



# Einführung in Matlab

## 1. Grundlagen

Prof. Dr. Christiane Zarfl, Dipl.-Inf. Willi Kappler

---



## 1 Einleitung

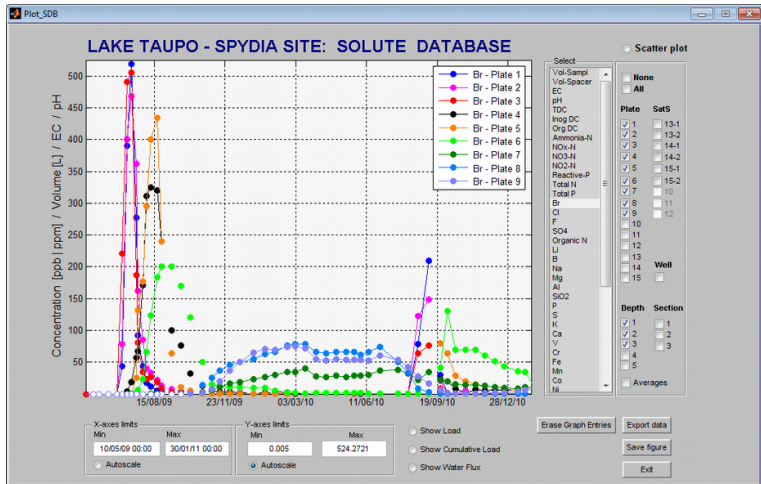
## 2 Grundlagen



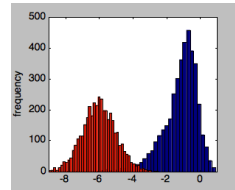
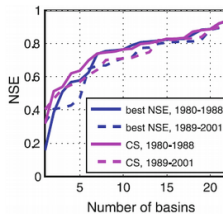
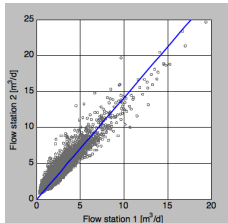
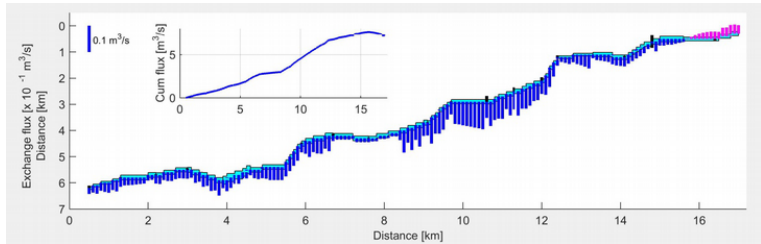
- Einfache Rechnungen
- Datenaufbereitung & -speicherung
- Visualisierung von Daten & Simulationsergebnissen
- einfaches Lösen von (linearen) Gleichungssystemen, Differentialgleichungen (DGL), DGL-Systemen...
- Wiederverwendung von häufig gebrauchten Berechnungen (Programmierung)
- Datenanalyse (z.B. Regression) & Statistik
- Komplexe Modellierung von Umweltsystemen



## Visualisierung von Daten & Simulationsergebnissen

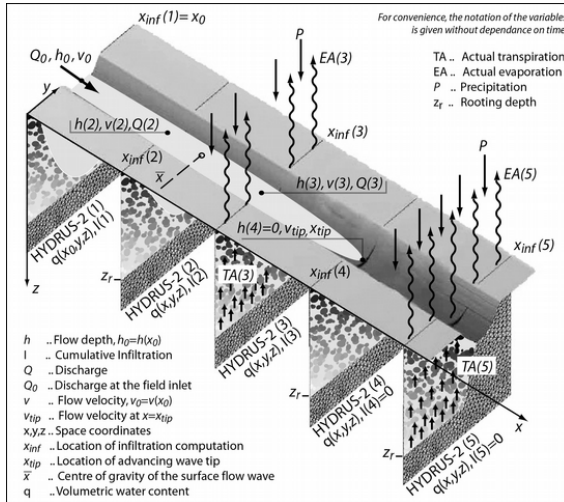


## Datenanalyse & Statistik





## Modellierung von Umweltsystemen





- Hardware:
  - BYOD (eigenes Gerät mitbringen)
  - Geo-Notebooks (Raum S245)
  - CIP Pool Rechner (Raum S310)
- Software:
  - Auf Institutshardware bereits vorinstalliert
  - ZDV: Matlab [▶ hier](#) herunterladen
  - GNU Octave [▶ Webseite](#) (Open Source, aber nicht 100% kompatibel)
- Auf Institutshardware bitte zuerst ein eigenes Verzeichnis anlegen!



