4.8.10

AI25BTECH11014 - Gooty Suhas

October 1, 2025

Question

Find the foot of the perpendicular from

$$\mathbf{P} = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$$

to the plane

$$\mathbf{N}^T\mathbf{x} = 1, \quad \mathbf{N} = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}$$

Also find the distance $\|\mathbf{P} - \mathbf{Q}\|$ and the image of $\mathbf{P}$ treating the plane as a mirror.

Foot of Perpendicular

We use:

$$\mathbf{Q} = \mathbf{P} - \frac{\mathbf{N}^T\mathbf{P} - 1}{\mathbf{N}^T\mathbf{N}}\mathbf{N}$$

Compute:

$$\mathbf{N}^T\mathbf{P} = \begin{pmatrix} 2 & -1 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} = 5 \quad \Rightarrow \quad \mathbf{N}^T\mathbf{P} - 1 = 4$$

$$\mathbf{N}^T\mathbf{N} = \begin{pmatrix} 2 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} = 6 \Rightarrow \mathbf{Q} = \mathbf{P} - \frac{2}{3}\mathbf{N} = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} - \frac{2}{3} \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{5}{3} \\ \frac{8}{3} \\ \frac{1}{3} \end{pmatrix}$$

Distance

$$\|\mathbf{P}-\mathbf{Q}\| = \left\|\frac{2}{3}\mathbf{N}\right\| = \frac{2}{3}\sqrt{6}$$

This is the perpendicular distance from $\mathbf{P}$ to the plane.

Image of $\mathbf{P}$

The reflected image is:

$$\mathbf{R} = \mathbf{P} - 2 \cdot \frac{\mathbf{N}^T\mathbf{P} - 1}{\mathbf{N}^T\mathbf{N}}\mathbf{N} = \mathbf{P} - \frac{4}{3}\mathbf{N} = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} - \frac{4}{3}\begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{3} \\ \frac{10}{3} \\ -\frac{1}{3} \end{pmatrix}$$

Final Answer

$$\boxed{\mathbf{Q} = \begin{pmatrix} \frac{5}{3} \\ \frac{8}{3} \\ \frac{1}{3} \end{pmatrix}} \qquad \boxed{\|\mathbf{P} - \mathbf{Q}\| = \frac{2}{3}\sqrt{6}} \qquad \boxed{\text{Image of } \mathbf{P} : \begin{pmatrix} \frac{1}{3} \\ \frac{10}{3} \\ -\frac{1}{3} \end{pmatrix}}$$

Python Code — SymPy

```python
from sympy import Matrix, sqrt

P = Matrix([3, 2, 1])
N = Matrix([2, -1, 1])

num = N.dot(P) - 1
den = N.dot(N)

Q = P - (num / den) * N
dist = (num / den) * sqrt(den)
R = P - 2 * (num / den) * N

print("Q =", Q)
print("Distance =", dist)
print("Image =", R)
```

C Code — Matrix Only (1/2)

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

int main() {
    double P[3], N[3];
    for(int i = 0; i < 3; i++)
        scanf("%lf", &P[i]);
    for(int i = 0; i < 3; i++)
        scanf("%lf", &N[i]);

    double dotPN = 0, dotNN = 0;
    for(int i = 0; i < 3; i++) {
        dotPN += P[i] * N[i];
        dotNN += N[i] * N[i];
    }
```

C Code — Matrix Only (2/2)

```
    double scalar = (dotPN - 1) / dotNN;
    double Q[3], R[3];
    for(int i = 0; i < 3; i++) {
        Q[i] = P[i] - scalar * N[i];
        R[i] = P[i] - 2 * scalar * N[i];
    }

    double dist = scalar * sqrt(dotNN);

