DIF Clément

LOXOL Nicolas

Compte rendu TP : Interpolation polynomiale

1. Rappel des méthodes :

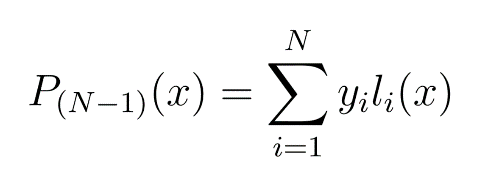
Les méthodes d'interpolation et d'approximation ont pour objectif de retrouver l'équation mathématique d'une courbe traversant une liste de points mesurés. Nous allons vous présenter deux méthodes d’interpolation.

Interpolation de Lagrange :

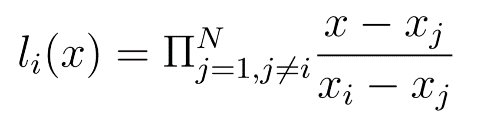
Soit Pn<N (x) = anxn + an-1xn-1 + … + a1x + ao un polynôme avec 1<i<N. On sait que le point (xi,yi) vérifie Pn<N (xi) = yi .

On veut calculer les coefficient an, an-1, ..., a0

On écrit le polynôme d’une manière différente :



Ou li est la fonction définie par :



On peut remarquer que li(xj) vaut 0 pour j 6= i et 1 sinon. La fonction li(x) est le produit de N −1 polynômes de degré 1, c'est donc un polynôme de degré N −1

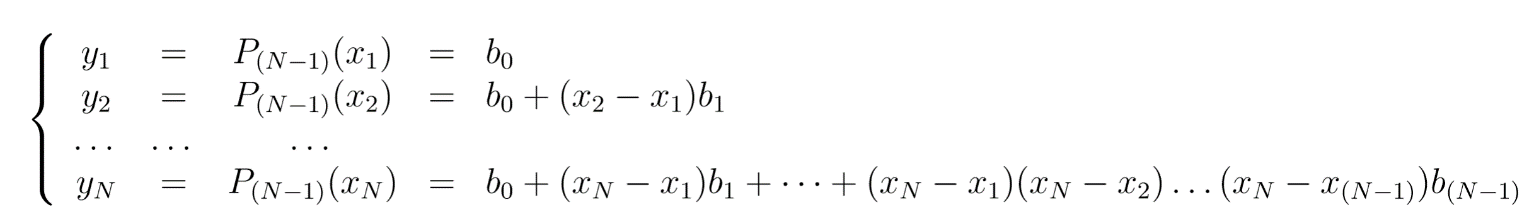
Interpolation de Newton :

Le polynôme de Newton est aussi un polynôme d’interpolation. Celui – ci peut s’écrire :



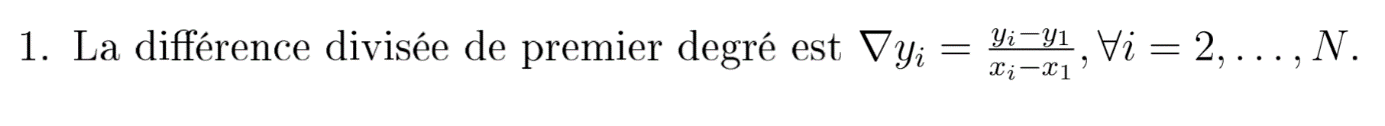
On recherche les coefficients bo,b1,.. tels que PN-1 (xi) = yi avec 1<i<N.

