plus $2n+1$ vérifiant (2.13). Le polynôme $r = p - q$, qui est aussi de degré au plus $2n+1$, admet chaque x_i comme racine double car $r(x_i) = r'(x_i) = 0$. Il possède donc au moins $2n+2$ racines, r est donc forcément le polynôme nul. ■

Théorème 22 (Erreur dans l'interpolation d'Hermite)

Si f est une fonction de classe $\mathcal{C}^{2n+2}([a, b])$, $a = \min_{0\leq i\leq n} x_i, b = \max_{0\leq i\leq n} x_i$, alors pour tout x dans $[a, b]$, il existe ξ_x dans $]a, b[$ tel que

$$f(x) - p_{2n+1}(x) = \frac{f^{(2n+2)}(\xi_x)}{(2n+2)!} \prod_{i=0}^{n} (x - x_i)^2.$$

Démonstration. Si x est un des points $x_0, x_1, \cdots, x_n$, le résultat est évident. Sinon, posons

$$g(t) = f(t) - p_{2n+1}(t) - \Big(f(x) - p_{2n+1}(x)\Big) \prod_{i=0}^{n} \left(\frac{t - x_i}{x - x_i}\right)^2. \tag{2.14}$$

On a alors $g(x) = 0$ et pour tout entier i entre 0 et n, $g(x_i) = 0$.

En appliquant le théorème de Rolle à g, on montre l'existence de $c_0 < c_1 < \cdots < c_n$ avec c_i différent de x_j pour tout $i \neq j$ tels que $g'(c_i) = 0$ pour tout i. De plus, on a $g'(x_i) = 0$

pour tout i. Il existe donc $2n+1$ points de $[a,b]$ deux à deux distincts annulant g'. Alors en appliquant le lemme 1 à g', il existe $\xi \in]a,b[$ tel que $(g')^{(2n+1)}(\xi) = g^{(2n+2)}(\xi) = 0$. Donc il existe $\xi \in]a,b[$ tel que

$$f^{(2n+2)}(\xi) - 0 - \Big(f(x) - p_{2n+1}(x)\Big) \frac{(2n+2)!}{\prod\limits_{i=0}^{n}(x-x_i)^2} = 0. \blacksquare$$

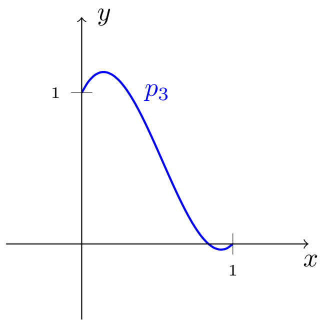
Remarque 23

- Nous pouvons définir des formules d'interpolation mixte Lagrange-Hermite où les valeurs des dérivées ne sont utilisées que pour certains points.
- Nous pouvons également faire intervenir des dérivées d'ordre plus élevée.

Exemple 19

Construisons un polynôme de degré ≤ 3 tel que $p(0) = 1, p'(0) = 2, p(1) = 0, p'(1) = 1$.
On a $n = 1$, $x_0 = 0$, $f(x_0) = 1$, $f'(x_0) = 2$, $x_1 = 1$, $f(x_1) = 0$, $f'(x_1) = 1$ et on a

$$\ell_0(x) = \frac{x - x_1}{x_0 - x_1} = -(x-1), \quad \ell_1(x) = \frac{x - x_0}{x_1 - x_0} = x,$$
$$\ell_0'(x_0) = \frac{1}{x_0 - x_1} = -1, \quad \ell_1'(x_1) = \frac{1}{x_1 - x_0} = 1.$$



Donc

$$H_0(x) = \Big(1 - 2(x-0)(-1)\Big)(-(x-1))^2 = 2x^3 - 3x^2 + 1,$$
$$H_1(x) = \Big(1 - 2(x-1)(1)\Big)x^2 = -2x^3 + 3x^2,$$

Alors

$$p_3(x) = (2x^3 - 3x^2 + 1)(1) + (-2x^3 + 3x^2)(0) + (x-0)(-(x-1))^2(2) + (x-1)(x)^2(1)$$
$$= 5x^3 - 8x^2 + 2x + 1. \blacksquare$$

Algorithme 4 : Polynôme d'interpolation d'Hermite

Données : 1. Les $(n+1)$ points $\{(x_i, f(x_i)), i = 0, \cdots, n\}$ et les valeurs $f'(x_i)$.

