AN INTRODUCTION TO LINEAR ALGEBRA USING PYTHON

Summer 2021
Zoom Lecture: Tu: 2:00-3:35 p.m.
National Science Foundation (NSF) Center for Integrated Quantum Materials (CIQM), DMR -1231319
Dr. Steven L. Richardson (srichards22@comcast.net)
Professor Emeritus of Electrical Engineering, Department of Electrical and Computer Engineering, Howard University, Washington, DC
and
Faculty Associate in Applied Physics, John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA

PROBLEM SET I
(due May 25, 2021)

Problem 1

As discussed in lecture show that for the case of simultaneously solving a system of **two** linear equations that

$$y = \frac{b_2 a_{11} - b_1 a_{21}}{a_{11} a_{22} - a_{12} a_{21}} = \frac{\begin{vmatrix} a_{11} & b_1 \\ a_{21} & b_2 \end{vmatrix}}{|\mathrm{A}|}$$

Problem 2

As discussed in lecture we will study the case of simultaneously solving a system of **three** linear equations

$$a_{11}\, x + a_{12}\, y + a_{13}\, z = b_1 \tag{1}$$

$$a_{21}\, x + a_{22}\, y + a_{23}\, z = b_2 \tag{2}$$

$$a_{31}\, x + a_{32}\, y + a_{33}\, z = b_3 \tag{3}$$

Let us first multiply (1) by -a_{22} and then multiply (2) by a_{12}. You will get two new equations which you can combine to eliminate y. Now solve this new equation to obtain the following equation

$$x = \frac{a_{22}b_1 - a_{12}b_2}{a_{11}a_{22} - a_{12}a_{21}} + z\left[\frac{a_{23}a_{12} - a_{13}a_{22}}{a_{11}a_{22} - a_{12}a_{21}}\right] \tag{4}$$

Let us now multiply (1) by -a_{21} and then multiply (2) by a_{11}. You will get two new equations which you can combine to eliminate x. Now solve this new equation to obtain the following equation

$$y = \frac{a_{11}b_2 - a_{21}b_1}{a_{11}a_{22} - a_{12}a_{21}} - z\left[\frac{a_{23}a_{11} - a_{13}a_{21}}{a_{11}a_{22} - a_{12}a_{21}}\right] \tag{5}$$

Now place (4) and (5) into (3) and carefully solve for z. You should obtain the following result

$$z = \frac{-a_{31}a_{22}b_1 + a_{31}a_{12}b_2 - a_{11}a_{32}b_2 + a_{21}a_{32}b_1 + a_{11}a_{22}b_3 - a_{12}a_{21}b_3}{a_{11}a_{22}a_{33} + a_{13}a_{21}a_{32} + a_{12}a_{31}a_{23} - a_{13}a_{22}a_{31} - a_{12}a_{21}a_{33} - a_{11}a_{23}a_{32}} \tag{6}$$

Note that the denominator is what we have defined in lecture as the determinant of the 3 x 3 matrix **A**. Using this definition show that the numerator can also be expressed as a determinant involving the constants b_1, b_2, and b_3 so that we can write

$$z = \frac{\begin{vmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ a_{31} & a_{32} & b_3 \end{vmatrix}}{|\mathbf{A}|} \tag{7}$$

I leave it up to you to find x using a similar strategy. As a hint return to (1) and (2) and find the appropriate multipliers to eliminate y and find z = z(x). Repeat this process and find an equation for y = y(x), Take these two new equations and substitute them into (3) and then solve for x to show that

$$x = \frac{\begin{vmatrix} b_1 & a_{12} & a_{13} \\ b_2 & a_{22} & a_{23} \\ b_3 & a_{32} & a_{33} \end{vmatrix}}{|\mathbf{A}|} \tag{8}$$

