

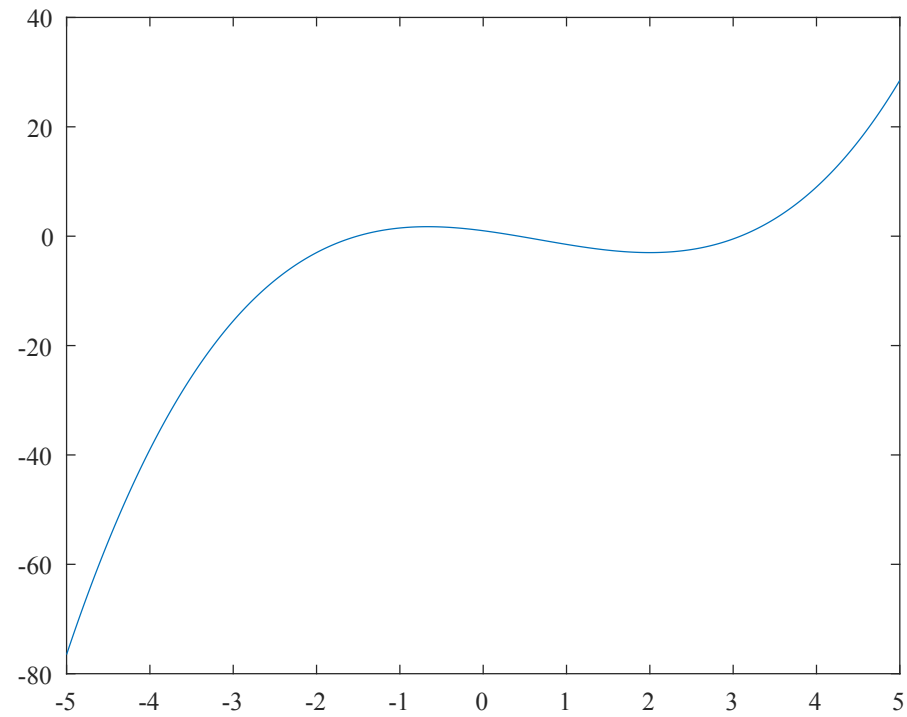
# Visualisierung mit Matlab

# Visualisierung mit Matlab

- Visualisierung mathematischer Funktionen und Signale
- Gestaltung der Diagramme
- Darstellung von 3-D Diagrammen

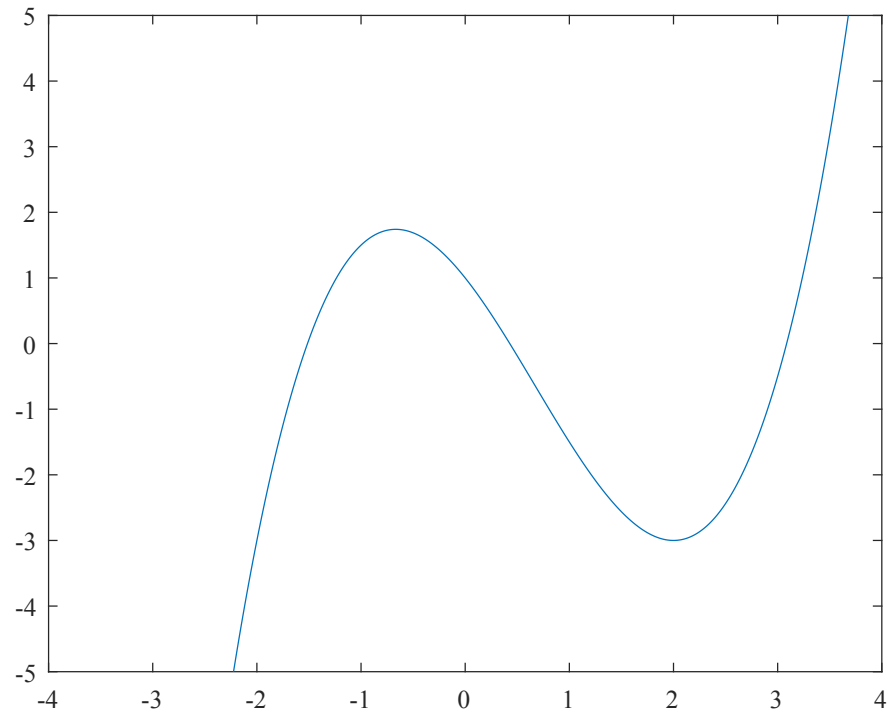
## Visualisieren mathematischer Funktionen

