Matgeo Presentation - Bonus Problem

ee25btech11063 - Vejith

September 11, 2025

Question

Given 3 vectors $\mathbf{A}$,$\mathbf{B}$,$\mathbf{C}$ are coplanar then show $\det(\mathbf{M}) = 0$ where $\mathbf{M} = (\mathbf{A}\ \mathbf{B}\ \mathbf{C})$

Solution

Equation of plane through 3 coplanar points is

$$\mathbf{n}^T\mathbf{x} = 0 \tag{0.1}$$

$$\implies \mathbf{n}^T\mathbf{A} = \mathbf{n}^T\mathbf{B} = \mathbf{n}^T\mathbf{C} = 0 \tag{0.2}$$

$$\mathbf{M} = \begin{pmatrix}\mathbf{A} & \mathbf{B} & \mathbf{C}\end{pmatrix} \tag{0.3}$$

$$\implies \mathbf{n}^T\mathbf{M} = \begin{pmatrix}\mathbf{n}^T\mathbf{A} & \mathbf{n}^T\mathbf{B} & \mathbf{n}^T\mathbf{C}\end{pmatrix} \tag{0.4}$$

$$\implies \mathbf{n}^T\mathbf{M} = \begin{pmatrix}0 & 0 & 0\end{pmatrix} \tag{0.5}$$

$$\implies \mathbf{n}^T\mathbf{M} = \mathbf{0} \tag{0.6}$$

From (0.6) it means $\mathbf{M}$ has a non trivial vector in $it's$ null space

$$\implies rank(\mathbf{M}) < 3. \tag{0.7}$$

For a $3\times$ 3 square matrix like $\mathbf{M}$ if $\det(\mathbf{M}) \neq 0$ means $\mathbf{M}$ is invertible which means $\mathbf{M}$ is a full rank matrix
$\implies$ rank($\mathbf{M}$)=3.(if $\det(\mathbf{M}) \neq 0$)

Solution

From (0.7) rank($\mathbf{M}$)$<$3
$\implies$ $\mathbf{M}$ is not invertible
$\implies$ det($\mathbf{M}$)=0
proof 2:
3 vectors $\mathbf{A}$,$\mathbf{B}$,$\mathbf{C}$ are coplanar means they are linearly dependent.
let′s assume

$$\mathbf{C} = \alpha\mathbf{A} + \beta\mathbf{B}. \tag{0.8}$$

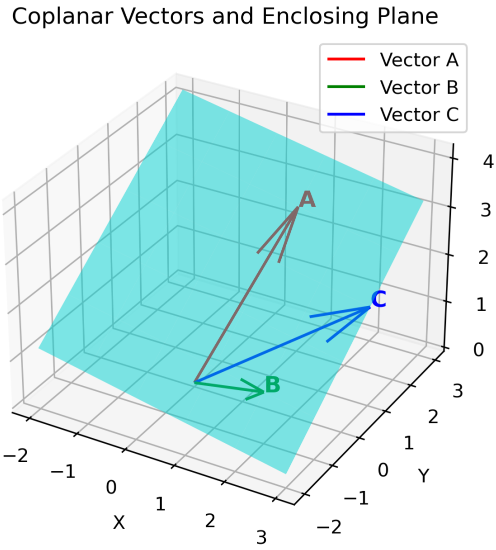
$$\det(\mathbf{M}) = \det((\mathbf{A} \quad \mathbf{B} \quad \mathbf{C}) \tag{0.9}$$

$$= \det((\mathbf{A} \quad \mathbf{B} \quad \alpha\mathbf{A} + \beta\mathbf{B}) \tag{0.10}$$

$$= \alpha\det((\mathbf{A} \quad \mathbf{B} \quad \mathbf{A}) + \beta\det((\mathbf{A} \quad \mathbf{B} \quad \mathbf{B}) = 0 \tag{0.11}$$

$$\implies \det(\mathbf{M}) = 0 \tag{0.12}$$

Plot


Coplanar Vectors and Enclosing Plane
Vector A
Vector B
Vector C
A
B
C
X
Y
−2
−1
0
1
2
3
4

C Code: coplanar.c

```c
#include <stdio.h>

#include <stdio.h>

// Function to compute scalar triple product (box product)
float boxProduct(float A[3], float B[3], float C[3]) {
    return A[0] * (B[1]*C[2] - B[2]*C[1])
         - A[1] * (B[0]*C[2] - B[2]*C[0])
         + A[2] * (B[0]*C[1] - B[1]*C[0]);
}

int main() {
    FILE *fp;
    float A[3], B[3], C[3];
    float box;

    // Input 3 vectors
    printf("Enter vector A (x y z): ");
    scanf("%f %f %f", &A[0], &A[1], &A[2]);

    printf("Enter vector B (x y z): ");
    scanf("%f %f %f", &B[0], &B[1], &B[2]);

    printf("Enter vector C (x y z): ");
    scanf("%f %f %f", &C[0], &C[1], &C[2]);

    // Compute box product
    box = boxProduct(A, B, C);

    // Open file coplanar.dat for writing
    fp = fopen("coplanar.dat", "w");
    if (fp == NULL) {
        printf("Error opening file!\n");
```

C Code: coplanar.c

```
        return 1;
    }

    fprintf(fp, "Scalar␣triple␣product␣(Box␣Product)␣=␣%.2f\n", box);

    if (box == 0)
        fprintf(fp, "Vectors␣are␣coplanar.\n");
    else
        fprintf(fp, "Vectors␣are␣NOT␣coplanar.\n");

    fclose(fp);

    printf("Result␣written␣to␣coplanar.dat\n");

    return 0;
}
```

Python: plot.py

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

# ==== Define the vectors ====
A = np.array([1, 2, 3]) # example vector A
B = np.array([2, -1, 1]) # example vector B
C = np.array([3, 1, 2]) # example vector C

# Compute normal to the plane (B-A)  (C-A)
normal = np.cross(B - A, C - A)

# Equation of plane: n(X - A) = 0  nX = nA
d = np.dot(normal, A)

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

# Plot vectors A,B,C from origin (with labels for legend)
ax.quiver(0, 0, 0, A[0], A[1], A[2], color='r', label='Vector A')
ax.quiver(0, 0, 0, B[0], B[1], B[2], color='g', label='Vector B')
ax.quiver(0, 0, 0, C[0], C[1], C[2], color='b', label='Vector C')

# Add labels at arrow tips
ax.text(A[0], A[1], A[2], "A", color='r', fontsize=12, weight='bold')
ax.text(B[0], B[1], B[2], "B", color='g', fontsize=12, weight='bold')
ax.text(C[0], C[1], C[2], "C", color='b', fontsize=12, weight='bold')

# Create grid for the plane
xx, yy = np.meshgrid(range(-2, 4), range(-2, 4))
zz = (d - normal[0]*xx - normal[1]*yy) / normal[2]
```

Python: plot.py

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

# Labels
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')
ax.set_title("Coplanar Vectors and Enclosing Plane")

# Show legend box
ax.legend()

# ==== Save the figure ====
plt.savefig("coplanar_vectors_plane.png", dpi=300, bbox_inches='tight')

plt.show()
```