Institut für Analysis und Numerik 19.06.2020
Universität Magdeburg
Thomas Richter thomas.richter@ovgu.de
Henry von Wahl henry.vonwahl@ovgu.de

**Blatt 10 zur Vorlesung Numerische Mathematik**
**Sommersemester 2020**

**Abgabe bis 26. Juni, 23:59**

**Approximation** Die Gauss-Quadratur war direkte Anwendung der Interpolation. Wir haben eine möglichst hohe Ordnung durch optimale Wahl der Stützstellen erreicht. Wir gehen nun zurück zum allgemeinen Approximationsproblem: wie können wir eine Funktion $f$ mit Hilfe eines Polynoms $p \in P$ möglichst gut approximieren, so dass

$$\|f - p\| = \min_{q \in P} \|f - q\|.$$

Wir suche die sogenannte *Bestapproximation*. Falls wir die $L^2$-Norm wählen, so sprechen wir von der Gauss-Aproximation. Interessanter, aber auch viel schwerer wird es, wenn wir die Maximum-Norm wählen, dann sprechen wir von der Tschebyscheff-Approximation. In beiden Fällen unterscheidet sich das Vorgehen von der Interpolation. Wir werden das Polynom nicht mehr auf Basis von lokalen Interpolationseigenschaften bestimmen.

**Gauss-Approximation** Bei der Gauß-Approximation geht es um die beste Approximation einer Funktion $f$ durch ein Polynom $p \in P$ in der $L^2$-Norm. Es zeigt sich, dass dieses Problem sehr einfach auf Basis der orthogonalen Polynome gelöst werden kann. Es kommen hier die gleichen Legendre-Polynome vor, die auch bei der Gauss-Quadratur eine Rolle gespielt haben. Man lese Abschnitt *8.7 Bestapproximation* bis einschließlich Abschnitt *8.7.1 Gauß-Approximation - beste Approximation in der* $L^2$*-Norm* bis vor *Diskrete Gauß-Approximation* und gebe kurze Antworten

1. Was ist korrekt: jede Approximationsaufgabe kann auch als Interpolationsaufgabe formuliert werden, oder, jede Interpolationsaufgabe kann auch als Approximationsaufgabe formuliert werden?

2. Man nenne 2 äquivalente Bedingungen dafür, dass $p$ die Bestapproximation zu $f$ in der $L^2$-Norm unter allen Funktionen $q \in P$ ist.

3. Was ist die Gramsche Matrix bzgl. einer Basis? Warum ist diese stets regulär?

4. Warum ist es im Allgemeinen nicht sinnvoll, die Bestapproximation als Lösung eines linearen Gleichungssystems zu bestimmen?

5. Ist die folgende Aussage korrekt: Die quadratische Gauss-Approximation $p \in P_2$ zu einer Funktion $f$ hat bzgl. der Maximums-Norm immer einen geringeren Fehler $(\text{zu } f)$ als jede beliebige quadratische Interpolation $i_h f \in P_2$?

**Diskrete Gauss-Approximation, Verallgemeinerungen** Wir überspringen diese Abschnitte.

**Bestapproximation in der Maximumsnorm** Die Tschebyscheff-Approximation ist die Bestapproximation in der Maximumsnorm. Diese Aufgabe ist weitaus schwerer zu lösen und der Grund liegt in der Beschaffenheit der Maximumsnorm (hier ist ein grober Fehler im Buch, wo die Maximumsnorm mit dem min definiert ist)

$$\|f\|_\infty = \max_{[a,b]} |f(x)|,$$

welche nicht von einem Skalarprodukt induziert ist. Wir können daher nicht auf das Konzept der Orthogonalität zurückgreifen.
Stattdessen werden wir zunächst eine alternative Charakterisierung der Bestapproximation herleiten. Inder $L^2$-Norm war dies die Orthogonalitätsbeziehung $(f - p, q) = 0$ für alle $q \in P$. Bei der Maximumsnorm wird dies der *Satz von Kolmogoroff* (*8.90*) liefern. Das hier gewonnene Kriterium ist jedoch alles andere als einfach zu realisieren, so dass wir in *Satz 8.92* (*Tschebyscheffscher Alternantensatz*) ein noch konkreteres Kriterium angeben. Für den Fehler der Best-Approximation $e(x) = f(x) - p(x)$, mit $p \in P_n$, muss gelten, dass er in $n + 2$ Stellen $x_0, \dots, x_{n+1}$ sein Maximum $|e(x_i)|$ annimmt und zwar jeweils mit wechselndem Vorzeichen, also $e(x_i) = -e(x_{i+1})$.
Man lese Abschnitt *8.7.2 Tschebyscheff-Approximation* bis vor *Remez-Algorithmus*. Dabei können die Beweis zu Sätzen 8.90, 8.91 sowie 8.92 übersprungen werden. Nach Satz 8.92 kommt vor *Der Remez-Algorithmus* noch ein kurzer aber wichtiger Absatz, der beschreibt, wie bei Kenntnis einer Alternante die Bestapproximation konstruiert werden kann.
Man gebe kurze Antworten

1. Wieso ist es im Beweis zu *Satz 8.89* (*Existenz der Tschebyscheff-Approximation*) wichtig, zunächst die Menge $S_f$ einzuführen? D.h., warum wird nicht direkt das Minimum von $F(\phi)$ für $\phi \in S$ betrachtet?

2. Man argumentierre, dass die Funktion $p(x) = -\frac{1}{8} + x$ die lineare Bestapproximation zu $f(x) = x^2$ auf $[0, 1]$ ist. Hierzu kann entweder das Kriterium des Satzes von Kolmogoroff nachgewiesen werden, oder aber man versucht sofort zu zeigen, dass die Funktion $p$ die beste Approximation ist.

