Homework 4

January 30, 2024

This homework assignment is about Gaussian Elimination (section 2.1).

For the written exercises, you should upload a scanned PDF to Gradescope and then follow the prompts given by Gradescope to assign certain pages of your PDF document to the correct problems.

For the coding exercises, you will be prompted to upload your python file directly to Gradescope. Please make sure you submit only one file and that your variables are assigned to the correct variable names.

The course syllabus found on Canvas has information on how homework is graded and how homework should be presented and submitted. Please let me or the TAs know if you have any questions or concerns.

This assignment is due on Thursday, February 1 at 11:59pm.

Written Assignment

In section 2.1 of our textbook complete the following problems: 2, 3a, 3b, 3c, 3d, 7a, 7b, and 7e.

- Note that book is referring to "Naive Gaussian elimination" meaning we are not allowed to use the pivoting elementary row operation. (i.e. we are not allowed to use swap rows as an elementary row operation. Only multiplication and replacement.)
- For the written exercises, you do **NOT** need to write out every step of the algorithm, as we did in class on Wednesday 1-24. To show your work, you should list each elementary row operation you use and the matrix that results from that elementary row operation.
- The answers listed in the back of the textbook for 2 and 3a and 3b are correct, however the answers listed in the back of the book for problem 7a and 7b are not correct. Remember that to get credit for these problems you must show your work, a solution that only lists the answer will not receive credit.

Coding Assignment

The coding assignment for this week will ask you to write a function in python that executes Naive Gaussian Elimination. There is a template in the week 4 module that you may use to write this function if you please. You may also refer to the pseudocode on page 77 of our textbook. Our python function should include $\mathbf{A}$ and $\mathbf{b}$ as arguments and return $\mathbf{x}$.

Autograder updates:

- The autograder is configured to treat $\mathbf{b}$ and $\mathbf{x}$ as 1D arrays in python. So if you wrote your Naive_Gauss function where $\mathbf{b}$ and $\mathbf{x}$ are 2D arrays, please modify it so they are 1D arrays. Note that this change is necessary only for the autograder purposes.
- At the very beginning of your Naive_Gauss function, please convert your arguments A and b from arrays of ints to arrays of floats. For example we can use:

```
A = A.astype(‘‘float”)
b = b.astype(‘‘float”)
```

Once you have written the Naive Gaussian Elimination function in python, write a python script that has your Naive Gaussian Elimination function defined at the beginning of the script. Next we will call the function to solve three systems in the form $\mathbf{Ax} = \mathbf{b}$.

1. Consider our example problem from lecture:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 6 & 10 \\ 3 & 14 & 28 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ -8 \end{bmatrix}.$$

a.) Create the matrix $\mathbf{A}$ as a 2D numpy array and assign it to the variable A1.

b.) Create the matrix $\mathbf{b}$ and assign it to the variable b1.

c.) Call your Naive Gaussian Elimination function with arguments A1 and b1 and assign its return value to the variable x1.

```
x1 = Naive_Gauss(A1,b1)
```

2. Recall the problem 3 of the written assignment from homework 2. In this problem, we considered a cross-section of a metal beam, with negligible heat flow in the direction perpendicular to the plate. We derived a system of equations with T_1, T_2, T_3, and T_4 representing the temperature of four interior nodes on that plate. This system was derived based on the assumption that the temperature at a node is approximately equal to the average of the four nearest nodes and we were also given the temperature of the exterior nodes. This system can be written as a matrix equation $\mathbf{Ax} = \mathbf{b}$ where

$$\mathbf{A} = \begin{bmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 30 \\ 60 \\ 40 \\ 70 \end{bmatrix},$$

and the vector of unknowns is:

$$\mathbf{x} = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}.$$

a.) Create the matrix $\mathbf{A}$ as a 2D numpy array and assign it to the variable A2.

b.) Create the matrix $\mathbf{b}$ and assign it to the variable b2.

c.) Call your Naive Gaussian Elimination function with arguments A2 and b2 and assign its return value to the variable x2.

3. This is problem number 4 from computer exercises for section 2.1 on page 81 of our textbook. Worded in a slightly different way.

a.) Consider the 15×15 matrix $\mathbf{A}$ where $a_{ij} = -1 + 2\min(i, j)$.

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 3 & 3 & \cdots & 3 \\ 1 & 3 & 5 & \cdots & 5 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 3 & 5 & \cdots & 29 \end{bmatrix}$$

Use a nested for loop to create the matrix $\mathbf{A}$ as a 2D array in python and assign it to the variable A3. You can use the np.min() function in your for loop b.) Assign the variable b3 in the following way:

```
b3 = A3@np.ones(15)
```

c.) Call your Naive Gaussian Elimination function with arguments A3 and b3 and assign its return value to the variable x3.