Matgeo Presentation - Problem 2.4.16

ee25btech11021 - Dhanush sagar

August 30, 2025

Problem Statement

Given two points

$$\mathbf{A} = \begin{pmatrix} 0 \\ 7 \\ -10 \end{pmatrix}, \qquad \mathbf{B} = \begin{pmatrix} 1 \\ 6 \\ -6 \end{pmatrix}, \qquad \mathbf{C} = \begin{pmatrix} 4 \\ 9 \\ -6 \end{pmatrix}.$$

(a) prove the given points forms isosceles triangle

(b) prove the given points forms right angled triangle

solution

solution : We consider the vectors

$$\mathbf{A} = \begin{pmatrix} 0 \\ 7 \\ -10 \end{pmatrix}, \qquad \mathbf{B} = \begin{pmatrix} 1 \\ 6 \\ -6 \end{pmatrix}, \qquad \mathbf{C} = \begin{pmatrix} 4 \\ 9 \\ -6 \end{pmatrix}.$$

Form the difference vectors $\mathbf{B} - \mathbf{A}$ and $\mathbf{C} - \mathbf{A}$.

$$\mathbf{B} - \mathbf{A} = \begin{pmatrix} 1 - 0 \\ 6 - 7 \\ -6 - (-10) \end{pmatrix} \tag{1.1}$$

$$\mathbf{B} - \mathbf{A} = \begin{pmatrix} 1 \\ -1 \\ 4 \end{pmatrix} \tag{1.2}$$

solution

$$\mathbf{C} - \mathbf{A} = \begin{pmatrix} 4 - 0 \\ 9 - 7 \\ -6 - (-10) \end{pmatrix} \tag{1.3}$$

$$\mathbf{C} - \mathbf{A} = \begin{pmatrix} 4 \\ 2 \\ 4 \end{pmatrix} \tag{1.4}$$

Place these as columns in the 3×2 matrix M.

$$M = \begin{pmatrix} \mathbf{B} - \mathbf{A} & \mathbf{C} - \mathbf{A} \end{pmatrix} \tag{1.5}$$

$$M = \begin{pmatrix} 1 & 4 \\ -1 & 2 \\ 4 & 4 \end{pmatrix} \tag{1.6}$$

solution

Compute the 2×2 minor using rows 1 and 2.

$$\det \begin{pmatrix} 1 & 4 \\ -1 & 2 \end{pmatrix} = 1 \cdot 2 - 4 \cdot (-1) \tag{1.7}$$

$$\det \begin{pmatrix} 1 & 4 \\ -1 & 2 \end{pmatrix} = 2 + 4 = 6 \neq 0 \tag{1.8}$$

Hence $\operatorname{rank}(M) = 2$, so $\mathbf{B} - \mathbf{A}$ and $\mathbf{C} - \mathbf{A}$ are linearly independent. Therefore $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are not collinear and determine a triangle.

$$\mathbf{AB} = \mathbf{B} - \mathbf{A} = \begin{pmatrix} 1 \\ -1 \\ 4 \end{pmatrix}, \tag{2.1}$$

solution A

$$\mathbf{BC} = \mathbf{C} - \mathbf{B} = \begin{pmatrix} 4 - 1 \\ 9 - 6 \\ -6 - (-6) \end{pmatrix} \tag{2.2}$$

$$\mathbf{BC} = \begin{pmatrix} 3 \\ 3 \\ 0 \end{pmatrix}, \tag{2.3}$$

$$\mathbf{CA} = \mathbf{A} - \mathbf{C} = \begin{pmatrix} 0 - 4 \\ 7 - 9 \\ -10 - (-6) \end{pmatrix} \tag{2.4}$$

$$\mathbf{CA} = \begin{pmatrix} -4 \\ -2 \\ -4 \end{pmatrix}. \tag{2.5}$$

solution

$$\|\mathbf{AB}\|^2 = (\mathbf{B} - \mathbf{A})^T(\mathbf{B} - \mathbf{A}) \tag{2.6}$$

$$\|\mathbf{AB}\|^2 = 1^2 + (-1)^2 + 4^2 \tag{2.7}$$

$$\|\mathbf{AB}\|^2 = 18 \tag{2.8}$$

$$\|\mathbf{BC}\|^2 = (\mathbf{C} - \mathbf{B})^T(\mathbf{C} - \mathbf{B}) \tag{2.9}$$

$$\|\mathbf{BC}\|^2 = 3^2 + 3^2 + 0^2 \tag{2.10}$$

$$\|\mathbf{BC}\|^2 = 18 \tag{2.11}$$

solution

$$\|\mathbf{CA}\|^2 = (\mathbf{A} - \mathbf{C})^T(\mathbf{A} - \mathbf{C}) \tag{2.12}$$

$$\|\mathbf{CA}\|^2 = (-4)^2 + (-2)^2 + (-4)^2 \tag{2.13}$$

$$\|\mathbf{CA}\|^2 = 36 \tag{2.14}$$

$$\|\mathbf{AB}\| = \|\mathbf{BC}\| = 3\sqrt{2}, \tag{2.15}$$

$$\|\mathbf{CA}\| = 6 \tag{2.16}$$

Therefore the non-collinear vectors $\mathbf{A}, \mathbf{B}, \mathbf{C}$ determine a triangle, and since two sides are equal, that triangle is **isosceles** (with equal sides **AB** and **BC**).

