Prof. Dr. Dirk Lebiedz
M.Sc. Pascal Heiter
Dipl.-Math. oec. Klaus Stolle

Universität Ulm
Institut für Numerische Mathematik
Wintersemester 2013/14

Numerische Lineare Algebra - Matlabblatt 5

(Besprechung: Freitag, 17. Januar 2014)

Aufgabe 10 *(Compressed Column Storage Format)* (6 Punkte)

Dünnbesetzte Matrizen werden in speziellen Speicherformaten abgelegt um den Speicheraufwand zu reduzieren. Im Skript, Anhang D.2 wird das **C**ompressed **R**ow **S**torage, kurz CRS Format erklärt. MATLAB nutzt im Gegensatz dazu das CSS Format (Compressed Column Storage), was nichts anderes als das CRS für die transponierte Matrix ist. Es werden also folgende Vektoren benutzt:

- `val` - Wertevektor, welches die Nichtnulleinträge der Matrix A speichert (von oben nach unten, dann links nach rechts)
- `row_ind` - Indexvektor mit den Zeilenindices korrespondierend zu den Einträgen in `val`
- `col_ptr` - Indexvektor mit den Zeiger auf die Spalten, also eine Liste mit den Indices, ab wann eine neue Spalte beginnt.

a) Schreiben Sie eine Funktion

```
[val, row_ind, col_ptr] = CCS(A)
```

die für eine gegebene quadratische Matrix $A \in \mathbb{R}^{n\times n}$ das CCS Format berechnet. Testen Sie ihre Routine mit Hilfe des Skriptes `test_CCS.m`, welches auf der Homepage verfügbar ist.

Hinweis: Sie dürfen annehmen, dass es keine Nullspalten gibt. Nutzen Sie die Funktion `find`.

b) Erweitern Sie das Skript `test_complexity.m` derart, dass die Laufzeiten für die Berechnung des CCS Formats jeweils für das matlabinterne *full* Format und das *sparse* Format doppelt logarithmischen geplottet werden. Achten Sie auf eine saubere Beschriftung des Schaubildes.

c) Schreiben Sie eine Funktion

```
result = MatVecCCS(val,row_ind,col_ptr,x)
```

welches das Matrix-Vektor Produkt Ax berechnet, wobei A im CCS Format übergeben wird. Testen Sie ihre Routine mit Hilfe des Skriptes `test_MatVecCCS.m`, welches auf der Homepage verfügbar ist.

d) Schreiben Sie eine Funktion

```
A = CCStoFull(val,row_ind,col_ptr)
```

welches die Matrix A im gegebenen CCS Format in das volle Format konvertiert. Testen Sie ihre Routine mit Hilfe des Skriptes `test_CCStoFull.m`, welches auf der Homepage verfügbar ist.

Aufgabe 11 *(QR-Zerlegung)* (6 Punkte)

Die Aussage von Aufgabe 19 auf Blatt 5 lässt sich auf nichtquadratische Matrizen $A^{m\times n}$ mit $m \geq n$ und $\operatorname{rang}(A) = n$ erweitern.

a) Schreiben Sie eine Funktion

```
[Q,R] = myqr(A)
```

die zu einer gegeben Matrix $A^{m\times n}$ mit $\operatorname{rang}(A) = n$ die QR-Zerlegung von A mit Hilfe der Cholesky-Zerlegung von $A^T A$ berechnet - siehe dazu Blatt 5, Aufgabe 19.

b) Schreiben Sie ein Testskript `test_myqr.m`, welches Ihre Routine für folgende Matrizen testet

- `A = rand(10)`
- `A = [rand(10); hilb(10)]`
- `A = rand(20,3)`.

Aufgabe 12 *(Ausgleichsprobleme)* (8 Punkte)

Lineare Ausgleichsprobleme der Form

$$\text{(P)} \quad \min_{x \in \mathbb{R}^n} ||y - Ax||_2^2$$

kann man u.a. direkt über die Normalengleichung

$$A^T Ax = A^T y$$

lösen oder mittels QR-Zerlegung.

a) Schreiben Sie eine Funktion

```
x = leastSquares(A,y,option)
```

die zu einer gegeben Matrix $A^{m \times n}$ mit $\operatorname{rang}(A) = n$ und $y \in \mathbb{R}^n$ das lineare Ausgleichsproblem (P) mit verschiedenen Methoden löst. Dabei wird der Name der Methode (= `option`) als `String` übergeben. Die Methoden sind

- die Normalengleichung mittels Choleskyzerlegung lösen (`option = 'cholesky'`)
- die QR-Zerlegung von A berechnen und damit das Minimierungsproblem auf ein lineares Gleichungssystem transformieren (vgl. Mitschrieb aus der Vorlesung) - dabei soll zur Berechnung der QR-Zerlegung die Matlab-interne Funktion `qr` verwendet werden (`option = 'qr'`) und als Alternative ihre eigene Funktion `myqr` aus Aufgabe 11 (`option = 'myqr'`).

Testen Sie ihre Funktion mit Hilfe des Skriptes `test_leastSquares.m`, welches auf der Homepage verfügbar ist.

Hinweis: Verwenden Sie für den Vergleich von Strings die Funktion `strcmp`.

b) Erweitern Sie das Skript `compare_leastSquares.m` derart, dass die einzelnen Fehler der verschiedenen Methode logarithmisch geplottet werden. Achten Sie auf eine saubere Beschriftung und interpretieren Sie das Ergebnis. Erklären Sie die Vor- und Nachteile der einzelnen Methoden.

c) Schreiben Sie ein Skript `linearRegression.m`, welches für folgende Daten $D = \{(x_i, y_i)\} \subset \mathbb{R}^2$ mit

$$D := \{(0,0),(1,0),(1,3),(2,0),(2,1),(2,2),(3,2),(3,3),(4,4),(4,5),(5,7),(7,4)\}$$

die Regressionsgerade

$$g(x) = m^* x + c^*$$

berechnet. Es gilt also

$$(m^*, c^*) = \arg \min_{(m,c) \in \mathbb{R}^2} \sum_{i=1}^{|D|} (y_i - (mx_i + c))^2.$$

Plotten Sie D und die Ausgleichsgerade g auf [0,7] und achten Sie dabei auf eine saubere Beschriftung.