Adamson University
College of Engineering
Computer Engineering Department

LINEAR ALGEBRA

Laboratory No. # 3

VECTOR OPERATIONS

Score

CRITERIA	Exceeds Expectations	Meets Expectations	Needs Improvement	Unsatisfactory
Functionality (60 points)				
Completeness (20 points)				
Structure (20 points)				

Remarks: ______________________________

____

Submitted by:
Manlulu, Emmanuel L.
TTh 7:00 – 10:00 / 58013

Submitted to
Engr. Maria Rizette Sayo
Instructor

Date Performed:
13/09/2023

Date Submitted
13/09/2023

Objective

1. Recall knowledge on vector operations while being familiar with new operations such as products.
2. Visualize vector operations.
3. Perform vector operations using Python.

Algorithm

1. Type the main title of this activity as "Vector Operations"
2. On your GitHub, create a repository name Linear Algebra 58019
3. On your Colab, name your activity as Python Exercise 3.ipynb and save a copy to your GitHub repository

Discussion

We have dealt with some of the vector operations in the last module, now we will dwell into more operations. In this laboratory, we will tackle addition, multiplication, division and the inner product of a vector.

Coding Activity 3

Vector Addition

We have encountered vector before especially with your last activity. Vector addition is simply the element-wise addition of the scalar values of the vectors. Let's take the following vectors as a sample:

$$A = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}, B = \begin{bmatrix} 31-2 \end{bmatrix}$$

So if do a vector addition of these two vectors we'll get:

$$A + B = \begin{bmatrix} 4 \\ 3 \\ -2 \end{bmatrix}$$

We can programmatically solve this using np.add() or simply using +.

```
A = np.array([1,2,0])
B = np.array([3,1,-2])
C = np.array([0,0,1])

A+B+C

np.add(A,np.add(B,C))

np.sum(B)
```

Vector Subtraction

Vector subtraction is similar to your vector addition but you would need to scale the second vector using a negative scalar that is usually -1. So if we subtract vector B from vector A we get:

$$A - B = \begin{bmatrix} -2 \\ 1 \\ 2 \end{bmatrix}$$

In Python, this can be achieved by using np.subtract() or –

```
np.subtract(A,B)

A-B
```

Vector Multiplication

Vector multiplication, like addition does its operations element-wise. So basic vector multiplication can be achieved by multiplying the elements or the scalars of the vectors individually. So:

$$A * B = \begin{bmatrix} 3 \\ 2 \\ 0 \end{bmatrix}$$

We can implement this in code by using np.multiply() or simply operating with *.

```
np.multiply(A,B)

A*B
```

Vector Division

Dividing a vector by another is uncommon, but if the situation specifies that each elements or scalars of the vector would be divided individually we can perform this by:

$$A./B = \begin{bmatrix} \frac{1}{3} \\ 2 \\ 0 \end{bmatrix}$$

Take note that the notation we used here is ./ for element-wise division, this notation is based in a MATLab notation for element-wise division. This can be achieved in Python using np.divide() or /.

```
np.divide(A,B)

A/B

C = np.array([3.9,1.8,7.7])
D = np.array([1,1,1])
C//D

np.array(A/B,dtype=int)
```

```
import numpy as np

# Vector Addition
A = np.array([1,2,0])
B = np.array([3,1,-2])
C = np.array([0,0,1])

A+B+C

np.add(A,np.add(B,C))

print("Sum: ", np.sum(B))
```

```
Sum:  2
```

```
#Vector Subtraction
A-B

print("Difference: ", np.subtract(A,B))
```

```
Difference:  [-2  1  2]
```

```
#Vector Multiplication
A*B

print("Product: ", A*B)
```

```
Product:  [3 2 0]
```

```
#Vector Division

A/B

C = np.array([3.9,1.8,7.7])
D = np.array([1,1,1])
C//D
```