

Matgeo Presentation - Problem 1.6.13

ee25btech11063 - Vejith

August 26, 2025

Question

The points $(0, 5)$, $(0, -9)$ and $(3, 6)$ are not collinear.

Solution

Solution:

point	Name
(0, 5)	Point A
(0, -9)	Point B
(3, 6)	Point C

Table: Variables Used

The collinearity matrix is given by (0.1)

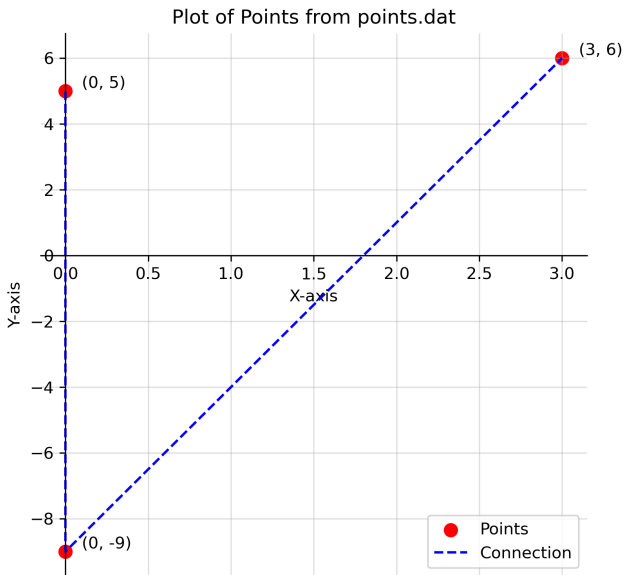
$$(\mathbf{B} - \mathbf{A} \quad \mathbf{C} - \mathbf{A})^T = \begin{pmatrix} 0 & -14 \\ 3 & 1 \end{pmatrix} \quad (0.2)$$

(0.3)

3 points are collinear if the rank of collinearity matrix is 1. but for above matrix by applying any row reduction also we can't create zero rows in matrix.

Solution

⇒ given 3 points A,B,C are not collinear and they form a triangle.



C Code: points.c

```
#include <stdio.h>

int main() {
    FILE *fp;

    // -----
    // Question 1.6.13
    // -----

    fp = fopen("points.dat", "w");
    fprintf(fp, "%d,%d,%d\n", 0, 5, 0); // A
    fprintf(fp, "%d,%d,%d\n", 0, -9, 0); // B
    fprintf(fp, "%d,%d,%d\n", 3, 6, 0); // C
    fclose(fp);
    return 0;
}
```

Python: call_c.py

```
import subprocess

# Compile the C program
subprocess.run(["gcc", "points.c", "-o", "points"])

# Run the compiled C program
result = subprocess.run(["./points"], capture_output=True, text=True)

# Print the output from the C program
print(result.stdout)
```

Python: plot.py

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

# Load the file, using comma as delimiter
points = np.loadtxt("points.dat", delimiter=",")

# Take only the first two columns (x, y)
x = points[:, 0]
y = points[:, 1]

# --- Plot ---
plt.figure(figsize=(6, 6))
plt.scatter(x, y, color="red", s=60, label="Points")
plt.plot(x, y, linestyle="--", color="blue", label="Connection")

# Annotate each point
for xi, yi in zip(x, y):
    plt.text(xi + 0.1, yi + 0.1, f"({xi:g},{yi:g}")

# Axes setup
ax = plt.gca()
ax.spines["left"].set_position("zero")
ax.spines["bottom"].set_position("zero")
ax.spines["right"].set_color("none")
ax.spines["top"].set_color("none")

plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.title("Plot of Points from points.dat")
plt.grid(True, alpha=0.4)
plt.legend()

# Save the figure
```

Python: plot.py

```
plt.savefig("points_plot.png", dpi=300, bbox_inches="tight")  
  
# Show the figure  
plt.show()
```