EECS 16A Designing Information Devices and Systems I
Fall 2018 Homework 3

This homework is due September 14, 2018, at 23:59.
Self-grades are due September 18, 2018, at 23:59.

Submission Format
Your homework submission should consist of **two** files.

- `hw3.pdf`: A single PDF file that contains all of your answers (any handwritten answers should be scanned) as well as your IPython notebook saved as a PDF.

  If you do not attach a PDF of your IPython notebook, you will not receive credit for problems that involve coding. Make sure that your results and your plots are visible.

- `hw3.ipynb`: A single IPython notebook with all of your code in it.

  In order to receive credit for your IPython notebook, you must submit both a "printout" and the code itself.

Submit each file to its respective assignment on Gradescope.

1. Counting Solutions

For each of the following systems of linear equations, determine if there is a unique solution, no solution, or an infinite number of solutions. **Show your work**. If there is a unique solution, find it. If there are an infinite number of solutions, describe the space of solutions.

(a) **(PRACTICE)**

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

Solution:

$$\left[\begin{array}{ccc|c} 1 & 1 & 1 & 3 \\ 2 & 2 & 2 & 5 \end{array}\right] \rightarrow \left[\begin{array}{ccc|c} 1 & 1 & 1 & 3 \\ 0 & 0 & 0 & -1 \end{array}\right] \longleftarrow R_2 - 2R_1 \mapsto R_2$$

No solution. The fact that there are fewer equations than there are unknowns immediately means that it is not possible to have a unique solution; however, this does not guarantee that there is a solution to begin with. From Gaussian Elimination, we can see that these equations are inconsistent and we have a row of zero equating to a nonzero value. In other words, no values of x, y, and z can satisfy both equations simultaneously.

(b)

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

Solution:

Because there are two equations and three unknowns, we immediately see that there can be no unique solution. The question then becomes if there are an infinite number of solutions, or no solution at all. Using Guassian Elimination, we can add an additional equation which provides no unique information

$$\left[\begin{array}{ccc|c} 0 & -1 & 2 & 1 \\ 2 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

This gives us the square matrix formulation we were first introduced to, and can see after rearranging the matrix into upper triangular form that we have a zero pivot. That said, the equations do not contradict each other, so we can find the space of solutions:

$$y = 2z - 1$$
$$x = 1 - \frac{1}{2}z$$

Infinite solutions, $\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}t + 1 \\ 2t - 1 \\ t \end{bmatrix} \forall t \in \mathbb{R}$

(c) **(PRACTICE)**

$$\begin{array}{rcrl} x & + & 2y & = 3 \\ 2x & - & y & = 1 \\ 3x & + & y & = 4 \end{array}$$

Solution:

$$\left[\begin{array}{cc|c} 1 & 2 & 3 \\ 2 & -1 & 1 \\ 3 & 1 & 4 \end{array}\right] \to \left[\begin{array}{cc|c} 3 & 1 & 4 \\ 2 & -1 & 1 \\ 3 & 1 & 4 \end{array}\right] \longleftarrow R_1 + R_2 \mapsto R_1$$

$$\to \left[\begin{array}{cc|c} 3 & 1 & 4 \\ 2 & -1 & 1 \\ 0 & 0 & 0 \end{array}\right] \longleftarrow R_3 - R_1 \mapsto R_3$$

$$\to \left[\begin{array}{cc|c} 3 & 1 & 4 \\ 5 & 0 & 5 \\ 0 & 0 & 0 \end{array}\right] \longleftarrow R_1 + R_2 \mapsto R_2$$

$$\to \left[\begin{array}{cc|c} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{array}\right] \longleftarrow R_1 - \frac{3}{5}R_2 \mapsto R_1$$

$$\to \left[\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{array}\right] \longleftarrow \text{switch } R_1, R_2$$

Unique solution, $\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. Notice how even with the redundant equation, we still have enough information to uniquely find x and y! If this is unclear, the system of linear equations at the end of the

Gaussian Elimination above simply reads out

$$\begin{aligned} x &= 1 \\ y &= 1 \\ 0 &= 0 \end{aligned}$$

Do not add new columns! Because each column represents a variable, we should not add columns of zeros in an attempt to reshape our matrix into a square. In this probelem, the first equation does not provide any new information, but no more information is necessary to solve for x and y.

(d)

$$\begin{array}{rcrl} x & + & 2y & = 3 \\ 2x & - & y & = 1 \\ x & - & 3y & = -5 \end{array}$$

Solution:

$$\left[\begin{array}{cc|c} 1 & 2 & 3 \\ 2 & -1 & 1 \\ 1 & -3 & -5 \end{array}\right] \rightarrow \left[\begin{array}{cc|c} 1 & 2 & 3 \\ 2 & -1 & 1 \\ 2 & -1 & -2 \end{array}\right] \longleftarrow R_3 + R_1 \mapsto R_3$$

$$\rightarrow \left[\begin{array}{cc|c} 1 & 2 & 3 \\ 2 & -1 & 1 \\ 0 & 0 & 3 \end{array}\right] \longleftarrow R_2 - R_3 \mapsto R_3$$

No solution. We can think of this to mean that there are no values of x and y which satisfy the conditions in all three equations simultaneously, because in order to satisfy all three equations, the last row $0 = 3$ would need to be true. Even though we have more equations than unknowns, that does not guarantee that a unique solution, or any solutions, exist.

