Prof. Dr. Karsten Urban
Dipl.-Math. oec. Klaus Stolle
Institut für Numerische Mathematik
Universität Ulm

SoSe 2017
Blatt 07
02.06.2017

Angewandte Numerik 1

Besprechung in den Tutorien in der Woche vom 12.06.2017 bis 16.06.2017

Für dieses Übungsblatt gibt es 14 Theorie- und 15 Matlab-Punkte, sowie 4 Theorie- und 8 Matlab-Zusatzpunkte. Punkte, die mit einem * gekennzeichnet sind, sind Zusatzpunkte.
Die 60-Prozent-Grenzen liegen aktuell (inklusive Blatt 07) bei 88,8 Theoriepunkten und 64,2 Matlabpunkten.

Aufgabe 29 *(Programmieraufgabe: Householder-Spiegelung mit kompakter Speicherung)* (5M+5M*+3M* Punkte)

a) In Aufgabe 28 vom letzten Übungsblatt 6 haben Sie eine Funktion `[Q, R] = qrHouseholder(A)` geschrieben, die die QR-Zerlegung einer Matrix $A \in \mathbb{R}^{m\times n}$ mittels Householder-Spiegelungen berechnet. Ändern Sie diese Funktion zu einer Funktion `M = qrHhKompakt(A)` ab. `M = qrHhKompakt(A)` soll wie `[Q, R] = qrHouseholder(A)` die QR-Zerlegung der Matrix A mittels Householder-Spiegelungen berechnen, dabei aber den rechten oberen Dreiecksteil der Matrix A sukzessive mit der rechten oberen Dreiecksmatrix R überschreiben sowie im linken unteren Dreiecksteil der Matrix A die wesentlichen Komponenten der Householder-Vektoren $\omega^{(j)}$ speichern. Der Rückgabewert M soll diese überschriebene Matrix sein.

b) Betrachten Sie nun wieder das lineare Ausgleichsproblem $\|Ax - b\|_2 \to \min$ aus Aufgabe 20 von Übungsblatt 5 (Wasserstand der Nordsee). Da Sie dieses lineare Ausgleichsproblem mit Ihrer Funktion `M = qrHhKompakt(A)` lösen wollen, ohne die Matrizen Q und R explizit zu berechnen, können Sie Ihre Funktion `[x, uR] = solveQR(Q, R, b)` aus Aufgabe 22 nicht anwenden. Schreiben Sie also eine Funktion `[x, uR] = solveHhKompakt(M, b)`, die die Lösung `x` eines linearen Ausgleichsproblems berechnet, ohne die Matrizen Q und R explizit aufzustellen. Dabei sind `M` die von Ihrer Funktion `M = qrHhKompakt(A)` erzeugte Matrix, `b` der Vektor der Messwerte und `uR` das unvermeidbare Residuum.

c) Lösen Sie das lineare Ausgleichsproblem aus Aufgabe 20 mit Ihren Funktionen `M = qrHhKompakt(A)` und `[x, uR] = solveHhKompakt(M, b)`. Vergleichen Sie Ihre Ergebnisse mit den Ergebnissen vom Übungsblatt 5.

Aufgabe 30 *(Lineares Ausgleichsproblem und die Singulärwertzerlegung)* (8T+6T Punkte)

Betrachten Sie das lineare Ausgleichsproblem

$$\min_{x\in\mathbb{R}^3} \left\| \frac{1}{27}\begin{pmatrix} 16 & 52 & 80 \\ 44 & 80 & -32 \\ -9 & -36 & -72 \\ -16 & -16 & 64 \end{pmatrix} x - \begin{pmatrix} 4 \\ 1 \\ 3 \\ 0 \end{pmatrix} \right\|_2 .$$

Die Singulärwertzerlegung $A = U\Sigma V^T$ der Matrix A ist gegeben durch

$$U = \frac{1}{45}\begin{pmatrix} 32 & 5 & 24 & -20 \\ 4 & 40 & 3 & 20 \\ -27 & 0 & 36 & 0 \\ 16 & -20 & 12 & 35 \end{pmatrix}, \qquad \Sigma = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \qquad V^T = \frac{1}{9}\begin{pmatrix} 1 & 4 & 8 \\ 4 & 7 & -4 \\ -8 & 4 & -1 \end{pmatrix}.$$

Bestimmen Sie

a) die Lösung x^* des linearen Ausgleichsproblems mit der kleinsten euklidischen Norm und

b) alle Lösungen des linearen Ausgleichsproblems.

