4.7.22

Sai Sreevallabh - EE25BTECH11031

September 30, 2025

Question

In what direction should a line be drawn through the point $(1, 2)$ so that its point of intersection with the line $x + y = 4$ is at a distance $\sqrt{63}$?

Theoretical Solution

The given point is $\mathbf{P} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.

The given line can be represented as $\mathbf{n}^\top \mathbf{x} = c$, where

$$\mathbf{n} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad c = 4 \tag{1}$$

A parametric point on the line passing through the point $\mathbf{P}$ is given by

$$\mathbf{r} = \mathbf{P} + \lambda \mathbf{m} \tag{2}$$

where $\mathbf{m} = \begin{pmatrix} 1 \\ m \end{pmatrix}$

Theoretical Solution

Plugging in the parametric form of the point in the line equation, we get:

$$\mathbf{n}^\top (\mathbf{P} + \lambda \mathbf{m}) = c \tag{3}$$

$$\lambda = \frac{c - \mathbf{n}^\top \mathbf{P}}{\mathbf{n}^\top \mathbf{m}} \tag{4}$$

Replacing this value of λ in the equation of the parametric point, we get it to be

$$\mathbf{r} = \mathbf{P} + \left(\frac{c - \mathbf{n}^\top \mathbf{P}}{\mathbf{n}^\top \mathbf{m}} \right) \mathbf{m} \tag{5}$$

Theoretical Solution

Substituting the values, we get

$$\sqrt{63} = \left|\frac{4-3}{1+m}\right| \sqrt{1+m^2} \tag{6}$$

Squaring on both sides:

$$63 = \frac{1+m^2}{(1+m)^2} \tag{7}$$

Theoretical Solution

This is an equation in m. Upon solving this equation, we get:

$$m = \frac{-63 \pm 5\sqrt{5}}{62} \tag{8}$$

$\therefore$ The direction vector $\mathbf{m}$ can take two values:

$$\mathbf{m} = \begin{pmatrix} 1 \\ \frac{-63+5\sqrt{5}}{62} \end{pmatrix} \quad \text{or} \quad \mathbf{m} = \begin{pmatrix} 1 \\ \frac{-63-5\sqrt{5}}{62} \end{pmatrix}$$

C Code - Function to Find y Coordinate of P

```c
#include <stdio.h>
#include <math.h>

void solve_quad(double a, double b, double c, double sols[2]){

    double D = b*b - 4*a*c;

    sols[0] = (-b+pow(D,0.5))/(2*a);
    sols[1] = (-b-pow(D,0.5))/(2*a);

}
```

Python Code - Using Shared Object

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


c_lib = ctypes.CDLL("./code.so")

c_lib.solve_quad.argtypes = [ctypes.c_double, ctypes.c_double,
    ctypes.c_double, ctypes.c_double*2]

a,b,c = 31.0, 63.0, 31.0
P = np.array([1,2])
sol = (ctypes.c_double*2)(0.0,0.0)

c_lib.solve_quad(a,b,c,sol)
```

Python Code - Using Shared Object

```
m1 = sol[0]
m2 = sol[1]

t = (2+m1)/(m1+1)
k = (2+m2)/(m2+1)

A = np.array([t,4-t], dtype = np.double)
B = np.array([k, 4-k], dtype = np.double)

plt.plot([-5,8], [9,-4], c='black', label = "$x+y=4$")
plt.plot([P[0],A[0]], [P[1], A[1]], c='green')
plt.plot([P[0],B[0]], [P[1], B[1]], c='red')
```

Python Code - Using Shared Object

