

## Homework 7: Projections; Regression with Linear Algebra

EECS 245, Fall 2025 at the University of Michigan

due Friday, October 24th, 2025 at 11:59PM Ann Arbor Time

Write your solutions to the following problems by either typing them up or handwriting them on another piece of paper. Homeworks are due to Gradescope by 11:59PM on the due date. See the [syllabus](#) for details on the slip day policy.

Homework will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should always explain and justify your conclusions, using sound reasoning. Your goal should be to convince the reader of your assertions. If a question does not require explanation, it will be explicitly stated.

Before proceeding, make sure you're familiar with the [collaboration policy](#).

Total Points:  $35 + 13 + 14 + 27 = 89$

**Note:** This homework is a bit different from previous homeworks in that it only has 4 problems, but each one is rather involved and requires you to read a fair bit. **Tip:** Do the problems in order, since they build upon each other.

### Problem 1: Projecting onto the Column Space (35 pts)

The big idea in [Chapter 2.10](#), which we will also revisit in Chapter 3.1, is that of projecting a vector  $\vec{y} \in \mathbb{R}^n$  onto the column space of an  $n \times d$  matrix  $X$ . That is, unlike in previous homeworks, we **don't** assume  $\vec{y}$  is a linear combination of  $X$ 's columns, and instead aim to find the vector in  $\text{colsp}(X)$  that is closest to  $\vec{y}$ .

As we will show in [Chapter 2.10](#), if  $X$ 's columns are the vectors  $\vec{x}^{(1)}, \vec{x}^{(2)}, \dots, \vec{x}^{(d)}$ , then the projection of  $\vec{y}$  onto  $\text{colsp}(X)$  is the vector

$$\vec{p} = X\vec{w}^* = w_1^* \vec{x}^{(1)} + w_2^* \vec{x}^{(2)} + \dots + w_d^* \vec{x}^{(d)}, \quad \text{where} \quad \underbrace{\vec{w}^* = (X^T X)^{-1} X^T \vec{y}}_{\text{if } X^T X \text{ is invertible}}$$

The vector  $\vec{w}^*$  minimizes the norm of the error vector,  $\vec{e} = \vec{y} - \vec{p}$ , and as we will see in coming problems and homeworks, it contains **optimal model parameters** for linear regression, when we fill our  $X$  (carefully) with our input variables and  $\vec{y}$  with our output variables.

Taking another look at the formula  $\vec{p} = X\vec{w}^*$ , we see that it's equivalent to

$$\vec{p} = X\vec{w}^* = X(X^T X)^{-1} X^T \vec{y} = P\vec{y}$$

where  $P = X(X^T X)^{-1} X^T$  is called the **projection matrix**. Multiplying  $P\vec{y}$  is equivalent to projecting  $\vec{y}$  onto  $\text{colsp}(X)$ . In this problem, you'll explore properties of  $P$  and  $\vec{w}^*$ .

- a) (4 pts) In this part only, suppose  $X$  is an  $n \times 1$  matrix, i.e. it is a vector. Then,
- (i) What is the value of  $\vec{w}^*$ , and how does it relate to what we learned in [Chapter 2.3](#)?
  - (ii) What is the value of the matrix  $P$ , and how does it relate to what we learned in [Homework 5, Problem 4](#)?

- b) (4 pts) Show that  $P$  is both symmetric (meaning that  $P^T = P$ ) and idempotent (meaning that  $P^2 = P$ ). Then, explain in English how  $P$ 's idempotence relates to the linear transformation of projecting  $\vec{y}$  onto  $\text{colsp}(X)$ .

- c) (2 pts) Recall that  $X$  is an  $n \times d$  matrix (meaning it's not necessarily square), which makes  $P = X(X^T X)^{-1} X^T$  an  $n \times n$  matrix.

Fill in the blanks:  $X^T X$  is invertible if and only if  $X$ 's columns are .....

- d) (2 pts) In the rare case that  $X$  is an  $n \times n$  square matrix, and  $\text{rank}(X) = n$ , what is  $P$ ? What does this say about the relationship between  $\vec{y}$ ,  $\vec{p}$ , and  $\text{colsp}(X)$ ?

*Hint: Use the fact that  $(AB)^{-1} = B^{-1} A^{-1}$ .*

- e) (8 pts) In (i) and (ii), find  $\vec{w}^*$ ,  $\vec{p}$ , and  $\vec{e} = \vec{y} - \vec{p}$ , and verify that  $\vec{e}$  is orthogonal to  $\text{colsp}(X)$  by showing that it is orthogonal to each of  $X$ 's columns.