- “*Matlab* ist eine kommerzielle Software des Unternehmens *The MathWorks, Inc.* zur Lösung mathematischer Probleme und zur grafischen Darstellung der Ergebnisse.” (Quelle: [▶ Wikipedia](#)).
- Matlab leitet sich ab von **MAT**rix **LAB**oratory.
- Wir benutzen Matlab als (numerische) Programmiersprache.
- Wie ein Taschenrechner oder Excel arbeitet Matlab numerisch (mit Zahlenwerten, also nicht symbolisch wie ein [▶ CAS](#)).
- Anders als bei einem Taschenrechner können Zahlenwerten Variablennamen zugewiesen werden.
- Im Programm werden die Variablennamen als Platzhalter für die Werte verwendet.





- kennen Sie den Aufbau der Oberfläche der Software Matlab.
- benutzen Sie die Matlab-Hilfe, um für Sie nützliche Funktionen und Informationen selbstständig zu finden.
- führen Sie einfache Rechnungen mit Matlab durch.
- können Sie Variablen in Matlab definieren und verwenden.
- kennen Sie die Vorteile der Verwendung von Vektoren und können diese in Matlab definieren und für Rechnungen verwenden.



## gegenwärtiges Verzeichnis


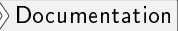

The screenshot shows the MATLAB R2014a environment. The 'Current Folder' pane on the left lists files: 'Hausaufgaben', 'Woche1', 'Woche2', 'Woche3', and 'template.m'. An arrow points from the text 'Dateien im gegenwärtigen Verzeichnis' to this pane. The 'Editor' pane in the center shows the 'template.m' script. A yellow highlight covers the variable declarations: `x = 0:0.1:5;`, `y1 = min(x);`, `y2 = min(3*x);`, and `y3 = min(0.4*x);`. An arrow points from the text 'Variablen im Arbeitsspeicher' to this highlighted section. The 'Workspace' pane on the right shows the variables: `x` (1x51 double, 0 to 5), `y1` (1x51 double, -0. to 0.9), `y2` (1x51 double, -0. to 0.9), and `y3` (1x51 double, 0 to 0.9). An arrow points from the text 'Eingabeverlauf' to this pane. The 'Command Window' at the bottom shows the prompt `>>` and the command `template`. An arrow points from the text 'Eingabefenster mit "Prompt" (>>)' to this window.

↑  
Dateien im  
gegenwärtigen  
Verzeichnis


↑  
Variablen im  
Arbeitsspeicher

↑  
Eingabeverlauf

↑  
Eingabefenster mit  
"Prompt" (>>)

- Niemand kann alle Befehle kennen, deshalb ist die (ausführliche) Hilfe in Matlab so wichtig.
- Allgemeines Hilfe-Fenster:   (  )
- Information zu einem Befehl:
  - `doc <Befehlsname>` (Info in einem extra Fenster)
  - `help <Befehlsname>` (Info im Befehlsfenster)
- Beispiele: Im Prompt eingeben:
  - `help sin`
  - `help exp`
  - `doc plot`



- Alle Anweisungen werden nach dem Prompt (`>>`) eingegeben und mit  (Return) bestätigt. Matlab nennt das Ergebnis `ans` (für answer):

```
1 12/3 + 7*5 - 1
```

```
1 ans = 38
```

- Addition ( $\oplus$ ), Subtraktion ( $\ominus$ ), Multiplikation ( $\otimes$ ) und Division ( $\oslash$ ) wie im Taschenrechner (Matlab kennt “Punkt vor Strich-Rechnung”).
- Braucht man einen **Ausdruck öfters**, so kann man ihn als **Variable** definieren:

```
1 a = 48/3 - 3^2
```

```
1 a = 7
```

- Variablen werden im Arbeitsspeicher (Workspace) gespeichert (s. Arbeitsspeicher-Fenster).