(e)

$$\left[\begin{array}{rcrl} x & - & y & = 2 \\ 5x & - & 5y & = 10 \\ 3x & - & 3y & = 6 \end{array}\right]$$

Solution:

We can see that all three equations tell us the exact same information; that is, $x - y = 2$. For completeness

$$\left[\begin{array}{cc|c} 1 & -1 & 2 \\ 5 & -5 & 10 \\ 3 & -3 & 6 \end{array}\right] \rightarrow \left[\begin{array}{cc|c} 1 & -1 & 2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}\right] \longleftarrow R_2 - 5R_1 \mapsto R_2 \text{ and } R_3 - 3R_1 \mapsto R_3$$

Infinite solutions, $\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t \\ t-2 \end{bmatrix} \forall t \in \mathbb{R}$

2. Elementary Matrices

In lecture, we learned about an important technique for solving systems of linear equations called Gaussian elimination. It turns out that each row operation in Gaussian elimination can be performed by multiplying the augmented matrix on the left by a specific matrix called an *elementary matrix*. For example, suppose we want to row reduce the following augmented matrix:

$$\mathbf{A} = \left[\begin{array}{cccc|c} 1 & -2 & 0 & -5 & 15 \\ 0 & 1 & 0 & 3 & -7 \\ -2 & -3 & 1 & -6 & 9 \\ 0 & 1 & 0 & 2 & -5 \end{array}\right] \tag{1}$$

What matrix do you get when you subtract the 4th row from the 2nd row of **A** (putting the result in row 2)? (You don't have to include this in your solution.) Now, try multiplying the original **A** on the left by

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(You don't have to include this in your solutions either.) Notice that you get the same thing.

$$\mathbf{EA} = \left[\begin{array}{cccc|c} 1 & -2 & 0 & -5 & 15 \\ 0 & 0 & 0 & 1 & -2 \\ -2 & -3 & 1 & -6 & 9 \\ 0 & 1 & 0 & 2 & -5 \end{array}\right]$$

E is a special type of matrix called an *elementary matrix*. This means that we can obtain the matrix **E** from the identity matrix by applying an elementary row operation – in this case, subtracting the 4th row from the 2nd row.

In general, any elementary row operation can be performed by left multiplying by an appropriate elementary matrix.

In other words, you can perform a row operation on a matrix **A** by first performing that row operation on the identity matrix to get an elementary matrix (see below), and then left multiplying **A** by the elementary matrix (like we did above).

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \xrightarrow{R_2 - R_4 \mapsto R_2} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \mathbf{E}$$

(a) Write down the elementary matrices required to perform the following row operations on a 4×5 augmented matrix.

i. $R_1 \mapsto R_3$
$R_3 \mapsto R_1$

ii. $-5R_3 \mapsto R_3$

iii. $3R_2 + R_4 \mapsto R_4$
$R_1 - R_2 \mapsto R_1$

*Hint: For the last one, note that if you want to perform two row operations on the matrix **A**, you can perform them both on the identity matrix and then left multiply **A** by the resulting matrix.*

Solution:

We obtain each of the desired elementary matrices by performing the row operations on a 4×4 identity matrix.

i. Switching rows 1 and 3:

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

ii. Multiplying row 3 by -5:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

iii. Adding 3×row 2 to row 4 and subtracting row 2 from row 1:

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

Note that we obtained this last matrix by applying two elementary row operations to the identity matrix. We could have performed each elementary row operation on individual identity matrices and then multiplied them together to achieve the same result. In this case, the order of the matrices did not matter; however, this is not true in general.

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

(b) In lecture we emphasized using Gaussian Elimination to reach an upper triangular form to determine the number of solutions for a given system of linear equations. When there is a unique solution, however, it is useful to determine exactly what that solution is by continuing Gaussian Elimination to reach a "fully reduced" form like that shown below. An example of this process can be found in Note 1, Example 1.7.

Compute a matrix **E** (by hand) that fully row reduces the augmented matrix **A** given in Equation (1)– that is, find **E** such that **EA** is a diagonal matrix with 1s along the diagonal. Show that this is true by multiplying out **EA**. When an augmented matrix is in this final form, it will have the form

$$\mathbf{EA} = \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & b_1 \\ 0 & 1 & 0 & 0 & b_2 \\ 0 & 0 & 1 & 0 & b_3 \\ 0 & 0 & 0 & 1 & b_4 \end{array}\right]$$

Once you have found the required elementary matrices,

i. use IPython to find the matrix **E**

ii. verify by hand that multiplying **E** and **A** gives you the identity matrix augmented with constants (as **EA** shown above).

Hint: As before, note that you can either ***apply a set of row operations to the same identity matrix*** *or* ***apply them to separate identity matrices and then multiply the matrices together****. Make sure, though, that you apply the row operations and multiply the matrices in the correct order.*

Solution:

We first need to row reduce $\mathbf{A}$ by hand to find the required row operations. The following row operations will do the trick (though you could have used a different set that does the same thing.)

- Step 1: Add $2\times$**Row 1** to Row 3
- Step 2: Add $2\times$**Row 2** to Row 1, add $7\times$**Row 2** to Row 3, and subtract **Row 2** from Row 4
- Step 3: Add **Row 4** to Row 1, add $3\times$**Row 4** to Row 2, and add $5\times$**Row 4** to Row 3
- Step 4: Multiply **Row 4** by -1

Note that we have grouped the row operations together, so that each step involves adding a scalar multiple of a particular row (bolded above) to the other rows. This will make calculating the elementary matrices for each step easier.

Applying each of these sets of row operations to a 4×4 identity matrix gives us the following matrices:

- Step 1:

$$\mathbf{E}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Step 2:

$$\mathbf{E}_2 = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 7 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

- Step 3:

$$\mathbf{E}_3 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Step 4:

$$\mathbf{E}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

We now multiply these matrices together as follows:

$$\mathbf{E} = \mathbf{E}_4\mathbf{E}_3\mathbf{E}_2\mathbf{E}_1 = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 1 \end{bmatrix}} \underbrace{\begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 7 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 2 & 7 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

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

Note the order in which we multiplied the matrices. $\mathbf{E}_1$ gets applied first, so it is furthest to the right (i.e. it will act on the augmented matrix first), etc. Also, note that we could have applied the row operations to the identity matrices in different groups. For example, we could have written an elementary matrix for each individual row operation and multiplied all of them together, making sure to maintain the correct order. We also could have applied all of the row operations, in the correct order, to a single identity matrix. **The important thing is that we maintain the correct order of row operations – either when we're applying them to an individual identity matrix or multiplying the elementary matrices together.** You should have done these multiplications in the IPython notebook.

