4.3.35

Navya Priya - EE25BTECH11045

October 1,2025

Question

Find the intercepts made by the plane $2x - 3y + 5z + 4 = 0$ on the co-ordinate axis

Theoretical Solution

The above equation of plane can be written as

$$\mathbf{n}^\top \mathbf{x} = c$$

where

$$\mathbf{n} = \begin{pmatrix} 2 \\ -3 \\ 5 \end{pmatrix} \tag{1}$$

$$c = -4 \tag{2}$$

X-intercept

Let the x-intercept of the given plane be of the form $\begin{pmatrix} a \\ 0 \\ 0 \end{pmatrix}$. Substituting this in the above equation gives

$$\begin{pmatrix} 2 \\ -3 \\ 5 \end{pmatrix}^{\top} \begin{pmatrix} a \\ 0 \\ 0 \end{pmatrix} = -4 \tag{3}$$

$$a = -2 \tag{4}$$

$$\therefore \text{ The } x-\text{intercept is } \begin{pmatrix} -2 \\ 0 \\ 0 \end{pmatrix}$$

Y-intercept

Let the y-intercept of the given plane be of the form $\begin{pmatrix} 0 \\ b \\ 0 \end{pmatrix}$. Substituting this in the above equation gives

$$\begin{pmatrix} 2 \\ -3 \\ 5 \end{pmatrix}^{\top} \begin{pmatrix} 0 \\ b \\ 0 \end{pmatrix} = -4 \tag{5}$$

$$\mathrm{b} = \frac{4}{3} \tag{6}$$

$$\therefore \text{ The } y - \text{intercept is } \begin{pmatrix} 0 \\ \frac{4}{3} \\ 0 \end{pmatrix}$$

Z-intercept

Let the z-intercept of the given plane be of the form $\begin{pmatrix} 0 \\ 0 \\ c \end{pmatrix}$. Substituting this in the above equation gives

$$\begin{pmatrix} 2 \\ -3 \\ 5 \end{pmatrix}^{\top} \begin{pmatrix} 0 \\ 0 \\ c \end{pmatrix} = -4 \tag{7}$$

$$c = \frac{-4}{5} \tag{8}$$

$$\therefore \text{ The } z - \text{intercept is } \begin{pmatrix} 0 \\ 0 \\ \frac{-4}{5} \end{pmatrix}$$

C code

```c
#include <stdio.h>

void find_intercepts(double *x_int, double *y_int, double *z_int,
    double a,double b,double c,double d) {
    // Plane equation: ax + by + cz = d
    // X-intercept -> y=0, z=0 -> ax=d
    *x_int = d.0 / a.0;

    // Y-intercept -> x=0, z=0 -> by-d=0
    *y_int = d.0 / b.0;

    // Z-intercept -> x=0, y=0 -> cz-d=0
    *z_int = d.0 / c.0;
}
```

CallC.py

```
    import ctypes
# Load the shared library (change to .dll if using Windows)
lib = ctypes.CDLL("./plane_intercepts.so")
# Define function prototype
lib.find_intercepts.argtypes = [ctypes.POINTER(ctypes.c_double),
                                ctypes.POINTER(ctypes.c_double),
                                ctypes.POINTER(ctypes.c_double)]
# Prepare variables
x = ctypes.c_double()
y = ctypes.c_double()
z = ctypes.c_double()
# Call C function
lib.find_intercepts(ctypes.byref(x), ctypes.byref(y), ctypes.
    byref(z))
# Print results
print("X-intercept:", (x.value, 0, 0))
print("Y-intercept:", (0, y.value, 0))
print("Z-intercept:", (0, 0, z.value))
```

Plot.py

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

# Plane equation: 2x - 3y + 5z + 4 = 0
x_int = -4/2 # X-intercept
y_int = 4/3 # Y-intercept
z_int = -4/5 # Z-intercept

# Create 3D plot
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# Meshgrid for plane
xx, yy = np.meshgrid(np.linspace(-4, 4, 50), np.linspace(-4, 4,
    50))
zz = (-2*xx + 3*yy - 4) / 5

