MatGeo Presentation - Problem 12.389

EE25BTECH11064 - Yojit Manral

Question

$$x + 2y + z = 4$$
$$2x + y + 2z = 5$$
$$x - y + z = 1$$

The system of algebraic equations given above has

(a) a unique solution of $(x = 1, y = 1, z = 1)$.

(b) only the two solutions of $(x = 1, y = 1, z = 1)$ and $(x = 2, y = 1, z = 0)$.

(c) infinite number of solutions.

(d) no feasible solution.

Solution

→ The above equations can be written in matrix form as

$$\begin{pmatrix} 1 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & -1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 4 \\ 5 \\ 1 \end{pmatrix} \tag{0.1}$$

$$\mathbf{A} \quad \mathbf{x} = \mathbf{B} \tag{0.2}$$

→ Using the augmented matrix $(\mathbf{A}|\mathbf{B})$

$$\left(\begin{array}{ccc|c} 1 & 2 & 1 & 4 \\ 2 & 1 & 2 & 5 \\ 1 & -1 & 1 & 1 \end{array}\right) \xrightarrow[R_3 \leftrightarrow R_3 - R_1]{R_2 \leftrightarrow R_2 - 2R_3} \left(\begin{array}{ccc|c} 1 & 2 & 1 & 4 \\ 0 & 3 & 0 & 3 \\ 0 & -3 & 0 & -3 \end{array}\right) \tag{0.3}$$

$$\xrightarrow[R_1 \leftrightarrow R_1 - (2/3)R_2]{R_3 \leftrightarrow R_3 + R_2} \left(\begin{array}{ccc|c} 1 & 0 & 1 & 2 \\ 0 & 3 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{array}\right) \tag{0.4}$$

$$\implies x + z = 2 \text{ and } y = 1 \tag{0.5}$$

→ So, the system of equations has infinite solutions given by the intersection of the planes $x + z = 2$ and $y = 1$.

Solution

$\rightarrow$ Therefore, *(c) infinite number of solutions.* is the correct option.

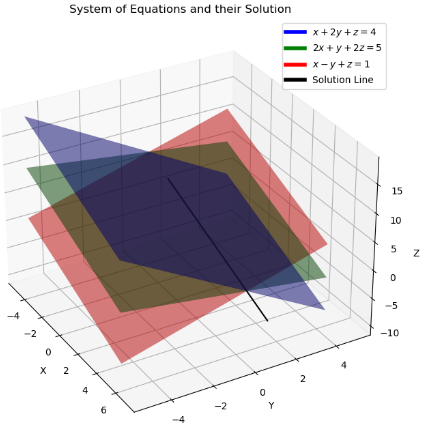

Figure: Plot of system of equations and solution line

File: points.c

```c
#include <stdio.h>

int main() {
  FILE *fp;

  // -------------------
  // Question 12.389
  // -------------------


  fp = fopen("points.dat", "w");
  fprintf(fp, "%d,%d,%d,%d\n", 1, 2, 1, 4); // 1
  fprintf(fp, "%d,%d,%d,%d\n", 2, 1, 2, 5); // 2
  fprintf(fp, "%d,%d,%d,%d\n", 1, -1, 1, 1); // 3
  fclose(fp);
  return 0;
}
```

File: call_c.py

```python
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)
```

File: plot.py

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from matplotlib.lines import Line2D

# Define the equations
def plane1(x, y):
    return 4 - x - 2 * y

def plane2(x, y):
    return (5 - 2*x - y) / 2

def plane3(x, y):
    return 1 - x + y

# Create meshgrid for x and y
x_vals = np.linspace(-5, 5, 10)
y_vals = np.linspace(-5, 5, 10)
X, Y = np.meshgrid(x_vals, y_vals)

# Compute Z values for each plane
Z1 = plane1(X, Y)
Z2 = plane2(X, Y)
Z3 = plane3(X, Y)

# Plotting the planes and solution
fig = plt.figure(figsize=(12, 8))
ax = fig.add_subplot(111, projection='3d')
```

File: plot.py

```
# Plot each plane
surf1 = ax.plot_surface(X, Y, Z1, color='blue', alpha=0.5, rstride=100, cstride=100)
surf2 = ax.plot_surface(X, Y, Z2, color='green', alpha=0.5, rstride=100, cstride=100)
surf3 = ax.plot_surface(X, Y, Z3, color='red', alpha=0.5, rstride=100, cstride=100)

# Plot the solution line (x = 2 - z, y = 1)
z_vals = np.linspace(-5, 5, 10)
x_sol = 2 - z_vals
y_sol = np.ones_like(z_vals)

line = ax.plot(x_sol, y_sol, z_vals, color='black', label='Solution Line')

# Labels
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')

ax.set_title('System of Equations and their Solution')

# Manually add legend
ax.legend(handles=[
    Line2D([0], [0], color='blue', lw=4, label=r'$x + 2y + z = 4$'),
    Line2D([0], [0], color='green', lw=4, label=r'$2x + y + 2z = 5$'),
    Line2D([0], [0], color='red', lw=4, label=r'$x - y + z = 1$'),
    Line2D([0], [0], color='black', lw=4, label='Solution Line')
])

# Show the plot
ax.view_init(30, -30)
plt.show()
```