8.2.3

Naman Kumar-EE25BTECH11041

1 Oct,2025

Question)

$$y^2 = -8x \tag{1}$$

Since it is a parabola we have

Eccentricity	e=1
Eigenvalue	$\lambda_1 = 0$
Determinant	$\|V\| = 0$

General equation of conic

$$\mathbf{x}^\mathbf{T}\mathbf{V}\mathbf{x} + 2\mathbf{u}^\mathbf{T}\mathbf{x} + f = 0 \tag{2}$$

Solution

For the equation (1), we can write

$$\mathbf{x}^{\mathbf{T}} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \mathbf{x} + 2 \begin{pmatrix} 4 \\ 0 \end{pmatrix}^T \mathbf{x} = 0 \tag{3}$$

$\mathbf{V}$	$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$
$\mathbf{u}$	$4e_1$
f	0

Solution

using general equations we know for any conic

$$\mathbf{V} = \|\mathbf{n}\|^2\mathbf{I} - e^2\mathbf{n}\mathbf{n}^T \tag{4}$$

$$\mathbf{u} = ce^2\mathbf{n} - \|\mathbf{n}\|^2\mathbf{F} \tag{5}$$

$$f = \|\mathbf{n}\|^2\|\mathbf{F}\|^2 - c^2e^2 \tag{6}$$

Let

$\mathbf{n}$	$\begin{pmatrix} a \\ b \end{pmatrix}$	normal vector to directrix
$\mathbf{F}$	$\begin{pmatrix} g \\ h \end{pmatrix}$	focus
c		be constant of directrix

Solution

Firstly in (4)

$$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{n}^T\mathbf{n}\mathbf{I} - (1)^2\mathbf{n}\mathbf{n}^T \tag{7}$$

$$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} a^2+b^2 & 0 \\ 0 & a^2+b^2 \end{pmatrix} - \begin{pmatrix} a^2 & ab \\ ab & b^2 \end{pmatrix} \tag{8}$$

$$\mathbf{n} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \mathbf{e_1} \tag{9}$$

Solution

In (5)

$$4\mathbf{e_1} = c(1)^2\mathbf{e_1} - (1)^2\begin{pmatrix} g \\ h \end{pmatrix} \tag{10}$$

$$\begin{pmatrix} g \\ h \end{pmatrix} = (c-4)\mathbf{e_1} = (c-4)\begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{11}$$

$$\mathbf{F} = \begin{pmatrix} c-4 \\ 0 \end{pmatrix} \tag{12}$$

Solution

In (6)

$$0 = (1)^2 \begin{pmatrix} c-4 \\ 0 \end{pmatrix}^T \begin{pmatrix} c-4 \\ 0 \end{pmatrix} - c^2(1) \tag{13}$$

$$c^2 + 16 - 8c - c^2 = 0 \tag{14}$$

$$c = 2 \tag{15}$$

$$\mathbf{F} = -2\mathbf{e_1} \tag{16}$$

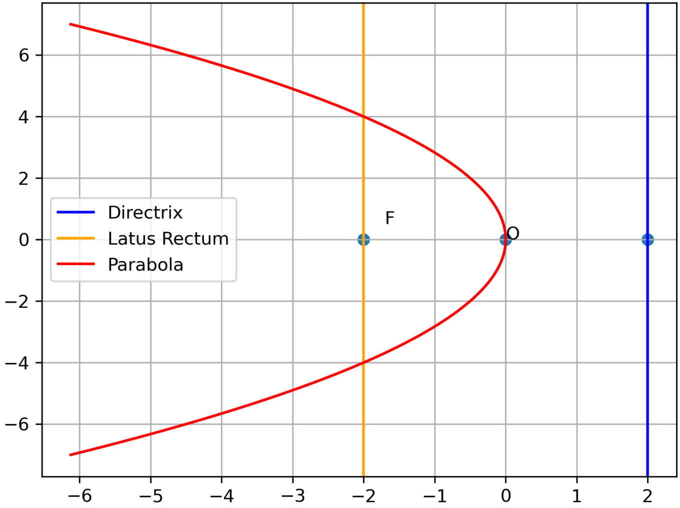
Solution

Directrix is

$$\mathbf{n}^{\mathbf{T}}\mathbf{x} = c \tag{17}$$

$$\mathbf{e_1}^T\mathbf{x} = 2 \tag{18}$$

Figure



Direct Python

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

y = np.linspace(-7,7,200)
x = -y*y/8

plt.axvline(x=2, color='blue', label="Directrix")
plt.axvline(x=-2, color='orange', label="Latus Rectum")

xp = np.array([0,2,-2])
yp = np.array([0,0,0])
plt.scatter(xp,yp)
```

Direct Python

```
plt.annotate('O', xy=(0, 0))
plt.annotate('F', xy=(-2, 0), xytext=(-1.7,0.5))

plt.plot(x,y, label="Parabola", color='red')
plt.grid()
plt.legend()
plt.savefig("figure.png", dpi=300)
plt.show()
```

C code

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

int generate_points(double x[], double y[], int n) {
    double xmin = -10.0; // start of x range
    double xmax = 0.0; // parabola is defined for x <= 0
    double step = (xmax - xmin) / (n/2); // half because we store
        y

    int idx = 0;
```

C code

```c
    for (int i = 0; i <= n/2 && idx < n-1; i++) {
        double xval = xmin + i * step;
        double yval = sqrt(-8.0 * xval);

        x[idx] = xval;
        y[idx] = yval;
        idx++;

        x[idx] = xval;
        y[idx] = -yval;
        idx++;
    }

    return idx;
}
```

Python code with shared object

```
# main.py
import ctypes
import numpy as np
import matplotlib.pyplot as plt

# Load the shared object
lib = ctypes.CDLL("./libparabola.so")

# Define function prototype
lib.generate_points.argtypes = [
    np.ctypeslib.ndpointer(dtype=np.float64, ndim=1, flags="
        C_CONTIGUOUS"),
    np.ctypeslib.ndpointer(dtype=np.float64, ndim=1, flags="
        C_CONTIGUOUS"),
    ctypes.c_int
]
lib.generate_points.restype = ctypes.c_int
```

Python code with shared object

```python
# Allocate arrays
n = 4000
x = np.zeros(n, dtype=np.float64)
y = np.zeros(n, dtype=np.float64)

# Call the C function
count = lib.generate_points(x, y, n)

# Slice to actual filled points
x = x[:count]
y = y[:count]
```

Python code with shared object

```
# Plot parabola
plt.figure(figsize=(6,6))
plt.scatter(x, y, s=5, c='b', label=r"$y^2=-8x$")
plt.axhline(0, color='k', linewidth=0.8)
plt.axvline(0, color='k', linewidth=0.8)
plt.xlabel("x-axis")
plt.ylabel("y-axis")
plt.title("Parabola: $y^2 = -8x$")
plt.legend()
plt.grid(True)
plt.axis("equal")
plt.show()
```