$$y = f(x) = \frac{x^3}{2} - x^2 - 2x + 1$$



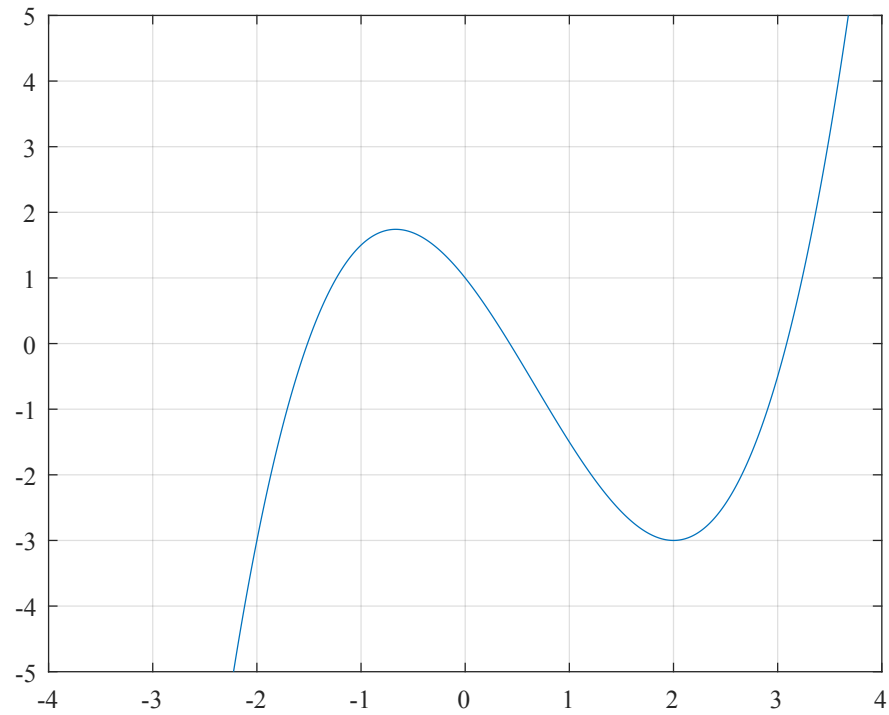
## Individuelle Gestaltung der Diagramme: Zoom

$$y = f(x) = \frac{x^3}{2} - x^2 - 2x + 1$$



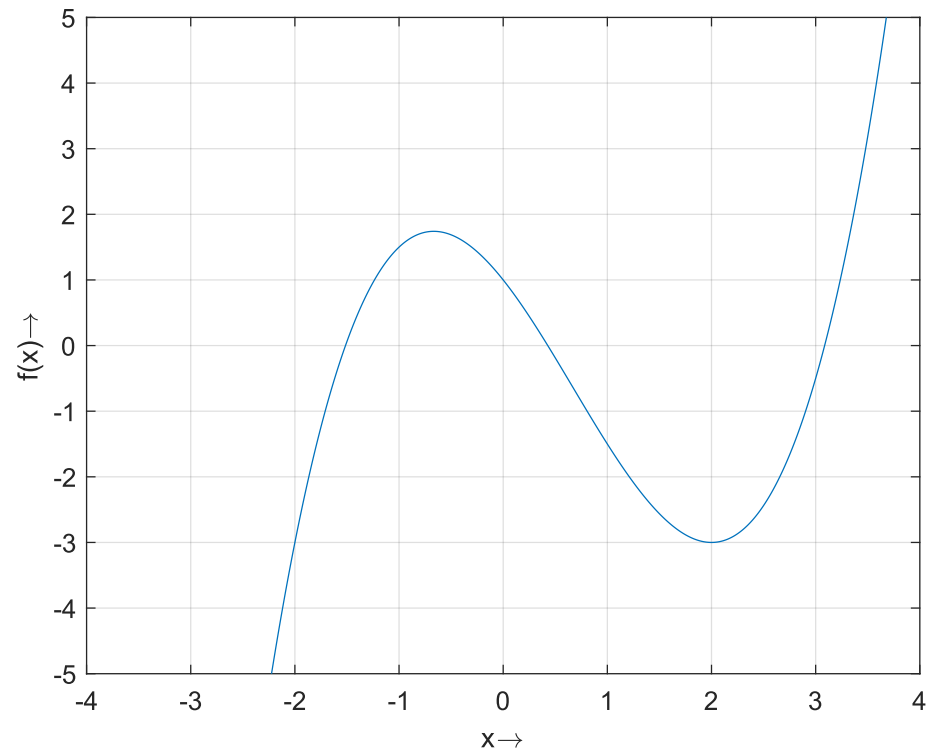
## Individuelle Gestaltung der Diagramme: Gitternetz

$$y = f(x) = \frac{x^3}{2} - x^2 - 2x + 1$$



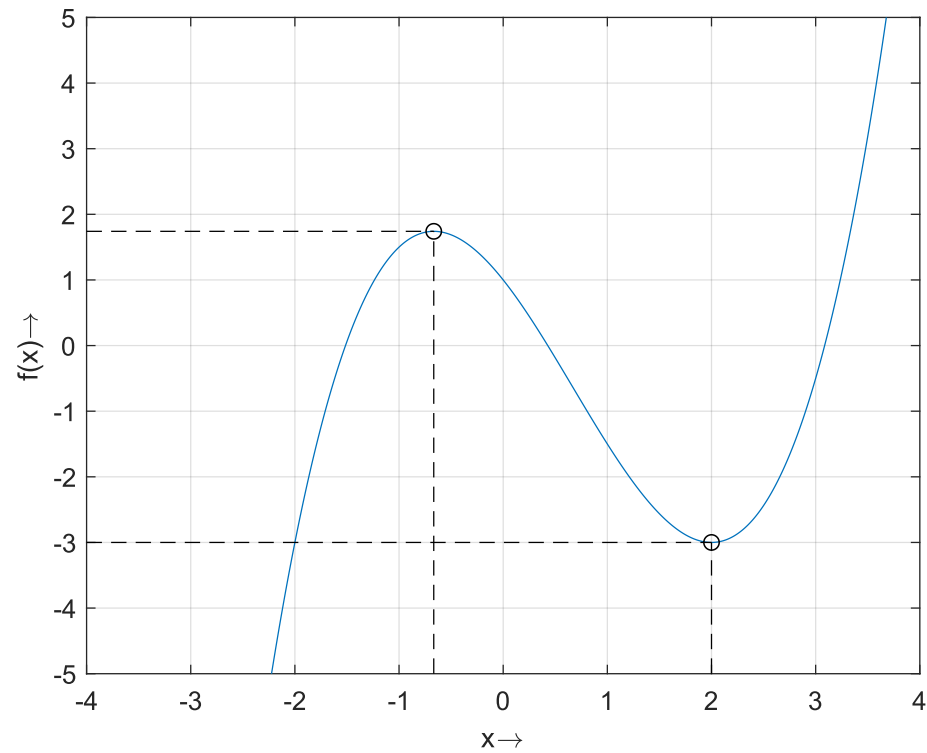
## Individuelle Gestaltung der Diagramme: Achsenbeschriftung

