QTM2600 Course Notes

Nathan Karst
Division of Mathematics and Science
Babson College

October 8, 2013

Contents

Chapter 1

Linear systems

1.1 Linear systems and their solutions

Bad news: the amusement park you manage has lost the breakdown of adult, child, senior, and student tickets sales for the day. The good news is that you do have some information:

- By counting the total number of tickets remaining in the office, you know that the park sold 960 total tickets.
- From a partial count of sales, you know that you sold 768 adult and child tickets.
- Since discount tickets (child, student, and senior) are processed differently than full-priced tickets, you have a record of how many of these tickets you sold: 704.
- By counting the money in the register, you know that the total revenue for the day was $10,240. The price of an adult ticket is $20, the price of a child's ticket is $5, the price of a student ticket is $10, and the price of a senior ticket is $15.

How could we use these data to piece together how many of each type of ticket were sold? We should probably start by defining what we don't know. Let a, c, s, and e be the number of adult, child, student, and senior tickets, respectively. If we phrase the first idea above in terms of these variables we have

$$a + c + s + e = 960.$$

We can apply the same methodology to each of the pieces of information we know to construct the following system of equations:

$$\begin{array}{rcrcrcrcrl}
a & + & c & + & s & + & e & = & 960 & (\textit{total tickets}) \\
a & + & c & & & & & = & 768 & (\textit{adult and child tickets}) \\
 & & c & + & s & + & e & = & 704 & (\textit{discount tickets}) \\
20a & + & 5c & + & 10s & + & 15e & = & 10240 & (\textit{total revenue}).
\end{array}$$

You might remember solving systems of liner equations like this one by sequentially eliminating variables through (1) multiplying an equation by a constant; (2) adding and subtracting rows. In both of these operations, the coefficients of the variables, not the variables themselves, are the important feature of the equations. For instance, when you add two equations together, you're actually adding coefficients corresponding to the same variable, and when you multiply an equation by a constant, you actually multiply all the coefficients by the chosen constant. So solving systems of linear equations using elimination is all about manipulating the coefficients of the system of equations.

1.1.1 Coefficient matrices and augmented matrices

One important feature of the linear system is the coefficients of the variables on the left side of each equation. We can capture these coefficients in a compact form through a *coefficient matrix*. For instance, the coefficient matrix of our example is

$$C = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 20 & 5 & 10 & 15 \end{bmatrix}.$$

Reading Check 1: Pause for a second and try to articulate how you think we created this matrix. Then compare your description to the one below.

To form the coefficient matrix of a linear system, we create one column for each variable and one row for each equation, and place at the intersection of a row i and a column j the coefficient of the j^{th} variable in the i^{th} equation.

But in addition to the coefficients of the variables of the left side of the equation, the linear system has another important component: the values on the right side of the equations. We can capture these too using by constructing the *augmented matrix* of the linear system. Here the augmented matrix is given by

$$A = \left[\begin{array}{cccc|c} 1 & 1 & 1 & 1 & 960 \\ 1 & 1 & 0 & 0 & 768 \\ 0 & 1 & 1 & 1 & 704 \\ 20 & 5 & 10 & 15 & 10240 \end{array}\right].$$

Reading Check 2: Take another second and try to articulate how you think we created this matrix. Then compare your description to the one below.

Notice that the first four columns of the augmented matrix are identical to the coefficient matrix and that the entry in the i^{th} row of the last column of the augmented matrix is just the right hand side of the i^{th} equation.

Example 1: Find the augmented matrix of the following linear system:

$$\begin{array}{rcrcrcl} 13a & - & 17b & + & 19c & = & 3 \\ 2a & + & 8b & - & 10c & = & 5 \\ a & + & 2b & + & 3c & = & 9. \end{array}$$

Example 2: Find a linear system corresponding to the following augmented *matrix:*

$$\left[\begin{array}{cc|c} 1 & 2 & 6 \\ 2 & 4 & 12 \\ 3 & 5 & 11 \end{array}\right].$$

Example 3: Find the augmented matrix of the following linear system:

$$\begin{array}{ccccccc} 6x_1 & + & 3x_2 & + & 2x_3 & = & 12 \\ 5x_1 & + & 4x_2 & + & 3x_3 & = & 11 \end{array}$$

Example 4: Find a linear system corresponding to the following augmented matrix:

$$\left[\begin{array}{cc|c} 1 & 2 & 6 \\ 3 & 4 & 12 \\ 3 & 5 & 11 \end{array}\right].$$

Let's turn our attention back to our ticket example. Now that we've had some practice converting from systems of linear equations into augmented matrices and vice versa, let's think about the form of the augmented matrix of our solution $a = 256$, $c = 512$, $s = 64$ and $e = 128$.

Reading Check 3: Take a second and come up with a hypothesis about how the augmented matrix of the solution will look. Then compare your guess with the solution below.

In the first equation of the system, $a = 256$, the coefficient of a is 1, and the coefficients of the other three variables are 0. Similarly, for the second equation in the system, $c = 512$, the coefficient of c is 1 and the coefficients of the other three variables are 0. We can repeat this same type of logic for all the equations in the solution. Then applying the same reasoning that we used the the four examples we just completed, we can write

$$A_{sol} = \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 256 \\ 0 & 1 & 0 & 0 & 512 \\ 0 & 0 & 1 & 0 & 64 \\ 0 & 0 & 0 & 1 & 128 \end{array}\right].$$

I think you might agree right off the bat that this matrix has some interesting structure. In particular, there seems to be a pattern containing 1 and 0. Before we start talking about this pattern formally, let's define a few notions that we'll see are critical to understanding the number of solutions of a linear system. A leading (left-most) 1 is called a *pivot.* A column containing a pivot is called a *pivot column.*

1.1.2 Row reduced echelon form

We say a matrix is in *row reduced echelon form* (RREF) if

- the left-most nonzero in any row is a 1
- each pivot is the only nonzero in its column
- each pivot occurs to the right of any pivot above it
- any row containing only zeros is below every row containing a nonzero

Reading Check 4: Confirm that A_{sol} is in RREF.

If you've verified this previous statement, then you'll agree that in at least the tickets case, we formed the RREF of an augmented matrix when we were looking for the solutions of the associated system of linear equations. This is a key idea! In fact, it's a general idea, too: **to find the solutions to any system of linear equations, we find the RREF of its augmented matrix.**

The problem here, of course, is that actually computing the RREF by hand is a total pain. And we've only done it for a simple system of 4 equations in 4 unknowns. In applications, we can have much, much larger systems whose RREFs are practically incomputable by hand. We need a computational tool to help us out. Matlab is the industry standard for things like this. If you don't have access to Matlab, there are several free alternatives that'll get the job done, *e.g.*, Octave.

Let's start by learning how to enter a matrix into Matlab. Before we talk about specifics, let's see an example:

$$\left[\begin{array}{cc|c} 1 & 2 & 6 \\ 2 & 4 & 12 \\ 3 & 5 & 11 \end{array}\right] \leftrightarrow \texttt{[1, 2, 6; 2, 4, 12; 3, 5, 11]}.$$

Reading Check 5: Take a second and try to figure out the rules for entering matrices in Matlab. Then test your intuition against the explanation below.

We enclose matrices in square brackets, enter one row at a time, separate columns with commas, and separate rows with semicolons. In fact, if we enter the command on the right, Matlab will give us a rough picture of what it thinks we meant:

```
>> [1, 2, 6; 2, 4, 12; 3, 5, 11]

ans =

     1     2     6
     2     4    12
     3     5    11
```

(Remember, the vertical line is just to help us remember that the matrix we're looking at is an augmented matrix rather than a coefficient matrix.)

Reading Check 6: Try entering the matrices associated with the examples from earlier into Matlab.

It would be nice to have access to the matrices we've entered, even if we've run some commands since we entered the matrix. To do this, we're going to assign the matrix to a variable name. For instance,

```
>> A = [1, 2, 6; 2, 4, 12; 3, 5, 11]

A =

     1     2     6
     2     4    12
     3     5    11
```

Assignments in Matlab always assign the value to the right of the equals sign to the variable to the left of the equals sign. So here, we're taking the value represented by the matrix we've entered and assigning it to the variable `A`. If we want to get access to `A` later, we can always just type `A` and verify that the value of this variable is still what we think it is:

```
>> A

A =

     1     2     6
     2     4    12
     3     5    11
```

One way in which this is particularly useful is to find the RREF of a matrix. Whenever you're in doubt of how to use a Matlab command, try typing `help` followed by the command name. This will bring up some text describing how the command works, including what it expects you to provide and what you can count on it providing you. For instance,

```
RREF   Reduced row echelon form.
    R = RREF(A) produces the reduced row echelon form of A.
 ...
```

(The text goes on here, as these type of help entries often do, but usually the most important stuff is at the top.) So, we know that if we type `rref(A)`, then Matlab will return a matrix `R` that represents the RREF of `A`. Let's give it a shot.

```
>> rref(A)
```

```
ans =

     1     0    -8
     0     1     7
     0     0     0
```

Reading Check 7: Using the properties of RREF that we set out earlier, verify that this matrix is actually in RREF.

Reading Check 8: Use Matlab to compute the RREF of each of the example augmented matrices we saw earlier. Then transform each of the RREF matrices into a system of linear equations. Make a hypothesis about how many solutions each linear system has.

Let's start with Example 1.1.1. In Matlab, we enter

```
>> W = [13 17 19 3; 2 8 10 5; 1 2 3 9];
>> rref(W)

ans =

    1.0000         0         0    2.4773
         0    1.0000         0  -16.2727
         0         0    1.0000   13.0227
```

Before we start analyzing the output of `rref`, let's notice a few things about how we input the matrix W. First, there aren't any commas separating the entries of the rows. Commas are just for our convenience; sometimes it's really help to have them there to keep everything straight, but we don't have to have them every time. The semicolons that mark the end of the row are not optional, however. Second, notice that we've added a semicolon at the very end of the line. This suppresses the output command. So Matlab still assigns the matrix to the variable `W`, but it doesn't show you what it's done. It's a good way to save space and avoid clutter when you're pretty sure you know what Matlab's done. If you want to verify that `W` actually contains what you think it does, just type `W` in the command line, and Matlab will show you what it has stored in that variable.

```
>> W

W =

    13    17    19     3
     2     8    10     5
     1     2     3     9
```

If x_1, x_2, and x_3 are the variables corresponding to the first three columns of the augmented matrix in order from left to right, then our RREF matrix represents the linear system $x_1 = 2.4773$, $x_2 = -16.2727$, $x_3 = 13.0227$. So the linear system has a unique solution. Notice that just as with our tickets example, each variable is a pivot column, and the right-most column is not a pivot column. We'll gather up some facts about pivot columns and their relationship to the number solutions to a linear system and then gather up these results into some bigger ideas at the end of this section.

1.1.3 Redundancies and inconsistencies

In Example 1.1.1, we have

```
>> X = [1 2 6; 2 4 12; 3 5 11;];
>> rref(X)

ans =

     1     0    -8
     0     1     7
     0     0     0
```

If we assume our variables are x_1 and x_2 in the same spirit as the last example, the RREF matrix corresponds to the system $x_1 = -8$, $x_2 = 7$ and $0 = 0$. This last equation is *redundant*, meaning that it's not offering any new information. After all, $0 = 0$ is always true. We can see that two of the original equations were redundant; the first equation $x_1 + 2x_2 = 6$ and the second equation $2x_1 + 4x_2 = 12$ are really the scaled versions of one another. Redundancy can get more complicated as we'll see in future sections, but the idea will remain the same: we don't have as much information as we might think we have. Before we move on, notice that each variable column is a pivot column and that the right-most column is not a pivot column.

1.1.4 Free variables

In Example 1.1.1, we have

```
>> Y = [6 3 2 12; 5 4 3 11];
>> rref(Y)

ans =

    1.0000         0   -0.1111    1.6667
         0    1.0000    0.8889    0.6667
```

Assuming that our variables are x_1, x_2, x_3 as we've been doing, our linear system reads $x_1 - 0.\overline{11}x_3 = 1.\overline{66}$ and $x_2 + 0.\overline{88}x_3 = 0.\overline{66}$. The reason this example is different than anything we've seen before is that we have no way of knowing what x_3 should be. But we can confirm that as soon as we choose x_3, we can uniquely determine the values of x_1 and x_2.

Reading Check 9: Choose a few values of x_3 and compute the resulting values of x_1 and x_2.

We call x_3 a *free variable* exactly because we have the freedom to choose it. And because there are infinitely many choices for x_3, there are infinitely many solutions to this system of linear equations! Note that this is the first example in which one of the variable columns is not a pivot column.

In example 1.1.1, we have

```
>> Z = [1 2 6; 3 4 12; 3 5 11];
>> rref(Z)

ans =

     1     0     0
     0     1     0
     0     0     1
```

Assuming that our variables are x_1 and x_2, this RREF matrix corresponds to $x_1 = 0$, $x_2 = 0$ and $0 = 1$. In other words, for this system of three equations to be true, it must be the case that $x_1 = 0$, $x_2 = 0$ and $0 = 1$. Since we don't live in a numerical universe where $0 = 1$, there is no solution to this linear system! Notice that this is the first case in which the right-most column is a pivot column.

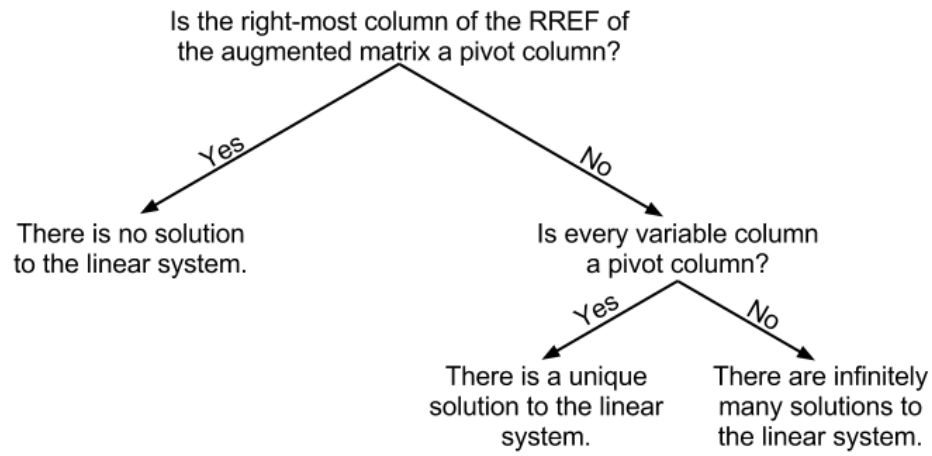
Reading Check 10: Convince yourself using the definition of RREF that if the right-most column is a pivot column then the system of linear equations has no solution.

1.1.5 Summary flow chart

Let's pull our observations about the relationship between the positions of pivot columns and the number of solutions to a linear system into a decision tree.

1.2 Vector and Matrix Equations

So far we've been using the term "matrix" loosely, but before we begin talking about how vector equations and matrix equations relate to linear systems, let's take some time to formalize some terms and introduce some others. A $r \times c$ *matrix* is a rectangular array of numbers with r rows and c columns. We call $r \times c$ the dimensions of the variables. A *column vector* is a matrix with just one column, and a *row vector* is a matrix with just one row. The notation for vectors varies, but both bold font like $\mathbf{x}$ and a little arrow on top like $\vec{x}$ both indicate that x is a vector. (Typically, we only use the arrow when we can't clearly indicate that a variable is in bold case, like when we're writing on a blackboard.) We call the entries of a vector *components*, and we typically think



of ordering the components from top to bottom in the case of a column vector or from left to right in the case of a row vector, so that saying something like "the first component of $\mathbf{x}$" makes sense. As we'll see, the components of vectors will often have specific meanings in the models we'll construct, and so it's helpful to be able to refer to them each specifically without any confusion.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \begin{array}{l} \leftarrow 1^{st} \text{ component} \\ \leftarrow 2^{nd} \text{ component} \\ \qquad \vdots \\ \leftarrow n^{th} \text{ component} \end{array}$$

If $\mathbf{x}$ has n components we say $\mathbf{x}$ is in $\mathbb{R}^n$, or $\mathbf{x} \in \mathbb{R}^n$ in mathematical shorthand. If you haven't seen it before, the symbol $\mathbb{R}$ stands for the real numbers, that is, all the normal numbers with (possibly) infinitely many digits to the left and right of the decimal point that use all the time. The space $\mathbb{R}^n$ is just the collection of all vectors with n components which are real numbers.

We can add vectors. For instance

$$\begin{bmatrix} a \\ b \end{bmatrix} + \begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} a+c \\ b+d \end{bmatrix}.$$

We say that vector addition is done "component-wise", meaning we add components in the same position. Notice that this means that adding vectors of different sizes doesn't make sense.

We can also scale vectors by multiplying them by a constant. For instance, if k is some real number, then

$$k \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} ka \\ kb \end{bmatrix}.$$

Again, the multiplication affects each component of the vector individually.

But we still haven't figured out why we should care about vectors. To get a handle on this, let's go back to our original example about amusement park tickets.:

$$\begin{array}{rcrcrcrcl} a & + & c & + & s & + & e & = & 960 \\ a & + & c & & & & & = & 768 \\ & & c & + & s & + & e & = & 704 \\ 20a & + & 5c & + & 10s & + & 15e & = & 10240 \end{array}$$

Up till now, we've been thinking about grouping these entries in rows. But imagine we group them in columns instead. Then we can see

$$\begin{array}{rcrcrcrcl} a & + & c & + & s & + & e & = & 960 \\ a & + & c & & & & & = & 768 \\ & & c & + & s & + & e & = & 704 \\ 20a & + & 5c & + & 10s & + & 15e & = & 10240 \end{array} \leftrightarrow a\begin{bmatrix}1\\1\\0\\20\end{bmatrix} + c\begin{bmatrix}1\\1\\1\\5\end{bmatrix} + s\begin{bmatrix}1\\0\\1\\10\end{bmatrix} + e\begin{bmatrix}1\\0\\1\\15\end{bmatrix} = \begin{bmatrix}960\\768\\704\\10240\end{bmatrix}$$

$$\leftrightarrow a\mathbf{v}_1 + c\mathbf{v}_2 + s\mathbf{v}_3 + e\mathbf{v}_4 = \mathbf{b}$$

So we've turned our system of linear equations into a single *vector equation.* Notice that each of the vectors on the left side of this equation is a column of the coefficient matrix. Notice that $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4$ are the first 4 columns of the coefficient matrix in order and $\mathbf{b}$ is the right side of the linear system. We say $\mathbf{b}$ is a *linear combination* of the vectors $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4$ with weights a, c, s, e, respectively.

1.2.1 Linear Combinations and Spans

Linear combinations are one of the most important ideas of the early part of this course, so it's worth it to take a little time and make sure we have a solid intuitive understanding of what linear combinations mean. But let's start an example that's a little more basic than the one we've been considering. Imagine we have a single vector

$$\mathbf{v}_1 = \begin{bmatrix}1\\-1\end{bmatrix}.$$

While we won't drive ourselves crazy with the geometry of linear algebra, it really is convenient to sometimes think about matrices and vector geometric. For instance, if we think of the first coordinate of $\mathbf{v}_1$ as the x coordinate, and the second component of $\mathbf{v}_1$ as the y coordinate, we can plot the vector in the $x - y$ plane as seen in figure 1.1.

A linear combination in general is a weighted sum vectors. But here, since we have only one vector, any linear combination looks like $x_1\mathbf{v}_1$. Imagine for

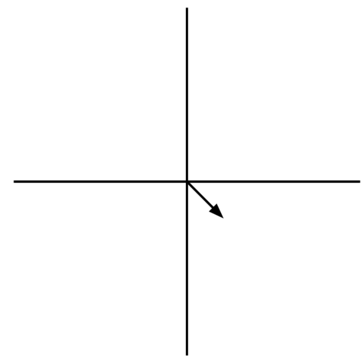


Figure 1.1: Graphical representation of $\mathbf{v}_1$

instance that our weight x_1 is -3. Then

$$-3\mathbf{v_1} = \begin{bmatrix} -3 \\ 3 \end{bmatrix}.$$

We can plot this vector, too. We can see the results in figure 1.2.

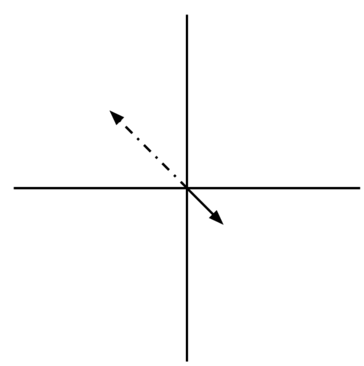


Figure 1.2: Graphic representation of $\mathbf{v}_1$ (solid) and $-3\mathbf{v}_1$ (dot-dashed).

Reading Check 11: Try draw $x_1\mathbf{v}_1$ for several more values of x_1. Try positive, negative, big, and small values. What pattern do you see emerge?

Reading Check 12: Can every vector in $\mathbb{R}^2$ be represented by a linear combination of $\mathbf{v}_1$?

If we allow x_1 to vary across all real numbers, we start getting a clear picture of a line representing all the possible $x_1\mathbf{v}_1$ for any choice of x_1. Mathematically, we call this the *span* of $\mathbf{v}_1$ and often denote it as $span(\mathbf{v}_1)$. We can see a graphical representation of $span(\mathbf{v}_1)$ in figure 1.3.

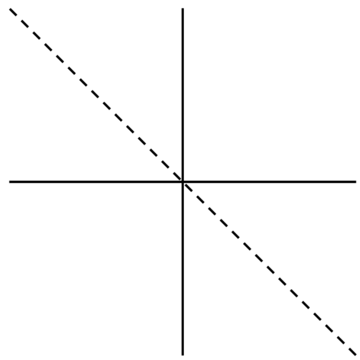


Figure 1.3: Graphic representation $span(\mathbf{v}_1)$ (dashed).

But what if we have two vectors, say

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \quad \mathbf{v}_2 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}?$$

If we consider about the first, second, and third components as x, y, and z, then we can think of these vectors appearing in three-dimensional space as we see in figure We can imagine taking a linear combination of these vectors, $x_1\mathbf{v}_1+x_2\mathbf{v}_2$. For instance, if we set $x_2 = 0$, and scale x_1 along the real numbers, we get the line $span(\mathbf{v}_1$, and if we set $x_1 = 0$ and scale x_2, we get another line $span(\mathbf{v}_2)$. We can see these in figure 1.5. But the whole span of $\mathbf{v}_1$ and $\mathbf{v}_2$ together is more than the sum of the parts. We could take a little of $\mathbf{v}_1$ and a little of $\mathbf{v}_2$ to make a new vector, *e.g.*, $\mathbf{v}_3 = 0.5\mathbf{v}_1 + 0.5\mathbf{v}_2$, which lies in the span of the two vectors together but in neither of the individual spans. If we allow x_1 and x_2 to vary across the real numbers independently, we get a much bigger set of possible linear combinations. This collection $span(\mathbf{v}_1, \mathbf{v}_2)$ can be seen in figure 1.3. Generally speaking, the *span* of a set of vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ is the set of all linear combinations $x_1\mathbf{v}_1 + x_2\mathbf{v}_2 + \ldots + x_n\mathbf{v}_n$.

Understanding span is key to understanding whether a linear system has a solution. We've seen that every linear system is equivalent to a vector equation.

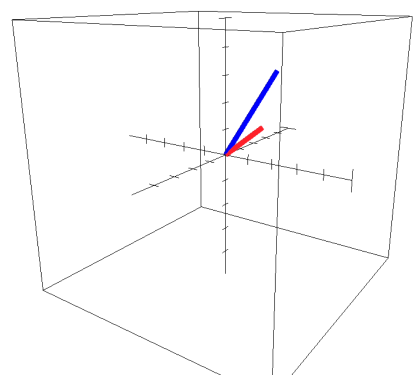


Figure 1.4: Graphic representation $\mathbf{v}_1$ (red) and $\mathbf{v}_2$ (blue).

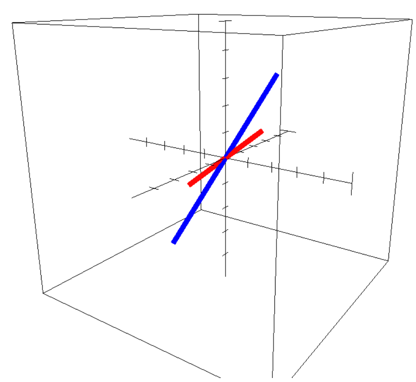


Figure 1.5: Graphic representation $span(\mathbf{v}_1)$ (red) and $span(\mathbf{v}_2)$ (blue)

Determining whether the vector equation $x_1\mathbf{v}_1 + x_2\mathbf{v}_2 + \ldots + x_n\mathbf{v}_n = \mathbf{b}$ has a solution is the same thing as determining whether $\mathbf{b}$ is in $span(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$.

Reading Check 13: Use the definitions in this section to convince yourself that this last statement is true.

1.2.2 Matrix multiplication

Linear combinations like those we've been dealing with come up so often that mathematicians have invented a very compact notation to describe them; it

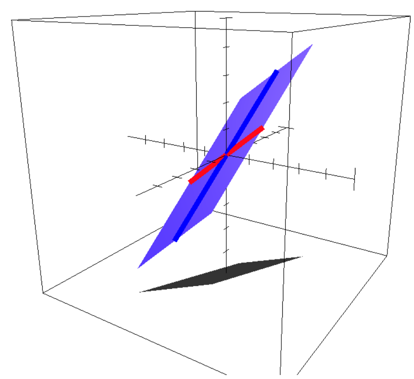


Figure 1.6: Graphic representation $span(\mathbf{v}_1)$ (red), $span(\mathbf{v}_2)$ (blue), and $span(\mathbf{v}_1, \mathbf{v}_2)$ (purple).

really saves on the writing.

$$\begin{array}{ccccccccc} a & + & c & + & s & + & e & = & 960 \\ a & + & c & & & & & = & 768 \\ & & c & + & s & + & e & = & 704 \\ 20a & + & 5c & + & 10s & + & 15e & = & 10240 \end{array}$$

Up till now, we've been thinking about grouping these entries in rows. But imagine we group them in columns instead. Then we can see

$$\begin{array}{ccccccccc} a & + & c & + & s & + & e & = & 960 \\ a & + & c & & & & & = & 768 \\ & & c & + & s & + & e & = & 704 \\ 20a & + & 5c & + & 10s & + & 15e & = & 10240 \end{array} \leftrightarrow a\begin{bmatrix} 1 \\ 1 \\ 0 \\ 20 \end{bmatrix} + c\begin{bmatrix} 1 \\ 1 \\ 1 \\ 5 \end{bmatrix} + s\begin{bmatrix} 1 \\ 0 \\ 1 \\ 10 \end{bmatrix} + e\begin{bmatrix} 1 \\ 0 \\ 1 \\ 15 \end{bmatrix} = \begin{bmatrix} 960 \\ 768 \\ 704 \\ 10240 \end{bmatrix}$$

$$\leftrightarrow \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 20 & 5 & 10 & 15 \end{bmatrix} \begin{bmatrix} a \\ c \\ s \\ e \end{bmatrix} = \begin{bmatrix} 960 \\ 768 \\ 704 \\ 10240 \end{bmatrix}$$

We call this last line a *matrix equation*. Notice that the matrix in the equation is just the coefficient matrix of the linear system. The vector on the left side of the equation just contains our variables, and the vector on the right side contains the right side of the linear system.

It's also worth pointing out that we've done something kind of interesting and important here: we've shown that there are three equivalent ways to represent a system of linear equations: as the linear system itself, as a vector equation, and as a matrix equation. Each of these is useful in different ways, and we'll investigate the properties of all of them over the coming sections.

Without too much introduction, imagine we have the following

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

Imagine that the matrix on the left is a coefficient matrix of some linear system, and that the vector on the right represents some particular chosen values of the variables in that linear system. The left side of the first equation in the linear system would read

$$1x_1 + 2x_2 + 3x_3$$

and with our chosen values of x_1, x_2, x_3, the left side of the first equation would be equal to

$$1(7) + 2(7) + 3(7) = 42.$$

Notice that this is the same as taking the first row, flipping it so the first component is at the top, multiplying component wise with the vector, and adding the results.

Reading Check 14: Try multiplying the second row with the vector. You should get 105 as your answer.

Reading Check 15: Try the following matrix-vector multiplication:

$$\begin{bmatrix} 3 & 5 \\ 7 & 9 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Matlab can really help us out with this type of thing. We already know how to input matrices and vectors, and multiplying them is as easy as tossing an asterisk between the two. For instance, the answer to the previous question should be

```
>> [3 5; 7 9] * [1;2]

ans =

    13
    25
```

For the following examples, feel free to do some by hand, but make sure you check with Matlab.

Reading Check 16: Try the following matrix-vector multiplication:

$$\begin{bmatrix} 1 & 2 \\ 3 & 5 \\ 7 & 11 \end{bmatrix} \begin{bmatrix} 4 \\ 6 \end{bmatrix}.$$

Reading Check 17: Try the following matrix-vector multiplication:

$$\begin{bmatrix} 11 & 9 & 7 \\ 5 & 3 & 1 \end{bmatrix} \begin{bmatrix} 8 \\ 6 \\ 4 \end{bmatrix}.$$

Reading Check 18: Try the following matrix-vector multiplication:

$$\begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}.$$

1.3 Homogeneous and inhomogeneous Equations

The *homogeneous* equation $A\mathbf{x} = \mathbf{0}$ can tell us a lot about the *nonhomogenous* equation $A\mathbf{x} = \mathbf{b} \neq \mathbf{0}$. To get started, let's note that $\mathbf{x} = 0$ is always a solution to $A\mathbf{x} = \mathbf{0}$.

Reading Check 19: Using the definition of matrix-vector multiplication from the last section, convince yourself that this is true.

We call $\mathbf{x} = \mathbf{0}$ the *trivial solution* to the equation, exactly because it's such an easy solution to find. A natural next question is whether there exists a *nontrivial solution* $\mathbf{x} \neq \mathbf{0}$. Let's take a particular example.

$$\begin{aligned} 2.0x_1 + 3.0x_2 + 6.0x_3 &= 0 \\ 9.0x_1 + 2.0x_2 + 6.0x_3 &= 0 \\ 1.0x_1 + 6.0x_2 + 8.0x_3 &= 0 \end{aligned}$$

Reading Check 20: Determine whether this particular system has a nontrivial solution.

Reading Check 21: Make a hypothesis as to when homogeneous systems may or may not have nontrivial solutions.

Any homogeneous system is consistent, because $\mathbf{x} = \mathbf{0}$ is always a solution. Then following the flow chart we developed in the first section, we can claim that $A\mathbf{x} = \mathbf{0}$ has a unique solution if every column of A has a pivot and has infinitely many solutions if some column of A does not have a pivot. If the equation has a unique solution, it is the trivial solution, and if the equation has infinitely many solutions, then the system has nontrivial solutions. So $A\mathbf{x} = \mathbf{0}$ has a nontrivial solution if and only if A has at least 1 free variable.

But what about inhomogeneous equations? To get some intuition, let's draw a few pictures. Check out $x_1 + x_2 = 0$ and $x_1 + x_2 = 2$ in figure 1.7.

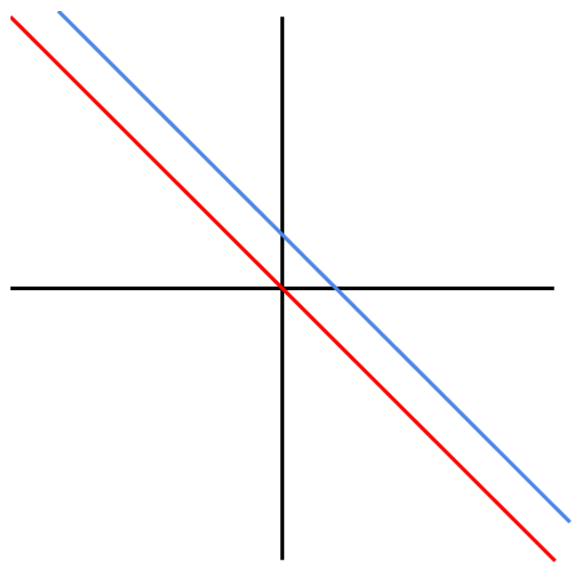


Figure 1.7: Graphical representation of the linear equations $x_1 + x_2 = 0$ (red) and $x_1 + x_2 = 2$ (blue).

The solution set of the homogeneous equation passes through the origin because the trivial solution $x_1 = x_2 = 0$ satisfies the equality. Notice that the solution set to the inhomogeneous equation is parallel to the solution set for the homogeneous equation. In this 2-dimensional case, we can just think of this as just "changing the x_2-intercept", but the notion of parallel solution sets is a general on. For another example, consider the homogeneous equation

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

We saw in the last section that the span of the columns of this matrix forms a plane in 3-dimensional space. We could also consider the inhomogeneous equation

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \mathbf{x} = - \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}.$$

Your intuition might be telling you that the solution set to this equation will also form a plane in 3-dimensional space, and you'd be right. We can see both solution sets in figure 1.8. Again we see that the solution sets are parallel.

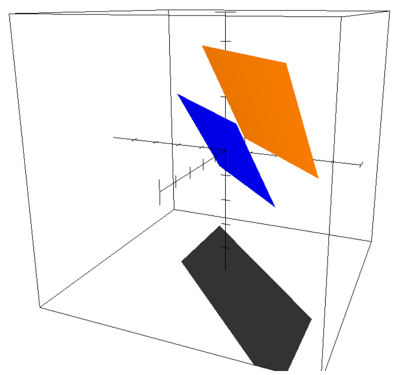


Figure 1.8: Graphical representation of the homogeneous solution set (blue) and inhomogeneous solution set (orange).

But why are these solutions sets parallel? Well, if they were not, then they would intersect, and so both solution sets would contain some vector $\mathbf{y}$. In symbols, $A\mathbf{y} = \mathbf{0}$ and $A\mathbf{y} = \mathbf{b} \neq \mathbf{0}$. But then $\mathbf{0} = A\mathbf{y} = \mathbf{b}$, but we already said that $\mathbf{b} \neq 0$. So it can't be the case that the two solutions sets share any vectors.

