

Learning MATLAB by doing MATLAB (1)

Christian Mehl*

Andreas Steinbrecher[†]

5. November 2002

Dieses kleine MATLAB-Tutorial setzt auf "Learning by Doing". Gib jeweils die hinter > angegebenen Befehle im MATLAB Command Window ein und beobachte, was bei der Ausgabe passiert.

1. Variablen, Vektoren, Matrizen

>a=7	a wird als Skalar interpretiert (oder 1×1 -Matrix)
>b=[1,2,3]	nach Komma: neues Element in derselben Zeile, also hier $b \in \mathbb{R}^{1,3}$
>c=[1+2,3,3]	
>d=[7 7 2]	Leerzeichen haben gleiche Bedeutung wie Kommata
>e=[7 a 2]	
>f=[1;2;3;4]	Semikolon: Beginn einer neuen Zeile, also hier $f \in \mathbb{R}^{4,1}$
>g=f(2)	fragt das 2. Element des Spaltenvektors f ab
>E=[1 2 3;2 1 3]	MATLAB unterscheidet zwischen Groß- und Kleinschreibung
>h=E(1,2)	fragt das (1,2)-Element von E an
>E	
>F(3,4)=7	MATLAB interpretiert F zunächst als 3×4 -Matrix. Nicht benannte Elemente werden auf Null gesetzt.
>F(4,3)=2	Jetzt brauchen wir eine 4. Zeile!
>F(1,2)=3	Belegt das (1,2)-Element von F mit 3.
>F(1:2,3:4)=[1 3;2 7]	1 : 2 bedeutet Zeile 1 bis Zeile 2; 3 : 4 bedeutet Spalte 3 bis Spalte 4
>who	gibt aus, welche Variablen belegt sind
>clear a b	löscht die Variablen a und b
>who	
>clear	löscht alle Variablen
>help clear	gibt Hilfe zum Befehl <code>clear</code>
>A=[1 2];	Semikolon am Ende unterdrückt die Ausgabe
>A	
>pi	
>A=eye(3)	3×3 -Einheitsmatrix
>b=[1 2 3]	
>B=diag(b)	Diagonalmatrix
>C=diag([1 7 8])	
>D=diag([1;7;8])	geht auch mit Spaltenvektoren
>E=ones(4)	4×4 -Matrix mit Einsen
>F=ones(2,3)	2×3 -Matrix mit Einsen
>G=zeros(4)	
>H=zeros(2,3)	
>I=[A B;zeros(3) A]	Blockmatrix
>C	
>C'	Transponierte von C
>w(3)=5	MATLAB interpretiert w als Zeilenvektor. Das 3. Element wird gleich 5 gesetzt.

*Institut für Mathematik, Technische Universität Berlin, D-10635 Berlin, Germany, (mehl@math.tu-berlin.de)

[†]Institut für Mathematik, Technische Universität Berlin, D-10635 Berlin, Germany, (anst@math.tu-berlin.de)

2. Einfache Operationen

```
>clear
>A=[1 2 3;2 1 0]
>B=[2 2;1 0;0 1]
>C=[0 1 0;5 1 3]
>size(A)           Gibt Anzahl der Zeilen und Spalten von A als Zeilenvektor aus.
>[m,n]=size(A)    So bekomme ich sie einzeln.
>b=[2 1 3]
>x=[2;1;3]
>A.*C             elementweise Matrixmultiplikation
>who              ans (für "answer") ist die unbenannte Ausgabevariable
>D=A+C           Matrixaddition
>E=A+B           Fehler, da Dimensionen nicht passen!
>E=A-B'           $E = A - B^T$ 
>g=A*x           Matrix mal Vektor
>g=A*b           Fehler!
>A
>b
>C=[1 2 3]'
```

3. Matrixmanipulationen

```
>clear
>A=[1 2;3 4]
>A(3,2)=7         Hinzunahme einer dritten Zeile!
>A(1:2,2)         (1:2,2): 1. bis 2. Element der 2. Spalte
>A(3,1:2)         (3,1:2): 1. bis 2. Element der 3. Zeile
>B(3:4,3)=[5;6]  MATLAB erzeugt eine Matrix B bei der das 3. und 4. Element
                  der 3. Spalte 5 bzw 6 ist. Alle anderen Einträge sind Null.
>C(4:5,4)=A(1:2,2)
>B(:,3)          3. Spalte von B
>d=C(1,:)        1. Zeile von C
>E=[1 2 3;4 5 6;7 8 9;10 11 12]
>E(1:2:4,3)      (1:2:4,3): Sucht in der 3. Spalte vom 1. bis 4. Element
                  jedes 2. Element heraus.
>F=[1 2 3 4 5;6 7 8 9 10;11 12 13 14 15]
>G=F(1:2:3,1:2:5) Sucht in der 1., 3. und 5. Spalte jeweils vom
                  1. bis 3. Element jedes 2. Element heraus.
>H=[1 3;9 11]
>det(H)          Die Determinante.
>b=[99 100 101]
>F(1,1:3)=b
>A
>A(1,1:3)=b      Das (2,3)- und das (3,3)-Element werden neu hinzugefügt.
```