$$y = f(x) = \frac{x^3}{2} - x^2 - 2x + 1$$



## Individuelle Gestaltung der Diagramme: Hervorheben der Extrempunkte

$$y = f(x) = \frac{x^3}{2} - x^2 - 2x + 1$$



## Der `plot` Befehl

Gegeben sei folgende Messreihe

Zeit	0.0	0.5	1.0	1.5	2.0	2.5	3.0
Messwerte	2.1	5.7	9.8	7.2	1.4	3.7	5.0

**Aufgabe:** Stellen Sie diese Messwerte grafisch über der Zeit dar

**Vorbereitung:** Anlegen zweier Vektoren für `zeit` und `messwerte`

```
>> zeit = 0:0.5:3;
```

```
>> messwerte = [2.1 5.7 9.8 7.2 1.4 3.7 5.0];
```

**Durchführung:**

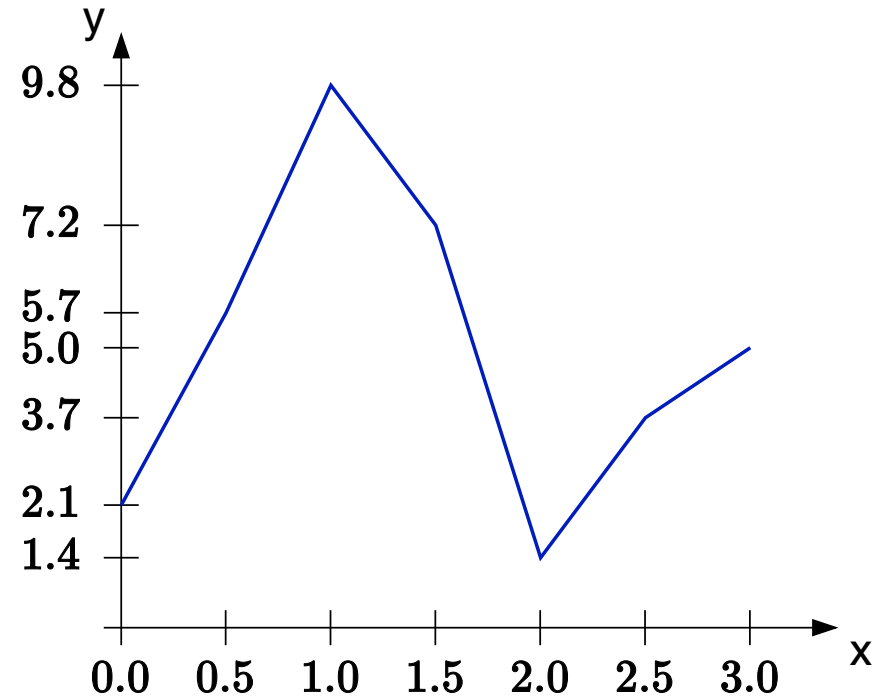
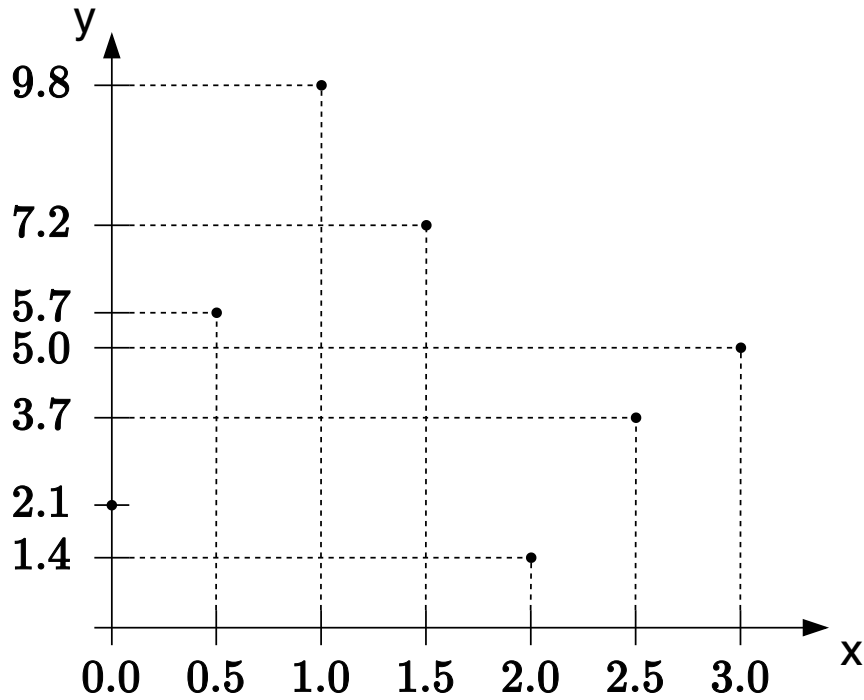
```
>> plot(zeit, messwerte)
```



