Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 8

System of Linear Equation

Submitted by:
Joaquin, Marc Christopher C.

Instructor:
Engr. Dylan Josh D. Lopez

December 19, 2020

I. Objectives

This laboratory activity aims to implement the principles and techniques of system of linear equation. In addition, it aims to be familiarized to solve system of linear equations using various linear algebra techniques in Python Programmming.

II. Methods

The practices of this activity are first to apply system of linear equations in order to enhance the skills of the programmer with this topic. With this technique, it implies that this numpy library has a variety of functions that can be useful for solving ton of datas and it teaches the programmer on how this function works in a program. It will really help him to understand and definitely, he will apply that as well. Aside from that, the practice of this activity is to dig deeper for other functionalities of this system of linear equation. The deliverables of this activity is to solve all of the tasks using various linear algebra techniques and it must be done in Python Programming. With the techniques provided in this activity and real world example given, it will help the programmer to understand it easily so he will achieved the objective. Using Numpy library, it will lead the programmer to achieve it such as np.array() method, np.linalg.inv() method, and np.linalg.solve() method. The np.array() method is used to create array[1]. The np.linalg.inv() method computes the (multiplicative) inverse of a matrix[2]. At last, the np.linalg.solve() method used for solving a linear matrix equation, or system of linear scalar equations[3].

III. Results

The task given to the programmer is to have an example that applies the system of linear equation which has at least two questions and must apply as well the np.linalg.solve. The example will be shown in Figure 1.

The Joaqeco Family is looking for a Physical Therapist who can treat their *Padre de Pamilya* because he got stroke last week. They are in the middle class society who are in debts because of the hospital bills of their father so they need to look for PT with low budget rating deal. Dei, The patient's older son, has a friend named Paulo and they've been friends for so long since 4 years and Dei is aware that Paulo is a Physical Therapist too, who've been practicing it for 10 years. Dei asked Paulo regarding with the session deal for his father's therapy and Paulo offered him a 10 + 3 session bundle (8 provided excercises per session) for the therapy with just 10,000 Php only. Kriselle, the patient's daughter, found a Physical Therapist too online near to their place. The Physical Therapist offered her 10 session bundle (18 provided excercises per session), which is the normal session for therapy in stroke and it costs of 21,000 Php. Dei and Kriselle gathered ideas who will be the one to treat their father's condition. If they choose Paulo's offer, how much his rating per excercise in each session? How about the other Physical Therapist's offer if they would grab it? If they tried to ask for bonus deal like what Paulo's offer yet same price and numbers of excercise, Who has the low budget rating deal that they must choose?

Figure 1: Real Life Example

In this example given by the programmer, it is all about the rate difference of the two offers by the Physical Therapist for their father, who is a stroke patient. They must look for one as soon as possible in order to improve their father's condition with lower budget deal price.

LET

X = *Therapy session*

Y = *Number of Activities*

Paulo's Offer

13x + 4y = 10000

Physical Therapist 1

10x + y = 21000

Figure 2: Linear Equation

As the reader will see, the x serves as the number of therapy session and the y serves as the number of Activities per session. Paulo's offer to the Joaqeco Family, he has 13 session and every session, the therapy or the exercise will consume 4 hours, on the other hand, the physical therapist offers a 10 session and every session, the exercise activities will consume an hour.

```
#Money rating for the two therapist.
quantities = np.array([
    [13, 8 ],
    [10, 18],
])

cost = np.array([
    [10000],
    [21000],
])

prices = np.linalg.inv(quantities) @ cost
print('The rating per excercise of Paulo is: PHP {:.2f}'.format(float(prices[0])))
print('The rating per excercise of the Physical Therapist is: PHP {:.2f}'.format(float(prices[1])))
```

```
The rating per excercise of Paulo is: PHP 77.92
The hour rating per excercise of the Physical Therapist  is: PHP 1123.38
```

