Université de Toulouse III, FSI - Département de Mathématiques
Licence 2 parcours spécial, module "Calcul scientifique" , 2013-2014

Projet "Interpolation par splines cubiques" Novembre 2013

Dans ce projet, on s'intéresse à la possibilité de trouver une courbe passant par des points donnés. On a bien sûr vu en cours des méthodes d'interpolation de base que l'on va coder ici et comparer à d'autres méthodes ayant des propriétés intéressantes. On a en particulier la volonté de tracer une courbe *très régulière* passant par les points donnés. On va ici fabriquer une méthode d'interpolation par splines cubiques, mise en place pour du dessin assisté par ordinateur. Le nom même de "spline" est à rapprocher du mot "cerce", latte souple en bois bien de fil qui sert à tracer les courbes quelconques (courbes harmonieuses passant par un certain nombre de points mais ne pouvant être tracées à l'aide du compas), utilisée par les menuisiers, charpentiers...

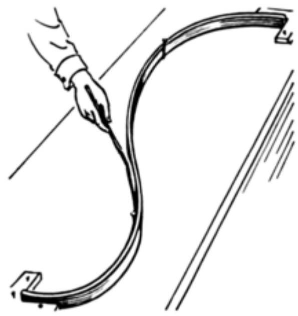

FIGURE 1 – Cerce

1 Un petit détour par l'interpolation de Hermite pour 2 points

Vous vous souvenez sans doute que, étant donnés n réels distincts $x_1 < x_2 < \cdots < x_n$, et n valeurs correspondantes $y_1, \ldots, y_n$ (à noter qu'il peut y avoir des valeurs égales pour le coup), l'interpolation de Lagrange vise à construire un polynôme P de degré $n-1$ tel que pour tout $i = 1 \ldots n,\ P(x_i) = y_i$. Dans cette premieère partie, on propose d'imposer aussi la valeur des dérivées aux points $x_1, \ldots x_n$. On se donne pour cela des valeurs $z_1, \ldots, z_n$ et on cherche à montrer qu'il existe un unique polynône Q de degré inférieur ou égal à $2n-1$ tel que

$$\forall i = 1, \ldots, n,\ Q(x_i) = y_i, \quad \text{et} \quad Q'(x_i) = z_i.$$

Pour la suite du projet, on cherche à déterminer le polynôme d'interpolation de Hermite de degré 3, associé à 2 points $x_G < x_D$ et qui vérifie donc, étant donnés y_G, y_D, z_G, z_D quatre réels donnés :

$$H(x_G) = y_G,\ H(x_D) = y_D,\ H'(x_G) = z_G,\ H'(x_D) = z_D.$$

On cherchera une expression de H, nécessaire pour la suite du projet sous la forme

$$H(X) = Q_0(X) + (X - x_G)(X - x_D)Q_1(X)$$

où Q_0 et Q_1 sont des polynômes de degré inférieur ou égal à 1 que l'on déterminera dans la base des polynômes $X - x_G$ et $X - x_D$.
Calculer alors $H"(x_G)$ et $H"(x_D)$ en fonction des données x_G, x_D, y_G, y_D, z_G et z_D.

2 Existence et unicité de la spline cubique contrainte interpolante.

On se donne des points distincts : $x_1 < \cdots < x_n$ (pas nécessairement équidistants) ainsi que n valeurs $y_1, \ldots y_n$, et deux réels supplémentaires z_1 et z_n. Le but du projet est de construire l'unique fonction $\pi : [x_1, x_n] \to \mathbb{R}$ vérifiant :

(P1) π est de classe $\mathcal{C}^2$ sur $[x_1, x_n]$

(P2) π interpole les valeurs données, *i.e* $\forall i = 1, \ldots, n,\ \pi(x_i) = y_i$

(P3) La restriction de π à chaque intervalle $[x_i, x_{i+1}]$ est un polynôme de degré inférieur ou égal à 3

(P4) $\pi'(x_1) = z_1$ et $\pi'(x_n) = z_n$.