Let's see this idea one more way. Imagine $\mathbf{k}$ is any solution to $A\mathbf{x} = \mathbf{0}$ and $\mathbf{p}$ is a solution to $A\mathbf{x} = \mathbf{b} \neq \mathbf{0}$. Consider $A(\mathbf{k}+\mathbf{p})$. We've shown that matrix-vector multiplication distributes, and so we can claim with confidence that $A(\mathbf{k}+\mathbf{p}) = A\mathbf{k} + A\mathbf{p} = \mathbf{0} + \mathbf{b}$. So $\mathbf{k} + \mathbf{p}$ is a solution to the inhomogeneous equation for any choice of $\mathbf{k}$ satisfying the homogeneous equation. In other words, there are at least as many solutions to the inhomogeneous equation as there are to the homogeneous equation, and the solution set for the inhomogeneous equation is just the solution set of the homogeneous equation shifted (the mathematical word is *translated*) by $\mathbf{p}$.

But let's tie up the big idea: we learn a lot about the solution set of the equation $A\mathbf{x} = \mathbf{b}$ simply by looking at the solution set of $A\mathbf{x} = \mathbf{0}$, and studying the inhomogeneous equation is typically a lot easier.

	Coal	Electricity	Steel
Coal	0	0.56	0.43
Electricity	0.72	0.11	0.40
Steel	0.28	0.33	0.17

Table 1.1: Consumptions (in percent) in our manufacturing sector economic model. The intersection of row i and column j is the fraction of industry i's output that was sold to industry j. Notice that this implies that the column should each sum to 1.

1.3.1 Application of homogeneous equations: high dimensional break-even

Suppose that a model of an the manufacturing sector of an economy consists of just three industries: coal, electricity and steel production. Define p_c, p_e and p_s to be the total annual output of the coal, electricity and steel industries, respectively. In this model, we'll assume that all output is consumed. A natural question that should come up any time you're dealing with a real world application is "what are the units?" And it's a great question here, too. For our purposes, we'll think of our annual production in terms of their currency value; let's use dollars in this example. But as you might suspect, these industries don't operate in isolation. The coal industry buys steel to build new mines, the steel industry buys electricity to power its plants, and so on. We can capture these interrelated rates of consumption in Table 1.3.1. Here the intersection of row i and column j is the fraction of industry i's output that was sold to industry j. So, for instance, the electricity buys 72% of the coal industry's annual output (as measured in dollars).

One question we might ask about a system like this is how much each industry should produce so that every industry exactly breaks even. Remember that the break-even point is where revenue equals cost. Let's start with cost. How much does the coal industry spend every every? Well, it buys 0% of its own output, 56% of the electricity industries output, and 43% of the steel industries output. Using symbols instead of words, this sentence reads

$$0p_c + 0.56p_e + 0.43p_s.$$

We could perform the same conversion for the other two industries. Let's think about forming a *cost vector* that we'll call $\mathbf{c}$. Here, the components of our cost vector will be, in order, the cost incurred by the coal, electricity and steel industries, respectively.

$$
\begin{aligned}
\mathbf{c} = \begin{bmatrix} \text{cost of coal industry} \\ \text{cost of electricity industry} \\ \text{cost of steel industry} \end{bmatrix} &= \begin{bmatrix} 0.00p_c + 0.56p_e + 0.43p_s \\ 0.72p_c + 0.11p_e + 0.40p_s \\ 0.28p_c + 0.33p_e + 0.17p_s \end{bmatrix} \\
&= \begin{bmatrix} 0.00 & 0.56 & 0.43 \\ 0.72 & 0.11 & 0.40 \\ 0.28 & 0.33 & 0.17 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} \\
&= C\mathbf{p},
\end{aligned}
$$

where C is the matrix and $\mathbf{p}$ is the vector of annual outputs. To see why this matrix-vector product expression is powerful, we need to discuss the revenue vector. By the definition of the problem statement, the total annual revenue of the coal industry is just p_c, and similarly for the remaining two industries. Let's think about defining a revenue vector $\mathbf{r}$ in a similar way to the cost vector, so that the first component of $\mathbf{r}$ is the revenue of the coal industry, the second of the electricity industry, and the third of the steel industry. Replacing the words with symbols,

$$
\begin{aligned}
\mathbf{r} = \begin{bmatrix} \text{revenue of coal industry} \\ \text{revenue of electricity industry} \\ \text{revenue of steel industry} \end{bmatrix} &= \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} \\
&= R\mathbf{p}.
\end{aligned}
$$

When all industries break-even simultaneously, we have

$$
\mathbf{c} = \begin{bmatrix} \text{cost of coal industry} \\ \text{cost of electricity industry} \\ \text{cost of steel industry} \end{bmatrix} = \begin{bmatrix} \text{revenue of coal industry} \\ \text{revenue of electricity industry} \\ \text{revenue of steel industry} \end{bmatrix} = \mathbf{r}.
$$

But how do we actually *solve* for this case? How can we determine what value(s) or p_c, p_e and p_s lead to all industries breaking even? Well, we have matrix-vector product expressions for both $\mathbf{r}$ and $\mathbf{c}$, both of which contain the variables p_c, p_e and p_s that we're after. Maybe substituting these matrix-vector products is a good place to start.

$$
\begin{gathered}
\begin{bmatrix} 0.00 & 0.56 & 0.43 \\ 0.72 & 0.11 & 0.40 \\ 0.28 & 0.33 & 0.17 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} \\
C\mathbf{p} = R\mathbf{p}.
\end{gathered}
$$

For this to be a useful, we need to note that matrix-vector multiplication *distributes*. You might be a little rusty on these old terms, and for good reason; we take these properties for granted all the time. Distributivity is the property that says, for instance, that $3x - 5x = (3-5)x$. We say that the multiplication by x is *distributed* across the terms. So what does that have to do with our situation here? Well, moving both matrix-vector products to the left side of the equation gives

$$\begin{bmatrix} 0.00 & 0.56 & 0.43 \\ 0.72 & 0.11 & 0.40 \\ 0.28 & 0.33 & 0.17 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$
$$C\mathbf{p} - R\mathbf{p} = \mathbf{0}$$

This is equivalent to saying that at the break even point, the cost minus the revenue is equal to zero for each industry independently. But, since matrix-vector multiplication distributes, we can write

$$\begin{aligned} C\mathbf{p} - R\mathbf{p} &= (C-R)\mathbf{p} \\ &= \left(\begin{bmatrix} 0.00 & 0.56 & 0.43 \\ 0.72 & 0.11 & 0.40 \\ 0.28 & 0.33 & 0.17 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix}. \end{aligned}$$

But this brings to another new topic: how do we add and subtract matrices? Well, if you had to define how to add or subtract two matrices of the same size, how would you do it? Seriously, take a second and think about it. Got an idea? Good. If you thought, "I'd line up the matrices and add or subtract the equivalent positions in each matrix", you're on the right track. If you thought something else, let me know; I'd love to hear your idea.

So let's see how to do this matrix subtraction.

$$\begin{aligned} \left(\begin{bmatrix} 0.00 & 0.56 & 0.43 \\ 0.72 & 0.11 & 0.40 \\ 0.28 & 0.33 & 0.17 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} &= \begin{bmatrix} 0.00-1 & 0.56 & 0.43 \\ 0.72 & 0.11-1 & 0.40 \\ 0.28 & 0.33 & 0.17-1 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix} \\ &= \begin{bmatrix} -1 & 0.56 & 0.43 \\ 0.72 & -0.89 & 0.40 \\ 0.28 & 0.33 & -0.83 \end{bmatrix} \begin{bmatrix} p_c \\ p_e \\ p_s \end{bmatrix}. \end{aligned}$$

But remember, the reason we started down this trail was to figure out the solutions to $C\mathbf{p} = R\mathbf{p}$. Now we know that these solutions are the same as the solutions to $(C-R)\mathbf{p} = \mathbf{0}$. We can easily do this computation in Matlab.

```
EDU>> C = [0.00 0.56 0.43; 0.72 0.11 0.40; 0.28 0.33 0.17]

C =
```

```
         0    0.5600    0.4300
    0.7200    0.1100    0.4000
    0.2800    0.3300    0.1700

EDU>> R = [1 0 0; 0 1 0; 0 0 1]

R =

     1     0     0
     0     1     0
     0     0     1
```

Now that we have the matrices C and R in Matlab, we can confirm that we didn't make any arithmetic mistakes earlier in our subtraction $C - R$.

```
EDU>> C-R

ans =

   -1.0000    0.5600    0.4300
    0.7200   -0.8900    0.4000
    0.2800    0.3300   -0.8300
```

But how do we make the augmented matrix of the linear system $(C - R)\mathbf{p} = \mathbf{0}$. We know from the previous section that we want the matrix $(C - R)$ the the column vector $\mathbf{0}$ append on the right side. On straightforward option is to type this all in manually.

```
EDU>> M = [-1.00 0.56 0.43 0.00; 0.72 -0.89 0.48 0.00; 0.28 0.33 -0.83 0.00]

M =

   -1.0000    0.5600    0.4300         0
    0.7200   -0.8900    0.4800         0
    0.2800    0.3300   -0.8300         0
```

But as you might imagine, this can be really cumbersome for large matrices. A much faster and more direct way is to make a new matrix with $C - R$ placed next to $\mathbf{0}$.

```
EDU>> M = [C-R [0;0;0]]

M =

   -1.0000    0.5600    0.4300         0
    0.7200   -0.8900    0.4000         0
    0.2800    0.3300   -0.8300         0
```

Regardless of how we computed the matrix M, we know what we have to do to find the solutions to the linear system: `rref` it!

```
EDU>> rref(M)

ans =

    1.0000         0   -1.2463         0
         0    1.0000   -1.4577         0
         0         0         0         0
```

The value of p_s is not constrained by any of the equations from the reduced row echelon form, and so it is a free variable. Only once we have chosen a particular value for p_s can the values of p_c and p_e be computed. In general, though, `rref(M)` tells us some interesting things about the nature of the solutions to this system. For instance, regardless of the annual output from the steel industry, for all industries to break even, the coal industry's output must be 124.63% that of the steel industry. Similarly, the electricity industry's annual output must be 145.77% of the steel industries output. Even if there is not a unique solution to a system of linear equations, we can often still tell quite a bit about what the solution space looks like.

1.3.2 Application of inhomogeneous equations: network flow and supply chain management

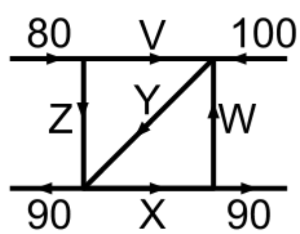


Figure 1.9: Graphical representation of a supply chain.

Imagine your business has a supply chain as depicted in figure 1.9. Arrows pointing into the network represent units of raw goods entering the manufacturing process, and arrows exiting the network represent goods going to consumers. At each junction (mathematicians call them *nodes* or *vertices*), some goods come in and some goods go out. For instance, at the top right node, we have 100 units of raw materials, W units of one product, and V units of another product all coming together to make Y units of the output product. Our key assumption here is that at each node, the total number of units entering the node is the same as the total number of units leaving the node. This way, we can write a system of linear equations, one for each node, that describes the relationships between the products in the supply chain.

$$\begin{aligned} (\text{in}) &= (\text{out}) \\ 80 &= v + z \\ 100 + v + w &= y \\ x &= w + 90 \\ y + z &= 90 + x \end{aligned}$$

By now we should be comfortable converting a linear system into its associated matrix equation

$$\begin{bmatrix} -1 & 0 & 0 & 0 & -1 \\ 1 & 1 & 0 & -1 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \\ x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -80 \\ -100 \\ 90 \\ 90 \end{bmatrix}.$$

Notice that the matrix equation is inhomogeneous. We then produce the RREF of the augmented matrix in order to learn about the solutions of the linear system

```
>> rref([-1 0 0 0 -1 -80; 1 1 0 -1 0 -100; 0 -1 1 0 0 90; 0 0 -1 1 1 90])

ans =

     1     0     0     0     1    80
     0     1     0    -1    -1  -180
     0     0     1    -1    -1   -90
     0     0     0     0     0     0
```

Here we see that we have not one but *two* free variables, y and z. Let's take a closer look at the first equation of the RREF, $v = 80 - z$. If we assume we can't have negative units of goods flowing through our supply chain, this equation implies that $z \leq 80$. Since v and z have a physical interpretation, their values can be constrained *even if one of them is a free variable.* We can carry this analysis a bit further by looking at the final nontrivial equation in the RREF, $x = y + z - 90$. Since we decided $z \leq 80$, we have $x \leq y - 10$, and so $y \geq 10$ even though y is technically a free variable.

It's worth saying again: if linear systems in which the variables have some sort of physical meaning, it's often the case that variables which RREF would indicate are completely free actually have constrained values due to the underlying assumptions of the linear system.

1.4 Linear Independence

Let's consider the supply chain diagram found in figure TODO. Units of products or raw materials move along each line in the graph, with the direction of movement indicated by arrows and the number of units represented either by a fixed quantity (*e.g.*, 60, 70, 80) or a variable (*e.g.*, $x_1, x_2, \ldots, x_6$). We will call the product whose arrow is labeled by x_i product i; for instance, the top-most edge represents product 1. Arrows coming into the system represent raw materials, and arrows going out of the system represent final goods going to consumers. Internal arrows represent intermediate products that will either be sold to end consumers or will be used as components in a new product. At each lettered node, a new product is formed. For instance, at node C, we have x_2 units of a product 2 combine with 100 units of raw materials to form x_3 units of product 3. We'll also impose the constraint that all units, whether raw materials or product, that go into a node, must come out. At node A, any units of product 6 that have not been sold are recycled and combined with 60 units of raw materials to restart the process with x_1 units of product 1.

The question we'll try to be answering here is this: what are internal production levels $x_1, x_2, \ldots, x_6$ that need to be maintained in order to exactly satisfy customer demand, represented by outgoing arrows, subject to raw material availability, represented by incoming arrows? It seems like a natural enough question from a business perspective. It would be wasteful to make more products than you can sell, and you certainly don't want to have raw materials you've purchased not going into products. But how do we go about trying to solve something like this? And how can we be sure that there's a solution at all?

Let's start by writing down what we know. We know that at each node, for every unit of product that comes in, a unit of product must come out. So at each lettered node, we can just equate the number of units coming in to the number of units coming out. This gives us a system of linear equations in $x_1, x_2, \ldots, x_6$.

$$\begin{aligned} x_6 + 60 &= x_1 \\ x_1 &= x_2 + 70 \\ x_2 + 100 &= x_3 \\ x_3 &= 90 + x_4 \\ x_4 + 80 &= x_5 \\ x_5 &= 80 + x_6 \end{aligned} \quad \leftrightarrow \quad \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} -60 \\ 70 \\ -100 \\ 90 \\ -80 \\ 80 \end{bmatrix}.$$

We know by now that to solve this system of linear equations, we would just `rref` it's augmented matrix.

```
EDU>> rref([-1 0 0 0 0 1 -60; 1 -1 0 0 0 0 70; 0 1 -1 0 0 0 -100; ...
0 0 1 -1 0 0 90; 0 0 0 1 -1 0 -80; 0 0 0 0 1 -1 80])
```

```
ans =

     1     0     0     0     0    -1    60
     0     1     0     0     0    -1   -10
     0     0     1     0     0    -1    90
     0     0     0     1     0    -1     0
     0     0     0     0     1    -1    80
     0     0     0     0     0     0     0
```

So x_6 is a free variable, and there are infinitely many solutions to this linear system. But what if we change the inputs just a little bit? For instance, imagine 61 units of raw materials comes into node A.

```
EDU>> rref([-1 0 0 0 0 1 -61; 1 -1 0 0 0 0 70; 0 1 -1 0 0 0 -100; ...
0 0 1 -1 0 0 90; 0 0 0 1 -1 0 -80; 0 0 0 0 1 -1 80])

ans =

     1     0     0     0     0    -1     0
     0     1     0     0     0    -1     0
     0     0     1     0     0    -1     0
     0     0     0     1     0    -1     0
     0     0     0     0     1    -1     0
     0     0     0     0     0     0     1
```

So this slightly different system has no solutions, because the rightmost column contains a pivot. It may seem a little weird that making such a small change in the assumptions of the model took us from infinitely many solutions to no solutions. As it turns out, there's a lot of relevant mathematics to describe why this happens.

1.4.1 Counting solutions

Let's start by thinking about perhaps the easiest linear system we can, $A\mathbf{x} = \mathbf{0}$. The vector $\mathbf{x} = \mathbf{0}$ is always a solution to this equation. Since it's so easy to satisfy the equality with this choice of $\mathbf{x}$, we call $\mathbf{x} = \mathbf{0}$ the *trivial solution* to $A\mathbf{x} = \mathbf{0}$. A much more interesting question is whether there are any nonzero solutions to the equation. But before we get into whether these exist, let's talk about why such a thing should be interesting.

Imagine that there are infinitely many $\mathbf{k}$ such that $A\mathbf{k} = \mathbf{0}$, and let's assume too that there is a vector $\mathbf{q}$ such that $A\mathbf{q} = \mathbf{b}$ for some fixed vector $\mathbf{b}$. Remembering our matrix arithmetic properties from section **??**, we know that $A(\mathbf{k} + \mathbf{q}) = A\mathbf{k} + A\mathbf{q} = \mathbf{0} + \mathbf{b} = \mathbf{b}$. In words, if $\mathbf{k} + \mathbf{q}$ is a solution to $A\mathbf{x} = \mathbf{b}$, too. So if there are infinitely many $\mathbf{k}$ such that $A\mathbf{k} = \mathbf{0}$, then there are infinitely many $\mathbf{x}$ such that $A\mathbf{x} = \mathbf{b}$ provided that just one solution $\mathbf{q}$ exists. (Note that if this initial $\mathbf{q}$ doesn't exist, then none of the rest is true!)

Let's turn the tables and assume that there are infinitely many solutions to $A\mathbf{x} = \mathbf{b}$ for the same fixed $\mathbf{b}$. Choose just one of these and call it $\mathbf{q}$, and let $\mathbf{r}$ range over all solutions. By the same good old matrix arithmetic properties, we have $A\mathbf{q} - A\mathbf{r} = \mathbf{b} - \mathbf{b} = \mathbf{0}$ and $A\mathbf{q} - A\mathbf{r} = A(\mathbf{q} - \mathbf{r})$ so that $\mathbf{q} - \mathbf{r}$ is a solution to $A\mathbf{x} = \mathbf{0}$. Since there are infinitely many choices for $\mathbf{r}$, there must be infinitely many solutions to $A\mathbf{x} = \mathbf{0}$.

But why are we bothering with this at all? Well, the previous two paragraphs tell us that we can learn a lot about the number of solutions to $A\mathbf{x} = \mathbf{b}$ just by looking at the solutions to the (much simpler) system $A\mathbf{x} = \mathbf{0}$. And for a bonus, we have the nice property that $rref([A|\mathbf{0}]) = [rref(A)|\mathbf{0}]$ since scaling and adding rows can never change a rightmost column full of zeros. So here's the take away message: to determine how many solutions to $A\mathbf{x} = \mathbf{b}$ exist, a good place to start is to look at `rref(A)`. And why is this good again? Well, since `rref(A)` is independent of the particular choice of $\mathbf{b}$, we're getting information about *all possible* systems with coefficients described by A at the *same time*.

1.4.2 Rank

Let's update our previous results about solutions to linear systems using the new information we've just uncovered. A linear system $A\mathbf{x} = \mathbf{b}$

- a unique solution if and only if the `rref(A)` has a pivot in every row and column
- either no solution or infinitely many solutions if `rref(A)` otherwise

This formulation seems to imply that the number of pivots in `rref(A)` is an important quantity in determining how many solutions a linear system has. In fact, it's so important it gets its own name. We call the number of pivots contained in `rref(A)` the *rank of A*, denoted $rank(A)$. The Matlab command to compute the rank of the coefficient matrix A is just `rank(A)`. We can use the coefficient matrix from our supply chain network as an example.

```
EDU>> rank([-1 0 0 0 0 1; 1 -1 0 0 0 0; 0 1 -1 0 0 0; ...
0 0 1 -1 0 0; 0 0 0 1 -1 0; 0 0 0 0 1 -1])

ans =

     5
```

Note that since each row can contain at most one pivot, for a $r \times c$ coefficient matrix A, we have $rank(A) \leq r$. Similar reasoning applied to columns shows that $rank(A) \leq c$. Some interesting results follow from this.

- the linear system $A\mathbf{x} = \mathbf{b}$ has a unique solution if and only if $r = c = rank(A)$
- if $c > r \geq rank(A)$, then the linear system $A\mathbf{x} = \mathbf{b}$ must have free variables and therefore has infinitely many solutions.

But what about the remaining case where $r > c \geq rank(A)$? Well, this case covers a lot of ground, and we could have any one of the three types of solution spaces: unique solution, no solution and infinitely many solutions. For a unique solution, consider the linear system

$$\begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \\ 6 \end{bmatrix}$$
$$A_1\mathbf{x} = \mathbf{b}_1$$

We can compute the solution to this system in Matlab.

```
EDU>> rref([1 2 3; 4 5 6; 2 4 6])

ans =

     1     0    -1
     0     1     2
     0     0     0
```

Notice that $rank(A_1) = 2$, and that the system has a unique solution $x_1 = -1, x_2 = 2$.

For the no solution case, consider slightly altering the vector $\mathbf{b}_1$.

$$\begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \\ 7 \end{bmatrix}$$
$$A_0\mathbf{x} = \mathbf{b}_0$$

Again, we compute the solution in Matlab.

```
EDU>> rref([1 2 3; 4 5 6; 2 4 7])

ans =

     1     0     0
     0     1     0
     0     0     1
```

While $rank(A_0) = rank(A_1) = 2$, we now have an inconsistency in the system, captured in the third row of the reduced row echelon form of the augmented matrix.

For the infinitely many solutions case, consider the system

$$\begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \\ 9 \end{bmatrix}$$
$$A_\infty\mathbf{x} = \mathbf{b}_\infty$$

We still have $rank(A_\infty) = rank(A_0) = rank(A_1) = 2$, but now the reduced row echelon form of the augmented matrix is

```
EDU>> rref([1 2 3; 2 4 6; 3 6 9])

ans =

     1     2     3
     0     0     0
     0     0     0
```

Notice that x_2 is now a free variable, and so the system has infinitely many solutions.

1.4.3 Linear independence

There's another way to think about the non-trivial solutions of $A\mathbf{x} = \mathbf{0}$. We say that a collection of two or more vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c$ are *linearly independent* if

$$\begin{bmatrix} | & | & \cdots & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_c \\ | & | & \cdots & | \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_c \end{bmatrix} = \mathbf{0}$$

has only the trivial solution. If there exists a nontrivial linear combination of the columns equalling $\mathbf{0}$, then we say the vectors are *linearly dependent*. Using linear independence terminology, our results concerning the solutions of the system $A\mathbf{x} = \mathbf{b}$ with A a $r \times c$ matrix become

- has a unique solution if $r = c$ and the columns of A are linearly independent
- if the columns of A are linearly dependent, then the system has infinitely many solutions
- if the columns of A are linearly independent and $r > c$, then the system has either infinitely many solutions or no solution.

1.4.4 Span

We say that a vector $\mathbf{b}$ is in the *span* of the vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c$ if the linear system

$$\begin{bmatrix} | & | & \cdots & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_c \\ | & | & \cdots & | \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_c \end{bmatrix} = \mathbf{b}$$

has at least one solution. We call the collection of all such $\mathbf{b}$ the *span of* $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c$. Said a different way, the linear system $A\mathbf{x} = \mathbf{b}$ has at least one solution if and only if $\mathbf{b}$ lies in the span of the columns of A.

A linear system $A\mathbf{x} = \mathbf{b}$ with A an $n \times n$ matrix has a unique solution if and only if

- $A\mathbf{x} = \mathbf{0}$ has only the trivial solution
- $rank(A) = n$
- A has n linearly independent columns
- the span of the columns of A is $\mathbb{R}^n$

Chapter 2

Input-output models and invertibility

2.1 Leontief Input-Output Model

We've already seen examples of supply networks in which product A is both used to make product B and is sold itself directly to the consumer. We can formalize these notions by calling the former *intermediate demand* and the latter *final demand*. Let's investigate systems featuring intermediate and final demands in a slightly different context than supply chain networks.

Imagine we have two industries, manufacturing and services, and suppose that in order to make 1 unit of output, the manufacturing sector must consume 0.4 units of its own output, and 0.2 units of service industry output. (Here "units of output" could be measured in whatever way we want so long as the measurement method is consistent across the industries.) Similarly, suppose that in order to make 1 unit of output, the services industry must consume 0.7 units of the manufacturing industries output, and 0.1 units of its own output. Let x_1, x_2 be the number of units produced by the manufacturing and services industries, respectively. Then the intermediate demand necessary to create these units out production is

$$\begin{bmatrix} 0.4 & 0.7 \\ 0.2 & 0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Suppose now that we introduce a final demand vector $\mathbf{d}$ representing the number of units of production of both the manufacturing and services industry that are demanded not by other industries, but by the public at large. Ideally, the total number of units produced by both industries must be equal to the sum of the intermediate and final demands; this is just the familiar idea of supply equaling demand in equilibrium. Mathematically, we can express this idea as follows

$$\begin{aligned} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \begin{bmatrix} 0.4 & 0.7 \\ 0.2 & 0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \\ \mathbf{x} &= C\mathbf{x} + \mathbf{d} \end{aligned}$$

Suppose that we have somehow measured of estimated the demands contained in $\mathbf{d}$, and we're trying to determine our ideal production levels $\mathbf{x}$. Rearranging the equation a little bit gives

$$\begin{aligned} \mathbf{x} &= C\mathbf{x} + \mathbf{d} \\ (I - C)\mathbf{x} &= \mathbf{d}. \end{aligned}$$

This type of linear system is called a *Leontief input-output model.*

In the past, we've solved systems of linear equations like this input-output model using a `rref` in Matlab. We know how to interpret the results of this computation in terms of unique solutions, free variables and all the rest. But `rref` isn't an ideal solution generally for a couple of reasons. First, an individual `rref` computation doesn't help us do another `rref` computation more efficiently;

for a different demand vector $\mathbf{d}'$, we would simply repeat the entire process. Second, `rref` is almost useless in a theoretic context; sometimes, like it or not, pushing symbols around leads to serious discoveries about fundamental properties of a given system. Here's my claim: it would be great if we could find a matrix A such that

$$\begin{aligned} A(I - C)\mathbf{x} &= A\mathbf{d} \\ I\mathbf{x} = \mathbf{x} &= A\mathbf{d}. \end{aligned}$$

If you think about it, this is exactly what's happening every time you solve for x in an equation involving only real numbers. For instance when solving the equation $7x = 14$, you find a number a such that $a \cdot 7x = 1x = x$, namely $a = 1/7$. While it's true that such an inverse element a will exist over the real numbers, it's not the case that an inverse element will always exist when we're dealing with matrices. If the inverse of a matrix C exists, we denote it C^{-1}. Just as if real numbers, the matrix C^{-1} satisfies $C^{-1}C = I = CC^{-1}$.

But you might already have some issues with this concept. For instance,

- What does AB mean when A and B are matrices? How do define matrix-matrix multiplication? Is it related to the concept of matrix-vector multiplication that we've already seen?
- For a matrix C, does the inverse C^{-1} exist? How can we know if it does or doesn't?
- If the inverse C^{-1} does exist, how can we compute it? We want concepts we can *use*, and if we can't compute the inverse, we can't put it to use.

We're going to deal with each of these issues in turn, and by the end, we'll have solid answers for each question.

2.1.1 Matrix-matrix multiplication

Suppose that A is a $r \times n$ matrix and B is a $n \times c$ matrix with columns $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_c$. The matrix multiplication of A and B is defined as

$$\begin{aligned} AB &= A \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \ldots & \mathbf{b}_c \end{bmatrix} \\ &= \begin{bmatrix} A\mathbf{b}_1 & A\mathbf{b}_2 & \ldots & A\mathbf{b}_c \end{bmatrix}. \end{aligned}$$

The product AB will be $r \times c$. Notice that matrix-matrix multiplication AB makes sense only if AB has dimensions $(r \times n)(n \times c)$ so that the "inner dimensions" of A and B are identical.

Reading Check 22: Verify the previous two claims.

Matrix-matrix multiplication in Matlab is easy: just enter each matrix as you normally would and place a `*` in between. In all the following examples, before actually doing the matrix multiplication AB in Matlab or by hand, determine what the dimensions of the product will be.

Example 5: $A = \begin{bmatrix} 1 & 2 \\ 3 & -4 \end{bmatrix}, \quad B = \begin{bmatrix} 7 & 3 & -1 \\ -2 & 4 & 9 \end{bmatrix}$

Example 6: $A = \begin{bmatrix} 7 & -2 \\ 4 & 3 \\ -1 & 9 \end{bmatrix}, \quad B = \begin{bmatrix} 3 & 5 \\ 1 & 2 \end{bmatrix}$

Example 7: $A = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$

Example 8: $A = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad B = \begin{bmatrix} 3 & 4 & 5 \end{bmatrix}$

You may have noticed that the last two examples were especially strange. In Example 2.1.1, we took two vectors and produced a *scalar* (*i.e.*, a single number). This is known as the *inner product* of the vectors. The inner product of vectors $\mathbf{v}$ and $\mathbf{w}$ is usually denoted $\mathbf{v} \circ \mathbf{w}$ or $< \mathbf{v}, \mathbf{w} >$. In Example 2.1.1, we took two vectors and produced a matrix. This is known as the *outer product* of the vectors. The standard notation for the outer product of vectors $\mathbf{v}$ and $\mathbf{w}$ is $\mathbf{v} \otimes \mathbf{w}$. We'll see both of these concepts in the course, but we can use the inner product right away. We can define matrix multiplication another way by saying that the (i, j) entry of matrix AB is the inner product of row i in A and column j in B.

Reading Check 23: Verify that the previous claim is true.

Matrix multiplication might be the first time in a person's mathematical life that arithmetic starts getting a little strange. But before we get to all the weird stuff, let's enumerate some things in matrix multiplication that work the same way they do in multiplication of real numbers:

1. $A(BC) = (AB)C$ (*associativity*)
2. $A(B + C) = AB + AC$ (*left distributivity*)
3. $(A + B)C = AC + BC$ (*right distributivity*)
4. $I_r A = A = AI_c$ assuming A is $r \times c$(*identity*)

But there are some things that don't work the way we might expect, too.

1. In general, $AB \neq BA$. (*non-commutative*)
2. In general, $AB = AC$ does not imply that $B = C$.
3. In general, $AB = 0$ does not imply that $A = 0$ or $B = 0$.

The one to really worry about the most is the first one. We can't just rearrange the order in a product. We won't dwell on these too long since you'll be doing a problem about each of these.

Let's use our knew knowledge of matrix multiplication to do a quick example about inverses.

Example 9: Show that $CD = DC = I_2$, so that $D = C^{-1}$, where

$$C = \begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix}, \quad D = \begin{bmatrix} 2 & -3 \\ -3 & 5 \end{bmatrix}.$$

Example 10: Show that $EF = FE = I_2$, so that $E = F^{-1}$, where

$$E = \begin{bmatrix} 7 & 2 \\ 3 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & -2 \\ -3 & 7 \end{bmatrix}.$$

Before we dive into computing the inverse of a matrix, let's gather up some preliminary results. For now, let's consider an arbitrary $r \times c$ matrix A. Generally speaking, the matrix A takes a vector of length c and produces a vector of length r. Another way of saying this is that the domain of C is $\mathbb{R}^c$ and the range is $\mathbb{R}^r$. Remember, for an inverse of A to exist, we need $A\mathbf{x} = \mathbf{b}$ to have a unique solution $\mathbf{x} \in \mathbb{R}^c$ for every $\mathbf{b}\mathbb{R}^r$. Also remember that there can be at most one pivot per row and at most one pivot per column.

Suppose $c > r$. Then since there is at most one pivot per row, there exists some column without a pivot. Then A has a free variable, and so $A\mathbf{x} = \mathbf{b}$ has infinitely many solutions for some $\mathbf{b} \in \mathbb{R}^r$. So A cannot have an inverse.

Suppose $c < r$. Then since there is at most one pivot per column, there exists some row without a pivot. Then $rref(A)$ has a row of all zeros, and so $A\mathbf{x} = \mathbf{b}$ has no solution for some $\mathbf{b} \in \mathbb{R}^r$. And so A cannot have an inverse here either.

Reading Check 24: Make sure you believe the following two paragraphs. It might help to make some small examples if you're still in doubt.

Combining the two previous paragraphs, we can say that for a matrix A to have any hope of having an inverse, it must be the case that $c = r$. In this case, we call A a *square* matrix.

But even if A is $n \times n$, we can still run into problems. For a ridiculous case, imagine that A is a square matrix containing all zeros. Then A maps every input vector $\mathbf{x}$ to the output vector $\mathbf{y} = \mathbf{0}$. So clearly A is not a one-to-one mapping in this case. We've seen that in general if the columns of A are not linearly independent, then `rref(A)` contains a free variable, and so there are some $\mathbf{y}$ in the range of A which have infinitely many solutions $\mathbf{x}$ satisfying $A\mathbf{x} = \mathbf{y}$. Hence, if the columns of A are linearly dependent, then A is not an one-to-one mapping, and hence there can be no inverse for A.

So suppose that the columns of A are linearly independent. Then $A\mathbf{x} = \mathbf{0}$ has only the trivial solution, and so `rref(A)` can have no free variables. Since A is square, this implies that `rref(A)` is an identity matrix. Therefore, for every $\mathbf{b} \in \mathbb{R}^n$, there is a unique solution $\mathbf{x}$ such that $A\mathbf{x} = \mathbf{b}$. This is exactly what we want in order for the inverse of A to exist!