# Wie funktioniert der `plot` Befehl?

Syntax: `plot(x, y)`

x	0.0	0.5	1.0	1.5	2.0	2.5	3.0
y	2.1	5.7	9.8	7.2	1.4	3.7	5.0



## Beispiele zum `plot` Befehl I

```
>> x = 0:0.1:10;
```

```
>> y1 = sin(x);
```

```
>> y2 = cos(x);
```

### Verschiedene Linienfarben

```
>> plot(x, y1, 'r')
```

```
>> plot(x, y1, 'g')
```

```
>> plot(x, y1, 'b')
```

### Mehrere Plots in einem Bild

```
>> plot(x, y1, x, y2)
```

```
>> plot(x, y1, 'g', x, y2, 'm')
```

## Beispiele zum `plot` Befehl II

### Setzen von Markern

```
>> plot(x, y1, 'x', x, y2, 'o')
```

### Verschiedene Linientypen

```
>> plot(x, y1, '--', x, y2, '-')
```

### Kombination der Linienformate

```
>> plot(x, y1, 'g:s', x, y2, 'b--^')
```

### Hinzufügen von Legendem

```
>> legend('Sinus', 'Cosinus')
```

## Beispiele zum `plot` Befehl III

Mehrere Plots in einem Bild durch mehrfaches Aufrufen von `plot`

Gegenüberstellung verschiedener Kurven

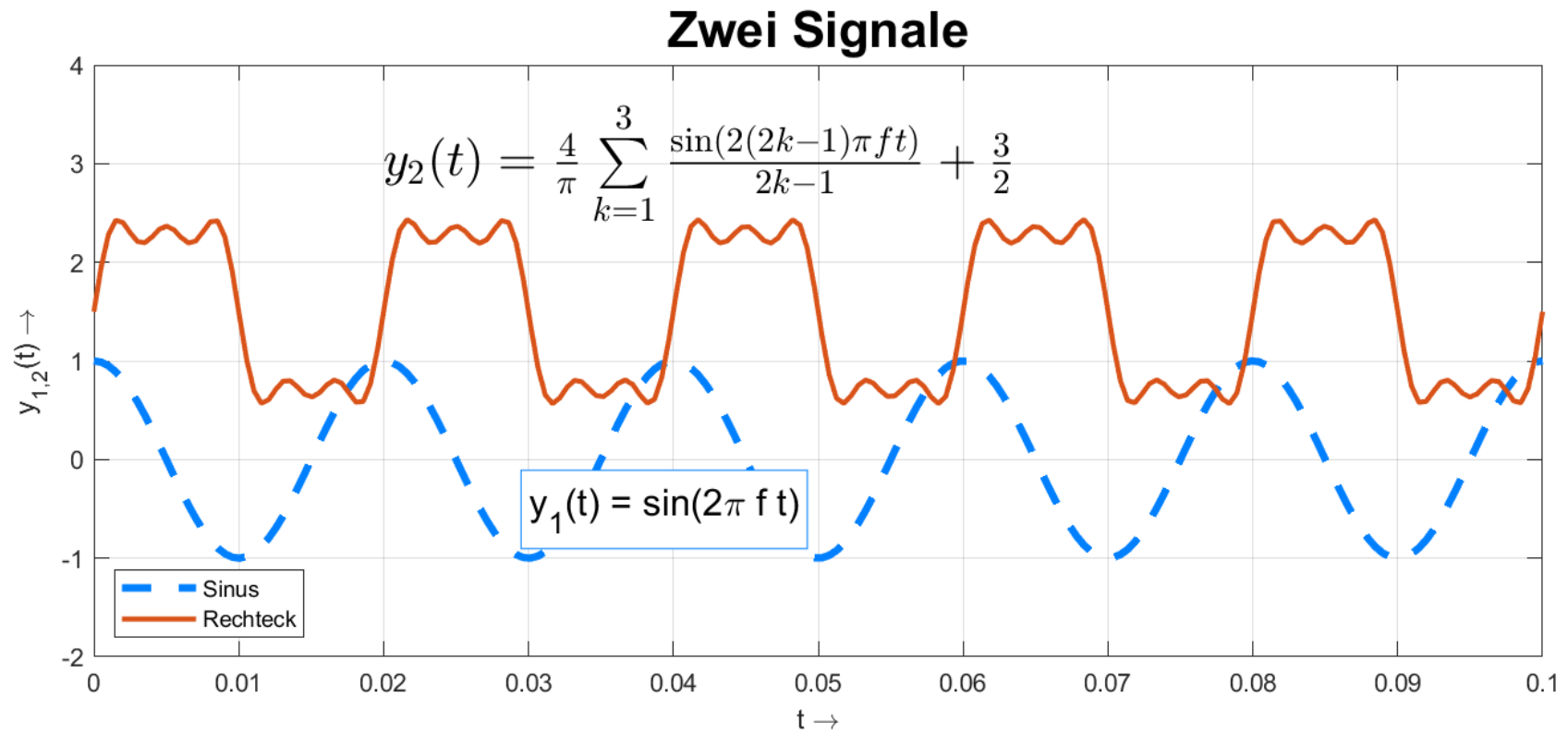
```
>> subplot(121)
```

```
>> plot(x, y1, 'g')
```

```
>> subplot(122)
```

```
>> plot(x, y2, 'b')
```

