Analyse Numérique

Semestre de Printemps 2022- Section MA

Prof. Annalisa Buffa

Séance 5 - 25 mars 2022

Exercice 1 (Matlab)

Considérons une fonction $f : [a,b] \to \mathbb{R}$ avec $f \in C^0([a,b])$, dont l'intégrale est $I(f) = \int_a^b f(x)\mathrm{d}x$. L'approximation de $I(f)$ à travers une formule d'interpolation simple s'écrit

$$Q_S(f) = \sum_{j=1}^{S} b_j f(c_j),$$

où b_j sont les poids de quadrature, et c_j les noeuds de quadrature, avec $j = 1, \dots, S$. Le type d'approximation polynomiale de la fonction $f(x)$ dans $[a,b]$ détermine la formule de quadrature spécifique. On observe que, par simplicité, ces formules sont en général définies sur l'intervalle de référence $[\bar{a}, \bar{b}] = [-1, 1]$ (ou parfois $[0,1]$), où les noeuds de quadrature $\bar{c}_j$ et les poids $\bar{b}_j$ sont donnés. Ensuite, les noeuds et poids de quadrature correspondants à un intervalle générique $[a,b]$ sont donnés par :

$$c_j = \frac{a+b}{2} + \frac{(b-a)}{2}\bar{c}_j, \qquad b_j = \frac{(b-a)}{2}\bar{b}_j, \qquad \text{pour } j = 1, \dots, S.$$

Les formules de quadrature de Gauss-Legendre (simples) constituent une famile de formules d'interpolation, chacune déterminée par S, le nombre de noeuds et poids de quadrature ; la formule de quadrature de Gauss-Legendre correspondant à $S \geq 1$ a ordre égal à $2S$. Les noeuds et poids de quadrature des formules de quadrature de Gauss-Legendre sur l'intervalle de référence $[\bar{a}, \bar{b}] = [-1, 1]$ pour $S = 1, 2, 3$ sont donnés dans le tableau suivant :

S	$\{\bar{c}_j\}$	$\{\bar{b}_j\}$
1	$\{0\}$	$\{2\}$
2	$\{-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\}$	$\{1, 1\}$
3	$\{-\frac{\sqrt{15}}{5}, 0, \frac{\sqrt{15}}{5}\}$	$\{\frac{5}{9}, \frac{8}{9}, \frac{5}{9}\}$

1. Écrire la fonction Matlab `gauss_legendre_simple_quadrature.m` qui implémente l'approximation de $I(f)$ grâce à la formule de quadrature de Gauss-Legendre simple pour $S = 1, 2, 3$. Utilisez le template inclu dans `gauss_legendre_simple_quadrature_template.m`. Les inputs de la fonction sont : `fun` (fonction définissant $f(x)$), les extrémités de l'intervalle a, b, et S, le nombre de noeuds et poids de quadrature.
2. Considérer la fonction $f(x) = \sin\left(\frac{7}{2}x\right) + e^x - 1$ avec $a = 0$ et $b = 1$ ($f \in C^\infty(]a,b])$), pour laquelle $I(f) = \frac{2}{7}\left(1 - \cos\left(\frac{7}{2}\right)\right) + e - 2$. Utiliser la fonction Matlab implémentée au point précédent pour approximer l'intégrale $I(f)$ grâce aux formules de quadrature de Gauss-Legendre pour $S = 1, 2, 3$. Reporter les valeurs des intégrales approximées en comparaison avec $I(f)$.

3. Poser $f(x) = x^d$, $a = 0$ et $b = 1$, avec $d \in \mathbb{N}$. Nous obtenons $I(f) = \frac{1}{d+1}$. En utilisant la fonction Matlab implémentée au premier point, vérifier les degrés d'exactitude des formules de Gauss-Legendre pour $S = 1, 2, 3$ en approximant l'intégrale $I(f)$ pour différentes valeurs de $d = 0, \ldots, 6$. Justifier les résultats obtenus, puis comparer et discuter les résultats avec les valeurs approchées des intégrales obtenues par les formules de quadrature du point milieu, du trapèze et de Simpson.

Exercice 2

Lorsque l'on veut intégrer une fonction numériquement, il est parfois important d'inclure les extrémités de l'intervalle parmi ses S noeuds de quadrature. En effet, si l'on utilise une telle formule de manière composite (c'est-à-dire en divisant l'intervalle d'intégration en un certain nombre de sous-intervalles, puis en appliquant la formule de quadrature sur chaque sous-intervalle), alors le dernier noeud d'intégration du i-ème sous-intervalle sera égal au premier noeud du $(i + 1)$-ème sous-intervalle. Cela permet d'économiser quelques calculs.
Une telle formule de quadrature (ayant les extrêmités de l'intervalle parmi ses S noeuds de quadrature) a ordre $2S - 2$, et elle est appelée formule de Gauss-Lobatto.