So we can add the existence of a matrix inverse to our list of equivalent conditions concerning linear independence of the columns of a matrix. Here we assume that A is $n \times n$.

- The equation $A\mathbf{x} = \mathbf{b}$ has a unique solution $\mathbf{x}$ for every $\mathbf{b} \in \mathbb{R}^n$.
- The equation $A\mathbf{x} = \mathbf{0}$ has only the trivial solution $\mathbf{x} = \mathbf{0}$.
- The columns of a A are linearly independent.
- `rref(A)` contains no free variables.
- A has n pivots.
- The inverse A^{-1} exists.

So far we have determined what matrix-matrix multiplication is and how to tell whether a matrix A has an inverse. But how do actually construct the matrix inverse?

2.2 The matrix inverse

2.2.1 Building intuition

Before we get into the nitty gritty, let's try to form a hypothesis about what a matrix inverse of a 2×2 matrix A will look like. In Matlab, try

```
EDU>> A = [5 3; 3 2]

A =

     5     3
     3     2

EDU>> A^(-1)
```

What do you see? How are the entries of A^{-1} related to the entries of A?

Next try

```
EDU>> A = [3 1; 5 2]

A =

     3     1
```

```
     5     2

EDU>> A^(-1)
```

Does your hypothesis from the first example fit the second example?
One last example:

```
EDU>> A = [4 7; 2 4]

A =

     4     7
     2     4

EDU>> A^(-1)
```

How is your hypothesis holding up now? If it failed, can you revise it to make sense of all three examples?

2.2.2 A first matrix inverse

So far we know that a matrix inverse of A exists if and only if A is square and has linearly independent columns. But this doesn't help much with the actual computation. Consider the general 2×2 matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

Let's make two observations:

$$\begin{aligned} A \begin{bmatrix} 1 \\ 0 \end{bmatrix} &= \begin{bmatrix} a \\ c \end{bmatrix} \\ A \begin{bmatrix} 0 \\ 1 \end{bmatrix} &= \begin{bmatrix} b \\ d \end{bmatrix} \end{aligned}$$

If A^{-1} exists, we can multiply both sides of both equations by A^{-1} on the left.

$$\begin{aligned} A^{-1} A \begin{bmatrix} 1 \\ 0 \end{bmatrix} &= A^{-1} \begin{bmatrix} a \\ c \end{bmatrix} \\ A^{-1} A \begin{bmatrix} 0 \\ 1 \end{bmatrix} &= A^{-1} \begin{bmatrix} b \\ d \end{bmatrix}. \end{aligned}$$

To find out what A^{-1} actually is, let's first define its components and then solve for them using the relations we just found. Let

$$A^{-1} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}.$$

Remember, we're given A, so we know exactly the values of a, b, c and d. We don't know e, f, g or h; these are variables. Our relations from above become

$$\begin{aligned} A^{-1}\begin{bmatrix} a \\ c \end{bmatrix} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}\begin{bmatrix} a \\ c \end{bmatrix} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \begin{bmatrix} ae + cf \\ ag + ch \end{bmatrix} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \end{aligned}$$

and

$$\begin{aligned} A^{-1}\begin{bmatrix} a \\ c \end{bmatrix} = \begin{bmatrix} e & f \\ g & h \end{bmatrix}\begin{bmatrix} b \\ d \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \begin{bmatrix} be + df \\ bg + dh \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{aligned}$$

We have 4 equations in the variables e, f, g and h. We can gather these up in to a single system of linear equations.

$$\begin{aligned} \begin{bmatrix} ae + cf \\ ag + ch \end{bmatrix} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \begin{bmatrix} be + df \\ bg + dh \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \Rightarrow \begin{bmatrix} a & c & 0 & 0 \\ 0 & 0 & a & c \\ b & d & 0 & 0 \\ 0 & 0 & b & d \end{bmatrix}\begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix} &= \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \end{aligned}$$

We can use a tool like WolframAlpha to solve this system. We find

$$\begin{aligned} e &= \frac{d}{ad - bc} \\ f &= \frac{-b}{ad - bc} \\ g &= \frac{-c}{ad - bc} \\ h &= \frac{a}{ad - bc}. \end{aligned}$$

But remember, the whole point of this exercise was to find the components of the inverse matrix A^{-1}. We can now just read them off.

$$\begin{aligned} A^{-1} &= \begin{bmatrix} e & f \\ g & h \end{bmatrix} \\ &= \frac{1}{ad - bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}. \end{aligned}$$

We call $ad - bc$ the *determinant* of A, denoted $\det(A)$. Notice that the inverse of A that we've computed here only makes sense if $\det(A) \neq 0$.

Reading Check 25: Show that we've found is accurate by computing AA^{-1} and $A^{-1}A$.

Notice that the same type of reasoning used in computing the inverse of a given 2×2 matrix could be used to explicitly construct the inverse of any $n \times n$ matrix!

Reading Check 26: Find the inverse of the arbitrary 3×3 matrix

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

2.2.3 Computing production

Now that we have the inverse of an arbitrary 2×2 matrix, computing the total production necessary to satisfy the demand in a Leontief input-output model should be a piece of cake. For instance, imagine that the total demand is 100 units from the manufacturing industry and 200 units from the services industry. Then our matrix equation looks like

$$\left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.4 & 0.7 \\ 0.2 & 0.1 \end{bmatrix} \right) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 100 \\ 200 \end{bmatrix}$$
$$A\mathbf{x} = \mathbf{d}$$

So we first need to compute A^{-1} and then multiply by $\mathbf{d}$. Since we've already done the tough stuff, we can literally just substitute values in order for find the inverse.

$$\begin{aligned} A^{-1} &= \frac{1}{(0.6)(0.9) - (0.7)(0.2)} \begin{bmatrix} 0.9 & 0.7 \\ 0.2 & 0.6 \end{bmatrix} \\ &= 2.5 \begin{bmatrix} 0.9 & 0.7 \\ 0.2 & 0.6 \end{bmatrix} \\ &= \begin{bmatrix} 2.25 & 1.75 \\ 0.5 & 1.5 \end{bmatrix} \end{aligned}$$

Not so bad, right? But a lot of times, things don't work out so neatly. In those cases, it's usually better just to use Matlab. We can compute the inverse of A using Matlab in a couple different ways.

```
EDU>> A = eye(2) - [0.4 0.7; 0.2 0.1]

A =

    0.6000   -0.7000
   -0.2000    0.9000

EDU>> inv(A)

ans =

    2.2500    1.7500
    0.5000    1.5000

EDU>> A^(-1)

ans =

    2.2500    1.7500
    0.5000    1.5000
```

All three methods agree, so we know we're doing everything correctly. So to compute the number of units needed to satisfy demand **d**, we just need to multiply.

```
>> d = [100;200]

d =

   100
   200

>> inv(A) * d

ans =

  575.0000
  350.0000
```

So the manufacturing industry and service industry must produce 625 units and 100 units, respectively, in order to satisfy both intermediate and final demand exactly. This type of operation comes up so frequently that Matlab has provided an even easier way to compute $\mathbf{x} = A^{-1}\mathbf{d}$.

```
>> A \ d

ans =
```

```
   575
   350
```

Notice that this is a backslash, not forward slash as we typically use in division!

2.2.4 Consequences of linearity to sensitivity

How does the production solution change if the final demand from industry 2 increases by 1 unit? Well,

```
EDU>> (eye(2) - C) \ [100;201]

ans =

  576.7500
  351.5000
```

In words, we need 576.75 total units from industry 1 and 351.5 total units from industry 2. Combining the two preceding results, we can write a numerical form of the change in production associated with an increase in final demand from industry 2 of 1 unit:

$$\begin{aligned}
\Delta \mathbf{x}_2 &= \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 100 \\ 201 \end{bmatrix} - \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 100 \\ 200 \end{bmatrix} \\
&= \begin{bmatrix} 576.75 \\ 351.5 \end{bmatrix} - \begin{bmatrix} 575 \\ 350 \end{bmatrix} \\
&= \begin{bmatrix} 1.75 \\ 1.5 \end{bmatrix}.
\end{aligned}$$

We can think of $\Delta \mathbf{x}_2$ as the sensitivity of production to changes in final demand of output from industry 2. But this sensitivity information is contained within the inverse of $I - C$. We can see this by developing the original expression for $\Delta \mathbf{x}_2$ along a different track.

$$\begin{aligned}
\Delta \mathbf{x}_2 &= \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 100 \\ 201 \end{bmatrix} - \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 100 \\ 200 \end{bmatrix} \\
&= \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \left(\begin{bmatrix} 100 \\ 201 \end{bmatrix} - \begin{bmatrix} 100 \\ 200 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}.
\end{aligned}$$

So the change in production associated with an increase of 1 unit of final demand of product 2 is the second column of the matrix $(I - C)^{-1}$.

Reading Check 27: Convince yourself that for any $r \times c$ matrix A, the matrix multiplication $A[0, \ldots, 0, 1, 0, \ldots, 0]^T$, where the single 1 is located in the j^{th} component, $1 \leq j \leq c$, returns the j^{th} column of A.

A simple Matlab computation of $(I - C)^{-1}$ confirms that the two interpretations are identical.

```
EDU>> inv(eye(2) - C)

ans =

    2.2500    1.7500
    0.5000    1.5000
```

Notice that this is general, meaning we could repeat a similar procedure and determine the additional production necessary to accommodate 1 additional unit of demand from any of the industries, and that this additional production would appear as a column of the inverse of $(I - C)$.

2.2.5 Parameterized Leontief input-output models

The Leontief input-output model that we considered previously took the form

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.4 & 0.7 \\ 0.2 & 0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$
$$\mathbf{x} = C\mathbf{x} + \mathbf{d},$$

where $\mathbf{x}$ represented the number of units produced of each good, and $\mathbf{d}$ represented the final demand. Just so we have some terminology to throw around, remember that we called $C\mathbf{x}$ the *intermediate demand*.

We reframed the idea of finding a specific production level $\mathbf{x}$ that would perfectly satisfy demand as a matrix inverse problem, in particular the problem of finding the inverse of the matrix $I - C$.

Just as in our investigations of eigenvalues and eigenvectors, it is often a productive exercise to think of one of the entries of C as a tunable parameter. This serves two functions: first, it let's us represent uncertainty as to the exact values of the matrix C. Remember, the (i, j) entry of C represents the number of units of product i that are necessary to produce 1 unit of product j.

Reading Check 28: Remind yourself why the preceding statement is true.

We could easily imagine that this value fluctuates or is not known definitively, and so having some control over the value of the (i, j) entry of C could give us some information about how the system behaves as various intermediates demands change. Another reason to introduce tunable parameters is more mathematical: generalizing the matrix C will help us build intuition about when and how inverses exist or fail to exist.

Let's consider an input-output model where the number of units of product 1 necessary to make 1 unit of product 2 is represented by the variable k. Since an industry should be adding value by creating a new product, it's safe to assume that $k < 1$, and since it doesn't make sense to have negative values of production, we can also feel good about bounding $k \geq 0$. Our model now takes the form

$$\begin{aligned} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \begin{bmatrix} 0.4 & k \\ 0.2 & 0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \\ \mathbf{x} &= C_k \mathbf{x} + \mathbf{d}, \end{aligned}$$

Here's a natural question: for what values of k does there exist a unique production level that satisfies final demand exactly? Mathematically, we're asking ourselves the following question: for what values of k does an inverse of $I - C_k$ exist.

We have many different and equivalent ways that characterize whether the inverse of a particular matrix exists. In some ways, the only real mathematics involved in the process of determining whether an inverse exists or not is having some insight into which formulation will give use the easiest route to determining the existence of the inverse. For instance, we could try to prove that the determinant of the matrix is nonzero. Or, alternatively and equivalently, we could try to prove that the columns of the matrix are linearly independent. Or we could try to prove that the kernel of the matrix contains only the zero vector. All of these (and more) and perfectly valid, but one or more will usually prove to be easier than the others.

Since computing the determinant of a 2×2 matrix is easy, let's start there. Remember, we're trying to investigate the matrix $I - C_k$, not just C_k itself. So the question becomes this: for what values of k does the following statement hold:

$$\det \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.4 & k \\ 0.2 & 0.1 \end{bmatrix} \right) \neq 0$$

Simplifying the matrices first and then taking the determinant gives

$$\det(I - C_k) = (0.6)(0.9) - (0.2)k.$$

Reading Check 29: Verify the previous claim.

Notice that the determinant is linear in k, and hence there will be one and only one value of k for which $\det(I - C_k)$ equals any given value, and in particular only one value of k for which the determinant is zero. A little bit of mathematical elbow grease will give us our answer.

$$\begin{aligned} 0 &= .54 - .2k \\ k &= 2.7. \end{aligned}$$

Bringing this result back to the context of the problem, there will exist a unique production level which uniquely satisfies *any* final demand vector $\mathbf{d}$ so long as $k \neq 2.7$.

Reading Check 30: Repeat the following procedure assuming that the (2,1) entry of C is unknown.

It's not much of a stretch to imagine that a different entry in the matrix C is unknown. For instance, let's imagine that entry (2,2) of C is unknown. In terms of our model, this means that we're unsure how many units of product 2 will be recycled to make a single unit of product 2. Our model now takes the form

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.4 & 0.7 \\ 0.2 & k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$
$$\mathbf{x} = B_k\mathbf{x} + \mathbf{d},$$

(Note: I'm using B_k here just to clearly denote that the matrix we're considering here is not exactly the same as the C_k considered before.) Again, let's determine when the inverse of $I - B_k$ by using determinants. Here the determinant condition for invertability takes the form

$$\det\left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.4 & 0.7 \\ 0.2 & k \end{bmatrix}\right) \neq 0$$

Simplifying and performing the determinant computation gives

$$\det(I - B_k) = (0.6)(1-k) - (0.7)(0.2)$$

Again, the determinant is linear in k, and so there is one and only one value of k for which $I - B_k$ is not invertible:

$$\begin{aligned} 0 &= (0.6)(1-k) - (0.7)(0.2) \\ k &= 1 - \frac{0.14}{0.6} \\ &\approx 0.77 \end{aligned}$$

Reading Check 31: Verify the previous computation.

Here, the computed value of k certainly does lie within the bounds we set up earlier. In other words, this value of k could conceivably come up in the real world. But what would it mean if it would? Remember, systems of linear equations either have a unique solution, no solution or infinitely many solutions. We've just proved that if k in this situation has a particular value, then it is definitely the case that the a unique solution to the Leontief input-output model does not exist. Therefore, we can conclude that there are either infinitely many production levels that perfectly satisfy a given final demand, or no production level that perfectly satisfies a given final demand. Which situation actually occurs completely depends on the given final demand.

How about a more difficult example? Consider a 3 industry Leontief input-output model of the form

$$\begin{aligned}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} &= \begin{bmatrix} 0.3 & 0.2 & 0.4 \\ 0.3 & 0.5 & 0.4 \\ 0.3 & 0.2 & k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \\ \mathbf{x} &= C_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

As with the previous example, there are a lot of different ways that we could determine when $I - C_k$. It's tough to say which is going to be the easiest, but let's try to start with analyzing the determinant condition. But we've never talked about how to take the determinant of a 3×3 matrix by hand! Have no fear! WolframAlpha eats problems like this for breakfast. Let's try entering the following statement into WolframAlpha:

```
det IdentityMatirx[3]-{{0.3, 0.2, 0.4},{0.3, 0.5, 0.4},{0.3, 0.2, k}}
```

Now, we could've used the statement

```
det {{1-0.3, -0.2, -0.4},{-0.3, 1-0.5,-0.4},{-0.3, -0.2, 1-k}}
```

You might think one or the other is a little better. Regardless of which one we end up choosing, the result is

$$\det(I - C_k) = 0.126 - 0.29k.$$

Then the determinant condition of invertibility tells us that $I - C_k$ is non-invertible if and only if

$$\begin{aligned} 0 &= 0.126 - 0.29k \\ k &\approx 0.434. \end{aligned}$$

Now, for the grain of salt. We've shown that a matrix inverse of $I - C_k$ fails to exist if and only if k is one particular value. In the real world, such a narrow window of badness is very rarely realized. That being said, sometimes very weird things can happen if a matrix is "close" to be non-invertible. In other words, we could see strange behavior of our production levels if k is very near the point at which a matrix inverse fails to exist. The point here is that we as mathematically oriented business people need to be aware of the failures of our models and do our best to mitigate those failures.

2.3 Leontief price equation

Let's imagine that industry i charges p_i dollars (or other unit of currency) for each unit of its output. Considering a Leontief system with 2 different industries, we can bundle these prices into a *price vector* $\mathbf{p}$, where the i^{th} component of p is p_i, the price charged by industry i for 1 unit of its output. For a concrete example, let's go back to the manufacturing and services example that we investigated previously. Remember, this model took the form

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.4 & 0.7 \\ 0.2 & 0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$
$$\mathbf{x} = C\mathbf{x} + \mathbf{d}$$

Imagine now that industry 1 (manufacturing) charges \$200 per unit of its output, and industry 2 (services) charges \$100 per unit of its output. Since industry 1 requires 0.4 units of output from industry 1 and 0.2 units of output from industry 2, industry 1 incurs a total cost of $(0.4)(200) + (0.2)(100)$ in order to make one unit of output. We can write this in symbols, too.

$$(0.4)(200) + (0.2)(100) = \begin{bmatrix} 0.4 & 0.2 \end{bmatrix} \begin{bmatrix} 200 \\ 100 \end{bmatrix}$$

Similarly, we could write the cost incurred to make a single unit of output from industry 2 as

$$(0.7)(200) + (0.1)(100) = \begin{bmatrix} 0.7 & 0.1 \end{bmatrix} \begin{bmatrix} 200 \\ 100 \end{bmatrix}$$

In fact, we can bundle up these costs into a single matrix-vector quantity.

$$\begin{bmatrix} 0.4 & 0.2 \\ 0.7 & 0.1 \end{bmatrix} \begin{bmatrix} 200 \\ 100 \end{bmatrix}$$

But notice that the matrix here is the matrix C "flipped" somehow. We've turned the rows of C into the columns of this new matrix. As it turns out, this is a very common operation in linear algebra, and we'll see that it leads to all sorts of interesting properties and applications.

2.3.1 The matrix transpose

To get started a little more formally, given a matrix A, the matrix transpose A^T is formed by taking the first column of A as the first row of A^T, the second column of A as the second row of A^T and so on. Here're some examples:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}, \quad A^T = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}, \quad A^T = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad A^T = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix}, \quad I^T = I$$

The transpose has all sorts of interesting properties. Let's start with some simpler ones. For instance, $(A^T)^T = A$. If A and B are matrices (or vectors) of the same size, then $(A+B)^T = A^T + B^T$. Scalar multiplication also works well with the transpose, in the sense that $(cA)^T = cA^T$ for any scalar c. But there are properties of transposition that aren't so intuitive. For instance, given suitably sized matrices A and B, we have $(AB)^T = B^T A^T$. To see why this is true, we'll need to use our definition of matrix multiplication: remember that $AB(i,j)$ is $<(i^{th} \text{ row of } A), (j^{th} \text{ column of } B)>$. Then

$$\begin{aligned} (AB)^T(i,j) &= <(j^{th} \text{ row of } A), (i^{th} \text{ column of } B)> \\ &= <(i^{th} \text{ row of } B^T), (j^{th} \text{ column of } A^T)> \\ &= B^T A^T(i,j) \end{aligned}$$

Example 11: Let

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 3 \\ 5 & 7 \\ 9 & 11 \end{bmatrix}.$$

Let's confirm that $(AB)^T(1,2) = (B^T A^T)(1,2)$. *Since rows and columns swap in transposition,* $(AB)^T(1,2) = AB(2,1)$. *So* $(AB)^T(1,2)$ *is the inner product* *of row 2 from A and column 1 from B*

$$(AB)^T(1,2) = \begin{bmatrix} 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 5 \\ 9 \end{bmatrix}$$
$$= 4 + 25 + 54 \qquad\qquad = 83.$$

For the other side, we have $(B^T A^T)(1,2)$ *is the inner product of the row 1 of* B^T *and column 2 of* A^T*. But notice that row 1 of* B^T *is the same as column 1 of* B*, and column 2 of* A^T *is the same as row 2 of* A*. So*

$$(B^T A^T)(1,2) = \begin{bmatrix} 1 & 5 & 9 \end{bmatrix} \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$$
$$= 4 + 25 + 54$$
$$83.$$

We could repeat this same procedure to prove that $(AB)^T$ *and* $B^T A^T$ *are componentwise equal.*

Using the previous two properties, we can actually determine the exact form of the inverse of A^T provided that it exists. To see how this works, let's first assume that a matrix A has an inverse A^{-1}. Then maybe there's an inverse of A^T, too. If A^T has an inverse B then

$$\begin{aligned} BA^T &= I \\ (BA^T)^T &= I^T \\ (A^T)^T B^T &= I \\ AB^T &= I \\ B^T &= A^{-1} I \\ B &= (A^{-1})^T \end{aligned}$$

In other words, the inverse of the transpose is the transpose of the inverse of A. In symbols, $(A^T)^{-1} = (A^{-1})^T$.

2.3.2 Computing solutions

Now, cost is only part of any economic equation. What we're really interested in is what price each industry needs to charge in order to cover depreciation, the wages of its employees, *etc.* and also make a fixed amount of profit. We'll be measuring all of these quantities per unit of output. (This is important, because it allows us to make fair comparisons.) All together, the wages, depreciation, profit, *etc.* can be summed into a single number: the *added value* v_i of industry

i. Bundling up the added values of every industry, we can form an added value vector $\mathbf{v}$. Then given a consumption matrix C and a value added vector $\mathbf{v}$, we would like to find at price vector $\mathbf{p}$ such that

$$\mathbf{p} = C^T\mathbf{p} + \mathbf{v}.$$

How does this equation mean? Well, here the prices of 1 unit of output from each industry have been set so that every industry exactly covers both its costs incurred from buying other industries' goods, represented by $C^T\mathbf{p}$, and its added value. But how can we find such a special price vector? We can follow a similar path as we did in the original Leontief model.

$$(I - C^T)\mathbf{p} = \mathbf{v}.$$

If the matrix $(I - C^T)$ is invertible, then

$$\mathbf{p} = (I - C^T)^{-1}\mathbf{v}.$$

Let's revisit our two industry example, but this time, rather than using a fixed price vector, let's imagine having a fixed value added vector $\mathbf{v} = [50, 25]^T$.

Reading Check 32: Describe in words what $\mathbf{v}$ *means in terms of quantities from our two industry model.*

Then our price equation takes the form

$$\begin{aligned} \mathbf{p} &= C^T\mathbf{p} + \mathbf{v} \\ \mathbf{p} &= \begin{bmatrix} 0.4 & 0.2 \\ 0.7 & 0.1 \end{bmatrix}\mathbf{p} + \begin{bmatrix} 50 \\ 25 \end{bmatrix} \\ \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.4 & 0.2 \\ 0.7 & 0.1 \end{bmatrix}\right)\mathbf{p} &= \begin{bmatrix} 50 \\ 25 \end{bmatrix} \end{aligned}$$

Just as a refresher, let's remind ourselves how to solve this system of linear equations the old way, that is, the way we solved systems of linear equations before we knew about the concept of inversion. Short story: `rref` the augmented matrix.

```
EDU>> rref([.6 -.2 50; -.7 .9 25])

ans =

     1     0   125
     0     1   125
```

But if we had a bigger matrix, this might be a little bit of a pain. If we're sure that $I - C^T$ has an inverse, we can compute the solution price vector $\mathbf{p}$ using matrix inversion. But before we do, we need to learn how to take the transpose of a matrix in Matlab. Fortunately for everyone involved, this is pretty easy

```
EDU>> C = [0.4 0.7; 0.2 0.1]

C =

    0.4000    0.7000
    0.2000    0.1000

EDU>> C'

ans =

    0.4000    0.2000
    0.7000    0.1000
```

The symbol to do the transpose, in case it's unclear, is the single quote. You can also type `help transpose` in Matlab to learn more about the syntax.

Now we're ready to actually do our matrix inversion. Remember that there are two ways to do this in Matlab. The first way looks a lot like what we would write analytically:

```
EDU>> inv(eye(2) - C') * [50;25]

ans =

   125
   125
```

The second looks a little bit different, and more like division notation than inverse notation.

```
EDU>> (eye(2) - C') \ [50;25]

ans =

  125.0000
  125.0000
```

Regardless of how you get to this point, we've concluded that for the manufacturing and service industries to have added values of \$50 and \$25 per unit of output, respectively, then both should charge \$125 dollars per unit of output.

2.3.3 Consequences of linearity to sensitivity

Using the same example we've been working with, let's imagine that the value added per unit in industry 1 is $v_1 = 17$ and the value added per unit in industry 2 is $v_2 = 24$. What are the appropriate prices given all of our assumptions? Matlab can crunch this type of thing, no problem:

```
EDU>> C = [0.4 0.7; 0.2 0.1]

C =

    0.4000    0.7000
    0.2000    0.1000

EDU>> (eye(2) - C') \ [17;24]

ans =

   50.2500
   65.7500
```

So for all industries to be simultaneously satisfied, industry 1 should sell 1 unit of its output for \$50.25 and industry 2 should sell 1 unit of its output for \$65.75. (Side note: If you're more comfortable simplifying the matrix expression before you put it into Matlab, that's totally fine. Myself, I'm pretty bad at arithmetic, so when there's a chance, I let Matlab do it for me.)

Now, what happens to the price solution if the value added per unit of industry 1 changes slightly from $v_1 = 17$ to $v_1 = 18$ while the value added per unit in industry 2 remains constant at $v_2 = 24$?

```
EDU>> (eye(2) - C') \ [18;24]

ans =

   52.5000
   67.5000
```

So for all industries to be simultaneously satisfied, industry 1 should sell 1 unit of its output for \$52.25, and industry 2 should sell 1 unit of its output for \$67.50. Combining the previous results, we can write an expression for the *change* in prices when the value added of industry 1 is increased by 1.

$$\begin{aligned}\Delta\mathbf{p}_1 &= \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 18 \\ 24 \end{bmatrix} - \begin{bmatrix} 0.6 & -0.7 \\ -0.2 & 0.9 \end{bmatrix}^{-1} \begin{bmatrix} 17 \\ 24 \end{bmatrix} \\ &= \begin{bmatrix} 52.50 \\ 67.50 \end{bmatrix} - \begin{bmatrix} 50.25 \\ 65.75 \end{bmatrix} \\ &= \begin{bmatrix} 2.25 \\ 1.75 \end{bmatrix}\end{aligned}$$

We can think of this vector as the sensitivity of the prices of each industry to changes in the value added per unit of industry 1. The larger the absolute values of the entries of this vector $\Delta\mathbf{p}_1$, the bigger the changes in price when industry 1 changes its value added just a little bit.

The really surprising thing is the this price change information is encoded in the matrix $(I - C^T)^{-1}$! To see this, let's start with the identical first line above and follow a different mathematical course:

$$\begin{aligned}\Delta\mathbf{p}_1 &= \begin{bmatrix}0.6 & -0.7\\ -0.2 & 0.9\end{bmatrix}^{-1}\begin{bmatrix}18\\24\end{bmatrix} - \begin{bmatrix}0.6 & -0.7\\ -0.2 & 0.9\end{bmatrix}^{-1}\begin{bmatrix}17\\24\end{bmatrix}\\ &= \begin{bmatrix}0.6 & -0.7\\ -0.2 & 0.9\end{bmatrix}^{-1}\left(\begin{bmatrix}18\\24\end{bmatrix} - \begin{bmatrix}17\\24\end{bmatrix}\right)\\ &= \begin{bmatrix}0.6 & -0.7\\ -0.2 & 0.9\end{bmatrix}^{-1}\begin{bmatrix}1\\0\end{bmatrix}.\end{aligned}$$

But for *any* matrix A, the matrix multiplication $A[1, 0, \ldots, 0]^T$ returns the first column of A. And so $\Delta\mathbf{p}_1$ is the first column of the matrix $(I - C^T)^{-1}$.

Simply computing the inverse of $I - C^T$ shows us that our analysis is correct.

```
EDU>> inv(eye(2) - C')

ans =

    2.2500    0.5000
    1.7500    1.5000
```

This is a good place to note that only the *difference* of the value added vectors mattered in the preceding computations; their absolute levels make absolutely no impact.

Reading Check 33: Repeat the preceding steps using the value added vectors $\mathbf{v} = [1, 2000]^T$ *and* $\mathbf{v}' = [1, 2001]^T$ *in order to show that the second column of* $(I - C^T)^{-1}$ *represents the sensitivity of prices to changes in the value added per unit of industry 2.*

2.3.4 Convenient properties

If we are concerned about both the production and the prices in a Leontief model, we can use one of our transpose properties to make this solving even simpler. Notice that

$$\begin{aligned}(I - C^T)^{-1} &= (I^T - C^T)^{-1}\\ &= ((I - C)^T)^{-1}\\ &= ((I - C)^{-1})^T\end{aligned}$$

Let's see an example.

```
EDU>> C
```

```
C =

    0.2000    0.5000    0.1000    0.3000
    0.4000    0.1000    0.4000    0.4000
    0.1000    0.1000    0.1000    0.1000
    0.2000    0.2000    0.3000         0

EDU>> inv((eye(4) - C'))

ans =

    3.5516    2.6187    0.8511    1.4894
    2.7823    3.3879    0.8511    1.4894
    2.4386    2.5968    1.8440    1.5603
    2.4223    2.4004    0.7801    2.1986

EDU>> inv((eye(4) - C))

ans =

    3.5516    2.7823    2.4386    2.4223
    2.6187    3.3879    2.5968    2.4004
    0.8511    0.8511    1.8440    0.7801
    1.4894    1.4894    1.5603    2.1986
```

This means that if we have computed $(I-C)^{-1}$ to solve $\mathbf{x} = C\mathbf{x}+\mathbf{d}$ then we can easily compute $(I - C^T)^{-1}$ (by simply taking the transpose of $(I - C)^{-1}$) in order to solve $\mathbf{p} = C^T\mathbf{p} + \mathbf{v}$. Moreover, if since the sensitivity of prices to changes in value added is related to the columns of $(I - C^T)^{-1}$), we could also determine the sensitivity of prices to changes in value added by looking the rows of $(I - C)^{-1}$.

For instance, given

$$C = \begin{bmatrix} 0.2 & 0.5 & 0.1 & 0.3 \\ 0.4 & 0.1 & 0.4 & 0.4 \\ 0.1 & 0.1 & 0.1 & 0.1 \\ 0.2 & 0.2 & 0.3 & 0 \end{bmatrix}.$$

What is the sensitivity of the price of product 3 to change in value added of product 2 of 1 dollar per unit? Well, from earlier work, we know that the sensitivity of all prices to changes in value added of product 2 is given by column 2 of $(I - C^T)^{-1}$.

```
EDU>> inv((eye(4) - C'))
```

```
ans =

    3.5516    2.6187    0.8511    1.4894
    2.7823    3.3879    0.8511    1.4894
    2.4386    2.5968    1.8440    1.5603
    2.4223    2.4004    0.7801    2.1986
```

So the price per unit of product 3 will change by 2.5968 dollars per unit if the price of product 2 increase by 1 dollar per unit. But since $(I - C^T)^{-1} = ((I - C)^{-1})^T$, we could also look at entry (2,3) of $(I - C)^{-1}$ and find the same information.

```
EDU>> inv((eye(4) - C))

ans =

    3.5516    2.7823    2.4386    2.4223
    2.6187    3.3879    2.5968    2.4004
    0.8511    0.8511    1.8440    0.7801
    1.4894    1.4894    1.5603    2.1986
```

The point here is that you can save a lot of time if you know some shortcuts that are provided by the properties of matrices. You could arrive at the same answer a lot of different ways; some are just much easier than others.

Chapter 3

Eigenpairs and dynamical systems

3.1 Dynamical systems

Imagine you're trying to plan out new stations for your bike rental startup. Your business model is interesting in that unlike many car rental services, customers don't have to return the bike where they picked it up. One of the headaches that you regularly deal with is bikes stacking up in popular locations. It's a thin line to walk: you don't have to have customers unable to return their bike because the station is full, but you don't want to invest in large stations that are underutilized. We'll see that you can use linear algebra to determine the long-term distribution of bikes across all your locations given some moderate constraints.

3.1.1 Transition matrices

To get a handle on the situation, let's first deal with a case that there are just two locations, Location 1 and Location 2. Suppose that we've collected data that indicate that 80% of the bikes rented at Location 1 are returned to Location 1, and the remaining 20% are returned to Location 2. We have similar data dealing with Location 2: 60% of the bikes rented at Location 2 are returned to Location 2, and the remaining 40% are returned to Location 1. We call a situation such as this one in which multiple interdependent quantities change together over time a *dynamical system*.

We'll assume for simplicity that every bike is rented and returned every day. Let's define x_0 and y_0 to be the initial number of bikes at Location 1 and Location 2, respectively. Then the number of bikes housed at Location 1 on day 1 is

$$x_1 = 0.8x_0 + 0.4y_0,$$

and the number of bikes housed a Location 2 on day 1 is

$$y_1 = 0.2x_0 + 0.6y_0.$$

We can gather these equations up into a matrix equation.

$$\begin{aligned} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} &= \begin{bmatrix} 0.8 & 0.4 \\ 0.2 & 0.6 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \\ \mathbf{b}_1 &= T\mathbf{b}_0. \end{aligned}$$

We call T the *transition matrix* of the system. Notice that T serves a really useful purpose: it updates (or *transitions*) the state of the bike distribution at time $t = 0$ to the bike distribution at time $t = 1$. If we assume that the same rental-return percentages hold on day $t = 1$, then the bike distribution on day $t = 2$ will be

$$\begin{aligned} \mathbf{b}_2 &= T\mathbf{b}_1 \\ &= T(T\mathbf{b}_0) \\ &= T^2\mathbf{b}_0. \end{aligned}$$

So if T moves the state of our business one day into the future, T^2 moves the state two days into the future. The trace of our variables over time is called the *trajectory* of the dynamical system. Note that the trajectory depends on $\mathbf{b}_0$.