To show that $\mathbf{E}$ does, in fact, row reduce $\mathbf{A}$, we calculate $\mathbf{EA}$.

$$\mathbf{EA} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & -2 & 0 & 3 \\ 2 & 2 & 1 & 5 \\ 0 & 1 & 0 & -1 \end{bmatrix} \left[\begin{array}{cccc|c} 1 & -2 & 0 & -5 & 15 \\ 0 & 1 & 0 & 3 & -7 \\ -2 & -3 & 1 & -6 & 9 \\ 0 & 1 & 0 & 2 & -5 \end{array}\right]$$

$$= \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 \end{array}\right]$$

3. Mechanical Inverses

For each of the following matrices, state whether the inverse exists. If so, find the inverse, $\mathbf{A}^{-1}$. If not, show why no inverse exists. **Solve the inverses by hand. You may use IPython for parts (a)-(d) to visualize how the matrix A changes a vector.**

(a) In addition to finding the inverse (if it exists), describe how the original matrix, $\mathbf{A}$, changes a vector it's applied to. For example, if $\mathbf{A}\vec{b} = \vec{c}$, then $\mathbf{A}$ could scale $\vec{b}$ by 2 to get $\vec{c}$, or $\mathbf{A}$ could reflect $\vec{b}$ across the x axis to get $\vec{c}$, etc. *Hint:* It may help to plot a few examples to recognize the pattern.

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

Solution:

The inverse does exist.

$\mathbf{A}^{-1} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

The original matrix $\mathbf{A}$ flips the x and y components of the vector. Any correct equivalent sequence of operations warrants full credit. Notice how the inverse does the exact same thing–that is, it switches the x and y components of the vector it's applied to. This makes sense–switching x and y twice on a vector $\begin{bmatrix} x \\ y \end{bmatrix}$ gives us $\begin{bmatrix} x \\ y \end{bmatrix}$.

(b) **(PRACTICE)** In addition to finding the inverse (if it exists), describe how the original matrix, $\mathbf{A}$, changes a vector it's applied to. For e.g., if $\mathbf{A}\vec{b} = \vec{c}$, then $\mathbf{A}$ could scale $\vec{b}$ by 2 to get $\vec{c}$, or $\mathbf{A}$ could reflect $\vec{b}$ across the x axis to get $\vec{c}$, etc. *Hint:* It may help to plot a few examples to recognize the pattern.

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

Solution:

The inverse does exist.

$\mathbf{A}^{-1} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$

The original matrix $\mathbf{A}$ reflects the vector across the y-axis, i.e. it multiplies the vector's x-component by a factor of -1. Reflecting the vector across the y-axis again with $\mathbf{A}^{-1} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ will give you the original vector.

(c) **(PRACTICE)** In addition to finding the inverse (if it exists), describe how the original matrix, $\mathbf{A}$, changes a vector it's applied to. For e.g., if $\mathbf{A}\vec{b} = \vec{c}$, then $\mathbf{A}$ could scale $\vec{b}$ by 2 to get $\vec{c}$, or $\mathbf{A}$ could reflect $\vec{b}$ across the x axis to get $\vec{c}$, etc. *Hint:* It may help to plot a few examples to recognize the pattern.

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

Solution:

We see here that the inverse does not exist because the first row (and second column) are the zero vector so the columns are linearly dependent. Since the columns of the matrix are linearly dependent, the inverse does not exist.

The original matrix $\mathbf{A}$ removes the x component of the vector it's applied to and keeps the same y component. Graphically speaking, this matrix can be thought of as taking the "shadow" of the vector on the y-axis if you were to shine a light perpendicular to the y-axis.

(d) In addition to finding the inverse (if it exists), describe how the original matrix, $\mathbf{A}$, changes a vector it's applied to. For e.g., if $\mathbf{A}\vec{b} = \vec{c}$, then $\mathbf{A}$ could scale $\vec{b}$ by 2 to get $\vec{c}$, or $\mathbf{A}$ could reflect $\vec{b}$ across the x axis to get $\vec{c}$, etc.

Hint: It may help to plot a few examples to recognize the pattern. What does the result look like when you apply this matrix to $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$? Draw it out. In the iPython notebook for this homework, we have provided you with a rotation function that may help to visualize what is happening.

$\mathbf{A} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

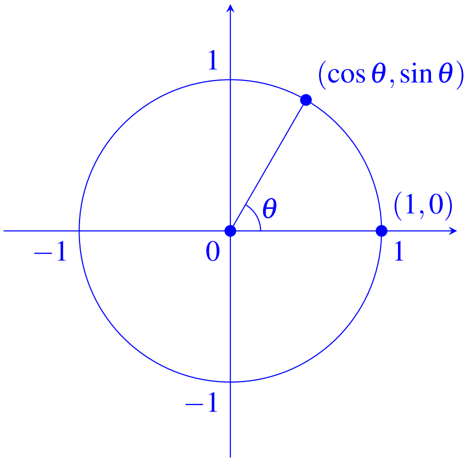
Solution:

The inverse does exist.

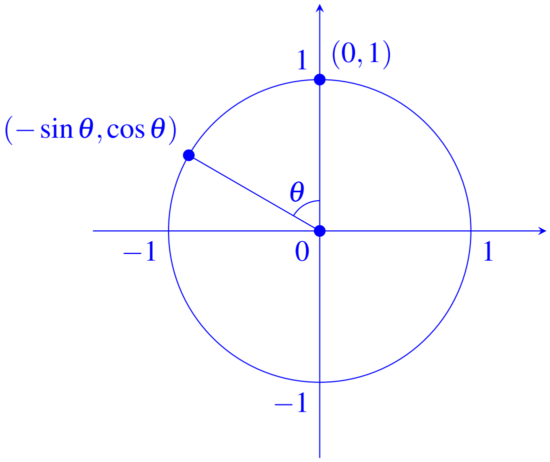
$\mathbf{A}^{-1} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$

The original matrix $\mathbf{A}$ is the two-dimensional rotation matrix. We'll look at the hint; namely, what the output looks like when we apply the matrix to $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$. In the diagrams below, we can see that applying the rotation matrix to a vector will rotate the vector by θ in the counter-clockwise direction.

