Lernziele

In diesem Praktikum sollen Sie üben und lernen:

- Berechnung von Quadratur-Formeln
- Grafische Veranschaulichung von Orthogonalpolynomen und ihren Nullstellen

Praktikumsaufgabe 1 - Berechnung von Quadratur-Formeln

In der Vorlesung haben Sie die Quadratur-Formel

$$\int_a^b \omega(x)q(x)\,\mathrm{d}x = \omega_1^{(n)} q(x_1^{(n)}) + \ldots + \omega_n^{(n)} q(x_n^{(n)}), \quad q \in \mathbb{P}_{2n-1} \tag{1}$$

kennengelernt, wobei wir mit $x_1^{(n)}, \ldots, x_n^{(n)}$ die Nullstellen des Orthogonalpolynoms p_n zu der Gewichtsfunktion $\omega(x)$ und mit $\omega_k^{(n)}$ die Quadratur-Gewichte bezeichnen. Die monische Orthogonalpolynome p_n erfüllen dabei folgende Drei-Term-Rekursion

$$\begin{aligned} p_{k+1}(x) &= (x-\alpha_k)p_k(x) - \beta_k\, p_{k-1}(x), \quad k = 0,1,... \\ p_{-1}(x) &= 0, \quad p_0(x) = 1. \end{aligned} \tag{2}$$

Die Koeffizienten der Drei-Term-Rekursion sind gegeben durch

$$\begin{aligned} \alpha_k &= \frac{(x\,p_k, p_k)_\omega}{(p_k,p_k)_\omega}, \quad k = 0,1,... \\ \beta_0 &= (p_0,p_0)_\omega \beta_k \qquad = \frac{(p_k,p_k)_\omega}{(p_{k-1},p_{k-1})_\omega}, \quad k = 1,2,... \end{aligned} \tag{3}$$

Mit Hilfe der Koeffizienten der Drei-Term-Rekursion können die Quadraturpunkte und Gewichte wie folgt bestimmt werden:
Die Quadraturpunkte $x_k^{(n)}$ sind die Eigenwerte der Matrix

$$J_n = \begin{pmatrix} \alpha_0 & \sqrt{\beta_1} & & & \\ \sqrt{\beta_1} & \alpha_1 & \sqrt{\beta_2} & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \sqrt{\beta_{n-1}} \\ & & & \sqrt{\beta_{n-1}} & \alpha_{n-1} \end{pmatrix}$$

und die Gewichte sind gegeben durch $\omega_k^{(n)} = \mu_0 \left(\frac{v_1^{(k)}}{\|v\|}\right)^2$, wobei $v^{(k)}$ den k-ten Eigenvektor bezeichnet.

(i) Vervollständigen Sie die MATLAB-Funktion `getCoeff`, welche die Koeffizienten α_k und β_k der Drei-Term-Rekursion mittels Formel (3) berechnet. (Sie dürfen die Funktion `scalar` verwenden.)

(ii) Schreiben Sie eine MATLAB-Funktion, welche zu gegebenen Koeffizienten α_k, β_k und Momenten μ_k die Quadraturpunkte und Gewichte über das oben beschriebene Eigenwertproblem berechnet.
(Aufruf: `[xk,wk] = computeGaussNodesWeights(alpha,beta,m)`)

(iii) Schreiben Sie eine MATLAB-Funktion `val = evalOrthoPoly(alpha,beta,x)`, welche die Orthogonalpolynome p_k, $k = 0,...,n$, mittels der Drei-Term-Rekursion (2) an den Stellen `x` auswertet. Zurückgegen werden soll eine Matrix `val`, welche spaltenweise die Funktionswerte der Orthogonalpolynome $p_0,...,p_n$ enthält.

(iv) Vervollständigen Sie das Skript `main1.m`, welches zu gegebener Gewichtsfunktion die Momente `m` mit der Funktion `moment.m`, die Koeffizienten `alpha` und `beta`, sowie die Nullstellen der Orthogonalpolynome berechnet. Stellen Sie anschließend p_n sowie die Nullstellen von p_n graphisch dar. Wählen Sie hierzu $n \in \{5, 8, 15\}$. Was fällt Ihnen auf?

