TELECOM Nancy *(ESIAL)*
Maths Numériques

feuille 5 : interpolation (fin)

Exercice 1. *recherche du bon intervalle*

Lorsqu'on utilise une interpolation polynomiale par morceaux on a besoin de trouver le bon intervalle. Comment faire cette opération rapidement lorsque :

1. les abscisses d'interpolation sont équidistantes ($x_{i+1} - x_i = h_i = Cte,\ \forall i$) ?
2. les abscisses ne sont pas équidistantes ? Dans ce cas, la bonne méthode est la dichotomie et si vous ne l'avez pas encore vue en informatique elle est très simple : supposons que l'abscisse d'évaluation x est comprise entre les abscisses x_{i_l} et et x_{i_h} (au début de l'algorithme $i_l = 1$ et $i_h = n$) on a donc le choix entre $i_h - i_l$ intervalles (au début $n-1$ intervalles) mais avec un seul test on peut retirer la moitiée de ces intervalles de la recherche : il s'agit de tester si x par rapport à l'abscisse x_{i_m} où $i_m = \frac{i_l+i_h}{2}$ (la division étant entière). En fonction du résultat du test, on met à jour i_l ou bien i_h et l'on recommence l'opération tant que $i_h - i_l > 1$.

(a) écrire cet algorithme ;

(b) calculer sa complexité ; pour cela on bornera le nombre d'itérations : soit N le nombre d'intervalles au début (ici $N = n-1$), on pose N_k le nombre d'intervalles restant à examiner à la suite de l'itération k de l'algorithme (par exemple si N est pair on a $N_1 = N/2$ et si N est impair on aura soit $N_1 = (N-1)/2$ soit $N_1 = (N+1)/2$). On notera $N_0 = N$.
   - Montrer que :
   $$N_k \leq \lceil N_{k-1}/2 \rceil, \text{ pour } k \geq 1$$
   - On considère maintenant la suite $(\tilde{N}_k)_{k\geq 0}$ définie par $\tilde{N}_0 = N$ et $\tilde{N}_k = \lceil \tilde{N}_{k-1}/2 \rceil, k \geq 1$ (il est clair que $\tilde{N}_k \geq N_k, \forall k \geq 0$). Soit K le plus petit entier tel que $2^K \geq N$ (on a donc $2^{K-1} < N$), montrer que si on part avec $\tilde{N}_0 = 2^K$ alors K est le plus petit entier pour lequel $(\tilde{N}_K) = 1$. En déduire que le nombre d'itérations de l'algorithme est borné par $\lceil lb(N) \rceil$ où lb désigne le log base 2 (8 itérations suffisent pour 256 intervalles, 16 itérations pour 65536 intervalles, 32 itérations pour 4294967296 intervalles, ...).

Exercice 2. *Interpolation de Lagrange-Hermite par morceaux et interpolation par spline cubique*

1. Montrer que le polynôme d'interpolation de Lagrange-Hermite p construit avec deux points, cad à partir de $x_0,\ x_1,\ y_0,\ y_1,\ y_0',\ y_1'$ et tel que $p(x_i) = y_i$ et $p'(x_i) = y_i'$ pour $i = 0, 1$ s'écrit $p(x) = y_0 H_0(x) + y_1 H_1(x) + y_0' K_0(x) + y_1' K_1(x)$ avec :
$$\begin{array}{ll} H_0(x) = \frac{(h_1+2(x-x_0))(x-x_1)^2}{h_1^3} & K_0(x) = \frac{(x-x_0)(x-x_1)^2}{h_1^2} \\ H_1(x) = \frac{(-h_1+2(x-x_1))(x-x_0)^2}{(-h_1)^3} & K_1(x) = \frac{(x-x_1)(x-x_0)^2}{(-h_1)^2} \end{array}$$
où $h_1 = x_1 - x_0$. On remarquera la symétrie suivante : si on pose $H_0(x) := H(x, x_0, x_1)$ et $K_0(x) := K(x, x_0, x_1)$ alors $H_1(x) = H(x, x_1, x_0)$ et $K_1(x) := K(x, x_1, x_0)$. Ceci permet de limiter les calculs par la suite.
2. Soit f une fonction 4 fois continûment dérivable sur $[x_0, x_1]$ et telle que $f(x_0) = y_0,\ f(x_1) = y_1,\ f'(x_0) = y_0'$ et $f'(x_1) = y_1'$. Montrer que :
$$\|p - f\|_\infty \leq \frac{h_1^4}{384} \|f^{(4)}\|_\infty, \text{ où } \|f\|_\infty := \max_{x\in[x_0,x_1]} |f(x)|$$
3. La fonction f est maintenant définie sur un intervalle $[a, b]$ et est toujours quatre fois continûment dérivable sur cet intervalle. L'intervalle $[a, b]$ est divisé en n sous-intervalles $I_i := [x_{i-1}, x_i]$ avec $x_0 := a < x_1 < \dots < x_{n-1} < x_n := b$. Une possibilité pour obtenir un interpolant par morceaux qui soit une fois continûment dérivable est d'utiliser sur chaque intervalle I_i un polynôme de Lagrange-Hermite p_i construit uniquement avec ces 2 points. On a donc $y_i = f(x_i)$ et $y_i' = f'(x_i),\ i = 0, \dots, n$. On appelle g la fonction interpolante ainsi obtenue ($g_{|I_i} := p_i$).

