Weekly Assignment 4

RC

‘r Sys.Date()

```
import sympy as sy
import sympy.vector as sv

N = sv.CoordSys3D('N')
t = sy.Symbol('t', real = True)
```

Question 1

Curvature of a circle with radius of 3 cm

- $r(t)$ curve
- $T = \frac{r'(t)}{||r'(t)||}$ normalized tangent vector
- $\kappa = ||\frac{dT}{ds}|| = \frac{||T'(t)||}{||r'(t)||} = \frac{||r' \times r''||}{||r'||^3}$ Curvature

```
r = 3*sy.cos(t)*N.i + 3*sy.sin(t)*N.j

r_prime = sy.diff(r)

T = r_prime / r_prime.magnitude().trigsimp()

T_prime = sy.diff(T)

k = T_prime.magnitude() / r_prime.magnitude()

print(f"curvature = {k.trigsimp()} cm^-1")

# alternate method
```

```
## curvature = 1/3 cm^-1
```

```
r_prime_prime = sy.diff(r_prime)

k = r_prime.cross(r_prime_prime).magnitude() / r_prime.magnitude()**3

print(f"curvature = {k.trigsimp()} cm^-1")
```

```
## curvature = 1/3 cm^-1
```

Question 2

Unit tangent vector and principle unit normal vector at $t = \pi$

```
r = sy.cos(3*t)*N.i + 2*sy.sin(3*t)*N.j + N.k
```

```python
r_prime = sy.diff(r)

UT = r_prime / r_prime.magnitude().trigsimp()

PUN = sy.diff(UT) / sy.diff(UT).magnitude()

print(f"T(t) = {UT}")
```

```
## T(t) = (-3*sin(3*t)/sqrt(36 - 27*sin(3*t)**2))*N.i + (6*cos(3*t)/sqrt(36 - 27*sin(3*t)**2))*N.j
```

```python
print(f"T(pi) = {UT.subs(t, sy.pi)}")
```

```
## T(pi) = (-1)*N.j
```

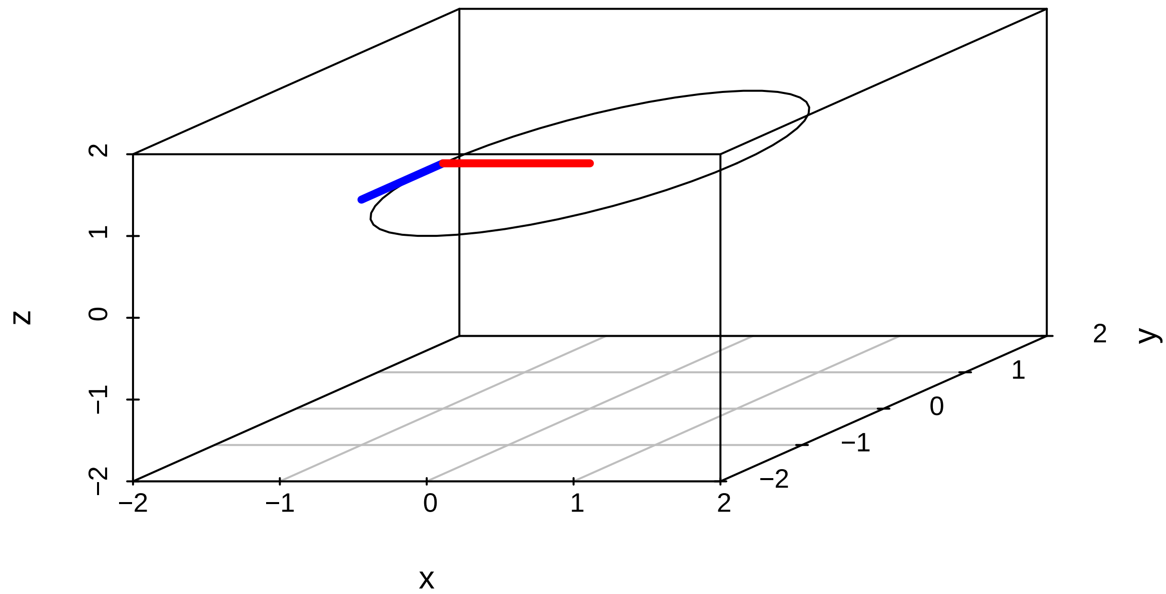
```python
print(f"N(t) = {PUN.simplify()}")
```

```
## N(t) = (-2*cos(3*t)/(sqrt(-1/(3*sin(3*t)**2 - 4)**3)*(3*sin(3*t)**2 - 4)**2))*N.i + (-sin(3*t)/(sqrt
```

```python
print(f"N(pi) = {PUN.subs(t, sy.pi)}")
```

```
## N(pi) = N.i
```

```r
f <- function(t) c(cos(3*t), 2*sin(3*t), 1)
X <- as.data.frame(t(sapply(seq(0, 2*pi/3, length=51), f)))
names(X) <- c("x", "y", "z")
sp1 <- scatterplot3d::scatterplot3d(X$x, X$y, X$z, xlim = c(-2,2), ylim=c(-2,2), zlim=c(-2,2),
                                    xlab="x", ylab="y", zlab="z", type="l")
UT <- function(t) f(pi) + t*c(0, -1, 0)
Y <- as.data.frame(t(sapply(seq(0, 1, length=51), UT)))
sp1$points3d(Y$V1, Y$V2, Y$V3, type="l", col="blue", lwd=4)
PUN <- function(t) f(pi) + t*c(1, 0, 0)
Z <- as.data.frame(t(sapply(seq(0, 1, length=51), PUN)))
sp1$points3d(Z$V1, Z$V2, Z$V3, type="l", col="red", lwd=4)
```