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$



$$\begin{bmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}-\sin\theta \\ \cos\theta\end{bmatrix}$$



This is an important matrix to remember, but if you ever forget what the form of the rotation matrix is, you know the definition of matrix-vector multiplication in terms of the columns of the matrix

$$\begin{aligned}\mathbf{A}\vec{x} &= \begin{bmatrix} | & | & & | \\ \vec{a}_0 & \vec{a}_1 & \cdots & \vec{a}_n \\ | & | & & | \end{bmatrix}\begin{bmatrix}x_0\\x_1\\\vdots\\x_n\end{bmatrix} \\ &= \vec{a}_0x_0 + \vec{a}_1x_1 + \cdots + \vec{a}_nx_n \\ &= \sum_{i=0}^{n}\vec{a}_ix_i\end{aligned}$$

We can generalize this to matrix-matrix multiplication and go one column of **V** at a time (we changed from x to v to avoid confusion in notation):

$$\begin{aligned}\mathbf{AV} &= \mathbf{A}\begin{bmatrix} | & | & & | \\ \vec{v_0} & \vec{v_0} & \cdots & \vec{v_m} \\ | & | & & | \end{bmatrix} \\ &= \begin{bmatrix} | & | & & | \\ \mathbf{A}\vec{v_0} & \mathbf{A}\vec{v_1} & \cdots & \mathbf{A}\vec{v_m} \\ | & | & & | \end{bmatrix}\end{aligned}$$

So if we apply this to the vectors from the problem $\begin{bmatrix}1\\0\end{bmatrix}$ and $\begin{bmatrix}0\\1\end{bmatrix}$, we find that our vectors combine to make the identity matrix! Plugging this and the vectors we solved for above into the equation for **AV** above:

$$\begin{aligned} A\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix} &= \begin{bmatrix}\cos\theta & \sin\theta \\ -\sin\theta & \cos\theta\end{bmatrix} \\ A &= \begin{bmatrix}\cos\theta & \sin\theta \\ -\sin\theta & \cos\theta\end{bmatrix}\end{aligned}$$

(e) $\mathbf{A} = \begin{bmatrix}1 & 1\\2 & 0\end{bmatrix}$

Solution:

We can use Gauss-Jordan method to find the inverse of the matrix.

$$\left[\begin{array}{cc|cc}1 & 1 & 1 & 0\\2 & 0 & 0 & 1\end{array}\right] \sim \left[\begin{array}{cc|cc}1 & 1 & 1 & 0\\0 & 2 & 2 & -1\end{array}\right] \sim \left[\begin{array}{cc|cc}1 & 1 & 1 & 0\\0 & 1 & 1 & -\frac{1}{2}\end{array}\right] \sim \left[\begin{array}{cc|cc}1 & 0 & 0 & \frac{1}{2}\\0 & 1 & 1 & -\frac{1}{2}\end{array}\right]$$

Inverse exists: $\mathbf{A}^{-1} = \begin{bmatrix}0 & \frac{1}{2}\\1 & -\frac{1}{2}\end{bmatrix}$

(f) $\mathbf{A} = \begin{bmatrix}1 & 0 & 0\\0 & 2 & 2\\1 & 4 & 4\end{bmatrix}$

Solution:

Inverse does not exist because the second and third column are equal, i.e., they are linearly dependent. Since the columns of the matrix are linearly dependent, the inverse does not exist.

(g) **PRACTICE:** $\mathbf{A} = \begin{bmatrix}1 & 0 & 0\\0 & -1 & 1\\0 & 1 & -1\end{bmatrix}$

Solution:

Inverse does not exist because the third column is the negative of the second column, i.e., they are linearly dependent. Since the columns of the matrix are linearly dependent, the inverse does not exist.

(h) **PRACTICE:** $\mathbf{A} = \begin{bmatrix}-1 & 1 & -\frac{1}{2}\\1 & 1 & -\frac{1}{2}\\0 & 1 & 1\end{bmatrix}$

Solution:

We can use Gauss-Jordan method to find the inverse of the matrix.

$$\left[\begin{array}{ccc|ccc} -1 & 1 & -\frac{1}{2} & 1 & 0 & 0 \\ 1 & 1 & -\frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right] \sim \left[\begin{array}{ccc|ccc} 0 & 2 & -1 & 1 & 1 & 0 \\ 1 & 1 & -\frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right] \sim \left[\begin{array}{ccc|ccc} 0 & 0 & -3 & 1 & 1 & -2 \\ 1 & 1 & -\frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$

$$\sim \left[\begin{array}{ccc|ccc} 0 & 0 & -3 & 1 & 1 & -2 \\ 1 & 1 & -\frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array}\right] \sim \left[\begin{array}{ccc|ccc} 0 & 0 & -3 & 1 & 1 & -2 \\ 1 & 0 & -\frac{1}{2} & -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ 0 & 1 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array}\right] \sim \left[\begin{array}{ccc|ccc} 0 & 0 & 1 & -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \\ 1 & 0 & -\frac{1}{2} & -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ 0 & 1 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array}\right]$$

$$\sim \left[\begin{array}{ccc|ccc} 0 & 0 & 1 & -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \\ 1 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 1 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array}\right] \sim \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 1 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 1 & -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \end{array}\right]$$

Inverse exists: $\mathbf{A}^{-1} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$

(i) **(PRACTICE)**

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

Hint 1: What do the linear (in)dependence of the rows and columns tell us about the invertibility of a matrix? Hint 2: We're reasonable people!

Solution:

Inverse does not exist because $\text{column}_1 + \text{column}_2 + \text{column}_3 = \text{column}_4$, which means that the columns are linearly dependent. Since the columns of the matrix are linearly dependent, the inverse does not exist.

4. (PRACTICE) Powers Of Nilpotent Matrices

The following matrices are examples of a special type of matrix called a nilpotent matrix. What happens to each of these matrices when you multiply it by itself repeatedly? Multiply them to find out. Why do you think these are called "nilpotent" matrices? (Of course, there is nothing magical about 3×3 or 4×4 matrices. You can have nilpotent square matrices of any dimension greater than 1.)