1. (à la main) Construire la formule de quadrature de Gauss-Lobatto

$$\int_{-1}^{1} f(x) \approx A_0 f(-1) + A_1 f(x_1) + A_2 f(1),$$

pour l'intervalle $[-1, 1]$. Il faut écrire et résoudre quatre équations pour trouver x_1, A_0, A_1 et A_2.

2. (Matlab) La formule de quadrature de Gauss-Lobatto à 5 noeuds sur $[-1, 1]$ est donnée par les noeuds et poids suivants :

Noeuds	Poids
-1	$1/10$
$-\frac{1}{7}\sqrt{21}$	$49/90$
0	$32/45$
$\frac{1}{7}\sqrt{21}$	$49/90$
1	$1/10$

(a) Écrire une fonction gauss_lobatto5 qui prend pour argument les extrêmités a et b d'un intervalle et une fonction f, et qui retourne l'approximation de l'intégrale de f sur $[a, b]$ donnée par la formule de Gauss-Lobatto à 5 noeuds. Utiliser le template donné dans `gauss_lobatto5.m`.

(b) Écrire une fonction gauss_lobatto5_composite qui prend pour argument les extrêmités a et b d'un intervalle, un nombre M de sous-intervalles et une fonction f ; et qui retourne l'approximation de l'intégrale de f sur $[a, b]$ donnée par la formule de Gauss-Lobatto composite à 5 noeuds sur M intervalles. Cette fonction doit utiliser la fonction gauss_lobatto5 de la question précédente. Utiliser le template donné dans `gauss_lobatto5_composite.m`.

(c) Calculer l'intégrale de $f(x) = \cos(x)^2 + \exp(x) - 1$ entre 0 et 2π en utilisant la formule de Gauss-Lobatto composite avec 5 noeuds sur M intervalles, avec M qui prend les valeurs 2^i, $i = 0, 1, 2, 3, 4, 5$. Quel est l'ordre de convergence attendu, par rapport à $h := \frac{2\pi}{M}$? En sachant que l'intégrale exacte est égale à $-1 + \exp(2\pi) - \pi$, vérifier que l'ordre de convergence soit celui attendu. Pour cela, faire un graphe de l'erreur de quadrature en fonction de h.

Exercice 3 (à la main)

Nous avons vu dans le cour que pour définir une formule de quadrature optimale à S noeuds, nous avons besoin d'un polynôme $M(t)$ de degré S tel que $\int_0^1 M(t)g(t)\,\mathrm{d}t = 0$, pour tout polynôme $g \in \mathbb{P}_{S-1}$, i.e. de degré au plus $S-1$. Cela est vrai si nous voulons une formule de quadrature d'ordre $2S$ pour calculer l'intégrale $\int_0^1 f(t)\,\mathrm{d}t$. Et dans ce cas, nous avons vu que nous pouvons prendre $M(t)$ comme étant le polynôme de Legendre de degré S, car les polynômes de Legendre sont orthogonaux par rapport au produit scalaire $< f, g >:= \int_0^1 f(t)g(t)\,\mathrm{d}t$.
Supposons maintenant que nous voulons trouver une formule de quadrature optimale pour calculer l'intégrale $\int_0^1 w(x)f(x)\,\mathrm{d}x$, où $w(x)$ est une fonction poids. C'est-à-dire que nous voulons trouver $\{c_i\}_{i=1}^s \subset [0,1]$ et $\{b_i\}_{i=1}^s \subset \mathbb{R}$ tels que $\int_0^1 w(x)p(x)\,\mathrm{d}x = \sum_{i=1}^S b_i p(c_i)$, pour tout $p \in \mathbb{P}_{2S-1}$. Avec le même raisonnement que précédemment (où on avait $w(x) \equiv 1$), nous avons besoin d'un polynôme $M(t)$ de degré S tel que $\int_0^1 w(t)M(t)g(t)\,\mathrm{d}t = 0$, pour tout polynôme $g \in \mathbb{P}_{S-1}$. Nous cherchons donc les polynômes ϕ_S qui sont orthogonaux par rapport au produit scalaire $< f, g >_w:= \int_0^1 w(t)f(t)g(t)\,\mathrm{d}t$.

1. En commençant par $\varphi_0(x) \equiv 1$ et en utilisant le procédé d'orthogonalisation de Gram-Schmidt, construire les polynômes orthogonaux de degré 0, 1 et 2 dans l'intervalle $[0,1]$ avec la fonction poids $w(x) = -\ln(x)$.
2. Déterminer les points et poids de quadrature pour la fonction poids $w(x) = -\ln(x)$ sur l'intervalle $[0,1]$, pour $S = 1$ et $S = 2$.

