

# Einführung in Matlab

**Christopher Mink, Klemens Schmitt**

AP/FP-Seminarreihe

Fachschaft Physik, TU Kaiserslautern

**15. Februar 2017**

# Softwarekurse zum AP/FP

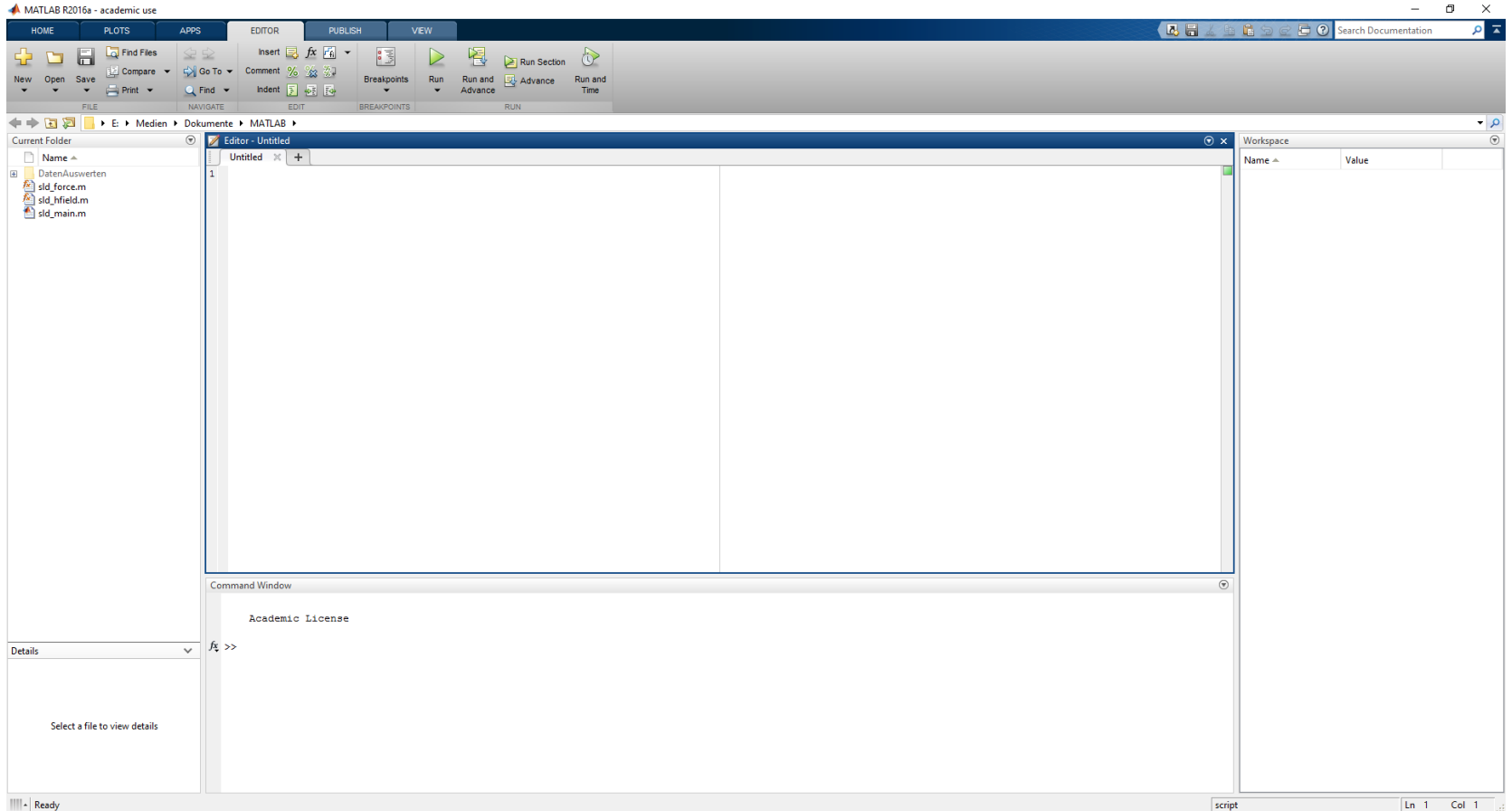
- **3 Programme:**
  - Heute: Matlab
  - Morgen: gnuplot
  - Übermorgen: Mathematica
- **Struktur**
  - Einführender Vortrag
  - Zeit zum eigenen Testen

