Homework 4 Solutions

CAS CS 132: Geometric Algorithms

Due: **Thursday October 5, 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.

Note. Solutions written here may be lengthy because they are expository, and may not reflect that amount of detail that you were expected to write in your own solutions.

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**.

- 1.8.3, 1.8.7, 1.8.9, 1.8.17, 1.8.31
- 1.9.1-10, 1.9.15, 1.9.22

1 Zero Rows in Echelon Forms

(10 points) Consider an arbitrary system of linear equations with n unknowns and m equations. Further suppose that

- it has a unique solution;
- it has at least as many equations as unknowns ($m \geq n$).

Write down an expression in terms of m and n for the number of all-zero rows which appear in the reduced echelon form of its augmented matrix. Justify your answer.

Solution. $m - n$. If the system has a unique solution, no variable appears free, or equivalently every column has a pivot. In particular, the pivots *must* appear in the first n rows in order to maintain that leading entries appear to the right of those which appear above them. All entries below a pivot are zero, so the last $m - n$ rows must be all-zeros.

2 Non-Linear Transformation

(10 points) Show that the transformation $T : \mathbb{R}^4 \to \mathbb{R}^4$ given by

$$T\left(\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}\right) = \begin{bmatrix} \min(v_1, 100) \\ \min(v_2, 100) \\ \min(v_3, 100) \\ \min(v_4, 100) \end{bmatrix}$$

is **not** linear.

Solution. This transformation does not satisfy either linearity condition. Take

$$\mathbf{v} = \begin{bmatrix} 51 \\ 51 \\ 51 \\ 51 \end{bmatrix}$$

Then

$$T(2\mathbf{v}) = \begin{bmatrix} 100 \\ 100 \\ 100 \\ 100 \end{bmatrix}$$

whereas

$$2T(\mathbf{v}) = \begin{bmatrix} 102 \\ 102 \\ 102 \\ 102 \end{bmatrix}$$

This also shows that additivity does not hold.

3 Affine Hyperplanes

(10 points) Consider the following linear equation.

$$x + y + z = 5$$

The plane represented by this equation does not include the origin (such a plane is called *affine*). Write down vectors $\mathbf{b}$, $\mathbf{v}_1$, and $\mathbf{v}_2$ such that every point in the plane represented by the above equation (that is, every point (a, b, c) such that $a+b+c = 5$) can be written as $\mathbf{b}+\mathbf{v}$ where $\mathbf{v}$ is in $\mathsf{span}\{\mathbf{v}_1, \mathbf{v}_2\}$. In set notation,

$$\{(x, y, z) : x + y + z = 5\} = \{\mathbf{b} + \mathbf{v} : \mathbf{v} \in \mathsf{span}\{\mathbf{v}_1, \mathbf{v}_2\}\}.$$

(Hint. Choose $\mathbf{b}$ to be a solution to the equation $x + y + z = 5$ and then find vectors spanning the plane given by $x+y+z = 0$. Also note that vectors which span a plane given by a linear equation must also be solutions to that equation.)

Solution.

$$\mathbf{b} = \begin{bmatrix} 5 \\ 0 \\ 0 \end{bmatrix} \qquad \mathbf{v}_1 = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \qquad \mathbf{v}_2 = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}$$

4 Matrix of a Transformation (Algebraic)

(10 points) Let $T : \mathbb{R}^3 \to \mathbb{R}^3$ be a linear transformation such that

$$T\left(\begin{bmatrix}2\\1\\0\end{bmatrix}\right) = \begin{bmatrix}3\\-3\\4\end{bmatrix} \qquad T\left(\begin{bmatrix}1\\1\\0\end{bmatrix}\right) = \begin{bmatrix}8\\0\\2\end{bmatrix} \qquad T\left(\begin{bmatrix}2\\1\\2\end{bmatrix}\right) = \begin{bmatrix}-1\\-1\\-1\end{bmatrix}$$

Find the matrix implementing T.