(i)  $X = \begin{bmatrix} 2 & 1 \\ 0 & -3 \\ 0 & 0 \end{bmatrix}$ ,  $\vec{y} = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}$

(ii)  $X = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & 0 \end{bmatrix}$ ,  $\vec{y} = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}$

- (iii) In **only one** of the subparts above, it is true that the sum of the components of  $\vec{e}$  is 0. Which one, and why? This is a **hugely** important result, and one that will 100% appear on Midterm 2. *Hint: The answer is not that  $\vec{y}$  is in  $\text{colsp}(X)$ ; in both parts, it is **not**.*

- f) (2 pts) What linear combination of  $\begin{bmatrix} 1 \\ 2 \\ -1 \\ 0 \end{bmatrix}$  and  $\begin{bmatrix} 1 \\ 0 \\ 1 \\ -3 \end{bmatrix}$  is closest to  $\begin{bmatrix} 3 \\ 1 \\ 2 \\ 4 \end{bmatrix}$ ?

- g) (3 pts) Find the point on the plane  $x - y - 2z = 0$  that is closest to the point  $(5, 0, 3)$ .

*Hint: Start by thinking of this as a projection problem. What are  $X$  and  $\vec{y}$ ?*

h) (6 pts) Consider the matrix

$$A = \begin{bmatrix} 3 & 6 & 6 \\ 4 & 8 & 8 \end{bmatrix}$$

- (i) How many vectors  $\vec{w}$  are there that minimize  $\|\vec{y} - A\vec{w}\|^2$ ? 0, 1, finitely many, or infinitely many?
  - (ii) Find the matrix  $P_c$  that projects vectors in  $\mathbb{R}^2$  onto the column space of  $A$ . *Hint: The formula for  $P$  will fail here, since  $A^T A$  will not be invertible. Start by explaining why  $A^T A$  is not invertible. Then, look at Homework 5 for inspiration.*
  - (iii) Find the matrix  $P_r$  that projects vectors in  $\mathbb{R}^3$  onto the row space of  $A$ .
  - (iv) Find the product  $P_c A P_r$  and explain the result.
- i) (4 pts) Suppose that  $Q$  is an  $n \times d$  matrix whose columns are orthonormal, meaning that its columns are orthogonal to each other and have length 1. Note that we are not assuming that  $Q$ 's rows are orthonormal too, meaning that  $Q$  is not necessarily an orthogonal matrix using the definition from Homework 5 or Chapter 2.9.

The fact that  $Q$ 's columns are orthonormal greatly simplifies the formulas involved in projecting  $\vec{y}$  onto  $\text{colsp}(Q)$ .

- (i) What is the value of  $\vec{w}^*$ ?
- (ii) What is the value of  $\vec{p}$ ?
- (iii) If  $Q$ 's rows were orthonormal too, how would the answers to (i) and (ii) simplify further, and why?

## Problem 2: Same, but Different (13 pts)

In [Chapter 1.4](#), we were introduced to one of many formulas for the optimal slope,  $w_1^*$ , and optimal intercept,  $w_0^*$ , for the simple linear regression model  $h(x_i) = w_0 + w_1 x_i$  when using squared loss:

$$w_1^* = r \frac{\sigma_y}{\sigma_x} \quad w_0^* = \bar{y} - w_1^* \bar{x}$$

The end goal of Chapter 2 has been to give us the tools to revisit the simple linear regression model in terms of linear algebra, so that we can extend our model to allow for multiple input variables. As we will see in Chapter 3.1 and in Lecture 15, the solution is to define the  $n \times 2$  “**design matrix**”  $X$  and observation vector  $\vec{y} \in \mathbb{R}^n$  as follows:

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}, \quad \vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Then, leveraging the results from Problem 1 of this homework (and in Chapter 2.10), we can express the optimal parameter vector  $\vec{w}^*$  as  $\vec{w}^* = (X^T X)^{-1} X^T \vec{y} = \begin{bmatrix} w_0^* \\ w_1^* \end{bmatrix}$ . Optimal predictions come from multiplying the design matrix by the optimal parameter vector, i.e.  $X \vec{w}^*$ .

**It’s not immediately obvious why the components of  $\vec{w}^*$  should have anything to do with the correlation, means of  $x$  and  $y$ , and standard deviations of  $x$  and  $y$ .** In this problem, we will prove that both of these formulations are equivalent, for any dataset  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ .

- (5 pts) Express the matrix  $(X^T X)^{-1}$  using constants and/or summations involving  $x_i$  and/or  $y_i$ .
- (3 pts) Prove that

$$(X^T X)^{-1} = \frac{1}{n\sigma_x^2} \begin{bmatrix} \sigma_x^2 + \bar{x}^2 & -\bar{x} \\ -\bar{x} & 1 \end{bmatrix}$$

*Hint: Start by proving  $\sum_{i=1}^n x_i^2 = n\sigma_x^2 + n\bar{x}^2$ .*

- (5 pts) Finally, prove that

$$(X^T X)^{-1} X^T \vec{y} = \begin{bmatrix} \bar{y} - r \frac{\sigma_y}{\sigma_x} \bar{x} \\ r \frac{\sigma_y}{\sigma_x} \end{bmatrix}$$

*Hint: Start by proving that  $\sum_{i=1}^n x_i y_i = nr\sigma_x\sigma_y + n\bar{x}\bar{y}$ .*

Note that the second component of the vector above is  $w_1^* = r \frac{\sigma_y}{\sigma_x}$  and the first component of the vector above is  $w_0^* = \bar{y} - r \frac{\sigma_y}{\sigma_x} \bar{x} = \bar{y} - w_1^* \bar{x}$ , as we first saw in Chapter 1.4! Beautiful.

### Problem 3: Billy the Waiter (14 pts)

This problem involves writing code and submitting it to the Gradescope autograder. The goal of this problem is to give you a taste of how linear algebra can be used to implement linear regression in code, and show you how to build models that involve multiple features (including categorical variables).

There are two ways to access the supplemental Jupyter Notebook:

- **Option 1:** Click [here](#) to open `hw07.ipynb` on DataHub. Before doing so, read the instructions on the [Tech Support](#) page on how to use the DataHub.
- **Option 2:** Set up a Jupyter Notebook environment locally, use `git` to clone our course repository, and open `homeworks/hw07/hw07.ipynb`. For instructions on how to do this, see the [Tech Support](#) page of the course website.

**This problem is entirely autograded; to receive credit for Problem 3 of this homework, you'll need to submit your completed notebook to the autograder on Gradescope.** Your submission time for Homework 7 is the **latter** of your PDF and code submission times.

#### Problem 4: Orthogonalization (27 pts)

Recall, a set of vectors  $\vec{q}_1, \vec{q}_2, \dots, \vec{q}_d \in \mathbb{R}^n$  are orthonormal if they are pairwise orthogonal, meaning  $\vec{q}_i \cdot \vec{q}_j = 0$  for all  $i \neq j$ , and each vector is a unit vector, meaning  $\|\vec{q}_i\| = 1$  for all  $i$ .

In this problem, we'll discover how to **turn a linearly independent set of vectors into an orthonormal set of vectors with the same span**, i.e. how to "orthogonalize" a set of vectors.

$$\begin{array}{ccc} \vec{v}_1, \vec{v}_2, \dots, \vec{v}_d & \rightarrow & \vec{q}_1, \vec{q}_2, \dots, \vec{q}_d \\ \text{linearly independent set of vectors} & & \text{orthonormal set of vectors with the same span} \end{array}$$

Orthonormal vectors (and matrices containing them) are convenient to work with. For example, if  $Q$ 's columns are orthonormal, then  $Q^T Q$ , the matrix containing the dot products of the columns of  $Q$ , is the identity matrix.

The algorithm that produces this orthonormal set of vectors is called the **Gram-Schmidt process**. It exploits the fact that **when you project  $\vec{y}$  onto  $\vec{x}$ , the error vector  $\vec{e} = \vec{y} - \vec{p}$  is orthogonal to  $\vec{x}$** . Before proceeding, revisit the section in [Chapter 2.3 titled "Orthogonal Decomposition"](#) for a useful visualization.

To illustrate how the algorithm works, let's use as an example the vectors

$$\vec{v}_1 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \quad \vec{v}_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad \vec{v}_3 = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$$

These are three linearly independent vectors in  $\mathbb{R}^3$ , though they are not orthogonal. These vectors span some subspace  $S$ . Our goal is to find an orthonormal set of vectors that spans the same  $S$ . (In this case,  $S$  is all of  $\mathbb{R}^3$ , but in general this process works even if  $d < n$ .)

In what follows, let  $\text{proj}_{\vec{x}}(\vec{y})$  be the projection of  $\vec{y}$  onto  $\vec{x}$ , i.e.  $\text{proj}_{\vec{x}}(\vec{y}) = \frac{\vec{y} \cdot \vec{x}}{\vec{x} \cdot \vec{x}} \vec{x}$ .

- **Iteration 1:** Set  $\vec{Q}_1 = \vec{v}_1$ .