# Übersicht

- 1) Grundlagen
- 2) Daten erzeugen
- 3) Plotten
- 4) Fitten
- 5) Daten importieren

# Download Matlab

- Im Uni-VPN: <https://serviceportal.rhrk.uni-kl.de/>
- Software-Download → Matlab für Studierende
- Anleitung befolgen
  
- Alternativ: Open-Source-Programm „Octave“



MATLAB R2016a - academic use

HOME PLOTS APPS EDITOR PUBLISH VIEW

Find Files Insert fx  
New Open Save Compare Go To Comment % Indent Breakpoints Run Run and Advance Run and Time  
Print Find Indent Breakpoints Run Run and Advance Run and Time

FILE NAVIGATE EDIT BREAKPOINTS RUN

Current Folder: E:\Medien\ Dokumente\MATLAB

Editor - Untitled

Workspace

Command Window

Academic License

f>>

Ready script Ln 1 Col 1

# Skript erzeugen und ausführen

- Editor mit grafischer Oberfläche
- Neues Skript (\*.mat) erstellen
- Code einfügen
- „Run“ oder F5 drücken
- Beachte:
  - Zeilen, die mit einem Semikolon enden, erzeugen keine Ausgabe
  - Das Symbol % erzeugt Kommentare
  - „Help“ bzw. F1 ist dein Freund!

# Vektoren erzeugen

```
>> A = [1 2 3]
A =
     1     2     3
>> B = [1; 2; 3]
B =
     1
     2
     3
```

# Vektoren intelligent erzeugen

```
% Dies ist ein Kommentar

% Werte in Intervall erzeugen
y = 1:5;
z = 1:.2:12; % .2 = 0.2: Schrittweite

% So geht das ebenfalls:
x = linspace(-pi,pi,1000);
```



# Matrizen erzeugen

```
>> C = [1 2; 3 4; 5 6]
```

```
C =
```

```
1 2
```

```
3 4
```

```
5 6
```

# Matrizen intelligent erzeugen

```
% 20x10 Matrix mit 1 in allen Einträgen  
k = ones(20, 10)  
  
% 10x10 Null-Matrix  
l = zeros(10);  
  
% 20x20 Einheitsmatrix  
m = eye(20);  
  
% 5x5 Matrix mit Diagonalelementen 1 bis 5  
n = diag(1:5);
```

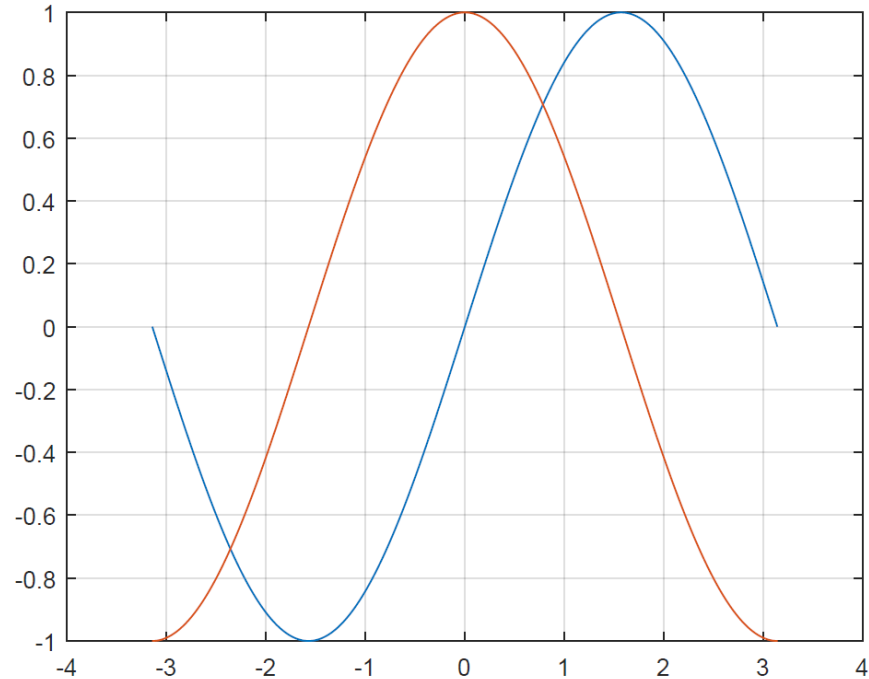
# Skalar- und Matrixprodukte

```
>> A*B
ans = 14
>> B*A
ans =
     1     2     3
     2     4     6
     3     6     9
>> B' .* A
ans =
     1     4     9
```