Hinweise:
„Viele (alle) Wege führen nach Rom!“ – Zu einigen dieser Wege wollen wir Ihnen hier ein paar Hinweise geben:

i) Sie können Bemerkung 4.5.8 und Lemma 4.5.9 verwenden.

ii) Beachten Sie, dass sich die Matrizen Q, R und S in Bemerkung 4.5.8 aus der vorhandenen Singulärwertzerlegung $A = U\Sigma V^T \Leftrightarrow U^T A = \Sigma V^T$ einfach berechnen lassen.

iii) Alternativ können Sie die Lösung x^* des linearen Ausgleichsproblems mit der kleinsten euklidischen Norm nach Satz 4.5.7 berechnen.

iv) Alle Lösungen des linearen Ausgleichsproblems können Sie alternativ über den Beweis zu Satz 4.5.7 erhalten.

Aufgabe 31 *(Programmieraufgabe: Bildkompression mit der Singulärwertzerlegung)*

(5M+2T*+2T*+2M+3M Punkte)

Gegeben sei $A \in \mathbb{R}^{m\times n}$ mit Singulärwertzerlegung $A = U\Sigma V^T$ und den Singulärwerten $\sigma_1 \geq \cdots \geq \sigma_r > 0$, mit $r = \text{rang}(A)$. Ferner seien $u_i \in \mathbb{R}^m, i = 1, \ldots, m$ die Spaltenvektoren von U und $v_j \in \mathbb{R}^n, j = 1, \ldots, n$ die Spaltenvektoren von V. Für $0 < k < r$ definieren wir die Matrix

$$A_k = \sum_{\ell=1}^{k} \sigma_\ell u_\ell v_\ell^T$$

als Approximation von A.
Die Approximationseigenschaften von A_k lassen sich beispielsweise zur Datenkompression nutzen. Ein zweidimensionales Schwarzweißbild kann man im Rechner als Matrix speichern, bei der jeder Eintrag der Grauwert des Pixels in der entsprechenden Position auf dem Bild ist.

a) Laden Sie in einem Matlab-Skript mit

```
load gatlin
image (X)
colormap(map)
```

ein (in Matlab bereits gegebenes) Schwarzweißbild, welches sechs Numeriker (von links: J. H. Wilkinson, W. Givens, G. Forsythe. A. Housholder, G. Henrici, F. L. Bauer) auf der so genannten Gatlinburg-Tagung zeigt. Die Datei `gatlin` enthält die Matrix `X`, die Farbskala `map` sowie die Bildunterschrift `caption`. Berechnen Sie die Singulärwertzerlegung der Matrix `X` (Matlab-Befehl `svd`, dies kann ein bisschen dauern). Rekonstruieren Sie das Bild mit $A_{40} = \sum_{i=1}^{40} \sigma_i u_i v_i^T$ und zeigen Sie die Rekonstruktion wieder an.

b) Wieviel Speicherplatz benötigt das Originalbild (dies können Sie in Ihrem Workspace sehen)? Welche Daten müssen Sie speichern, um A_k rekonstruieren zu können? Und wieviel Speicherplatz benötigen diese Daten?

c) Was beobachten Sie, wenn Sie k vergrössern oder verkleinern? Welches k ist für das menschliche Auge ausreichend? Wieviel Speicherplatz können Sie durch diese Bildkompression sparen?

d) Veranschaulichen Sie sich, wie die Singulärwerte kleiner werden. Stellen Sie dazu das Abfallen der Singulärwerte grafisch dar. Erklären Sie Ihre Beobachtungen aus Teilaufgabe c) anhand dieses Schaubildes.

e) Wählen Sie $k \leq \min\{m, n\}$. Berechnen Sie zusätzlich für jedes i mit $1 \leq i \leq k$ den Fehler $\|X - A_i\|_2$. Plotten Sie diese Fehler und die ersten k Singulärwerte in ein Schaubild mit logarithmierter y-Achse. Was fällt Ihnen auf? Stellen Sie einen Zusammenhang der beiden Kurven her.

   Hinweis: Für eine Matrix C lässt sich die Spektralnorm $\|C\|_2$ berechnen durch den Matlab-Befehl `norm(C, 2)`.

Hinweise:
Die Programmieraufgaben sind in Matlab zu erstellen. Der Source Code muss strukturiert und dokumentiert sein. Senden Sie **spätestens 24 Stunden vor Ihrem Tutorium** alle Matlab-Files und alle Ergebnisse in einer E-mail mit dem Betreff **Loesung-Blatt07** an **angewandte.numerik@uni-ulm.de**.