Une fonction vérifiant les propriétés *(P1)-(P2)-(P3)* est appelée *spline cubique interpolante*. Si de plus elle vérifie *(P4)* (*i.e* que l'on impose la valeur des dérivées aux bords) alors on parle de spline *contrainte*.

— **Étape 1 :** On commence par montrer que la fonction π est entièrement déterminée par la donnée des valeurs $\pi(x_i)$ et $\pi'(x_i)$ pour $i = 1 \ldots n$. On se donne donc $(y_i)_{i=1\ldots n}$ et $(z_i)_{i=1\ldots n}$. Déterminer la fonction π vérifiant P2, P3 et P4 sachant que $\pi'(x_i) = z_i$ pour $i = 1 \ldots n$. Pour cela, on détermine π sur $[x_i, x_{i+1}]$ en utilisant les polynômes de Hermite de la 1ere partie.

— **Étape 2 :** On cherche maintenant à déterminer les valeurs $z_2, \ldots z_{n-1}$ que π' doit prendre aux noeuds x_i pour que la fonction π soit de classe $\mathcal{C}^2$ (et vérifie donc la propreété *(P1)*). Calculer π'' sur $[x_i, x_{i+1}]$ et donc sur $[x_{i-1}, x_i]$ et en déduire les conditions de raccord de la dérivée seconde en x_i. On obtient le système linéaire $n-2 \times n-2$ suivant :

$$\begin{pmatrix} 4 & 1 & 0 & \ldots & 0 \\ 1 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \ldots & 0 & 1 & 4 \end{pmatrix} \begin{pmatrix} z_2 \\ \vdots \\ \vdots \\ \vdots \\ z_{n-1} \end{pmatrix} = \begin{pmatrix} 3\frac{y_2-y_1}{x_2-x_1} + 3\frac{y_3-y_2}{x_3-x_2} - z_1 \\ \vdots \\ 3\frac{y_i-y_{i-1}}{x_i-x_{i-1}} + 3\frac{y_{i+1}-y_i}{x_{i+1}-x_i} \\ \vdots \\ 3\frac{y_{n-1}-y_{n-2}}{x_{n-1}-x_{n-2}} + 3\frac{y_n-y_{n-1}}{x_n-x_{n-1}} - z_n \end{pmatrix}. \qquad (1)$$

— **Étape 3 :** A partir de la formule calculée à l'étape 1 et des valeurs de $z_2, \ldots, z_{n-1}$ obtenues en résolvant le système (1), implémenter une fonction Python prenant en argument les données du problème : le vecteur $X = (x_1, \ldots, x_n)$, le vecteur $Y = (y_1, \ldots, y_n)$ ainsi que z_1 et z_n et donnant en sortie la fonction π qui donne la spline cubique interpolante contrainte.

3 Et maintenant...

Munis de cette fonction de splines cubiques, on peut montrer l'intérêt de l'interpolation par splines cubiques par rapport à l'interpolation de Lagrange "classique".
On pourra par exemple approcher la fonction de Runge : $x \in [-1,1] \mapsto \frac{1}{1+x^2}$ par son interpolée de Lagrange et son interpolée spline cubique en 7 points équidistants par exemple. Qu'observe-t-on ? Que se passe-t-il si on interpole avec 15 points ? Et si on utilise de l'interpolation par splines plutôt ?

On peut aussi s'amuser un peu... on peut essayer de jouer au designer, entrer des points à la souris en Python et trouver une spline contrainte passant par ces points, tuer en modifiant les valeurs des dérivées aux bords...
NB : Comment entrer des points à la souris avec Python ? (et récupérer leurs coordonnées) :

```
import matplotlib.pyplot as plt
import numpy as np
#t = np.arange(10)
plt.plot(0,0)
Nb=input("Combien de Points ?")
print("Please click")
x = plt.ginput(Nb)
print("clicked", x)
plt.show()
```