2.3.12

Anshu kumar ram - EE25BTECH11009

3rd September 2025

Question

Find the angle which the line

$$\frac{x}{1} = \frac{y}{-1} = \frac{z}{0}$$

makes with the positive direction of the Y-axis.

Solution Step 1

The line can be written as

$$L : \mathbf{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} + k \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$$

So, the direction vector is

$$\mathbf{v} = \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}, \quad \mathbf{e_2} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Solution Step 2

Compute dot product:

$$\mathbf{v}^T\mathbf{e_2} = \begin{pmatrix} 1 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = -1 \tag{1}$$

Norms:

$$\|\mathbf{v}\| = \sqrt{1^2 + (-1)^2 + 0^2} = \sqrt{2} \tag{2}$$

$$\|\mathbf{e_2}\| = 1 \tag{3}$$

Solution Step 3

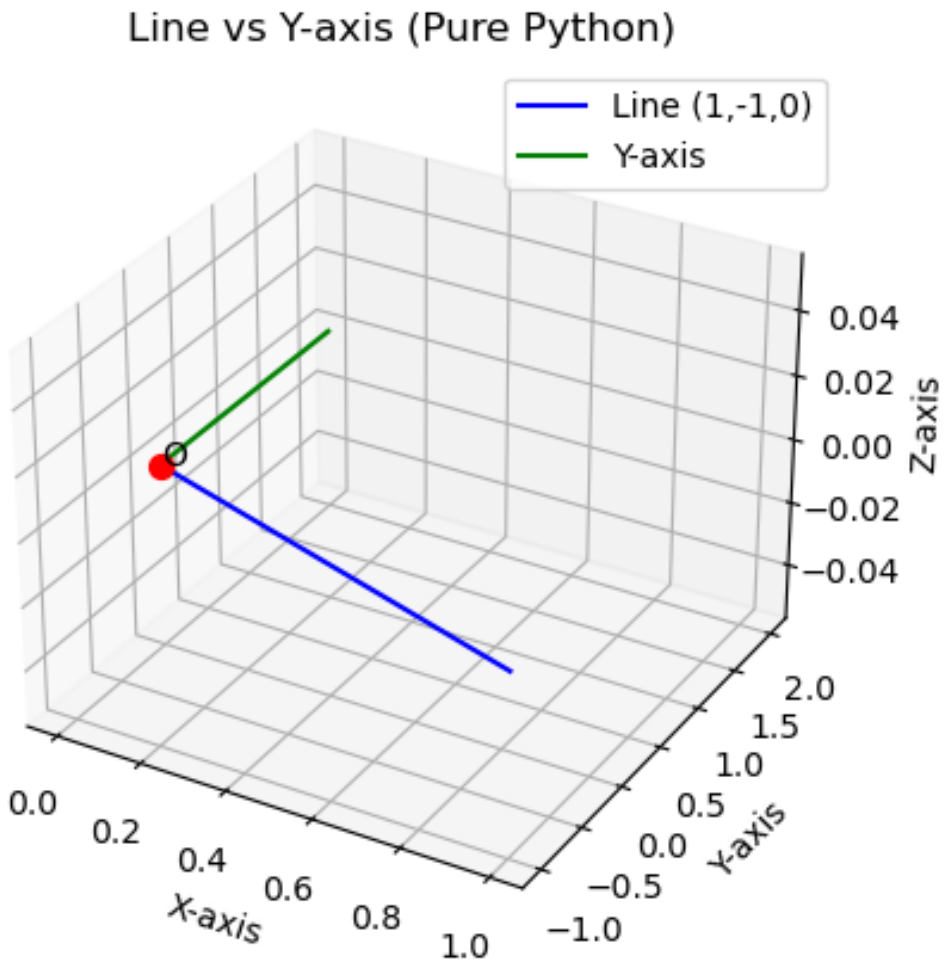
Angle:

