4.11.36

Vaishnavi - EE25BTECH11059

October 11, 2025

Question

Find the coordinates of the point where the line through the points $(3, -4, -5)$ and $(2, -3, 1)$ crosses the plane determined by the points $(1, 2, 3)$, $(4, 2, -3)$ and $(0, 4, 3)$.

Solution

Variable	Value
A	$(3, -4, -5)$
B	$(2, -3, 1)$
P	$(1, 2, 3)$
Q	$(4, 2, -3)$
R	$(0, 4, 3)$

Table: Variables Used

Solution

Let eq of plane be

$$\mathbf{n}^{\mathbf{T}}\mathbf{x} = 1 \tag{1}$$

As $\mathbf{P}, \mathbf{Q}, \mathbf{R}$ lie on the plane

$$\mathbf{n}^{\mathbf{T}}\mathbf{P} = 1 \tag{2}$$

$$\mathbf{n}^{\mathbf{T}}\mathbf{Q} = 1 \tag{3}$$

$$\mathbf{n}^{\mathbf{T}}\mathbf{R} = 1 \tag{4}$$

$$\begin{pmatrix} P^T \\ Q^T \\ R^T \end{pmatrix} \mathbf{n} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \tag{5}$$

solution

From eq $(2),(3),(4) and (5)$

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 4 & 2 & -3 & 1 \\ 0 & 4 & 3 & 1 \end{array}\right) \xrightarrow{R_2 \to R_2 - 4R_1} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & -6 & -15 & -3 \\ 0 & 4 & 3 & 1 \end{array}\right) \tag{6}$$

$$\xrightarrow{R_2 \to -\frac{1}{3}R_2} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 2 & 5 & 1 \\ 0 & 4 & 3 & 1 \end{array}\right) \tag{7}$$

Solutions

$$\xrightarrow{R_3 \to R_3 - 2R_2} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 2 & 5 & 1 \\ 0 & 0 & -7 & -1 \end{array}\right) \tag{8}$$

$$\xrightarrow{R_3 \to -\frac{1}{7}R_3} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 2 & 5 & 1 \\ 0 & 0 & 1 & \frac{1}{7} \end{array}\right) \tag{9}$$

$$\xrightarrow{R_2 \to R_2 - 5R_3} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 2 & 0 & \frac{2}{7} \\ 0 & 0 & 1 & \frac{1}{7} \end{array}\right) \tag{10}$$

Solutions

$$\xrightarrow{R_2 \to \frac{1}{2} R_2} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & 0 & \frac{1}{7} \\ 0 & 0 & 1 & \frac{1}{7} \end{array}\right) \tag{11}$$

$$\xrightarrow{R_1 \to R_1 - 3R_3} \left(\begin{array}{ccc|c} 1 & 2 & 0 & \frac{4}{7} \\ 0 & 1 & 0 & \frac{1}{7} \\ 0 & 0 & 1 & \frac{1}{7} \end{array}\right) \tag{12}$$

$$\xrightarrow{R_1 \to R_1 - 2R_2} \left(\begin{array}{ccc|c} 1 & 0 & 0 & \frac{2}{7} \\ 0 & 1 & 0 & \frac{1}{7} \\ 0 & 0 & 1 & \frac{1}{7} \end{array}\right) \tag{13}$$

Solution

$$\mathbf{n} = \begin{pmatrix} \frac{2}{7} \\ \frac{1}{7} \\ \frac{1}{7} \end{pmatrix} \tag{14}$$

hence eq of plane is

$$\begin{pmatrix} \frac{2}{7} & \frac{1}{7} & \frac{1}{7} \end{pmatrix} \mathbf{x} = 1 \tag{15}$$

let a point on line **AB** be

$$\mathbf{c} = k\mathbf{A} + (1-k)\mathbf{B} \tag{16}$$

$$\mathbf{n}^{\mathbf{T}}(k\mathbf{A} + (1-k)\mathbf{B}) = 1 \tag{17}$$

$$\begin{pmatrix} \frac{2}{7} & \frac{1}{7} & \frac{1}{7} \end{pmatrix} \begin{pmatrix} 2+k \\ -3-k \\ 1-6k \end{pmatrix} = 14 + 2k - 3 - k + 1 - 6k = 7 \tag{18}$$