Question 3

Tangential and normal components of acceleration at $t = 0$

```
r = sy.exp(t)*sy.cos(2*t)*N.i + sy.exp(t)*sy.sin(2*t)*N.j + sy.exp(t)*N.k

r_prime = sy.diff(r)

r_prime_prime = sy.diff(r_prime)

UT = r_prime / r_prime.magnitude().trigsimp()

PUN = sy.diff(UT) / sy.diff(UT).magnitude()

acc = r_prime_prime.subs(t, 0)

tang_comp = r_prime_prime.subs(t, 0).dot(UT.subs(t, 0)) * UT.subs(t, 0)

norm_comp = r_prime_prime.subs(t, 0).dot(PUN.subs(t, 0)) * PUN.subs(t, 0)

# check
acc.magnitude() - sy.sqrt(tang_comp.dot(tang_comp) + norm_comp.dot(norm_comp)) == 0
```

```
## True
```

```
print(f"r''(0) = {acc}")
```

```
## r''(0) = (-3)*N.i + 4*N.j + N.k
```

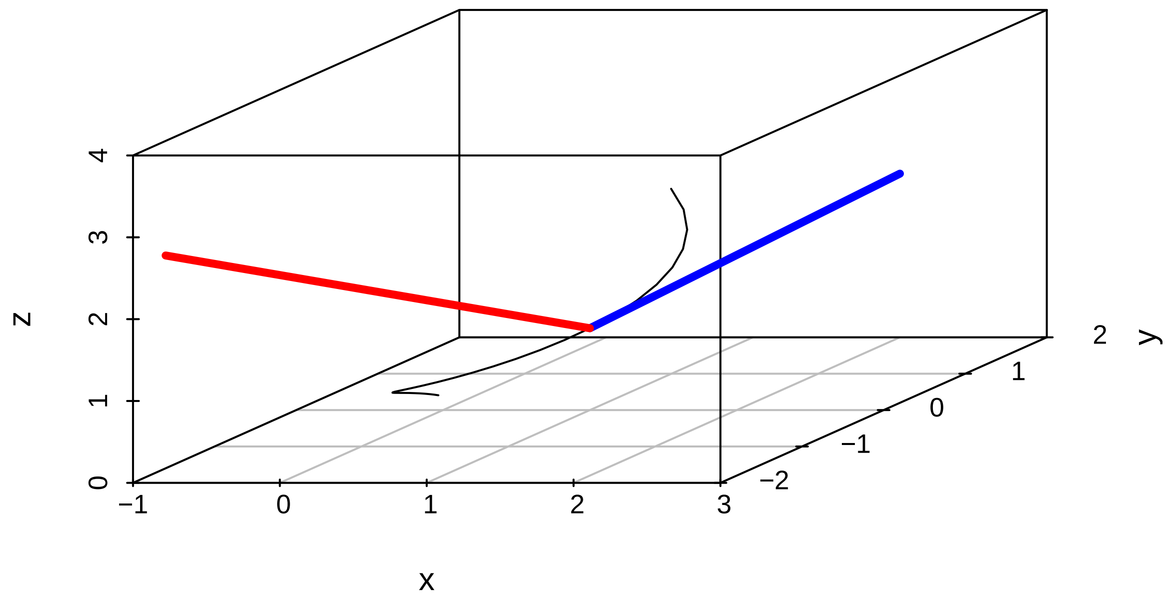
```
print(f"tangential component of acceleration at (t=0) = {tang_comp}")
```

```
## tangential component of acceleration at (t=0) = N.i + 2*N.j + N.k
```

```
print(f"normal component of acceleration at (t=0) = {norm_comp}")
```

```
## normal component of acceleration at (t=0) = (-4)*N.i + 2*N.j
```

```
f <- function(t) c(exp(t)*cos(2*t), exp(t)*sin(2*t), exp(t))
X <- as.data.frame(t(sapply(seq(-2, 2, length=51), f)))
names(X) <- c("x", "y", "z")
sp1 <- scatterplot3d::scatterplot3d(X$x, X$y, X$z, xlim = c(-1,3), ylim=c(-2,2), zlim=c(0,4),
                              xlab="x", ylab="y", zlab="z", type="l")
UT <- function(t) f(0) + t*c(1, 2, 1)
Y <- as.data.frame(t(sapply(seq(0, 1, length=51), UT)))
sp1$points3d(Y$V1, Y$V2, Y$V3, type="l", col="blue", lwd=4)
PUN <- function(t) f(0) + t*c(-4, 2, 0)
Z <- as.data.frame(t(sapply(seq(0, 1, length=51), PUN)))
sp1$points3d(Z$V1, Z$V2, Z$V3, type="l", col="red", lwd=4)
```