To make things more concrete, let's see an example. Imagine that the initial distribution of 300 total bikes is even between the two locations so that $x_0 = y_0 = 150$. Using our formation from above, we see that

```
EDU>> T * [150; 150]

ans =

   180
   120
```

so that there are 180 bikes at Location 1 and 120 bikes at Location 2. What about after 2 days of activity?

```
EDU>> T * [180; 120]

ans =

   192
   108
```

We also could have computed the same answer using T^2 and $\mathbf{b}_0 = [150, 150]^T$.

```
EDU>> T^2 * [150; 150]

ans =

  192.0000
  108.0000
```

We still haven't determined what's happening in the long term. We could always calculate $T^t\mathbf{b}_0$ for large t. For instance,

```
EDU>> T^100*[150;150]

ans =

  200.0000
  100.0000
```

But this doesn't give us an idea of how the distribution changes over time. Notice that this is *really* close to the distribution we saw at $t = 2$. To trace the bike distribution over time, we'll need another piece of Matlab machinery: the `for` loop.

3.1.2 The for loop

Imagine we wanted to successively calculate the distribution of bikes at each of the locations over 4 consecutive days. We've already decided that the distribution of bikes of day t is $T^t\mathbf{b}$. We could type in 4 different commands, but this wouldn't scale well if we wanted to compute the distribution at each of the first 100 or 1000 days. We can use a Matlab command to simplify the process. (You can copy and paste this code directly into the command line, or you can open a new Matlab script by going to `File > New > Blank M-File` and then pasting; to run this piece of code in the M-file, press `F5`.)

```
b = [150; 150];
for t = 1:4
    T^t * b
end
```

Before we examine the output, let's walk our way through to the code. We initialize the variable `b` to our initial distribution of bikes. The next line, which we read in words as "for `t` from 1 to 4", begins the `for` loop. Matlab executes statements found inside the loop, that is, on lines between the `for` line and the `end` line sequentially from top to bottom. When Matlab reaches the `end` statement, it increments `t` by 1, goes back to the top of the loop, and completes the whole process again. Matlab stops executing, or "exits", the loop when `t` goes outside of the range we've specified, here outside of the range 1 to 4. So the first time Matlab passes through the loop, Matlab executes `T^1 * b`, the second time it executes `T^2 * b`, the third time `T^3 * b` and so on. (Notice that we could easily change the time span over which we calculate the bike distribution by simply increasing the 4 to a 10 or 100 or whatever we'd like; the loop would work exactly the same.)

When we run this piece of code, we see

```
ans =

   180
   120


ans =

  192.0000
  108.0000


ans =

  196.8000
  103.2000
```

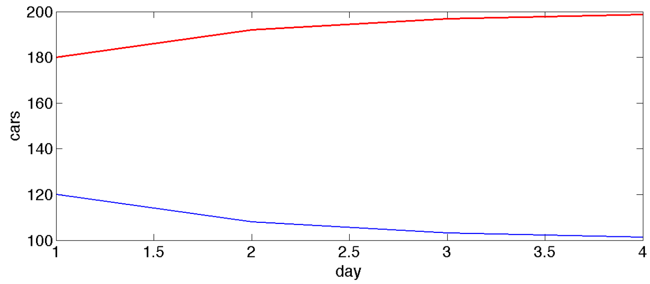


Figure 3.1:

```
ans =

  198.7200
  101.2800
```

Notice that since we did not put a semicolon at the end of the line reading `T^t*b`, Matlab will print the result of each of these computations. But even this printout doesn't really help us *see* what's going on here. For this, we can use the `plot` command.

3.1.3 Plotting

Let's use the our base `for` loop with just a few more commands.

```
b = [150; 150];
B = zeros(2,4);
time = 1:4;
for t = 1:4
    B(:,t) = T^t * b;
end
plot(time,B(1,:),'r'); hold on
plot(time,B(2,:),'b')
xlabel('day')
ylabel('cars')
```

This may a look more complicated, but most of the additional commands are just fluff to make the figure look decent. In the meat of the code, we begin by initializing `B` to be a 2×4 matrix full of zeros. This is where we'll keep the data on the distribution of bikes. Each column will represent a specific day, and each row will represent a specific location. We calculate `T^t*b` just as we did in the previous example, but here we assign the vector we've computed to be column t of matrix B. The colon symbol "all components" so that `B(:,t)` means "all rows, column `t`". Similarly, in the `plot` commands, `B(1,:)` means "row 1, all columns" and `B(2,:)` means "row 2, all columns". The full command `plot(time,B(1,:),'r')` plots the first row vector of `B` as y values against the vector `time` as the x values. The optional argument `'r'` just makes the line red. Remember, if you're unsure what a command does or how to use it properly, type `help` and then the command's name in Matlab. When we run this code, we get Figure 3.1.

3.1.4 Steady state vectors

From the figure, it seems like the distribution of bikes is moving towards a steady state in which there are 200 bikes at Location 1 and 100 bikes at Location 2. Mathematically, a steady state vector $\mathbf{v}$ for which $T\mathbf{v} = \mathbf{v}$. In words, the transition matrix which moves us from time t to time $t+1$ does not affect the steady state vector; the state of the system is the same at time t as it is at time $t+1$. We can verify that $[200, 100]^T$ is in fact a steady state using Matlab.

```
>> [0.8 0.4; 0.2 0.6] * [200; 100]

ans =

   200
   100
```

We'll see in the next section that the idea of a steady state vector is in fact a special case of the more general equation $T\mathbf{v} = \lambda\mathbf{v}$. In this more inclusive case, the matrix T acts to scale $\mathbf{v}$ by a factor of λ. In the case of a steady state vector $\lambda = 1$.

3.2 Eigenvectors and eigenvalues

We saw in the last section that steady state vectors at which $T\mathbf{v} = \mathbf{v}$ can play an important role in dynamical systems. We can learn even more about the long term behavior of dynamical systems by expanding this concept slightly. Given a matrix T, an *eigenvector* $\mathbf{v}$ associated with *eigenvalue* λ satisfies $T\mathbf{v} = \lambda\mathbf{v}$. Conceptually, the matrix T acts to scale $\mathbf{v}$ by λ. In the case of a steady state vector, $\lambda = 1$, but there are many other and interesting cases to consider. *Example 12:*

$$A = \begin{bmatrix} 2 & 0 \\ 0 & -3 \end{bmatrix} \text{ with } (\mathbf{v}_1, \lambda_1) = \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, 2 \right) \text{ and } (\mathbf{v}_2, \lambda_2) = \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, -3 \right)$$

Example 13:

$$B = \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} \text{ with } (\mathbf{v}_1, \lambda_1) = \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, 3 \right) \text{ and } (\mathbf{v}_2, \lambda_2) = \left(\begin{bmatrix} 1 \\ -1/2 \end{bmatrix}, 0 \right)$$

Example 14:

$$C = \begin{bmatrix} 13 & -4 \\ 4 & 7 \end{bmatrix} \text{ with } (\mathbf{v}_1, \lambda_1) = \left(\begin{bmatrix} -2 \\ 1 \end{bmatrix}, 15 \right) \text{ and } (\mathbf{v}_2, \lambda_2) = \left(\begin{bmatrix} 1 \\ 2 \end{bmatrix}, 5 \right)$$

3.2.1 Computing eigenthings

A lot of time and effort has gone into developing algorithms to find eigenvectors and eigenvalues efficiently, at least in a numeric sense. In Matlab, we write `[V,D] = eig(A)`. This command returns two square matrices V and D such that

$$V = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{bmatrix}$$
$$D = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix}$$

and $A\mathbf{v}_i = \lambda_i \mathbf{v}_i$ for every eigenpair. For instance, we can use Matlab to compute the eigenthings from the last example above.

```
>> [V,D] = eig([13 -4; -4 7])

V =

   -0.4472   -0.8944
   -0.8944    0.4472


D =

     5     0
     0    15
```

But why don't the columns of V look the the eigenvectors $\mathbf{v}_1 = [-2, 1]^T$ and $\mathbf{v}_2 = [1, 2]^T$ we found earlier? Well, if $(\mathbf{v}, \lambda)$ is an eigenpair of A, then so is

$(c\mathbf{v}, \lambda)$ for any constant c. To see this, we need only verify that $A(c\mathbf{v}) = \lambda(c\mathbf{v})$ is true. So Matlab is giving us just *one* scaling of each of the eigenvectors. Notice that the first eigenvector that Matlab has given us is just $-0.4472\mathbf{v}_2$. Similarly, the second eigenvector Matlab has generated is $0.4472\mathbf{v}_1$. (Matlab in fact is giving us the unique scaled version of each eigenvector that has length 1. We'll talk about this morning when we talk about orthogonality and norms in a later chapter.)

3.2.2 Applications to dynamical systems

So why should we care about eigenthings? Imagine that we have a 2×2 transition matrix T with eigenpairs $(\mathbf{v}_1, \lambda_1)$ and $(\mathbf{v}_2, \lambda_2)$. Let's assume that we can write some initial condition $\mathbf{x}_0$ as a linear combination of the eigenvectors so that $\mathbf{x}_0 = c_1\mathbf{v}_1 + c_2\mathbf{v}_2$. Then

$$\begin{aligned}
\mathbf{x}_1 &= T\mathbf{x}_0 \\
&= T(c_1\mathbf{v}_1 + c_2\mathbf{v}_2) \\
&= c_1 T\mathbf{v}_1 + c_2 T\mathbf{v}_2 \\
&= c_1\lambda_1\mathbf{v}_1 + c_2\lambda_2\mathbf{v}_2.
\end{aligned}$$

So T has scaled the $\mathbf{v}_1$ component of $\mathbf{x}_0$ by λ_1 and the $\mathbf{v}_2$ component of $\mathbf{x}_0$ by λ_2. We can observe a similar action on the next iteration of time.

$$\begin{aligned}
\mathbf{x}_2 &= T\mathbf{x}_1 \\
&= T(c_1\lambda_1\mathbf{v}_1 + c_2\lambda_2\mathbf{v}_2) \\
&= c_1\lambda_1 T\mathbf{v}_1 + c_2\lambda_2 T\mathbf{v}_2 \\
&= c_1\lambda_1^2\mathbf{v}_1 + c_2\lambda_2^2\mathbf{v}_2.
\end{aligned}$$

So we have accumulated a factor of λ_1 in front of the first eigenvector and a factor of λ_2 in front of the second eigenvector. If we were to repeat this for n iterations, we would arrive at

$$\mathbf{x}_2 = c_1\lambda_1^n\mathbf{v}_1 + c_2\lambda_2^n\mathbf{v}_2.$$

If $|\lambda_i| > 1$, then the contribution of $\mathbf{v}_i$ grows exponentially over time. If $|\lambda_i| < 1$, then the contribution of $\mathbf{v}_i$ shrinks exponentially over time. And if $|\lambda_i| = 1$, then the contribution of $\mathbf{v}_i$ remains constant over time.

This has some profound implications for the behavior of dynamical systems. We call the eigenvalue of a matrix T with largest absolute value the *dominant eigenvalue* of T. Since the dominate eigenvalue has the largest absolute value, the contribution of its associated eigenvector to the state of the system grows faster than any other eigenvector's contribution. Soon, the state of the system looks very much like a scaled version of the dominate eigenvector. The dominate eigenvector does in fact dominate the long term dynamics of the system. Mathematically, if we have eigenpairs $(\mathbf{v}_1, \lambda_1), (\mathbf{v}_2, \lambda_2), \ldots, (\mathbf{v}_n, \lambda_n)$ ordered so that $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n$. Then for large t,

$$\mathbf{x}_t \approx c_1\lambda^t\mathbf{v}_1.$$

3.2.3 An example

For a more concrete example, let's go back to our transition matrix from the previous section

$$T = \begin{bmatrix} 0.8 & 0.4 \\ 0.2 & 0.6 \end{bmatrix}.$$

Matlab tells us that the eigenpairs of the matrix are given by

```
EDU>> [V,D] = eig([0.8 0.4; 0.2 0.6])

V =

    0.8944   -0.7071
    0.4472    0.7071


D =

    1.0000         0
         0    0.4000
```

After scaling the eigenvectors, we have eigenpairs $(\mathbf{v}_1, \lambda_1) = ([2,1]^T, 1)$ and $(\mathbf{v}_2, \lambda_2) = ([-1,1]^T, 0.4)$, and the dominating eigenpair is the first one. We can write an initial condition $\mathbf{x}_0 = [150, 150]^T$ as a linear combination of the eigenvectors by solving

$$\begin{bmatrix} 2 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 150 \\ 150 \end{bmatrix}.$$

We can easily solve this system using Matlab.

```
EDU>> [2 -1; 1 1] \ [150; 150]

ans =

   100
    50
```

So $\mathbf{x}_0 = 100\mathbf{v}_1 + 50\mathbf{v}_2$. Then going from time $t = 0$ to time $t = 1$ using our transition matrix T gives

$$\begin{aligned} \mathbf{x}_1 &= T\mathbf{x}_0 \\ &= T(100\mathbf{v}_1 + 50\mathbf{v}_2) \\ &= 100T\mathbf{v}_1 + 50T\mathbf{v}_2 \\ &= 100(1)\mathbf{v}_1 + 50(0.4)\mathbf{v}_2. \end{aligned}$$

Incrementing time again, we compute $\mathbf{x}_2$.

$$\begin{aligned}\mathbf{x}_2 &= T\mathbf{x}_1 \\ &= T(100(1)\mathbf{v}_1 + 50(0.4)\mathbf{v}_2) \\ &= 100(1)T\mathbf{v}_1 + 50(0.4)T\mathbf{v}_2 \\ &= 100(1)^2\mathbf{v}_1 + 50(0.4)^2\mathbf{v}_2.\end{aligned}$$

It we continue to apply this same methodology out to the k^{th} time step, we would find

$$\mathbf{x}_k = 100(1)^k\mathbf{v}_1 + 50(0.4)^k\mathbf{v}_2.$$

Since $(0.4)^k \approx 0$ for large k, we have

$$\mathbf{x}_k \approx 100(1)^k\mathbf{v}_1 = 100\begin{bmatrix}2\\1\end{bmatrix} = \begin{bmatrix}200\\100\end{bmatrix}$$

for large k. Notice that since the growing or shrinking of each component is exponential, we actually don't have to look very far into the future to see good convergence to the dominant eigenvector.

The conclusion here is that we can learn everything about the long-term behavior of a dynamical system simply by investigating its eigenpairs, and in particular its dominating eigenpair. Now, we have swept under the rug an assumption that there is a single dominating eigenpair, but this need not always be the case; it could be that there are two eigenvectors both with the same eigenvalue. We will investigate these edge cases in the future.

3.3 The characteristic equation

In the last section, we saw that eigenpairs (and in particular *dominant* eigenpairs) can tell us a lot about the long term behavior of a dynamical system. We also saw that Matlab can compute eigenvalues and eigenvectors numerically. But at this point it's still unclear where these quantities come from. In fact, at this point it's pretty much magic. There's a lot of quantitative reasoning behind eigenpairs, at we'll delve into these ideas in this section. For an eigenpair $(\mathbf{v}, \lambda)$ to exist for a given matrix A, it must be that

$$A\mathbf{v} = \lambda\mathbf{v}.$$

Rearranging this equation can give us a whole new understanding.

$$(A - \lambda I)\mathbf{v} = \mathbf{0}.$$

So $(\mathbf{v}, \lambda)$ is an eigenpair of A if and only if $(A-\lambda I)\mathbf{v} = \mathbf{0}$ has a nontrivial solution $\mathbf{v}$. This insight allows us to connect eigenpairs to all of the material concerning the solutions of linear systems that we've developed in the past several chapters. A particularly useful way to think about the problem is to equate the existence of an eigenpair $(\mathbf{v}, \lambda)$ of a matrix A to the idea that $\det(A - \lambda I) = 0$. We can compare this formulation to the examples we saw in the previous section.

3.3.1 Computing eigenvalues

Let's consider the example matrices we saw in the last section and rather than just stating their eigenpairs out of thin air, calculate them methodically.

Define

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

We've seen that $\lambda = 2$ and $\lambda = -3$ are eigenvalues of A. But let's consider the eigenpairs of A in a different light. For $(\mathbf{v}, \lambda)$ to be an eigenpair of A, we must have

$$\begin{aligned} A\mathbf{v} &= \lambda\mathbf{v} \\ (A - \lambda I)\mathbf{v} &= 0 \\ \left(\begin{bmatrix} 2 & 0 \\ 0 & -3 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \right) \mathbf{v} &= \mathbf{0}. \end{aligned}$$

The last equation holds with $\mathbf{v} \neq \mathbf{0}$ if and only if $\det(A - \lambda I) = 0$. We can easily compute this determinant.

$$\begin{aligned} \det(A - \lambda I) &= 0 \\ \det \left(\begin{bmatrix} 2 & 0 \\ 0 & -3 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \right) &= 0 \\ \det \left(\begin{bmatrix} 2 - \lambda & 0 \\ 0 & -3 - \lambda \end{bmatrix} \right) &= 0 \\ (2 - \lambda)(-3 - \lambda) &= 0. \end{aligned}$$

So $\det(A - \lambda I) = 0$, and hence an eigenpair $(\mathbf{v}, \lambda)$ exists, if and only if $\lambda = 2$ or $\lambda = -3$. So the eigenvalue of A are $\lambda = 2$ and $\lambda = -3$, just as we found earlier. But note that under this formulation, we know that these are the *only* eigenvalues of A.

For another example, define

$$B = \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}.$$

Then $(\mathbf{v}, \lambda)$ is an eigenpair of B if and only if $B\mathbf{v} = \lambda\mathbf{v}$. Writing this out in more detail, we see

$$\begin{aligned} (B - \lambda I)\mathbf{v} &= \mathbf{0} \\ \left(\begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right)\mathbf{v} &= \mathbf{0} \\ \left(\begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right)\mathbf{v} &= \mathbf{0} \\ \begin{bmatrix} 1-\lambda & 2 \\ 1 & 2-\lambda \end{bmatrix}\mathbf{v} &= \mathbf{0}. \end{aligned}$$

Again, this last equation holds with nonzero $\mathbf{v}$ if and only if $\det(B - \lambda I) = 0$. The determinant is given by

$$\begin{aligned} \det(B - \lambda I) &= (1-\lambda)(2-\lambda) - 2 \\ &= \lambda^2 - 3\lambda \\ &= \lambda(\lambda - 3). \end{aligned}$$

So an eigenpair $(\mathbf{v}, \lambda)$ exists if and only if $\lambda = 0$ or $\lambda = 3$.

For a final example, define

$$C = \begin{bmatrix} 13 & -4 \\ -4 & 7 \end{bmatrix}.$$

Then setting $\det(C - \lambda I) = 0$ gives

$$\begin{aligned} \det\left(\begin{bmatrix} 13-\lambda & -4 \\ -4 & 7-\lambda \end{bmatrix}\right) &= 0 \\ 91 - 20\lambda + \lambda^2 - 16 &= 0 \\ \lambda^2 - 20\lambda + 75 &= 0 \\ (\lambda - 15)(\lambda - 5) &= 0. \end{aligned}$$

So the eigenvalues of C are $\lambda = 15$ and $\lambda = 5$.

While we've only completed examples with 2×2 matrices here, this is a very general process. We call $\det(A - \lambda I)$ the *characteristic equation of* A. The characteristic equation of a matrix A will always be a polynomial in λ, and the roots of this polynomial are the eigenvalues of A.

But this brings us to an important question: if Matlab can just compute the eigenpairs of A, why should we care about the characteristic equation? Well, Matlab is very good at computing things numerically, but what if one of the entries A is not a number, but a variable? For an example, consider the predator-prey model of owls and rats with transition matrix

$$T = \begin{bmatrix} 0.5 & 0.4 \\ -p & 1.1 \end{bmatrix}.$$

We saw in Studio 8 that the parameter p is the predation rate which measures the average number of rats (in thousands) eaten by one owl. We also saw in this studio problem that for some values of p both species go extinct, while for other values of p both grow in the long term. A natural question is what value of p separates the two scenarios. Let λ be the dominant eigenvalue of T. If $|\lambda| > 1$, then the species grow together, and if $|\lambda| < 1$, then the populations go both extinct. So the dividing case is $|\lambda| = 1$. Using the characteristic equation, we can solve for the eigenvalues, even with the uncertainty generated by the parameter p.

$$\begin{aligned}\det(T - \lambda I) &= \det\left(\begin{bmatrix} 0.5 & 0.4 \\ -p & 1.1 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}\right) \\ &= \det\left(\begin{bmatrix} 0.5-\lambda & 0.4 \\ -p & 1.1-\lambda \end{bmatrix}\right). \qquad = (0.5-\lambda)(1.1-\lambda) + 0.4p \\ &= \lambda^2 - 1.6\lambda + 0.4p + 0.55.\end{aligned}$$

Setting $\det(T - \lambda I)$ equal to 0 and solving for λ produces (for instance, with WolframAlpha or any other symbolic calculator)

$$\begin{aligned}\lambda_1 &= \frac{1}{10}(8 + \sqrt{9 - 40p}) \\ \lambda_1 &= \frac{1}{10}(8 - \sqrt{9 - 40p}).\end{aligned}$$

Recalling that the separating case is $|\lambda| = 1$ for the dominant eigenvalue λ, we can solve for the critical predation rate that separates the populations thriving together from the populations going extinct.

$$\begin{aligned}1 &= \frac{1}{10}(8 + \sqrt{9 - 40p}) \\ p &= \frac{1}{8} = 0.125.\end{aligned}$$

So the populations go extinct $p > 0.125$ and grow together towards a constant ratio if $p < 0.125$. Again, these computations aren't intended to be done by hand; use whatever symbolic calculator you're comfortable with. This result agrees well with the intuition we built in Studio 8. We saw that a predation rate of $p = 0.1$ led the populations growing together over time, and a predation rate of $p = 0.2$ led to both populations going extinct.

3.3.2 Computing eigenvectors

We've seen how to compute the eigenvalues of a matrix A using the characteristic equation, but it's still unclear how to compute the associated eigenvectors. This, too, follows a deterministic methodology. We'll consider some of the same examples.

Let's go back to our matrix

$$B = \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}.$$

We know $\lambda = 3$ is an eigenvalue, but what about a corresponding eigenvector? An eigenvector associated eigenvalue $\lambda = 3$ is a nontrivial solution to

$$\begin{aligned} B\mathbf{v} &= 3\mathbf{v} \\ (B - 3I)\mathbf{v} &= \mathbf{0} \\ \begin{bmatrix} 1-3 & 2 \\ 1 & 2-3 \end{bmatrix} \mathbf{v} &= \mathbf{0}. \end{aligned}$$

Remember that if a system is homogeneous, we can solve the system just by taking the RREF of the coefficient matrix.

$$rref\left(\begin{bmatrix} -2 & 2 \\ 1 & -1 \end{bmatrix}\right) = \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix}.$$

Writing $\mathbf{v} = [v_1, v_2]^T$, we can write this result as

$$\begin{aligned} v_1 - v_2 &= 0 \\ 0 &= 0. \end{aligned}$$

Then $v_1 = v_2$, and substituting these values back into $\mathbf{v}$ gives $\mathbf{v} = [v_2, v_2]^T = v_2[1, 1]^T$. So $[1, 1]^T$ and all its multiples are eigenvectors with eigenvalue $\lambda = 3$, just as we found in the previous section.

What about the other eigenvalue $\lambda = 0$? Here we have

$$\begin{aligned} rref\left(\begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} - 0I\right) &= rref\left(\begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}\right) \\ &= \begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix}. \end{aligned}$$

Again setting $\mathbf{v} = [v_1, v_2]^T$, the corresponding linear system reads

$$\begin{aligned} v_1 + 2v_2 &= 0 \\ 0 &= 0. \end{aligned}$$

So $\mathbf{v} = [v_1, v_2]^T = [-2v_2, v_2]^T = v_2[-2, 1]^T$. We conclude that $\mathbf{v} = [-2, 1]^T$ and all its multiples are eigenvectors associated with eigenvalue $\lambda = 0$, just as we found in the previous section.

3.3.3 Eigenvalue zero and invertibility

This example brings us to an important point: if $(\mathbf{v}, \lambda = 0)$ is an eigenpair of matrix A, then $A\mathbf{v} = 0\mathbf{v} = \mathbf{0}$ has a nontrivial solution $\mathbf{v}$. So we can connect this condition to our list of necessary and sufficient conditions on the invertibility of A. An $n \times n$ matrix A is invertible if and only if

- There exists A^{-1} such that $A^{-1}A = AA^{-1} = I_n$.
- $rref(A) = I_n$.
- A has n pivots.
- The columns of A are linearly independent.
- The columns of A span $\mathbb{R}^n$.
- $\det(A) \neq 0$.
- $A\mathbf{v} = \mathbf{0}$ has only the trivial solution $\mathbf{v} = \mathbf{0}$.
- $\lambda = 0$ is *not* an eigenvalue of A.

3.4 Complex Eigenvalues

We've seen that the roots of the characteristic polynomial $\det(A - \lambda I)$ give the eigenvalues λ of A. But in what we seen so far, we've been conveniently ignoring a simple fact: polynomials can have complex eigenvalues. In this section we'll see what complex eigenvalues mean in the context of dynamical systems.

3.4.1 A first example

Define

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

Using the tools we created in the last section, we can compute the characteristic polynomial of the matrix.

$$\begin{aligned} \det(T - \lambda I) &= \det\left(\begin{bmatrix} -\lambda & -1 \\ 1 & -\lambda \end{bmatrix}\right) \\ &= \lambda^2 + 1. \end{aligned}$$

Recalling that $i = \sqrt{-1}$, we can conclude that T has eigenvalues $\lambda = \pm i$. But what does T actually *do* to vectors? Let's consider one particular example,

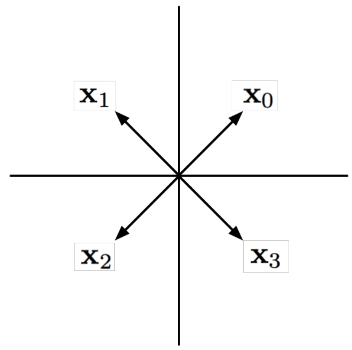


Figure 3.2: Transformation of $\mathbf{x}_0$ under successive applications of T.

$\mathbf{x}_0 = [1, 1]^T$. Then

$$\begin{aligned}\mathbf{x}_1 &= T\mathbf{x}_0 \\ &\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} -1 \\ 1 \end{bmatrix}.\end{aligned}$$

We can see this transformation of $\mathbf{x}_0$ into $\mathbf{x}_1$ graphically in figure **??**. Notice that T has rotate $\mathbf{x}_0$ 90 degrees counterclockwise.

What happens if we continue to sequentially apply T?

$$\begin{aligned}\mathbf{x}_2 &= T\mathbf{x}_1 \\ &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}. \\ \mathbf{x}_3 &= T\mathbf{x}_2 \\ &\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \\ \mathbf{x}_4 &= T\mathbf{x}_3 \\ &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \mathbf{x}_0.\end{aligned}$$

So after four applications of the matrix T, we've arrive back where started. Graphically, after applying 4 sequential rotations of 90 degrees each, we have

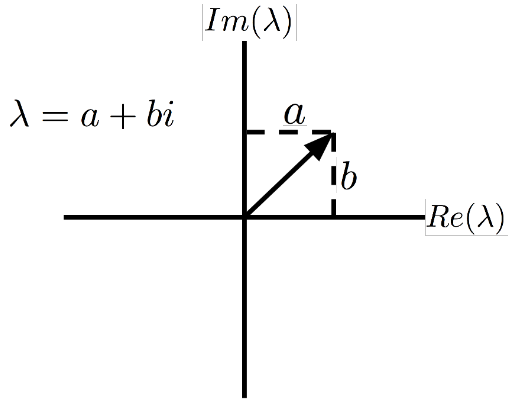


Figure 3.3: Complex eigenvalue

completed one 360 degree rotation. If we consider the matrix T as a transition matrix of a dynamical system, we arrive at a really startling conclusion: if we continue to apply T, we never end up at a steady state population distribution given the initial condition $\mathbf{x}_0$. Remember that if the eigenvalues of a transition matrix are real, we *always* arrive at some long term steady state *ratio* of the populations. But this isn't the case here! Here, the populations would continue "rotating" together forever.

Reading Check 34: Choose your own initial condition $\mathbf{x}_0 = [x_1, x_2]^T$ *and compute four sequential applications of* T. *Do you arrive back where you started?*

3.4.2 Review of complex numbers and their properties

A complex number has the form $\lambda = a + bi$, where a and b are real numbers. We call a the *real part* of λ, denoted $Re(\lambda)$, and b the *imaginary part* of λ, denoted $Im(\lambda)$. The absolute value of λ is defined slightly different than we've seen in real numbers, but the motivation is the same. Whether a real number x is positive or negative, it's absolute value $|x|$ represents its distance from the origin on the number line. We use the same ideas to motivative the definition of the absolute value of a complex number. Let $\lambda = a + bi$ and consider the graphical description of λ as seen in figure 3.3. Here we think of the real part as the "x" component and the imaginary part as the "y" component. Then the distance of the point (a, b) is just $|\lambda| = \sqrt{a^2 + b^2}$. Note that for a real number $x = x + 0i$, we have $|x| = \sqrt{x^2 + 0^2} = \sqrt{x^2} = |x|$. So the complex number definition of absolute value agrees perfectly with the definition we're used to from real numbers.

Complex numbers have a lot of interesting properties that we can exploit. But to understand many of them, we need to introduce a new concept. For a complex number $\lambda = a + bi$, we define the *complex conjugate* of λ to be $\bar{\lambda} = a - bi$. You might have first seen the complex conjugate in the context of solutions to the quadratic equation; if there are complex roots, they occur in complex conjugate pairs. The perhaps surprising thing is that eigenvalues occur in complex conjugate pairs, too. But to see why this is true, we'll have to build up some machinery first. We'll prove the following statements in a studio problem: for complex numbers z and w,

- $\overline{z + w} = \bar{z} + \bar{w}$
- $\overline{zw} = \bar{z}\bar{w}$
- $|\bar{z}| = |z|$
- $|z|^2 = \bar{z}z$

Also note that if $x = x + 0i$ is a real number, then $\bar{x} = x - 0i = x$. This also implies that if a vector $\mathbf{v}$ or matrix A is real, then $\bar{\mathbf{v}}$ and $\bar{A}$ are real, respectively.) of Using these facts, we can tackle the idea of complex conjugate eigenvalue pairs.

Let's assume that $(\mathbf{v}, \lambda)$ is an eigenpair of a real matrix A, with both $\mathbf{v}$ and λ complex. (When we say a vector (or matrix) is complex, we mean that it has components/entries that are complex numbers.) Then $(\bar{\mathbf{v}}, \bar{\lambda})$ is an eigenpair of A, too. We can simply verify that this is true. But we'll have to use many of the facts that we established above.

$$\begin{aligned} A\bar{\mathbf{v}} &= \overline{A\mathbf{v}} \\ &= \overline{\lambda\mathbf{v}} \\ &= \bar{\lambda}\bar{\mathbf{v}}. \end{aligned}$$

So $(\bar{\mathbf{v}}, \bar{\lambda})$ is an eigenpair of A, too.

Reading Check 35: For each line of the preceding series of equations, indicate which property of complex numbers allowed us to move to the next line

3.4.3 A more general example

Define

$$C = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}.$$

Note that the first example we considered in this section is a specific case of this type of matrix in which $a = 0$ and $b = 1$. We can find the eigenvalues of C

using the characteristic equation machinery we developed in the last section.

$$\begin{aligned}\det(C - \lambda I) &= \det\left(\begin{bmatrix} a-\lambda & -b \\ b & a-\lambda \end{bmatrix}\right) \\ &= (a-\lambda)(a-\lambda) - (-b)(b) \\ &= (a-\lambda)^2 + b^2.\end{aligned}$$

Then solving for the eigenvalues using the quadratic formula gives $\lambda_{1,2} = a \pm bi$. So the eigenvalues here occur in complex conjugate pairs as we saw earlier must be true.

Now consider the graphical representation of λ as seen in figure 3.3. We have $r = |\lambda| = \sqrt{a^2+b^2}$, and ϕ as the lesser included angle. We can rewrite C by factoring out r.

$$\begin{aligned}C &= r\begin{bmatrix} \frac{a}{r} & -\frac{b}{r} \\ \frac{b}{r} & \frac{a}{r} \end{bmatrix} \\ &= \begin{bmatrix} r & 0 \\ 0 & r \end{bmatrix}\begin{bmatrix} \frac{a}{r} & -\frac{b}{r} \\ \frac{b}{r} & \frac{a}{r} \end{bmatrix}.\end{aligned}$$

We can simplify the second matrix in the product by remembering that the cosine of an angle equals "adjacent over hypotenuse" and the sine of an angle equals "opposite over hypotenuse".

$$\begin{aligned}C &= \begin{bmatrix} r & 0 \\ 0 & r \end{bmatrix}\begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \\ &= S_r R_\phi.\end{aligned}$$

Let's think about what these matrices do. The matrix S_r is a *scaling matrix* because $S_r\mathbf{x}$ simply scales both components of $\mathbf{x}$ by r. The matrix R_ϕ is a *rotation matrix* because $R_\phi\mathbf{x}$ rotates $\mathbf{x}$ by ϕ radians counterclockwise. We can piece these actions together to think about how C modifies a vector $\mathbf{x}$.

$$\begin{aligned}C\mathbf{x} &= S_r R_\phi \mathbf{x} \\ &= S_r \mathbf{x}_{\text{rotated by } \phi} \\ &= \mathbf{x}_{\text{rotated by } \phi \text{ and scaled by } r}\end{aligned}$$

So complex eigenvalues induce *rotations and scalings.*

Let's see a specific example. Define

$$C = \begin{bmatrix} 3 & -4 \\ 4 & 3 \end{bmatrix}.$$

Then $r = |\lambda| = \sqrt{3^2+4^2} = 5$. Moreover $\sin\phi = 4/5$ and $\cos\phi = 3/5$ so that $\phi = \sin^{-1}(4/5) \approx 0.92$. So C represents a rotation counterclockwise by $\phi = 0.92$

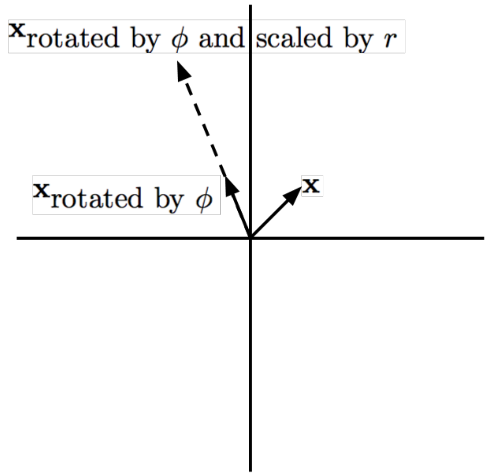


Figure 3.4: The stages of the action of the matrix C.

radians followed by a scaling of all components by $r = 5$. Let's consider how C affects a single vector $\mathbf{x} = [1, 1]^T$.

$$\begin{aligned} C\mathbf{x} &= \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} 3/5 & -4/5 \\ 4/5 & 3/5 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} -1/5 \\ 7/5 \end{bmatrix} \\ &= \begin{bmatrix} -1 \\ 7 \end{bmatrix}. \end{aligned}$$

The rotation and sequential scaling of $\mathbf{x}$ can be seen in figure 3.4.