Finally use a similar strategy to solve for y. As a hint return to (1) and (2) and find the appropriate multipliers to eliminate x and find x = x(y). Repeat this process and find an equation for z = z(y), Take these two new equations and substitute them into (3) and then solve for y to show that

$$y = \frac{\begin{vmatrix} a_{11} & b_1 & a_{13} \\ a_{21} & b_2 & a_{23} \\ a_{31} & b_3 & a_{33} \end{vmatrix}}{|\mathrm{A}|} \tag{9}$$

Problem 3

Use **Cramer's rule** to solve the following equations:

$$\begin{aligned} x &+ y &= 2 \\ 3x &- 2y &= 5 \end{aligned}$$

Problem 4

Use **Cramer's rule** to solve the following equations:

$$\begin{aligned} x + 2y + 3z &= -5 \\ -x - 3y + z &= -14 \\ 2x + y + z &= 1 \end{aligned}$$

Problem 5

Use **Cramer's rule** to solve the following equations:

$$\begin{aligned} x + y + z &= 2 \\ 2x - y - z &= 1 \\ x + 2y - z &= -3 \end{aligned}$$

Problem 6

Use **Cramer's rule** to solve the following equations:

$$\begin{aligned} x + y - z &= 2 \\ 2x - y + 3z &= 5 \\ 3x + 2y - 2z &= 5 \end{aligned}$$

Problem 7

Use **Cramer's rule** to solve the following equations:

$$\begin{array}{rcrcr} \boldsymbol{x} & + & \boldsymbol{2y} & = & \mathbf{3} \\ \boldsymbol{2x} & + & \boldsymbol{4y} & = & \mathbf{1} \end{array}$$

What goes wrong here? Why?

Python Exercise 1

Getting Started With Python

1. Google Anaconda Navigator which is a desktop graphical user interface which allows us to install applications and easily manage packages such as Python on your desktop or laptop computer.

2. Click on Install Anaconda which is highlighted on the first page.

3. You next have the option of installing Anaconda on either a Windows, MacOS X, or Linux platform.

4. You will install the free Individual Edition of Anaconda on your particular computer platform.

5. As of today (May 14, 2021), you will want to both download and install the 64-Bit Graphical Installer for Python 3.8 on your particular computer platform (Windows, MacOS X, or Linux). Please be patient as this takes a while to install all of the required package scripts.

6. Note that during the installation process you will be prompted to ask if you wish to download and install the PyCharm IDE package. You really do not need this package for the purpose of this course, but you might want it for future applications in data science and machine learning.

7. Activate the green Anaconda icon on your computer. In MacOS X, it is now located in the Applications folder.

8. We now see that the Anaconda Navigator is a very powerful tool which allows you to access tons of important applications. We only want one: click on the launch button for the Jupyter Notebook. Note that this is not a typo as it is really "Jupyter" and not the planet "Jupiter". The "py" comes from Python, a very powerful high-level general purpose programming language written over 30 years ago by Guido van Rossum.

9. You should see a Jupyter notebook with "Files","Running", and "Cluster" at the top left. Find the"New" button on the top right and click on it to see "Python 3". Click on "Python 3".

10. If you now see a Jupyter notebook page with a green input line (In [ ]), you now have successfully installed a Jupyter Notebook on your computer platform with access to Python 3. This is where we want to be!

11. Type in your first Python command as follows: introduction = "Welcome to Python ". Then click on the "Run" button at the top.

12. Next on a new input line type in the following: print (introduction). Press the "Run" button again and you should get the output: Welcome to Python 3. Congratulations, you have successfully used Python to run your first command!

13. We need to make some important editorial comments at this stage. You need to explore the many, many things that you can do in a Jupyter Notebook throughout this course. You can do this by yourself by trial and error and using the Web for more detailed instructions. You do not have to go out and purchase a book on "Jupyter Notebooks for Dummies"! You have all of this free information and technology at your fingertips, so please use it!

14. Next, this is not a course on Python, Anaconda, or Jupyter Notebooks! This is a course on learning linear algebra using some tools from Python with the Jupyter Notebook as the graphical interface for your computer platform.

