Presentation5
PYTHON TO C

P.Srijith Reddy
EE19BTECH11041
IIT Hyderabad

September 17, 2019

Problem Statement

Let **A**,**B**,**C** be three unit vectors such that

$$\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = \frac{\sqrt{3}}{2}(\mathbf{B} + \mathbf{C}) \tag{2.1}$$

If **B** is not parallel to **C**,then find the angle between **A** and **B**.

Finding angle between **A** and **B**

Given that **A**,**B**,**C** are unit vectors and

$$\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = \frac{\sqrt{3}}{2}(\mathbf{B} + \mathbf{C}) \tag{3.1}$$

We know that

$$\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = (\mathbf{A}^T\mathbf{C})\mathbf{B} - (\mathbf{A}^T\mathbf{B})\mathbf{C} \tag{3.2}$$

PROOF

Let

$$\mathbf{A} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \begin{bmatrix} i & j & k \end{bmatrix} \tag{4.1}$$

$$\mathbf{B} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} \begin{bmatrix} i & j & k \end{bmatrix} \tag{4.2}$$

$$\mathbf{C} = \begin{bmatrix} c_x \\ c_y \\ c_z \end{bmatrix} \begin{bmatrix} i & j & k \end{bmatrix} \tag{4.3}$$

Now

$$\mathbf{B} \times \mathbf{C} = \begin{bmatrix} i & j & k \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{bmatrix} \tag{4.4}$$

$$\mathbf{B} \times \mathbf{C} = (b_y c_z - b_z c_y)i - (b_x c_z - b_z c_x)j + (b_x c_y - b_y c_x)k \tag{4.5}$$

Now

$$\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = \begin{bmatrix} i & j & k \\ a_x & a_y & a_z \\ b_y c_z - b_z c_y & -(b_x c_z - b_z c_x) & b_x c_y - b_y c_x \end{bmatrix} \tag{4.6}$$

Now expanding $\mathbf{A}\times(\mathbf{B} \times \mathbf{C})$ *and considering* $\mathbf{x}$ *component only*

$\implies \mathbf{i} : a_y(b_x c_y - b_y c_x) + a_z(b_x c_z - b_z c_x)$
$=(\mathrm{a}_y c_y + a_z c_z)b_x - (a_y b_y + a_z b_z)c_x$
$=(\mathrm{a}_y c_y + a_z c_z)b_x - (a_y b_y + a_z b_z)c_x + ((a_x c_x)b_x - (a_x b_x)c_x))$
$=(\mathrm{a}_x c_x + a_y c_y + a_z c_z)b_x - (a_x b_x + a_y b_y + a_z c_z)c_x$
$=\mathbf{i}((\mathrm{A}^T C)b_x - (A^T B)c_x)$

Similarly we can show for **j** and **k** components.

Hence

$$\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = (\mathbf{A}^T\mathbf{C})\mathbf{B} - (\mathbf{A}^T\mathbf{B})\mathbf{C} \tag{4.7}$$

Now using equation (3.1) and equation(3.2)

$$(\mathbf{A}^T\mathbf{C})\mathbf{B} - (\mathbf{A}^T\mathbf{B})\mathbf{C} = \frac{\sqrt{3}}{2}(\mathbf{B} + \mathbf{C}) \tag{4.8}$$

Rearranging the terms we get the following equation

$$(\mathbf{A}^T\mathbf{C} - \frac{\sqrt{3}}{2})\mathbf{B} = (\mathbf{A}^T\mathbf{B} + \frac{\sqrt{3}}{2})\mathbf{C} \tag{4.9}$$

Applying cross product on both sides with **B** we get

$$(\mathbf{A}^T\mathbf{C} - \frac{\sqrt{3}}{2})\mathbf{B} \times \mathbf{B} = (\mathbf{A}^T\mathbf{B} + \frac{\sqrt{3}}{2})\mathbf{C} \times \mathbf{B} \tag{4.10}$$

Since

$$\mathbf{B} \times \mathbf{B} = 0 \tag{4.11}$$

therefore

$$(\mathbf{A}^T\mathbf{B} + \frac{\sqrt{3}}{2})\mathbf{C} \times \mathbf{B} = 0 \tag{4.12}$$

Either

$$(\mathbf{A}^T\mathbf{B} + \frac{\sqrt{3}}{2}) = 0 \tag{4.13}$$

or

$$\mathbf{C} \times \mathbf{B} = 0 \tag{4.14}$$

But since it is given that **B** is not parallel to **C**,

$$\mathbf{C} \times \mathbf{B} \neq 0 \tag{4.15}$$

So,

$$(\mathbf{A}^T\mathbf{B} + \frac{\sqrt{3}}{2}) = 0 \tag{4.16}$$

implies

$$\mathbf{A}^T\mathbf{B} = -\frac{\sqrt{3}}{2} \tag{4.17}$$

$$\mathbf{A}^T\mathbf{B} = \|\mathbf{A}\|\|\mathbf{B}\|COS(\theta_1) \tag{4.18}$$

where θ_1 is angle between **A** and **B**. Since,

$$\|\mathbf{A}\| = \|\mathbf{B}\| = 1 \tag{4.19}$$

$$cos(\theta_1) = -\frac{\sqrt{3}}{2} \tag{4.20}$$

$$\theta_1 = \frac{5\pi}{6} \tag{4.21}$$

Similarly, we get equation(3.17)by substituting equation(3.8) in equation(3.4).