- Ein Semikolon (;) am Ende der Eingabezeile unterdrückt die Ausgabe des Ergebnisses.
- Matlab **unterscheidet** zwischen Groß- und Kleinbuchstaben!
- Potenzieren wird **vor** einer **Multiplikation** oder **Division** ausgewertet, sonst gilt “Punkt-vor-Strich”; runde Klammern “(” und “)” um die Reihenfolge der Berechnung zu steuern.
- Mehrere Anweisungen in einer Zeile sind zulässig:
  - Sind sie durch ein Komma getrennt, so folgt eine Ausgabe.
  - Werden sie durch ein Semikolon getrennt, so folgt keine Ausgabe:

```
1 b = (3+5)*6;  
2 c = (b/3)^2, d = 1/c^2
```

```
1 c = 256  
2 d = 1.5259e-05
```




- Regeln bei der Definition von Variablen:
- Das erste Zeichen muss ein Buchstabe sein (Keine Zahl!)
- Keine Sonderzeichen (außer Unterstrich)
- Max. Zeichenlänge (abhängig vom Computer)
- **Vorsicht:** Variablennamen identisch mit Funktionen ist erlaubt, hat aber Seiteneffekte!
- Variablen haben einen bestimmten **Typ**, z.B. Ganzzahl, Fließkommazahl, Vector, Matrix, ...





- `who`: gibt eine Liste der Variablen im Arbeitsspeicher aus
- `whos`: gibt zusätzliche Information (Typ, Größe, Speicherbedarf)
- `clear <Variable>`: löscht die Variable
- `clear all`: löscht alle Variablen
- `clc`: löscht den Inhalt des Befehlsfensters




Geben Sie nacheinander folgende Anweisungen ein. Überlegen Sie vorher was wird Matlab ausgeben?

`u = 2, v = 5;` 

`(u+6) / 4` 

`y = x+1` 

`y = 3u` 

Welche der folgenden Variablennamen sind **nicht** zulässig?

`anzahl`, `Summe_a+b`, `5_Tageskarte`, `dauer_phase3`, `sin`





- Es gibt in Matlab bereits “eingebaute” Funktionen (viel mehr als im Taschenrechner und Excel).
- Z.B. die Wurzelfunktion (square root): `a = sqrt(2)/2`
- Nähere Informationen zur Funktion `sqrt` erhält man mit `help sqrt`.
- Funktionen können keinen, einen oder mehrere **Eingabeparameter** haben.
- Funktionen können keinen, einen oder mehrere **Rückgabewerte** haben.
- Wie bei Variablen auch haben Funktionen einen bestimmten Typ. D.h. die Ein- und Ausgabewerte **müssen** vom Typ her passen.



Berechnen Sie den natürlichen Logarithmus von  $1.36$

Berechnen Sie auch den Logarithmus zur Basis  $10$  von  $1.36$

(Zusatz: Wie berechnen Sie den Logarithmus zur Basis  $3$  von  $1.36$ ?)



Berechnen Sie  $\cos(\pi)$  und  $\cos(\pi/2)$

$\pi$  ist in Matlab bereits eingebaut und wird mit `pi` bezeichnet

**Vorsicht:** Das Argument der trigonometrischen Funktionen (`sin`, `cos`, `tan`, `cot`) wird von Matlab immer im **Bogenmaß** interpretiert

Berechnen Sie den Kosinus von  $180^\circ$  und  $90^\circ$



- Eine Gruppierung von mehreren Zahlenwerten nennt man einen **Vektor**
- Eine zweidimensionale Gruppierung von Zahlen nennt man eine **Matrix**
- **Es folgt:** Eine Zahl ist sowohl ein spezieller Vektor (der Länge 1), als auch eine spezielle Matrix der Dimension  $1 \times 1$ .
- **Ein Vektor der Länge  $n$**  ist eine spezielle Matrix der Dimension  $n \times 1$  (Spaltenvektor) oder  $1 \times n$  (Zeilenvektor)
- Matlab (**MAT**rix **LAB**oratory) kennt intern nur Matrizen!
- Die Berechnung der Wurzel in Matlab von vorhin:

```
a = sqrt(2)/2
```

Hier ist  $a$  eine  $1 \times 1$  Matrix (Selbst die Konstante 2 wird als eine konstante  $1 \times 1$  Matrix interpretiert)



- Vektoren werden in Matlab immer in eckigen Klammern eingegeben: “[” und “]”
- Die Elemente des Vektors sind durch Leerzeichen (oder ein Komma) zu trennen (es ergibt sich ein Zeilenvektor, Semikolon als Trenner ergibt einen Spaltenvektor).
- Um in Matlab Tipparbeit zu sparen können Vektoren verwendet werden. Möchte man z.B. die Quadratwurzel mehrerer Werte berechnen, so geht man folgendermaßen vor:
- Zeilenvektor in einer Variablen definieren: `x = [0 2 4 6 8 10]`.
- Die Quadratwurzelfunktion auf den gerade definierten Zeilenvektor `x` anwenden: `y = sqrt(x)`.
- Das Ergebnis in der Variablen `y` ist nun ebenfalls ein Zeilenvektor und enthält die einzelnen Quadratwurzel der oben aufgeführten Zahlen.