# Ein Komplexes Beispiel

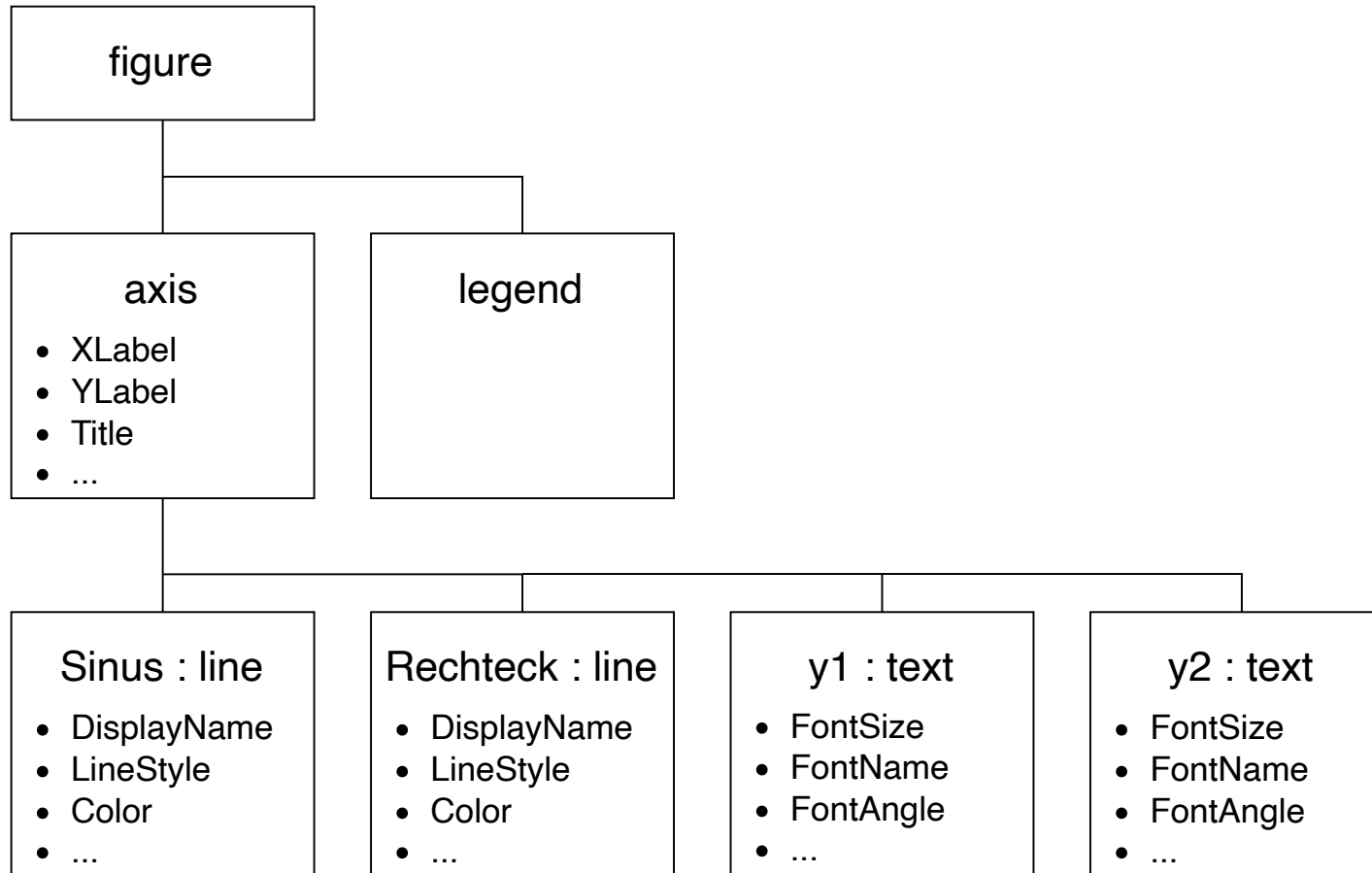


## Zwei Signale: Source Code

```
%%
t = linspace(0, 100e-3, 200);
y1 = cos(2*pi*50*t);
y2 = (sin(2*pi*50*t) + 1/3*sin(2*pi*150*t) + 1/5*sin(2*pi*250*t)) + 3/2;

%%
figure(1), clf
h = plot(t, y1, '--');
set(h, 'LineWidth', 3)
set(h, 'Color', [0, 0.5, 1])
hold on
plot(t, y2, 'LineWidth', 2)
ylim([-2, 4])
set(gcf, 'Position', [900, 500, 1000, 400])
grid on
h = title('Zwei Signale');
set(h, 'FontSize', 20)
xlabel('t \rightarrow')
ylabel('y_{1,2}(t) \rightarrow')
set(gca, 'YTick', -2:2:4)
legend('Sinus', 'Rechteck', 'Location', 'SW')
h = text(0.03, -0.5, 'y_1(t) = sin(2\pi f t)');
set(h, 'FontSize', 15)
set(h, 'BackgroundColor', 'w')
set(h, 'EdgeColor', [0, 0.5, 1])
text(0.02, 3, '$y_2(t) = \frac{4}{\pi} \sum \limits_{k=1}^3 \frac{\sin(2(2k-1)\pi f t)}{2k-1} + \frac{3}{2}$', ...
     'Interpreter', 'Latex', ...
     'FontSize', 20)
saveas(gcf, 'two_signals.png')
```

# Objektstruktur



## 3-Dimensionale Diagramme

### 3-D Linienplot: `plot3(x, y, z)`

- Erweiterung des 2-Dimensionalen Plots auf eine dritte Dimension
- Darstellung und Verbinden der Punkte-Tripel  $(x_i, y_i, z_i)$

### 3-D Mesh-Plot:

- Darstellung der Funktion  $z = f(x, y)$
- Verschiedene Darstellungsmöglichkeiten
  - Mesh-Plot: `mesh(X, Y)`
  - Surface-Plot: `surf(X, Y)`
  - Höhenlinien-Plot: `contour(X, Y)`
  - etc.