$$\mathbf{A}^T\mathbf{C} = \frac{\sqrt{3}}{2} \tag{4.22}$$

$$\mathbf{A}^T\mathbf{C} = \|\mathbf{A}\|\|\mathbf{C}\|COS(\theta_2) \tag{4.23}$$

where θ_2 is angle between **A** and **C**. Since,

$$\|\mathbf{A}\| = \|\mathbf{C}\| = 1 \tag{4.24}$$

$$cos(\theta_2) = \frac{\sqrt{3}}{2} \tag{4.25}$$

$$\theta_2 = \frac{\pi}{6} \tag{4.26}$$

code in python

```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
from funcs import *
import numpy as np
import math
m = -(np.sqrt(3))/2#=A.T@B
a=1#=np.linalg.norm(A)
b=1#=np.linalg.norm(B)
k=m/(a*b)
theta=math.acos(k)*180/np.pi
print (theta)
#defining lines : x(k) = A + k*l
O = np.array([0,0,0]).reshape((3,1))
l1 = np.genfromtxt("A.dat").reshape((3,1))
O = np.array([0,0,0]).reshape((3,1))
l2 = np.genfromtxt("C.dat").reshape((3,1))
```

code in python

```
#creating x,y for 3D plotting
len = 10
xx, yy = np.meshgrid(range(len), range(len))
#setting up plot
fig = plt.figure()
ax = fig.add_subplot(111,projection='3d')
O = np.array([0,0,0]).reshape((3,1))
l3= np.genfromtxt("B.dat").reshape((3,1))
A = np.genfromtxt("A.dat").reshape((3,1))
B = np.genfromtxt("B.dat").reshape((3,1))
C = np.genfromtxt("C.dat").reshape((3,1))
l1_p = line_dir_pt(l1,O)
l2_p = line_dir_pt(l2,O)
l3_p = line_dir_pt(l3,O)
z=xx*0
```

code in python

```
ax.scatter(O[0],O[1],O[2],'o',label="Point O")
ax.scatter(A[0],A[1],A[2],'o',label="Point A")
ax.scatter(C[0],C[1],C[2],'o',label="Point C")
ax.scatter(B[0],B[1],B[2],'o',label="Point B")
#ax.plot_surface(xx, yy, z, color='r',alpha=1)

#plotting line
plt.plot(l1_p[0,:],l1_p[1,:],l1_p[2,:],label="Line L1")
plt.plot(l2_p[0,:],l2_p[1,:],l2_p[2,:],label="Line L2")
plt.plot(l3_p[0,:],l3_p[1,:],l3_p[2,:],label="Line L3")

#corresponding z for planes
#plotting planes

#show plot
plt.xlabel('$x$');plt.ylabel('$y$')
plt.legend(loc='best');plt.grid()
plt.show()
```

code in c used for uploading

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "coeffs.h"
int main() //main function begins
{
double**B=createMat(3,1);
B[0][0]=-sqrt(3)/2;B[1][0]=0.5;B[2][0]=0;
printf("B=\n");savetxt(B,"B.dat",3,1);
print(B,3,1);
double**C=createMat(3,1);
C[0][0]=sqrt(3)/2;C[1][0]=0.5;C[2][0]=0;
printf("C=\n");savetxt(C,"C.dat",3,1);
```

code in c used for uploading

```
print(C,3,1);//Let P be parallel to A ,so P=C+qB, q comes out to be −1
double **P=linalg_sub(C, B, 3, 1);
printf("P=\n");
print(P,3,1);
double m=linalg_norm(P, 3);
double s=1/m;
double **A=linalg_scalmatmul(P, s,3, 1);
printf("A=\n");
print(A,3,1);savetxt(A,"A.dat",3,1);
return 0;
}
```

Now we are going to plot one set of vectors satisfying given condition and

$$\mathbf{A} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{5.1}$$

$$\mathbf{B} = \begin{pmatrix} -\sqrt{3}/2 \\ 1/2 \\ 0 \end{pmatrix} \tag{5.2}$$

$$\mathbf{C} = \begin{pmatrix} \sqrt{3}/2 \\ 1/2 \\ 0 \end{pmatrix} \tag{5.3}$$

satisfy the given condition.

Plot

The code in

https://github.com/SRIJITH01/Srijith/blob/master/presentation2.py

plots Fig. 1.

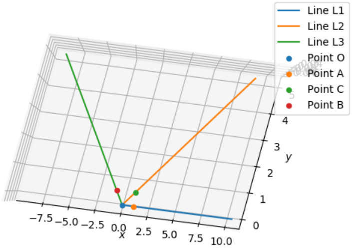

Figure: The above plot is for set of vectors staisfying given condition .