(a) **(PRACTICE)** Calculate $\mathbf{C}^3$ by hand. Make sure you show what $\mathbf{C}^2$ is along the way.

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

Solution:

A nilpotent matrix is a matrix that becomes all 0's when you raise it to some power, i.e. repeatedly multiply it by itself.

$$\begin{aligned}
\mathbf{C}^2 &= \begin{bmatrix}0 & 1 & 0\\0 & 0 & 2\\0 & 0 & 0\end{bmatrix}\begin{bmatrix}0 & 1 & 0\\0 & 0 & 2\\0 & 0 & 0\end{bmatrix}\\
&= \begin{bmatrix}0 & 0 & 2\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}\\
\mathbf{C}^3 &= \begin{bmatrix}0 & 1 & 0\\0 & 0 & 2\\0 & 0 & 0\end{bmatrix}\begin{bmatrix}0 & 0 & 2\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}\\
&= \begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}
\end{aligned}$$

For the purposes of this homework, the above is all you need to know. For those interested, this particular matrix can be used to describe polynomial differentiation where the variables are $\{1, x, x^2\}$:

$$\frac{d}{dx}[1] = 0$$

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

$$\frac{d}{dx}[x] = 1$$

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

$$\frac{d}{dx}\left[x^2\right] = 2x$$

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

$$\frac{d}{dx}\left[x^2 + x + 1\right] = 2x + 1$$

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

and applying it twice

$$\frac{d^2}{dx^2}\left[x^2 + x + 1\right] = 2$$

and then thrice

$$\frac{d^3}{dx^3}\left[x^2 + x + 1\right] = 0$$

Generalizing this across coefficients for each of the terms of the polynomial, applying the third derivative to a second order polynomial will always yield a result of 0!

(b) **(PRACTICE)** Calculate $\mathbf{A}^4$ by hand. Make sure you show what $\mathbf{A}^2$ and $\mathbf{A}^3$ are along the way.

$$\mathbf{A} = \begin{bmatrix} 0 & 2 & 1 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Solution:

$$\mathbf{A}^2 = \begin{bmatrix} 0 & 2 & 1 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 & 1 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A}^3 = \begin{bmatrix} 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 & 1 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 6 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A}^4 = \begin{bmatrix} 0 & 0 & 0 & 6 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 & 1 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(c) **(PRACTICE)** Calculate $\mathbf{B}^4$. You are allowed to use iPython to find $\mathbf{B}^2$ and $\mathbf{B}^3$—write out in your homework what they are. Calculate the final multiplication by hand.

$$\mathbf{B} = \begin{bmatrix} 3 & 4 & 2 & 1 \\ -5 & -6 & -3 & -1 \\ 6 & 7 & 3 & 2 \\ 2 & 2 & 1 & 0 \end{bmatrix}$$

Solution:

$$\mathbf{B}^2 = \begin{bmatrix} 3 & 4 & 2 & 1 \\ -5 & -6 & -3 & -1 \\ 6 & 7 & 3 & 2 \\ 2 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} 3 & 4 & 2 & 1 \\ -5 & -6 & -3 & -1 \\ 6 & 7 & 3 & 2 \\ 2 & 2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 4 & 1 & 3 \\ -5 & -7 & -2 & -5 \\ 5 & 7 & 2 & 5 \\ 2 & 3 & 1 & 2 \end{bmatrix}$$

$$\mathbf{B}^3 = \begin{bmatrix} 3 & 4 & 1 & 3 \\ -5 & -7 & -2 & -5 \\ 5 & 7 & 2 & 5 \\ 2 & 3 & 1 & 2 \end{bmatrix} \begin{bmatrix} 3 & 4 & 2 & 1 \\ -5 & -6 & -3 & -1 \\ 6 & 7 & 3 & 2 \\ 2 & 2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ -2 & -2 & 0 & -2 \\ 2 & 2 & 0 & 2 \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

$$\mathbf{B}^4 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ -2 & -2 & 0 & -2 \\ 2 & 2 & 0 & 2 \\ 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 & 4 & 2 & 1 \\ -5 & -6 & -3 & -1 \\ 6 & 7 & 3 & 2 \\ 2 & 2 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

5. Properties of Pump Systems

Throughout this problem, we will consider a system of reservoirs connected to each other through pumps. An example system is shown below in Figure 1, represented as a graph. Each node in the graph is marked with a number and represents a reservoir. Each edge in the graph represents a pump which moves a fraction of the water from one reservoir to the next at every time step. The fraction of water is written on top of the edge.

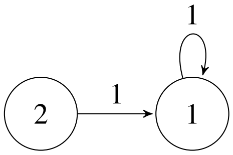


Figure 1: Pump system

(a) Consider the system of pumps shown above in Figure 1. Let $x_i[n]$ represent the amount of water in reservoir i at time step n. Find a system of equations that represents every $x_i[n+1]$ in terms of all the different $x_i[n]$. **Solution:**

$$x_1[n+1] = x_1[n] + x_2[n]$$
$$x_2[n+1] = 0$$

(b) For the system shown in Figure 1, find the associated state transition matrix. That is find the matrix $\mathbf{A}$ such that:

$$\vec{x}[n+1] = \mathbf{A}\vec{x}[n], \text{ where } \vec{x}[n] = \begin{bmatrix} x_1[n] \\ x_2[n] \end{bmatrix}$$

Solution:

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

(c) Suppose that the reservoirs are initialized to the following water levels: $x_1[0] = 0.5, x_2[0] = 0.5$. In a completely alternate universe, the reservoirs are initialized to the following water levels: $x_1[0] = 0.3, x_2[0] = 0.7$. For both initial states, what are the water levels at timestep 1 ($\vec{x}[1]$)? Use your answer from part (b) to compute your solution.

Solution:

$$\vec{x}[1] = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1[0] \\ x_2[0] \end{bmatrix} = \begin{bmatrix} x_1[1] \\ x_2[1] \end{bmatrix}$$

$$\vec{x}[1] = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \vec{x}[1] = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0.3 \\ 0.7 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

(d) If you observe the reservoirs at timestep 1, can you figure out what the initial ($\vec{x}[0]$) water levels were? Why or why not?