Solution. First note that the vectors

$$\begin{bmatrix}2\\1\\0\end{bmatrix} \qquad \begin{bmatrix}1\\1\\0\end{bmatrix} \qquad \begin{bmatrix}2\\1\\2\end{bmatrix}$$

are linearly independent, so T is completely determined by its behavior on these inputs. In order to find the matrix for T, we have to figure out the values of T on the standard basis. This means writing the standard basis as a linear combination of the vectors above, and then using linearity to determine the value of T on these linear combinations. Each basis vector in turn:

$$T\left(\begin{bmatrix}1\\0\\0\end{bmatrix}\right) = T\left(\begin{bmatrix}2\\1\\0\end{bmatrix} - \begin{bmatrix}1\\1\\0\end{bmatrix}\right) = T\left(\begin{bmatrix}2\\1\\0\end{bmatrix}\right) - T\left(\begin{bmatrix}1\\1\\0\end{bmatrix}\right) = \begin{bmatrix}3\\-3\\4\end{bmatrix} - \begin{bmatrix}8\\0\\2\end{bmatrix} = \begin{bmatrix}-5\\-3\\2\end{bmatrix}$$

$$T\left(\begin{bmatrix}0\\1\\0\end{bmatrix}\right) = T\left(\begin{bmatrix}1\\1\\0\end{bmatrix} - \begin{bmatrix}1\\0\\0\end{bmatrix}\right) = T\left(\begin{bmatrix}1\\1\\0\end{bmatrix}\right) - T\left(\begin{bmatrix}1\\0\\0\end{bmatrix}\right) = \begin{bmatrix}8\\0\\2\end{bmatrix} - \begin{bmatrix}-5\\-3\\2\end{bmatrix} = \begin{bmatrix}13\\3\\0\end{bmatrix}$$

$$T\left(\begin{bmatrix}0\\0\\2\end{bmatrix}\right) = T\left(\begin{bmatrix}2\\1\\2\end{bmatrix} - \begin{bmatrix}2\\1\\0\end{bmatrix}\right) = T\left(\begin{bmatrix}2\\1\\2\end{bmatrix}\right) - T\left(\begin{bmatrix}2\\1\\0\end{bmatrix}\right) = \begin{bmatrix}-1\\-1\\-1\end{bmatrix} - \begin{bmatrix}3\\-3\\4\end{bmatrix} = \begin{bmatrix}-4\\2\\-5\end{bmatrix}$$

$$T\left(\begin{bmatrix}0\\0\\1\end{bmatrix}\right) = T\left((0.5)\begin{bmatrix}0\\0\\2\end{bmatrix}\right) = (0.5)T\left(\begin{bmatrix}0\\0\\2\end{bmatrix}\right) = (0.5)\begin{bmatrix}-4\\2\\-5\end{bmatrix} = \begin{bmatrix}-2\\1\\-2.5\end{bmatrix}$$

All together, this gives us the matrix

$$\begin{bmatrix}-5 & 13 & -2\\-3 & 3 & 1\\2 & 0 & -2.5\end{bmatrix}$$

5 Matrix of a Transformation (Geometric)

A. (5 points) Find the (3×3) matrix for the linear transformation which reflects vectors through the x_1x_2-plane. (Hint. The x_3-coordinate should be negated.)

B. (10 points) Find the (3×3) matrix for the linear transformation which rotates vectors about the line generated by span of the vector $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ by 120 degrees. You can choose which direction you want to rotate by, we will accept either answer. (Hint. Graph the span as a line. Think carefully about the way this transformation affects the standard basis vectors.)

Solution.

A. We have to determine how this tranformation affects the standard basis. The vectors

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

are unaffected. The only vector which is affected is

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

which after reflection becomes

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

All together, this gives the matrix

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

B. This problem is tricky on the surface, but is made easier by the fact that the rotation is by 120 degrees. The vectors $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ extends diagonally from the origin and the standard basis vectors are rotationally symmetric about this line. After 90 degrees, the basis vectors switch places with each other:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \mapsto \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \mapsto \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \mapsto \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

or

$$\begin{bmatrix}1\\0\\0\end{bmatrix} \mapsto \begin{bmatrix}0\\0\\1\end{bmatrix} \qquad \begin{bmatrix}0\\1\\0\end{bmatrix} \mapsto \begin{bmatrix}1\\0\\0\end{bmatrix} \qquad \begin{bmatrix}0\\0\\1\end{bmatrix} \mapsto \begin{bmatrix}0\\1\\0\end{bmatrix}$$

depending on which direction you rotate. The first case gives the matrix

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

6 Grade Breakdown

Let A be a matrix with n rows and 6 columns. Each row of A contains the **unweighted** percentage scores (out of 100) of one student on 4 homework assignments (columns 1 through 4) a midterm (column 5) and a final (column 6).

