

## AN INTRODUCTION TO LINEAR ALGEBRA USING PYTHON

Summer 2021

Zoom Lecture: Tu: 2:00-4:00 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 XIII (due Tuesday, August 17, 2021)

#### Problem 1

We showed in lecture that the projection  $\vec{p}$  of the vector  $\vec{b}$  on the two-dimensional plane in  $\mathbf{R}^3$  is given by

$$\vec{p} = A(A^T A)^{-1} A^T \vec{b} = P\vec{b}$$

where  $\vec{P}$  is the projection matrix. It looks like you should be able to rewrite this expression as

$$\vec{p} = AA^{-1}(A^T)^{-1}A^T \vec{b} = I\vec{b} = P\vec{b}$$

so that you would obtain

$$P = I.$$

What has gone awry?

#### Problem 2

(a) Write down the matrix which projects a vector in  $\mathbf{R}^3$  on the the xz-plane without performing any computations.

(b) Check you answer by finding a basis for this plane and use the results from Problem 2(a).

**Problem 3**

(a) Write down the matrix which projects a vector in  $\mathbf{R}^3$  on the the yz-plane without performing any computations.

(b) Check you answer by finding a basis for this plane and use the results from Problem 3(a).

**Problem 4**

(a) Find the projection matrix  $\mathbf{P}$  for projecting a vector in  $\mathbf{R}^3$  on to the plane

$$x - 4y + 2z = 0$$

(b) Now use this projection matrix  $\mathbf{P}$  to project the vector  $\vec{\mathbf{b}}$  on to this plane where

$$\vec{\mathbf{b}} = \begin{pmatrix} 1 \\ 0 \\ 4 \end{pmatrix}$$

**Problem 5**

(a) Find the projection matrix  $\mathbf{P}$  for projecting a vector in  $\mathbf{R}^3$  on to the plane

$$x - y + z = 0$$

(b) Now use this projection matrix  $\mathbf{P}$  to project the vector  $\vec{\mathbf{b}}$  on to this plane where

$$\vec{\mathbf{b}} = \begin{pmatrix} 2 \\ 4 \\ -1 \end{pmatrix}$$

**Problem 6**

Find the least squares solution of a quadratic equation which passes through the origin for the three points

$$\vec{\mathbf{a}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\vec{\mathbf{b}} = \begin{pmatrix} 2 \\ 5 \end{pmatrix}$$

$$\vec{\mathbf{c}} = \begin{pmatrix} -1 \\ -2 \end{pmatrix}$$

**Problem 7**

Find the least squares solution for the following system of linear equations

$$x_1 - x_2 = 4$$

$$3x_1 + 2x_2 = 1$$

$$-2x_1 + 4x_2 = 3$$

**Problem 8**

Find the least squares solution of a linear equation for the four points

$$\vec{a} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\vec{b} = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$$

$$\vec{c} = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$$

$$\vec{d} = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$$

**Problem 9**

Find the least squares solution of a quadratic equation for the four points

$$\vec{a} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\vec{b} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$$

$$\vec{c} = \begin{pmatrix} 3 \\ 1 \end{pmatrix}$$

$$\vec{d} = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

**Problem 10: All Good Things Must Come to an End**

We have reached the end of this short on-line summer course, “An Introduction to Linear Algebra Using Python.” I want to acknowledge the many textbooks and on-line sources I have freely and liberally drawn from to develop both my lectures and problem sets. I list these references by starting with the textbook I have most frequently consulted and going in descending order from there. I will not make any editorial comments on the pros and cons of each textbook, but if you are interested in my personal viewpoints please directly contact me by email!

1. Mike X. Cohen, “Linear Algebra: Theory, Intuition, and Code”, 1st ed., Sincxpress BV, Middletown, DE, 2021.
2. Donald A. McQuarrie, “Mathematical Methods for Scientists and Engineers”, University Science Books, 2003.
3. Howard Anton and Robert C. Busby, “Contemporary Linear Algebra”, John Wiley & Sons, New York, 2003.
4. Gilbert Strang, “Introduction to Linear Algebra”, 4th ed., Wellesley-Cambridge Press, Wellesley, MA, 2009.
5. Khan Academy, “Linear Algebra”, (<https://youtube.com/playlist?list=PLFD0EB975BA0CC1E0>).
6. Dr. H. Trevor Johnson-Steigelman, The University of Maryland at College Park, (<http://youtu.be/JOSK7Zy8iko>)

### Python Exercise 13

1. Here is a Python script which uses the function `np.linalg.lstsq` to find the least squares solution to a linear matrix equation. It also plots the initial raw data with the fitted line. At this point in the course you should be able to go through each line and figure out what is going on using your elementary knowledge of Python and suitable on-line references.

```
import numpy as np
import matplotlib.pyplot as plt
x = np.array([0, 1, 2, 3])
y = np.array([-1, 0.26, 3.9, 7.1])
A = np.vstack([x, np.ones(len(x))]).T
A
m, c = np.linalg.lstsq(A, y, rcond=None)[0]
m, c
(1.0 -0.95) # may vary
plt.plot(x, y, 'o', label='Original data', markersize=10)
plt.plot(x, m*x + c, 'r', label='Fitted line')
plt.legend()
plt.show()
```

You should play with this script and adapt it to verify the least squares solutions in Problem Set XIII. You may commit some mistakes but that is part of learning.

2. Go back to all the problems in Problem Set XIII and compute the error in each of the projection and least square approximation problems. Now write a Python script to do the same and confirm your results.

3. Now you are ready for our final and most difficult Python script so far. Without using the function `np.linalg.lstsq` from our previous Python exercises, it uses the theory of Problem 1 to find the least squares solution of a set of data fitted to a line. It is worth the trouble to research how each line in the script works and then play with it for new problems. You might even use a quadratic function as a fitting function and verify the appropriate problems in Problem Set XIII. Have fun! Clearly there is much more to learn about Python, but our purpose in this course was to give you a taste of its power and encourage you to learn more.

```
import numpy as np
from scipy import optimize
import matplotlib.pyplot as plt
plt.style.use('seaborn-poster')
# generate x and y
x = np.linspace(0, 1, 101)
y = 1 + x + x * np.random.random(len(x))
# assemble matrix A
A = np.vstack([x, np.ones(len(x))]).T
# turn y into a column vector
y = y[:, np.newaxis]
# Direct solution of least squares problem
alpha = np.dot((np.dot(np.linalg.inv(np.dot(A.T,A)),A.T)),y)
print(alpha)
# plot the results
plt.figure(figsize = (10,8))
plt.plot(x, y, 'b.')
plt.plot(x, alpha[0]*x + alpha[1], 'r')
plt.xlabel('x')
plt.ylabel('y')
plt.show()
```