Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 8

System of Linear Equations

Submitted by:

Columba, Lorenzo Miguel L.

Instructor:

Engr. Dylan Josh D. Lopez

December 21, 2020

I. Objectives

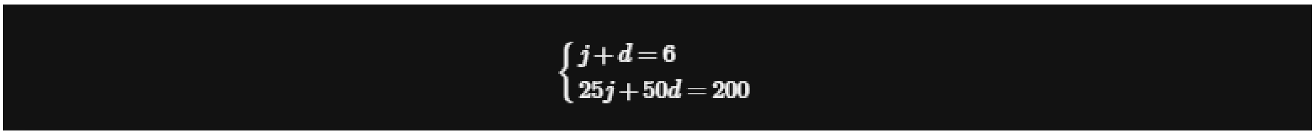
This laboratory activity aims to be familiarized in system of linear equations and solving using the various linear algebra techniques using Python programming.

II. Methods

The practice of this activity is to be familiar with how the system of linear equations works on Python programming and it implies the various techniques of linear algebra and how to solve the linear equation more quickly. The deliverables of this activity is to create a scenario or word problem that includes a system of linear equations and providing an output which was achieved by creating a matrix with the usage of NumPy's present functions to show how it solves rather than in manual computing. The NumPy's functions, I used is np.array for creating an array for matrix [1], followed by np.linalg.inv() to compute the multiplicative inverse of a matrix or inverse function with a dot product from the library of NumPy's present functions. [2] and Lastly, np.linalg.solve() to validate if the np.linalgb.inv() have the same result and to know what is their comparison. [3]

III. Results

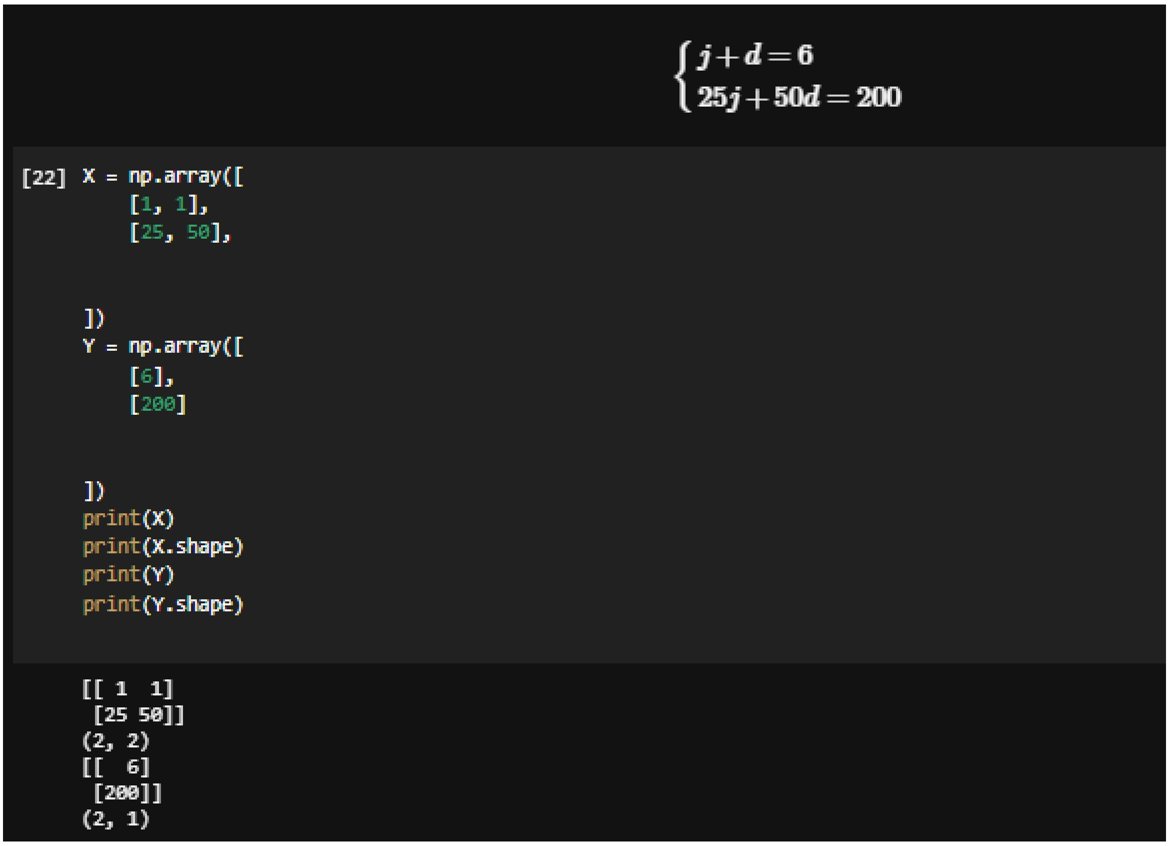
For this activity the instructor wants us to try and create a 2 minimum equations and solve them in using inverse equation.



$$\begin{cases} j + d = 6 \\ 25j + 50d = 200 \end{cases}$$

Figure 1: System of linear equation problems

Figure 1 shows the system of linear equation I created from the scenario or word problem which is “You’re going to the mall with your friends and you have $200 to spend from your recent birthday money. You discover a store that has all jeans for $25 and all dresses for $50. You really, really want to take home 6 items of clothing because you “need” that many new things. Wouldn’t it be clever to find out how many pairs of jeans and how many dresses you can buy so you use the whole $200 (tax not included – your parents promised to pay the tax)?”



