10.4.1

Vishwambhar - EE25BTECH11025

12th october, 2025

Question

Find the equations of the tangent and the normal, to the curve $16x^2 + 9y^2 = 145$ at the point (x_1, y_1), where $x_1 = 2$ and $y_1 > 0$.

Let

Let the point of contact of tangent and conic be $\mathbf{q}$ and also point of intersection of normal and conic be $\mathbf{q}$.
Given

$$\mathbf{q} = \begin{pmatrix} 2 \\ k \end{pmatrix} ; k > 0 \tag{1}$$

Ellipse

Let the equation of given ellipse in quadratic form be:

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

$$\tag{3}$$

where,

$$V = \begin{pmatrix} \frac{16}{145} & 0 \\ 0 & \frac{9}{145} \end{pmatrix} \tag{4}$$

$$\mathbf{u} = \begin{pmatrix} 0//0 \end{pmatrix} f = -1 \tag{5}$$

Tangent

Since $\mathbf{q}$ lies on the ellipse:

$$\mathbf{q}^\top V\mathbf{q} + f = 0 \tag{6}$$

$$\mathbf{q} = \begin{pmatrix} 2 \\ 3 \end{pmatrix} \tag{7}$$

The tangent equation can be given by

$$(V\mathbf{q} + \mathbf{u})^\top \mathbf{x} + \mathbf{u}^\top \mathbf{q} + f = 0 \tag{8}$$

Normal

The normal equation can be given by

$$\left(V\mathbf{q}+\mathbf{u}\right)^{\top}R\left(\mathbf{x}-\mathbf{q}\right)=0 \tag{9}$$

$$R=\begin{pmatrix}0 & -1\\ 1 & 0\end{pmatrix} \tag{10}$$

After substituting values we get tangent equation in normal form as:

$$\begin{pmatrix}\frac{32}{145}\\ \frac{27}{145}\end{pmatrix}^{\top}\mathbf{x}-1=0 \tag{11}$$

Cocnlusion

After substituting values we get normal equation in normal form as:

$$\begin{pmatrix} \frac{-27}{145} \\ \frac{32}{145} \end{pmatrix}^{\top} \mathbf{x} - \frac{42}{145} = 0 \tag{12}$$

C Code

```
#include <stdio.h>
void give_data(double *A, double *u, double *c, double *p, double
     *m, double *points){
    A[0] = 16.0/145.0;
    A[1] = 0;
    A[2] = 0;
    A[3] = 9.0/145.0;
    u[0] = 0;
    u[1] = 0;
    c[0] = -1;
    p[0] = 2;
    p[1] = 3;
    m[0] = 32 * 2;
    m[1] = 18 * 3;
    points[0] = A[0];
    points[1] = A[3];
    points[2] = c[0];
    points[3] = p[0];
    points[4] = p[1];
```

Python code 1

```
import ctypes as ct
import numpy as np
from numpy.lib import scimath as np_scimath
lib = ct.CDLL("./problem.so")
lib.give_data.argtypes = [
    ct.POINTER(ct.c_double), ct.POINTER(ct.c_double),
    ct.POINTER(ct.c_double), ct.POINTER(ct.c_double),
    ct.POINTER(ct.c_double), ct.POINTER(ct.c_double)
]
pointsA = ct.c_double * 4
pointsu = ct.c_double * 2
pointsc = ct.c_double * 1
pointsp = ct.c_double * 2
pointsm = ct.c_double * 2
points = ct.c_double * 5
```

Python code 1

```
A = pointsA()
u = pointsu()
c = pointsc()
p = pointsp()
m = pointsm()
data = points()
lib.give_data(A, u, c, p, m, data)
A = np.array([[A[0], A[1]], [A[2], A[3]]])
u = np.array([[u[0]], [u[1]]])
p = np.array([[p[0]], [p[1]]])
m = np.array([[m[0]], [m[1]]])
c = c[0]
```

Python code 1

```
a1 = float(m.T @ A @ m)
b1 = float(2 * (p.T @ A @ m + u.T @ m))
c1 = float(p.T @ A @ p + 2 * u.T @ p + c)
D = b1**2 - 4 * a1 * c1
t1 = (-b1 + np_scimath.sqrt(D)) / (2 * a1)
t2 = (-b1 - np_scimath.sqrt(D)) / (2 * a1)
A_point = p + t1 * m
B_point = p + t2 * m
def send_data():
    return (data, float(A_point[0]), float(A_point[1]), float(
        B_point[0]), float(B_point[1]))
```

Python code 2

```
import matplotlib.pyplot as plt
from call import send_data
import numpy as np
data, Ax, Ay, Bx, By = send_data()
x = np.linspace(-3.010, 3.010, 2500)
y = np.sqrt(145/9-(16*x**2)/9)
Xt = np.linspace(-5, 5, 100)
Yt = (145/27) * (1 - (32/145)*Xt)
Xn = np.linspace(-5, 5, 100)
Yn = (54/64) * (Xn - 2) + 3
```

Python code 2

```
plt.plot(x, y, "r")
plt.plot(x, -y, "r")
plt.plot(Xt, Yt, "g")
plt.plot(Xn, Yn, "b--")

plt.plot(2, 3, "ko")
plt.text(2.1, 3.1, "(2,3)", color="black")
plt.text(2.36, -2.77, r'$16x^2+9y^2=145$', color = "black")
plt.text(-3.04, 9.18, r'$32x+27y=145$', color="black")
plt.text(4.35,4.95,r'$-27x+32y=42$', color="black")
```

Python code 2

```
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.axis("equal")
plt.grid(True)
plt.savefig("../figs/plot.png")
plt.show()
```

Plot

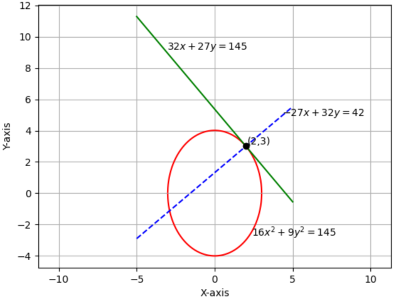

Figure: Plot of the ellipse, tangent and normal