3.4.4 Complex eigenvalues and dynamical systems

It's still unclear why we should care about complex eigenvalues, at least in any applicable sense. Let's consider one application to dynamical systems. We learned in a previous section that the eigenvalue that maximizes $|\lambda|$ dominants the long term behavior of the system. This is true even when the dominant eigenvalue is complex!

3.5 PageRank

Arguably the most innovative feature of the world wide web is its hyperlink structure. Pages link to other pages, and these links can tell us quite a bit about the "importance" of each page. It's still unclear what the term "importance" actually means at this point, as there are many ways one could define the term in the context of the web. In this section, we'll develop several different methodologies for ranking web pages in order of importance. We'll start with a basic definition and work our way up to the PageRank definition as developed by Google founders Brin and Page. To give ourselves a concrete example, we'll always consider the web featured in figure 3.5. Admittedly, this is a very simple example. But we'll see that it is more than sufficient to point out the problems with each of the methods we'll propose.

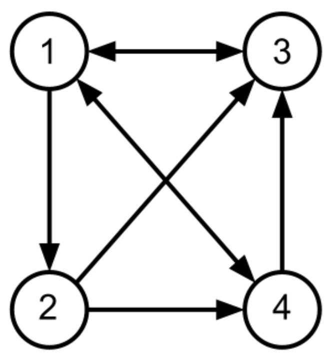


Figure 3.5: Basic web network.

In this diagram, an arrow from page i to page j represents a link on page i that points to page j. A bidirectional arrow indicates that page i links to page j and vice versa. If you happen to have some experience with discrete mathematics, you'll recognize this web as an example of a *directed graph*. Notice that we do allow cycles in this context, so that page i linking to page j linking to page k linking back to page i is admissible.

3.5.1 Method 1: count in the in-links

Probably the most straightforward way to try to measure importance is by simply counting the number of incoming links to a page and using this number as the page's rank. After all, if pages are linking to yours, then your page surely has something of value on it. The ranking vector using this metric is $\mathbf{x} = [2, 1, 3, 2]^T$. So according to Method 1, page 3 is the most important. But a ranking method as simple as this is sure to have flaws. Imagine that the owner of page 2 is upset about being ranked last under Method 1. Imagine he creates three new pages, page 5, page 6 and page 7, and makes each of them link to page 2 as seen in figure 3.6.

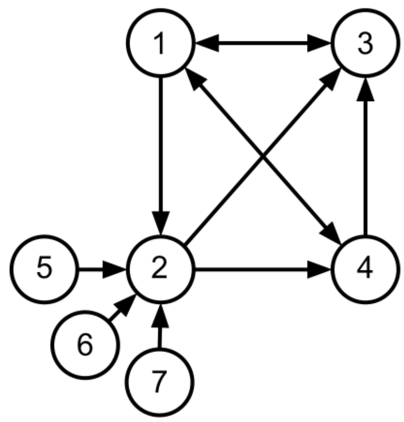


Figure 3.6: Network with malicious web pages that inflate the importance of page 2 under Method 1.

So suddenly page 2 has skyrocketed from last rank to first rank with the addition of pages 5, 6 and 7. This seems like a problem for a couple of reasons. First, someone easily cheated the proposed ranking method, and we'd like to avoid that happening. But this example is hinting at what might be even a larger problem: even if 5, 6 and 7 were legitimate pages and not created to warp the rankings, their votes shouldn't count for very much, because no one links to them! There's a big difference between getting referred by CNN.com, which a lot of people link to, or referred by my grandmother's blog about her cat, which, surprisingly, few people link to.

3.5.2 Method 2: consider rank of referrers

We can develop a more detailed and nuanced ranking methodology by considering the rank of the referrer. If a referrer has a higher rank, her link to your site should count more than a link from a page with lower rank. For one easy example of this type of thinking, let's imagine that the rank of a page is just the sum of the ranks of the pages that link to it. Applying this definition to the network seen in 3.5, we arrive at a system of linear equations in the rankings

x_i.

$$\begin{aligned} x_1 &= x_3 + x_4 \\ x_2 &= x_1 \\ x_3 &= x_1 + x_2 + x_4 \\ x_4 &= x_1 + x_2 \end{aligned} \quad \leftrightarrow \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$
$$\leftrightarrow \mathbf{x} = A\mathbf{x}.$$

If you've had some experience with graph theory, you'll recognize the matrix A as the adjacency matrix of the underlying web graph. Using the vocabulary we've developed over the previous sections, we can read off this last line by saying the ranking $\mathbf{x}$ must be an eigenvector of A with associated eigenvalue $\lambda = 1$. It might be strange that eigenpairs are showing up at this stage in the development, but this is just a testament to the utility and pervasiveness of eigenvalues in linear systems.

There are problems with this method, too. We've seen in the past several sections that eigenpairs can be delicate things. For instance, for $\lambda = 1$ be an eigenvalue of A, it must be the case that $\lambda = 1$ is a root of the characteristic equation $\det(A - \lambda I) = 0$. There's no good reason why this should be the case. Stated another way, there's no good reason why there should be a ranking $\mathbf{x}$ that fits into the method we've developed. This is fundamentally a problem of *existence*, as opposed to *uniqueness* which we'll see in Method 3, of the ranking.

Method 1 also points to a bigger flaw in our underlying mentality. Somehow, pages that link to many other pages are influencing the ranking much more than pages that link to just a few others. This might be exactly the opposite of your intuition. Pages that link to just a few others might be much for selective about what pages they recommend, and therefore their recommendation should really count. Compare this to a page that links to many, many other pages. Somehow each of these recommendations feels like it should be worth less.

3.5.3 Method 3: give each page an equal vote

We'd like the keep the idea that the value of a link should correspond somehow to the rank of the referrer, but we'd like to get rid of the fact that some pages somehow have more influence than others. Let's imagine that a page has to distribute its rank evenly among all the pages it links to. For instance, in our running example, page 1 links to three pages, page 2, page 3 and page 4. So page 1 contributes $(1/3)x_1$ to the rank of each of the three pages it links to. We can write a new system of liner equations that describes this method.

$$\begin{aligned} x_1 &= x_3 + \tfrac{1}{2}x_4 \\ x_2 &= \tfrac{1}{3}x_1 \\ x_3 &= \tfrac{1}{3}x_1 + \tfrac{1}{2}x_2 + \tfrac{1}{2}x_4 \\ x_4 &= \tfrac{1}{3}x_1 + \tfrac{1}{2}x_2 \end{aligned} \quad \leftrightarrow \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$
$$\leftrightarrow \mathbf{x} = S\mathbf{x}.$$

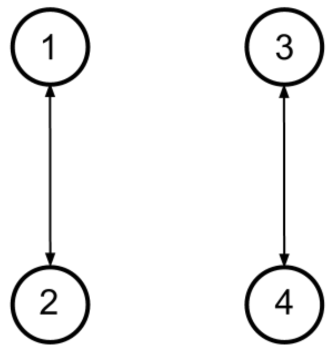


Figure 3.7: A web with disconnected groups of pages.

Notice that just as in Method 2, if a ranking $\mathbf{x}$ exists, then it must be an eigenvector of a matrix, here S, with associated eigenvalue $\lambda = 1$. But the matrix S here has more structure than the matrix A we used in Method 1. Here S is a column stochastic matrix. We know from Studio 9 that any stochastic matrix has $\lambda = 1$ as an eigenvalue. So using this method, we are sure to produce a ranking $\mathbf{x}$ that satisfies all the constraints we've set out.

The problem with this method is that there may be more than one equally valid ranking. This would cause all sorts of problems. How do you know which ranking to trust? How do you differentiate between them? But rather than try to answer these questions, let's see an example that admits multiple rankings, and then think about how we might cure the disease rather than alleviate the symptoms.

Consider the 4 page network seen in figure 3.7. A Method 3 ranking of this network satisfies

$$\begin{aligned} x_1 &= x_2 \\ x_2 &= x_1 \\ x_3 &= x_4 \\ x_4 &= x_3 \end{aligned} \quad \leftrightarrow \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

We can verify that both $\mathbf{x}_1 = [1, 1, 0, 0]^T$ and $\mathbf{x}_2 = [0, 0, 1, 1]^T$ are eigenvectors of the matrix, each having associated eigenvalue $\lambda = 1$. So we've computed two totally different rankings of the four pages, one of them ignoring pages 3 and 4, and the other ignoring pages 1 and 2. This is really getting at the heart of the issue. Somehow we actually have two *separate* networks in play here. But how do we compare completely separate networks? The pages in the groups aren't linking between the groups, so it's impossible for us to trace the influence of a page in one group to any page in the other.

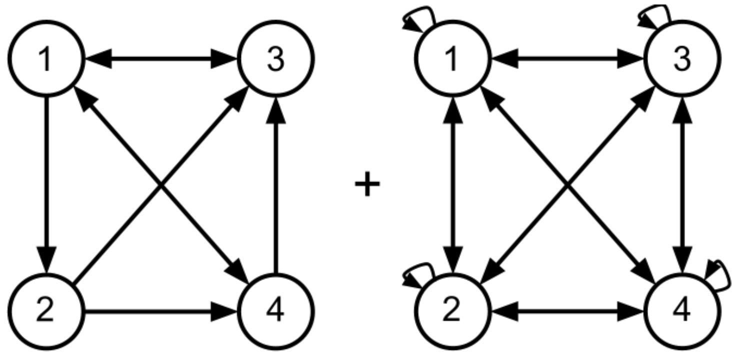


Figure 3.8: The real links (left) and the artificial links (right) present in Method 4.

3.5.4 Method 4: connect the components

To get rid of disjoint groups of web pages, we'll simply connect them all together. But the exact way in which we do this can make things really easy or really hard for us. Imagine that we artificially add links between every web page and every web page (including itself) in addition to the actual link structure of the network. Since these links aren't real in the sense that one page isn't really recommending another, we'd like the impact of these pages to be small on the overall ranking. Remember, we're just trying to get rid of the disconnected components in the page network so that (hopefully) we'll have a unique ranking in the end. Imagine we give each one of these new, fake edges coming into page i weight α/N, where α is a number between 0 and 1, and N is the total number of pages in the network, in the computation of the rank of page i. We call α the *damping factor*. We could define a new matrix $H = S + \alpha(1/N)$, but this matrix wouldn't be stochastic and have all the nice properties that come along with that. Instead, we define

$$G = (1-\alpha)S + \alpha\frac{1}{N}$$

which we call the *Google matrix* of the network. A ranking $\mathbf{x}$ according to this method satisfies $\mathbf{x} = G\mathbf{x}$.

Reading Check 36: Convince yourself that G is column stochastic and has only positive entries given that S is column stochastic $\alpha \in (0,1)$.

We still haven't resolved the question as to whether $\mathbf{x}$ is unique. To do this, we can bust out a golden oldie theorem.

Perron-Frobenius (1907): Suppose that A is a $n \times n$ matrix with strictly positive entries. Then (1) A has a unique dominant eigenpair. (2) The dominant eigenvector has all positive entries. (3) If in addition A is column stochastic, then $\lambda = 1$ is the dominant eigenvalue of A.

This may not seem like a tremendous fact, but it is. Well, it at least made Brin and Page billionaires many times over. Taken all together, we've concluded that the unique ranking we want is actually the dominant eigenvector of G. This is great news, because we know the state vector $\mathbf{x}_k = G\mathbf{x}_{k-1}$ of a dynamical system with state transition matrix G very closely resembles the dominant eigenvector when k is even just moderately large. (In fact, we saw that the convergence to the dominant eigenvector is *exponential* in k.)

Let's see this example applied to our network. We first need to initialize our Google matrix.

```
EDU>> S = [0 0 1 1/2; 1/3 0 0 0; 1/3 1/2 0 1/2; 1/3 1/2 0 0 ];
EDU>> alpha = 0.15; N = 4; % the value of alpha here is typical
EDU>> G = (1-alpha)*S + alpha*(1/N);
```

If you're feeling brave, take a look at G by removing the semicolon from the last line. It looks like a mess, despite it's relatively straightforward definition. Let's make our initial ranking $\mathbf{x}_0 = 1/4[1, 1, 1, 1]^T$, that is the uniform initial ranking

```
EDU>> x = (1/4)*ones(4,1);
```

Then the rank after successive iterations is

```
EDU>> x = G*x

x =

    0.3562
    0.1083
    0.3208
    0.2146

EDU>> x = G*x

x =

    0.4014
    0.1384
    0.2757
    0.1845

EDU>> x = G*x

x =

    0.3502
    0.1512
    0.2885
    0.2101
```

```
EDU>> x = G*x

x =

    0.3720
    0.1367
    0.2903
    0.2010
```

The rank of each page bounces around, but it does seem like they're settling down. If we compute out to 15 iterations we arrive at a steady-state vector out to 4 decimal places.

```
EDU>> x = G*x

x =

    0.3682
    0.1418
    0.2880
    0.2021
```

The PageRank is telling us that Google thinks that page 1 is the most important, followed by page 3, then page 4 and finally page 2.

Doing matrix-vector multiplication is relatively cheap, especially because the matrix S in the real world contains mostly zeros, give that of the billions and billions of web pages, each one links only to a few others. Since matrix-vector multiplication distributes, we have $G\mathbf{x} = (1-\alpha)S\mathbf{x}+(\alpha/N)\mathbf{x}$. This implies that we can do the matrix-vector multiplication in the case in which the matrix S has a lot of zeros (rather than the case with the matrix G which by construction has no zeros.) This is just one of the linear algebraic tweaks we can use to make Google's ranking algorithm even faster.

3.5.5 Methods 3 and 4, revisited: the random surfer

Seeing that the we can iterate towards the PageRank of a network by using the Google matrix G as our transition matrix might have got you thinking that there might an interesting interpretation of G in some sort of physical terms. And you'd be exactly right. But before we get to that, let's start with a slightly easier example in which we look at the related matrix S.

Imagine that we have a surfer who starts on page 1 at time $t = 0$ and uniformly randomly choses her next page from the set of pages that page 1 links to. There is probability 1/3 that she is on each of pages 2,3, and 4 at time $t = 1$. Similarly, we could imagine that our surfer started on page 4 at time $t = 0$, in which case there would be probability 1/2 that she would end up on each of pages 1 and 3 at time $t = 1$. If the surfer continues clicking links uniformly randomly at each time step, she traces out what we call a *random walk* on the network. A natural question is "where does she end up?". But since we're dealing with a random surfer, we need to think more along the lines of "with what probability is the surfer on a particular page in the long term?" The matrix S is exactly the transition matrix of this system. Now, the i^{th} component of a vector $\mathbf{x}_k$ represents the probability that the random surfer is on page i at time k. For instance, imagine we start on page 1 with probability 1, so that our initial vector is $\mathbf{x}_0 = [1, 0, 0, 0]^T$. Then after one click, the probability that the surfer is on each page is

$$\begin{aligned}\mathbf{x}_1 &= S\mathbf{x}_0 \\ &= \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{bmatrix}.\end{aligned}$$

This exactly what we found using our graphical interpretation of the problem. So the dominant eigenvalue of S (if it is unique) represents the probability distribution of the surfer's location, exactly because the matrix S describes the random walk we've laid out. (To be honest, I'm not sure if Brin and Page thought about the random surfer or the dominant eigenvector approach to this problem first. I'm severely biased and hope that they did the mathematics first and then thought of the random surfer as a convenient way to explain the concept to investors. But somehow I think this is wrong, and that they used the intuition they gained by dealing with the random surfer to continue to make progress on the mathematical side.)

The problems arising from the disconnected components that we laid out when discussing Method 3 might make a lot more sense in the context of the random surfer. If web page network has multiple disconnected groups, then if the surfer starts in one of them, she has no chance of ever getting to any of the pages in any other group.

Out solution of adding "fake" edges from every page to every page (including itself), each have weight α/N has a really interesting interpretation, too. Imagine that at each step with probability α, the random surfer doesn't choose a link on the page she's currently visiting, but instead choose a new page uniformly randomly from the entire web. (Note that this includes the possibility that she chooses the page that she's currently on.) And conversely, with probability $(1-\alpha)$ she continues on the random walk, just as we described above. Then clearly the random surfer can get to any page, even if there are disconnected groups of pages on the web. Here, the state transition matrix of the dynamical system is $G = (1-\alpha)S + \alpha/N$. Let's take another look at this equation in the new light. The first term in the sum represents the "continue to surf" behavior, and the second term represents the "stop surfing and pick a new starting point" behavior.

Chapter 4

Orthogonal projections and least-squares regression

Over the next few weeks, we'll work our way towards an incredibly useful set of technologies for dealing with real data: regression. You've probably seen at this point a few specific flavors of regression: linear, multilinear, polynomial, exponential. In this chapter, we'll develop the theory necessary to deal with *any* type of regression that is linear in the unknown coefficients. Along the way we'll pick some items that are interesting and useful in their own right, including the ideas of the fundamental subspaces of a matrix and orthogonality.

4.1 Vector spaces and subspaces

A vector space, informally, is a collation of items called *vectors* that are closed under addition and scalar multiplication. Said another way, if $\mathbf{v}$ and $\mathbf{w}$ are two arbitrary vectors in our vector space, then so are $\mathbf{v}+\mathbf{w}$ and $c\mathbf{v}$ for any scalar c. We'll see that while we've been thinking about vectors as rows or columns of real numbers, we can expand the definition to a more abstract setting to deal with all sorts of new, useful, and interesting cases. But before we do, let's see a more formal definition of the vector space. A collection of vectors V is a *vector* *space* if for every $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ chosen from V, and scalars c and d, we have

$$\mathbf{u}+\mathbf{v}=\mathbf{v}+\mathbf{u} \quad \text{(commutativity)}$$
$$(\mathbf{u}+\mathbf{v})+\mathbf{w}=\mathbf{u}+(\mathbf{v}+\mathbf{w}) \quad \text{(associativity)}$$
$$\mathbf{u}+\mathbf{0}=\mathbf{u} \quad \text{(identity)}$$
$$\mathbf{u}+(-\mathbf{u})=\mathbf{0} \quad \text{(inverse)}$$
$$c(\mathbf{u}+\mathbf{v})=c\mathbf{u}+c\mathbf{v} \quad \text{(distributivity)}$$
$$(c+d)\mathbf{u}=c\mathbf{u}+d\mathbf{u} \quad \text{(distributivity)}$$
$$(cd)\mathbf{u}=c(d\mathbf{u}) \quad \text{(associativity)}$$
$$1\mathbf{u}=\mathbf{u} \quad \text{(identity).}$$

Note that all of this is a particularly long-winded way of saying that both addition and scalar multiplication are "well behaved" in the sense that they act the way we have come to expect numbers (and vectors) to act.

We've been dealing with vectors spaces for a long while at this point. The collection $\mathbb{R}^n$, vectors with n components, each containing a real number, is probably the best known and most used vector space. So why are we going back to such a basic definition now? There are two main reasons. First, we'll see that the idea of vector spaces extends way past $\mathbb{R}^n$. Making this connection will allow us to use all the tools we've developed to work on $\mathbb{R}^n$ is a bunch of new contexts that seem like they might have been difficult to understand and handle without all these ready-made linear algebraic machinery. Second, we can start thinking about how vector spaces are related to one another. One particularly natural and handy idea is that of the vector subspace.

Informally, a vector space H of a vector space V is itself a vector space that is entirely contained within V. Given a collection of vectors H that is contained entirely within a vector space V, we can use what we'll call the *two step subspace test* to determine if H is a subspace of V. We define the test as follows. Let $\mathbf{v}$ and $\mathbf{w}$ be arbitrary vectors in H.

- Verify that $\mathbf{v}+\mathbf{w}$ is in H.

- Verify that $c\mathbf{v}$ is in H.

If both of these conditions are true for every choice of vectors $\mathbf{v}, \mathbf{w}$ and scalar c, then H is indeed a subspace of V.

For an example, let's consider $H = span([1,2]^T)$ in $\mathbb{R}^2$. (Recall that H is the collection of vectors that look like $c[1,2]^T$ for any scalar c.) Let's apply the two step subspace test to see whether H is a subspace of $\mathbb{R}^2$. Define $\mathbf{v} = [k, 2k]^T$

and $\mathbf{w} = [\ell, 2\ell]^T$. We can verify that $\mathbf{v} + \mathbf{w}$ is in H, too.

$$\begin{aligned}\mathbf{v} + \mathbf{w} &= \begin{bmatrix} k \\ 2k \end{bmatrix} + \begin{bmatrix} \ell \\ 2\ell \end{bmatrix} \\ &= k \begin{bmatrix} 1 \\ 2 \end{bmatrix} + \ell \begin{bmatrix} 1 \\ 2 \end{bmatrix} \\ &= (k + \ell) \begin{bmatrix} 1 \\ 2 \end{bmatrix} \in H.\end{aligned}$$

For the second step in the subspace test, we just note that $c[1, 2]^T \in H$ for any scalar c just by the definition of span. So H is indeed a subspace of $\mathbb{R}^2$. We can a graphical representation of this idea of subspaces in Figure 4.1.

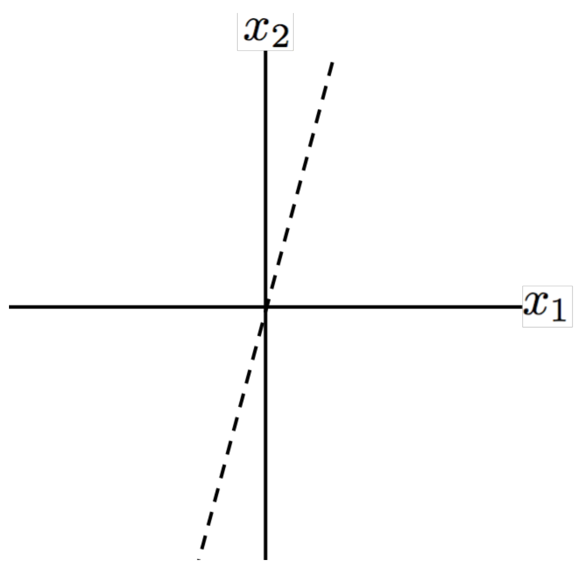


Figure 4.1: Span of $[1, 2]^T$ (dotted line) in $\mathbb{R}^2$

We've actually already encountered two important vector subspaces, though we haven't had the terminology to call them as such.

4.1.1 The kernel of A

The kernel of a matrix A, denoted $\ker(A)$, is the collection of all vectors $\mathbf{x}$ such that $A\mathbf{x} = \mathbf{0}$. In the language we've been using thus far, the kernel of A is the set of all solutions to the homogeneous equation $A\mathbf{x} = \mathbf{0}$. We can use the two step subspace test to verify that the kernel of A is in fact a subspace of the domain of A. Let $\mathbf{v}$ and $\mathbf{w}$ be two vectors from $\ker(A)$. We first need to verify that $\mathbf{v} + \mathbf{w}$ is in the kernel of A, too.

$$\begin{aligned} A(\mathbf{v} + \mathbf{w}) &= A\mathbf{v} + A\mathbf{w} \\ &= \mathbf{0}. \end{aligned}$$

So $\mathbf{v} + \mathbf{w}$ is a solution to $A\mathbf{x} = \mathbf{0}$, too. Said another way, the vector $\mathbf{v} + \mathbf{w}$ is the in kernel of A. For the second step in the test, we need to verify that $c\mathbf{v}$ is in the kernel.

$$\begin{aligned} A(c\mathbf{v}) &= cA\mathbf{v} \\ &= c\mathbf{0} \\ &= \mathbf{0}. \end{aligned}$$

So $c\mathbf{v}$ is in the kernel of A is indeed a subspace of the domain of A.

We can actually formulate the kernel of A explicitly using RREF. For a concrete example, consider the matrix

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

We can find the solutions to the homogeneous equation $A\mathbf{x} = \mathbf{0}$ by taking the RREF of the coefficient matrix A.

```
>> A = [1 -1; 1 -1];
>> rref(A)

ans =

     1    -1
     0     0
```

Defining $\mathbf{x} = [x_1, x_2]^T$ and reading off the first line, we have $x_1 - x_2 = 0$, so that $x_1 = x_2$ for any vector in the kernel of A. Then vectors in the kernel have the form $[x_1, x_2]^T = [x_2, x_2]^T = x_2[1, 1]^T$. Since x_2 is free in this solution, it can assume any value. Said another way, $\ker(A) = span([1, 1]^T)$. A graphical representation of these ideas can be seen in figure 4.2.

4.1.2 The image of A

Along with the kernel of A, there is another fundamental subspace of A: the image of A. The image of A, denoted $im(A)$, is defined as the set of all linear combinations of the columns of A. Said another way, the image of a matrix A is the span of the columns of A. The image of A is a subspace of the range of A. We can apply the two step subspace test here, too. Imagine that $\mathbf{v}$ and $\mathbf{w}$ are both in the image A. This implies that we can find solutions $\mathbf{x_1}$ and $\mathbf{x}_2$ such

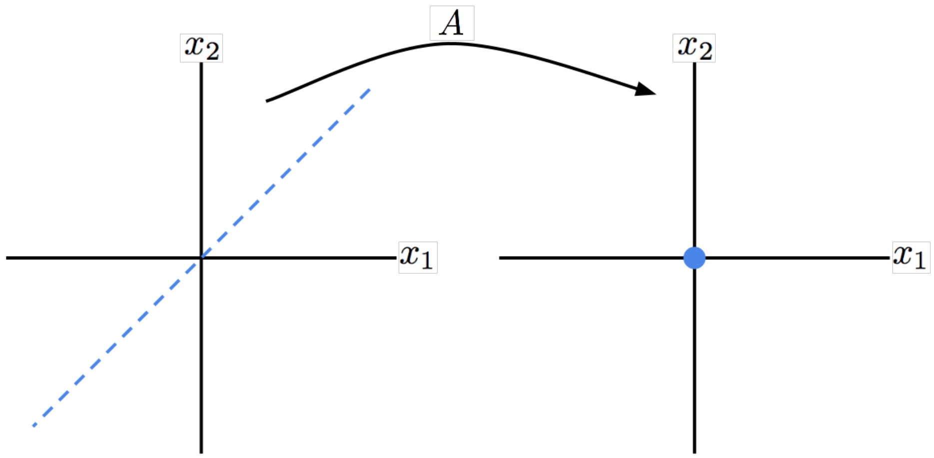


Figure 4.2: All vectors in the kernel of A (blue dashed line) get mapped to the zero vector (blue dot) by A.

that $A\mathbf{x}_1 = \mathbf{v}$ and $A\mathbf{x}_2 = \mathbf{w}$. We first need to verify that $\mathbf{v} + \mathbf{w}$ is in the image of A, too. This amounts to find a solution $\mathbf{x}_3$ such that $A\mathbf{x}_3 = \mathbf{v} + \mathbf{w}$.

$$\begin{aligned}\mathbf{v} + \mathbf{w} &= A\mathbf{x}_1 + A\mathbf{x}_2 \\ &= A(\mathbf{x}_1 + \mathbf{x}_2).\end{aligned}$$

So setting $\mathbf{x}_3 = \mathbf{x}_1 + \mathbf{x}_2$, we have our desired solution. For the second step, we need to verify that $c\mathbf{v}$ is in the image of A. This amounts to finding a solution $\mathbf{x}_4$ such that $A\mathbf{x}_4 = c\mathbf{v}$.

$$\begin{aligned}c\mathbf{v} &= c(A\mathbf{x}_1) \\ &= A(c\mathbf{x}_1.\end{aligned}$$

So setting $\mathbf{x}_4 = c\mathbf{x}_1$, we have our desired solution.

For an example, let's again consider the matrix

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

The image of A in this case all vectors of the form $\mathbf{v} = c_1[1,1]^T + c_2[-1,-1]^T$.

In this particular case we can simplify the form of any vector in the image of A.

$$\begin{aligned}\mathbf{v} &= c_1 \begin{bmatrix}1\\1\end{bmatrix} + c_2 \begin{bmatrix}-1\\-1\end{bmatrix}\\ &= c_1 \begin{bmatrix}1\\1\end{bmatrix} - c_2 \begin{bmatrix}1\\1\end{bmatrix}\\ &= (c_1 - c_2)\begin{bmatrix}1\\1\end{bmatrix}\end{aligned}$$

So any linear combination of the columns of A looks like a single constant $c_3 = c_1 - c_2$ times the first column of A. In other words, $im(A) = span([1, 1]^T)$. We can see a graphical representation of this example in figure 4.3.

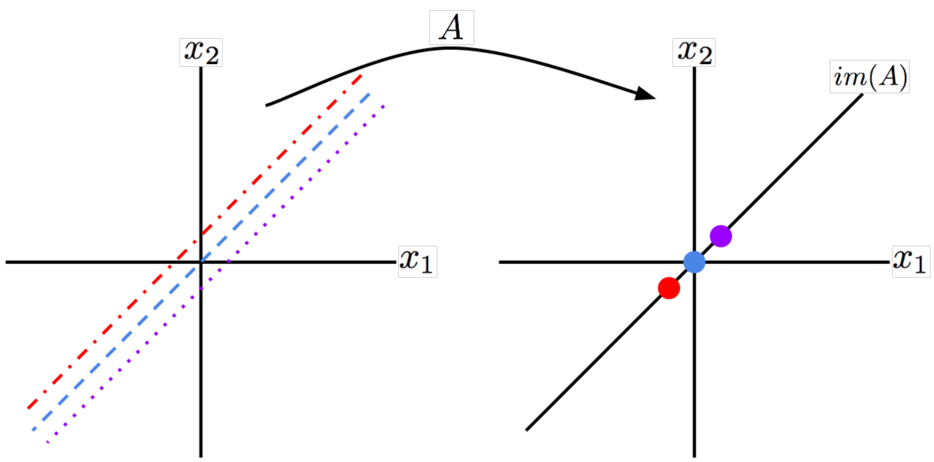


Figure 4.3: The image of A in this example is a 1-dimensional subspace of $\mathbb{R}^2$. The kernel of A (blue dashed line) is mapped to $\mathbf{0}$ by A. All vectors on the red dot-dashed line are mapped to the red dot below the x_1 axis, and all vectors on the purple dotted line are mapped to the purple dot above the x_1 axis. The entire $x_1 \times x_2$ plane on the left maps to the solid black line representing $im(A)$ on the right.

4.1.3 Connecting new to old

You might be starting to get the impression that in linear algebra, a lot of different statements about a matrix are equivalent. For instance, we've been building a list of equivalent conditions for the invertibility of A. An $n \times n$ matrix A is invertible if and only if

- There exists A^{-1} such that $A^{-1}A = AA^{-1} = I_n$.
- $rref(A) = I_n$.
- A has n pivots.
- The columns of A are linearly independent.
- The columns of A span $\mathbb{R}^n$.
- $\det(A) \neq 0$.
- $A\mathbf{v} = \mathbf{0}$ has only the trivial solution $\mathbf{v} = \mathbf{0}$.
- $\lambda = 0$ is *not* an eigenvalue of A.

But notice that if the columns of A span $\mathbb{R}^n$, then $im(A) = \mathbb{R}^n$, and that if $A\mathbf{v} = \mathbf{0}$ has only the trivial solution, then $\ker(A) = \mathbf{0}$. So these new concepts of the kernel and image of A are very closely tied to many of the topics we've seen thus far.

4.2 The Vector Norm and Orthogonality

We encountered the idea of inner products when we were first dealing with matrix-vector multiplication. Recall that the inner product of $\mathbf{v}$ and $\mathbf{w}$ in $\mathbb{R}^n$ is defined as

$$\mathbf{v}^T\mathbf{w} = \mathbf{w}^T\mathbf{v} = v_1w_1 + v_2w_2 + \ldots + v_nw_n.$$

For short hand, we often denote the inner product of $\mathbf{v}$ and $\mathbf{w}$ as $\mathbf{v} \circ \mathbf{w}$ or $< \mathbf{v}, \mathbf{w} >$. (We will typically use the former "dot product" notation, but you may see the latter notation in other texts; I want you to be comfortable with both.)

Inner products have a lot of convenient algebraic properties that we'll leverage throughout this section. For instance,

- $(\mathbf{u} + \mathbf{v}) \circ \mathbf{w} = \mathbf{u} \circ \mathbf{w} + \mathbf{v} \circ \mathbf{w}$
- $(c\mathbf{u}) \circ \mathbf{v} = c(\mathbf{u} \circ \mathbf{v})$.

You'll prove these facts on the studio associated with this section.