# Plotten

```
x = linspace(-pi,pi,1000)
y = sin(x);
y2 = cos(x);
plot(x,y,x,y2);

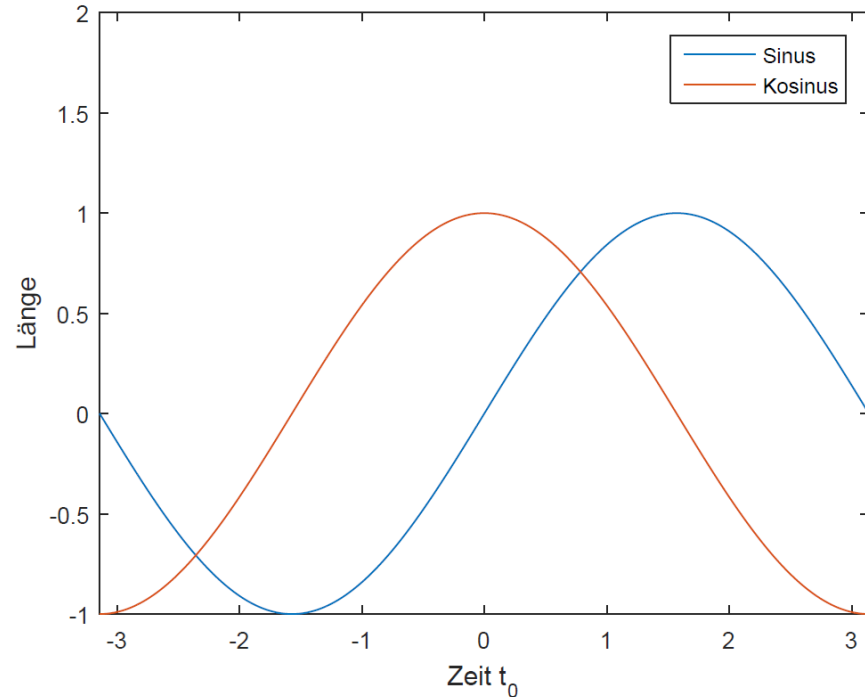
% alternativ
plot(x,y)
hold on; % Graph nicht überschreiben
plot(x,y2);
hold off;
```