## 3D-Linienplot

$$\varphi = 0 \dots 4\pi \quad x = \cos(\varphi) \quad y = \sin(\varphi) \quad z = \varphi$$

```
>> phi = linspace(0, 4*pi, 1000);
```

```
>> x = cos(phi);
```

```
>> y = sin(phi);
```

```
>> z = phi;
```

```
>> plot3(x, y, z)
```

```
>> grid on
```

```
>> xlabel('x\rightarrow')
```

```
>> ylabel('y\rightarrow')
```

```
>> zlabel('z\rightarrow')
```

## 3D-Mesh-Plot I

**Aufgabe: Plotten sie die Funktion**

$$f(x, y) = \frac{\cos(x) \cdot \sin(y)}{x^2 + y^2 + 1} \quad x = -10 \dots 10 \quad y = -5 \dots 5$$

**Vorbereitung: Anlegen zweier Vektoren für  $x$  (Länge  $N$ ) und  $y$  (Länge  $M$ )**

```
>> x = linspace(-10, 10, 100);
```

```
>> y = linspace(-5, 5, 50);
```

**Anordnung der Funktionswerte als Matrix über alle Kombinationen der Werte von  $x$  und  $y$**

$$\mathbf{Z} = \begin{bmatrix} f(x_1, y_1) & f(x_2, y_1) & f(x_3, y_1) & \cdots & f(x_N, y_1) \\ f(x_1, y_2) & f(x_2, y_2) & f(x_3, y_2) & \cdots & f(x_N, y_2) \\ f(x_1, y_3) & f(x_2, y_3) & f(x_3, y_3) & \cdots & f(x_N, y_3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f(x_1, y_M) & f(x_2, y_M) & f(x_3, y_M) & \cdots & f(x_N, y_M) \end{bmatrix}$$

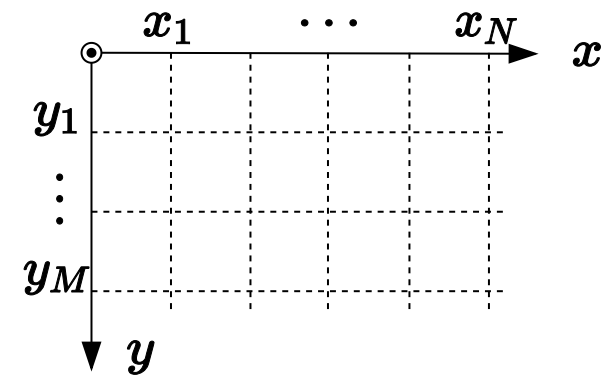
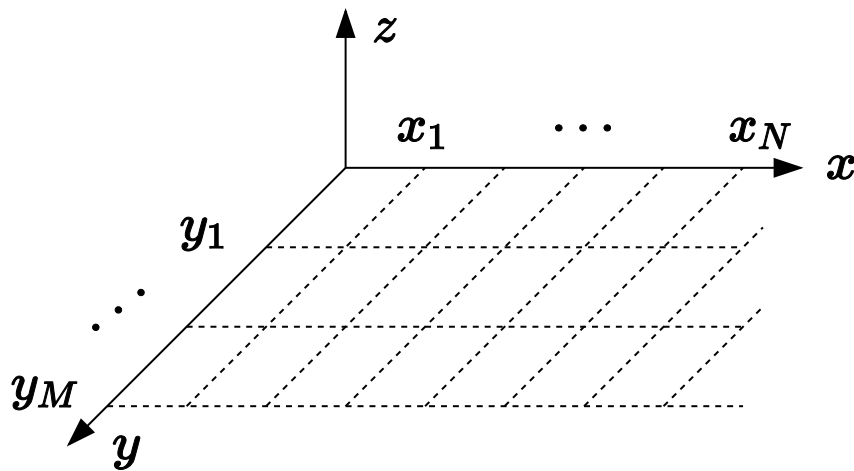
## 3D-Mesh-Plot II

### Berechnung der Matrix der Funktionswerte

$$\mathbf{Z} = f(\mathbf{X}, \mathbf{Y}) \quad \mathbf{X} = \begin{bmatrix} x_1 & \cdots & x_N \\ \vdots & \ddots & \vdots \\ x_1 & \cdots & x_N \end{bmatrix} \quad \mathbf{Y} = \begin{bmatrix} y_1 & \cdots & y_1 \\ \vdots & \ddots & \vdots \\ y_M & \cdots & y_M \end{bmatrix} \quad \mathbf{X}, \mathbf{Y}, \mathbf{Z} \in \mathbb{R}^{M \times N}$$

