4.2.16

EE25BTECH11018 - Darisy Sreetej

October 1 , 2025

Question

Find the equation of the plane passing through the points having position vectors $\hat{i}+\hat{j}-2\hat{k}, 2\hat{i}-\hat{j}+\hat{k}, \hat{i}+2\hat{j}+\hat{k}$.Write the equation of the plane passing through a point $(2,3,7)$ and parallel to the plane obtained above. Hence, find the distance between the two parallel planes.

Formula

The distance between the parallel planes is given by

$$\text{Distance} = \frac{|d_1 - d_2|}{\|\mathbf{n}\|} \tag{1}$$

where $\mathbf{n}^\top \mathbf{x} = d_1$ and $\mathbf{n}^\top \mathbf{x} = d_2$ are the parallel planes

Table:

A	$\begin{pmatrix} 1 \\ 1 \\ -2 \end{pmatrix}$
B	$\begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}$
C	$\begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$
P	$\begin{pmatrix} 2 \\ 3 \\ 7 \end{pmatrix}$

Obtaining the plane

Let the equation of plane be

$$\mathbf{n}^\top \mathbf{x} = C_1 \tag{2}$$

A,B,C satisfies this equation,

$$\mathbf{n}^\top \mathbf{A} = C_1, \mathbf{n}^\top \mathbf{B} = C_1, \mathbf{n}^\top \mathbf{C} = C_1 \tag{3}$$

$$\begin{pmatrix} \mathbf{A} \\ \mathbf{B} \\ \mathbf{C} \end{pmatrix}^\top \mathbf{n} = \begin{pmatrix} C_1 \\ C_1 \\ C_1 \end{pmatrix} \tag{4}$$

Using augmented matrix,

$$\left(\begin{array}{ccc|c} 1 & 1 & -2 & C_1 \\ 2 & -1 & 1 & C_1 \\ 1 & 2 & 1 & C_1 \end{array}\right) \tag{5}$$

$R_2 = R_2 - 2R_1$
$R_3 = R_3 - R_1$

$$\left(\begin{array}{ccc|c} 1 & 1 & -2 & C_1 \\ 0 & -3 & 5 & -C_1 \\ 0 & 1 & 3 & 0 \end{array}\right) \tag{6}$$

$R_2 \iff R_3$

$$\left(\begin{array}{ccc|c} 1 & 1 & -2 & C_1 \\ 0 & 1 & 3 & 0 \\ 0 & -3 & 5 & -C_1 \end{array}\right) \tag{7}$$

$R_3 = R_3 + 3R_2$

$$\left(\begin{array}{ccc|c} 1 & 1 & -2 & C_1 \\ 0 & 1 & 3 & 0 \\ 0 & 0 & 14 & -C_1 \end{array}\right) \tag{8}$$

$$14z + C_1 = 0 \implies z = \frac{-C_1}{14} \tag{9}$$

$$y + 3z = 0 \implies y = \frac{3C_1}{14} \tag{10}$$

$$x + y - 2z = C_1 \implies x = \frac{9C_1}{17} \tag{11}$$

Let $C_1 = 14$

$$\mathbf{n} = \begin{pmatrix} 9 \\ 3 \\ -1 \end{pmatrix}, C_1 = 14 \tag{12}$$

Equation of the plane

$$\begin{pmatrix} 9 \\ 3 \\ -1 \end{pmatrix}^\top \mathbf{x} = 14 \tag{13}$$

Obtaining the Parallel plane

For finding parallel plane passing through P ,

$$\mathbf{n}^\top \mathbf{x} = C_2 \tag{14}$$

$$\mathbf{n}^\top \mathbf{P} = C_2 \tag{15}$$

$$C_2 = \begin{pmatrix} 9 & 3 & -1 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 7 \end{pmatrix} \tag{16}$$

$$C_2 = 20 \tag{17}$$

Equation of plane parallel to given plane passing through point P is

$$\begin{pmatrix} 9 \\ 3 \\ -1 \end{pmatrix}^\top \mathbf{x} = 20 \tag{18}$$

Distance between the planes

The 2 planes obtained are parallel since their normal vectors are the same
The normal vector of the planes $\mathbf{n}$