3. Wieviele und welche Punkte hat die Menge $E(f, p)$ für das eben diskutierte Beispiel auf $[0, 1]$?

4. Man zeige, dass diese Punkte eine Alternante im Sinne von Satz 8.92 bilden.

5. Es sei $(0, 1/2, 1)$ die Alternante zu $f(x) = x^2$. Wie sieht das lineare Gleichungssystem zur Bestimmung der Bestapproximation $p(x) = \alpha_0 + \alpha_1 x$ mit $\alpha_2 = \max \|f - p\|$ aus?

**Der Remez-Algorithmus** Der Remez-Algorithmus ist ein Verfahren zum Suchen einer Alternante. Da der Abschnitt im Buch nicht sonderlich leserlich ist, wird der Algorithmus hier noch einmal vorgestellt.
Wir suchen zu einer Funktion $f(x)$ auf $[a, b]$ die Bestapproximation $p \in P_n$ gemäß

$$p(x) = \sum_{i=0}^{n} \alpha_i x^i.$$

Ziel ist es, eine Alternante $x_0, x_1, \dots, x_{n+1}$ zu ermitteln, also Punkte, in denen gilt:

$$e(x) \coloneqq f(x) - p(x), \quad e(x_i) = -e(x_{i+1}) \text{ für } i = 0, \dots, n,$$

sowie $\max_{[a,b]} |e(x)| = |e_i|$ für $i = 0, \dots, n+1$.
Der Algorithmus geht folgendermaßen vor:

- Wir gehen davon aus, dass durch $(x_0, x_1, \dots, x_{n+1})$ eine Approximation an die Alternante gegeben ist. Wenn wir keine bessere Approximation haben, dann verteilen wir die Punkte gleichmäßig im Intervall.

- Dann bestimmen wir zu diesen Punkten das zugehörige Polynom $p(x)$ nach der Vorschrift

  $$p(x_k) + (-1)^k E = f(x_k) \quad \Leftrightarrow \quad \sum_{i=0}^{n} \alpha_i x_k^i + (-1)^k E = f(x_k), \quad k = 0, \dots, n+1$$

  Dies ist ein LGS in den $n+2$ Unbekannten $\alpha_n, \dots, \alpha_0, E$ und mit $n+2$ Gleichungen. Dabei ist $E$ die Approximation an den maximalen Fehler. Man achte darauf, dass die `np.poly1d`-Funktion die Monomkoeffizienten in absteigender Reihenfolge erwartet. Im Skript ist das LGS in aufsteigender Reihenfolge formuliert.

- Wir prüfen nun, ob $p$ wirklich die Bestapproximation ist, indem wir überprüfen, ob die Extremstellen von $e(x)$ die Alternanteneigenschaft besitzen. Hierzu
  - Bestimmen wir alle Extrempunkte von $e(x) = f(x) - p(x)$, d.h. die Nullstellen der Ableitung $e'(x)$ im Intervall $[a, b]$ sowie die beiden Eckpunkte.
  - Falls wir hierdurch nicht $n+2$ Punkte identifizieren, so bricht diese einfache Form des Algorithmus ab.
  - Ansonsten gehen wir davon aus, dass wir $n+2$ Punkte finden, diese nennen wir $a = y_0, y_1, \dots, y_n, y_{n+1} = b$

- Wir könnten nun überprüfen, ob durch diese $n+2$ Punkte eine hinreichend gute Approximation an eine Alternante gegeben ist. Dieses überspringen wir jedoch und machen einfach eine feste Anzahl an Iterationen

- Wir setzen den Algorithmus fort und wählen in der nächsten Iteration die Alternantenapproximation $y_0, \dots, y_{n+1}$.

## Übungsaufgaben

**Aufgabe 10.1**

**a)** Es sei $f \in L^2(-1,1)$ und $p \in P_n$ die Gauss-Approximation zu $f$. Man zeige die folgende Stabilitätsabschätzung

$$\|p\|_{L^2(-1,1)} \leqslant \|f\|_{L^2(-1,1)}$$

**b)** Es sei $f \in C^1[-1,1]$ und $S \subset C^1[-1,1]$ ein endlich-dimensionaler Teilraum. Man zeige, dass die sogenannte *Ritz-Projektion* $p \in S$, definiert durch

$$\|(f-p)'\|_{L^2(-1,1)} \leqslant \|(f-q)'\|_{L^2(-1,1)} \quad \forall q \in S$$

für jedes $f$ existiert und durch die Variationsgleichung

$$\int_{-1}^{1} \big(f'(x) - p'(x)\big) q'(x) \, dx = 0 \quad \forall q \in S$$

charakterisiert ist. Unter welcher Bedingung an den Raum $S$ ist diese Ritz-Projektion eindeutig?

*Hinweis: Die Ritz-Projektion ist die Grundlage der Finite Elemente Approximation für die numerische Approximation elliptischer partieller Differentialgleichungen.*

**Programmieraufgabe 10.2**

Man implementiere den Remez-Algorithmus nach der Video-Vorlage und bestimme die Bestapproximationen $p \in P_n$ für $n = 0, 1, 2, 4$ auf dem Intervall $[0,1]$ zu der Funktion

$$f(x) = x^{10}$$

In der Programmvorlage sind einige Bestandteile vorgegeben. Da $f(x)$ selbst ein Polynom ist, können wir sehr effizient mit `numpy.poly1d` sowie `numpy.polyder` arbeiten, um Polynome und deren Ableitungen darzustellen.
Man führe jeweils 3 Schritte des Remez-Algorithmus durch und gebe die Funktion $f$ sowie die gefundene Approximation $p_n$ und auch den Fehler $f - p_n$ graphisch aus.