While we didn't dwell on inner products when we first saw them, the concept gives us all sorts of interesting tools. We'll see over the course of the next several sections that many of these tools are particularly useful when developing the core ideas behind regression. The three applications we'll cover in this section are length (norm) of a vector, the distance between two vectors, and the concept of orthogonality.

4.2.1 Norm

You probably ran into the idea of the length of a line segment fairly early on in your mathematical development. As a refresher, the length of the line segment connecting the origin to the point (a, b) is $\ell = \sqrt{a^2 + b^2}$. This is a consequence of the old-fashioned Pythagorean theorem you saw in your first Euclidean geometry class. We can see a graphical representation of this idea in figure 4.4.

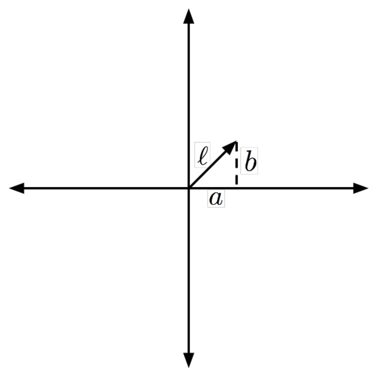


Figure 4.4: The norm of $[a, b]^T$ is just it's Euclidean length ℓ. This idea generalizes to higher dimensions.

If we think of the point (a, b) as a vector $\mathbf{x} = [a, b]^T$ in $\mathbf{R}^2$, we can define the *norm* (or *length*) of $\mathbf{x}$ to be is Euclidean length. Mathematically, we write $\|\mathbf{x}\| = \sqrt{a^2 + b^2}$. The key connection to make here is that the definition of the norm of $\mathbf{x}$ is actually a statement about the inner product of $\mathbf{x}$ with itself.

$$\begin{aligned}\|\mathbf{x}\| &= \sqrt{a^2 + b^2} \\ &= \sqrt{\mathbf{x} \circ \mathbf{x}}.\end{aligned}$$

We can easily generalize this idea to vectors living in higher dimensions. We define the norm of a vector $\mathbf{x} \in \mathbb{R}^n$ to be

$$\begin{aligned}\|\mathbf{x}\| &= \sqrt{\mathbf{x} \circ \mathbf{x}} \\ &= \sqrt{x_1^2 + x_2^2 + \ldots + x_n^2}.\end{aligned}$$

Note that since $\|\mathbf{x}\| = \mathbf{x} \circ \mathbf{x}$, we know that $\mathbf{x} \circ \mathbf{x} \geq 0$ and $\mathbf{x} \circ \mathbf{x} = 0$ if and only if $\mathbf{x} = \mathbf{0}$.

4.2.2 Distance between two vectors

Imagine we have two vectors $\mathbf{u}$ and $\mathbf{v}$ in $\mathbf{R}^2$. It might be useful to have some notion of how far away these vectors are from one another. In the real numbers, we defined the distance between x and y on the real number line to be $|x - y|$, that is, the absolute value of the difference of the numbers. Here we'll have to do things slightly differently, but looking at the difference of the vectors turns out to be a good place to start.

But what does the difference $\mathbf{u} - \mathbf{v}$ look like? One easy way to get a handle on this is to think about the vector $-\mathbf{v}$ in isolation for a moment. The vector is just $\mathbf{v}$ in the "opposite direction". Noting that $\mathbf{u} - \mathbf{v} = \mathbf{u} + (-\mathbf{v})$, we can start to make sense of the vector difference at hand. In figure 4.5, we can see that the distance between $\mathbf{u}$ and $\mathbf{v}$ is just $\|\mathbf{u} - \mathbf{v}\|$. While the sketch we've done here is in $\mathbf{R}^2$, this idea holds generally in $\mathbf{R}^n$.

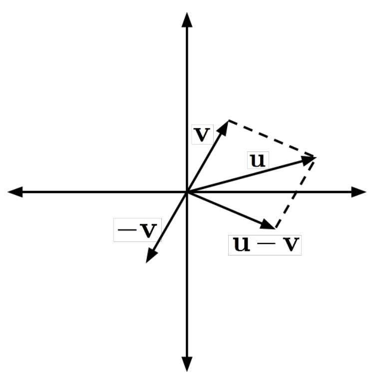


Figure 4.5: The distance between $\mathbf{u}$ and $\mathbf{v}$ is $\|\mathbf{u} - \mathbf{v}\|$.

For a concrete example, consider two vectors in $\mathbf{R}^2$ defined as $\mathbf{u} = [5, 5]^T$ and $\mathbf{v} = [1, 2]^T$. What is the distance between these two vectors? Using the tools we just created,

$$\begin{aligned}
\|\mathbf{u} - \mathbf{v}\| &= \left\| \begin{bmatrix} 5 \\ 5 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right\| \\
&= \left\| \begin{bmatrix} 4 \\ 3 \end{bmatrix} \right\| \\
&= \sqrt{4^2 + 3^2} \\
&= 5.
\end{aligned}$$

For an example in higher dimensions, consider $\mathbf{u} = [4, 19, 5]^T$ and $\mathbf{v} =$

$[2, 18, 7]^T$ in $\mathbb{R}^3$. Then the distance between the two vectors is

$$\begin{aligned}\|\mathbf{u}-\mathbf{v}\| &= \left\| \begin{bmatrix} 4 \\ 19 \\ 5 \end{bmatrix} - \begin{bmatrix} 2 \\ 18 \\ 7 \end{bmatrix} \right\| \\ &= \left\| \begin{bmatrix} 2 \\ 1 \\ -2 \end{bmatrix} \right\| \\ &= \sqrt{2^2 + 1^2 + (-2)^2} \\ &= 3.\end{aligned}$$

You might be starting to think that this sort of calculation becomes intractable (or at least very inconvenient) if the vectors have many components. Here the `norm` command in Matlab can be very useful. In Matlab, our previous example reads

```
>> norm([4;19;5] - [2;18;7])

ans =

     3
```

We'll see that vectors with norm 1, that is $\|\mathbf{x}\| = 1$, are key to the development of regression (and many other linear algebraic techniques). We'll deal with these in much further detail in the next section.

4.2.3 Orthogonality

You may remember from your first geometry course that line segments at right angles to one another gave you a huge advantage in proving things in a problem. The same is true in higher dimensions, but we need a slightly more involved definition to encapsulate the same idea. We see two vectors $\mathbf{u}$ and $\mathbf{v}$ are *orthogonal* to one another if $\mathbf{u} \circ \mathbf{v} = 0$.

For low-dimensional example, consider the definitions of $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{R}^2$ as seen in figure 4.6. We can clearly see that the two are at a right angle to one another. We can verify that the two are also orthogonal using our new definition.

$$\begin{aligned}\mathbf{u} \circ \mathbf{v} &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \circ \begin{bmatrix} -1 \\ 1 \end{bmatrix} \\ &= (1)(-1) + (1)(1) \\ &= 0.\end{aligned}$$

For an example in higher dimensions, consider the definitions of $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{R}^3$. Here it's more difficult to visualize the fact that these two vectors are

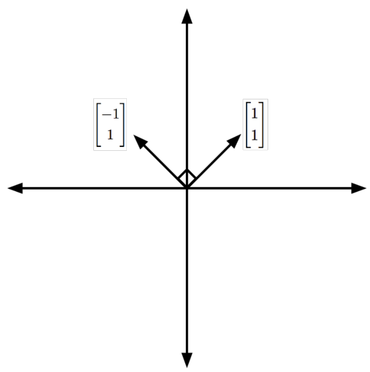


Figure 4.6: In low dimensions, we can think of orthogonality corresponding to right angles.

at a right angle to one another, we it's still very easy to verify that they are orthogonal.

$$\begin{aligned} \mathbf{u} \circ \mathbf{v} &= \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} \circ \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} \\ &= (2)(-1) + (1)(1) + (1)(1) \\ &= 0. \end{aligned}$$

This really brings us to the necessity of defining orthogonality rather than just relying on the old definition of right angles. In high dimensions, it's increasingly difficult to make sense of the idea of an angle between vectors, but it continues to be easy to talk about vectors being orthogonal to one another.

4.2.4 The orthogonal complement

With orthogonality in hand, we can bring forward one of the key ideas in the development of regression: the orthogonal complement of a vector space. Given a subspace W of a vector space V, we can think about the collection of all vectors from V that are orthogonal to *every* vector in W. We call this set of vectors the *orthogonal complement* of W, denoted $W^\perp$ and pronounced "W perp". For two examples of what an orthogonal complement might look like, consider figure 4.7.

What's fascinating (and useful) is that $W^\perp$ is a subspace of V, too! We can verify this claim using the two step subspace test we developed in the previous section. Let $\mathbf{u}$ and $\mathbf{v}$ be two vectors in $W^\perp$. To prove that $W^\perp$ is a subspace of V, we need to confirm that both $\mathbf{u} + \mathbf{v}$ and $c\mathbf{u}$ are members of $W^\perp$. First

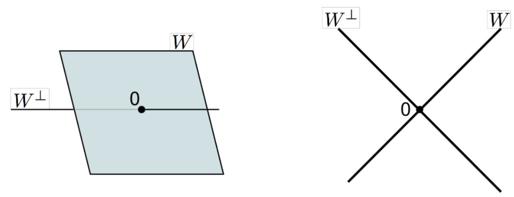


Figure 4.7: Two examples of subspaces W and their orthogonal complement. Every vector in $W^\perp$ is orthogonal to every vector in W.

note that by the definition of $W^\perp$, $\mathbf{u} \circ \mathbf{w} = 0$ and $\mathbf{v} \circ \mathbf{w} = 0$ for every $\mathbf{w} \in W$. We can use these facts to confirm that $\mathbf{u} + \mathbf{v}$ is also a member of the orthogonal complement of W.

$$\begin{aligned}(\mathbf{u} + \mathbf{v}) \circ \mathbf{w} &= \mathbf{u} \circ \mathbf{w} + \mathbf{v} \circ \mathbf{w} \\ &= 0 + 0 = 0.\end{aligned}$$

So the sum $\mathbf{u} + \mathbf{v}$) is part of $W^\perp$, since it is orthogonal to every $\mathbf{w} \in W$. We can use a similar strategy to show that $c\mathbf{u}$ is in the orthogonal complement of W.

$$\begin{aligned}(c\mathbf{u}) \circ \mathbf{w} &= c(\mathbf{u} \circ \mathbf{w}) \\ &= c(0) = 0.\end{aligned}$$

We've shown that $W^\perp$ passes the two step subspace test, and so $W^\perp$ is a subspace of V.

Let's see an example. Consider $W = span([1, 2]^T)$, a subspace of $\mathbb{R}^2$ as you proved over the course of the studio problems attached to the previous section. Any $\mathbf{x} \in W^\perp$ satisfies $\mathbf{x} \circ \mathbf{w}$ for every $\mathbf{w} \in W$. But note that any such $\mathbf{w}$ has the form $c_1[1, 2]^T$. A little arithmetic will give us a convenient expression for $\mathbf{x}$.

$$\begin{aligned}0 &= \mathbf{x} \circ \mathbf{w} \\ &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \circ c_1 \begin{bmatrix} 1 \\ 2 \end{bmatrix} \\ &= c_1 \left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 2 \end{bmatrix} \right) \\ &= (x_1)(1) + (x_2)(2) \\ -2x_2 &= x_1.\end{aligned}$$

This implies that any vector $\mathbf{x}$ in the orthogonal complement $W^\perp$ has the form

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= \begin{bmatrix} -2x_2 \\ x_2 \end{bmatrix} \\ &= x_2 \begin{bmatrix} -2 \\ 1 \end{bmatrix}.\end{aligned}$$

Note that this holds for *any* choice of x_2! We can conclude that $W^\perp = span([-2, 1]^T)$, which is a subspace of $\mathbb{R}^2$.

Perhaps the most important property of orthogonal complements is one that connects them to the ideas the image and kernel of a matrix A that we saw in the previous section. Here we make the bold claim $im(A)^\perp = \ker A^T$ for any $m \times n$ matrix A. Before we dive into showing why this is true, let's take a second to think about what this means. The left side of this equation is the set of all vectors that are perpendicular to every linear combination of the columns of A, and the right side is the set of all solutions to $A^T\mathbf{x} = \mathbf{0}$. These may seem like very separate ideas, and it's exactly because they seem so separate that the theorem is so powerful.

We'll show that these two collections of vectors are equal by showing that any vector $\mathbf{x}$ in $im(A)^\perp$ must be in $\ker A^T$ and vice versa. First, let's assume that $\mathbf{x} \in \ker A^T$, so that $A^T\mathbf{x} = \mathbf{0}$. By the definition of matrix-vector multiplication, this implies that $\mathbf{r} \circ \mathbf{x} = 0$ for every row $\mathbf{r}$ of A^T. But the rows of A^T are the columns of A. We conclude that $\mathbf{x}$ is orthogonal to every column of A. But if $\mathbf{x}$ is orthogonal to every column of A, then $\mathbf{x}$ is orthogonal to all linear combinations of columns of A. This may be easier to see once we write it out. Define $\mathbf{y} = c_1\mathbf{a}_1 + c_2\mathbf{a}_2 + \ldots c_n\mathbf{a}_n$, where $\mathbf{a}_i$ is the i^{th} column of A, so that $\mathbf{y}$ is a member of $im(A)$. Using the fact that $\mathbf{x}$ is orthogonal to each of the columns $\mathbf{a}_i$, we can see that $\mathbf{x}$ is orthogonal to $\mathbf{y}$, too.

$$\begin{aligned}\mathbf{x} \circ \mathbf{y} &= \mathbf{x} \circ (c_1\mathbf{a}_1 + c_2\mathbf{a}_2 + \ldots c_n\mathbf{a}_n) \\ &= c_1(\mathbf{x} \circ \mathbf{a}_1) + c_2(\mathbf{x} \circ \mathbf{a}_2) + \ldots + c_n(\mathbf{x} \circ \mathbf{a}_n) \\ &= 0.\end{aligned}$$

We can conclude at this point that every member $\mathbf{x}$ of $\ker A^T$ is also a member of $im(A)^\perp$. To finish off the proof, we need to show that any member of $im(A)^\perp$ is also a member of $\ker A^T$.

Redefine $\mathbf{x}$ to be a member of $im(A)^\perp$. This implies that $\mathbf{x} \circ \mathbf{y} = 0$ for every $\mathbf{y}$ that is a linear combination of the columns of A. If we consider some very particular linear combinations of the columns, we can make some headway.

$$\begin{aligned}
0 &= \mathbf{x} \circ (1\mathbf{a}_1 + 0\mathbf{a}_2 + \ldots + 0\mathbf{a}_n) = \mathbf{x} \circ \mathbf{a}_1 \\
0 &= \mathbf{x} \circ (0\mathbf{a}_1 + 1\mathbf{a}_2 + \ldots + 0\mathbf{a}_n) = \mathbf{x} \circ \mathbf{a}_2 \\
&\vdots \\
0 &= \mathbf{x} \circ (0\mathbf{a}_1 + 0\mathbf{a}_2 + \ldots + 1\mathbf{a}_n) = \mathbf{x} \circ \mathbf{a}_n.
\end{aligned}$$

Packaging these equations up as a matrix equation gives an interesting insight.

$$\mathbf{0} = \begin{bmatrix} \mathbf{x} \circ \mathbf{a}_1 \\ \mathbf{x} \circ \mathbf{a}_2 \\ \vdots \\ \mathbf{x} \circ \mathbf{a}_n \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_n \end{bmatrix} \mathbf{x} = A^T \mathbf{x}.$$

This implies that $\mathbf{x}$ is a member of $\ker A^T$. Since we've show that every member of $im(A)^\perp$ must be a member of $\ker A^T$ and vice versa, it must be the case that the collections of vectors are *identical*. So it really is the case that for *any* $m \times n$ matrix A, $im(A)^\perp = \ker A^T$!

We can try to tie this together with an example. Define A by

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

We can learn about the kernel of A^T by row reducing the coefficient matrix A^T.

```
>> rref([1 2; 1 2]')

ans =

     1     1
     0     0
```

The first line reads $x_1 + x_2 = 0$. So any vector $\mathbf{x}$ in the kernel of A^T has the form $\mathbf{x} = [x_1, x_2]^T = [-x_2, x_2]^T = x_2[-1, 1]^T$. Since x_2 is free, we can conclude that $\ker A^T = span([-1, 1]^T)$.

To confirm that $im(A)^\perp$ is the same subspace of $\mathbb{R}^2$, we'll derive a universal form for a vector $\mathbf{x} \in im(A)^\perp$. With $\mathbf{x}$ defined in this way, we have $\mathbf{x} \circ \mathbf{y} = 0$ for every $\mathbf{y}$ in the image of A. Any such $\mathbf{y}$ is a linear combination of the columns, a fact that we can use to our advantage.

$$\begin{aligned}
0 &= \mathbf{x} \circ \mathbf{y} \\
&= \mathbf{x} \circ \left(c_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix} + c_2 \begin{bmatrix} 2 \\ 2 \end{bmatrix} \right) \\
&= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \circ \begin{bmatrix} c_1 + 2c_2 \\ c_1 + 2c_2 \end{bmatrix} \\
&= x_1(c_1 + 2c_2) + x_2(c_1 + 2c_2) \\
-x_2 &= x_1.
\end{aligned}$$

So any vector $\mathbf{x}$ in the orthogonal complement of the image of A has the form $\mathbf{x} = [x_1, x_2]^T = [-x_2, x_2]^T = x_2[-1, 1]^T$. Since this identity holds for any choice of x_2, we can conclude that $\ker A^T = span([-1, 1]^T)$. We see concretely that $im(A)^T = span([-1, 1]^T = \ker A^T$.

4.3 Projections and Least-Squares Data Fitting

You have probably encountered the idea of data fitting in one of your previous applied mathematics courses. The general idea is that we have access to a bunch of data points, and we're looking for the linear (or higher order polynomial, exponential, logarithmic, logistic, multilinear, *etc.*) model that best fits the given data. Typically the "best model" is defined to be the one that minimizes the sum of squared errors between the values predicted by the model and the values observed in the data.

For an example, consider the following situation. Suppose we have (x, y) data points $(-2, 0), (-1, 2), (0, 1), (1, 3)$, and $(2, 5)$, as plotted in figure 4.8. Clearly there is a trend in the data, but just as clearly the data points do not all fall on a single line $y = \beta_0 + \beta_1 x$. It's an interesting problem (with a potential very useful solution) to determine what the best linear model for these data would be.

We'll see over the course of the section that while linear, polynomial, exponential, logistic and multilinear regression are often taught (and thought of) as separate techniques, they are all fundamentally the same in the linear algebraic context. A single tool, namely projection, generates all of these examples and more.

4.3.1 Projection

Let's assume that $W = span(\mathbf{u})$ is a subspace of V. Consider another vector $\mathbf{y} \notin W$. Since any vector $\mathbf{w} \in W$ has the for $c_1\mathbf{u}$, this directly implies that $\mathbf{y} \neq c_1\mathbf{u}$ for any weight c_1. We can see this configuration graphically in figure 4.9.

What is the closest vector in W to $\mathbf{y}$? Using the idea of the distance between two vectors that we developed in the last section, we can state this question

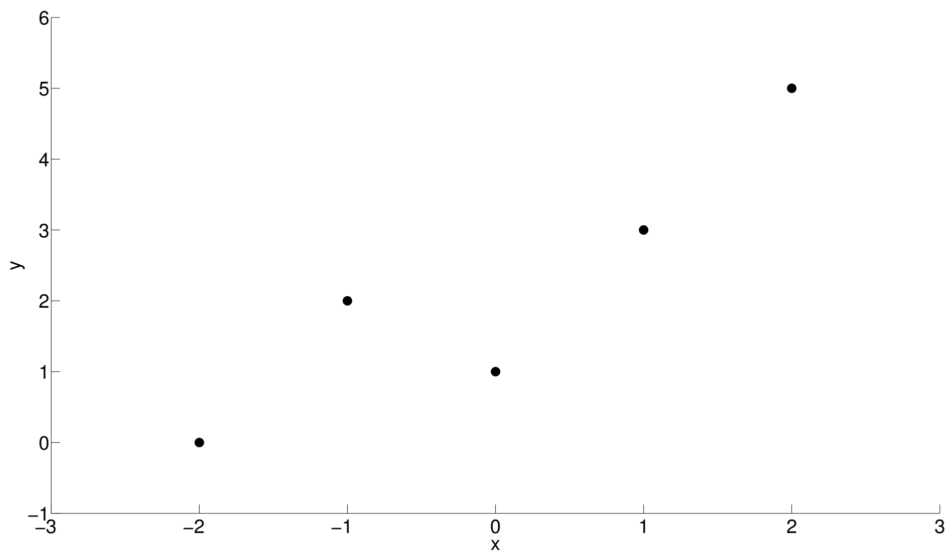


Figure 4.8: Data points $(-2,0), (-1,2), (0,1), (1,3)$, and $(2,5)$ are not collinear.

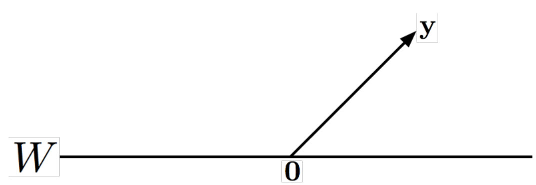


Figure 4.9: The vector $\mathbf{y}$ is not in the subspace $W = span(\mathbf{u})$.

another way: what is the vector $\hat{\mathbf{y}} \in W$ that minimizes $\|\mathbf{y} - \hat{\mathbf{y}}\|$? It may help to explore this idea graphically at first. Consider figure 4.10. Clearly the line that forms a right angle with W has the shortest length. But which vector in W is directly under $\mathbf{y}$? Remembering that right angles correspond to orthogonality, we can make some headway.

Since $W = span(\mathbf{u})$, any vector in the subspace has the form $c_1\mathbf{u}$. So we can define the vector closest to $\mathbf{y}$ as

$$\hat{\mathbf{y}} = c_1\mathbf{u}.$$

The vector $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to W. Said a different way, $\mathbf{y} - \hat{\mathbf{y}} \in W^{\perp}$. We

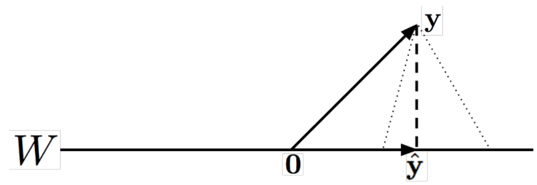


Figure 4.10: The closest vector $\hat{\mathbf{y}}$ to $\mathbf{y}$ in $W = span(\mathbf{u})$ lies directly beneath $\mathbf{y}$. The vector difference $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to W.

can work with this.

$$\begin{aligned}(\mathbf{y} - \hat{\mathbf{y}}) \circ \mathbf{u} &= 0 \\ \mathbf{y} \circ \mathbf{u} - c_1(\mathbf{u} \circ \mathbf{u}) &= 0 \\ \frac{\mathbf{y} \circ \mathbf{u}}{\mathbf{u} \circ \mathbf{u}} &= c_1\end{aligned}$$

Since the dot product of two vectors is just a scalar, so is the weight c_1. We can bring our results together by defining the closest vector in W to $\mathbf{y}$ as

$$\hat{\mathbf{y}} = \frac{\mathbf{y} \circ \mathbf{u}}{\mathbf{u} \circ \mathbf{u}}\mathbf{u}.$$

We call $\hat{\mathbf{y}}$ the *projection* of $\mathbf{y}$ onto W, denoted $\hat{\mathbf{y}} = proj_W \mathbf{y}$.

Let's consider a concrete example in low dimension. Define

$$\mathbf{y} = \begin{bmatrix} 7 \\ 6 \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}.$$

What is the vector closest to $\mathbf{y}$ in $W = span(\mathbf{u})$? Using our new definition of projection, we can compute this vector pretty easily.

$$\begin{aligned}\hat{\mathbf{y}} &= \frac{\mathbf{y} \circ \mathbf{u}}{\mathbf{u} \circ \mathbf{u}}\mathbf{u} \\ &= \frac{(7)(4) + (6)(2)}{(4)(4) + (2)(2)}\mathbf{u} \\ &= 2\mathbf{u}.\end{aligned}$$

So the closest vector to $\mathbf{y}$ in W is $2\mathbf{u} = [8, 4]^T$.

We call the distance $\|\mathbf{y} - \hat{\mathbf{y}}\|$ the *error* of the projection. This corresponds exactly to the idea of sum of squared errors that you might have seen in a earlier course. To see this, we just need to expand the error in terms of the components of the vectors.

$$\|\mathbf{y} - \hat{\mathbf{y}}\| = \sqrt{(b_1 - \hat{b}_1)^2 + (b_2 - \hat{b}_2)^2 + \ldots + (b_n - \hat{b_n})^2}.$$

(Note that the square root doesn't really matter here, because any solution that minimizes the sum of squared errors also minimizes the square root of the sum of squared errors because the square root function is invective and increasing.) Notice that the smaller the error, the closer the original vector was to its ultimate projection. Said another way, the smaller the error, the better the approximation of the projection of the original vector.

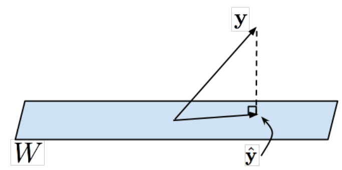


Figure 4.11: The closest vector $\hat{\mathbf{y}}$ to $\mathbf{y}$ in $W = span(\mathbf{u})$ lies directly beneath $\mathbf{y}$. The vector difference $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to W.

Projection in higher dimensions works in much the same way. Consider the projection of a vector $\mathbf{y} \in \mathbb{R}^3$ onto a 2-dimensional subspace $W = span(\mathbf{u}, \mathbf{v})$ of $\mathbb{R}^3$ featured in figure 4.11. Here we can still think of the projection lying directly below the original vector. We can think about the projection $\hat{\mathbf{y}}$ in at least three different ways, all of which give us different insights.

- The projection $\hat{\mathbf{y}}$ is the linear combination of $\mathbf{u}$ and $\mathbf{v}$ that is closest to the original vector $\mathbf{y}$.
- The projection $\hat{\mathbf{y}}$ is the closest vector in $W = span(\mathbf{u}, \mathbf{v})$ to the original vector $\mathbf{y}$.
- If we define a matrix A that has $\mathbf{u}$ and $\mathbf{v}$ as its columns the projection $\hat{\mathbf{y}}$ is the closest vector in $W = im(A)$ to the original vector $\mathbf{y}$.

This last mindset will prove to be the most useful, because it connects the idea of projections onto a subspace to all our results on images, kernels, and orthogonal complements.

For even higher dimensions, the projection of $\mathbf{y}$ onto span of a collection of vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k$ is the same as the projection of $\mathbf{y}$ onto the image of $A = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k]$, but the idea of the projection $\hat{\mathbf{y}}$ lying "directly below" the original vector $\mathbf{y}$ starts to fall apart as our geometric intuition degrades. In these cases, it's more convenient to think of orthogonality: the vector difference $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to $W = im(A)$. Using the notation we developed in the previous section, $\mathbf{y} - \hat{\mathbf{y}} \in W^{\perp} = im(A)^{\perp}$. We'll see that this idea is fundamental to the idea of least-squares solutions.

4.3.2 Least-squares problems

In least squares problems, we typically have a large number of data, and we want find out what linear (or polynomial, exponential, logarithmic, logistic, multilinear, *etc.*) model best fits the observed data. The fundamental problem here is that no single model fits the data perfectly, and so we must compromise. From a linear algebraic perspective, we can think of a collection of data that has no "perfect" fit as an iconsistent system. For an example, let's return to the setup we introduced at the beginning of the section.

Suppose we have (x, y) data points $(-2, 0), (-1, 2), (0, 1), (1, 3)$, and $(2, 5)$. Considering this problem as a linear system, we have

$$\begin{aligned} 0 &= \beta_0 - 2\beta_1 \\ 2 &= \beta_0 - \beta_1 \\ 1 &= \beta_0 \\ 3 &= \beta_0 + 1\beta_1 \\ 5 &= \beta_0 + 2\beta_1. \end{aligned}$$

We can reduce this system of equations to a single matrix equation.

$$\begin{bmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 3 \\ 5 \end{bmatrix}$$

$$A\beta = \mathbf{y}.$$

If we were to take the RREF of the augmented matrix here, we could clearly see that the system is inconsistent, which confirms our graphical intuition developed at the start of the section. But how do we proceed?

Let's define $\hat{\mathbf{y}} = \hat{\beta}_0 + \hat{\beta}_1\mathbf{x}$ to be the line that minimizes the sum of squared errors $\|\hat{\mathbf{y}} - \mathbf{y}\|^2$. As a matrix equation, the line of best fit is $\hat{\mathbf{y}} = A\hat{\beta}$. Since the vector $\mathbf{x}$ is given in the problem, we know the matrix A, and so in order to identify the line of best fit, we just need to find the coefficients in β that minimize $\|\hat{\mathbf{y}} - \mathbf{y}\|^2$. We know that projections minimizes $\|\hat{\mathbf{y}} - \mathbf{y}\|^2$, but it still might be unclear what subspace we're projecting onto here. We want $\hat{\mathbf{y}}$ to be a linear combination of the columns of A, namely the linear combination $\hat{\mathbf{y}} = A\hat{\beta}$. So we want $\hat{\mathbf{y}}$ to live in the image of A, which we'll call $W = im(A)$. This connection is what allows us to push forward.

Remember that if $\hat{\mathbf{y}} = proj_W\mathbf{y}$, then the vector difference $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to W. Linking this observation with the idea of orthogonal complements, we can state the condition as $\mathbf{y} - \hat{\mathbf{y}} \in W^{\perp}$. Here $W = im(A)$, and as we saw in the last section $im(A)^{\perp} = \ker(A^T)$. This directly implies that $A^T(\mathbf{y} - \hat{\mathbf{y}}) = \mathbf{0}$. This really gives us some leverage.

$$\begin{aligned} A^T(\mathbf{y} - \hat{\mathbf{y}}) &= \mathbf{0} \\ A^T\mathbf{y} - A^T\hat{\mathbf{y}} &= \\ A^T\mathbf{y} - A^TA\hat{\beta} &= \\ A^TA\hat{\beta} &= A^T\mathbf{y}. \end{aligned}$$

If A^TA is invertible, there is a unique line of best fit, and we can solve for the desired coefficients $\hat{\beta}$.

$$\hat{\beta} = (A^TA)^{-1}A^T\mathbf{y}.$$

If A^TA is not invertible, then there are infinitely many lines of best fit, and we can solve for one of the sets of desired coefficients by row-reducing the augmented matrix of $A^TA\hat{\beta} = A^T\mathbf{y}$.

Let's get back to our linear regression example. Here, we have

$$A^TA\hat{\beta} = A^T\mathbf{y}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ 1 \\ 3 \\ 5 \end{bmatrix}$$

Let's hope that A^TA has an inverse and try to compute $\hat{\beta} = (A^TA)^{-1}A^T\mathbf{y}$. (If A^TA is not invertible, Matlab will tell us through an error message after it tries to execute the `inv` command.)

```
>> A = [1, -2; 1, -1; 1, 0; 1, 1; 1, 2];
>> y = [0;2;1;3;5];
>> betaHat = inv(A'*A)*A'*y

betaHat =

    2.2000
    1.1000
```

The inverse of A^TA evidently exists because the command executed without any problems. We should interpret this output in the following way: there is a unique line $\hat{y} = 2.2 + 1.1x$ that minimizes the sum of squared errors for the given data. We can see a graphical representation out conclusion in figure 4.12.

