Equidistant Locus Problem

EE25BTECH11008 - Anirudh M Abhilash

September 12, 2025

Problem Statement

Find a relation between x and y such that the point

$$\mathbf{P}(x, y)$$

is equidistant from

$$\mathbf{A}(7, 1), \quad \mathbf{B}(3, 5).$$

Solution

Let

$$\mathbf{P} = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 7 \\ 1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 3 \\ 5 \end{pmatrix}.$$

Since $\mathbf{P}$ is equidistant from $\mathbf{A}$ and $\mathbf{B}$,

$$\|\mathbf{P} - \mathbf{A}\| = \|\mathbf{P} - \mathbf{B}\|.$$

Squaring both sides and using the inner product,

$$(\mathbf{P} - \mathbf{A})^\top(\mathbf{P} - \mathbf{A}) = (\mathbf{P} - \mathbf{B})^\top(\mathbf{P} - \mathbf{B}) \tag{1}$$

$$\mathbf{P}^\top\mathbf{P} - 2\mathbf{P}^\top\mathbf{A} + \mathbf{A}^\top\mathbf{A} = \mathbf{P}^\top\mathbf{P} - 2\mathbf{P}^\top\mathbf{B} + \mathbf{B}^\top\mathbf{B}. \tag{2}$$

Cancelling $\mathbf{P}^\top\mathbf{P}$,

$$2\mathbf{P}^\top(\mathbf{B} - \mathbf{A}) = \mathbf{B}^\top\mathbf{B} - \mathbf{A}^\top\mathbf{A}. \tag{3}$$

Solution (cont..)

Now,

$$\mathbf{B}-\mathbf{A} = \begin{pmatrix} 3 \\ 5 \end{pmatrix} - \begin{pmatrix} 7 \\ 1 \end{pmatrix} = \begin{pmatrix} -4 \\ 4 \end{pmatrix}, \quad \mathbf{B}^\top \mathbf{B} = 3^2+5^2 = 34, \quad \mathbf{A}^\top \mathbf{A} = 7^2+1^2 =$$

Thus,

$$2\begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} -4 \\ 4 \end{pmatrix} = 34 - 50 \tag{4}$$

$$-4x + 4y = -8 \tag{5}$$

$$y - x = -2. \tag{6}$$

Hence, the required relation is

$$\boxed{y = x - 2}$$

Python Code

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

A = np.array([7, 1])
B = np.array([3, 5])

x_vals = np.linspace(0, 10, 100)
y_vals = x_vals - 2

plt.plot(x_vals, y_vals, 'k-')
plt.text(9, 6, r'$y=x-2$', fontsize=15)
```

Python Code(Cont..)

```
plt.scatter(*A, color='red')
plt.scatter(*B, color='blue')

plt.text(A[0]+0.3, A[1], 'A-(7,1)')
plt.text(B[0]+0.3, B[1], 'B-(3,5)')

plt.axhline(0, color='black', lw=0.5)
plt.axvline(0, color='black', lw=0.5)
plt.axis('equal')
plt.show()
```

C Code

```
#include <stdio.h>

void perpendicular_bisector(double x1, double y1,
                            double x2, double y2,
                            double *a, double *b,
                            double *c) {
    double mx = (x1 + x2) / 2.0;
    double my = (y1 + y2) / 2.0;

    double dx = x2 - x1;
    double dy = y2 - y1;

    *a = dx;
    *b = dy;
    *c = -((*a) * mx + (*b) * my);
}
```

Using C Code in Python

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

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

lib.perpendicular_bisector.argtypes = (
    ctypes.c_double, ctypes.c_double,
    ctypes.c_double, ctypes.c_double,
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double),
    ctypes.POINTER(ctypes.c_double)
)

A, B = (7.0, 1.0), (3.0, 5.0)
```

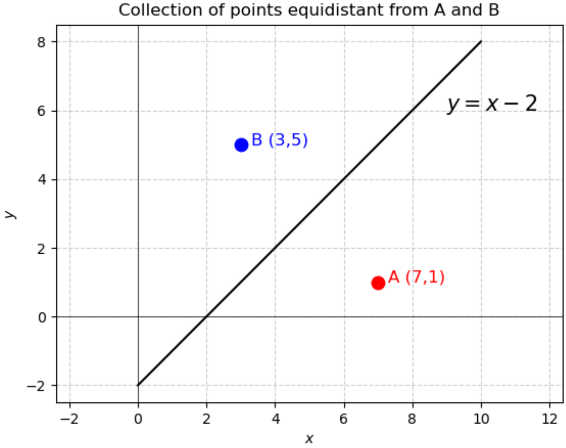
Using C Code in Python(Cont..)

```
a = ctypes.c_double()
b = ctypes.c_double()
c = ctypes.c_double()

lib.perpendicular_bisector(A[0], A[1], B[0], B[1],
    ctypes.byref(a),
    ctypes.byref(b),
    ctypes.byref(c)
    )

x = np.linspace(0, 10, 100)
y = (-a.value*x - c.value)/b.value

plt.plot(x, y, 'k-', label='Bisector')
```

Using C Code in Python(Cont..)

```
plt.text(A[0] + 0.3, A[1], r'A(7,1)', fontsize=12, color='red')
plt.text(B[0] + 0.3, B[1], r'B(3,5)', fontsize=12, color='blue')
plt.text(7, ( -a.value*6 - c.value)/b.value + 0.3,
         f"{a.value:.0f}x+{b.value:.0f}y+{c.value:.0f}=0",
         fontsize=12, color="black")
plt.scatter(*A, color='red', label='A(7,1)')
plt.scatter(*B, color='blue', label='B(3,5)')
plt.axhline(0, color='gray', lw=0.5)
plt.axvline(0, color='gray', lw=0.5)
plt.legend()
plt.show()
```

Plot (Python)


Collection of points equidistant from A and B
$y = x - 2$
B (3,5)
A (7,1)
y
x
8
6
4
2
0
−2
−2
0
2
4
6
8
10
12