In the first iteration, we simply take the first vector  $\vec{v}_1$  and copy it to  $\vec{Q}_1$ . From now on, each new vector will be constructed to be orthogonal to all previously constructed  $\vec{Q}_i$ 's.

$$\vec{Q}_1 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}.$$

- **Iteration 2:** Set  $\vec{Q}_2 = \vec{v}_2 - \text{proj}_{\vec{Q}_1}(\vec{v}_2)$

$\vec{Q}_2$  is the same as the error vector from projecting  $\vec{v}_2$  onto  $\vec{Q}_1$ , which we know is orthogonal to  $\vec{Q}_1$ . Don't believe me? Find  $\vec{Q}_2 \cdot \vec{Q}_1$  just by using the definition of  $\vec{Q}_2$  above.

$$\vec{Q}_2 = \underbrace{\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}}_{\vec{v}_2} - \underbrace{\frac{\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}}{\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}}} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} - \frac{2}{3} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1/3 \\ 2/3 \\ 1/3 \end{bmatrix}$$

- **Iteration 3:** Set  $\vec{Q}_3 = \vec{v}_3 - \text{proj}_{\vec{Q}_1}(\vec{v}_3) - \text{proj}_{\vec{Q}_2}(\vec{v}_3)$

When constructed this way,  $\vec{Q}_3$  is orthogonal to both  $\vec{Q}_1$  and  $\vec{Q}_2$ . Think of it this way:  $\text{span}(\{\vec{Q}_1, \vec{Q}_2\})$  is a plane in  $\mathbb{R}^3$ ; after projecting  $\vec{v}_3$  onto this plane, the remaining part of  $\vec{v}_3$  that is orthogonal to the plane is  $\vec{Q}_3$ . If that doesn't make sense, compute  $\vec{Q}_3 \cdot \vec{Q}_1$  and  $\vec{Q}_3 \cdot \vec{Q}_2$  and see for yourself what we get!

$$\vec{Q}_3 = \underbrace{\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}}_{\vec{v}_3} - \underbrace{\frac{\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}}{\begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}}} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} - \underbrace{\frac{\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 1/3 \\ 2/3 \\ 1/3 \end{bmatrix}}{\begin{bmatrix} 1/3 \\ 2/3 \\ 1/3 \end{bmatrix} \cdot \begin{bmatrix} 1/3 \\ 2/3 \\ 1/3 \end{bmatrix}}} \begin{bmatrix} 1/3 \\ 2/3 \\ 1/3 \end{bmatrix} = \begin{bmatrix} -1/2 \\ 0 \\ 1/2 \end{bmatrix}$$

If there were more  $\vec{v}_i$ 's, we'd continue this process, each time constructing a new  $\vec{Q}_i$  that is orthogonal to all previously constructed  $\vec{Q}_i$ 's by "subtracting off" the parts we've already accounted for through the earlier  $\vec{Q}_i$ 's.

Now,  $\vec{Q}_1, \vec{Q}_2, \vec{Q}_3$  are **orthogonal** to one another, but they are not yet unit vectors. To make them unit vectors, we simply need to divide each by its length.

$$\vec{q}_1 = \frac{\vec{Q}_1}{\|\vec{Q}_1\|} = \begin{bmatrix} 1/\sqrt{3} \\ -1/\sqrt{3} \\ 1/\sqrt{3} \end{bmatrix}, \quad \vec{q}_2 = \frac{\vec{Q}_2}{\|\vec{Q}_2\|} = \begin{bmatrix} 1/\sqrt{6} \\ 2/\sqrt{6} \\ 1/\sqrt{6} \end{bmatrix}, \quad \vec{q}_3 = \frac{\vec{Q}_3}{\|\vec{Q}_3\|} = \begin{bmatrix} -1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \end{bmatrix}$$

Now, the vectors  $\vec{q}_1, \vec{q}_2, \vec{q}_3$  are **orthonormal** to one another, and they span the same subspace  $S$  as the vectors  $\vec{v}_1, \vec{v}_2, \vec{v}_3$ !

In parts **a)** through **d)**, we'll refer to the vectors  $\vec{v}_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ ,  $\vec{v}_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$ , and  $\vec{v}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$ .

- a)** (6 pts) By hand, apply the Gram-Schmidt process to the vectors  $\vec{v}_1, \vec{v}_2, \vec{v}_3$  to find an orthonormal set of vectors  $\vec{q}_1, \vec{q}_2, \vec{q}_3$ , and show all of your work.

