Cours 9 MT09. Interpolation

Antoine Zurek

LMAC - UTC

I. Motivations

1. Approcher une fonction
2. Différentes notions d'approximation
3. Illustration du problème d'interpolation
4. Interpolation, mais interpolation polynomiale !

II. Interpolation polynomiale

1. Polynôme d'interpolation dans la base de Lagrange
2. Exemple d'interpolation via la base de Lagrange
3. Erreur d'interpolation
4. Une majoration de l'erreur
5. La base de Newton
7. Différences divisées et coefficients de Newton
8. Calcul pratique des différences divisées
9. Exemple d'interpolation dans la base de Newton
6. Newton vs Lagrange
10. Algorithme de Horner

Table of contents

I. Motivations

1. Approcher une fonction
2. Différentes notions d'approximation
3. Illustration du problème d'interpolation
4. Interpolation, mais interpolation polynomiale !

II. Interpolation polynomiale

1. Polynôme d'interpolation dans la base de Lagrange
2. Exemple d'interpolation via la base de Lagrange
3. Erreur d'interpolation
4. Une majoration de l'erreur
5. La base de Newton
7. Différences divisées et coefficients de Newton
8. Calcul pratique des différences divisées
9. Exemple d'interpolation dans la base de Newton
6. Newton vs Lagrange
10. Algorithme de Horner

1. Approcher une fonction

On souhaite calculer $f(t)$ pour de nombreuses valeurs de t, mais :

- f n'est connue qu'en quelques points expérimentaux.
- L'évaluation de f est fournie par un code de calcul coûteux.

Idée.

Remplacer f par une fonction $\widetilde{f}$ plus simple à évaluer . Autrement dit, on cherche une fonction $\widetilde{f}$ dans une classe "simple", notée $\tilde{F}$, de functions.

Exemples pour $\tilde{F}$. Un polynôme, un polynôme trigonométrique etc.

2. Différentes notions d'approximation

Pour des données $(t_i, f(t_i))_{i=0}^n$, plusieurs objectifs sont possibles.

Interpolation : Chercher $\widetilde{f}$ tq $\widetilde{f}(t_i) = f(t_i)$ pour tout i.

Moindres carrés : Minimiser $\sum_{i=0}^n \left(\widetilde{f}(t_i) - f(t_i)\right)^2$.

Approximation quadratique : Minimiser $\int_a^b |f(x) - \widetilde{f}(x)|^2 \, dx$.

Approximation uniforme : Minimiser $\max_{x\in[a,b]} |f(x) - \widetilde{f}(x)|$.

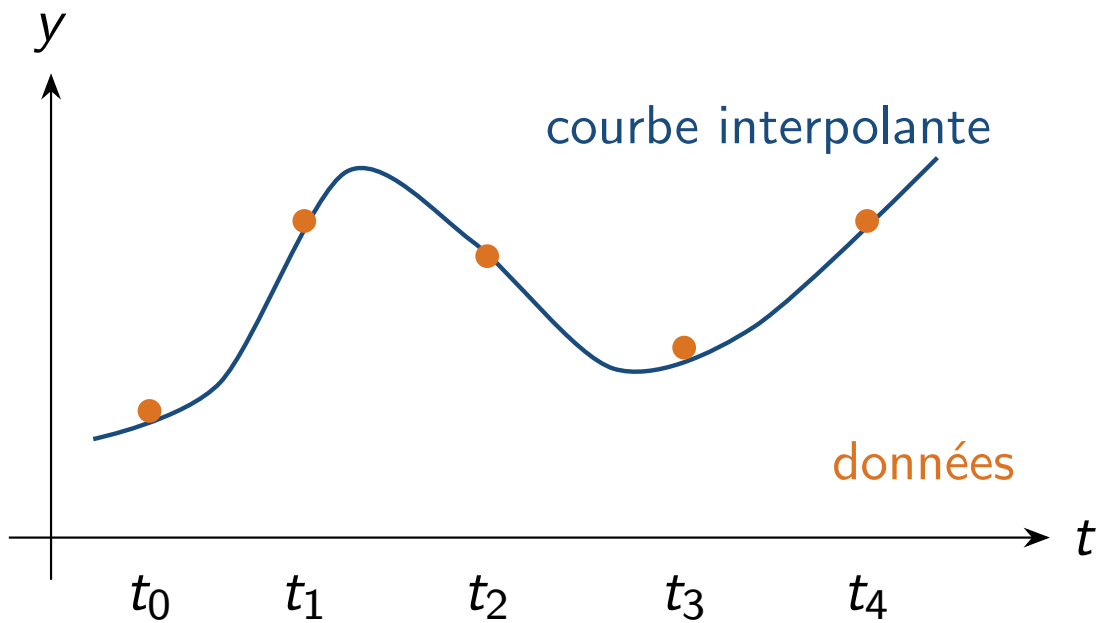
Dans ce chapitre, on étudie uniquement l'interpolation !

