Fachbereich Mathematik & Informatik
Freie Universität Berlin
Prof. Dr. Carsten Gräser, Tobias Kies

5. Übung zur Vorlesung

Numerik I

SoSe 2015

http://numerik.mi.fu-berlin.de/wiki/SS_2015/NumerikI.php

Abgabe: Fr., 29.05.2015, 12:00 Uhr Allgemeine Hinweise

Bitte beachten Sie die auf der Vorlesungshomepage angegebenen Hinweise zur Bearbeitung und Abgabe der Übungszettel, insbesondere der Programmieraufgaben.

1. Aufgabe (4 TP)
Bestimmten Sie für

$$A = \begin{pmatrix} 3 & -3 & 1 \\ 0 & 1 & 5 \\ 0 & 0 & 2 \\ 0 & 1 & 0 \\ 4 & -4 & -7 \end{pmatrix}$$

eine QR-Zerlegung von A mittels Givens-Rotationen. Geben Sie dabei auch die Rotationsmatrizen an, aus denen die Orthogonalmatrix $Q \in \mathbb{R}^{5\times 5}$ gebildet wird.

2. Aufgabe (4 TP)
Betrachten Sie die in der Vorlesung besprochenen Algorithmen zur Bestimmung einer QR-Zerlegung mittels *Givens-Rotation* und *Householder-Reflexion*. Verifizieren Sie folgende Abschätzungen bezüglich des Aufwandes unter der Voraussetzung, dass die Anzahl der zu approximierenden Punkte ungefähr mit der Anzahl der Fitting-Parameter übereinstimmt; also $m \approx n$.

a) Für die Berechnung mit Givens-Rotationen werden

$$\mathcal{O}(\tfrac{4}{3}n^3) \text{ Punktoperationen und } \mathcal{O}(\tfrac{1}{2}n^2) \text{ Quadratwurzeln}$$

benötigt.

b) Für die Berechnung mit Householder-Reflexionen werden

$$\mathcal{O}(\tfrac{2}{3}n^3) \text{ Punktoperationen}$$

benötigt.

3. Aufgabe (8 PP)

a) Implementieren Sie in `MATLAB` eine Funktion `[Q, R] = qr_givens(A)`, die die QR-Zerlegung einer $(m \times n)$-Matrix `A` mittels *Givens-Rotationen* berechnet. Dabei ist `Q` eine orthogonale $(m \times m)$-Matrix und `R` eine rechte obere $(m \times n)$-Dreiecksmatrix.

b) Implementieren Sie nun auch eine Funktion `[Q,R] = qr_householder(A)`, die wie oben die QR-Zerlegung einer Matrix `A` berechnet, aber nun auf Basis von *Householder-Reflexionen.*

c) Testen Sie Ihre beiden Implementationen anhand der Matrix

$$A = \begin{pmatrix} 3 & -3 & 1 \\ 0 & 1 & 5 \\ 0 & 0 & 2 \\ 0 & 1 & 0 \\ 4 & -4 & -7 \end{pmatrix}$$

und vergleichen Sie Ihr Ergebnis mit dem der `MATLAB`-eigenen Funktion `qr`.

d) Schreiben Sie eine `MATLAB`-Funktion `coeff = Ausgleich(A,b)`, die ein lineares Ausgleichsproblem für eine $(m \times n)$-Matrix `A` und einen Satz von m Zuständen `b` mit Householder-Reflexionen löst und das Ergebnis im n-Vektor `coeff` zurückgibt.

Die Datei `qr-prog.txt` auf der Homepage beschreibt den monatlichen Verlauf der Arbeitslosenquoten in Deutschland über die Jahre 2012 − 2014 mit Hilfe von Datenpaaren (t_i, b_i).[1]
Wir möchten das Verhalten dieser Entwicklung mit einer Modellfunktion beschreiben, die sowohl langfristige Änderungen als auch saisonale Schwankungen berücksichtigen kann. Als eine Möglichkeit hierfür definieren wir den $(2M+N+1)$-dimensionalen Ansatzraum $X_{M,N}$ aller in Frage kommenden Modellfunktionen durch

$$X_{M,N} := \operatorname{span}\{s_1, \dots, s_M, c_1, \dots, c_M, p_0, \dots, p_N\}$$

mit $M, N \in \mathbb{N}$ und

$$s_k(t) = \sin(\tfrac{2\pi}{12}kt), \qquad c_k(t) = \cos(\tfrac{2\pi}{12}kt), \qquad p_k(t) = t^k.$$

e) Stellen Sie für N, M beliebig (mit $N + 2M + 1 = n \leq m = 36$) das zum Problem

$$\min_{\varphi \in X_{M,N}} \sum_{i=1}^{m} (\varphi(t_i) - b_i)^2$$

zugehörige lineare Ausgleichsproblem auf. Lösen Sie das Problem für verschiedene Werte von N und M und plotten Sie ihre Ergebnisse zusammen mit den Datenpunkten (t_i, b_i). Was passiert, wenn M zu groß wird? Inwiefern eignet sich der polynomiale Anteil, um die langfristigen Tendenzen zu beschreiben?

---

[1] Quelle: `http://statistik.arbeitsagentur.de/Navigation/Statistik/Statistik-nach-Themen/Arbeitslose-und-gemeldetes-Stellenangebot/Arbeitslose/Arbeitslose-Nav.html`