15. Some of you may already be familiar with computer software packages like MATHEMATICA and MATLAB. Some of you may not. Note that they, like Python, also have Notebooks. In fact, MATLAB was explicitly written to learn matrix theory and linear algebra! So why are we not using these two packages in this course? This is a great question with a simple answer: Python is free, as MATLAB and MATHEMATICA are not! If you already have access to MATHEMATICA and MATLAB, then everything we can do in Python you should be able to do in those two languages. I would, however, recommend learning Python as it is a powerful, free language which is going to be with us for quite a while.

16. Some last comments are needed here. Again this is not a course in Python! You can do tons of interesting and important things in Python, but we are going to use a small number of tools from Python to help us master the concepts of linear algebra. In fact we are going to primarily use a number of modules from Python (e.g. NumPy and Matplotlib) to study linear algebra. These modules are so powerful that we will only use a few tools from them! We will learn the necessary tools from Python as we go along throughout the course. You will only learn Python by making mistakes and your Jupyter Notebook will allow you to do that without blowing up your computer! Linear algebra drives this course with Python as a very useful tool, not the other way around. You are encouraged to learn more and more about Python, Jupyter Notebooks, NumPy, Matplotlib, etc. as the course progresses and beyond. These are essential tools for future work in scientific computing, linear algebra, quantum computing, data science, and machine learning so live long and prosper. These are part of the armamentarium of programming tools for your future in STEM!

17. Enjoy Python and have fun!

1. Now let us first create a matrix in Python. We first need to import an essential module (or software package in Python) used throughout this course, NumPy.

On the first input line of your Jupyter Notebook type

import numpy as np # Everything after this asterisk is a comment. Here we rename NumPy as np for simplicity.

On the following input line type

A = np.array([[1,2,3], [4,5,6], [7,8,9]]) #this creates the 3 x 3 matrix called A

Now let us play with the following NumPy commands

print (A) # prints the matrix elements of A

print(A.shape) # tells you the dimension (number of rows and columns) of your matrix A

B =np.zeros((3,3)) #creates a 3 x 3 matrix whose matrix elements are all zeroes

print(B)

C= np.ones((1,2)) # create a 1 x 2 matrix whose matrix elements are all unity

print(C)

D = np.full((2,2), 7) # creates a matrix whose matrix elements are all a constant print(D)

F = np.random.random((2,2)) # create a matrix with random (actually pseudo-random) matrix elements

print(F)

2. To find the determinant of a matrix we use the mp.linalg.det function

import numpy as np

V = np.array([[6,1,1], [4, -2, 5], [2,8,7]])

print V # compare this with print(V)

np.linalg.det(V)

Now that you know how to find the determinant of a matrix in Python, let us go through some important properties of the determinant. We are just picking some particular examples here so these are not rigorous proofs but you get the idea. Rigorous proofs of these properties exist but they are beyond the scope of this course.

Given the matrix **A**

$$A = \begin{pmatrix} 3 & -1 & 2 \\ -2 & 1 & 1 \\ 1 & 4 & 3 \end{pmatrix}$$

Show that $|\mathbf{A}|$ is unchanged if we interchange the first row with the first column.

Take **A** and interchange any two rows or two columns. What happens to the determinant?

Take **A** and multiply each matrix element by some number. Show that the determinant of this new matrix is also multiplied by that number.

Show that $|\mathbf{A}|$ is unchanged if one row or column is added or subtracted to another.

Start with **A** and show that if any row or column is written as the sum or difference of two or more terms that the determinant of this new matrix can be written as the sum or difference of two or more determinants.

Invent a 3 x 3 matrix and show that its determinant vanishes if any two rows or columns are identical.

Now repeat all of the above determinant exercises for random matrices of order 3, 4, 5, and 10.

It goes without saying that you are not to use Python to solve any of the analytical problems in the first part of this problem set or any other problem set; otherwise, you will learn nothing in the course. It is tantamount to learning arithmetic in the first grade by simply using a calculator. Also feel free to experiment as you learn new Python commands.