### Matrizen der Stützstellen

```
>> [X, Y] = meshgrid(x, y);
```



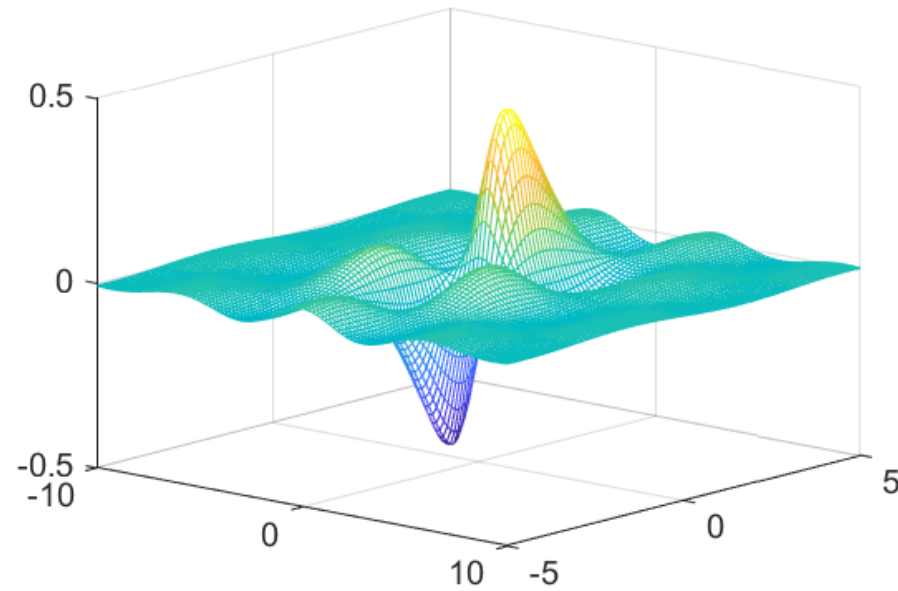
## 3D-Mesh-Plot III

### Berechnen der Funktionswerte

```
>> Z = cos(X) .* sin(Y) ./ (X.^2 + Y.^2 + 1);
```

### Darstellung der Funktion

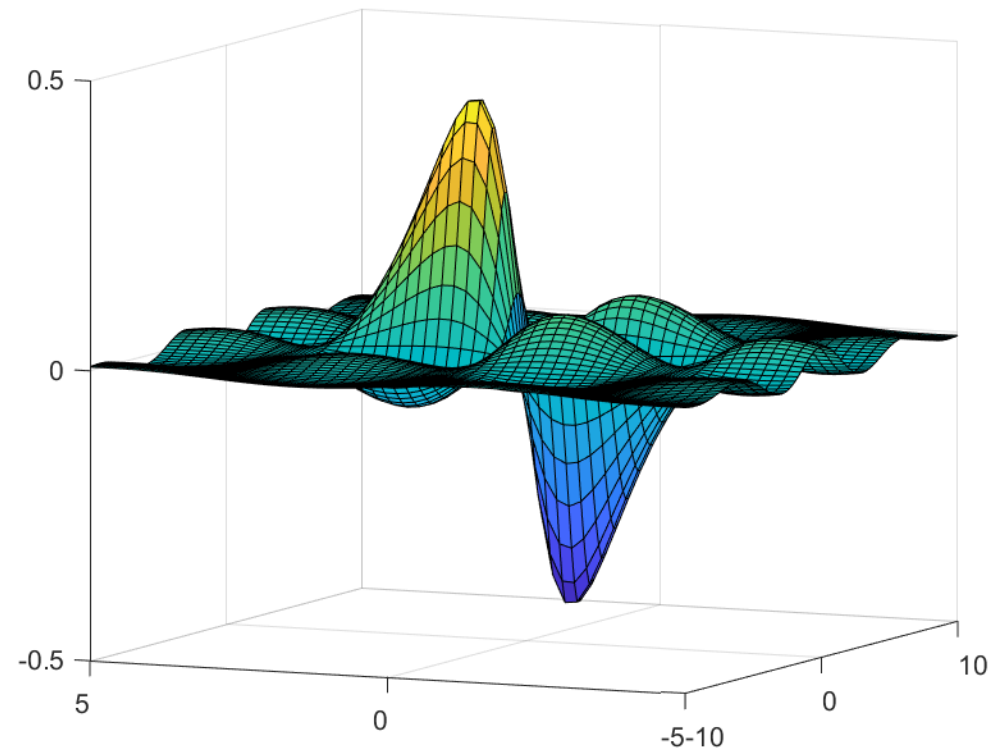
```
>> mesh(x, y, Z)
```



## 3D-Mesh-Plot IV

### Alternative Darstellung als Surface-Plot

```
>> surf(x, y, Z)
```

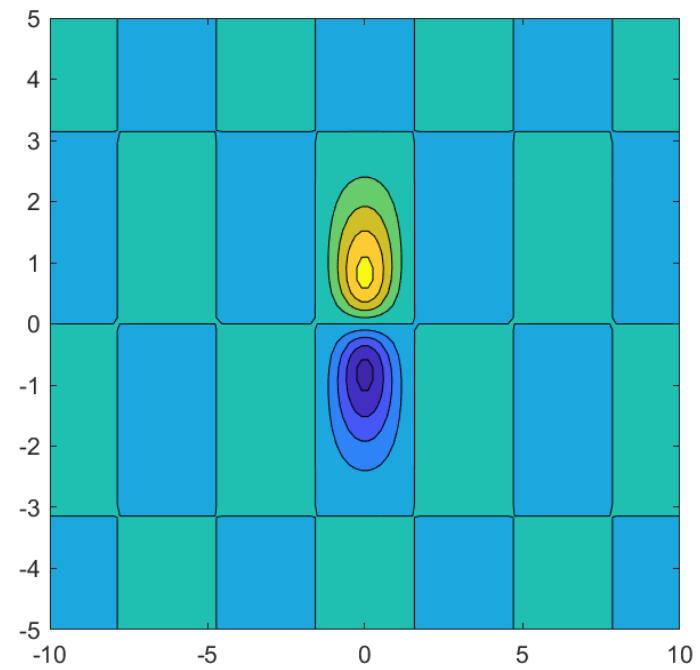
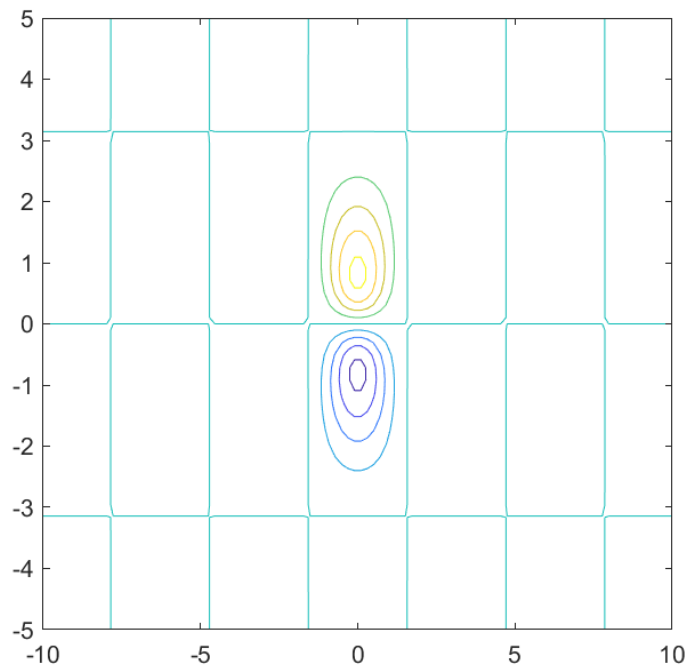