# Wertebereich

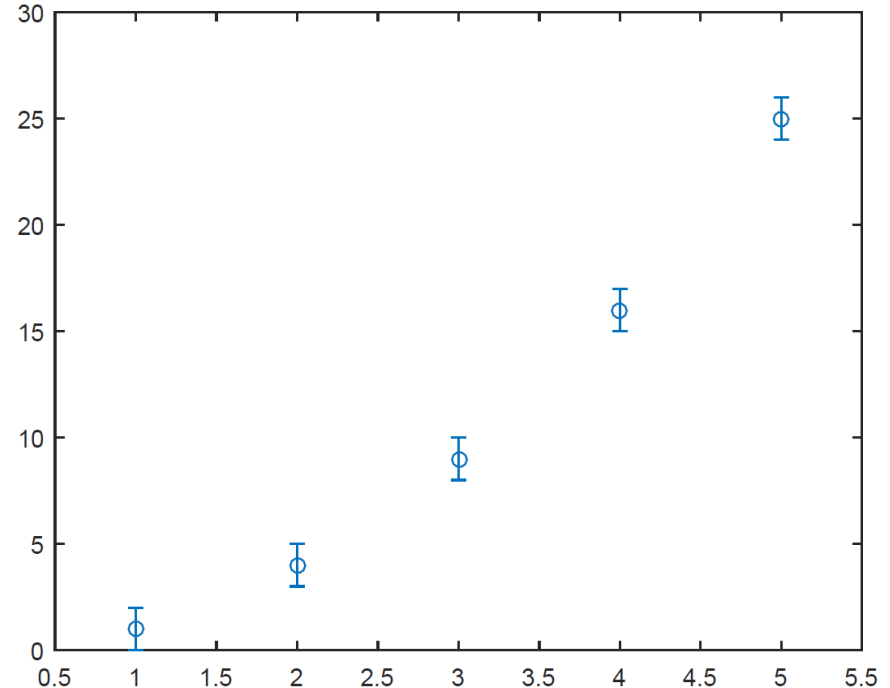
```
plot(x, sin(x), x, cos(x));  
xlim([-pi,pi]);  
ylim([-1,2]);  
legend('Sinus', 'Kosinus');  
xlabel('Zeit t_0');  
ylabel('Länge');
```

- Mit dem grafischen Menü können weitere Anpassungen per Hand vorgenommen werden.



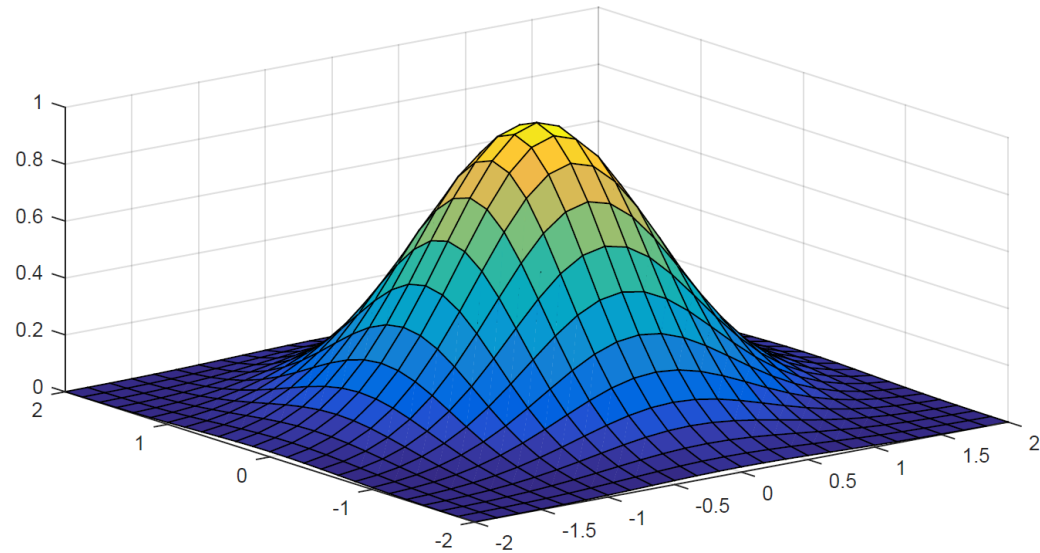
# Fehlerbalken

```
x = 1:5;  
y = x.^2; % Quadriere elementweise  
  
% Fehler: Reihenvektor mit 1ern  
z = ones(1, 5);  
  
errorbar(x, y, z, 'o');
```

