End Semester Examination

Name : Mohnish Devaraj

Subject Name: Data Science

Reg No: 39110636

Subject code: SCSA 3016

Roll No: 19S115398

Year : III Year

Date: 23rd May 2022

No. of pages: 6

360416003445017

(31) $B = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \end{bmatrix}$

Calculate the pseudo inverse for the matrix B.

Sol:

Given,

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \end{bmatrix}$$

Here all min ors of order two are zero

$\therefore$ rank(B) = 1

Since for the given matrix the rank is not equal to the number of rows or columns, therefore it has to be solved by singular value decomposition.

Then the pseudo inverse of A using SVD is $B^{+} = V\Sigma^{+}U^{T}$

$$= V\Sigma^{-1}U^{T}$$

$$B\cancel{A}^T = \begin{bmatrix} \cancel{2}1 & 2 \\ 2 & 4 \\ 3 & 6 \end{bmatrix}$$

$$B^T B = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \end{bmatrix} = \begin{bmatrix} 5 & 10 & 15 \\ 10 & 20 & 30 \\ 15 & 36 & 45 \end{bmatrix}$$

$$\det(\overset{B}{\cancel{A}} - \lambda I) = 0$$

$$\left[\begin{bmatrix} 5 & 10 & 15 \\ 10 & 20 & 30 \\ 15 & 30 & 45 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right] = 0$$

$$\begin{bmatrix} 5-\lambda & 10 & 15 \\ 10 & 20-\lambda & 30 \\ 15 & 30 & 45-\lambda \end{bmatrix} = 0$$

by solving, we get the eigen values as, $\lambda_1 = 70$, $\lambda_2 = 0$, $\lambda_3 = 0$.

$\therefore$ by substituting the values of λ, we get eigen vectors,

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} -2 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 \\ 6 \\ -5 \end{bmatrix}$$

The singular values of B are

$$\sigma_1 = \sqrt{\lambda_1} = \sqrt{70} \qquad \sigma_2 = \sqrt{\lambda_2} = \sqrt{0} \qquad \sigma_3 = \sqrt{\lambda_3} = \sqrt{0}$$

Normalized vectors are $v_1 = \begin{bmatrix} 1/\sqrt{14} \\ 2/\sqrt{14} \end{bmatrix}$ $v_2 = \begin{bmatrix} -2/\sqrt{5} \\ 1/\sqrt{5} \end{bmatrix}$ $v_3 = \begin{bmatrix} 3/\sqrt{70} \\ 6/\sqrt{70} \end{bmatrix}$

$$v_1 = \begin{bmatrix} 1/\sqrt{14} \\ 2/\sqrt{14} \\ 3/\sqrt{14} \end{bmatrix} \qquad v_2 = \begin{bmatrix} -2/\sqrt{5} \\ 1/\sqrt{5} \\ 0 \end{bmatrix} \qquad v_3 = \begin{bmatrix} 3/\sqrt{70} \\ 6/\sqrt{70} \\ -5/\sqrt{70} \end{bmatrix}$$

So, thus

$$V = \begin{bmatrix} 1/\sqrt{14} & -2/\sqrt{5} & 3/\sqrt{70} \\ 2/\sqrt{14} & 1/\sqrt{5} & 6/\sqrt{70} \\ 3/\sqrt{14} & 0 & -5/\sqrt{70} \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} \sqrt{70} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$u_1 = \frac{1}{\sigma_1} B v_1 = \begin{bmatrix} 1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix} \text{ and } u_2 = \frac{1}{\sigma_2} B v_2 = \begin{bmatrix} -2/\sqrt{5} \\ 1/\sqrt{5} \end{bmatrix}$$

$$U = \begin{bmatrix} 1/\sqrt{5} & -2/\sqrt{5} \\ 2/\sqrt{5} & 1/\sqrt{5} \end{bmatrix}$$

$$B^+ = V \Sigma^+ U^T$$

$$= \begin{bmatrix} 1/\sqrt{14} & -2/\sqrt{5} & 3/\sqrt{70} \\ 2/\sqrt{14} & 1/\sqrt{5} & 6/\sqrt{70} \\ 3/\sqrt{14} & 0 & -5/\sqrt{70} \end{bmatrix} \begin{bmatrix} 1/\sqrt{70} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{5} & 2/\sqrt{5} \\ -2/\sqrt{5} & 1/\sqrt{5} \end{bmatrix}$$

$$= \frac{1}{70} \begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \end{bmatrix}$$

(34) Write the python program for the following:

1) Find the binary search

2) Find common elements in python list using any one method.

Sol:

1) Algorithm:

i) The middle element is tested for the required element. If found, then its position is reported else the following test is made.

ii) If search element 'val' < 'middle' element, search the left half of the list else search the right half of the list

iii) Repeat step 1 and 2 on the selected half until the entry is found otherwise report failure

Program

```
def binary-search(list1, n):
    low = 0
    high = len (list1)-1
    mid = 0
    while low <= high:
        mid = (high + low)/2
        if list1 [mid] < n :
            low = mid +1
        elif list1 [mid] > n:
            high = mid -1
        else:
            return mid
    return -1

list1 = input ("Enter the list of numbers: ").split()
list1 = [int(x) for x in list1]
n = int (input ("Enter the search element: ")).

result = binary-search (list1, n)

if result != -1:
    print ("Element is present at index ", str(result))
else:
    print ("Element not found ")
```

Output :

```
Enter the list of numbers : 10 3 2 13 5 6
Enter the search element : 2
Element is present at index 2
```

2) Algorithm :

i, Given two list.

ii, Loop through two lists and get the match number using the if condition.

iii, if same number matches print the number, else display no common elements.

Using Set's & property:

Program:

```
def common-member (a,b):
    a_set = set(a)
    b-set = set(b)

    if (a_set & b_set):
        print(a_set & b_set)
    else:
        print("No common elements").

a=[1, 2, 3, 4, 5]

b=[5, 6, 7, 8, 9]

Common_member(a,b)

c=[1, 2, 3, 4, 5]

d=[6, 7, 8, 9]

Common_member(c,d)
```

Output:

```
{5}

No common elements.
```