Polynomial regression

Let's imagine that we have the same (x, y) data points $(-2, 0)$, $(-1, 2)$, $(0, 1)$, $(1, 3)$, and $(2, 5)$, but that we want to find the cubic function $y = \beta_0 + \beta_1 x + \beta_2 x^3 + \beta_3 x^3$ that minimizes the sum of squared errors between the predictions and the corresponding observations. We'll skip the step of writing this situation out as a system of linear equations and skip directly to a matrix equation formulation of

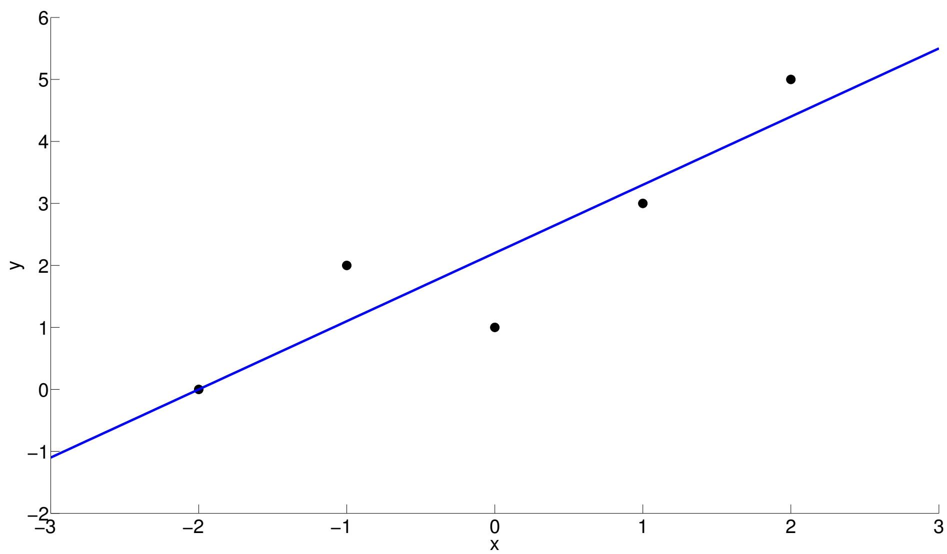


Figure 4.12: Data points $(-2,0), (-1,2), (0,1), (1,3)$, and $(2,5)$ and a sketch of the linear least-squares solution $\hat{y} = 2.2 + 1.1x$.

the problem.

$$\begin{bmatrix} 1 & -2 & (-2)^2 & (-2)^3 \\ 1 & -1 & (-1)^2 & (-1)^3 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 2^2 & 2^3 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 3 \\ 5 \end{bmatrix}$$
$$A\beta = \mathbf{y}.$$

As with the linear regression example, there are more equations than variables in this linear system, and so we should expect that for most values of $\mathbf{y}$, the linear system will be inconsistent. We can skip right to the chase and calculate the coefficients $\beta = (A^T A)^{-1} A^T \mathbf{y}$ using Matlab.

```
>> A = [1 -2 4 -8; 1 -1 1 -1; 1 0 0 0; 1 1 1 1; 1 2 4 8];
>> y = [0;2;1;3;5];
>> betaHat = inv(A'*A)*A'*y

betaHat =

    1.7714
    0.2500
    0.2143
    0.2500
```

This output is telling us that there is a unique cubic function $\hat{y} = 1.7714 + 0.25x + 0.2143x^2 + 0.25x^3$ that minimizes the sum of squared errors between the predicted values and the given data. We can see a sketch of the function in figure 4.13

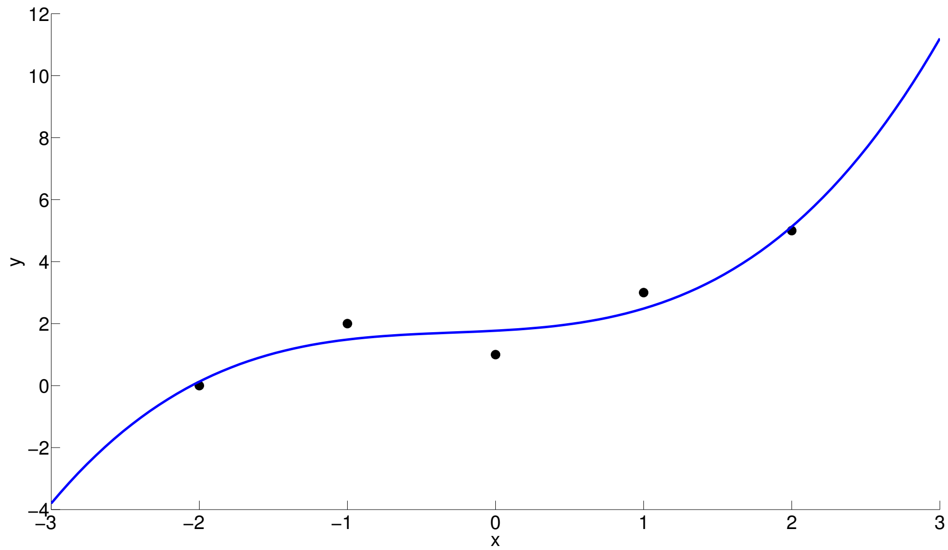


Figure 4.13: Data points $(-2, 0), (-1, 2), (0, 1), (1, 3)$, and $(2, 5)$ and a sketch of the cubic least-squares solution $\hat{y} = 1.7714 + 0.25x + 0.2143x^2 + 0.25x^3$.

Exponential regression

Let's again imagine that we have (x, y) data points $(-1, 2), (0, 1), (1, 3)$, and $(2, 5)$. (This is same list as before, only with the data point $(-2, 0)$ removed.) This time, we'll look for the exponential function $y = \beta_0 e^{\beta_1 x}$. The form of this model looks a little trickier, in particular because it's unclear how we'll formulate the linear system associated with the data points. We can manipulate the equation to put it into a more convenient form. We can certainly take the natural logarithm of both sides of the equation.

$$\begin{aligned} y &= \beta_0 e^{\beta_1 x} \\ \ln(y) &= \ln(\beta_0 e^{\beta_1 x}) \end{aligned}$$

Now we have to remember some properties of logarithms. Most fundamentally, logarithms are the mathematical tool that allow us to turn products into sums. In symbols, $\ln(xy) = \ln(x) + \ln(y)$. We can use this fact, and the property that $\ln(e^k) = k$, to further reduce the equation.

$$\begin{aligned} \ln(y) &= \ln(\beta_0 e^{\beta_1 x}) \\ &= \ln(\beta_0) + \beta_1 x. \end{aligned}$$

Now this final form may seem like a problem to you, most likely because the coefficient β_0 is inside the logarithm. But if we define a new variable $\beta_0' = \ln(\beta_0)$, we have an equation that is a linear combination of our coefficients.

$$\ln(y) = \beta_0' + \beta_1 x.$$

Performing this type of modification, both to the equation and to the coefficients that we're solving for themselves is perfectly legal mathematically. We just need to remember that the results we get from Matlab may need to be interpreted differently based on the changes we've made to the problem. We'll see how this works at the end of this example.

We can write the matrix equation associated with this new model and the given data points.

$$\begin{bmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} \beta_0' \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \ln(2) \\ \ln(1) \\ \ln(3) \\ \ln(5) \end{bmatrix}$$

Notice how similar the form of this system is to that of the linear regression model, despite the fact that the models are quite different. This is starting to show us that there are deep linear algebraic properties that are connecting all of these least-squares problems. The method for solving for the unknown coefficients β_0' and β_1 is exactly the same as the previous two examples.

```
>> A = [1, -1; 1, 0; 1, 1; 1, 2];
>> y = [2; 1; 3; 5];
>> y = log(y);
>> betaHat = inv(A'*A)*A'*y

betaHat =

    0.6579
    0.3847
```

Let's point out a couple items of interest before we discuss the results. In Matlab, `log` is the natural logarithm, not the base 10 logarithm like you might expect. Second, notice that you can take the logarithm of a vector component-wise; this may be easier than writing `log()` a bunch of times.

We'll need to work a little harder to interpret the results here. Remember that we defined $\beta_0' = \ln(\beta_0)$. Matlab is indicated that $\beta_0' = 0.6579$, which implies that the actual coefficient we're looking for is $\beta_0 = e^{0.6579} = 1.9307$. We can then safely claim that the unique exponential model that minimizes the sum of squared errors between its predictions and the observations is $\hat{y} = 1.9307e^{0.3847x}$. We can see a sketch of the function in figure 4.14.

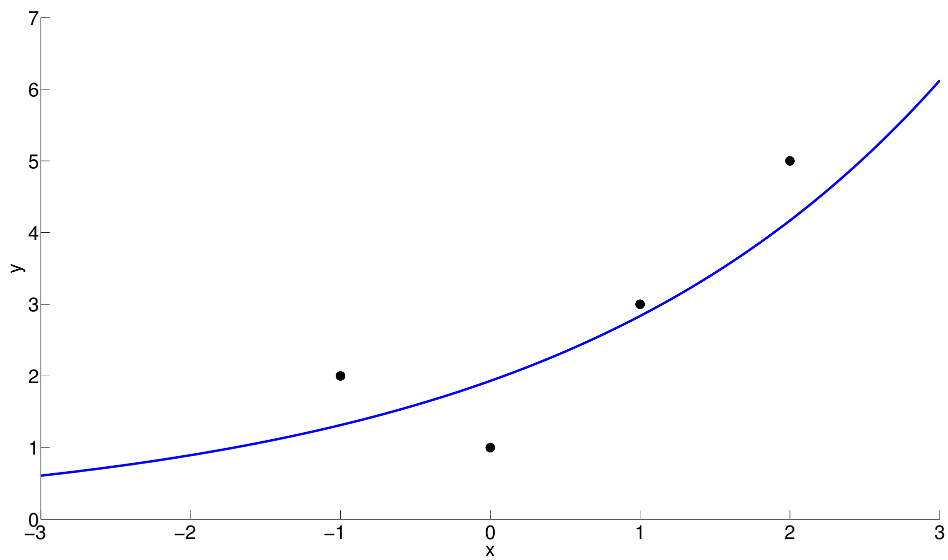


Figure 4.14: Data points $(-1,2),(0,1),(1,3)$, and $(2,5)$ and a sketch of the exponential least-squares solution $\hat{y} = 1.9307e^{0.3847}$.

Logistic regression

In probability, the cumulative density function $F(x)$ represents the probability that a random variable X will have value less than or equal to x. Stated symbolically, $F(x) = P(X \leq x)$. Estimating distributions is a really useful tool in all sorts of fields. And since the data that we collect may not always be consistent and/or we would like a simpler model that estimates but does not perfectly capture certain trends, finding the best estimate of the underlying distribution is important. One particularly popular model is

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x}+1},$$

Notice that $\pi(x)$ is between 0 and 1 for every x. Moreover, we see that $\pi(x) \to 1$ as $x \to \infty$ and $\pi(x) \to 0$ as $x \to -\infty$. So $\pi(x)$ seems like a pretty good candidate for a CDF.

It might be really unclear at this point how we'll turn this model into something that looks linear in the unknown coefficients β_0 and β_1. It will certainly require more elbow grease than we've needed in the previous examples.

$$\begin{aligned}
\pi(x) &= \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x}+1} \\
&= \frac{1}{1+e^{-\beta_0-\beta_1 x}} \\
\pi(x)+\pi(x)e^{-\beta_0-\beta_1 x} &= 1 \\
e^{-\beta_0-\beta_1 x} &= \frac{1-\pi(x)}{\pi(x)} \\
\beta_0+\beta_1 x &= \ln\left(\frac{\pi(x)}{1-\pi(x)}\right).
\end{aligned}$$

For an example application, let's consider the American household income distribution from 2010. Roughly 9.1% of households made less than \$10,000 in 2010, while roughly 87.9% of households made less than \$100,000 in 2010. The median income was \$49,445, which implies that 50% of American households made less than this income level. So our $(x, \pi(x))$ data points are (10,000, 0.091), (49,445, 0.50) and (100,000, 0.879). We can write a matrix equation capturing these data in the logistic model.

$$\begin{bmatrix} 1 & 10,000 \\ 1 & 49,445 \\ 1 & 100,000 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \ln\left(\frac{0.091}{1-0.091}\right) \\ \ln\left(\frac{0.5}{1-0.5}\right) \\ \ln\left(\frac{0.879}{1-0.879}\right) \end{bmatrix}$$

$$A\beta = \pi.$$

Matlab will give us the projection here exactly as it has in the previous examples.
s

```
>> A = [1 10000; 1 49445; 1 10000];
>> pi = [log(0.091/(1-0.091)); log(0.5/(1-0.5)); log(0.879/(1-0.879))];
>> betaHat = inv(A'*A)*A'*pi

betaHat =

   -0.1996
    0.0000
```

The string of all zeros seems a little weird. We can ask Matlab to display more digits of the decimal using the `format` command.

```
>> format long
>> betaHat = inv(A'*A)*A'*pi

betaHat =

  -0.199617177625391
   0.000004037155984
```

So the unique logistic model that best fits the given data is

$$\pi(x) = \frac{e^{-0.1996+4\times 10^{-6}x}}{e^{-0.1996+4\times 10^{-6}x} + 1}.$$

Generlization

You might be guessing that we could use the same framework to perform other types of regression, and you'd be right. But it still might be surprising just how much we can get out of what we've developed so far. The exact same techniques will find the least-squares solution(s) for any model of the form

$$y = \beta_0 f_0(x) + \beta_1 f_1(x) + \ldots + \beta_n f_n(x),$$

where each f_i is some function. If we hadn't had so much experience with least-squares problems and their solutions, this claim might be quite literally incredible. But with our results in hand, we know that no matter what the functions $f(x)$, we can form a linear system of the form $A\beta = \mathbf{y}$ in which the entries of A depend on the functions $f_1, f_2, \ldots, f_n$ and the exact x values of the data points we've collected. But other these domestic differences, the process works in exactly the same way.

Chapter 5

Coordinates and principal component analysis

The sections in this chapter are geared at developing *principal component analysis* (PCA), a technique that breaks a large data set into various components and then rates the components as to their contribution to the overall trends in the data. As we saw with regression, the road towards a big tool, here PCA, is dotted with all sorts of interesting and useful ideas. We'll first encounter the concepts of a *basis* for a subspace. As an aside, we'll learn to characterize a subspace using its *dimension*, and we'll connect the dimension of the image of a matrix A to the matrix *rank* of A. We'll see that changing the basis in which we describe a vector can be particularly handy in a lot of applications. We'll develop a number of specific examples, including changing to a orthonormal basis and an eigenbasis.

5.1 Bases

Recall that a set of vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ is linearly independent if the homogeneous equation

$$c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_k\mathbf{v}_k = \mathbf{0}$$

has only the trivial solution $c_1 = c_2 = \ldots = c_k = 0$. Said another way, the set $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ is linearly independent if none of the vectors can be written as a linear combination of the others. While we had thought about this concept in the past in terms of the solution sets of linear systems, we can also connect it the newer concepts of vector (sub)spaces.

We say a set of vectors $\mathcal{B} = \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ forms a *basis* of a subspace H of a vector space V if $\mathcal{B}$ is a linearly independent set, and $H = span(\mathcal{B})$.

For a first example, let $H = \mathbb{R}^3$. The collection $\mathcal{B}$ defined by

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

both spans H and is linearly independent. So $\mathcal{B}$ is in fact a basis of H. Imagine that we were to remove one of the vectors from $\mathcal{B}$ by defining

$$\mathcal{C} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\}.$$

The new collection $\mathcal{C}$ is still linearly independent, but it does not span H. For instance, any vector with a nonzero third component is not a linear combination of the vectors from $\mathcal{C}$. Imagine now that we were to add another vector to $\mathcal{B}$ by defining

$$\mathcal{D} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

The collection $\mathcal{D}$ spans H, but the collection is not linearly independent, because

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \mathbf{0}.$$

This example shows us two fundamental ideas about a basis of a subspace H: take a vector away from the basis and the collection no longer spans H; add another vector to a basis, and the collection of vectors becomes linearly dependent.

Note that a basis is not unique, that is, there may be many bases for the same space. For instance, we can construct two different bases for $\mathbb{R}^3$.

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}, \quad \mathcal{C} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

While the particular vectors in a basis for a space H can differ from basis to basis, the number of vectors in the basis does not. We say that the *dimension* of a subspace H is the number of vectors in any basis of H. We define the subspace $H = \mathbf{0}$ to have dimension zero.

5.1.1 Bases of the image and kernel

We can connect the idea of basis to the fundamental subspaces of a matrix A, namely the image and kernel of A. Define

$$\begin{aligned} A &= \begin{bmatrix} 1 & 1 & 0 \\ 1 & 3 & -2 \\ 1 & 5 & -4 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \end{bmatrix}. \end{aligned}$$

Let's start investigating this matrix by row reducing the coefficient matrix A.

```
>> rref(A)

ans =

     1     0     1
     0     1    -1
     0     0     0
```

Let's think about what this means for vectors in the kernel of A. Defining $\mathbf{x} = [x_1, x_2, x_3]^T$, the first line reads $x_1 + x_3 = 0$, and the second line reads $x_2 - x_3 = 0$. So any vector in the kernel of A has the form

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -x_3 \\ x_3 \\ x_3 \end{bmatrix} = x_3 \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}.$$

Since x_3 is free as indicated by the RREF, we have $ker(A) = span([-1, 1, 1]^T)$. Since $[-1, 1, 1]^T$ spans the kernel, and since any collection containing only a single vector is linearly independent, we conclude that $\mathcal{B} = \{[-1, 1, 1]^T\}$ is a basis for the kernel of A. Here the kernel is 1-dimensional. In general, we can compute one basis vector of the kernel of A per free variable in the RREF of A using the same methodology we have here.

How about the image of A? We can start with the definition $im(A) = span(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$. Note that since the kernel of A is not just $\mathbf{0}$, the columns of A are linearly dependent, and so $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ does not form a basis for the image of A. Any vector in the span has the form

$$\mathbf{x} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + c_3\mathbf{v}_3.$$

But note that since $[-1, 1, 1]^T$ is in the kernel of A, we have $-\mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3 = \mathbf{0}$ so that $\mathbf{v}_3 = -\mathbf{v}_1 + \mathbf{v}_2$. We can use this information to rewrite the general form of an element in the image of A.

$$\begin{aligned} \mathbf{x} &= c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + c_3(-\mathbf{v}_1 + \mathbf{v}_2) \\ &= (c_1 - c_3)\mathbf{v}_1 + (c_2 + c_3)\mathbf{v}_2. \end{aligned}$$

Note what's happened here: we originally assumed that $\mathbf{x}$ was an element of the span of all three columns of A, but we've actually shown that $\mathbf{x}$ is in the span of just two of them. This holds exactly because the columns of A are not linearly independent. In effect, by taking into account the information we learned about the kernel of A, we have whittled down the columns of A until we have a linearly independent set that spans the image of A. Hence, we have constructed a basis for the image of A, namely $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2\}$. Note that $\mathbf{v}_1$ and $\mathbf{v}_2$ were pivot columns of the row reduced form of A. This leads us to a general observation: the collection of pivot columns of a matrix A form a basis for the image of A. So the dimension of the image of A is equal to the number of pivot columns in the RREF of A.

We can synthesize our results on bases for the kernel and image of an $m \times n$ matrix A to form an interesting theorem. Note that each column of the RREF of A is either a pivot column or represents a free variable. Since the dimension of the kernel is equal to the number of free variables and the dimension of the image is equal to the number of pivots and the total number of columns is n, we have

$$dim(im(A)) + dim(ker(A)) = n.$$

This is the so-called *rank-nullity* theorem. The *rank* of a matrix is simply the number of pivots, and the *nullity* is simply the number of free variables. So in words, the rank-nullity theorem says that the sum of the rank and the nullity of a matrix is equal to the number of columns in the matrix. We can use the rank-nullity theorem to add two more statements to our list of equivalent conditions for the invertibility of A. An $n \times n$ matrix A is invertible if and only if any of the following are true.

- There exists A^{-1} such that $A^{-1}A = AA^{-1} = I_n$.
- $rref(A) = I_n$.
- A has n pivots.
- The columns of A are linearly independent.
- The columns of A span $\mathbb{R}^n$.
- $\det(A) \neq 0$.
- $A\mathbf{v} = \mathbf{0}$ has only the trivial solution $\mathbf{v} = \mathbf{0}$.
- $\lambda = 0$ is *not* an eigenvalue of A.
- $ker(A) = \mathbf{0}$.
- $im(A) = \mathbb{R}^n$.
- $dim(im(A)) = n$.
- $dim(ker(A)) = 0$.

5.2 Coordinates

Given a basis $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ of a vector space V, we can define the *coordinates* of any vector $\mathbf{x}$ defined by

$$\mathbf{x} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_k\mathbf{v}_k$$

as

$$[\mathbf{x}]_{\mathcal{B}} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix}.$$

Since the basis $\mathcal{B}$ spans V, we know that $\mathbf{x}$ can be expressed as a linear combination of the basis elements, and so these coordinates are guaranteed to exist. We also know they must be unique, because $\mathcal{B}$ is linearly independent.

For a first example, consider the standard basis of $\mathbb{R}^3$

$$\mathcal{E} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}.$$

Then defining $\mathbf{x} = [x_1, x_2, x_3]^T$, we can write the $\mathcal{E}$-coordinates of $\mathbf{x}$.

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + x_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$
$$\Rightarrow [\mathbf{x}]_{\mathcal{E}} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

So the standard basis coordinates of a vector are just the vector itself. While this might not seem too interesting, we'll see that it is useful.

For a slightly more involved example, consider the basis

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

Then the coordinates $[\mathbf{x}]_{\mathcal{B}}$ of the vector $\mathbf{x} = [1, 2, 3]^T$ satisfy

$$c_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + c_2 \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + c_3 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$P_{\mathcal{E}\leftarrow\mathcal{B}}[\mathbf{x}]_{\mathcal{B}} = \mathbf{x} = [\mathbf{x}]_{\mathcal{E}}.$$

So finding the $\mathcal{B}$-coordinates of $\mathbf{x}$ is equivalent to solving a linear system. We can do this any number of ways. Perhaps the easiest is to use the inverse of $P_{\mathcal{E}\leftarrow\mathcal{B}}$.

```
>> PBtoE = [1,1,1;0,1,1;0,0,1];
>> Bcoords = inv(PBtoE)*[1;2;3]

Bcoords =

    -1
    -1
     3
```

So we have $[\mathbf{x}]_{\mathcal{B}} = [-1, -1, 3]^T$.

Notice that the matrix $P_{\mathcal{E}\leftarrow\mathcal{B}}$ changes the coordinates of $\mathbf{x}$ from basis $\mathcal{B}$ to the standard basis. Perhaps not surprisingly, we call $P_{\mathcal{E}\leftarrow\mathcal{B}}$ the *change of coordinates matrix* from $\mathcal{B}$ to $\mathcal{E}$. The columns of this matrix are simply the vectors of the basis $\mathcal{B}$ written in the standard basis $\mathcal{E}$. Since $P_{\mathcal{E}\leftarrow\mathcal{B}}$ is always invertible (*i.e.*, not just in the example we did above), we have another expression for the $\mathcal{B}$-coordinates of $\mathbf{x}$.

$$[\mathbf{x}]_{\mathcal{B}} = P_{\mathcal{E}\leftarrow\mathcal{B}}^{-1}\mathbf{x}.$$

Since the matrix $P_{\mathcal{E}\leftarrow\mathcal{B}}^{-1}$ acts to change the $\mathcal{E}$-coordinates of $\mathbf{x}$ to $\mathcal{B}$-coordinates, we can draw an interesting equivalence.

$$P_{\mathcal{E}\leftarrow\mathcal{B}}^{-1} = P_{\mathcal{E}\leftarrow\mathcal{B}}.$$

In words, the matrix that transforms $\mathcal{E}$-coordinates into $\mathcal{B}$-coordinates is the inverse of the matrix that transforms $\mathcal{B}$-coordinates into $\mathcal{E}$-coordinates. We can confirm this numerically using the quantities we defined in our example.

```
>> PEtoB = inv(PBtoE)

PEtoB =

     1    -1     0
```

```
     0     1    -1
     0     0     1

>> PEtoB*[1;2;3]

ans =

    -1
    -1
     3

>> PBtoE*[-1;-1;3]

ans =

     1
     2
     3
```

Imagine now that we have two different basis $\mathcal{B}$ and $\mathcal{C}$ of the same vector space V. We can use our knowledge change of coordinates between $\mathcal{B}$ and $\mathcal{E}$ and between $\mathcal{C}$ and $\mathcal{E}$ to build up some intuition about changing coordinates directly between $\mathcal{B}$ and $\mathcal{C}$. From our work above, we know that we can express $\mathbf{x}$ both in $\mathcal{B}$-coordinates and $\mathcal{C}$-coordinates.

$$\begin{aligned}\mathbf{x} &= P_{\mathcal{E}\leftarrow\mathcal{B}}[\mathbf{x}]_{\mathcal{B}}\\ \mathbf{x} &= P_{\mathcal{E}\leftarrow\mathcal{C}}[\mathbf{x}]_{\mathcal{C}}.\end{aligned}$$

We can use this fact to define a matrix $P_{\mathcal{C}\leftarrow\mathcal{B}}$ that changes $\mathcal{B}$-coordinates into $\mathcal{C}$-coordinates.

$$\begin{aligned}P_{\mathcal{E}\leftarrow\mathcal{B}}[\mathbf{x}]_{\mathcal{B}} &= P_{\mathcal{E}\leftarrow\mathcal{C}}[\mathbf{x}]_{\mathcal{C}}\\ (P^{-1}_{\mathcal{E}\leftarrow\mathcal{C}}P_{\mathcal{E}\leftarrow\mathcal{B}})[\mathbf{x}]_{\mathcal{B}} &= [\mathbf{x}]_{\mathcal{C}}.\end{aligned}$$

The action performed by the matrix $P_{\mathcal{C}\leftarrow\mathcal{B}} = P^{-1}_{\mathcal{E}\leftarrow\mathcal{C}}P_{\mathcal{E}\leftarrow\mathcal{B}} = P_{\mathcal{C}\leftarrow\mathcal{E}}P_{\mathcal{E}\leftarrow\mathcal{B}}$ on the $\mathcal{B}$-coordinates of $\mathbf{x}$ is composed of two parts. It first converts $\mathcal{B}$-coordinates to $\mathcal{E}$-coordinates using $P_{\mathcal{E}\leftarrow\mathcal{B}}$. (Remember by "first" we mean the right-most component, because matrix multiplication in the sense we use it moves from right to left, not left to right like English text.) It then converts the resulting $\mathcal{E}$-coordinates into $\mathcal{C}$-coordinates using $P_{\mathcal{C}\leftarrow\mathcal{E}} = P^{-1}_{\mathcal{E}\leftarrow\mathcal{C}}$.

Let's seen an example. Define

$$\mathcal{C} = \left\{ \begin{bmatrix}1\\0\\1\end{bmatrix}, \begin{bmatrix}2\\1\\2\end{bmatrix}, \begin{bmatrix}1\\1\\0\end{bmatrix} \right\}.$$

Then we can define coordinate change matrices for both $\mathcal{B}$ and $\mathcal{C}$ to the standard basis $\mathcal{E}$.

```
>> PBtoE = [1 1 1; 0 1 1; 0 0 1];
>> PCtoE = [1 2 1; 0 1 1; 1 2 0];
```

We can assemble these matrices to form coordinate change matrices to convert $\mathcal{B}$-coordinates to $\mathcal{C}$-coordinates and vice versa.

```
>> PCtoB = inv(PBtoE)*PCtoE

PCtoB =

     1     1     0
    -1    -1     1
     1     2     0

>> PBtoC = inv(PCtoE)*PBtoE

PBtoC =

     2     0    -1
    -1     0     1
     1     1     0
```

We can test the validity of these matrices with an example. Let's calculate the $\mathcal{B}$-coordinates and $\mathcal{C}$-coordinates of the vector $\mathbf{x} = [1, 2, 3]^T$ using the old methodology.

```
>> x = [1;2;3];
>> Bcoords = inv(PBtoE)*x

Bcoords =

    -1
    -1
     3

>> Ccoords = inv(PCtoE)*x

Ccoords =

    -5
     4
    -2
```

Using our matrices $P_{\mathcal{C}\leftarrow\mathcal{B}}$ and $P_{\mathcal{B}\leftarrow\mathcal{C}}$, we can confirm that if we convert directly from $\mathcal{B}$-coordinates to $\mathcal{C}$-coordinates and arrive at the correct coordinates.

```
>> Bcoords_check = PCtoB*Ccoords
```

```
Bcoords_check =

    -1
    -1
     3

>> Ccoords_check = PBtoC*Bcoords

Ccoords_check =

    -5
     4
    -2
```

5.3 Diagonalization

In this section, we'll connect two of the big ideas we've seen so far in the course: eigenthings and change of coordinates. While we didn't have the language of bases at the time, we saw in our investigation of dynamical systems that representing an initial condition $\mathbf{x}_0$ as a linear combination of the eigenvectors of the transition matrix allowed us to discover concepts like the dominant eigenpair. This was only possible because the eigenvectors formed a basis of $\mathbb{R}^n$, and so every initial condition $\mathbf{x}_0$ could be represented as a unique linear combination of the eigenvectors. Here we'll formalize these connections and bring in new ones. We'll use all of these tools in our discussion of the singular value decomposition and principal component analysis.

5.3.1 Eigenbases

Imagine that a $n \times n$ matrix A has n linearly independent eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ with associated distinct eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. We now know that these vectors form a basis of $\mathbb{R}^n$ as they are linearly independent and span the space. We call $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ an *eigenbasis* of $\mathbb{R}^n$. The matrix that changes coordinates from $\mathcal{V}$ to the standard basis $\mathcal{E}$ is defined by

$$P_{\mathcal{E}\leftarrow\mathcal{V}} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{bmatrix}.$$

We'll abbreviate this change of coordinate matrix simply as P where no confusion should occur.

We can capture the information the $\mathcal{V}$-coordinates of the image of all eigenvectors in a single matrix equation. Suppose we define

$$D = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix},$$

so that the eigenvalues of A are located along the main diagonal of D. We can stack the standard coordinates

$$\begin{aligned} AP &= \begin{bmatrix} A\mathbf{v}_1 & A\mathbf{v}_2 & \dots & A\mathbf{v}_n \end{bmatrix} \\ PD &= \begin{bmatrix} \lambda_1\mathbf{v}_1 & \lambda_2\mathbf{v}_2 & \dots & \lambda_n\mathbf{v}_n \end{bmatrix} \\ \Rightarrow AP &= PD \end{aligned}$$

Since P is invertible, we can conclude that $A = PDP^{-1}$, where D is a diagonal matrix. We say that a matrix A is *diagonalizable* if there exists an invertible matrix P and a diagonal matrix D such that $A = PDP^{-1}$. So our work above has shown that a $n \times n$ matrix is diagonalizable if and only if it has n linearly independent eigenvectors.

Now, determining whether a matrix has a collection of linearly independent eigenvectors can be a tough proposition. We'll see later in this section that in special cases we can conclude that such an eigenbasis does exist. But in most cases, we just need to compute the eigenvalues and check whether they're independent. Notice that given a matrix A, the matrices P and D are exactly the matrices returned by the command `eig(A)` in Matlab.

Example 15: Is the matrix

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

diagonalizable?

To conclude that A is diagonalizable, we need to confirm that the eigenvectors of A are linearly independent. Matlab gives

```
>> A = [1 2 0; 0 3 0; 2 -4 2];
>> [P,D] = eig(A)

P =

         0    0.4472    0.4082
         0         0    0.4082
    1.0000   -0.8944   -0.8165
```

```
D =

     2     0     0
     0     1     0
     0     0     3
```

To confirm that the columns of P are linearly independent, we need only row reduce P. Matlab can help with this, too.

```
>> rref(A)

ans =

     1     0     0
     0     1     0
     0     0     1
```

Since there are no free variables in the RREF, we conclude that the eigenvectors of A are linearly independent. So A is diagonalizable. Just to make ourselves feel better, we can check that $A = PDP^{-1}$.

```
>> P*D*inv(P)

ans =

     1     2     0
     0     3     0
     2    -4     2
```

Example 16: Is the matrix

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

diagonalizable?

Again, we need to check the linear independence of the eigenvectors of A. Matlab gives

```
>> A = [3 1; 0 3];
>> [P,D] = eig(A)

P =

    1.0000   -1.0000
         0    0.0000
```

```
D =

     3     0
     0     3
```

You might be able to tell by eye that the columns of P are not linearly independent. But to check more formally, let's row reduce P.

```
>> rref(P)

ans =

     1    -1
     0     0
```

Since there is a free variable in the RREF of P, we conclude that the columns of P are *not* linearly independent. Thus A is *not* diagonalizable.

In general it is not possible to tell *a priori* whether an arbitrary matrix is diagonalizable. But for specific classes of matrices we can make more headway.

5.3.2 Symmetric matrices

Recall that a matrix A is symmetric if $A^T = A$. We saw in Studio 10 that symmetric matrices have real eigenvalues. But their interesting properties do not stop there. Let's build some intuition with a particular example. Define the symmetric matrix

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

We can compute the eigenpairs of A using Matlab.

```
>> A = [1 2; 2 1];
>> [P,D] = eig(A)

P =

   -0.7071    0.7071
    0.7071    0.7071


D =

    -1     0
     0     3
```

Not only are the eigenvectors of A linearly independent, they're actually orthogonal! While this may seem like it worked out this way because of the simple example we chose, it is in fact a property of all symmetric matrices. To convince ourselves of this, imagine that $\mathbf{v}_1$ and $\mathbf{v}_2$ are two eigenvectors of A with distinct eigenvalues λ_1 and λ_2, respectively. Then

$$\begin{aligned}\lambda_1(\mathbf{v}_1 \circ \mathbf{v}_2) &= \lambda_1 \mathbf{v}_1^T \mathbf{v}_2 \\ &= (\lambda_1 \mathbf{v}_1)^T \mathbf{v}_2 \\ &= (A\mathbf{v}_1)^T \mathbf{v}_2 \\ &= \mathbf{v}_1^T A^T \mathbf{v}_2.\end{aligned}$$

Now, since A is assumed to be symmetric, we have $A^T = A$.

$$\begin{aligned}\lambda_1(\mathbf{v}_1 \circ \mathbf{v}_2) &= \mathbf{v}_1^T A \mathbf{v}_2 \\ &= \mathbf{v}_1^T \lambda_2 \mathbf{v}_2 \\ &= \lambda_2(\mathbf{v}_1 \circ \mathbf{v}_2).\end{aligned}$$

This implies that $(\lambda_1 - \lambda_2)(\mathbf{v}_1 \circ \mathbf{v}_2) = 0$, which can only be true if $\lambda_1 = \lambda_2$ or $\mathbf{v}_1 \circ \mathbf{v}_2 = 0$. Since we can assumed the eigenvalues here are distinct, we have $\lambda_1 \neq \lambda_2$. Therefore it must be the case that the eigenvectors are orthogonal. We say that a matrix A is *orthogonally diagonalizable* if there exists an invertible matrix P whose columns are mutually orthogonal and a diagonal matrix D such that $A = PDP^{-1}$. Our work has shown that any symmetric matrix is orthogonally diagonalizable. Note that if we scale each eigenvector $\mathbf{v}_i$ so that $\|\mathbf{v}_i\|^2 = 1$, then change of coordinate matrix P becomes orthogonal so that $P^{-1} = P^T$. Matlab does this scaling automatically.

```
>> P'*P

ans =

    1.0000         0
         0    1.0000
```

We can verify that in this case $A = PDP^T$.

```
>> P*D*P'

ans =

    1.0000    2.0000
    2.0000    1.0000
```

Note that diagonalization, whether orthogonal or not, only applies to square matrices. To deal with rectangular matrices, we will need a more powerful tool, namely the singular value decomposition.

5.4 Principal Component Analysis

In 1993, the heights and weights of 25,000 children between the ages of 0 and 18 were measured in Hong Kong. Such a large sample is sure to have interesting information buried within it. At the same time, drawing conclusions from such a large collection of data might seem daunting. We've seen one tool, namely least-squares, that would allow us to glean some information from the data. In this section, we'll learn about another: principal component analysis. Using this technique, we'll decompose the data (using an orthogonal change of coordinates) into ranked components which are mutually uncorrelated. This in effect will reveal the dominant trend in the data, followed by the next most prevalent trend assuming all linear dependence on the dominant trend has been removed, the third most important trend assuming all linear dependence on the first two has been removed, and so on. Moreover, we'll be able to quantify just how much of the total variation of the data each trend describes.

