

(kurze) Einführung in GNU Octave

Florian Mücke
26. Oktober 2005

Octave ist eine Hochsprache, die in erster Linie für numerische Berechnungen bestimmt ist. Es bietet ein zweckmäßiges Kommandozeileninterface zur numerischen Lösung linearer und nichtlinearer Probleme. Die Syntax ist dabei weitestgehend MatLab kompatibel. And best of all: it's free!

Octave gibt's generell hier:

<http://www.octave.org>

für Windows gibt es allerdings optimierte Pakete wie z.B. hier:

<http://sourceforge.net/projects/octave/>

<http://www.site.uottawa.ca/~adler/octave/octave-2.1.42-p6a.exe>

Hierbei ist immer das Programm **Gnuplot** zur Ausgabe von Grafiken enthalten.

Einführung in die Befehlsyntax

Octave behandelt grundsätzlich alles als Matrizen. Vektoren sind dabei Matrizen mit nur einer Zeile oder Spalte. Man kann direkt in der Eingabeaufforderung von Octave rechnen. Octave behandelt dabei auch +, -, *, * und / als Matrixoperationen.

1. Vektoren

- Zeilenvektoren
`>> v=[1 2 3]` oder
`>> v=[1,2,3]`
- Spaltenvektoren
`>> v=[1;2;3]`

2. Matrizen

$M = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ wird erzeugt durch

```
>> M=[1 2;3 4]
```

3. Zugriff auf Matrixelemente

Man kann direkt über den Positionsindex auf die einzelnen Elemente von Vektoren und Matrizen zugreifen. Bereiche sind dabei genauso erlaubt. Dabei wird folgendermaßen auf die Elemente zugegriffen: `>> M(Zeilennummer, Spaltennummer)`

```
>> M(2) liefert  
ans = 2 als Ergebnis (ans ist die interne Variable für den Rückgabewert)
```

```
>> M(2,1) liefert  
ans = 3 als Ergebnis
```

```
>> M(:,2)=0 erzeugt  $M = \begin{bmatrix} 1 & 0 \\ 3 & 0 \end{bmatrix}$ 
```

Für einige wichtige $m \times n$ -Matrizen existieren eigene Befehle. Falls $m=n$ ist, ist nur ein Argument als Parameter anzugeben.

```
>> eye(m,n) erzeugt eine Matrix mit Einsen auf der Hauptdiagonalen (Einheitsmatrix für m=n)
```

```
>> zeros(m,n) erzeugt eine Nullmatrix
```

```
>> ones(m,n) erzeugt eine Matrix mit lauter Einsen
```

```
>> rand(m,n) erzeugt eine Matrix mit Zufallszahlen
```

4. wichtige Befehle und grundlegende Operationen

- Elementweise Operationen: alle Operationen mit vorangestelltem Punkt werden mit den einzelnen Elementen verknüpft.
`>> M.^2` quadriert jedes Element von M einzeln, anstatt $M \cdot M$ zu berechnen
- Transponieren: zum Transponieren von Matrizen wird ein simples Hochkomma angehängt.
`>> v = v'` transponiert v und speichert diesen wieder unter dem selben Namen
- wichtige Befehle
`>> help befehl` liefert Hilfe zum jeweiligen Befehl
`>> dot(u,v)` berechnet das Skalarprodukt von u und v
`>> norm(m)` berechnet die euklidische Norm von m – Näheres mit `help norm`
`>> abs(m)` liefert den Betrag von m
`>> diag(M)` liefert die Diagonalelemente von M
`>> diag(v)` erzeugt eine Matrix mit v als Diagonale
`>> inv(M)` berechnet die Inverse zu M
`>> whos` Anzeige aller angelegter Variablen und Funktionen
`>> clear` löscht alle Variablen (nicht aber die Funktionen)
`>> load/save Dateiname` lädt(speichert) die aktuelle Variablenumgebung aus(in) eine(r) Datei

5. Funktionen

```
>> function y = f(x)
y = x^2;
end

>> f(3)
ans = 9
```

Wir definieren also eine Funktion f , die das Quadrat der Eingabe berechnet. Der Rückgabewert y wird dabei einfach der Funktion x^2 zugewiesen. Damit das Zwischenergebnis y nicht angezeigt wird, kann man dessen Ausgabe mit einem Strichpunkt am Ende der Zeile unterbinden. $f(M)$ liefert dasselbe wie $M \cdot M$.

6. Skriptdateien

Man kann Octave Befehle einfach in Textdateien mit der Endung `.m` abspeichern und diese dann einfach in Octave aufrufen. Dazu muss sich die Datei im richtigen Verzeichnis befinden - das Standard-Benutzerverzeichnis von ist unter Windows XP z.B.:

```
C:\Programme\GNU Octave\octave_files
```

Dateien mit der Endung `.m` können dann direkt in Octave aufgerufen werden.

Beispiel `test.m`

```
eins = 1;
zwei = 2;
drei = eins + zwei
```

```
>> test
drei = 3
```

7. Links

- [Einführung in GNU Octave](#) von Hubert Selhofer, mit Änderungen von Marcel Oliver
- [Octave Manual](#) von John W. Eaton
- [ScaryOctave](#) Wiki