2. Un point x en lequel à évaluer le polynôme d'interpolation d'Hermite p_{2n+1}.

début
- $n+1$:= nombre de points d'interpolation
- **pour** $i \leftarrow 0$ **à** n **faire**
  - **si** $x = x_i$ `/* Vérifier si` $x = x_i, i = 0, 1, \cdots, n$. `*/`
  - **alors**
    - $p = f(x_i)$
    - **écrire** p `/* La valeur du polynôme d'Hermite en` x. `*/`
    - **stop**
- $p = 0$
- **pour** $i \leftarrow 0$ **à** n **faire**
  - $\ell_i = 1$
  - **pour** $j \leftarrow 0$ **à** n **faire**
    - **si** $i \neq j$ **alors**
      - $\ell_i = \ell_i * (x - x_j)/(x_i - x_j)$ `/* Calcul des` $\ell_i(x)$. `*/`
      - $c_i = c_i + 1/(x_i - x_j)$ `/* Calcul des` $\ell_i'(x_i)$. `*/`
  - $p = p + ((1 - 2(x - x_i)c_i) f(x_i) + (x - x_i) f'(x_i)) * \ell_i^2$ `/* Calcul de` $p(x)$. `*/`
- **écrire** p. `/* La valeur du polynôme d'interpolation d'Hermite en` x. `*/`

Résultat : La valeur du polynôme d'interpolation d'Hermite en x.

Code Matlab 4 : Polynôme d'interpolation d'Hermite

```
function p = polynome_hermite(xi, fxi, fpxi, x)
% Polynôme d'interpolation d'Hermite
% Entrée:
%   1)xi:  Un vecteur ligne contenant les abscisses xi.
%   2)fxi: Un vecteur ligne contenant les valeurs f(xi)
%          de la fonction à interpoler pour les xi correspondants.
%   3)fpxi: Un vecteur ligne contenant les valeurs f'(xi).
%   3)x :  Un point en lequel à évaluer le polynôme d'Hermite.
% Sortie:
```

```
%   1) p : La valeur du polynôme d'interpolation d'Hermite en x.
% Exemple d'appel:
%  >> xi=[-1 0 1 2 3]
%  >> fxi=[-2 -1 0 3 2]
%  >> fpxi=[1 2 1 4 -2]
%  >> p = polynome_hermite(xi, fxi, fpxi, 2.5)
n = length(xi);
    for i = 1 : n
        if x==xi(i)  %Vérifier si x est égal à l'un des xi.
            p=fxi(i)
            return
        end
    end
p=0;
L = zeros(1,n);
c = zeros(1,n);
    for i=1:n
      L(i)=1;
       for j=1:n
         if i≠j
          L(i)  = L(i)*(x-xi(j))/(xi(i)-xi(j));  %Calcul de L_i(x).
          c(i)  = c(i)+1/(xi(i)-xi(j));          %Calcul de L'_i(x_i).
         end
       end
       p = p + ( (1-2*(x-xi(i))*c(i))*fxi(i) + (x-xi(i))*fpxi(i) ...
          )*L(i)^2;  %Évaluation de p_{2n+1}(x).
    end
```

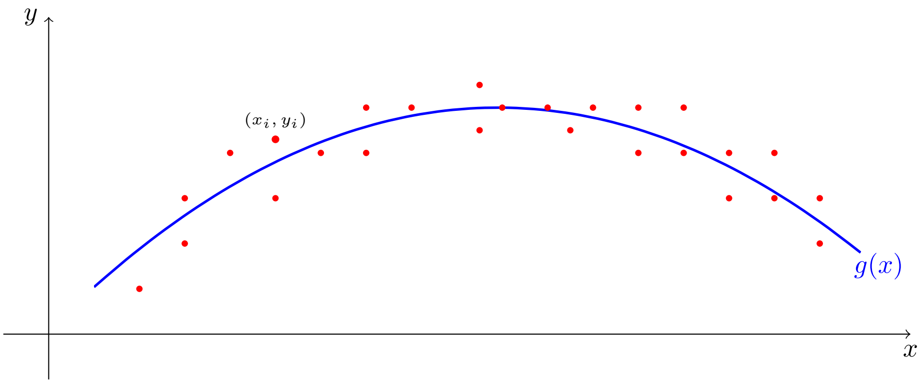
2.8 Approximation au sens des moindres carrés discrets

Si on se donne une famille de points du plan $\{(x_i, y_i), i = 0, \cdots, n\}$, les x_i étant distincts, alors il existe un unique polynôme $p(x)$ de degré inférieur ou égal à n tel que

$$p(x_i) = y_i, i = 0, \cdots, n,$$

qui est le polynôme d'interpolation des points (x_i, y_i).