Solution:

No, at timestep 1, configuration 1 and 2 are indistinguishable. This implies that the system is inherently linearly dependent, and so the transformation matrix is noninvertible and it's impossible to ascertain how the water was distributed in earlier states.

(e) Now generalize: if there exists a state transition matrix where two different initial state vectors lead to the same water levels/state vectors at a timestep in the future, can you recover the initial water levels? Prove your answer.
(*Hint: What does this say about the matrix* $\mathbf{A}$*?*)

Solution:

We are told that two different initial states, $\vec{x}[0]$ and $\vec{y}[0]$, lead to the same resulting state $\vec{x}[1]$.

$$\mathbf{A}\vec{x}[0] = \vec{x}[1] \qquad \mathbf{A}\vec{y}[0] = \vec{x}[1]$$

If we can recover the initial water levels, this would mean that there is some operation that can be performed on the resulting state vector to yield the previous state. This operation would, by definition, be the inverse of the matrix $\mathbf{A}$.

$$\vec{x}[0] = \mathbf{A}^{-1}\vec{x}[1] \qquad \vec{y}[0] = \mathbf{A}^{-1}\vec{x}[1]$$

However, we can see that the above two equations contradict each other. Therefore, $\mathbf{A}$ is not invertible, and you cannot recover the previous states from the resulting state.

(f) Suppose that we have 3 reservoirs and that there is a state transition matrix such that the entries of each column vector sum to one. Let the total amount of water in the system be s at timestep n. Show that the total amount of water at timestep $n+1$ will also be s. What is the physical interpretation about the total amount of water in the system? Prove this for 3 reservoirs first, then generalize to k reservoirs.
Hint: Consider the state vector at time n. **Solution:**

Let $\vec{x}[n] = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix}$ be the amount of water in each resevoir at timestep n. We know:

$$\sum_{i=1}^{k} x_i[n] = s$$

Let $\vec{a}_j$ be the j-th column of the state transition matrix $\mathbf{A}$.

$$\mathbf{A} = \begin{bmatrix} \vec{a}_1 & \vec{a}_2 & \cdots & \vec{a}_k \end{bmatrix}$$

We know that every column of $\mathbf{A}$ sums to one, so we know for all j,

$$\sum_{i=1}^{k} a_{ij} = 1$$

Now, consider the product $\mathbf{A}\vec{x}[n]$:

$$\mathbf{A}\vec{x}[n] = \begin{bmatrix} \vec{a}_1 & \vec{a}_2 & \cdots & \vec{a}_k \end{bmatrix} \begin{bmatrix} x_1[n] \\ x_2[n] \\ \vdots \\ x_k[n] \end{bmatrix} = \sum_{j=1}^{k} x_j[n]\vec{a}_j = \vec{x}[n+1]$$

Let's consider the sum of the elements in $\vec{x}[n+1]$:

$$\sum_{i=1}^{k} x_i[n+1] = \sum_{i=1}^{k} \left(\sum_{j=1}^{k} x_j[n] a_{ij} \right)$$

Switching the order of summation, so we sum along the columns first:

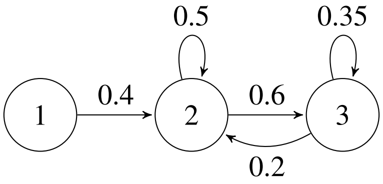
$$\sum_{i=1}^{k} \left(\sum_{j=1}^{k} x_j[n] a_{ij} \right) = \sum_{j=1}^{k} x_j[n] \left(\sum_{i=1}^{k} a_{ij} \right) = \sum_{j=1}^{k} x_j[n] = s$$

Since the sum of the elements of the vector $\vec{x}[n+1]$ is still s, the amount of water in the system does not change from one step to the next.

The physical interpretation of this statement is that the total amount of water in the system remains the same. There are no "leaks/drains" or "inlets/gains" in the system.

(g) **(PRACTICE)** Set up the state transition matrix $\mathbf{A}$ for the system of pumps shown below. Compute the sum of the columns of the state transition matrix. Is it greater than/less than/equal to 1? Explain what this $\mathbf{A}$ matrix physically implies about the total amount of water in this system.

Note: If there is no "self-arrow/self-loop," then the water does not return.



Solution:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 \\ 0.4 & 0.5 & 0.2 \\ 0 & 0.6 & 0.35 \end{bmatrix}$$

Note that the entries in the columns do *not* sum to one. This is physically interpreted as a "leak" – i.e., the total amount of water is not conserved.

(h) **(PRACTICE)** There is a state transition matrix where the entries of its rows sum to one. Prove that applying this system to a uniform vector will return the same uniform vector. A uniform vector is a vector whose elements are all the same.

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & a_{ij} & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x \\ \vdots \\ x \end{bmatrix} = \begin{bmatrix} x \\ \vdots \\ x \end{bmatrix}$$

Solution:

Consider the row interpretation of matrix multiplication: Each b_i is equal to the dot product of the row vector $\vec{\alpha}_i^T$ and the vector $\vec{x}$.

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & a_{ij} & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \vec{\alpha}_1^T \\ \vec{\alpha}_2^T \\ \vdots \\ \vec{\alpha}_n^T \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

For example, let's look at the first row b_1.

$$b_1 = \vec{\alpha}_1^T \vec{x} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

All values in the vector $\vec{x}$ are equal to x.

$$\begin{aligned} b_1 &= \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \end{bmatrix} \begin{bmatrix} x \\ x \\ \vdots \\ x \end{bmatrix} \\ &= a_{11}x + a_{12}x + \cdots + a_{1n}x \\ &= x(a_{11} + a_{12} + \cdots + a_{1n}) \end{aligned}$$

Remember that the values in $\vec{\alpha}_1^T$ sum to 1.

$$b_1 = x(a_{11} + a_{12} + \cdots + a_{1n}) = x \cdot 1 = x$$

As we can see, $b_1 = x$. This property carries through for each row, so all of them sum to 1. Thus $\vec{b} = \vec{x}$, which is what we wanted to prove.

