

## 2.3.16

EE25BTECH11013 - Bhargav

September 1, 2025

# Question

If  $\mathbf{p}$  is a unit vector and  $(\mathbf{x} - \mathbf{p}) \cdot (\mathbf{x} + \mathbf{p}) = 80$ , then find  $\|\mathbf{x}\|$ .

# Solution

We are given the equation:

$$(\mathbf{x} - \mathbf{p})^\top \cdot (\mathbf{x} + \mathbf{p}) = 80 \quad (1)$$

Expand the product:

$$\mathbf{x}^\top \mathbf{x} + \mathbf{x}^\top \mathbf{p} - \mathbf{p}^\top \mathbf{x} - \mathbf{p}^\top \mathbf{p} = 80 \quad (2)$$

Since the product is commutative ( $\mathbf{x}^\top \cdot \mathbf{p} = \mathbf{p}^\top \cdot \mathbf{x}$ ), the middle terms cancel out:

$$\mathbf{x}^\top \cdot \mathbf{x} - \mathbf{p}^\top \cdot \mathbf{p} = 80 \quad (3)$$

# Final Calculation

Since  $\mathbf{v}^T \cdot \mathbf{v} = \|\mathbf{v}\|^2$ .

$$\|\mathbf{x}\|^2 - \|\mathbf{p}\|^2 = 80 \quad (4)$$

We are given that  $\mathbf{p}$  is a **unit vector**, so its magnitude is 1.

$$\|\mathbf{p}\| = 1 \implies \|\mathbf{p}\|^2 = 1 \quad (5)$$

Substituting this value into the equation:

$$\|\mathbf{x}\|^2 - 1 = 80 \quad (6)$$

$$\|\mathbf{x}\|^2 = 81 \quad (7)$$

$$\|\mathbf{x}\| = 9 \quad (8)$$

**Therefore, the magnitude of vector  $\mathbf{x}$  is 9.**

# Verification Example

The theoretical solution can be verified by example.

Assume that  $\mathbf{p}$  is the unit vector  $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ .

Then from the code we get a possible vector  $\mathbf{x}$  would be  $\begin{pmatrix} 9 \\ 0 \end{pmatrix}$ .

The magnitude of the  $\mathbf{x}$  is verified to be 9.

```
#include <stdio.h>
#include <math.h>
void find_magnitude(double *x, double *x_norm) {
    double p[2] = {1.0, 0.0};
    double given_value = 80.0;
    double p_norm_sq = p[0]*p[0] + p[1]*p[1];
    double x_norm_sq = given_value + p_norm_sq;
    *x_norm = sqrt(x_norm_sq);
    x[0] = *x_norm;
    x[1] = 0.0;
}
```

```
import ctypes
import numpy as np
lib = ctypes.CDLL("./libmagnitude.so")
lib.find_magnitude.argtypes = [np.ctypeslib.ndpointer(dtype=np.float64,
    ndim=1, flags="C"), ctypes.POINTER(ctypes.c_double)]
lib.find_magnitude.restype = None
x = np.zeros(2, dtype=np.float64)
x_norm = ctypes.c_double()
lib.find_magnitude(x, ctypes.byref(x_norm))
print("Result from C:")
print("x =", x)
print("||x|| =", x_norm.value)
p = np.array([1.0, 0.0])
lhs = np.dot(x - p, x + p)
print("(x - p)^T (x + p) =", lhs)
print("||x|| =", np.linalg.norm(x))
```

```
import numpy as np
p = np.array([1, 0])
given_value = 80
p_norm_sq = np.dot(p, p)
x_norm_sq = given_value + p_norm_sq
x_norm = np.sqrt(x_norm_sq)
x = np.array([x_norm, 0])
lhs = np.dot(x - p, x + p)
print("||p||^2 =", p_norm_sq)
print("||x|| =", x_norm)
print("Example x =", x)
print("Verification (x - p)^T (x + p) =", lhs)
```