Licence Sciences et Technologies,
Calcul Numérique, TD4

Calcul Numérique, TD 4

Tout le travail demandé devra être réalisé sous la forme de "scripts" dont le nom sera `"TD4_xxx.m"` avec "xxx" le numero de la question pricipale (en chiffres romains), et d'un ensemble de fonctions que l'on placera dans un repertoire "TD4". On demande de bien commenter son code, et de ne pas oublier de nommer les auteurs du travail par un commentaire en début de chacun de ces fichiers. Il est par ailleurs conseillé d'utiliser des noms de fonctions significatifs, pour faciliter la lecture du travail.

I. Régression linéaire.

Pour illustrer la construction effectuée dans le cours pour réaliser l'approximation de données par une loi linéaire, nous allons construire des données bruitées et réaliser leur approximation.

(I.1) Définir un vecteur x de 100 nombres équirépartis entre 0 et 5. Créer un vecteur err de 100 nombres aléatoires compris entre 0 et 0.1, puis lui ajouter le vecteur $y = 2.1 * x + 0.7 + err$. Représenter graphiquement les données en utilisant des symboles 'x'.

(I.2) Réaliser une fonction "DistEuclid" dont les arguments seront a, b deux nombres réels et x, y deux vecteurs. La valeur retournée sera la distance entre y et $ax+b$ définie par :

$$d_{a,b}(x, y) = \sum_i (ax_i + b - y)^2.$$

(I.3) Représenter graphiquement la fonction $Dista : a \mapsto d_{a,2}(x, y)$, estimer la valeur a_0 réalisant le minimum de la distance.

(I.4) Rechercher par la même méthode b_0 qui réalise le minimum de $Distb : b \mapsto d_{a_0,b}(x, y)$.

(I.5) Représenter sur le même graphe les données (x, y) et leur approximation (x, a_0x+b_0). Que pensez-vous du résultat ? Vérifiez que vous retrouvez bien avec une précision raisonnable les valeurs de $a = \frac{<xy>-<x><y>}{<x^2>-<x>^2}$ et de $b =< y > -a < x >$ calculées dans le cours. Quelle erreur avez-vous commise lors de votre estimation ?

II. Recherche automatique de minimum.

Le but de cette partie est de rendre automatique la recherche des valeurs a_0 et b_0 de la question précédente. Pour rechercher le minimum d'une fonction f à une variable, on commence par l'estimer à l'aide de 3 valeurs : g_0, c_0 et d_0, telles que : $g_0 < c_0 < d_0$ avec $f(c_0) \leq f(g_0)$ et $f(c_0) \leq f(d_0)$. On a alors une estimation c_0 de optimum de f dans l'intervalle $[g_0, d_0]$ (avec une précision $|d_0 - g_0|$.

Pour améliorer cette estimation, nous allons construire une nouveau triplet de valeurs g_1, c_1 et d_1 avec les mêmes propriétés, mais tel que $|d_1 - g_1| < |d_0 - g_0|$. Pour ce faire, on choisit un point m dans l'intervalle $[g_0, d_0]$, puis on distingue 2 cas :

- Si $m \in [g_0, c_0]$, le nouveau triplet sera donné par la règle : Si $f(m) \leq f(c_0)$ alors $(g_1 = g_0, c_1 = m, d_1 = c_0)$ sinon $(g_1 = m, c_1 = c_0, d_1 = d_0)$.
- Si $m \in [c_0, d_0]$, le nouveau triplet sera donné par la règle : Si $f(m) \leq f(c_0)$ alors $(g_1 = c_0, c_1 = m, d_1 = d_0)$ sinon $(g_1 = g_0, c_1 = c_0, d_1 = m)$.

On continue ainsi à construire les triplets $(g_{k+1}, c_{k+1}, d_{k+1})$ en fonction de (g_k, c_k, d_k) jusqu'à ce que la précision soit suffisante.

(II.1) Verifier que les règles énoncées ci-dessus sont bien fonctionnelles, c'est à dire que le nouveau triplet construit à chaque étape a bien les mêmes propriétés que le triplet initial.

(II.2) Ecrire une fonction dont la première ligne sera :
`[g1,c1,d1] = PreciseMinimum( g0,c0,d0, fname)` ,
qui réalise le calcul du triplet suivant en utilisant la règle ci-dessus. Les evaluations de la fonction seront réalisées grâce à la commande Octave : `feval` dont vous devriez lire la page de manuel.

(II.3) Modifier cette dernière fonction en lui ajoutant 2 paramètres x et y pour pouvoir l'utiliser pour rechercher les minima des fonctions $Dista$ et $Distab$ du I.

(II.4) Ecrire un script qui donne les valeurs de a_0 et b_0 avec une precision de $1.0e-7$. Comparer à la valeur théorique de ces nombres. Les valeurs initiales des triplets seront déterminées grâce aux graphes de la partie I.

III. Sinusoïdes ou Carrés ?

IV. Ecrire une fonction SinN à deux arguments t (vecteur) et n (nombre), qui renvoie la valeur de $\sin(2*n*\pi*t)$.

V. Tracer sur un même graphe en utilisant "subplot", les 5 courbes représentatives de $SinN(1, tt)$, $SinN(2, tt)$, $SinN(5, tt)$, $SinN(10, tt)$et $SinN(20, tt)$ pour tt entre -1.5 et 1.5. L'idéal serait d'utiliser une boucle et un vecteur $nn = [1, 2, 5, 10, 20]$.

VI. Ecrire une fonction $SumSin$ permettant de calculer $\sum_{n=1}^{N} SinN(n, t)$ en fonction de N et t. Tracer sur un même graphe en utilisant "subplot", les 5 courbes représentatives de $SumSin(1, tt)$, $SumSin(2, tt)$, $SumSin(5, tt)$, $SumSin(10, tt)$ et $SumSin(20, tt)$, toujours pour tt entre -1.5 et 1.5. Que pensez-vous du résultat ?

VII. Après avoir défini correctement la fonction $IntSin$ qui calcule $\sum_{n=1}^{N} SinN(n, t)/(2\pi n)$. Tracer de même que précédemment les courbes représentatives de $IntSin(N, tt)$ pour tt entre 0.5 et 4.5 et N prenant les valeurs de $nn = [1, 2, 5, 10, 20]$. Quel est l'effet produit par la division par $2\pi n$?