MATGEO Presentation: 1.6.5

Subhodeep Chakraborty
ee25btech11055,
IIT Hyderabad.

September 5, 2025

1. Problem
2. Solution
   - Theory
   - Calculations
   - Plot
3. C Code
4. Python Code
   - Using C shared objects
   - Plot
   - In Pure Python
   - Plot

Problem Statement

Show that the points $(2, 3, 4)$, $(-1, -2, 1)$, $(5, 8, 7)$ are collinear.

Theory

For three points **A**, **B** and **C** to be collinear:

$$\operatorname{rank}\begin{pmatrix}\mathbf{B}-\mathbf{A} & \mathbf{C}-\mathbf{A}\end{pmatrix} = \operatorname{rank}\begin{pmatrix}\mathbf{B}-\mathbf{A} & \mathbf{C}-\mathbf{A}\end{pmatrix}^{\top} = 1 \tag{3.1}$$

Given:

$$\mathbf{A} \equiv (2, 3, 4) \tag{3.2}$$

$$\mathbf{B} \equiv (-1, -2, 1) \tag{3.3}$$

$$\mathbf{C} \equiv (5, 8, 7) \tag{3.4}$$

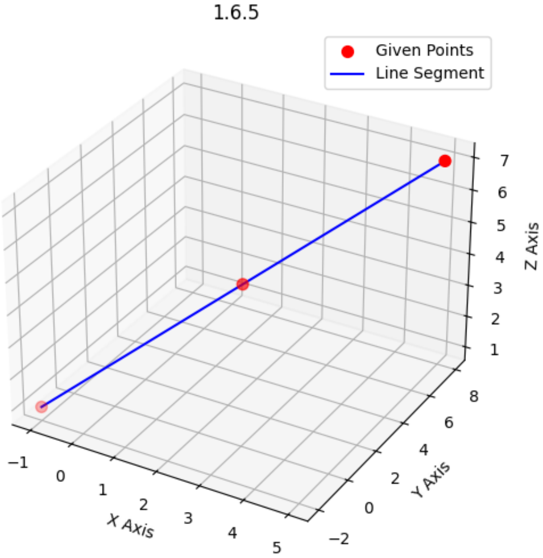
Calculations

The transpose of the collinearity matrix can be expressed as:

$$\begin{aligned}\begin{pmatrix}\mathbf{B}-\mathbf{A} & \mathbf{C}-\mathbf{A}\end{pmatrix}^\top &= \begin{pmatrix}-3 & -5 & -3\\ 3 & 5 & 3\end{pmatrix}\\ &\xleftrightarrow{R_2=R_2+R_1} \begin{pmatrix}-3 & -5 & -3\\ 0 & 0 & 0\end{pmatrix}\end{aligned}$$

which has rank 1. Using 3.1, we conclude that the given points are collinear.

Plot


1.6.5
● Given Points
Line Segment
X Axis
−1
0
1
2
3
4
5
Y Axis
−2
0
2
4
6
8
Z Axis
1
2
3
4
5
6
7

C code for generating points on line

```
void point_gen(const double* P1, const double* P2, double t, double*
     result_point) {
    result_point[0] = P1[0] + t * (P2[0] - P1[0]);
    result_point[1] = P1[1] + t * (P2[1] - P1[1]);
    result_point[2] = P1[2] + t * (P2[2] - P1[2]);
}
```

Python code for plotting using C

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

lib = ctypes.CDLL(”./line.so”)

get_point = lib.point_gen
get_point.argtypes = [
    ctypes.POINTER(ctypes.c_double), # P1
    ctypes.POINTER(ctypes.c_double), # P2
    ctypes.c_double, # t
    ctypes.POINTER(ctypes.c_double), # result_point
]
get_point.restype = None
```

Python code for plotting using C