$$2 - 5k = 7 \tag{19}$$

$$k = -1 \tag{20}$$

Solution

The point $\mathbf{c}$ is

$$\mathbf{c} = \begin{pmatrix} 1 \\ -2 \\ 7 \end{pmatrix} \tag{21}$$

Refer to Figure

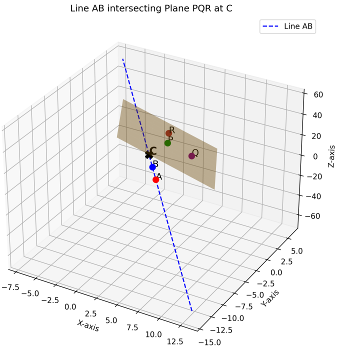

Figure:

Python Code

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Given points
A = np.array([3, -4, -5])
B = np.array([2, -3, 1])
P = np.array([1, 2, 3])
Q = np.array([4, 2, -3])
R = np.array([0, 4, 3])

# Direction vector of line AB
AB = B - A

# Normal vector to plane PQR = (Q-P) x (R-P)
n = np.cross(Q-P, R-P)

# Solve for intersection: A + t*AB lies in plane
t = np.dot(n, P-A) / np.dot(n, AB)
```

```
plane_points = P.reshape(3,1,1) + (Q-P).reshape(3,1,1)
    *U + (R-P).reshape(3,1,1)*V

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

# Extended line AB
t_vals = np.linspace(-10, 10, 200)
line_points = A.reshape(3,1) + np.outer(AB, t_vals)
ax.plot(line_points[0], line_points[1], line_points
    [2], 'b--', label= Line AB )

# Plot plane
ax.plot_surface(plane_points[0], plane_points[1],
    plane_points[2], alpha=0.4, color='orange')

# Plot points
ax.scatter(*A, color='red', s=60)
```

Python Code

```
ax.scatter(*P, color='green', s=60)
ax.text(*P,  P , fontsize=12)

ax.scatter(*Q, color='purple', s=60)
ax.text(*Q,  Q , fontsize=12)

ax.scatter(*R, color='brown', s=60)
ax.text(*R,  R , fontsize=12)

ax.scatter(*C, color='black', s=100, marker='X')
ax.text(*C,  C , fontsize=14, color='black', weight='
    bold')

# Labels
ax.set_xlabel( X-axis )
ax.set_ylabel( Y-axis )
ax.set_zlabel( Z-axis )
ax.set_title( Line AB intersecting Plane PQR at C )
```

C Code

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

#define N 3

// Gaussian elimination solver
void gaussElimination(double A[N][N], double b[N],
    double x[N]) {
    int i, j, k;
    double ratio;

    // Forward elimination
    for (i = 0; i < N - 1; i++) {
        for (j = i + 1; j < N; j++) {
            if (fabs(A[i][i]) < 1e-12) return;
```

C Code

```c
    ratio = A[j][i] / A[i][i];
            for (k = 0; k < N; k++) {
                A[j][k] -= ratio * A[i][k];
            }
            b[j] -= ratio * b[i];
        }
    }
    // Back substitution
    for (i = N - 1; i >= 0; i--) {
        x[i] = b[i];
        for (j = i + 1; j < N; j++) {
            x[i] -= A[i][j] * x[j];
        }
        x[i] /= A[i][i];
    }
}
```

C Code

```c
// Exposed function for Python
void solve_plane(double *out) {
    double A[N][N] = {
        {1, 2, 3},
        {4, 2, -3},
        {0, 4, 3}
    };
    double b[N] = {1, 1, 1};
    double n[N];

    gaussElimination(A, b, n);

    for (int i = 0; i < N; i++) {
        out[i] = n[i];
    }
}
```

Python and C Code

```
import ctypes
import numpy as np

# Load shared library
lib = ctypes.CDLL( ./code.so )

# Define function signature: void solve_plane(double *
    out)
lib.solve_plane.argtypes = [ctypes.POINTER(ctypes.
    c_double)]
lib.solve_plane.restype = None

# Prepare output array
out = (ctypes.c_double * 3)()
lib.solve_plane(out)

# Convert to numpy for convenience
normal_vec = np.array([out[i] for i in range(3)])
print( Normal vector: , normal_vec)
```