(a) Soit $t \in [a,b]$ décrire succinctement les deux étapes permettant de calculer $g(t)$.

(b) Montrer que :

$$\|g-f\|_\infty \leq \frac{h^4}{384}\|f^{(4)}\|_\infty, \text{ où } h := \max_i h_i \ (h_i := x_i - x_{i-1}) \text{ et } \|f\|_\infty := \max_{x\in[a,b]} |f(x)|$$

(c) En fait si la dérivée quatrième de f n'est pas identiquement nulle sur $[a,b]$ (et donc si $f \notin \mathcal{P}_3$) et si on utilise une partition uniforme de $[a,b]$ (cad $h_i := h = (b-a)/n$), on peut montrer que cette majoration est quasi optimale (i.e. $\|g-f\|_\infty \simeq \frac{h^4}{384}\|f^{(4)}\|_\infty$). Que se passe-t-il pour l'erreur si on utilise deux fois plus d'intervalles ?

4. Dans la pratique si les points (x_i, y_i) sont issus d'une expérience, on ne dispose généralement pas des dérivées y'_i. Une possibilité est de les estimer à partir des données (x_i, y_i), par exemple avec $y'_i := q'_i(x_i)$ où q_i est le polynôme qui interpole en x_{i-1}, x_i et x_{i+1}. Ceci conduit à l'interpolation cubique de Bessel (on obtient une convergence en $O(h^3)$). En fait il est possible de garder une convergence en $O(h^4)$ et d'avoir un interpolant deux continûment dérivable en utilisant une spline cubique s. Pour cela on pose $s_{|I_i} := p_i$ où p_i est le polynôme de Lagrange-Hermite construit avec $x_{i-1}, x_i, y_{i-1}, y_i$ et $d_{i-1} := y'_{i-1}, d_i := y'_i$ (dans la suite nous appelerons d_i les y'_i) ce qui assure déjà des raccords C^1. Cependant pour que s soit bien une spline cubique, il faut imposer les $n-1$ conditions supplémentaires suivantes :

$$s''(x_i-) = s''(x_i+), \quad (\text{soit } p''_i(x_i) = p''_{i+1}(x_i)), i = 1, \dots, n-1 \tag{1}$$

de sorte à obtenir des raccords C^2. Les calculs suivants ont pour but d'arriver rapidement à écrire les $n-1$ équations précédentes comme autant d'équations portant sur les d_i.

(a) Montrer (ou admettre) que :

$$H'_0(x) = \frac{2(x-x_1)(h_1 + 2(x-x_0) + (x-x_1))}{h_1^3} \text{ et } H''_0(x) = \frac{2(h_1 + 2(x-x_0) + 4(x-x_1))}{h_1^3}$$

(b) Par symétrie (cf question 1) en déduire que $H''_1(x) = \dfrac{2(h_1 - 2(x-x_1) - 4(x-x_0))}{h_1^3}$

(c) Sur le même principe, montrer que $K''_0(x) = \frac{4(x-x_1)+2(x-x_0)}{h_1^2}$ et en déduire $K''_1(x)$ par symétrie.

(d) Maintenant calculer $H''_0(x_0)$, $H''_0(x_1)$, $H''_1(x_0)$, $H''_1(x_1)$, $K''_0(x_0)$, $K''_0(x_1)$, $K''_1(x_0)$, $K''_1(x_1)$, et en déduire que :

$$p''_1(x_0) = y_0\left(\frac{-6}{h_1^2}\right) + y_1\left(\frac{6}{h_1^2}\right) + d_0\left(\frac{-4}{h_1}\right) + d_1\left(\frac{-2}{h_1}\right)$$

et que :

$$p''_1(x_1) = y_0\left(\frac{6}{h_1^2}\right) + y_1\left(\frac{-6}{h_1^2}\right) + d_0\left(\frac{2}{h_1}\right) + d_1\left(\frac{4}{h_1}\right)$$