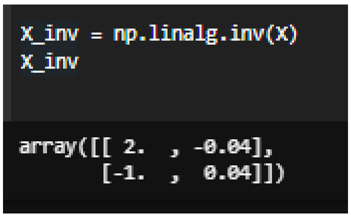
$$\begin{cases} j+d=6 \\ 25j+50d=200 \end{cases}$$

```
[22] X = np.array([
         [1, 1],
         [25, 50],


     ])
     Y = np.array([
         [6],
         [200]


     ])
     print(X)
     print(X.shape)
     print(Y)
     print(Y.shape)
```

```
     [[ 1  1]
      [25 50]]
     (2, 2)
     [[  6]
      [200]]
     (2, 1)
```

Figure 2: Code snippet of the word problem and linear equation.

Figure 2 shows the example of system of linear equation which in this block of code I create a 2 by 2 matrix and 2 by 1 matrix which explains the x and y equation. I can solve this by using the formula of the linear equation which is , to get how many jeans and dresses to bought, for jeans and dresses will be represented of x and y for items and money to spend.

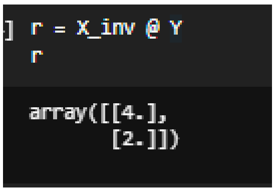


```
X_inv = np.linalg.inv(X)
X_inv
```

```
array([[ 2.  , -0.04],
       [-1.  ,  0.04]])
```

Figure 3: The inverse of X

In Figure 3 I use the function np.linalg.inv() to show the the inverse value of the x or the jeans and dresses

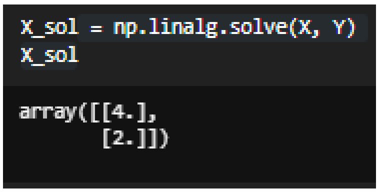


```
r = X_inv @ Y
r
```

```
array([[4.],
       [2.]])
```

Figure 4: The Answer in the problem using inverse

In figure 4 I use the dot product to solve the linear equation with the inverse and Y or the number of items and spent money.



```
X_sol = np.linalg.solve(X, Y)
X_sol
```

```
array([[4.],
       [2.]])
```

Figure 5: The code for solving the linear equation.

Figure 5 shows the other method and function in solving the system of linear equation using the NumPy’s present functions np.linalg.solve(). The reason why it has the same result with figure 3 is because this method is more precises which gives the exact solution for x which is well determined, full rank, linear equation as ax = b [3]. Also, sometimes in computing inverse equation np.linalgb.inv() will return error when it computes a large dense matrix [4].

IV. Conclusion

This laboratory helps me to understand how does system linear equation works in a real-life situation and familiarized the techniques and implementation in solving linearequations using various linear algebra techniques. Also, another usage of NumPy’s present functions using Python programming. I learned that this activity will help me to solve some practical problems and how does it make it easier than computing manually. Moreover, the essence of a system of linear equations is to understand the simulations and application into a real-life scenario it could be a geometric problem, it could make predictions, budget or knowing the price of an item, etc.[5]In solving the system of the linear equation it could be into 3 methods such as graphing, substitution, and elimination. This method determines which one you will choose in order to get your right answer and makes it easier for us programmers to compute such large scales of numbers or large dense matrix. [5]. Lastly, the system of linear equations is one of the fundamentals in the application of robotics that is used to create the concept of a robot that includes stability and design of the controllers which the sensors of the robots will be measured like in the arms or the camera of a robot.[6]

References

[1] W3Schools, (2020). About " NumPy Creating Arrays." Available:

https://www.w3schools.com/python/numpy_creating_arrays.asp Date Accesed: 12/14/2020

[2] The SciPy community, (2020). About " numpy.linalg.inv." Available:

https://numpy.org/doc/stable/reference/generated/numpy.linalg.inv.html Date Accesed: 12/14/2020

[3] The SciPy community, (2020). About " numpy.linalg.solve." Available:

https://numpy.org/doc/stable/reference/generated/numpy.linalg.solve.html Date Accesed: 12/14/2020

[4] GitHub, Inc, (2020). About " why using np.linalg.solve instead of np.linalg.inv?." Available:

https://github.com/ntucllab/striatum/issues/60 Date Accesed: 12/14/2020

[5] "Home," She Loves Math. [Online]. Available: https://www.shelovesmath.com/algebra/intermediate-algebra/systems-of-linear-equations/. [Accessed: 14-Dec-2020].

[6] [6] Luc Jaulin, (2020). About" Linear Systems." Available:

https://onlinelibrary.wiley.com/doi/10.1002/9781119081326.ch3 Date Accesed: 12/14/2020