ALGORITHMIQUE NUMÉRIQUE

<u>ÉLÉMENTS DE CORRECTION</u>

L3 - DEVOIR DU MARDI 10 AVRIL 2012

2 H

Sans document. Sans calculatrice.

Attention à la gestion de l'espace : toutes vos réponses doivent être consignées directement sur la feuille d'énoncé dans les cadres réservés.
Aucune autre copie ne sera prise en compte.

Vous disposez d'un formulaire en annexe.

Questions (5 pts)

1/ Quelles sont les deux grandes classes d'algorithmes numériques ? Donner un exemple pour chacune.

Cf. cours slide n°12

2/ On souhaite résoudre un système d'équations linéaires en utilisant une méthode directe. Le déterminant de la matrice A, caractéristique du système, est différent de 0 mais très proche. Que suggérez-vous (détaillez votre raisonnement) ?

On souhaite utiliser une méthode discrète, donc une des variantes du pivot de Gauss (seule méthode discrète de résolution de syst. linéaires vus en cours).
Le déterminant de A est proche de zéro, on risque donc une grande instabilité des calculs car le problème est mal conditionné. On a donc tout intérêt à privilégier la méthode du pivot de Gauss total, qui consiste à choisir à chaque étape le pivot le plus grand possible (en valeur absolue) dans la sous matrice considéré.

3/ Comment vérifier a priori qu'un système d'équation linéaire peut être résolu par une méthode itérative ?

D'après le cours (cf. slide 69-70), il suffit que le rayon spectral de la matrice B (où B est la matrice $M^{-1}N$ définie slide 65) soit strictement inférieur à 1.

Le rayon spectral étant difficile/long à calculer, on tentera le plus souvent de vérifier que la matrice A du système proposé soit à diagonale dominante (ou le plus proche possible).

4/ Peut-on inverser une matrice en utilisant la méthode de Gauss avec pivot partiel ? Si oui, expliquez comment. Si non, expliquez pourquoi.

Le pivot de Gauss consiste en général à résoudre une équation matricielle de type Ax=b où la matrice A et le vecteur b sont connus et le vecteur x est «le vecteur des inconnues».

Or, inverser une matrice consiste à trouver, lorsqu'elle existe, la matrice X (appelée aussi A^{-1}) vérifiant AX=I où I est la matrice identité. Ce système peut aussi s'écrire comme un ensemble de systèmes Ax_1=(10...0) ; Ax_2=(010..0) etc. où les xi sont les vecteurs colonnes constitutifs de la matrice X.

On peut donc très bien inverser une matrice A en utilisant la méthode du pivot de Gauss partiel (et même du pivot de Gauss simple ou total) pour tous les vecteurs colonnes constituant sa matrice inverse.

5/ Quelle méthode d'interpolation peut-on utiliser pour approximer le zéro d'une fonction *f* ? Comment ? Quelles doivent être alors les hypothèses sur *f* ?

On rappelle qu'approximer le zéro d'une fonction f consiste à trouver la valeur de x pour laquelle f(x) = 0 sur l'intervalle considéré. On ne peut donc pas utiliser directement une méthode d'interpolation qui permet en principe de retrouver une valeur de f(x) à partir d'un ensemble de points (x,y) donnés.
Mais on peut détourner le problème en «inversant» les rôles des abscisses et des ordonnées.

L'idée consiste donc à utiliser la méthode de Neville sur des points issus d'une discrétisation de f sur l'intervalle donné et pour x = 0 en inversant les rôles des «x» et des «y». Mais il faut s'assurer que la fonction f est bien monotonne sur cet intervalle.

Exercice 1 (5 pts)

1/ Calculez la valeur, en base `10`, du nombre réel `C2E14000` (codé en hexadécimal) selon la représentation IEEE754 simple précision ? Détaillez les calculs.

$(C2E14000)_{16}$

$=$

$(\underbrace{1}_{S}\underbrace{100\ 0010\ 1}_{E}\underbrace{110\ 0001\ 0100\ 0000\ 0000\ 0000}_{M})_2$

S=1 donc le nombre est négatif

0<E<max : il s'agit donc d'un nombre normalisé

On en déduit la valeur a de ce nombre :

$a = (-1)^S(1,M).2^{E-127} = -(1+2^{-1}+2^{-2}+2^{-7}+2^{-9})2^6$

$= -(2^6+2^5+2^4+2^{-1}+2^{-3}) = \boxed{-112,625}$

2/ Expliquer la notion de nombre réel dénormalisé dans le cadre de la représentation IEEE754. Donner la valeur binaire du plus petit nombre normalisé positif représentable en simple précision puis un exemple de nombre dénormalisé positif de valeur inférieure.