    printf("Q = (%.3f, %.3f, %.3f)\n", Q[0], Q[1], Q[2]);
    printf("Distance = %.3f\n", dist);
    printf("Image = (%.3f, %.3f, %.3f)\n", R[0], R[1], R[2]);
    return 0;
}
```

Python Code — With .so

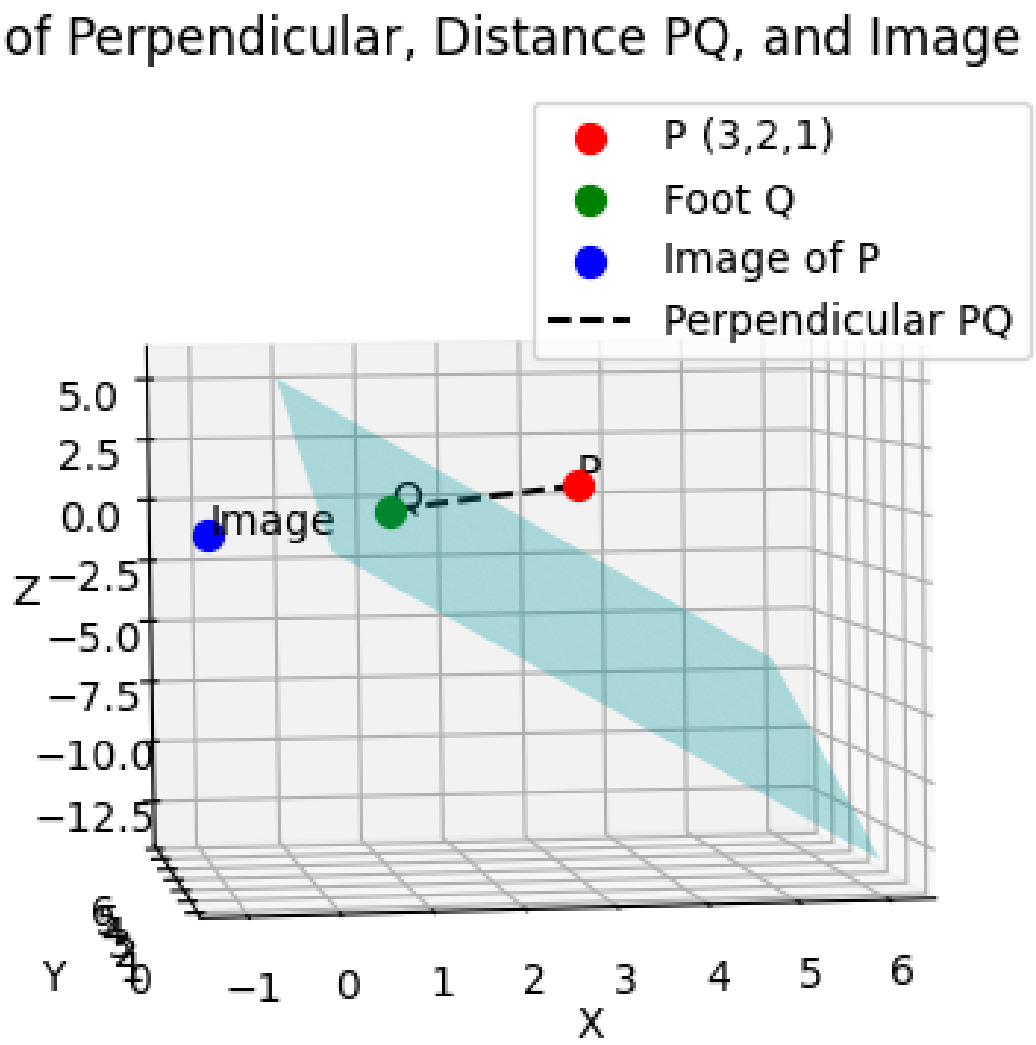
```python
import subprocess

P = [3.0, 2.0, 1.0]
N = [2.0, -1.0, 1.0]
inputs = P + N
input_str = ' '.join(map(str, inputs))

result = subprocess.run(
    ['./plane_solver'],
    input=input_str,
    capture_output=True,
    text=True
)

print(result.stdout.strip())
```

3D Plot


Foot of Perpendicular, Distance PQ, and Image of P
P (3,2,1)
Foot Q
Image of P
Perpendicular PQ
Image
Q
P
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12.5
Z
Y
−1
0
1
2
3
4
5
6
X