Si le nombre de points n est trop grand, ou si les ordonnées sont bruitées, on préfère en général chercher une fonction $g(x)$ qui, dans une classe donnée (polynômes, fractions rationnelles, polynômes trigonométriques, exponentielles, ...), approche "au mieux" les points (x_i, y_i), on parle alors d'approximation, de lissage ou bien de régression.



Soit donc une famille de fonctions

$$g_0(x), g_1(x), \cdots, g_m(x), \quad m \leq n,$$

linéairement indépendantes. Étant donné $m+1$ nombres réels $a_0, a_1, \cdots, a_m$ on peut introduire le nombre $E(a)$,

$$E(a) = \sum_{i=0}^{n} |g(x_i) - y_i|^2,$$

avec $g(x) = \sum_{i=0}^{n} a_i g_i(x)$. La quantité $E(a)$ représente la somme des erreurs quadratiques entre les valeurs données et celles prises par g aux points x_i. Le problème d'approximation se formule alors de la façon suivante :

$$\text{Trouver } a^* \in \mathbb{R}^{m+1}, \text{ tel que } E(a^*) \leq E(a) \text{ pour tout } a \text{ dans } \mathbb{R}^{m+1}.$$

La méthode des moindres carrés consiste à trouver les valeurs optimales des paramètres $a_0, a_1, \cdots$, a_m en minimisant la somme, $E(a)$ des erreurs quadratiques.

Par exemple si l'on désire faire de la régression polynomiale, c'est-à-dire prendre pour $g(x) = p(x)$ un polynôme de degré $\leq m$, on a

$$g_i(x) = x^i, \; i = 0, \cdots, m, \quad g(x) = p(x) = a_0 + a_1 x + \cdots + a_m x^m,$$

où les coefficients $a_i, \; i = 0, \cdots, m$ sont les inconnues du problème de moindres carrés, on doit minimiser la fonction

$$E(a) = \sum_{i=0}^{n} |a_0 + a_1 x_i + \cdots + a_m x_i^m - y_i|^2.$$

Pour minimiser $E(a)$ on cherche d'abord les points stationnaires, i.e. les points a qui vérifient

$$\frac{\partial E(a)}{\partial a_0} = \frac{\partial E(a)}{\partial a_1} = \cdots = \frac{\partial E(a)}{\partial a_m} = 0.$$

Puisque

$$\begin{aligned}
\frac{\partial E(a)}{\partial a_0} &= 2\sum_{i=0}^{n}\left(a_0 + a_1 x_i + \cdots + a_m x_i^m - y_i\right),\\
\frac{\partial E(a)}{\partial a_1} &= 2\sum_{i=0}^{n}\left(a_0 + a_1 x_i + \cdots + a_m x_i^m - y_i\right) x_i,\\
&\vdots\\
\frac{\partial E(a)}{\partial a_m} &= 2\sum_{i=0}^{n}\left(a_0 + a_1 x_i + \cdots + a_m x_i^m - y_i\right) x_i^m,
\end{aligned}$$

alors $\frac{\partial E(a)}{\partial a_0} = \frac{\partial E(a)}{\partial a_1} = \cdots = \frac{\partial E(a)}{\partial a_m} = 0$ équivalent à

$$\underbrace{\begin{pmatrix} n+1 & \sum_{i=0}^{n} x_i & \cdots & \sum_{i=0}^{n} x_i^m \\ \sum_{i=0}^{n} x_i & \sum_{i=0}^{n} x_i^2 & \cdots & \sum_{i=0}^{n} x_i^{m+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=0}^{n} x_i^m & \sum_{i=0}^{n} x_i^{m+1} & \cdots & \sum_{i=0}^{n} x_i^{2m} \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{pmatrix}}_{a} = \underbrace{\begin{pmatrix} \sum_{i=0}^{n} y_i \\ \sum_{i=0}^{n} y_i x_i \\ \vdots \\ \sum_{i=0}^{n} y_i x_i^m \end{pmatrix}}_{b},$$

soit encore

$$Aa = b.$$

Ce système a une solution unique à condition que les x_i soient distincts.