Figure 3: Input and output for the rating per excercise

In this figure, the reader will see that the values of the number of therapy and exercise combine in one array called *quantities* and the cost of each Physical Therapist is announced in an array called *cost*. As the result, the rating per exercise of Paulo's offer is PHP 77.92 while the Physical Therapist's offer regarding with rating per exercise is PHP 1123.38.

```
#What deal will the Joaqeco Family choose for less cost?
quantities = np.array([
    [13, 8 ],
    [13, 18],
])

cost = np.array([
    [10000],
    [21000],
])

prices = np.linalg.inv(quantities) @ cost
print('The rating per excercise of Paulo is: PHP {:.2f}'.format(float(prices[0])))
print('The hour rating per excercise of the Physical Therapist is: PHP {:.2f}'.format(float(prices[1])))
```

```
The rating per excercise of Paulo is: PHP 105.26
The hour rating per excercise of the Physical Therapist is: PHP 1078.95
```

Therefore, The Joaqueco Family has less cost for therapy to their father compare to the Physical Therapist.

Figure 4: Input and output for choosing what deal that has less cost.

The Joaqueco Family tried to ask to add more session yet same price and number of excercises still and when the Physical Therapist accept it, they try to compare the deal of Paulo and the other Physical Therapist and as the result, Paulo has less cost in terms of bundle deals for their father's situation.

```
#Using Linalg.solve.
linalg_solve = np.linalg.solve(quantities, cost)
print('The rating per excercise of Paulo is: PHP {:.2f}'.format(float(linalg_solve[0])))
print('The hour rating per excercise of the Physical Therapist is: PHP {:.2f}'.format(float(linalg_solve[1])))
```

```
The rating per excercise of Paulo is: PHP 105.26
The hour rating per excercise of the Physical Therapist is: PHP 1078.95
```

Figure 4: Using np.linalg.solve method

In this figure, it shows the second task given by this laboratory activity, which is to use the np.linalg.solve. The programmer noticed that when he used it, it is the same exact value of output compare to the np.linalg.solve. The difference between those two is first it is more convenient because the code is short and easily to understand while the np.linalg.inv is sort of short method and quite challenging to understand yet whether the programmer used either way, he will arrived with the same exact result.

IV. Conclusion

In this project, the programmer achieved the objective of this laboratory exercise,which is to be familiarized about the system of linear equation and how it works in a program. In addition, it helps him to dig deeper to understand the Numpy Library for in this activity are more on using that such as numpy.array, numpy.linalg.inv and numpy.linalg.solve. With this tasks given, the programmer gradually understand the essence of this library because with this, it helps the programmer to make his/her task much easier especially solving ton of datas. Nonetheless, The system of linear equation is a collection of two or more linear equations involving the same set of variables[4]. With numpy function, it supports two important operations in order to achieve the expected result, which are matrix inversion and matrix dot product. However, the system of linear equation can be used in robotics because in controlling it, a higher algorithm must be used. One of the steps in order to create robot is designing controller in order to describe the desired behavior of state space form. State space form requires three matrices and other more complex controllers have a forth matrix also [5]. With these matrices, it will then be used in a standardized design to implement the controller.

References

[1]*numpy.array — NumPy v1.19 Manual*. (n.d.). Retrieved December 19, 2020, from https://numpy.org/doc/stable/reference/generated/numpy.array.html

[2]*numpy.linalg.inv — NumPy v1.19 Manual*. (n.d.). Retrieved December 19, 2020, from https://numpy.org/doc/stable/reference/generated/numpy.linalg.inv.html

[3] *numpy.linalg.solve — NumPy v1.19 Manual*. (n.d.). Retrieved December 19, 2020, from https://numpy.org/doc/stable/reference/generated/numpy.linalg.solve.html

[4]*Solving Systems of Linear Equations with Python's Numpy*. (n.d.). Retrieved December 19, 2020, from https://stackabuse.com/solving-systems-of-linear-equations-with-pythons-numpy/

[5]*12.2.3. Linear Timeâ€“Invariant Control Systems — Robotics Programming Study Guide*. (n.d.). Retrieved December 19, 2020, from http://faculty.salina.k-state.edu/tim/robotics_sg/Control/controllers/linear.html