Praktikumsaufgabe 2 - Alternative Berechnung der Quadraturpunkte

Nachdem wir in Aufgabe 1 gesehen haben, dass die Berechnung der Koeffizienten der Drei-Term-Rekursion instabil ist, wollen wir in dieser Aufgabe alternative Möglichkeiten zur Berechnung der Koeffizienten für das logarithmische Gewicht $\omega(x) = -\log(x)$ auf dem Intervall $(0,1)$ untersuchen.
Eine Alternative bietet die Berechnung mittels Hankel-Determinanten Δ_k und Δ'_k, die definiert sind durch:

$$\Delta_{-1} = 1, \quad \Delta_0 = 1, \quad \Delta_n = \begin{vmatrix} \mu_0 & \mu_1 & \dots & \mu_{n-1} \\ \mu_1 & \mu_2 & \dots & \mu_n \\ \vdots & \vdots & & \vdots \\ \mu_{n-1} & \mu_n & \dots & \mu_{2n-2} \end{vmatrix}, \quad n = 1, 2, 3, ...$$

und

$$\Delta'_0 = 0, \quad \Delta'_1 = \mu_1, \quad \Delta'_n = \begin{vmatrix} \mu_0 & \mu_1 & \dots & \mu_{n-2} & \mu_n \\ \mu_1 & \mu_2 & \dots & \mu_{n-1} & \mu_{n+1} \\ \vdots & \vdots & & \vdots & \\ \mu_{n-1} & \mu_n & \dots & \mu_{2n-3} & \mu_{2n-1} \end{vmatrix}, \quad n = 1, 2, 3, ...$$

Die Koeffizienten α_k und β_k sind dann gegeben durch

$$\begin{aligned} \alpha_k &= \frac{\Delta'_{k+1}}{\Delta_{k+1}} - \frac{\Delta'_k}{\Delta_k}, \quad k = 0, 1, 2... \\ \beta_k &= \frac{\Delta_{k+1}\Delta_{k-1}}{\Delta_k^2}, \quad k = 0, 1, 2... \end{aligned} \tag{4}$$

(i) Schreiben Sie eine Funktion `[alpha,beta]=getCoeffHankel(m)` die zu den $2n$ Momenten $\mu_0, .., \mu_{2n-1}$ die Koeffizienten α_k, β_k, $k = 0, ..., n$ mit Formel (4) berechnet. Bis zu welcher Ordnung n können Sie jetzt die Koeffizienten der Drei-Term-Rekursion berechnen?

Als dritte Möglichkeit zur Berechnung der Koeffizienten betrachten wir modifizierte Momente

$$m_k := (P_k^{(a,b)}, 1)_\omega$$

bzgl. der Jakobi-Polynome $P_k^{(a,b)}$. Die Jakobi-Polynome sind die Orthogonalpolynome zur Gewichtsfunktion $\omega(x) = (1-x)^a(1+x)^b$. (Für $a = b = 0$ erhalten wir die Legendre Polynome). Die modifizierten Momente können mit der Funktion `modMoments.m` berechnet werden. Mit Hilfe des modifizierten Tschebyscheff-Algorithmus können aus den modifizierten Momenten $m_0, ..., m_{2n-1}$ und den Koeffizienten der Jakobi-Polynome $\alpha_k^{(Jac)}, \beta_k^{(Jac)}$, $k = 0, ..., 2n-2$, (werden von der Funktion `getCoeffJacobi` berechnet) die Koeffizienten der Orthogonalpolynome bestimmt werden.
(Aufruf: `[alpha,beta] = modCheb(m,alphaJ,betaJ)`.)

(ii) Vervollstäbndigen Sie das MATLAB-Skript `main2.m`, in welchem die Koeffizienten sowie die Nullstellen der Orthogonalpolynome zur Gewichtsfunktion $\omega(x) = -\log(x)$ auf dem Intervall $(0,1)$ mit allen drei Möglichkeiten berechnen werden.

(iii) Vergleichen Sie alle Methoden hinsichtlich des Fehlers in der Nullstellenberechnung (die 'exakten' Nullstellen liefert die Routine `[wk,xk] = gaussLog(n)`) sowie der Laufzeit für die Berechnung. Welche der Methoden ist die Beste?