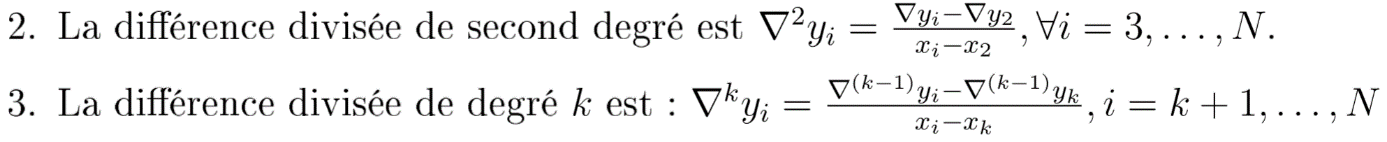
Si on remplace x par xi, on obtient le système linéaire suivant :



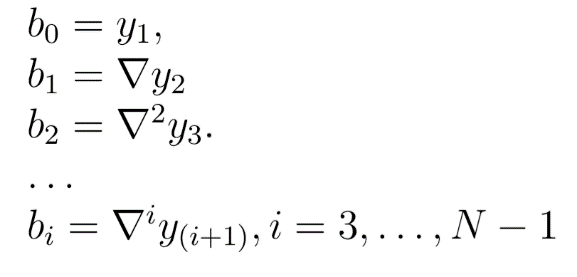
On peut remarquer que la matrice du système est triangulaire inférieure et qu'il peut être résolu par substitution connaissant les N points (xi,yi). Cependant la résolution par l'utilisation des différences divisées est plus rapide.

On utilise la définition de la différence divisée à l’ordre 1,2,…,k :

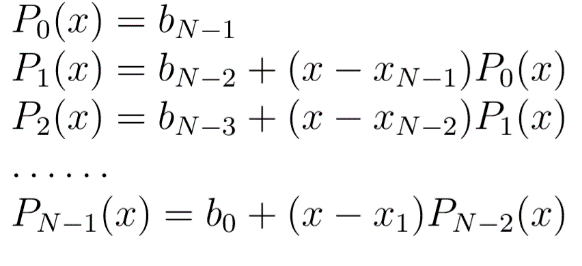




On obtient tous les coefficients du polynôme par les formules suivantes. Celle-ci étant déterminées par récurrence à partir du système défini précédemment.



Le polynôme PN-1 est alors obtenu de la manière suivante :



Remarque :

-Dans ce TP nous traiterons les fonctions polynomiales comme unique cas.

-Le théorème d’unisolvance (« Etant donnés un ensemble N de points (xi,yi), il existe un unique polynôme PN-1 de degré maximum (N-1) qui passe par les N points » ) répond à la question de l’unicité du polynôme.

-D’autres méthodes plus complexes réalisent l’interpolation. On a parfois recours à des découpages de l’ensemble de point en sous ensemble qu’on interpole. De plus on fait en sorte que les fonctions obtenues se touchent sur l’ensemble de point.

1. Présentation du programme :

Notre programme est constitué de plusieurs parties distinctes :

* Les includes afin d’utiliser les fonctions de la bibliothèque standard.
* Les prototypes qui se séparent en deux fonctions ; l’une réalisant l’interpolation grâce à la méthode de Lagrange l’autre grâce à Newton.
* La fonction main dans laquelle sont injectés nos jeux d’essais et les fonctions définies dans les prototypes.
* Les fonctions présentées dans les prototypes :
  + *lagrange* 🡪 cette fonction réalisent l’interpolation de Lagrange ; elle prend en paramètre (xi,yi) respectivement *double \*x* et *double \*y* puis le nombre de point (*int n*) et pour finir le point que l’on souhaite déterminer (*double a*).

Remarque :

-La variable *li* symbolise la fonction mathématique li(x) énoncé dans le premier paragraphe.

- A la fin de chaque boucle, on prend soin à remettre *li* à 1 pour éviter toute erreur de calcul.

-Nous retournons *somme* qui est à la fin de notre fonction la valeur prise par la fonction pour le point *a*.

1. Jeux d’essais et graphiques :

Nous utiliserons comme jeux d’essais les tableaux de donnée présent au verso du TP. Nous ajouterons a cela un tableau avec beaucoup plus de cellule donc plus de point (x,y) afin de constater l’oscillation des méthodes. Par conséquent nous ne ferons varier que le nombre de point.

Graphique :

Les graphiques ont été réalisé a partir du langage de programmation python celui-ci assurant une meilleure précision qu’un tableur tel que Excel.

1. Commentaires des jeux d’essais :
2. Conclusion :