$$\cos\theta = \frac{\mathbf{v}^T\mathbf{e_2}}{\|\mathbf{v}\|\|\mathbf{e_2}\|} = \frac{-1}{\sqrt{2}} \tag{4}$$

$$\theta = \cos^{-1}\left(-\frac{1}{\sqrt{2}}\right) = 135^\circ \tag{5}$$

$$\boxed{\theta = 135^\circ}$$

Plot


Line vs Y-axis (Pure Python)
Line (1,-1,0)
Y-axis
0.04
0.02
0.00
−0.02
−0.04
Z-axis
0.0
0.2
0.4
0.6
0.8
1.0
X-axis
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
Y-axis

C Code (angle_between)

```c
#include <stdio.h>
#include <math.h>

// Function to compute angle between two 3D vectors
double angle_between(double *u, double *v) {
    double dot = u[0]*v[0] + u[1]*v[1] + u[2]*v[2];
    double norm_u = sqrt(u[0]*u[0] + u[1]*u[1] + u[2]*u[2]);
    double norm_v = sqrt(v[0]*v[0] + v[1]*v[1] + v[2]*v[2]);
    double cos_theta = dot / (norm_u * norm_v);
    if(cos_theta > 1.0) cos_theta = 1.0;
    if(cos_theta < -1.0) cos_theta = -1.0;
    return acos(cos_theta);
}
```

Python + C (Part 1)

```
import ctypes
import numpy as np
import matplotlib.pyplot as plt

handc = ctypes.CDLL("./func.so")
handc.angle_between.argtypes = [
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double)
]
handc.angle_between.restype = ctypes.c_double

def np_to_c(arr):
    return arr.ctypes.data_as(ctypes.POINTER(ctypes.c_double))

v = np.array([1.0,-1.0,0.0], dtype=np.float64)
e2 = np.array([0.0,1.0,0.0], dtype=np.float64)

theta = handc.angle_between(np_to_c(v), np_to_c(e2))
theta_deg = np.degrees(theta)
```

Python + C (Part 2)

```
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

t = np.linspace(-2,2,10)
x_line, y_line, z_line = t*v[0], t*v[1], t*v[2]
ax.plot(x_line, y_line, z_line, label='Line (1,-1,0)')

ax.plot([0,0],[0,2],[0,0],'g',label='Y-axis')

ax.set_xlabel('X'); ax.set_ylabel('Y'); ax.set_zlabel('Z')
ax.legend()
ax.set_title(f"Angle  {theta_deg:.1f}")
plt.savefig("../figs/line_c.png")
plt.show()
```

Pure Python (Part 1)

```python
import sys
sys.path.insert(0, '/home/anshu-ram/matgeo/codes/CoordGeo')

import numpy as np
import matplotlib.pyplot as plt
from line.funcs import line_gen

v = np.array([1,-1,0]).reshape(-1,1)
e2 = np.array([0,1,0]).reshape(-1,1)

dot = float(v.T @ e2)
theta_rad = np.arccos(dot/(np.linalg.norm(v)*np.linalg.norm(e2)))
theta_deg = np.degrees(theta_rad)
print(f"Angle = {theta_deg:.2f} degrees")
```

Pure Python (Part 2)

```
A = np.zeros((3,1))
B = v
line_points = line_gen(A, B)

Y_end = np.array([0,2,0]).reshape(-1,1)
yaxis_points = line_gen(A, Y_end)

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot(line_points[0,:], line_points[1,:], line_points[2,:],
    label="Line")
ax.plot(yaxis_points[0,:], yaxis_points[1,:], yaxis_points[2,:],
    label="Y-axis")

ax.scatter(0,0,0,color="red",s=50)
ax.text(0,0,0,"O",fontsize=10)
ax.set_xlabel("X"); ax.set_ylabel("Y"); ax.set_zlabel("Z")
ax.legend(); plt.title("Line vs Y-axis")
plt.savefig("../figs/line.png"); plt.show()
```