Homework 8

CAS CS 132: Geometric Algorithms

Due: **Thursday November 9, 2023 at 11:59PM**

Submission Instructions

- Make the answer in your solution to each problem abundantly clear (e.g., put a box around your answer or used a colored font if there is a lot of text which is not part of the answer).

- Choose the correct pages corresponding to each problem in Gradescope. Note that Gradescope registers your submission as soon as you submit it, so you don't need to rush to choose corresponding pages. **For multipart questions, please make sure each part is accounted for.**

Graders have license to dock points if either of the above instructions are not properly followed.

Practice Problems

The following list of problems comes from *Linear Algebra and its Application 5th Ed* by David C. Lay, Steven R. Lay, and Judi J. McDonald. They may be useful for solidifying your understanding of the material and for studying in general. **They are optional, so please don't submit anything for them**.

- 2.8.1-5, 2.8.11-12, 2.8.15-16, 2.8.21-22, 2.8.25, 2.8.31-32, 2.8.35

- 2.9.3-6, 2.9.12, 2.9.17-21

1 Subspaces and Matrices (Basics)

For each of the following parts, justify your answer.

A. (2 points) If $A \in \mathbb{R}^{3\times 6}$, then for what value of n is $\mathrm{Nul}\, A$ is a subspace of $\mathbb{R}^n$?

B. (4 points) If $A \in \mathbb{R}^{10\times 13}$ matrix then what is the minimum dimension of $\mathrm{Nul}\, A$?

C. (3 points) If $A \in \mathbb{R}^{7\times 5}$ matrix and $\dim(\mathrm{Col}\, A) = 4$, what is $\dim(\mathrm{Nul}\, A)$?

D. (3 points) Determine if $\mathbf{v}$ is in $\mathrm{Nul}\, A$ where

$$
\mathbf{v} = \begin{bmatrix} 2 \\ -1 \\ 2 \end{bmatrix} \qquad A = \begin{bmatrix} -1 & 0 & 1 \\ 3 & 6 & 0 \\ 5 & 7 & 2 \end{bmatrix}
$$

E. (4 points) Determine if $\mathbf{v}$ is in $\mathrm{Col}\, A$, where $\mathbf{v}$ and A are as in the previous part.

F. (4 points) Determine $\mathrm{rank}\, A$ where

$$
A = \begin{bmatrix} 2 & 1 & -8 & 3 \\ -1 & 3 & 4 & 2 \\ 3 & 2 & -12 & 5 \\ 1 & -2 & -4 & -1 \end{bmatrix}
$$

Hint. Attempt to do this without performing any calculations.

Solution.

2 Bases for Null Spaces and Column Spaces

Consider the following matrix.

$$
\begin{bmatrix}
1 & 2 & 4 & 3 & -4 & 1 \\
-3 & -4 & -10 & -8 & 13 & -3 \\
5 & 6 & 16 & 10 & -13 & 9 \\
-7 & -8 & -22 & -12 & 13 & -9 \\
13 & 18 & 44 & 32 & -47 & 11
\end{bmatrix}
$$

A. (2 points) Find the reduced echelon form of A. You may (should) use Python. Describe the process you used. For your convenience:

```
1  np.array([[1., 2, 4, 3, -4, 1], [-3, -4, -10, -8, 13, -3], [5,
      6, 16, 10, -13, 9], [-7, -8, -22, -12, 13, -9], [13, 18,
      44, 32, -47, 11]])
```

B. (6 points) Find a basis for Col A.

C. (6 points) Find a basis for Nul A.

Solution.

3 Affine Spaces

Let A be an $m \times n$ matrix and let $\mathbf{v}$ be a vector in $\mathbb{R}^n$.

A. (6 points) Show that if $A\mathbf{v} = \mathbf{b}$ and $\mathbf{w} \in \operatorname{Nul} A$, then $\mathbf{v} + \mathbf{w}$ is a solution to the equation $A\mathbf{x} = \mathbf{b}$.

