Interdisziplinäres Zentrum
für Wissenschaftliches Rechnen (IWR)
Ruprecht–Karls–Universität Heidelberg

Dr. Peter Zaspel

Übungsblatt 5

24.05.2016 – 31.05.2016

Einführung in die Numerik SS 2016

Aufgabe 1. Gauß-Legendre-Polynome (4+4+4 Punkte)
Man zeige, dass die durch

$$\varphi_k(x) = \frac{k!}{(2k)!}\frac{d^k}{dx^k}(x^2-1)^k$$

definierten Polynome

a. orthogonal bezüglich des L^2-Skalarprodukts über $[-1,1]$ sind,

b. dass

$$||\varphi_k|| = \frac{k!^2}{(2k)!}\sqrt{\frac{2^{2k+1}}{2k+1}},$$

c. und dass

$$\varphi_k(1) = \frac{k!^2}{(2k)!}2^k.$$

Durch Normierung erhält man hieraus die sogenannten Gauß-Legendre-Polynome

$$L_k(x) := \frac{(2k)!}{k!^2 2^k}\varphi_k(x), \qquad L(1) = 1.$$

Hinweis: Man verwende partielle Integration (a.), die dritte Binomische Formel (b.) und die zweistufige Rekursionsformel für Legendre-Polynome (c.):

$$\begin{aligned}\varphi_0(x) &= 1\\ \varphi_1(x) &= x\\ \varphi_{k+1}(x) &= x\varphi_k(x) - \frac{k^2}{4k^2-1}\varphi_{k-1}(x),\ k = 1,2,....,n-1.\end{aligned}$$

Aufgabe 2. Optimale Approximation (2+2+2+2 Punkte)
Auf dem letzten Übungsblatt (Blatt 4, Aufgabe 1) hatten wir Funktionen betrachtet, die ein Orthonormalsystem des Vectorraumes T_n der trigonometrischen Polynome vom Grad $\leq n$ bilden. Es soll nun dieses

Wissen benutzt werden, um die Bestapproximation $g_n(x) \in T_n$ von vorgegebenen Funktionen $f(x) \in C[0, 2\pi]$ zu berechnen. Diese sollen den Fehler

$$||f - g_n||^2 = \int_0^{2\pi} (f(x) - g_n(x))^2 \,\mathrm{d}x$$

minimieren. Die zu approximierenden Funktionen lauten:

a. $f(x) = x$,

b. $f(x) = (x - \pi)^2$,

c. $f(x) = e^x$,

d. $f(x) = \begin{cases} 1, & 0 \leq x \leq \pi \\ 0, & \pi < x \leq 2\pi \end{cases}$.

Aufgabe 3. Praktische Aufgabe zur Gauß-Approximation mit Legendre-Polynomen (1+2+1+2+1+1+2 Punkte & 2+2+4 Bonuspunkte)
In dieser Übung wollen wir die Umsetzung der Gauß-Approximation über Legendre-Polynome studieren. Hierzu sei an Beispiel 2.4.6 aus der Vorlesung erinnert:

> Mittels Legendre-Polynomen lässt sich die Gauß-Approximation für Polynome schreiben als
> $$g(x) = \sum_{k=0}^{n} \left(\int_{-1}^{1} |L_k(x)|^2 dx \right)^{-1} \left(\int_{-1}^{1} f(\xi) L_k(\xi) d\xi \right) L_k(x)$$
> Die Berechnung der Koeffizienten erfolgt mittels numerischer Quadratur.

Hinweis: Bitte beachten Sie, dass der normierende Koeffizient $\left(\int_{-1}^{1} |L_k(x)|^2 dx \right)^{-1}$ in der Vorlesung versehentlich ausgelassen wurde. Dieser stammt daher, dass die Legendre-Polynome zwar an der Stelle 1 normiert sind, aber nicht ortho*normal* bezüglich des Skalarproduktes sind.

