Approximation polynômiale

Le but de ce projet est d'étudier quelques méthodes d'approximations de fonctions par des polynômes. Les applications sont multiples : représentation d'un signal numérique, résolution approchée d'équations où apparaît f, compression d'informations, représentation d'un objet physique...

Interpolation de Lagrange

On considère une fonction f dont on ne connaît que certaines valeurs $f(x_0), \ldots, f(x_n)$. Le principe de l'interpolation est d'approcher f par le polynôme P de degré au plus n tel que $\forall i,\ P(x_i) = f(x_i)$. On dit alors qu'on a interpolé f aux points $x_0, \ldots, x_n$.

Nous ne considérerons ici que des fonctions définies sur l'intervalle $[0, 1]$. Pour une telle fonction f et des points $x_0, \ldots, x_n$ de $[0, 1]$, le polynôme interpolateur de f est donné par

$$P_n(f)(X) = \sum_{i=0}^{n} f(x_i) \prod_{j \neq i} \frac{X - x_j}{x_i - x_j}$$

1. Écrire un programme *interpolation(f,l)*, où l sera une liste $[x_0, \ldots, x_n]$ de points de $[0, 1]$, qui renvoie le polynôme interpolateur de f en ces points.
2. Vérifier votre programme en le testant sur des fonctions f polynômiales puis sur d'autres fonctions (dont certaines varaitn beaucoup sur $[0, 1]$.
3. La qualité de l'approximation est mesurée par l'erreur $e = \sup_x |f(x) - P(f)(x)|$, où P est un polynôme interpolateur de f.
   Écrire un programme *erreur(f,l)* qui calcule pour une fonction f et une liste l l'erreur e_n de l'interpolation correspondante.

   Interpolation aux points de Lagrange :
   Les points de Lagrange sont des points équirépartis de l'intervalle $[0, 1]$. Pour un entier n donné, ce sont les points $\frac{k}{n}$ avec $k = 0, \cdots, n$.
4. Écrire un programme *Lagrange(n)* qui renvoie la liste des $n + 1$ points de Lagrange.
5. Tracer sur un même graphe une fonction f et ses approximations obtenues avec les points de Lagrange pour $n = 1, \ldots, 5$.
6. Représenter la suite des erreurs e_n en fonction de n.
7. Cette suite converge-t-elle vers 0 ? Décrire le phénomène (appelé phénomène de Runge) que l'on observe.

Interpolation aux points de Tchebychev :

Les points de Tchebychev sont, pour un entier n donné, les points définis par $x_k = \frac{1}{2} - \frac{1}{2}\cos((k+\frac{1}{2})\pi/(n+1))$ pour $k = 0, \ldots, n$.

8. Reprendre pour les points de Tchebychev toutes les questions de la partie précédente.
9. Comparer les qualités des interpolations avec les points de Lagrange et de Tchebychev.

Fonctions splines

Les fonctions splines permettent d'effectuer une interpolation à l'aide de courbes polynomiales par morceaux.

1. Recommencer l'étude précédente avec ces fonctions. Quels sont leurs avantages et inconvénients ?
2. Écrire un programme n'utilisant pas la commande *Spline* qui calcule la meilleure courbe spline (en un sens que l'on précisera) passant par trois points donnés.

Moindres carrés

On peut également approcher une fonction sans nécessairement faire d'interpolation. La méthode des moindres carrés permet de déterminer le polynôme le plus proche (en un certain sens) des points que l'on considère.

Recommencer l'étude précédente avec cette méthode. La comparer aux précédentes.

Fonctions utiles : PolynomialInterpolation, seq, Spline, LeastSquares