3. Illustration du problème d'interpolation

On se donne $n+1$ abscisses deux à deux distinctes : $t_0, \ldots, t_n$,

et les valeurs associées : $y_0, \ldots, y_n$.

On cherche une fonction p tq $\boxed{p(t_i) = y_i, \qquad i = 0, \ldots, n}$.



Problème. Il existe une infinité de fonctions satisfaisant ces $n+1$ contraintes ! Il faut fixer une classe $\tilde{F}$ de fonctions "simples".

4. Interpolation, mais interpolation polynomiale !

Dans la suite, on considère la classe

$$\tilde{F} = \mathbb{P}_n = \{p : \ p \text{ est un polynôme de degré } \leq n\}.$$

Rappel. $\dim(\mathbb{P}_n) = n + 1$.

Objectif. Pour des couples

$$\{(t_0, y_0), (t_1, y_1), \ldots, (t_n, y_n)\}$$

donnés, on cherche à construire $p_n \in \mathbb{P}_n$ tel que

$$p_n(t_i) = y_i, \qquad i = 0, \ldots, n.$$

4. Interpolation, mais interpolation polynomiale !

Dans la suite, on considère la classe

$$\tilde{F} = \mathbb{P}_n = \{p : \ p \text{ est un polynôme de degré } \leq n\}.$$

Rappel. $\dim(\mathbb{P}_n) = n + 1$.

Objectif. Pour des couples

$$\{(t_0, y_0), (t_1, y_1), \ldots, (t_n, y_n)\}$$

donnés, on cherche à construire $p_n \in \mathbb{P}_n$ tel que

$$p_n(t_i) = y_i, \qquad i = 0, \ldots, n.$$

Théorème 5.2.1.

Etant données $n + 1$ réels $t_0, \ldots, t_n$ distincts deux à deux et $n + 1$ réels $y_0, \ldots, y_n$, alors, il existe un unique polynôme $p \in \mathbb{P}_n$ tel que $p_n(t_i) = y_i$ pour tout $i \in \{0, \ldots, n\}$.

Table of contents

I. Motivations

1. Approcher une fonction
2. Différentes notions d'approximation
3. Illustration du problème d'interpolation
4. Interpolation, mais interpolation polynomiale !

II. Interpolation polynomiale

1. Polynôme d'interpolation dans la base de Lagrange
2. Exemple d'interpolation via la base de Lagrange
3. Erreur d'interpolation
4. Une majoration de l'erreur
5. La base de Newton
7. Différences divisées et coefficients de Newton
8. Calcul pratique des différences divisées
9. Exemple d'interpolation dans la base de Newton
6. Newton vs Lagrange
10. Algorithme de Horner

1. Polynôme d'interpolation dans la base de Lagrange

On considère $t_0, \ldots, t_n \in \mathbb{R}$ donnés et deux à deux distincts. Pour chaque $i = 0, \ldots, n$, on définit les polynômes de base de Lagrange

$$L_i(t) = \prod_{\substack{k=0 \\ k \neq i}}^{n} \frac{t - t_k}{t_i - t_k} \quad \forall t \in \mathbb{R}.$$

Remarque 1. $L_i \in \mathbb{P}_n$ et $L_i(t_j) = 1$ si $i = j$ et 0 sinon.