## 3D-Mesh-Plot V

### Erstellen von Contour-Plots

```
>> contour(x, y, Z)
```

```
>> contourf(x, y, Z)
```

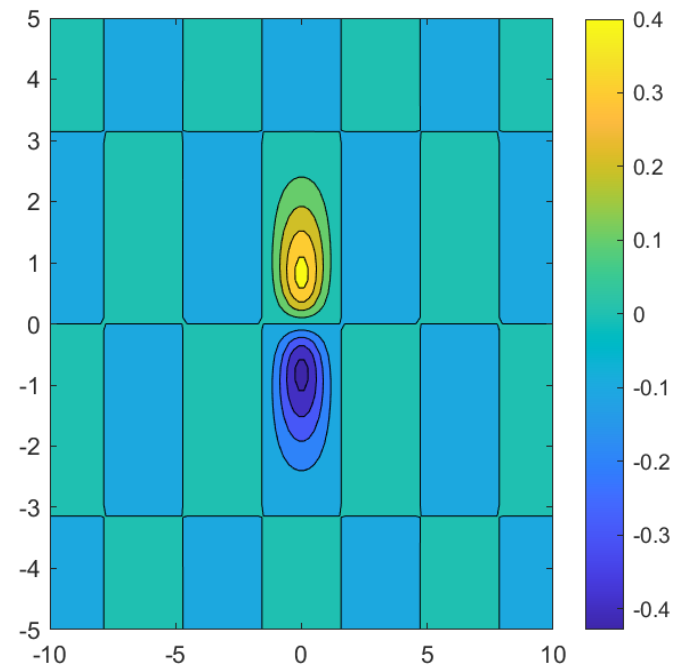
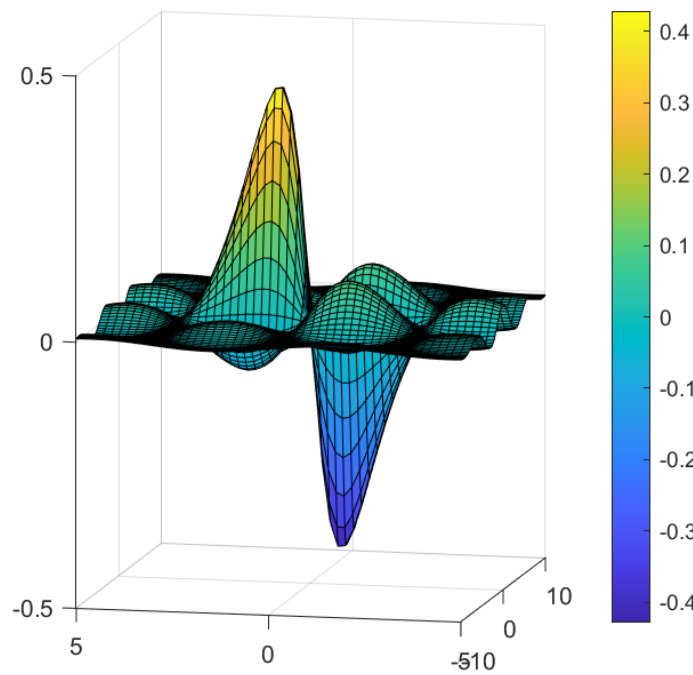


## 3D-Mesh-Plot VI

### Hinzufügen einer Color-Bar

```
>> surf(x, y, Z); colorbar
```

```
>> contourf(x, y, Z); colorbar
```



## Grafische Visualisierung in Matlab

- Verwenden des `plot`-Befehls
- Platzieren von mehreren Plots in einem Figure
- Gestalten der Plots (Achsenbeschriftung, Legenden, Texte, etc.)
- Verwenden der Properties einzelner Objekte (Linientypen und -farben, Marker, Schriftarten, Latex, etc.)
- Klassenstruktur der Plots (Figure, Axes, Lineplot, etc.)
- Visualisierung von 2D-Funktionen  $y = f(x)$
- Verwenden weiterer 2D-Diagrammtypen (Balkendiagramm, stem, Histogramm, etc.)
- Erstellen von 3D-Plots (Linienplot und Mesh-Plot)