# Plot plane
ax.plot_surface(xx, yy, zz, alpha=0.5, color='cyan')
```

Plot.py

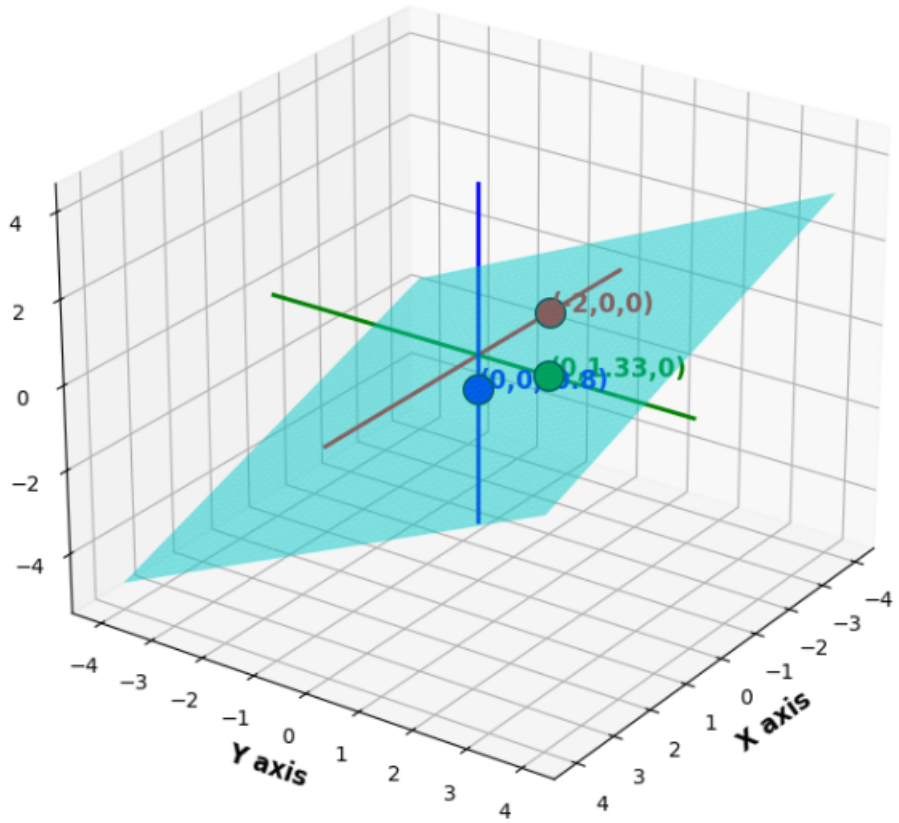
```
ax.plot([0, 0], [-4, 4], [0, 0], color='green', linewidth=2) # Y-
    axis
ax.plot([0, 0], [0, 0], [-4, 4], color='blue', linewidth=2) # Z-
    axis

# Plot intercepts with big markers
ax.scatter(x_int, 0, 0, color='red', s=200, marker='o', edgecolor
    ='k', zorder=5)
ax.scatter(0, y_int, 0, color='green', s=200, marker='o',
    edgecolor='k', zorder=5)
ax.scatter(0, 0, z_int, color='blue', s=200, marker='o',
    edgecolor='k', zorder=5)

# Add labels for intercepts
ax.text(x_int, 0, 0, f'(-2,0,0)', color='red', fontsize=12,
    weight='bold')
ax.text(0, y_int, 0, f'(0,1.33,0)', color='green', fontsize=12,
    weight='bold')
```

Plot.py

```
ax.text(0, 0, z_int, f'(0,0,-0.8)', color='blue', fontsize=12,
    weight='bold')

# Axes labels
ax.set_xlabel('X axis', fontsize=12, weight='bold')
ax.set_ylabel('Y axis', fontsize=12, weight='bold')
ax.set_zlabel('Z axis', fontsize=12, weight='bold')
ax.set_title('Plane 2x - 3y + 5z + 4 = 0 with Intercepts',
    fontsize=14, weight='bold')

# Equal aspect ratio
ax.set_box_aspect([1,1,0.8])

# Adjust view
ax.view_init(elev=25, azim=35)

plt.show()
```

Plot

From the graph, theoretical solution matches with the computational solution.



plane $2x - 3y + 5z + 4 = 0$ with intercepts