Question 4

Tangent vector, normal vector, and binormal vector at $t = \frac{\pi}{3}$

```
r = 4*sy.cos(t)*N.i + 4*sy.sin(t)*N.j + 2*t*N.k
```

```python
r_prime = sy.diff(r)

UT = r_prime / r_prime.magnitude().trigsimp()

PUN = sy.diff(UT) / sy.diff(UT).magnitude()

B = UT.cross(PUN)

print(f"T(t) = {UT}")
```

```
## T(t) = (-2*sqrt(5)*sin(t)/5)*N.i + (2*sqrt(5)*cos(t)/5)*N.j + (sqrt(5)/5)*N.k
```

```python
print(f"T(pi/3) = {UT.subs(t, sy.pi / 3)}")
```

```
## T(pi/3) = (-sqrt(15)/5)*N.i + (sqrt(5)/5)*N.j + (sqrt(5)/5)*N.k
```

```python
print(f"N(t) = {PUN.simplify()}")
```

```
## N(t) = (-cos(t))*N.i + (-sin(t))*N.j
```

```python
print(f"N(pi/3) = {PUN.subs(t, sy.pi / 3)}")
```

```
## N(pi/3) = (-1/2)*N.i + (-sqrt(3)/2)*N.j
```

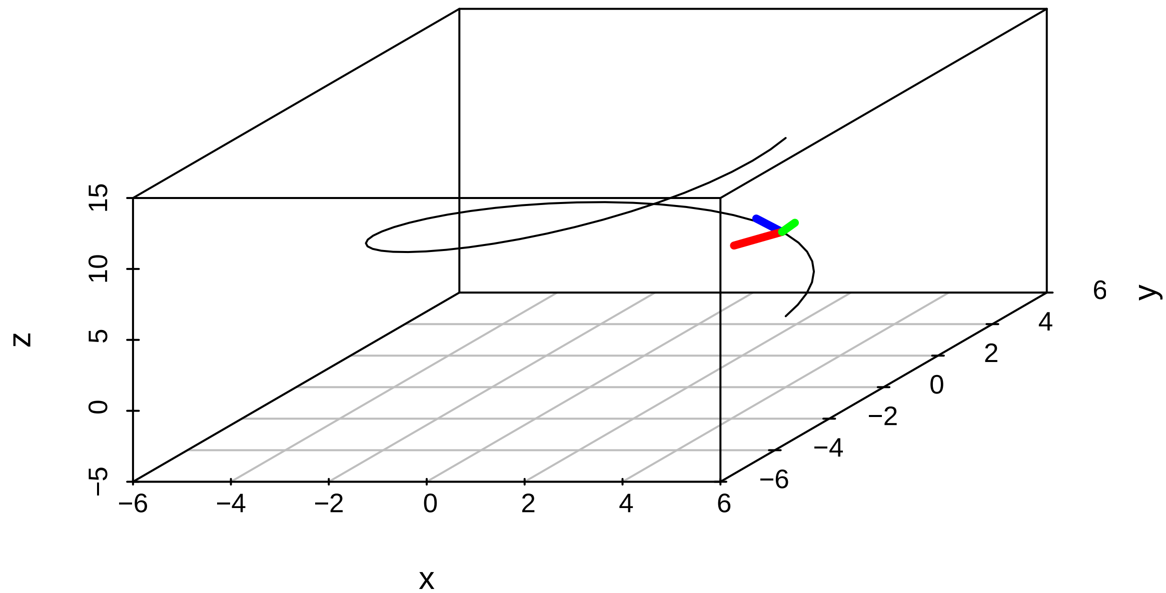
```python
print(f"B(t) = {B.trigsimp()}")
```

```
## B(t) = (sqrt(5)*sin(t)/5)*N.i + (-sqrt(5)*cos(t)/5)*N.j + (2*sqrt(5)/5)*N.k
```

```python
print(f"B(pi/3) = {B.subs(t, sy.pi / 3)}")
```

```
## B(pi/3) = (sqrt(15)/10)*N.i + (-sqrt(5)/10)*N.j + (2*sqrt(5)/5)*N.k
```

```r
f <- function(t) c(4*cos(t), 4*sin(t), 2*t)
X <- as.data.frame(t(sapply(seq(0, 2*pi, length=51), f)))
names(X) <- c("x", "y", "z")
sp1 <- scatterplot3d::scatterplot3d(X$x, X$y, X$z, xlim = c(-5,5), ylim=c(-5,5), zlim=c(-1,15),
                             xlab="x", ylab="y", zlab="z", type="l")
UT <- function(t) f(pi/3) + t*c(-sqrt(15)/5, sqrt(5)/5, sqrt(5)/5)
Y <- as.data.frame(t(sapply(seq(0, 1, length=51), UT)))
sp1$points3d(Y$V1, Y$V2, Y$V3, type="l", col="blue", lwd=4)
PUN <- function(t) f(pi/3) + t*c(-0.5, -sqrt(3)/2, 0)
Z <- as.data.frame(t(sapply(seq(0, 1, length=51), PUN)))
sp1$points3d(Z$V1, Z$V2, Z$V3, type="l", col="red", lwd=4)
BUN <- function(t) f(pi/3) + t*c(sqrt(15)/10, -sqrt(5)/10, 2*sqrt(5)/5)
B <- as.data.frame(t(sapply(seq(0, 1, length=51), BUN)))
sp1$points3d(B$V1, B$V2, B$V3, type="l", col="green", lwd=4)
```