solution (B)

To show the triangle is right-angled, compute the inner product of two adjacent side vectors **AB** and **BC**.

$$(\mathbf{AB})^T(\mathbf{BC}) = \begin{pmatrix}1 & -1 & 4\end{pmatrix}\begin{pmatrix}3\\3\\0\end{pmatrix} \tag{3.1}$$

$$(\mathbf{AB})^T(\mathbf{BC}) = 1\cdot 3 + (-1)\cdot 3 + 4\cdot 0 \tag{3.2}$$

$$(\mathbf{AB})^T(\mathbf{BC}) = 3 - 3 + 0 = 0. \tag{3.3}$$

Since the inner product is zero, $\mathbf{AB} \perp \mathbf{BC}$ and therefore the angle $\angle ABC$ is a right angle; the triangle is **right-angled at** B.

Final statement: The non-collinear vectors $\mathbf{A}, \mathbf{B}, \mathbf{C}$ determine a triangle which is both **isosceles** (with $\|\mathbf{AB}\| = \|\mathbf{BC}\|$) and **right-angled** (with $\mathbf{AB} \perp \mathbf{BC}$); hence the triangle is a *right isosceles* triangle with the right angle at vertex B.

C Source Code:gen point.c

```
#include <stdio.h>
// Function to write points into a file
void generate_points(const char *filename) {
    FILE *fp = fopen(filename, "w");
    if (fp == NULL) {
        printf("Error opening file!\n");
        return;
    }// Points A, B, C
    double A[3] = {0, 7, -10};
    double B[3] = {1, 6, -6};
    double C[3] = {4, 9, -6};
    fprintf(fp, "%lf %lf %lf\n", A[0], A[1], A[2]);
    fprintf(fp, "%lf %lf %lf\n", B[0], B[1], B[2]);
    fprintf(fp, "%lf %lf %lf\n", C[0], C[1], C[2]);
    fclose(fp);
}
```

Python Script: solve triangle.py

```
import ctypes
import numpy as np

# Load the shared object
lib = ctypes.CDLL("./gen_points.so")

# Call the C function to generate points.dat
lib.generate_points(b"points.dat")

# Load points from file
points = np.loadtxt("points.dat")
A, B, C = points

# Function to compute squared distance
def dist2(P, Q):
    return np.sum((P - Q) ** 2)
```

Python Script: solve triangle.py

```
# Squared lengths
AB2 = dist2(A, B)
BC2 = dist2(B, C)
CA2 = dist2(C, A)

print("Squared lengths:")
print("AB^2 =", AB2, " BC^2 =", BC2, " CA^2 =", CA2)

# Check isosceles (two sides equal)
isosceles = (AB2 == BC2) or (BC2 == CA2) or (CA2 == AB2)
print("Isosceles Triangle:", isosceles)

# Check right angle (Pythagoras theorem)
right_angle = (AB2 + BC2 == CA2) or (BC2 + CA2 == AB2) or (CA
print("Right Angled Triangle:", right_angle)
```

Python Script: plot triangle.py

```
import sys
sys.path.insert(0, '/home/dhanush-sagar/matgeo/codes/CoordGeo
import numpy as np
import matplotlib.pyplot as plt

# Local imports
from line.funcs import *
from triangle.funcs import *
from conics.funcs import circ_gen

# Load points
points = np.loadtxt("points.dat")
A, B, C = points
# Plot triangle
tri_coords = np.vstack((A, B, C, A))  # close loop
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
```

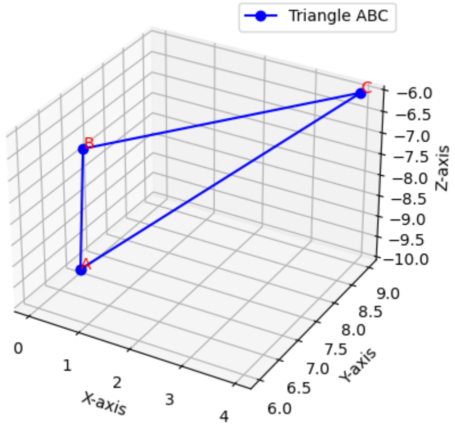
Python Script: plot triangle.py

```
ax.plot(tri_coords[:,0], tri_coords[:,1], tri_coords[:,2], 'b-

# Mark points
ax.text(A[0], A[1], A[2], "A", color='red')
ax.text(B[0], B[1], B[2], "B", color='red')
ax.text(C[0], C[1], C[2], "C", color='red')

ax.set_xlabel('X-axis')
ax.set_ylabel('Y-axis')
ax.set_zlabel('Z-axis')
ax.legend()
plt.savefig("triangle_plot.png")
plt.show()
```

Result Plot



Triangle ABC
A
B
C
X-axis
Y-axis
Z-axis
0
1
2
3
4
6.0
6.5
7.0
7.5
8.0
8.5
9.0
−6.0
−6.5
−7.0
−7.5
−8.0
−8.5
−9.0
−9.5
−10.0