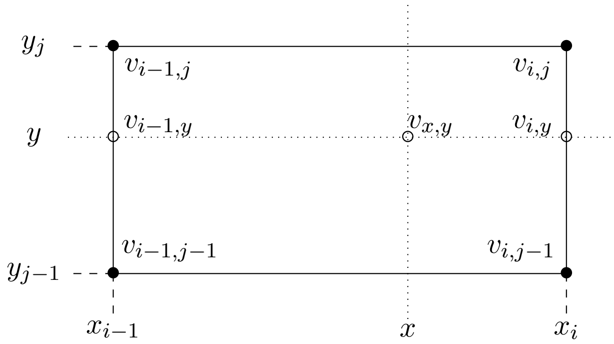
(e) En déduire pour $i \in [\![1, n-1]\!]$ les deux quantités $p''_i(x_i)$ et $p''_{i+1}(x_i)$ et montrer que l'équation (1) s'écrit :

$$\left(\frac{1}{h_i}\right)d_{i-1} + \left(\frac{2}{h_i} + \frac{2}{h_{i+1}}\right)d_i + \left(\frac{1}{h_{i+1}}\right)d_{i+1} = \frac{3}{h_i}\left(\frac{y_i - y_{i-1}}{h_i}\right) + \frac{3}{h_{i+1}}\left(\frac{y_{i+1} - y_i}{h_{i+1}}\right) \tag{2}$$

(f) En ressemblant ces $n-1$ équations on obtient un système linéaire mais avec $n+1$ inconnues. Pour obtenir un système carré, il existe plusieurs possibilités dont les deux suivantes : (i) si on connaît la dérivée exacte aux 2 extrémités, on impose alors $d_0 = f'(a)$ et $d_n = f'(b)$, (ii) on peut estimer ces dérivées avec deux polynômes d'interpolation $d_0 := q'_a(a)$ où q_a interpole $(x_i, y_i), i \in [\![0, 3]\!]$ et $d_n := q'_b(b)$ où q_b interpole $(x_i, y_i), i \in [\![n-3, n]\!]$. Ainsi d_0 et d_n ne sont plus des inconnues et on obtient un système linéaire carré. Ecrire la matrice A et le second membre de ce système. On peut montrer que A est inversible. En fait A est symétrique et définie positive et la méthode de Gauss sans échange d'équations convient parfaitement pour résoudre un tel système. On peut d'ailleurs adapter cette méthode au caractère symétrique et défini positif de la matrice (cf méthode de Choleski, exercice 7 feuille 2) ainsi qu'au caractère tridiagonal et obtenir une complexité en $O(n)$ au lieu de $n^3/3$.

Exercice 3. *Interpolation bilinéaire*

Ce type d'interpolation 2d est l'un des plus simples possibles et correspond au produit tensoriel de l'interpolation linéaire par morceaux 1d. On a donc une grille formée de points de la forme $(x_i, y_j)_{1\leq i\leq m, 1\leq j\leq n}$, les valeurs en ces points sont notées $v_{i,j}$.



Pour interpoler, c'est à dire pour obtenir la valeur en (x, y) (où $x \in [x_1, x_m]$ et $y \in [y_1, y_n]$), il faut d'abord repérer le rectangle qui contient ce point (cad les deux intervalles en "x" et en "y", ce qui peut se faire avec les techniques vues dans l'exercice 1) puis l'interpolation peut se comprendre intuitivement de la façon suivante :

— on commence par interpoler linéairement en "y", cad qu'on calcule :
  — $v_{i-1,y}$ par interpolation linéaire de $v_{i-1,j-1}$ et $v_{i-1,j}$
  — et $v_{i,y}$ par interpolation linéaire de $v_{i,j-1}$ et $v_{i,j}$;

— ensuite on interpole linéairement en "x" à partir de $v_{i-1,y}$ et $v_{i,y}$ pour obtenir $v_{x,y}$.

Mais on peut aussi commencer à interpoler en "x" ce qui donne la même chose.

