Fachbereich Mathematik & Informatik
Freie Universität Berlin
Prof. Dr. Ralf Kornhuber, Tobias Kies

11. Übung zur Vorlesung

Computerorientierte Mathematik I

WiSe 2017

http://numerik.mi.fu-berlin.de/wiki/WS_2017/CoMaI.php

Abgabe: Donnerstag, 1. Februar 2018, 14:00 Uhr

1. Aufgabe (4 Bonus TP)
Sei $A \in \mathbb{R}^{n \times n}$ eine invertierbare Tridiagonalmatrix und sei $b \in \mathbb{R}^n$. Geben Sie einen effizienten Algorithmus zur Lösung des Gleichungssystems $Ax = b$ an. Stellen Sie den Algorithmus dabei als (hinreichend verständlichen) Pseudo-Code dar.

Hinweis: Sie müssen nicht beweisen, dass dieser Algorithmus das Gleichungssystem löst.

2. Aufgabe (6 Bonus TP)
Seien $m \in \mathbb{N}$ Matrizen $A^{(i)} \in \mathbb{R}^{n \times n}$ mit LR-Zerlegungen $A^{(i)} = L^{(i)} R^{(i)}$ gegeben, sowie $b \in \mathbb{R}^n$. Wir definieren $A := \prod_{i=1}^{m} A^{(i)} = A^{(1)} \cdots A^{(m)}$.

a) Bestimmen Sie den Rechenaufwand, der nötig ist, um die Matrix A aus den Matrizen $A^{(i)}$ durch Ausmultiplizieren zu berechnen.

b) Bestimmen Sie den Rechenaufwand, der nötig ist, um das Gleichungssystem $Ax = b$ durch sukzessive Anwendung der Vor- bzw. Rücksubstitution auf Basis der Zerlegungen $A^{(i)} = L^{(i)} R^{(i)}$ zu berechnen.

c) Nehmen wir nun an, dass wir $k \in \mathbb{N}$ rechte Seiten $b^{(i)}$ gegeben haben, für die wir jeweils $Ax^{(i)} = b^{(i)}$ lösen möchten. Wann ist es hinsichtlich des Rechenaufwands sinnvoller, dieses Problem nach dem Ansatz aus b) zu lösen, und wann sollte man einen Ansatz entsprechend a) verfolgen, in dem die Matrix A explizit ausgerechnet wird?

Anmerkung: Die Vorwärtssubstitution zu einer LR-Zerlegung $A = LR$ und dem Gleichungssystem $Ax = b$ bezeichnet den Algorithmus für das Berechnen von $\tilde{b} = L^{-1} b$. Die Rückwärtssubstitution ist der Algorithmus für das Berechnen von $R^{-1} \tilde{b}$.

3. Aufgabe (8 Bonus PP)

a) Implementieren Sie eine Matlab-Funktion `B = lr(A)`, die die LR-Zerlegung einer quadratischen Matrix $A \in \mathbb{R}^{n \times n}$ mittels Gauß-Elimination ohne Zeilenpivotisierung berechnet und das Ergebnis kompakt in einer Matrix $B \in \mathbb{R}^{n \times n}$ zurückgibt. Gehen Sie dabei im Algorithmus möglichst speichersparend vor und verwenden Sie nicht die Matlab-eigene LR-Zerlegung. Sie dürfen in Ihrem Programm die Annahme machen, dass der Algorithmus für die Eingabe A durchführbar ist.

b) Schreiben Sie eine Funktion `x = backsolve(B,b)`, die zur Matrix B aus a) und zu einem Vektor $b \in \mathbb{R}^n$ die Rücksubstitution durchführt und die Lösung im Vektor x zurückgibt.

c) Testen Sie Ihr Programm, indem Sie ein Skript `run.m` schreiben, das die LR-Zerlegung für die Matrix `A = magic(4)./((1:4) + (1:4)')` mittels `lr` berechnet. Lösen Sie anschließend das Problem $Rx = b$ für $b = (1, 2, 3, 4)^T$ mittels `backsolve`.

   Geben Sie schließlich die Matrizen L und R aus der LR-Zerlegung für A sowie den Lösungsvektor x aus. Verwenden Sie dazu die Funktion `fprintf` und speichern Sie die Programmausgabe in der Textdatei `output.log`.

Zur Abgabe der Programme: Packen Sie die Dateien `lr.m`, `backsolve.m`, `run.m`, `output.log` in ein ZIP-Archiv. Benennen Sie das ZIP-Archiv mit Ihrem ZEDAT-Accountnamen und schicken Sie dieses per Email an Ihren Tutor. Achten Sie auch auf Groß- und Kleinschreibung bei den Dateinamen.