MATHEMATISCHES INSTITUT
PROF. DR. CHRISTIANE HELZEL
PAWEL BUCHMÜLLER

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

17. DEZEMBER 2015

Computergestützte Mathematik zur Linearen Algebra – 8. Übungsblatt

Aufgabe 29: (LU)

Befehle: `lu, error, \`

Schreiben Sie eine Funktion `MySolve(A,k,b)` zur Lösung der Gleichung

$$A^k x = b \text{ mit } A \in \mathbb{R}^{n\times n}, b \in \mathbb{R}^n, k \in \mathbb{N}_{>0}.$$

- Verwenden Sie die LU-Zerlegung.
- b darf auch mehrere rechte Seiten beinhalten.
- A^k soll **nicht** berechnet werden.
- Bevor die Berechnung beginnt, soll sichergestellt werden, dass A und b die richtige Größe haben und dass $k \in \mathbb{N}_{>0}$ ist.
- Testen Sie Ihre Funktion mit "A29Test.m". Zu finden unter: `http://www.am.uni-duesseldorf.de/~helzel/Lehre/WiSe1516/CompLinA/A29Test.m`

Hinweis: Interpretieren Sie das Problem als eine Folge von k aufeinander folgender LGS.

Aufgabe 30: (Eigenwert / Eigenvektor)

Befehle: `norm, eye, while, \`

Zur Bestimmung von Eigenwerten und Eigenvektoren gibt es viele iterative Algorithmen. Einer davon heißt Rayleigh-Quotient Iteration. Dabei „schätzt“ man zunächst einen Eigenvektor v_0. Durch die Iteration

$$\lambda_i = \frac{v_i^T A v_i}{v_i^T v_i}, \quad i = 0, 1, ...$$

$$v_{i+1} = \frac{(A - \lambda_i I)^{-1} v_i}{||(A - \lambda_i I)^{-1} v_i||}, \quad i = 0, 1, ...$$

Erhalten wir nun immer bessere Näherungen. Unterschreitet der Fehler eine zuvor definierte Toleranz, dann wird der Algorithmus beendet. Schreiben Sie eine Funktion `reyleigh(A,v0,tol)` zur Bestimmung eines Eigenwerts und eines Eigenvektors.

- $(A - \lambda_i I)^{-1}$ soll dabei nie explizit berechnet werden.
- Der \ Operator darf högstens ein Mal je Iteration zum Einsatz kommen.

Aufgabe 31: (Sudoku) Befehle: `setdiff, length, ceil`

Sudokuregeln: Bei einem Sudokuspiel der Größe N müssen in eine $N^2 \times N^2$ Matrix S natürliche Zahlen z zwischen 1 und N^2 eingetragen werden. Es ist zulässig die Zahl z in die Position $S_{i,j}$ einzutragen, falls die vier folgenden Bedingungen erfüllt sind:

(i) Die Position $S_{i,j}$ ist unbesetzt.

(ii) Die Zahl z steht nicht bereits in der j-ten Spalte von S.

(iii) Die Zahl z steht nicht bereits in der i-ten Zeile von S.

(iv) Die Zahl z steht nicht bereits in dem Unterblock von S, in dem auch $S_{i,j}$ ist. Siehe Abbildung für $N = 2$ und $N = 3$.

(a) Schreiben Sie eine MATLAB-Funktion `[BI, BJ] = SudBlock(S,i,j)` die, die Indizes des Unterblocks bestimmt in dem $S_{i,j}$ liegt.
Bsp.: Für $N = 3$ soll `SudBlock(S,4,7)` die Vektoren `BI = [4 5 6]` und `BJ = [7 8 9]` liefern.

(b) Schreiben Sie eine MATLAB-Funktion `vec = SudZug(S,i,j)`, die einen Vektor liefert, der alle nach den Bedingungen (ii) – (iv) zulässigen Werte für $S_{i,j}$ enthält.

		3	2
2	1	4	
4			

			1			2		
		2			3			
			9		5			
		8					7	
			7					5
	7	4				6		
					4			
6	2						4	8
3								

Aufgabe 32: (Sudoku - Teil 2)

Befehle: `find, length`

Schreiben sie eine MATLAB-Funktion `ST = sudoku(S)` die auf eine Sudoku Matrix angewendet wird und folgendes liefert

- `ST` ist eine Struktur der Länge K, wobei K die Anzahl der „0“ in S ist.
- `ST` hat drei Komponenten `ST.i`, `ST.j` und `ST.zug`.
- `ST.i` und `ST.j` sind die Koordinaten einer „0“ und `ST.zug` ein Vektor mit allen zulässigen Einträgen an dieser Stelle.

Hinweis: Das reicht noch nicht um das Rätsel zu lösen.

Besprechung in den Übungen am 4. Januar 2016 - 7. Januar 2016