$$\mathbf{n} = \begin{pmatrix} 9 \\ 3 \\ -1 \end{pmatrix} \tag{19}$$

The distance between the planes is given by this formula

$$\text{Distance} = \frac{|d_1 - d_2|}{\|\mathbf{n}\|} \tag{20}$$

Where $d_1 = 14$ and $d_2 = 20$

$$\|\mathbf{n}\| = \left(\sqrt{(9)^2 + (3)^2 + (-1)^2} \right) = \sqrt{91} \tag{21}$$

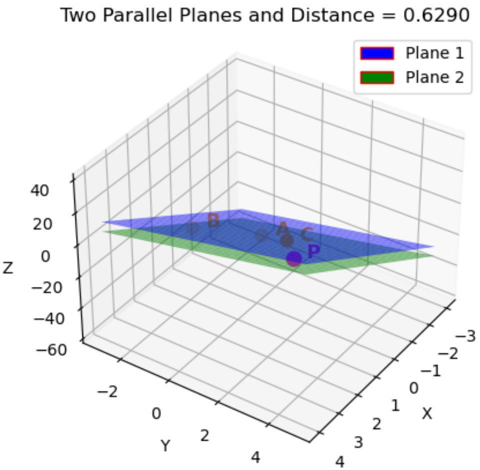
Substituting these values in the distance formula, we get

$$\therefore \text{Distance} = \frac{|14 - 20|}{\sqrt{91}} \tag{22}$$

$$\text{Distance} = \frac{6}{\sqrt{91}} \tag{23}$$

Therefore, the distance between the planes is $\frac{6}{\sqrt{91}}$

Plot



Two Parallel Planes and Distance = 0.6290
Plane 1
Plane 2
B
A
C
P
40
20
0
−20
−40
−60
Z
−2
0
2
4
Y
−3
−2
−1
0
1
2
3
4
X

C Code

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

void plane_from_points(double P1[3], double P2[3], double P3[3],
    double coeff[4]) {
    coeff[0] = (P2[1]-P1[1])*(P3[2]-P1[2]) - (P2[2]-P1[2])*(P3
        [1]-P1[1]);
    coeff[1] = (P2[2]-P1[2])*(P3[0]-P1[0]) - (P2[0]-P1[0])*(P3
        [2]-P1[2]);
    coeff[2] = (P2[0]-P1[0])*(P3[1]-P1[1]) - (P2[1]-P1[1])*(P3
        [0]-P1[0]);
    coeff[3] = -(coeff[0]*P1[0] + coeff[1]*P1[1] + coeff[2]*P1
        [2]);
}

void parallel_plane_through_point(double coeff[4], double Q[3],
    double coeff2[4]) {
    coeff2[0] = coeff[0];
```

C Code

```
coeff2[1] = coeff[1];
    coeff2[2] = coeff[2];
    coeff2[3] = -(coeff2[0]*Q[0] + coeff2[1]*Q[1] + coeff2[2]*Q
        [2]);
}

double norm(double *n) {
    return sqrt(n[0]*n[0] + n[1]*n[1] + n[2]*n[2]);
}

double plane_distance(double *n, double d1, double d2) {
    double num = fabs(d1 - d2);
    double denom = norm(n);
    return num / denom;
}
```

Python + C Code

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

lib = ctypes.CDLL("./plane_distance.so")

lib.plane_from_points.argtypes = [
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double)
]

lib.parallel_plane_through_point.argtypes = [
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double)
]
```

Python + C code

```
lib.plane_distance.argtypes = [
    ctypes.POINTER(ctypes.c_double),
    ctypes.c_double,
    ctypes.c_double
]
lib.plane_distance.restype = ctypes.c_double

A = [1, 1, -2] # Given vector A
B = [2, -1, 1] # Given vector B
C = [1, 2, 1] # Given vector C
P = [2, 3, 7] # Point P for the parallel plane

A_c = (ctypes.c_double * 3)(*A)
B_c = (ctypes.c_double * 3)(*B)
C_c = (ctypes.c_double * 3)(*C)
P_c = (ctypes.c_double * 3)(*P)

coeff1 = (ctypes.c_double * 4)()
```

