MATHEMATISCHES INSTITUT
PROF. DR. CHRISTIANE HELZEL
ANDREAS TROLL

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

7.12.2017

Computergestützte Mathematik zur linearen Algebra – 9. Übungsblatt

WICHTIG: Kommentieren Sie Ihren Quelltext. Ihre Skripte müssen durch ausführen des „run"-Befehls (grünes Dreieck bzw. F5) lauffähig sein.

Aufgabe 30: (*Code verstehen und umschreiben*)

Skriptname: *effizienter.py*
Sei $A \in \mathbb{R}^{N\times N}$. Die Funktion `PLR(A)` (siehe Vorlesung 9) berechnet die Matrizen P, L und R für die LR-Zerlegung $PA = LR$. Hierbei wird die Permutationsmatrix P als volle Matrix gespeichert, was Speicherverschwendung ist.

(a) Schreiben Sie eine Funktion `PLR_vek(A)`, welche die notwendigen Permutationen von Anfang an nur in einem Vektor p speichert und diesen statt P zurückgibt. Beispiel:

$$P = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \Leftrightarrow p = \texttt{np.array}([3, 1, 0, 2])$$

Tipp: `A[p]` ist ist äquivalent zu `P@A`.

(b) Testen Sie Ihre Funktion an geeigneten Matrizen.

Aufgabe 31: (*Inverse berechnen*)

Skriptname: *Inverse_durch_LR.py*

(a) Schreiben Sie eine Funktion `inverseLR(A)`, welche eine Matrix A übergeben bekommt und mit Hilfe der *LR-Zerlegung* die Inverse A^{-1} berechnet. Sollte A singulär oder nicht quadratisch sein, so soll eine Warnung ausgegeben werden und der Rückgabewert soll `None` sein.
Hinweis: Verwenden Sie keine eingebauten Funktionen, wie `numpy.linalg.inv` oder `numpy.linalg.solve`.
`SolveL`, `SolveR` und `PLR` (siehe Vorlesung 9) dürfen Sie verwenden.

(b) Testen Sie Ihre Funktion an geeigneten Matrizen und vergleichen Sie Ihre Ergebnisse mit `numpy.linalg.inv`.

Aufgabe 32: (*Determinante berechnen*)

Skriptname: *LR_mit_Det.py*

(a) Schreiben Sie auf Basis von `PLR` (Vorlesung 9) eine Funktion `PLR_det`, welche zusätzlich die Determinante von A berechnet und als letztes Argument zurückgegeben wird, falls Ihre Funktion `determinante=True` als zusätzliches Eingabeargument übergeben bekommt. `PLR_det(A)` soll also ausschließlich P, L und R berechnen und zurückgeben, `PLR_det(A,True)` soll zusätzlich noch die Determinante von A berechnen und mit zurückgeben.
Hinweis: Verwenden Sie keine zusätzlichen Funktionen, wie `numpy.linalg.det`. Vergessen Sie außerdem bei der Berechnung der Determinante die Permutationsmatrix P nicht!

(b) Testen Sie Ihre Funktion an geeigneten Beispielen und vergleichen Sie Ihre Ergebnisse mit `numpy.linalg.det`.

Aufgabe 33: (*Poisson-Gleichung*)

Skriptname: *poisson.py*
Gegeben sei die Poisson-Gleichung

$$-\Delta u(x,y) = f(x,y) \text{ für } (x,y) \in \Omega \text{ und } u(x,y) = g(x,y) \text{ für } (x,y) \text{ auf } \partial\Omega. \tag{1}$$

Hierbei ist $\Delta u(x,y) = \frac{\partial^2 u(x,y)}{\partial x^2} + \frac{\partial^2 u(x,y)}{\partial y^2}$, $\Omega \subset \mathbb{R}^2$ offen und beschränkt und $\partial\Omega$ der Rand von Ω.
Sei in dieser Aufgabe $\Omega = [0,1] \times [0,1]$, $x_k = \frac{k}{N-1}$ mit $k = 0, \ldots N-1$ und y_k analog zu x_k.
$\Delta u(x,y)$ kann man approximieren durch

$$\Delta u_{k,l} \approx (N-1)^2 (u_{k+1,l} + u_{k-1,l} + u_{k,l+1} + u_{k,l-1} - 4u_{k,l}) \text{ mit } u_{k,l} := u(x_k, y_l). \tag{2}$$

(a) Schreiben Sie eine Funktion `u=loese_poisson(f,N)`, welche eine approximierte Lösung u (ausgewertet an den Stellen x_k, y_l, mit $k, l = 0, \ldots, N-1$) bei vorgegebenen f und $g \equiv 0$ (siehe (1)) zurückgibt. `f` ist ein *functionhandle* mit der rechten Seite und für $\mathtt{N} \in \mathbb{N}$ siehe oben. Die Rückgabe `u` soll eine $(N \times N)$-Matrix mit den Einträgen $\mathtt{u}[\mathtt{k},\mathtt{l}] = u_{k,l}$ sein.
Tipp für die Vorgehensweise: Überlegen Sie sich, wie das zu lösende lineare Gleichungssystem $Au = f$ aussieht. A haben Sie (fast) schon in Aufgabe 28 (ab Freitag auch auf der Vorlesungsseite) erstellt. Vergleichen Sie dazu die Struktur der Matrix mit (2). Beachten Sie, dass $u(0, y_l) = u(1, y_l) = u(x_k, 0) = u(x_k, 1) = 0$ gilt, sie u also nur im Inneren des Gebietes bestimmen müssen.

(b) Testen Sie `loese_poisson(f,N)` an folgender Funktion

$$f(x,y) = 13\pi^2 \sin(2\pi x) \sin(3\pi y)$$

für $N = 5, 20, 50$ und vergleichen Sie Ihre Lösungen mit der exakten Lösung

$$u(x,y) = \sin(2\pi x) \sin(3\pi y).$$

Sie können beispielsweise die Normen der Unterschiede berechnen (welche Normen sind (un)geeignet?) oder Zeilen/Spalten der Lösungen plotten.

Besprechung in den Übungen vom 11.-15.12. 2017.