Remarque 2. $\mathbb{P}_n = \mathrm{Vect}\langle L_0, \ldots, L_n \rangle$.

1. Polynôme d'interpolation dans la base de Lagrange

On considère $t_0, \ldots, t_n \in \mathbb{R}$ donnés et deux à deux distincts. Pour chaque $i = 0, \ldots, n$, on définit les polynômes de base de Lagrange

$$L_i(t) = \prod_{\substack{k=0 \\ k \neq i}}^{n} \frac{t - t_k}{t_i - t_k} \quad \forall t \in \mathbb{R}.$$

Remarque 1. $L_i \in \mathbb{P}_n$ et $L_i(t_j) = 1$ si $i = j$ et 0 sinon.

Remarque 2. $\mathbb{P}_n = \mathrm{Vect}\langle L_0, \ldots, L_n \rangle$.

À retenir.

Pour des nœuds distincts $t_0, \ldots, t_n$ et des valeurs $y_0, \ldots, y_n \in \mathbb{R}$ donnée, l'unique polynôme $p_n \in \mathbb{P}_n$ vérifiant $p_n(t_i) = y_i$ est

$$p_n(t) = \sum_{i=0}^{n} y_i L_i(t).$$

2. Exemple d'interpolation via la base de Lagrange

On considère les données

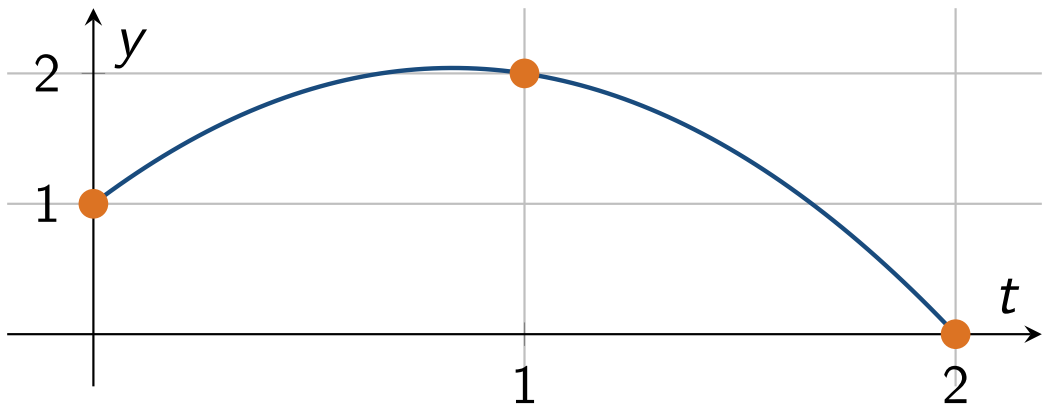
$$(t_0, y_0) = (0, 1), \qquad (t_1, y_1) = (1, 2), \qquad (t_2, y_2) = (2, 0).$$

Les polynômes de Lagrange sont

$$L_0(t) = \frac{(t-1)(t-2)}{2},$$
$$L_1(t) = -t(t-2),$$
$$L_2(t) = \frac{t(t-1)}{2}.$$

Ainsi,

$$p_2(t) = L_0(t) + 2L_1(t) = -\frac{3}{2}t^2 + \frac{5}{2}t + 1.$$



3. Erreur d'interpolation

On suppose maintenant que les données proviennent d'une fonction

$$y_i = f(t_i),$$

et l'on note : $e_n(t) = f(t) - p_n(t)$.

Théorème 5.2.2.