Python + C code

```
lib.plane_from_points(A_c, B_c, C_c, coeff1)
plane1 = np.array(coeff1[:])

coeff2 = (ctypes.c_double * 4)()
lib.parallel_plane_through_point(coeff1, P_c, coeff2)
plane2 = np.array(coeff2[:])

a, b, c, d1 = plane1
a2, b2, c2, d2 = plane2

n = np.array([a, b, c], dtype=np.double)
dist = lib.plane_distance(n.ctypes.data_as(ctypes.POINTER(ctypes.
    c_double)), d1, d2)
print(f"Distance between planes: {dist:.4f}")

xx, yy = np.meshgrid(np.linspace(-2, 3, 30), np.linspace(-2, 4,
    30))
zz1 = (-d1 - a * xx - b * yy) / c
zz2 = (-d2 - a2 * xx - b2 * yy) / c2
```

Python + C code

```
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.view_init(elev=25, azim=45)

ax.plot_surface(xx, yy, zz1, alpha=0.5, color='blue')
ax.plot_surface(xx, yy, zz2, alpha=0.5, color='green')
points = np.array([A, B, C, P])
labels = ["A", "B", "C", "P"]
for (x, y, z), label in zip(points, labels):
    ax.scatter(x, y, z, color='red', s=50)
    ax.text(x, y, z, label, color='black')

ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Z")
ax.set_title(f"Two Planes and Distance = {dist:.4f}")
plt.show()
```

Python code

```python
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.patches import Patch

def plane_from_points(A, B, C):
    AB = np.array(B) - np.array(A)
    AC = np.array(C) - np.array(A)
    n = np.cross(AB, AC)
    d = -np.dot(n, A)
    return n, d

def parallel_plane_through_point(n, d, P):
    d_new = -np.dot(n, P)
    return n, d_new

def plane_distance(n, d1, d2):
    return abs(d1 - d2) / np.linalg.norm(n)
```

Python code

```
# Given points
A = [1, 1, -2]
B = [2, -1, 1]
C = [1, 2, 1]
P = [2, 3, 7]

# Calculate plane coefficients
n1, d1 = plane_from_points(A, B, C)
n2, d2 = parallel_plane_through_point(n1, d1, P)
dist = plane_distance(n1, d1, d2)
print(f"Distance between planes: {dist:.4f}")

# Create mesh grid for plotting planes
xx, yy = np.meshgrid(np.linspace(-3, 4, 30), np.linspace(-3, 5,
    30))
zz1 = (-d1 - n1[0]*xx - n1[1]*yy) / n1[2]
zz2 = (-d2 - n2[0]*xx - n2[1]*yy) / n2[2]
```

Python code

```
# Plot setup
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.view_init(elev=30, azim=60)

# Plot planes with legend
ax.plot_surface(xx, yy, zz1, alpha=0.5, color='blue', label='
    Plane 1')
ax.plot_surface(xx, yy, zz2, alpha=0.5, color='green', label='
    Plane 2')

# Plot points directly (no legend entry)
ax.scatter(*zip(*[A, B, C]), color='orange', s=60)
ax.scatter(*P, color='magenta', s=80)
# Mark points clearly with labels
for point, label, color in zip([A, B, C, P], ["A", "B", "C", "P"
    ], ['orange', 'orange', 'orange', 'magenta']):
    ax.text(point[0], point[1], point[2], f' {label}', color=
        color, fontsize=12, fontweight='bold')
```

Python code

```python
# Create legend only for planes using proxy artists
legend_elements = [
    Patch(facecolor='blue', edgecolor='r', label='Plane 1'),
    Patch(facecolor='green', edgecolor='r', label='Plane 2'),
]

ax.legend(handles=legend_elements, loc='best')

ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Z")
ax.set_title(f"Two Parallel Planes and Distance = {dist:.4f}")

plt.show()
```