6. Audio File Matching

Each day a wide variety of quantities we interact with can be expressed as vectors. For example, an audio clip or a sound wave (continuous function in time) can be sampled at regular intervals to make a discrete sequence of values and represented in vector form.

This problem explores using inner products for measuring similarity between two sound signals represented as vectors. The same ideas here will be further developed in the third module of EE16A where we will learn about Locationing and GPS.

Let us consider a very simplified model for an audio signal; one that is composed of just two tones. One tone is represented by the value -1 and the other by the value $+1$. A vector of length n makes up the audio file.

(a) Say we want to compare two audio files of the same length n to decide how similar they are. First consider two vectors that are exactly identical $\vec{X}_1 = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^T$ and $\vec{X}_2 = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^T$. What is the inner product/dot product of these two vectors, i.e. $\vec{X}_1^T \vec{X}_2$? What if $\vec{X}_1 = \begin{bmatrix} 1 & 1 \cdots 1 \end{bmatrix}^T$ and $\vec{X}_2 = \begin{bmatrix} 1 & -1 & 1 & -1 \cdots 1 & -1 \end{bmatrix}^T$ (where the length of the vector is an even number)? For pairs of vectors of length n made of ±1's, does a larger dot product imply that the vectors are more similar or less similar?

Solution: From lecture, we define the dot product as the following:

$$\vec{u}^T \vec{v} = \sum_{i=1}^{n} u_i v_i$$

For the case where the two vectors $\vec{X}_1 = \vec{X}_2 = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^T$,

$$\vec{X}_1^{\ T} \vec{X}_2 = \sum_{i=1}^{n} 1(1) = n$$

When $\vec{X}_2 = \begin{bmatrix}1 & -1 & 1 & -1 & \cdots & 1 & -1\end{bmatrix}^T$ (with an even number of elements)

$$\vec{X}_1^T \vec{X}_2 = 1(1) + 1(-1) + \cdots + 1(1) + 1(-1) = 0$$

For pairs of vectors length n with values of ± 1, the larger the dot product, the more similar the vectors are. This notion of the dot product as a gauge of similarity between vectors will persist throughout the course.

In general, a dot product with a very negative value means the vectors are very different. However, it turns out humans are unable to perceive the "sign" of sound, so hearing a vector $\vec{X}$ sounds identical to hearing a vector $-\vec{X}$! That means that for the purposes of this problem, it's the *absolute value* of the dot product we care about. Don't take off points if you didn't mention absolute value.

(b) Next suppose we want to search for a short audio clip in a longer one. We might want to do this for an application like *Shazam* to be able to identify a song from a signature tune. Consider the vector of length 8, $\vec{X} = \begin{bmatrix}1 & -1 & -1 & 1 & 1 & 1 & -1 & 1\end{bmatrix}^T$. Let us label the elements of $\vec{X}$ so that $\vec{X} = \begin{bmatrix}x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8\end{bmatrix}^T$. Our goal is to find the short segment $\vec{Y} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^T$ in the longer vector (i.e. we want to find i, such that the sequence represented by $\begin{bmatrix}x_i & x_{i+1} & x_{i+2}\end{bmatrix}^T$ is the closest to $\vec{Y}$). Come up with an approach to do this. Can your approach be written as a matrix vector multiplication $\mathbf{A}\vec{x}$ where $\mathbf{A}$ is 6×8 and $\vec{x}$ is 8×1? Applying your technique, which i gives the best match for $\vec{Y} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^T$?

Solution: For each length-3 sub-sequence of $\vec{X}$ say $\vec{X}_i = \begin{bmatrix}x_i & x_{i+1} & x_{i+2}\end{bmatrix}^T$ (the subsequence which starts at position i), take the dot product of $\vec{X}_i$ and $\vec{Y}$. The length-3 sub-sequences $\vec{X}_i$ with the maximum-magnitude dot product with $\vec{Y}$ gives us the closest match with $\vec{Y}$. Computing the dot products:

$$i = 1 : \begin{bmatrix}1 & -1 & -1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = -1$$

$$i = 2 : \begin{bmatrix}-1 & -1 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = -1$$

$$i = 3 : \begin{bmatrix}-1 & 1 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = 1$$

$$i = 4 : \begin{bmatrix}1 & 1 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = 3$$

$$i = 5 : \begin{bmatrix}1 & 1 & -1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = 1$$

$$i = 6 : \begin{bmatrix}1 & -1 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = 1$$

We know that matrix-vector multiplication can be written as a series of dot products

$$A\vec{x} = \begin{bmatrix} - & \vec{a}_1{}^T & - \\ - & \vec{a}_2{}^T & - \\ & \vdots & \\ - & \vec{a}_n{}^T & - \end{bmatrix} \vec{x} = \begin{bmatrix} \vec{a}_1{}^T\vec{x} \\ \vec{a}_2{}^T\vec{x} \\ \vdots \\ \vec{a}_n{}^T\vec{x} \end{bmatrix}$$

so we can also implement the 6 dot products in one matrix multiplication. For instance, if $\vec{X} = \begin{bmatrix} x_1 & x_2 & \dots & x_8 \end{bmatrix}^T$ and $\vec{Y} = \begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix}^T$, the dot products (or correlation) $z_i = \vec{X}_i \cdot \vec{Y}$ can be represented as:

$$\begin{bmatrix} y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{bmatrix}$$

We then pick the z_i with the largest magnitude and the corresponding $\vec{X}_i$ gives us the required substrings. This is connected to the ideas of cross-correlation which will be explored later on.

Note:

We sometimes extend beyond the length of $\vec{X}$—that is, we can include $i = 7, 8$ and $i = -1, 0$. Depending on the situation, we can either (1) assume all elements outside those in the original vector $\vec{X}$ are 0, (we call this "zero-padding"), or (2) take a cyclical approach to the elements of $\vec{X}$. If you did this, the equivalent matrix-vector multiplication will not have precisely the dimensions we specified. However, **if you correctly performed either the cyclical correlation or zero-padding and your matrix-vector multiplication matches your equations, you should give yourself full credit**.