Exercice 4 (Matlab)

En Matlab, on peut calculer les polynômes d'interpolation en utilisant les commandes `polyfit` et `polyval`. Voyons plus en détail comment utiliser ces commandes.

`p = polyfit(x,y,n)` calcule les coefficients du polynôme de degré n qui interpole les valeurs $y = (y_0, \dots, y_n)$ aux noeuds $x = (x_0, \dots, x_n)$.

Exemple A. Pour interpoler les valeurs $\boldsymbol{y} = [3.38, 3.86, 3.85, 3.59, 3.49]$ aux noeuds $\boldsymbol{x} = [0, 0.25, 0.5, 0.75, 1]$ par un polynôme de degré 4, il faut utiliser les commandes Matlab/Octave suivantes :

```
>> x=[0:0.25:1]; %vecteur des points d'interpolation
>> y=[3.38 3.86 3.85 3.59 3.49]; %vecteur des valeurs
>> p1=polyfit(x,y,4)
p1 =
   1.8133 -0.1600 -4.5933 3.0500 3.3800
```

Donc `p1` est le vecteur des coefficients du polynôme interpolant : $\Pi_4(x) = 1.8133x^4 - 0.16x^3 - 4.5933x^2 + 3.05x + 3.38$.

Exemple B. Pour calculer le polynôme de degré n qui interpole une fonction f donnée dans $n+1$ noeuds, il faut d'abord construire le vecteur $\boldsymbol{y}$ en évaluant f dans les noeuds $\boldsymbol{x}$. Considérons le cas suivant :

```
>> f=@(x) cos(x);
>> x=[0:0.25:1];
>> y=f(x);
>> p=polyfit(x,y,4)
p =
```

```
   0.0362 0.0063 -0.5025 0.0003 1.0000
```

Remarque : si la dimension de $\boldsymbol{x}$ et $\boldsymbol{y}$ est $m+1 > n+1$ (n étant le degré du polynôme d'interpolation), la commande `polyfit(x,y,n)` retourne le polynôme interpolant de degré n au sens des moindes carrés. Dans le cas où $m+1 = n+1$, on trouve le polynôme d'interpolation "standard", puisque, dans ce cas, les deux coïncident.

`yval = polyval(p,xval)` calcule les valeurs y d'un polynôme de degré $\boldsymbol{n}$, dont les $\boldsymbol{n+1}$ coefficients sont mémorisés dans le vecteur p, aux points `xval`, c'est-à-dire :
`yval = p(1)*xval.^n +...+ p(n)*xval + p(n+1).`

Exemple C. Pour évaluer le polynôme trouvé dans l'exemple A au point $x = 0.4$ et, ensuite, pour en tracer son graphe, on peut utiliser les commandes suivantes :

```
>> xval=0.4;
>> yval=polyval(p1,xval)
yval =
   3.9012

>> xval=linspace(0,1,100);
>> yval=polyval(p1,xval);
>> plot(xval,yval)
```

On veut maintenant interpoler la fonction $f(x) = sin(x) + x$ sur l'intervalle $[0, 3\pi]$ en utilisant $n = 2, 3, 4, 5, 6$ noeuds équirépartis.

1. Utiliser la commande `polyfit` pour calculer les coefficients des polynômes interpolants $\Pi_n f$ pour les différentes valeurs de n.
2. Utiliser `polyval` pour évaluer les polynômes définis par les coefficients trouvés au point 1. aux points $xval = [0 : \pi/100 : 3\pi]$.
3. Afficher les graphes des différents polynômes ainsi que celui de la fonction $f(x)$ sur une même figure en utilisant les informations trouvées en 1. et 2.

Exercice 5 (Matlab)

On considère la fonction $f(x) = \sin(x)$ définie sur l'intervalle $I = [a, b] = [0, 3\pi]$.

1. En utilisant les commandes `polyfit` et `polyval`, calculer le polynôme d'interpolation $\Pi_n f$ de la fonction $f(x) = \sin(x)$ pour une distribution de noeuds uniforme dans $[0, 3\pi]$, dans les cas $n = 1, \ldots, 7$ (n étant le degré du polynôme d'interpolation). Comparer graphiquement le résultat avec la fonction donnée (utiliser au moins 1001 points pour les représentations).
2. Évaluer l'erreur
$$e_n(f) = \max_{x \in I} |f(x) - \Pi_n f(x)|$$
et visualiser le graphe de e_n en fonction de n.