Wir wollen nun die Funktion $f(x) = x^2 e^x$ auf dem Intervall $[-1, 1]$ approximieren. Als Basis für Ihre Ausarbeitung steht Ihnen das Programmgerüst in der Datei *gauss.cpp* im Moodle-Kurs zur Verfügung.

a. Im Programmgerüst finden Sie bereits eine fertig implementierte Funktion `double evaluateLegendrePolynomial(double x, int k)`, die die Auswertung von $L_k(x)$ durchführt. Diese Funktion verwendet die Rekursionsformel und Normierung aus Satz 2.4.7 um die Auswertung durchzuführen. Bitte machen Sie sich mit dem Konzept des rekursiven Funktionsaufrufes (ggf. mittels Internet / Büchern) vertraut, so dass Sie bei der Abgabe diese Funktion sowie `evaluateUnnormalizedLegendrePolynomial` und `factorial` erklären können.

b. Zur Berechnung der Bestapproximation benötigen Sie numerische Quadratur. Wir wollen dies mit der sehr einfachen summierten / zusammengesetzten Trapezregel durchführen. Ergänzen Sie die Funktion `double compositeTrapezoidalRule(double* f, double h, int n_quadrature)` derart, dass für ein gegebenes Array `f` mit $n_{quadrature} + 1$ äquidistanten Auswertungen (mit Abstand h) einer Funktion f die summierte Trapezregel ausgewertet wird und das Resultat als Rückgabewert der Funktion ausgebeben wird.

c. Die Funktion `double evaluateSquaredLegendrePolynomialNorm(int k, int n_quadrature)` ist bereits implementiert und setzt die numerische Auswertung des Terms $\int_{-1}^{1} |L_k(x)|^2 dx$ mittels der zuvor implementierten summierten Trapezregel um. Studieren Sie diese um sie erklären zu können.

d. Schreiben Sie die Funktion `double approximateWeight(int k, int n_quadrature)`, die das Gewicht $\int_{-1}^{1} f(\xi) L_k(\xi) d\xi$ unter Verwendung der summierten Trapezregel mit $n_{quadrature}+1$ Stützstellen für das Legendre-Polynom L_k numerisch auswertet.

e. Ergänzen Sie die Funktion `void approximateCoefficients(double* alpha, int n, int n_quadrature)`, die die Koeffizienten

$$\alpha_k := \left(\int_{-1}^{1} |L_k(x)|^2 dx \right)^{-1} \left(\int_{-1}^{1} f(\xi) L_k(\xi) d\xi \right), \qquad k = 0, ..., n$$

unter Verwendung der vorangegangenen Methoden auswertet und im Array `alpha` ablegt.

f. Schreiben Sie die Funktion `double evaluateBestApproximation(double x, double* alpha, int n)` die für ein gegebenes Array mit den Koeffizienten $\alpha_0, ..., \alpha_n$ und die Legendre-Polynome $L_0, ..., L_n$ die Bestapproximation g (siehe oben) an der Stelle x auswertet.

g. Die Hauptroutine ist bereits fertig implementiert. Sie berechnet die Bestapproximation für die Funktion f und wertet diese so aus, dass die Datei `gauss_approx.dat` erzeugt wird, die in der in der ersten Spalte die Auswertungsstellen, in der zweiten Spalte die Auswertung der Bestapproximation und in der dritten Spalte die exakte Funktionsauswertung beinhaltet. Plotten Sie jeweils die Bestapproximation und die exakte Lösung für $n = 1, 2, 4, 6$.

Bonus h. Beschäftigen Sie sich noch einmal mit der Auswertung der Legendre-Polynome und plotten Sie diese dazu für größer werdende $k \geq 6$. Vergleichen Sie die berechneten Funktionen mit Plots der exakt berechneten Polynome, wie Sie sie mit einer kurzen Internet-Recherche finden. Was fällt auf? (Bei der Abgabe zeigen Sie die die Auffälligkeiten z. B. als Plots vor)

Bonus i. Berechnen Sie die Skalarprodukte (L_i, L_j) für $i, j = 1, ..., 10$. Reduzieren Sie nun die Anzahl an Stützstellen in der Quadratur. Was fällt hier auf?

Bonus j. Versuchen Sie die Probleme aus **Bonus h.** durch eine alternative Berechnungsvorschrift / Implementierung der Legendre-Polynome z.B. aus ”Numerical Recipes in Cäbzuschwächen / zu beheben.

Abgabe: 31.05.2016, 14:00-14:15 Uhr. (Mappen in der Vorlesung)