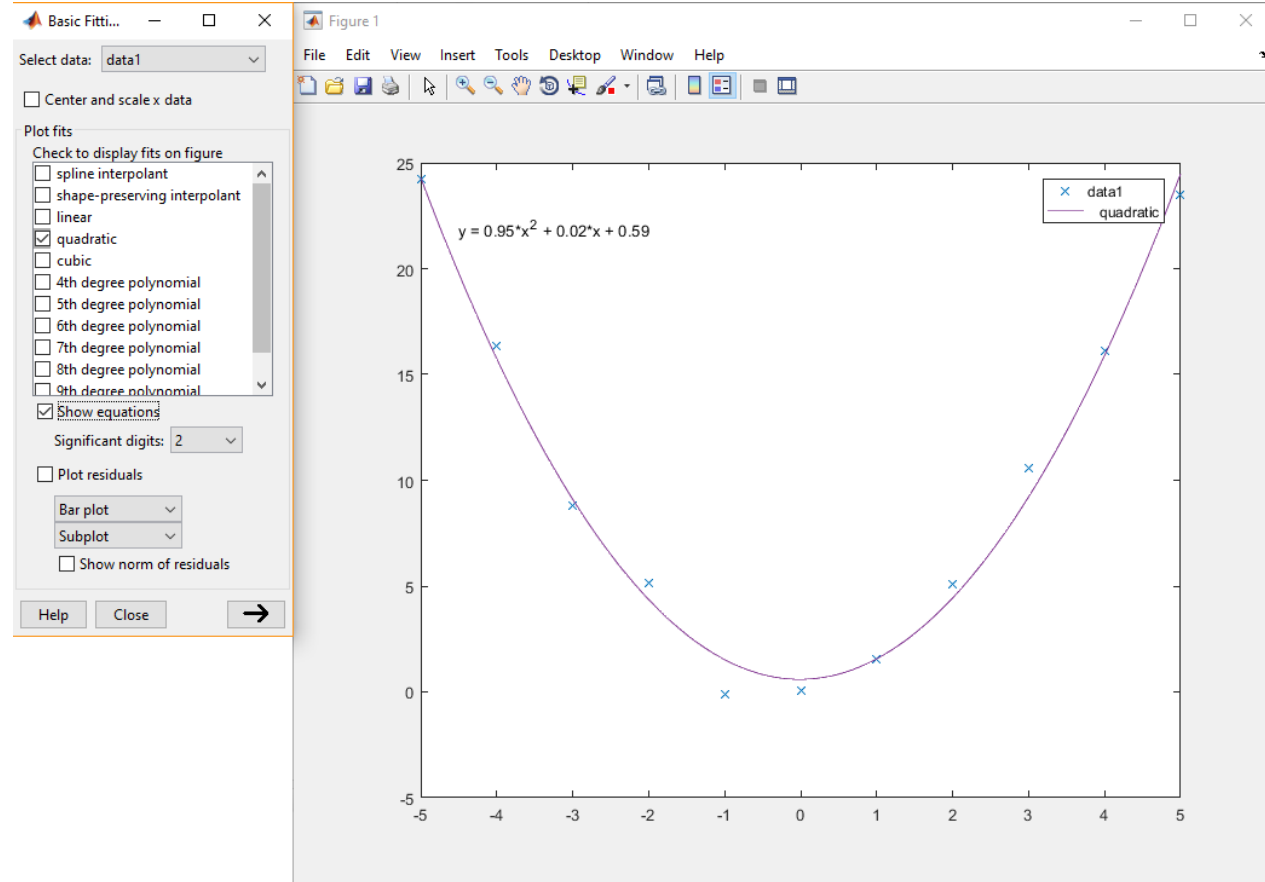


# 3D-Plots

```
x = linspace(-2, 2, 25);  
[X, Y] = meshgrid(x, x);  
z = exp(-(X.^2 + Y.^2));  
  
surf(x, x, z)
```



- Grafisch Fitten:  
Plotfenster → Tools →  
Basic Fitting
- Für mehr Kontrolle:  
textbasiert fitten, z.B.  
<https://de.mathworks.com/help/curvefit/fit.html>





# Textdatei importieren

```
% definiere Eingaben für den Import-Befehl
filename = 'Herzfrequenz.csv';
delimiterIn = ',';
headerlinesIn = 1;

% importiere Tabelle in Matlab-Objekt
tab = importdata(filename,delimiterIn,headerlinesIn);

% generiere Vektoren für x- und y-Werte aus importierter Tabelle
xcol = tab.data(:, 1);
ycol = tab.data(:, 2);
```

# Noch Fragen?

- Falls nein:
  - Installiert Matlab
  - Probiert euch an den gezeigten Problemstellungen
  - Stellt Fragen, wann auch immer etwas unklar ist!