Case 1: Zero Padding

In this case, we extend $\vec{X}$ beyond the signal of interest (the ±1 values) to rewrite it

$$\vec{X}_{\text{zero-padded}} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

For clarity, we maintain the same indexing as above.

$$i=-1:\begin{bmatrix}0 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix}=1$$

$$i=0:\begin{bmatrix}0 & 1 & -1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix}=0$$

$$i=7:\begin{bmatrix}-1 & 1 & 0\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix}=0$$

$$i=8:\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix}=1$$

This doesn't change the location of where the match occurs, but it will change your matrix-vector multiplication. Both of the following can be considered correct, though the resulting vector will be differently sized:

$$\begin{bmatrix}
y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3
\end{bmatrix}\vec{X}_{\text{zero-padded}}$$

$$\begin{bmatrix}
y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0\\
y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0\\
0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0\\
0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0\\
0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0\\
0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0\\
0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3\\
0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1
\end{bmatrix}\vec{X}$$

Case 2: Cyclical Correlation

Sometimes, instead of saying that $\vec{X}$ simply ends at its last element, we can act as though there's another copy of $\vec{X}$ attached before and after it. This situation applies when the sound defined by $\vec{X}$ is repeated several times in a row (see: the pop music industry), and we can describe this by extending $\vec{X}$ like so:

$$\vec{X}_{\text{cyclical}} = \begin{bmatrix} -1 \\ 1 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 1 \\ -1 \end{bmatrix}$$

All we've done here is take take the last two elements of $\vec{X}$ and placed them at the front, and taken the first two elements of $\vec{X}$ and placed them at the end. Maintaining the same indexing as the original vector,

$$i = -1 : \begin{bmatrix} -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 1$$

$$i = 0 : \begin{bmatrix} 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 1$$

$$i = 7 : \begin{bmatrix} -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 1$$

$$i = 8 : \begin{bmatrix} 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 1$$

Similar to the case with zero padding, there are several ways to write the matrix-vector multiplication which will provide the same information, albeit with differently sized output vectors.

$$\begin{bmatrix} y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 \end{bmatrix} \vec{X}_{\text{cyclical}}$$

$$\begin{bmatrix} y_3 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 \\ y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & y_1 \\ y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 \\ y_3 & 0 & 0 & 0 & 0 & 0 & y_1 & y_2 \\ y_2 & y_3 & 0 & 0 & 0 & 0 & 0 & y_1 \end{bmatrix} \vec{X}$$

(c) **(PRACTICE)** Now suppose our vector was represented using integers and not just by 1 and -1. We want to find the subsequence of the longer vector that is most similar to that of a sample vector. Say we wanted to locate the sequence closest in direction to $\vec{Y} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$ within $\vec{X} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix}^T$. Can you explain why the approach in part (b) won't work? Consider using the norm/magnitude of the short segments (the norm/magnitude of a vector is defined as $\|\vec{z}\| = \sqrt{z_0^2 + z_1^2 + \ldots + z_n^2}$). How might you use this quantity to modify your approach for the new vectors to focus on the direction rather than the magnitude of the vectors?

Solution: Applying the approach in part (b), we get the best match to be $\begin{bmatrix} 6 & 7 & 8 \end{bmatrix}^T$ as this has the largest dot product with $\vec{Y} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$.

One way to modify the previous approach is by considering how close the "directions" of $\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$ and any length-3 substring of $\vec{X}$ is. The unit vector in the direction of $\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$ is $\vec{Y}_u = \frac{1}{\sqrt{14}} \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$ and the unit vector in the direction of any length-3 substring $\vec{X}_i = \begin{bmatrix} x_i & x_{i+1} & x_{i+2} \end{bmatrix}^T$ is then given by $\vec{U}_i = \frac{1}{\|X_i\|} \begin{bmatrix} x_i & x_{i+1} & x_{i+2} \end{bmatrix}^T$. The unit vector $\vec{U}_i$ which has the maximum-magnitude dot product with $\vec{Y}_u$ (i.e., $\vec{U}_i \cdot \vec{Y}_u$) is the one most aligned with the vector $\vec{Y}$. Thus in our example for $\vec{X} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix}^T$ and $Y = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$, the unit vector of the length-3 substring $\vec{X}_1 = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$ has the maximum dot product ($\vec{U}_1 \cdot \vec{Y}_u = 1$).
What are other interesting ways to achieving this?

(d) In the IPython notebook, `prob3.ipynb`, complete part 1. **Solution:** See the solutions in the ipython notebook.

(e) In the IPython notebook, `prob3.ipynb`, complete part 2. **Solution:** See the solutions in the ipython notebook.

7. Write Your Own Question And Provide a Thorough Solution.

Writing your own problems is a very important way to really learn material. The famous "Bloom's Taxonomy" that lists the levels of learning is: Remember, Understand, Apply, Analyze, Evaluate, and Create. Using what you know to create is the top level. We rarely ask you any homework questions about the lowest level of straight-up remembering, expecting you to be able to do that yourself (e.g. making flashcards). But we don't want the same to be true about the highest level. As a practical matter, having some practice at trying to create problems helps you study for exams much better than simply counting on solving existing practice problems. This is because thinking about how to create an interesting problem forces you to really look at the material from the perspective of those who are going to create the exams. Besides, this is fun. If you want to make a boring problem, go ahead. That is your prerogative. But it is more fun to really engage with the material, discover something interesting, and then come up with a problem that walks others down a journey that lets them share your discovery. You don't have to achieve this every week. But unless you try every week, it probably won't ever happen.

8. Homework Process and Study Group

Who else did you work with on this homework? List names and student ID's. (In case of homework party, you can also just describe the group.) How did you work on this homework?

Solution:

I worked on this homework with...

I first worked by myself for 2 hours, but got stuck on problem 5, so I went to office hours on...

Then I went to homework party for a few hours, where I finished the homework.