$$\begin{array}{cccccc} H_1 & H_2 & H_3 & H_4 & M & F \end{array}$$

$$\begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{15} & p_{16} \\ p_{21} & p_{22} & p_{23} & p_{24} & p_{25} & p_{26} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{n1} & p_{n2} & p_{n3} & p_{n4} & p_{n5} & p_{n6} \end{bmatrix}$$

All homework assignments are worth the same amount. Consider the transformation T for this matrix.

A. (5 points) Suppose that homework assignments account for **50** percent of the final grade, the midterm accounts for **20** percent and the final accounts for **30** percent. Find a vector $\mathbf{v}$ such that $T(\mathbf{v})$ is the vector whose i-th entry is the final percentage grade of the i-th student. So, for example, if the i-th student recieved 90 percent on every homework assignment, 85 percent on the midterm, and 92 percent on the final, then the i-th entry of the output vector should be $90 * 0.5 + 85 * 0.2 + 92 * 0.3$.

B. (5 points) Find a vector $\mathbf{v}$ such that $T(\mathbf{v})$ is the vector whose i-th entry is the unweighted homework grade for student i. For the same example as above, the i-th entry would be 90.

Solution. This problem is a matter of understanding that applying a matrix transformation to a vector means taking a linear combination of its columns. In this case, the linear combination is the grade weighting.

A.

$$\mathbf{v} = \begin{bmatrix} 0.125 \\ 0.125 \\ 0.125 \\ 0.125 \\ 0.2 \\ 0.3 \end{bmatrix}$$

B.

$$\mathbf{v} = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \\ 0 \\ 0 \end{bmatrix}$$

7 Defining Matrix-Vector Multiplication Yourself (Programming)

(15 points) This week you will be writing your own definition of matrix-vector multiplication in Python. This will give you the opportunity to work with numpy arrays directly if you haven't already (all your previous assignments have only required indirect use of numpy arrays). **Read through the docstring of each function carefully.**

You are given starter code in the file `hw04prog.py`. **Don't change the name of this file when you submit.** Also don't change the names of the functions included in the starter code. **The only changes you should make are to fill in the TODO items in the starter code.** There are three functions you need to fill in.

- `inner_product`, which computes the inner product of two vectors. Recall that the inner product, also called the *dot product*, is defined as

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_1 \\ \vdots \\ v_n \end{bmatrix} = u_1v_1 + u_2v_2 + \cdots + u_nv_n = \sum_{i=1}^{n} u_iv_i$$

- `mat_vec_mult_ip`, which computes matrix-vector multiplication using the row-column rule and inner products:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} a_{1i}v_i \\ \sum_{i=1}^{n} a_{2i}v_i \\ \vdots \\ \sum_{i=1}^{n} a_{mi}v_i \end{bmatrix}$$

  Recall that the rows of a numpy array are themselves numpy arrays, so you should be able to use your `inner_product` function directly here.

- `mat_vec_mult_vs`, which computes matrix-vector multiplication, but using the definition we gave in class:

$$\begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \dots & \mathbf{a}_n \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = v_1\mathbf{a}_1 + v_2\mathbf{a}_2 + \cdots + v_n\mathbf{a}_n$$

Of course, the last two functions will compute the same vector, but they will do so in different ways. The first should compute each entry of the output vector individually, whereas the second should compute a linear combination of the columns of the given matrix using addition and scaling of numpy arrays.

For this assignment **you are not allowed use built in numpy functions for inner products or matrix-vector multiplication, like `np.inner` or `np.dot` or `@`.** The point is for you to implement your own.

You will upload the single python file `hw04prog.py` to Gradescope with your implementations of the required functions. We will be running autograder tests on your submission to determine its correctness. **You will not have access to the autograder tests.**