Si $f \in C^{n+1}([a,b])$ et si les nœuds et t appartiennent à $[a,b]$, alors il existe un point ξ_t dans le plus petit intervalle contenant $t, t_0, \ldots, t_n$ tel que

$$e_n(t) = \frac{f^{(n+1)}(\xi_t)}{(n+1)!} \prod_{i=0}^{n} (t - t_i).$$

Attention ! Comme ξ_t est inconnu, la formule fournit une estimation de l'erreur, pas sa valeur exacte !

4. Une majoration de l'erreur

Posons $E_n = \max_{t\in[a,b]} |f(t) - p_n(t)|$. Alors, la formule précédente donne

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

Or, comme $t, t_i \in [a, b]$, on obtient la majoration :

$$E_n \leq \frac{(b-a)^{n+1}}{(n+1)!} \max_{t\in[a,b]} |f^{(n+1)}(t)|.$$

Cette majoration peut tendre vers 0 lorsque n augmente.

Exemple. Si f est C^∞ !

Attention. Si les dérivées de f ne sont pas bornées, augmenter n ne change rien et peut apporter des oscillations, phénomène de Runge.

5. La base de Newton

Précédemment, nous avons construit l'unique polynôme d'interpolation dans $\mathbb{P}_n$ via les polynômes de la base de Lagrange. Cependant, nous nous intéressons dans la suite à une autre base de $\mathbb{P}_n$ qui permet de calculer ce polynôme de manière plus efficace. Bien entendu, comme le polynôme d'interpolation est unique, d'après le Théorème 5.2.1, nous devrions retrouver le même polynôme à la fin...

5. La base de Newton

Pour des nœuds ordonnés $t_0, \ldots, t_n$, on introduit :

$$1, \quad (t - t_0), (t - t_0)(t - t_1), \ldots, \quad \prod_{j=0}^{n-1} (t - t_j).$$

Ces $n + 1$ polynômes forment une base de $\mathbb{P}_n$, appelée base de Newton. Dans cette base, le polynôme interpolateur s'écrit :

$$p_n(t) = c_0 + c_1(t - t_0) + \cdots + c_n \prod_{j=0}^{n-1} (t - t_j).$$

Question. Relations entre $c_0, \ldots, c_n$ et $f(t_0), \ldots, f(t_n)$?

7. Différences divisées et coefficients de Newton

Pour $t_0, t_1, t_2, \ldots$ des points distincts. On appelle différence divisée d'ordre 0, 1, 2, $\ldots$, k, les expressions suivantes :

$$f[t_0] = f(t_0), \quad f[t_1] = f(t_1), \quad f[t_2] = f(t_2), \ldots \quad \text{ordre } 0,$$

$$f[t_0, t_1] = \frac{f[t_0] - f[t_1]}{t_0 - t_1}, \quad f[t_1, t_2] = \frac{f[t_1] - f[t_2]}{t_1 - t_2}, \ldots \quad \text{ordre } 1,$$

$$f[t_0, t_1, t_2] = \frac{f[t_0, t_1] - f[t_1, t_2]}{t_0 - t_2}, \ldots \quad \text{ordre } 2,$$

et à l'ordre $k \geq 3$

$$f[t_0, \ldots, t_k] = \frac{f[t_0, \ldots, t_{k-1}] - f[t_1, \ldots, t_k]}{t_0 - t_k}.$$

Remarque. une différence divisée ne dépend pas de l'ordre de ses arguments.

7. Différences divisées et coefficients de Newton

Théorème 5.2.4.

Soit f une fonction dont on connaît les valeurs en $n+1$ points distincts $t_0, \ldots, t_n$. Alors le polynôme d'interpolation de f dans la base de Newton s'écrit :

$$\begin{aligned} p_n(t) = & f[t_0] + f[t_0, t_1](t - t_0) \\ & + f[t_0, t_1, t_2](t - t_0)(t - t_1) \\ & + \cdots \\ & + f[t_0, \ldots, t_n] \prod_{j=0}^{n-1} (t - t_j). \end{aligned}$$

À retenir.