Question 5

Find the curvature of the space curve at $< 2, 4, 2 >$ which is $t = 2$

```
r = t*N.i + t**2*N.j + t**3/4*N.k

r_prime = sy.diff(r)

r_prime_prime = sy.diff(r_prime)

k = r_prime.cross(r_prime_prime).magnitude() / r_prime.magnitude()**3

print(f"k(t) = {k}")
```

```
## k(t) = sqrt(9*t**4/4 + 9*t**2/4 + 4)/(9*t**4/16 + 4*t**2 + 1)**(3/2)
```

```
print(f"k(2) = {k.subs(t, 2)}")
```

```
## k(2) = 7*sqrt(26)/676
```

Question 6

Calculate quantities at $t = 0$

```
r = sy.exp(2*t)*N.i + (sy.sin(2*t) - 4*t + t**2)*N.j + (sy.cos(t) + 3*t)*N.k

r_prime = sy.diff(r)
```

```
print(f"v(0) = r'(0) = {r_prime.subs(t, 0)}")
```

```
## v(0) = r'(0) = 2*N.i + (-2)*N.j + 3*N.k
```

```
r_prime_prime = sy.diff(r_prime)

print(f"a(0) = r''(0) = {r_prime_prime.subs(t, 0)}")
```

```
## a(0) = r''(0) = 4*N.i + 2*N.j + (-1)*N.k
```

```
UT = r_prime / r_prime.magnitude()
PUN = sy.diff(UT) / sy.diff(UT).magnitude()

print(f"T(0) = {UT.subs(t, 0)}")
```

```
## T(0) = (2*sqrt(17)/17)*N.i + (-2*sqrt(17)/17)*N.j + (3*sqrt(17)/17)*N.k
```

```
tang_comp = r_prime_prime.subs(t, 0).dot(UT.subs(t, 0)) * UT.subs(t, 0)

norm_comp = r_prime_prime.subs(t, 0).dot(PUN.subs(t, 0)) * PUN.subs(t, 0)

print(f"tangential component of acceleration at (t=0) = {tang_comp}")
```

```
## tangential component of acceleration at (t=0) = 2/17*N.i + (-2/17)*N.j + 3/17*N.k
```

```
print(f"normal component of acceleration at (t=0) = {norm_comp}")
```

```
## normal component of acceleration at (t=0) = 66/17*N.i + 36/17*N.j + (-20/17)*N.k
```

```
print(f"N(0) = {PUN.subs(t, 0)}")
```

```
## N(0) = (33*sqrt(1513)/1513)*N.i + (18*sqrt(1513)/1513)*N.j + (-10*sqrt(1513)/1513)*N.k
```

```
k = r_prime.cross(r_prime_prime).magnitude() / r_prime.magnitude()**3

print(f"k(0) = {k.subs(t, 0)}")

# check
```

```
## k(0) = 2*sqrt(1513)/289
```

```
r_prime_prime.subs(t,0).magnitude() - sy.sqrt(tang_comp.dot(tang_comp) + norm_comp.dot(norm_comp)) == 0
```

```
## True
```