

Hyperebenen-Animation

Nach 3-5 Minuten entsteht eine Datei "hyperplane_animation.gif im Download Verzeichnis des Rechners. Sie zeigt eine animierte Darstellung der sich um die z-Achse drehenden Hyperebenen.

Die gif-Datei ist fertig, falls die aktuelle Zeit am Ende angezeigt wird.

Dr. Hermann Völlinger, DHBW Stuttgart, September 2024

In [32]:

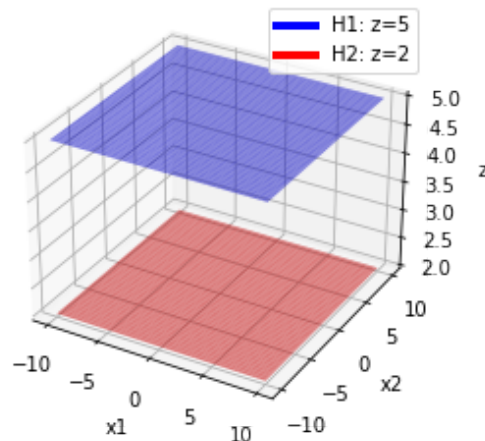
```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from mpl_toolkits.mplot3d import Axes3D
4 from matplotlib.animation import FuncAnimation
5 import os
6
7 # Import Library time to check execution date+time
8 import time
9 # print the date & time of the notebook
10 print('*****')
11 print("Actual date & time of the notebook:",time.strftime("%d.%m.%Y %H:%M"))
12 print('*****')
13
```

```
*****
Actual date & time of the notebook: 11.09.2024 18:05:04
*****
```

In [33]:

```
1 # Definition der Hyperebenen
2 def hyperplane1(x1, x2):
3     return 5 * np.ones_like(x1)
4
5 def hyperplane2(x1, x2):
6     return 2 * np.ones_like(x1)
7
8 # Erzeugung der Datenpunkte für den Plot
9 x1 = np.linspace(-10, 10, 100)
10 x2 = np.linspace(-10, 10, 100)
11 x1, x2 = np.meshgrid(x1, x2)
12 z1 = hyperplane1(x1, x2)
13 z2 = hyperplane2(x1, x2)
14
15 # Erstellen des 3D-Plots
16 fig = plt.figure()
17 ax = fig.add_subplot(111, projection='3d')
18
19 # Plotten der Hyperebenen
20 blue_surface = ax.plot_surface(x1, x2, z1, color='b', alpha=0.5)
21 red_surface = ax.plot_surface(x1, x2, z2, color='r', alpha=0.5)
22
23 # Achsenbeschriftung
24 ax.set_xlabel('x1')
25 ax.set_ylabel('x2')
26 ax.set_zlabel('z')
27
28 # Manuelle Legende
29 blue_patch = plt.Line2D([0], [0], color='b', lw=4, label='H1: z=5')
30 red_patch = plt.Line2D([0], [0], color='r', lw=4, label='H2: z=2')
31 ax.legend(handles=[blue_patch, red_patch])
32 import numpy as np
33 import matplotlib.pyplot as plt
34 from mpl_toolkits.mplot3d import Axes3D
35
36 plt.title('Rote Hyperenbene H1={x1,x2,2} und blaue Hyperebne H2={x1,x2,5}')
37
38 # Anzeigen des Plots
39 plt.show()
40
```

Rote Hyperenbene H1={x1,x2,2} und blaue Hyperebne H2={x1,x2,5}



```

In [34]: 1 # Funktion zur Aktualisierung des Blickwinkels (für die Animation)
2 def update(frame):
3     ax.view_init(elev=30, azim=frame)
4
5 # Erstellen der Animation
6 ani = FuncAnimation(fig, update, frames=np.arange(0, 360, 2), interval=10
7
8 # Pfad zum Downloads-Verzeichnis
9 download_dir = os.path.join(os.path.expanduser("~"), "Downloads", "hyperp
10
11 # Speichern der Animation als GIF
12 ani.save(download_dir, writer='imagemagick', fps=30)
13
14 # Anzeigen der Animation (optional)
15 # nach 3-5 Minuten entsteht eine Datei "hyperplane_animation.gif im Downl
16 # Sie zeigt eine animierte Darstellung der sich um die z-Achse drehenden
17 # Die gif-Datei ist fertig, falls die aktuelle Zeit unten angezeigt wird
18
19 #plt.title('Animation: Drehung der Hyperebenen um die Z-Achse')
20 plt.show()
21
22 # print the date & time of the notebook
23 print('*****')
24 print("Actual date & time of the notebook:",time.strftime("%d.%m.%Y %H:%M
25 print('*****')
26

```

MovieWriter imagemagick unavailable; using Pillow instead.

```

*****
Actual date & time of the notebook: 11.09.2024 18:06:52
*****

```