22111066
Vamshikiran Morlawar

CS724 - Assignment 1

Question 1

1) A = $\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$ What is C(A)?

   Vectors of A are linearly independent thus the column space of A is $\mathbf{R^2}$

2) B = $\begin{bmatrix}1 & 2\\2 & 4\end{bmatrix}$ What is C(B)?

   Vectors of B are linearly dependent as $v_1 = \begin{bmatrix}1\\2\end{bmatrix}$, $v_2 = \begin{bmatrix}2\\4\end{bmatrix}$ and we can see that $2v_1 = v_2$

   thus the column space of B is $\mathbf{R^1}$

3) D = $\begin{bmatrix}1 & 2 & 3\\0 & 0 & 4\end{bmatrix}$ What is C(D)?

   Two vectors of D are linearly dependent as $v_1 = \begin{bmatrix}1\\0\end{bmatrix}$, $v_2 = \begin{bmatrix}2\\0\end{bmatrix}$ and we can see that $2v_1 = v_2$

   thus the column space of D is $\mathbf{R^2}$

Question 2

a) We know user location coordinates (100, 100, 100), considering the 5 satellite locations at some points. We can get the distance between the points using the distance formula and as we know the signal from the satellite to the user has the speed of light i.e 3×10^8 m/s. So by using the Speed = Distance/Time we can get the time required for the signal to reach the user device.

   ```
   satellite_1_loc = np.array([120, 276, 9900])
   satellite_2_loc = np.array([20, 320, 9700])
   satellite_3_loc = np.array([50, 162, 10870])
   satellite_4_loc = np.array([90, 20, 10230])
   satellite_5_loc = np.array([160, 227, 10040])
   ```

   The time it takes for a signal to arrive from each one of these satellites to the user
   - Satellite 1 - 3.267200228534109e-05 s
   - Satellite 2 - 3.2009512475026 68e-05 s
   - Satellite 3 - 3.590098172597636e-05 s
   - Satellite 4 - 3.3767736080465925e-05 s
   - Satellite 5 - 3.31340591805136e-05 s

b) By using the linear equations with 5 satellites and time we got to verify the user location using the formula that we have derived ***$(A^TA)^{-1}A^Tb$***. We see that after the calculation we get the user location very close 100,100,100 i.e user location we've considered earlier.

$$(x-x_1)^2+(y-y_1)^2+(z-z_1)^2=r_1^2$$
$$(x-x_2)^2+(y-y_2)^2+(z-z_2)^2=r_2^2$$
$$(x-x_3)^2+(y-y_3)^2+(z-z_3)^2=r_3^2$$
$$(x-x_4)^2+(y-y_4)^2+(z-z_4)^2=r_4^2$$
$$(x-x_5)^2+(y-y_5)^2+(z-z_5)^2=r_5^2$$

$$A = \begin{bmatrix} 2(x_2-x_1) & 2(y_2-y_1) & 2(z_2-z_1) \\ 2(x_3-x_2) & 2(y_3-y_2) & 2(z_3-z_2) \\ 2(x_4-x_3) & 2(y_4-y_3) & 2(z_4-z_3) \\ 2(x_5-x_4) & 2(y_5-y_4) & 2(z_5-z_4) \end{bmatrix}$$

$$b = \begin{bmatrix} (r_1^2-r_2^2)-(x_1^2-x_2^2)-(y_1^2-y_2^2)-(z_1^2-z_2^2) \\ (r_2^2-r_3^2)-(x_2^2-x_3^2)-(y_2^2-y_3^2)-(z_2^2-z_3^2) \\ (r_3^2-r_4^2)-(x_3^2-x_4^2)-(y_3^2-y_4^2)-(z_3^2-z_4^2) \\ (r_4^2-r_5^2)-(x_4^2-x_5^2)-(y_4^2-y_5^2)-(z_4^2-z_5^2) \end{bmatrix}$$

Ax = b
$\hat{x} = (A^TA)^{-1}A^Tb$

Use the satellite locations and the times to find out the location of the user. Check whether it is coming exactly as (100,100,100)?

x - 100.00000000005934
y - 100.00000000007796
z - 100.00000000001299

c) Now we can add some random time errors to the times we got to check the impact in the localisation error. Let's add a time error ***e*** then the new distance that we will get be increased by ***e*speed_of_light***. And if we again use the linear equations using the formula we derived, we can find the coordinates of the user and find localisation errors that is calculate the distance between the actual user location and the calculated user location. A ***few nanoseconds*** increase in the time error has a ***1-3 meters*** error on localisation.

Check how much location inaccuracy it showing up?

time error - 2.8723355321197187e-09 s,
location inaccuracy - 0.8605003567173722 m

22111066
Vamshikiran Morlawar

d) Let's take multiple random time errors and using the same method as (c) calculate the corresponding localisation and plot the graph using a library like matplotlib to see the relation. Looking at the graph we see that its increasing plot i.e increase in time error increases the localisation error.