Les coefficients du polynôme d'interpolation de f dans la base de Newton sont précisément les

$$c_k = f[t_0, \ldots, t_k], \qquad k = 0, \ldots, n.$$

8. Calcul pratique des différences divisées

On remplit le tableau colonne par colonne :

ordre 0	ordre 1	ordre 2	$\cdots$	ordre n
$f[t_0]$				
$f[t_1]$	$f[t_0, t_1]$			
$f[t_2]$	$f[t_1, t_2]$	$f[t_0, t_1, t_2]$		
$\vdots$	$\vdots$	$\vdots$	$\ddots$	
$f[t_n]$	$f[t_{n-1}, t_n]$	$f[t_{n-2}, t_{n-1}, t_n]$	$\cdots$	$f[t_0, \ldots, t_n]$

- Chaque nouvelle colonne se déduit de la précédente.
- La diagonale supérieure donne les coefficients $c_0, c_1, \ldots, c_n$.
- Le calcul demande $n(n+1)/2$ divisions.

9. Exemple d'interpolation dans la base de Newton

Reprenons les données : $(0,1)$, $(1,2)$ et $(2,0)$. Alors

ordre 0	ordre 1	ordre 2
1		
2	$\dfrac{1-2}{0-1}=1$	
0	$\dfrac{2-0}{1-2}=-2$	$\dfrac{1-(-2)}{0-2}=-\dfrac{3}{2}$

Donc

$$c_0 = 1, \qquad c_1 = 1, \qquad c_2 = -\frac{3}{2},$$

et

$$p_2(t) = 1 + t - \frac{3}{2}t(t-1) = -\frac{3}{2}t^2 + \frac{5}{2}t + 1.$$

Remarque. On retrouve bien le polynôme d'interpolation de l'exemple précédent. C'est normal car le polynôme d'interpolation est unique !

Question. Intérêt de la base de Newton ?

6. Newton vs Lagrange

Définissons les polynômes tronqués (à l'ordre k)

$$p_k(t) = c_0 + c_1(t - t_0) + \cdots + c_k \prod_{j=0}^{k-1}(t - t_j) \quad \forall k = 0, \ldots, n.$$

Alors p_k interpole les $k + 1$ premières données :

$$p_k(t_i) = f(t_i), \qquad i = 0, \ldots, k,$$

et en plus

$$p_n(t) = p_k(t) + c_{k+1} \prod_{j=0}^{k}(t - t_j) + \ldots + \prod_{j=0}^{n-1}(t - t_j).$$

Ainsi, si l'on connaît p_{n-1} et que l'on ajoute le nœud t_n, on conserve les coefficients précédents et l'on ajoute seulement un terme :

$$p_n(t) = p_{n-1}(t) + c_n \prod_{j=0}^{n-1}(t - t_j).$$

10. Algorithme de Horner

La forme de Newton peut être réécrite sous forme emboîtée :

$$p_n(t) = c_0 + (t - t_0)\big(c_1 + (t - t_1)\big(c_2 + \cdots + (t - t_{n-1})c_n\big)\cdots\big).$$

Algorithme de Horner

1. Initialiser $p \leftarrow c_n$.
2. Pour $k = n-1, n-2, \ldots, 0$, effectuer
$$p \leftarrow c_k + (t - t_k)p.$$
3. Le résultat final est p_n.

10. Algorithme de Horner

La forme de Newton peut être réécrite sous forme emboîtée :

$$p_n(t) = c_0 + (t - t_0)\big(c_1 + (t - t_1)\big(c_2 + \cdots + (t - t_{n-1})c_n\big)\cdots\big).$$

Algorithme de Horner

1. Initialiser $p \leftarrow c_n$.
2. Pour $k = n-1, n-2, \ldots, 0$, effectuer
$$p \leftarrow c_k + (t - t_k)p.$$
3. Le résultat final est p_n.

À retenir.

La base de Lagrange est adaptée à l'expression explicite du polynôme, mais moins pratique pour le calcul de ses valeurs. En pratique, il est plus intéressant de travailler dans la base de Newton.