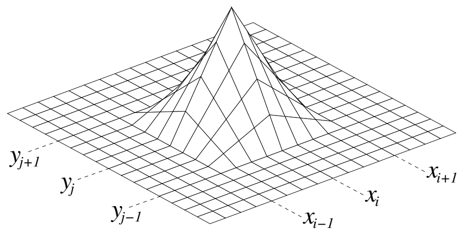
1. Montrer que cette interpolation conduit à :
$$v_{x,y} = v_{i-1,j-1}\phi_{i-1,j-1}(x, y) + v_{i-1,j}\phi_{i-1,j}(x, y) + v_{i,j-1}\phi_{i,j-1}(x, y) + v_{i,j}\phi_{i,j}(x, y) \tag{3}$$
$$\text{avec :}\quad \begin{array}{rcl} \phi_{i-1,j-1}(x,y) &=& \left(\frac{x-x_i}{x_{i-1}-x_i}\right)\left(\frac{y-y_j}{y_{j-1}-y_j}\right) \\ \phi_{i-1,j}(x,y) &=& \left(\frac{x-x_i}{x_{i-1}-x_i}\right)\left(\frac{y-y_{j-1}}{y_j-y_{j-1}}\right) \\ \phi_{i,j-1}(x,y) &=& \left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right)\left(\frac{y-y_j}{y_{j-1}-y_j}\right) \\ \phi_{i,j}(x,y) &=& \left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right)\left(\frac{y-y_{j-1}}{y_j-y_{j-1}}\right) \end{array}$$
ces fonctions $\phi_{i,j}$ sont parfois appelées "base locale" (sous-entendue sur le rectangle $[x_{i-1}, x_i] \times [y_{j-1}, y_j]$) par opposion aux fonctions de base globales définies ci-après.

2. On se rend compte qu'en se donnant un jeu de valeurs $(v_{i,j})_{1\leq i\leq m,1\leq j\leq n}$ on définit de cette façon une et une seule fonction définie sur $[x_1, x_n] \times [y_1, y_n]$ (et à valeur dans $\mathbb{R}$) qu'on appelera $\mathcal{V}(x, y)$ dont les $m \times n$ "paramètres" $v_{i,j}$ sont les d.d.l.. L'espace de fonctions $\mathcal{B}$ ainsi engendré semble être de dimension $m \times n$ et il est assez facile de montrer que c'est bien un espace vectoriel : si on se donne un autre jeu de valeurs $(w_{i,j})_{1\leq i\leq m,1\leq j\leq n}$, on définit une autre fonction disons $\mathcal{W}(x, y)$. Montrer que la fonction $(x, y) \longmapsto \alpha\mathcal{V}(x, y) + \beta\mathcal{W}(x, y)$ est bien dans $\mathcal{B}$ (ses "ddl" étant $(\alpha v_{i,j} + \beta w_{i,j})_{1\leq i\leq m,1\leq j\leq n}$).

3. La suite de l'exercice consiste à exhiber une base de cet espace. Montrer que :
$$v(x, y) = \sum_{i=1}^{m}\sum_{j=1}^{n} v_{i,j}\Phi_{i,j}(x, y), \quad \forall v \in \mathcal{B} \tag{4}$$

où si (x_i, y_j) est un point intérieur (cad $1 < i < m$ et $1 < j < n$) alors $\Phi_{i,j}(x,y) =$

$$\begin{cases} \left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right)\left(\frac{y-y_{j-1}}{y_j-y_{j-1}}\right) & \text{si } (x,y) \in [x_{i-1}, x_i] \times [y_{j-1}, y_j] \\ \left(\frac{x-x_{i+1}}{x_i-x_{i+1}}\right)\left(\frac{y-y_{j-1}}{y_j-y_{j-1}}\right) & \text{si } (x,y) \in [x_i, x_{i+1}] \times [y_{j-1}, y_j] \\ \left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right)\left(\frac{y-y_{j+1}}{y_j-y_{j+1}}\right) & \text{si } (x,y) \in [x_{i-1}, x_i] \times [y_j, y_{j+1}] \\ \left(\frac{x-x_{i+1}}{x_i-x_{i+1}}\right)\left(\frac{y-y_{j+1}}{y_j-y_{j+1}}\right) & \text{si } (x,y) \in [x_i, x_{i+1}] \times [y_j, y_{j+1}] \\ 0 & \text{ailleurs} \end{cases}$$



les fonctions de base correspondant aux points du bord répondant aussi à cette même définition là où elle a un sens. Aide : on peut vérifier que (4) donne la bonne expression pour chaque rectangle (on doit obtenir (3) en considérant $[x_{i-1}, x_i] \times [y_{j-1}, y_j]$).

4. (*) Montrer que $\mathcal{B} = \mathcal{C}_x \otimes \mathcal{C}_y$ où $\mathcal{C}_x$ est l'espace des fonctions "chapeaux" défini par les $(x_i)_{1 \le i \le m}$ et $\mathcal{C}_y$ l'espace des fonctions "chapeaux" défini par les $(y_j)_{1 \le j \le n}$

Remarque : si exhiber la base (naturelle) de cet espace de fonctions $\mathcal{B}$ est peu utile pour l'interpolation, c'est par contre important pour l'approximation au sens des moindres carrés.