```
plt.scatter([1,t,k],[2,4-t,4-k])
plt.annotate(
        "P(1,2)",
        xy=(1,2),
        xytext = (-15,-15),
        textcoords = "offset points"
        )
```

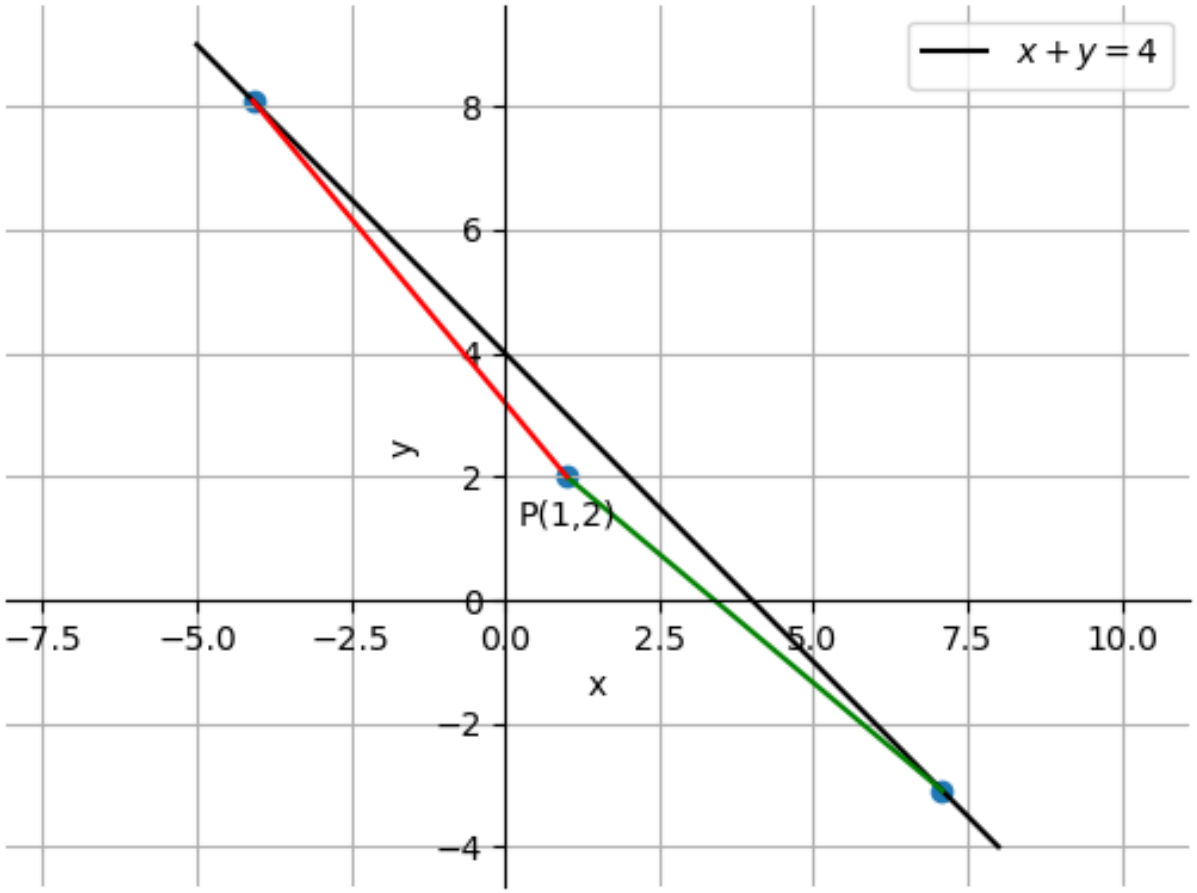
Python Code - Using Shared Object

```
ax = plt.gca()
ax.spines['top'].set_color('none')
ax.spines['left'].set_position('zero')
ax.spines['right'].set_color('none')
ax.spines['bottom'].set_position('zero')
plt.xlabel('$x$')
plt.ylabel('$y$')
plt.legend(loc='best')
plt.grid()
plt.axis('equal')

plt.savefig("../Figs/plot(py+C).png")

plt.show()
```

Plot-Using Both C and Python



Python Code

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

a,b,c = 31,63,31
P = np.array([1,2])

roots = np.roots([a,b,c])

t = (2+roots[0])/(roots[0]+1)

A = np.array([t,4-t], dtype = np.double)

k = (2+roots[1])/(roots[1]+1)

B = np.array([k, 4-k], dtype = np.double)
```

Python Code

```
plt.plot([-5,8], [9,-4], label = "$x+y=4$")
plt.plot([P[0],A[0]], [P[1], A[1]])
plt.plot([P[0],B[0]], [P[1], B[1]])

plt.scatter([1,t,k],[2,4-t,4-k])
plt.annotate(
        "P(1,2)",
        xy=(1,2),
        xytext = (-15,-15),
        textcoords = "offset points"
        )
```

Python Code

```
ax = plt.gca()
ax.spines['top'].set_color('none')
ax.spines['bottom'].set_position('zero')
ax.spines['right'].set_color('none')
ax.spines['left'].set_position('zero')
plt.xlabel('x')
plt.ylabel('y')
plt.legend(loc='best')
plt.grid()
plt.axis('equal')

plt.savefig("../Figs/plot(py).png")
plt.show()
```

Plot-Using Python only