Then, create the matrix  $Q = \begin{bmatrix} | & | & | \\ \vec{q}_1 & \vec{q}_2 & \vec{q}_3 \\ | & | & | \end{bmatrix}$  and confirm that  $Q^T Q = I$ , but that  $Q Q^T \neq I$ .

- b)** (3 pts) Suppose  $\vec{v}_4 = \begin{bmatrix} 2 \\ 2 \\ 3 \\ 3 \end{bmatrix}$ . If you were to apply the Gram-Schmidt process to the vectors  $\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4$ , what would the vector  $\vec{q}_4$  be? Why?

- c)** (4 pts) Consider  $\vec{y} = \begin{bmatrix} 3 \\ -1 \\ -2 \\ -2 \end{bmatrix}$ .  $\vec{y}$  is in  $\text{span}(\{\vec{v}_1, \vec{v}_2, \vec{v}_3\}) = \text{span}(\{\vec{q}_1, \vec{q}_2, \vec{q}_3\})$ .

Find scalars  $a, b$ , and  $c$  such that  $a\vec{q}_1 + b\vec{q}_2 + c\vec{q}_3 = \vec{y}$ , **without** solving a system of 3 equations and 3 unknowns. Instead, use the fact that  $\vec{q}_1, \vec{q}_2, \vec{q}_3$  are orthonormal.

*Hint: There's a relevant problem from [Lab 4](#); review that, along with [Chapter 2.3](#).*

- d)** (4 pts) Consider  $\vec{y} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$ . Unlike in **c)**,  $\vec{y}$  is **not** in  $\text{span}(\{\vec{v}_1, \vec{v}_2, \vec{v}_3\})$ .

Find the vector in  $\text{span}(\{\vec{v}_1, \vec{v}_2, \vec{v}_3\})$  that is closest to  $\vec{y}$ . **Do not** stack the  $\vec{v}_i$ 's into a matrix  $X$  and then use  $X(X^T X)^{-1} X^T \vec{y}$ . Instead, use the fact that  $\vec{q}_1, \vec{q}_2, \vec{q}_3$  are orthonormal and have the same span as  $\vec{v}_1, \vec{v}_2, \vec{v}_3$ . How does this simplify the problem?

*Hint: What did you discover in Problem 1i)?*

- e)** (5 pts) Open the **the supplemental Jupyter Notebook** we've created for Homework 7, which can either be found [here](#) on DataHub, or [here](#) in the course GitHub repository.

There, you're asked to implement the function `orthogonalize`, which takes in an  $n \times d$  matrix  $V$  whose columns are linearly independent, and returns a matrix  $Q$  whose columns are orthonormal and have the same span as  $V$ . This problem is **not autograded**. Rather, in your submission to this part, include a screenshot of your implementation and sample output in your PDF for Homework 7.



f) (5 pts) A QR decomposition of a matrix  $A$  is a factorization of the form

$$A = QR$$

where  $Q$  is an  $n \times d$  matrix with orthonormal columns and  $R$  is an  $d \times d$  **upper triangular** matrix (a matrix that has 0s below the diagonal).

For example, if  $A = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 1 & 2 \end{bmatrix}$ , a QR decomposition of  $A$  is

$$A = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 1 & 2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1/\sqrt{3} & 1/\sqrt{6} & -1/\sqrt{2} \\ -1/\sqrt{3} & 2/\sqrt{6} & 0 \\ 1/\sqrt{3} & 1/\sqrt{6} & 1/\sqrt{2} \end{bmatrix}}_Q \underbrace{\begin{bmatrix} \sqrt{3} & 2\sqrt{3}/3 & 2\sqrt{3}/3 \\ 0 & \sqrt{6}/3 & 5\sqrt{6}/6 \\ 0 & 0 & 1/\sqrt{2} \end{bmatrix}}_R$$

Finding the  $Q$  in a QR decomposition is straightforward: apply Gram-Schmidt to the columns of  $A$ , assuming  $A$ 's columns are linearly independent. The question is how to find  $R$ .

- (i) In the supplemental Jupyter Notebook, we've defined an arbitrary matrix  $A$  and call your `orthogonalize` function on it, and give you hints as to how to find  $R$ . Using the experimentation there, and what you know about  $Q$ , **explain how to find  $R$** .

- (ii) Find a QR decomposition of  $A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$ , made up of the same three vectors you worked with in parts **a)** through **d)** of this problem.

- (iii) Given a QR decomposition of  $A$ , explain how to find **another** QR decomposition of  $A$  with a (slightly) different  $Q$  and/or  $R$ .

The QR decomposition will be important when we study eigenvalues and eigenvectors in Chapter 5. These are the backbone of Google's (original) search algorithm, PageRank.