Par exemple pour $m = 1$, où on parle d'ajustement ou de régression linéaire

$$\begin{pmatrix} n+1 & \sum_{i=0}^{n} x_i \\ \sum_{i=0}^{n} x_i & \sum_{i=0}^{n} x_i^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{n} y_i \\ \sum_{i=0}^{n} y_i x_i \end{pmatrix}.$$

La solution de ce système est

$$\begin{aligned}
a_0 &= \frac{\left(\sum_{i=0}^{n} y_i\right)\left(\sum_{i=0}^{n} x_i^2\right) - \left(\sum_{i=0}^{n} y_i x_i\right)\left(\sum_{i=0}^{n} x_i\right)}{(n+1)\sum_{i=0}^{n} x_i^2 - \left(\sum_{i=0}^{n} x_i\right)^2},\\
a_1 &= \frac{(n+1)\left(\sum_{i=0}^{n} y_i x_i\right) - \left(\sum_{i=0}^{n} y_i\right)\left(\sum_{i=0}^{n} x_i\right)}{(n+1)\sum_{i=0}^{n} x_i^2 - \left(\sum_{i=0}^{n} x_i\right)^2}.
\end{aligned}$$

Exemple 20

Déterminons la ligne des moindres carrés approximant les données du tableau suivant.

x_i	0.5	0.5	1	1.5	2	2.5	3	3.5	4	4	4.5	4.75	5.5	6	6	6.5
y_i	0.5	1	1	1.25	1	1	1.5	1.5	1.5	2	2	1.75	2	2	2.5	2.25

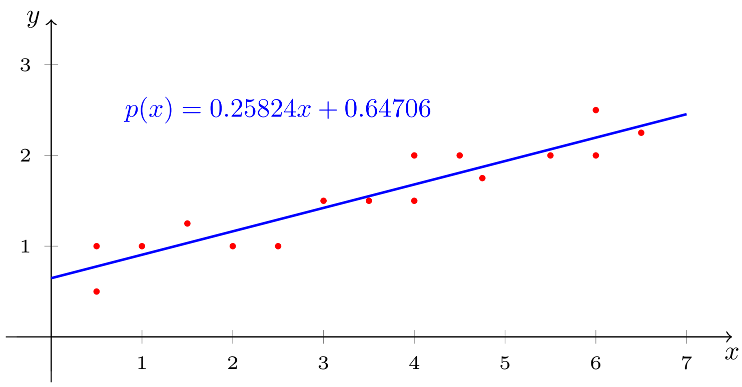
On a $n+1=16$, $\sum_{i=0}^{n} x_i = 55.75$, $\sum_{i=0}^{n} y_i = 24.75$, $\sum_{i=0}^{n} x_i^2 = 254.5625$, $\sum_{i=0}^{n} x_i y_i = 101.8125$.
Alors

$$a_0 = \frac{\left(\sum_{i=0}^{n} y_i\right)\left(\sum_{i=0}^{n} x_i^2\right)-\left(\sum_{i=0}^{n} y_i x_i\right)\left(\sum_{i=0}^{n} x_i\right)}{(n+1)\sum_{i=0}^{n} x_i^2-\left(\sum_{i=0}^{n} x_i\right)^2} = \frac{24.75\times 254.5625 - 101.8125\times 55.75}{16\times 254.5625-(55.75)^2} = 0.647\,06,$$

$$a_1 = \frac{(n+1)\left(\sum_{i=0}^{n} y_i x_i\right)-\left(\sum_{i=0}^{n} y_i\right)\left(\sum_{i=0}^{n} x_i\right)}{(n+1)\sum_{i=0}^{n} x_i^2-\left(\sum_{i=0}^{n} x_i\right)^2} = \frac{16\times 101.8125 - 55.75\times 24.75}{16\times 254.5625-(55.75)^2} = 0.258\,24,$$

et donc

$$p(x) = 0.258\,24x + 0.647\,06.$$



■

2.9 Exercices

En rédaction!