B. (5 points) Show that if $\mathbf{b} \neq 0$, then the solution set of $A\mathbf{x} = \mathbf{b}$ is *not* a subspace of $\mathbb{R}^n$.

C. (5 points) **(Extra Credit)** A set H is an **affine** subspace of $\mathbb{R}^n$ if there is a subspace U of $\mathbb{R}^n$ and a vector $\mathbf{o}$ such that

$$H = \{\mathbf{u} + \mathbf{o} \mid \mathbf{u} \text{ is in } U\}$$

Show that **if the solution set of** $A\mathbf{x} = \mathbf{b}$ **is nonempty** then it is an affine subspace. This means choosing a vector $\mathbf{o}$ and subspace U.

Solution.

4 Complement of the Column Space

A. (2 points) Find a vector which is *not* in Col A, where

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

B. (5 points) Find a vector which is *not* in Col A, where

$$A = \begin{bmatrix} 1 & 1 & 5 \\ -1 & 0 & -1 \\ 1 & 2 & 9 \end{bmatrix}$$

C. (8 points) Let A be a $5 \times n$ matrix such that rank $A = 4$, which has an LU decomposition where

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 4 & 1 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 \\ 0 & 3 & -3 & 0 & 1 \end{bmatrix}$$

Determine if $\mathbf{v}$ in Col A, where

$$\mathbf{v} = \begin{bmatrix} 2 \\ -5 \\ -11 \\ 5 \\ -12 \end{bmatrix}$$

Solution.

5 Problem (Programming)

(15 points) A lot of what doing computational linear algebra entails is building and manipulating matrices. NumPy provides a powerful interface for doing this, but it takes some time to become familiar with. In this problem, you will be constructing NumPy arrays using functions from the NumPy standard library. You are required to complete each part **with a single line of code**.

You are given starter code in the file `hw08prog.py`. **Don't change the name of this file when you submit.** Also don't change the names of any functions or variables provided in the starter code. **The only changes you should make are to fill in the provided TODO items.** You will upload a single file `hw08prog.py` to Gradescope.

For each of the following parts, fill in (at the corresponding TODO item in `hwprog08.py`) a **single line (fewer than 100 characters))** of Python code which builds the 2D NumPy arrays representing each matrix. In particular, you can't hardcode the arrays. This may require reading through some of the NumPy documentation.

A.

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

B.

$$
\begin{bmatrix}
4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 4
\end{bmatrix}
$$

C. *Hint.* Look at some of the examples in the documentation for `np.eye`.

$$
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

D. *Hint.* Look at `np.arange` and `np.diag`.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 \end{bmatrix}$$

E. *Hint.* Use matrix addition.

$$\begin{bmatrix} 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 4 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 6 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 8 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 10 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 12 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 14 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 16 \end{bmatrix}$$

F. *Hint.* Look at some of the examples in the documentation for `np.arange`.

$$\begin{bmatrix} 8 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 7 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 7 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

G.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

H. *Hint.* Look at `np.ones`.

$$\begin{bmatrix} 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$

I. *Hint.* Look at `np.hstack`.

$$\begin{bmatrix} 3 & 3 & 3 & 3 & 1 & 0 & 0 & 0 \\ 3 & 3 & 3 & 3 & 0 & 1 & 0 & 0 \\ 3 & 3 & 3 & 3 & 0 & 0 & 1 & 0 \\ 3 & 3 & 3 & 3 & 0 & 0 & 0 & 1 \end{bmatrix}$$

J. *Hint.* Look also at `np.vstack`.

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -2 & -2 & -2 & -2 \\ 0 & 1 & 0 & 0 & -2 & -2 & -2 & -2 \\ 0 & 0 & 1 & 0 & -2 & -2 & -2 & -2 \\ 0 & 0 & 0 & 1 & -2 & -2 & -2 & -2 \end{bmatrix}$$

K. *Hint.* Remember you can index matrices on lists.

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

L.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

M. *Hint.* Look at `np.tril`.

$$\begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$