```
DoubleArray3 = ctypes.c_double * 3
P1_arr = DoubleArray3(-1, -2, 1)
P2_arr = DoubleArray3(5, 8, 7)

t_values = np.linspace(0, 1, 100)
line_points_x, line_points_y, line_points_z = [], [], []

for t in t_values:
    result_arr = DoubleArray3()

    get_point(P1_arr, P2_arr, t, result_arr)

    line_points_x.append(result_arr[0])
    line_points_y.append(result_arr[1])
    line_points_z.append(result_arr[2])

original_points = np.array([[-1, -2, 1], [2, 3, 4], [5, 8, 7]])
```

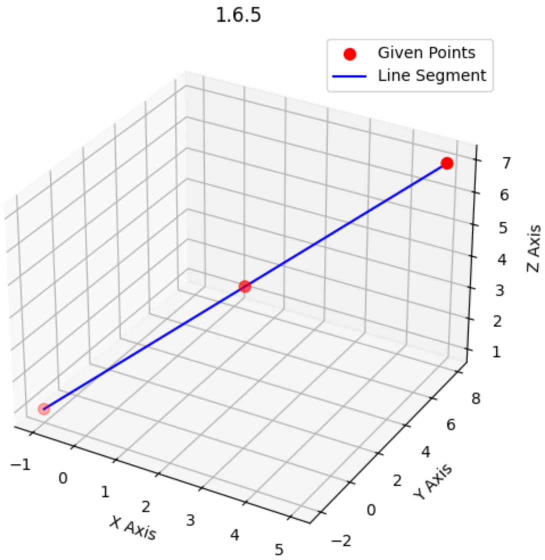
Python code for plotting using C

```
fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(111, projection=”3d”)

ax.scatter(
    original_points[:, 0],
    original_points[:, 1],
    original_points[:, 2],
    color=”red”,
    s=50,
    label=”Given-Points”,
)
```

Python code for plotting using C

```
ax.plot(
    line_points_x,
    line_points_y,
    line_points_z,
    color=”blue”,
    label=”Line-Segment”,
)

ax.set_xlabel(”X-Axis”)
ax.set_ylabel(”Y-Axis”)
ax.set_zlabel(”Z-Axis”)
ax.set_title(”1.6.5”)
ax.legend()
ax.grid(True)

plt.savefig(”../figs/plot.png”)
plt.show()
```

Plot


1.6.5
● Given Points
Line Segment
−1
0
1
2
3
4
5
X Axis
−2
0
2
4
6
8
Y Axis
1
2
3
4
5
6
7
Z Axis

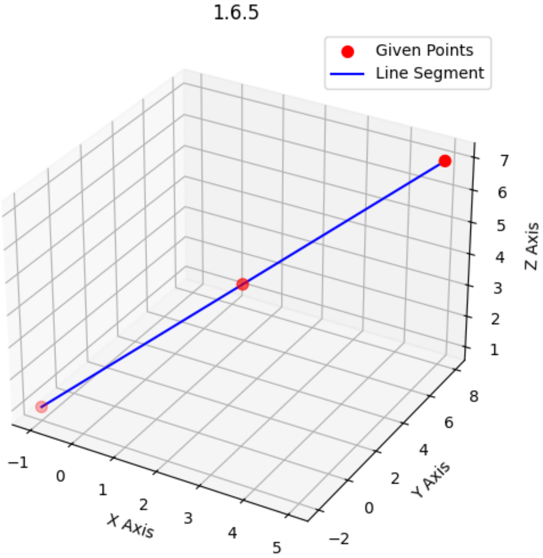
Pure Python Code for Plotting

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

points = np.array([[−1, −2, 1], [2, 3, 4], [5, 8, 7]])

x_coords = points[:, 0]
y_coords = points[:, 1]
z_coords = points[:, 2]

fig = plt.figure(figsize=(8, 6))
ax = fig.add_subplot(111, projection="3d")

ax.scatter(x_coords, y_coords, z_coords, color="red", s=50, label="Given-
    Points")
ax.plot(x_coords, y_coords, z_coords, color="blue", label="Line-Segment"
    )
```

Pure Python Code for Plotting

```
ax.set_xlabel("X-Axis")
ax.set_ylabel("Y-Axis")
ax.set_zlabel("Z-Axis")
ax.set_title("1.6.5")
ax.legend()
ax.grid(True)

plt.savefig("../figs/python.png")
plt.show()
```

Pure Python Plot


● Given Points
Line Segment
X Axis
Y Axis
Z Axis