Le format dénormalisé permet de représenter de plus petits nombres (en valeur absolue) que le format normalisé, en contrepartie d'une perte de précision.

La formule de conversion devient alors $(-1)^S(0,M)2^{1-E}$

Le format dénormalisé est indiqué par la conjonction E=0 et M≠0.

Le plus petit dénormalisé $0\ \overbrace{0...0}^{7}1\ \overbrace{0...0}^{23} = 2^{-126}$

On notera que ce plus petit normalisé majore tous les normalisés dont par exemple le plus grand (positif) d'entre eux : $0\ 0...0\ 1...1 \approx 2^{-126}$

Exercice 2 (5pts)

Exprimer la fonction *f* décrite par les trois points suivants sous forme quadratique, en utilisant la méthode de Lagrange :

x	-4	0	4
f(x)	0	-24	16

$$f(x) = y_1 l_1(x) + y_2 l_2(x) + y_3 l_3(x)$$

$$= -24 \frac{(x+4)(x-4)}{(0+4)(0-4)} + 16 \frac{(x+4)x}{(4+4)4}$$

$$= (x+4)\left[\frac{3}{2}(x-4) + \frac{x}{2}\right] = 2(x+4)(x-3)$$

$$= \boxed{2(x+4)(x-3)}$$

$$= \boxed{2x^2 + 2x - 24}$$

Exercice 3 (5 pts)

Ecrire un algorithme (en langage algo ou C) qui donne les coefficients de la courbe issue de l'application d'une régression linéaire sur un nuage de points de taille quelconque.

Éléments de correction :

1 - Retrouver les formules de la régression linéaire à partir de la formule générale de l'approximation polynômiale, avec un degré 1.

2 - Appliquer directement la formule à partir de deux tableaux (les coordonnées des points). Fournir en sortie les deux coefficients a et b de la droite d'interpolation

FORMULAIRE

Formule du polynôme P_{n-1} de Lagrange pour l'interpolation de n points :

$$P_{n-1}(x) = \sum_{i=1}^{n} y_i \ell_i(x) \qquad \text{où} \qquad \ell_i(x) = \prod_{j=1, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$$

Système d'équations linéaires donnant l'expression générale des coefficients a_i du polynôme d'approximation de la méthode des moindres carrés :

$$\begin{bmatrix} n & \sum x_i & \sum x_i^2 & \dots & \sum x_i^m \\ \sum x_i & \sum x_i^2 & \sum x_i^3 & \dots & \sum x_i^{m+1} \\ \vdots & \vdots & \vdots & & \vdots \\ \sum x_i^m & \sum x_i^{m+1} & \sum x_i^{m+2} & \dots & \sum x_i^{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum y_i x_i \\ \vdots \\ \sum y_i x_i^m \end{bmatrix}$$

Formule de Jacobi donnant les valeurs des coefficients du vecteur solution à l'étape `k+1` en fonction des valeurs du même vecteur à l'étape précédente `k` :

$$x_i^{(k+1)} \leftarrow \frac{1}{A_{ii}} \left(b_i - \sum_{j=1, j\neq i}^{n} A_{ij} x_j^{(k)} \right), \; i = 1, 2, \dots, n$$

Formule de Gauss-Seidel donnant les valeurs des coefficients du vecteur solution à l'étape `k+1` en fonction des valeurs du même vecteur à l'étape précédente `k` :

$$x_i^{(k+1)} \leftarrow \frac{1}{A_{ii}} \left(b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} A_{ij} x_j^{(k)} \right), \; i = 1, 2, \dots, n$$

Coefficients a_i (i=`1...n`) du polynôme de Newton (égaux aux différences divisées respectives de degré i`-1` de y_i) :

$$a_1 = y_1, \; a_2 = \nabla y_2, \; a_3 = \nabla^2 y_3 \; \dots \; a_n = \nabla^{n-1} y_n$$

où `n` est le nombre de points. La formule générale de calcul des différences divisées de degré i est :

$$\nabla^j y_i = \frac{\nabla^{j-1} y_i - \nabla^{j-1} y_j}{x_i - x_j}, \; i = j+1, j+2, \dots, n$$