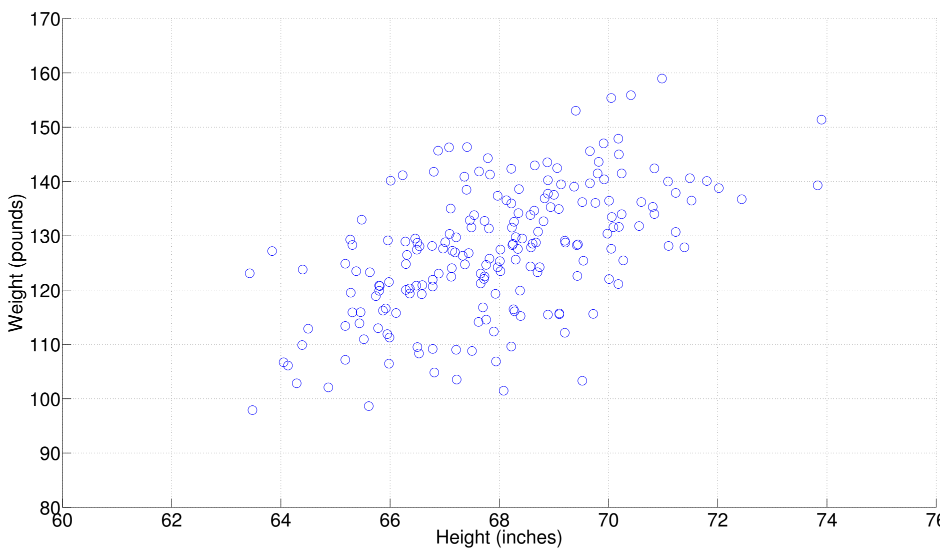


Figure 5.1: Height and weight data from 200 observations

5.4.1 Pre-processing data

For simplicity and tractability in our first example, we'll just deal with a 200 person subset of the data described above. As in statistics, we use the variable $N = 200$ to describe the number of observations we have. Imagine we bundle these observations in a matrix X,

$$X = \begin{bmatrix} h_1 & w_1 \\ h_2 & w_2 \\ \vdots & \vdots \\ h_{200} & w_{200} \end{bmatrix} = \begin{bmatrix} \mathbf{h} & \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{200} \end{bmatrix},$$

so that each row contains the height and weight data (in order) of a particular child. It will be helpful have notation describing both the rows and the columns of the observation matrix X; let $\mathbf{x}_i$ be row i, which describes the height and weight of a particular child, and let $\mathbf{h}$ and $\mathbf{w}$ be the columns containing *all* height and weight data, respectively. To help us get an intuitive sense of what the data look like, we can inspect Figure 5.1 in which the 200 data points are plotted.

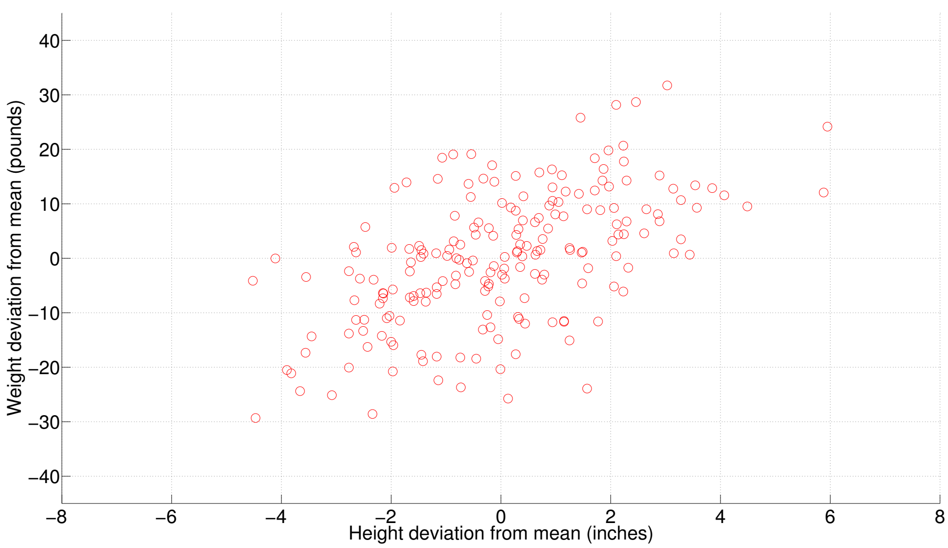


Figure 5.2: Demeaned height and weight data from 200 observations

Typically when we are exploring a data set, we're interested in its variability. This variability is opposed to the mean about which the variation occurs. To get at the variation of the data more directly, we need to de-mean the data by subtracting the mean height $\overline{h}$ from the height data $\mathbf{h}$ and the mean weight $\overline{w}$ from the weight data $\mathbf{w}$. The mean height in this data set is $\overline{h} = 67.95$ and the mean weight is $\overline{w} = 127.22$. (This can be computed easily by using `mean(X)`, with X defined as above, in Matlab.) We can define the *mean-deviation form* of the data by

$$B = \begin{bmatrix} h_1 - \overline{h} & w_1 - \overline{w} \\ h_2 - \overline{h} & w_2 - \overline{w} \\ \vdots & \vdots \\ h_{200} - \overline{h} & w_{200} - \overline{w} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{h}} & \hat{\mathbf{w}} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{x}}_1 \\ \hat{\mathbf{x}}_2 \\ \vdots \\ \hat{\mathbf{x}}_{200} \end{bmatrix},$$

(Try `B = X - ones(200,1)*mean(X)` in Matlab. The harder part is convincing yourself that `ones(200,1)*mean(X)` has $\overline{h}$ in the first column and $\overline{w}$ in the second column.) Let's see what these demeaned data look like in Figure 5.2. Notice that the "shape" of the data is the same as it was before, but now the data are horizontally and vertically centered about the origin.

5.4.2 Sample covariance matrix

We can get even more specific about the variation in the data. Recall that the sample covariance between two random variables $\mathbf{x}$ and $\mathbf{y}$ is defined by

$$\sigma_s(\mathbf{x}, \mathbf{y}) = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \overline{x})(y_i - \overline{y}).$$

We can use the notation we've developed here and some linear algebra to greatly simply this expression.

$$\begin{aligned} \sigma_s(\mathbf{x}, \mathbf{y}) &= \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \overline{x})(y_i - \overline{y}) \\ &= \frac{1}{N-1} \sum_{i=1}^{N} \hat{x}_i \hat{y}_i \\ &= \frac{1}{N-1} (\hat{\mathbf{x}} \circ \hat{\mathbf{y}}). \end{aligned}$$

So we conclude that covariance between two data sets is really a statement about the inner product of their mean deviation forms. We can efficiently package all the covariance information contained in the data set into the *covariance matrix* S in which $S(i,j)$ represents the covariance between variable i and variable j. The covariance of a variable with itself is known as its *variance.* With the height-weight data set, the covariance matrix is

$$S = \frac{1}{N-1} \begin{bmatrix} \hat{\mathbf{h}} \circ \hat{\mathbf{h}} & \hat{\mathbf{h}} \circ \hat{\mathbf{w}} \\ \hat{\mathbf{w}} \circ \hat{\mathbf{h}} & \hat{\mathbf{w}} \circ \hat{\mathbf{w}} \end{bmatrix}$$

But notice that we can repackage S into a very convenient form by using our knowledge of matrix multiplication.

$$\begin{aligned} S &= \frac{1}{N-1}\begin{bmatrix} \hat{\mathbf{h}} \\ \hat{\mathbf{w}} \end{bmatrix}\begin{bmatrix} \hat{\mathbf{h}} & \hat{\mathbf{w}} \end{bmatrix} \\ &= \frac{1}{N-1}B^T B. \end{aligned}$$

Computing this is a breeze in Matlab.

```
>> S = 1/(N-1)*B'*B

S =

    3.7650   12.9240
   12.9240  143.0645
```

Let's confirm that this result matches the intuition we've built around the figures we've seen so far. The variance of the height data is 3.77, while the variance of the weight data is 143.06. These are quite different, and so we should see such a difference reflected in Figure 5.2. Indeed we do, since the height data has a spread of only about 8 inches, while the weight data has a spread of roughly 60 pounds. We see this differing spreads reflected in the differences in the variances.

We can sum the individual variances of the variables to compute the *total variance*. Recall that the sum of the main diagonal elements of a matrix is known as its *trace*. Then the trace of the covariance matrix is the total variance of the data set. Here the total variance is $tr(S) = 3.77 + 143.06 = 146.8296$. In Matlab, we can write `trace(S)` to compute the total variance.

5.4.3 Eliminating covariance

The covariance between the height and weight data show that there is clearly some dependence between the two variables. While this is not surprising, it might be inconvenient in our analysis of the data. To wring the most information from the observations we have, we need *uncorrelated* variables, that is, variables which are linearly independent from one another. In this case, the covariance is zero. Mathematically, we would like to find a change of coordinates such that S becomes a diagonal matrix. We can bring all the tools we've manufactured in the previous sections to bear on this problem.

From our earlier work, we know that the basis that diagonalizes S is exactly its eigenbasis. Said another way, we can diagonalize S using a matrix P whose columns are the eigenvectors of S. This gives

$$S = PDP^{-1},$$

where D is a diagonal matrix containing the eigenvalues of S along its main diagonal. But when we first developed diagonalization, there was some doubt that as to whether an arbitrary matrix could be diagonalized. We can sidestep this concern here by noting that the covariance matrix S is symmetric:

$$S^T = \frac{1}{N-1}(B^T B)^T = \frac{1}{N-1}B^T(B^T)^T = \frac{1}{N-1}B^T B = S.$$

Hence, S is certainly diagonalizable, and moreover we know that the eigenvectors of S are mutually orthogonal.

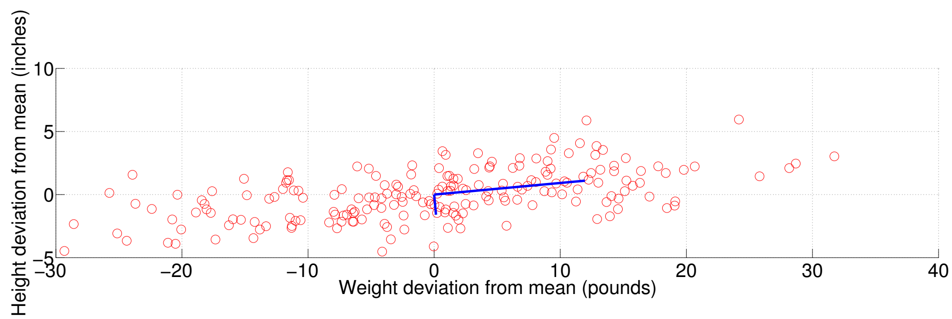


Figure 5.3: Principal components of the data set, with the length of each vector dictated by the standard deviation $\sqrt{\lambda_i}$. We have switched the axes and forced the aspect ratio to 1:1 in order to show the orthogonality of the principal components in a convenient form.

We call the eigenvectors of S the *principal components* of the data set, and we rank a principal component according to the relative size of its associated eigenvalue. For instance, given our data set, we have

```
>> [P,D] = eig(S)

P =

   -0.9958    0.0916
    0.0916    0.9958


D =

    2.5761         0
         0  144.2535
```

So the first principal component $\mathbf{u}_1 = [0.0916, 0.9958]^T$ represents the new variable

$$\hat{y}_1 = 0.0916\hat{h} + 0.9958\hat{w},$$

and the second principal component $\mathbf{u}_2 = [-0.9958, 0.0916]^T$represents the new variable

$$\hat{y}_2 = -0.9958\hat{h} + 0.0916\hat{w}.$$

We can see these principal components graphically in Figure 5.3.

We should interpret the first principal component as the dominant trend in the data. More specifically it shows that the ratio of weight to height in the children observed is roughly $0.9958/0.0916 = 10.8712$. Said another way, the dominant trend predict that for every inch a child grows, she will gain 10.8712 pounds. The second principal component is the most dominant trend after all linear dependence with the first principal component as been removed from each data point. Accordingly, the second principal component predicts something completely different than the first principal component: for every inch a child grows, she *loses* $0.0920 = 0.0916/0.9958$ pounds. Any data point can be represented by a linear combination of the uncorrelated trends represented by the principal components.

5.4.4 Capturing variance

Principal component analysis has one more important thing to tells us about how our data are distributed among the various principal components. First, let's make the observation that the total variance in the original data set, $tr(S)$, and the total variance in the data set after the change in coordinates, $tr(D)$, are equal:

```
>> trace(S)

ans =

  146.8296

>> trace(D)

ans =

  146.8296
```

This is no coincidence, as we can show. We'll have to use a fact about the trace of a product of matrices: $tr(AB) = tr(BA)$. Then note that

$$\begin{aligned} tr(S) &= tr(P(DP^{-1})) \\ &= tr((DP^{-1})P) \\ &= tr(D). \end{aligned}$$

So the trace of any two similar matrices are equal. (Note in particular that P does not have to be orthogonal as it is here.)

The percentage of the total variance captured by the new variable $\hat{y}_i$ is $\lambda_i/tr(S)$. For instance, the new variable $\hat{y}_1 = 0.0916\hat{h} + 0.9958\hat{w}$ represented by the first principal component corresponding to $\lambda_1 = 144.2535$ represents 98.25% of the total variance in the data. The new variable $\hat{y}_2 = -0.9958\hat{h} + 0.0916\hat{w}$ represented by the second principal component corresponding to $\lambda_2 = 2.5761$

captures the remaining 1.75% of the total variance. So we could effectively keep only the first principal component and keep much of the observed variance in the data. In this way, very high-dimensional data can be reduced to more manageable dimensionality by keeping the principal components that together account for a chosen percentage (*e.g.*, 90%, 99%, 99.99%) of the total variance.

Appendix A

Studio problems

A.1 Studio 1.1

(0) Write the augmented matrix, the row reduced echelon form matrix, and the general solutions of the linear systems below.

(a)

$$\begin{aligned} 3x_1 + 6x_2 &= -3 \\ 5x_1 + 7x_2 &= 10 \end{aligned}$$

(b)

$$\begin{aligned} x_1 - 5x_2 + 4x_3 &= -3 \\ 2x_1 - 7x_2 + 3x_3 &= -2 \\ -2x_1 + x_2 + 7x_3 &= -1 \end{aligned}$$

(c) Do the three lines $2x_1 + 3x_2 = -1$, $6x_1 + 5x_2 = 0$ and $2x_1 - 5x_2 = 7$ have a common point of intersection? If so, what is it? If not, how are you sure?

(1) You have 32 coins in denominations of pennies (1 cent each), nickels (5 cents each), and dimes (10 cents each) worth \$1.00 (100 cents) in total.

(a) Write a systems of a linear equations that describes this system.

(b) Solve the system of linear equations you found in the previous part of this question.

(c) How many coins of each type do you have?

(2) Find the interpolating polynomial $p(t) = a_0 + a_1 t + a_2 t^2$ for the data (1,6), (2,15), (3,28). That is, find a_0, a_1 and a_2 such that

$$
\begin{aligned}
a_0 + a_1(1) + a_2(1)^2 &= 6 \\
a_0 + a_1(2) + a_2(2)^2 &= 15 \\
a_0 + a_1(3) + a_2(3)^2 &= 28
\end{aligned}
$$

(3) (a) A system of linear equations with fewer equations than unknowns is sometimes called *underdetermined*. Can such a system have a unique solution?

(b) A system of linear equations with more equations than unknowns is sometimes called *overdetermined*. Can such a system be consistent?

(c) Suppose the coefficient matrix of a system of linear equations has a pivot in each row. Is the system consistent? Explain your answer.

(d) Suppose the coefficient matrix of a system of linear equations has a pivot in each column. Is the system consistent? Explain your answer.

(4) **(a)** Generate a random coefficient matrix with 3 row and 3 columns using the `rand(3,3)` command in Matlab. Does the matrix you generated correspond to linear system with a unique solution? How do you know?

(b) Repeat the procedure above 10 times. Do you notice a trend? Make a hypothesis about whether a random linear system of 3 equations in 3 unknowns has a unique solution.

(c) Is it possible that a linear system with 3 equations in 3 unknowns has no solutions? Stated another way, can a system of 3 linear equations in 3 variables be inconsistent? Does this fit with your hypothesis from above?

A.2 Studio 1.2

(0) **(a)** Does the vector $\mathbf{b} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ lie in the span of the vectors $\mathbf{v}_1 = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$, $\mathbf{v}_2 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$, $\mathbf{v}_3 = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}$?

(b) Determine whether there is a solution to the matrix equation

$$\begin{bmatrix} 72.0 & 56.0 & 8.0 \\ 74.0 & 69.0 & 74.0 \\ 95.0 & 13.0 & 11.0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 3.27 \\ 33.67 \\ 90.16 \end{bmatrix}.$$

(c) Determine whether the linear system

$$\begin{aligned} 46.0x_1 + 11.0x_2 + 56.0x_3 &= 47.22 \\ 36.0x_1 + 100.0x_2 + 30.0x_3 &= 98.42 \end{aligned}$$

is consistent.

(1) Let A be a $m \times n$ matrix. Prove that if $A\mathbf{x} = \mathbf{b}$, then $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$.

(2) Let A be an $m \times n$ matrix. Prove that if the matrix equation $A\mathbf{x} = \mathbf{b}$ has a solution $\mathbf{x} \in \mathbb{R}^n$ for every choice of $\mathbf{b} \in \mathbb{R}^m$, then A must have a pivot position in every row.

(3) Let A be a $m \times n$ matrix, and $\mathbf{u}$ and $\mathbf{v}$ be vectors in $\mathbb{R}^n$. Prove that $A(\mathbf{u} + \mathbf{v}) = A\mathbf{u} + A\mathbf{v}$.

(4)

$$B = \begin{bmatrix} 1 & 4 & 1 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & 7 \\ 2 & 9 & 5 & -7 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 4 & 1 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & 7 \\ 2 & 9 & 6 & -7 \end{bmatrix},$$

$$D = \begin{bmatrix} 1 & 4 & 1 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & 7 \end{bmatrix}, \quad E = \begin{bmatrix} 1 & 4 & 1 \\ 0 & 1 & 3 \\ 0 & 2 & 6 \\ 2 & 9 & 6 \end{bmatrix}$$

(a) Can every vector in $\mathbb{R}^4$ be written as a linear combination of the columns of B?

(b) Do the columns of C span $\mathbb{R}^4$?

(c) Does the matrix equation $D\mathbf{x} = \mathbf{b}$ have a solution for every $\mathbf{b} \in \mathbb{R}^3$?

(d) Do the columns of E span $\mathbf{R}^4$?

(5) Define

$$A = \begin{bmatrix} 10 & -7 & 1 & 4 & 6 \\ -8 & 4 & -6 & -10 & -3 \\ -7 & 11 & -5 & -1 & -8 \\ 3 & -1 & 10 & 12 & 12 \end{bmatrix}.$$

(a) Do the columns of A span $\mathbb{R}^4$?

(b) Find a column of A that can be deleted and yet have the remaining matrix columns still span $\mathbb{R}^4$.

(c) Can you delete more than one column and yet have the remaining matrix columns still span $\mathbb{R}^4$?

A.3 Studio 1.3

(0) Suppose weve been tracking 5 industries over the last several years, and constructed the following matrix describing how individual industries buy the output of other industries.

$$\mathbf{C} = \begin{bmatrix} 0.1 & 0.6 & 0.2 & 0 & 0.3 \\ 0.1 & 0.2 & 0 & 0.7 & 0.1 \\ 0.3 & 0 & 0.1 & 0.1 & 0.2 \\ 0.4 & 0.1 & 0.4 & 0.2 & 0.3 \\ 0.1 & 0.1 & 0.3 & 0 & 0.1 \end{bmatrix}$$

As weve seen before with these type of matrices, the matrix entry (i, j) is the percent of the output of industry j which is bought by industry i.

(a) Explain in words why the sums of each column must be equal to 1.

(b) Explain in words by the row sums need not be equal to 1.

(c) Suppose the industry i has a total output p_i, measured in dollars. How much should each industry produce in order for all industries to have their costs exactly balance their revenue?

(1) You and a friend have opened a boutique cupcake shop, and together you're trying to get a handle on your business total cost as a function of the total number of cupcakes produced. You currently have three data points: you know that the fixed cost of your business hovers around \$700 per week; in the first week you produced 100 cupcakes and your total cost was \$775; in the second week you produced 200 cupcakes and your total cost was \$800.

(a) Your cofounder obviously didn't have a Babson education: he thought that he had found a line that perfectly fit all three data points. Use linear algebraic reasoning to convince him that this can't be the case.

(b) After giving it another try, your cofounder claims that he has found several quadratic models $C(x) = c_0 + c_1x + c_2x^2$ which perfectly match the data. Either prove your cofounder right by producing an infinite family of models that perfectly predict the data, or disprove his claim by showing that there is a unique solution or no solution to this problem. (Not that simply producing a solution is not enough! How do you know there aren't more, for instance?)

(c) Motivated by your success with the quadratic modeling exercise, you've recently been wondering if there's a cubic total cost model $C(x) = c_0 + c_1x + c_2x^2 + c_3x^3$ that will even perfectly predict the data. Use linear algebraic reasoning to show how many of these solutions must exist.

(2) An economy has four sectors: agriculture, manufacturing, services and transportation. Agriculture sells 20% of its output to manufacturing, 30% to services, 30% to transportation and retains the rest. Manufacturing sells 35% of its output to agriculture, 35% to services, 20% to transportation and retains the rest. Services sells 10% of its output to agriculture, 20% to manufacturing, 20% to transportation and retains the rest. Transportation sells 20% of its output to agriculture, 30% to manufacturing, 30% to services and retains the rest.

(a) Construct the exchange matrix for this economy.

(b) Find a set of equilibrium prices for the economy if the value of transportation is \$10.00 per unit.

(c) The services sector launches a successful "eat farm fresh" campaign, and increases its share of the output from the agricultural sector to 40%, where as the share of agricultural production going to manufacturing falls to 10%. Construct the exchange matrix for this new economy.

(d) Find a set of equilibrium prices for this new economy if the value of transportation if still \$10.00 per unit. What effect has the service sectors campaign had on the equilibrium prices for the sectors of this economy? w

A.4 Studio 1.4

(0) Show that the columns of a matrix A are linearly dependent if A has more columns than rows.

(1) Give an example of a matrix A which has fewer columns than rows and whose columns are linearly dependent.

(2) Suppose A is a matrix which has the property that for any $\mathbf{b} \in \mathbb{R}^m$, there exists at *most* one solution $\mathbf{x} \in \mathbb{R}^n$. Explain why the columns of A must be linearly independent.

(3) Let's take another look at the 5 industry economic model we examined in the last studio. The matrix describing how these industries bought from and sold to one another was

$$\mathbf{C} = \begin{bmatrix} 0.1 & 0.6 & 0.2 & 0 & 0.3 \\ 0.1 & 0.2 & 0 & 0.7 & 0.1 \\ 0.3 & 0 & 0.1 & 0.1 & 0.2 \\ 0.4 & 0.1 & 0.4 & 0.2 & 0.3 \\ 0.1 & 0.1 & 0.3 & 0 & 0.1 \end{bmatrix}$$

(a) Are the columns of C linearly independent?

(b) How many, if any, production vectors $\mathbf{p}$ lead to a given total cost vector $\mathbf{c}$?

(c) Last time we were trying to solve the equation $(C - R)\mathbf{p} = \mathbf{0}$ in order to determine the total production levels $\mathbf{p}$ necessary for all industries to break even at the same time. We defined R to be the identity matrix with 5 rows and columns I_5. Are the columns of $C - R$ linearly independent?

(d) How many, if any, production vectors $\mathbf{p}$ lead to a break even scenario?

(e) Generate another cost matrix C, say for 3 industries. Are the columns of C linearly independent? Are the columns of $C - R$ linearly independent?

(f) Can you logically justify (meaning without Matlab) why in general the columns of C would be linearly independent but the columns of $C - R$ would not?

A.5 Studio 2.1

(0) Define matrices

$$A = \begin{bmatrix} 2.0 & 9.0 & 22.0 & 51.0 \\ 12.0 & 96.0 & 47.0 & 45.0 \\ 23.0 & 93.0 & 47.0 & 4.0 \end{bmatrix}, \quad B = \begin{bmatrix} 14.0 & 84.0 & 81.0 \\ 66.0 & 93.0 & 39.0 \\ 31.0 & 3.0 & 58.0 \end{bmatrix}, \quad C = \begin{bmatrix} 76.0 & 96.0 & 13.0 \\ 34.0 & 2.0 & 65.0 \\ 92.0 & 11.0 & 67.0 \\ 17.0 & 66.0 & 27.0 \end{bmatrix}$$

(a) Is AB defined? If so, what is its size?

(b) Is BA defined? If so, what is its size?

(c) Is AC defined? If so, what is its size?

(d) Is CA defined? If so, what is its size?

(e) Is ABC defined? If so, what is its size?

(f) Is CBA defined? If so, what is its size?

(1) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using RREF the total production necessary to satisfy the final demands represented by the vector

$$\mathbf{d} = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(2) If $AB = 0$ but neither A nor B is zero, we call A and B *zero divisors.* Note that there are no zero divisors in the real numbers, so this isn't something we may have encountered so far. Matrices can be zero divisors. For instance, if A and B are

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \end{bmatrix} \quad B = \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & 1 \\ -1 & -1 \end{bmatrix}$$

show that AB is the zero matrix, so that both A and B are zero divisors.

(3) **(a)** Unlike the real numbers, matrices do not typically commute, meaning that $AB \neq BA$ in general. To verify this statement, generate two random 3×3 matrices A and B and verify that they do not commute, that is, that $AB \neq BA$.

(b) Above we said that matrices do not commute *in general*. But some matrices do commute. Come up with an example of matrix that commutes with any square matrix A.

(4) In the real numbers, if $xy = xz$, then $y = z$, because we can cancel the x on each side. But with matrices, things are a little more complicated.

$$A = \begin{bmatrix} 3 & 4 & 5 & 6 \\ 7 & 8 & 9 & 10 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 3 & 4 & 5 \\ 6 & 7 & 10 & 11 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & 1 \\ -1 & -1 \end{bmatrix}$$

(a) Show that $AC = BC$, and note that $A \neq B$.

(b) Think about how zero divisors and cancellation are related. Use this to come up with another example of matrices A and B such that $AC = BC$ but $A \neq B$.

A.6 Studio 2.2

(0) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vector

$$\mathbf{d} = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(1) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 11 \\ 20 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(b) Confirm that the difference in the productions you found in the previous part is the first column of $(I - C)^{-1}$.

(c) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 21 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(d) Confirm that the difference in the productions you found in the previous part is the second column of $(I - C)^{-1}$.

(e) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 20 \\ 31 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(f) Confirm that the difference in the productions you found in the previous part is the third column of $(I - C)^{-1}$.

(g) Show that the additional production necessary to satisfy one additional unit of final demand for industry i is exactly the i^{th} column of $(I - C)^{-1}$.

(2) **(a)** Consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} k & 0.5 \\ 0.6 & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= C_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse C_k^{-1} exist?

(b) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & 0.5 \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= B_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse B_k^{-1} exist?

(c) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & \ell \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= A_{\ell,k} \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of ℓ and k does the inverse $A_{\ell,k}^{-1}$ exist?

(3) Generally, we can write a polynomial of degree $n-1$ as

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}.$$

In order for $p(x)$ to match our n data points exactly, it must be the case that $p(x_i) = y_i$ for every $i = 1, 2, \ldots, n$. But each of these equalities amounts to a linear combination of the coefficients $a_0, a_1, \ldots, a_{n-1}$. We can encode these linear combinations in a matrix equation:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$V\mathbf{a} = \mathbf{y}.$$

A.7 Studio 2.3

(0) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vector

$$\mathbf{d} = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(1) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 11 \\ 20 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(b) Confirm that the difference in the productions you found in the previous part is the first column of $(I - C)^{-1}$.

(c) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 21 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(d) Confirm that the difference in the productions you found in the previous part is the second column of $(I - C)^{-1}$.

(e) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 20 \\ 31 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

(f) Confirm that the difference in the productions you found in the previous part is the third column of $(I - C)^{-1}$.

(g) Show that the additional production necessary to satisfy one additional unit of final demand for industry i is exactly the i^{th} column of $(I - C)^{-1}$.

(2) **(a)** Consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} k & 0.5 \\ 0.6 & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= C_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse C_k^{-1} exist?

(b) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & 0.5 \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= B_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse B_k^{-1} exist?

(c) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & \ell \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= A_{\ell,k} \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of ℓ and k does the inverse $A_{\ell,k}^{-1}$ exist?

(3) Generally, we can write a polynomial of degree $n-1$ as

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}.$$

In order for $p(x)$ to match our n data points exactly, it must be the case that $p(x_i) = y_i$ for every $i = 1, 2, \ldots, n$. But each of these equalities amounts to a linear combination of the coefficients $a_0, a_1, \ldots, a_{n-1}$. We can encode these linear combinations in a matrix equation:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$V\mathbf{a} = \mathbf{y}.$$

A.8 Studio 3.2

(0) Suppose that your car rental company has 2 locations, L1 and L2, that together house 400 cars. Data indicate that on average 90% of the cars rented at L1 are returned to L1, and 80% of cars rented at L2 are returned to L2. (Thankfully, all cars are returned.)

(a) Write the transition matrix of this dynamical system.

(b) Assume that cars are rented and returned weekly, and that each week every car in both locations is rented. Calculate the distribution of cars after week 1, week 3, and week 10, given an even initial distribution of cars.

(c) Try several other initial conditions. Is the long term behavior of the car distribution the same?

(d) Calculate the eigenpairs of the transition matrix, and use them to predict the long term behavior of the system. What is the ratio of the number of cars at each location?

(e) Suppose now that you have 600 total cars. Is the long term behavior of the system the same as before? What is the ratio of the number of cars at each location?

(1) In one ecological model, the population of owls o_t and population of rats r_t (in thousands) at time t (in months) is related to the populations at time $t+1$ through the following matrix equation:

$$\begin{bmatrix} o_{t+1} \\ r_{t+1} \end{bmatrix} = \begin{bmatrix} .5 & .4 \\ -p & 1.1 \end{bmatrix} \begin{bmatrix} o_t \\ r_t \end{bmatrix}$$
$$\mathbf{x}_{t+1} = T\mathbf{x}_t.$$

The entry p is known as the *predation rate*.

(a) What is the meaning of each entry of matrix?

(b) What are the units of the predation rate p? In other words, how is p being measured in this model?

(c) Assume that $p = 0.1$. What happens to the populations in the long term?

(d) Assume that $p = 0.2$. What happens to the populations in the long term?

(2) One approach to conservation is through so called *stage-based population modeling*. Typically in these models, we consider on the female members of the species, because in biological terms males are often cheap; there are many males, and most of them are not going to reproduce anyway. For instance, female orcas have three stages: yearlings, juveniles, and mature. The yearly state transition matrix for the female orca population is

$$T = \begin{bmatrix} 0 & 0.0043 & 0.1132 \\ 0.9775 & 0.9111 & 0 \\ 0 & 0.0736 & 0.9534 \end{bmatrix}$$

(a) Interpret $T(2,1)$, $T(3,3)$, $T(1,3)$ and $T(3,2)$ in terms of the stage-based population model.

(b) What is the long term behavior of the population of female orcas? What is the ratio of juveniles to adults in the long term? What is the ratio of yearlings to adults in the long term?

(3) Imagine that we model our business based on two types of customers: one-time customers and repeat customers. These populations are disjoint, so that every current customer is either a one-time or a repeat, but no customer is both. Naturally (and hopefully), a one-time customer can become a repeat customer. From data you've gathered, you know that each month 40% of your one-time customers remain one-time customers. Around 10% of your repeat customers refer a new customer each month. You also know that on average 95% of repeat customers continue to buy your goods, and that on average 30% of one-time customers convert to repeat customers. (A common metric that I've heard is that if a customer has not bought something from you in 3 months then they are removed from the customer group.)

(a) Write a transition matrix for this model.

(b) High end industries often decide that repeat customers are the segment on which they want to focus. After all, the pool containing their potential clientele is small, so it makes sense to work hard to keep any customers you have. Cheap products often rely on the fact that they will have a large number of constantly changing one-time customers to support their business. Whatever your strategy, it's important make sure that you know what you're getting yourself into. In the long term, what will the ratio of one-time to repeat customers be for the business in this model?

(4) Suppose your business has a three tiered customer loyalty program. Every customer opting in to the program is assigned to either the bronze, silver or gold category. Customers do not need to progress through the levels in order; for instance, a customer can go directly from being a bronze category member to being a gold category member. Users can also slip; for instance, a customer can go from being a silver category member to being a bronze category member. Let b_t, s_t and g_t be number of customers in each of these populations in month t. Through data collection and analysis, you have proposed a model for the rates at which customers transition between these class from month to month. You can express your model using the following matrix equation:

$$\begin{bmatrix} b_{t+1} \\ s_{t+1} \\ g_{t+1} \end{bmatrix} = \begin{bmatrix} 0.4 & 0.6 & 0.1 \\ 0.6 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.8 \end{bmatrix} \begin{bmatrix} b_t \\ s_t \\ g_t \end{bmatrix}$$
$$\mathbf{x}_{t+1} = \mathbf{C}\mathbf{x}_t$$

(a) Describe in words the meaning of entry (3,1) of $\mathbf{C}$.

(b) Find the eigenvalues and associated eigenvectors of $\mathbf{C}$.

(c) In the long term, what percentage of the total number of customers enrolled in your loyalty program do you expect to be in the gold category?

A.9 Studio 3.3

(0) Define

$$C = \begin{bmatrix} 13 & -4 \\ -4 & 7 \end{bmatrix}.$$

(a) Use the characteristic equation to determine the eigenvalues of C.

(b) Using the eigenvalues you found in the previous part, compute the eigenvectors of the matrix.

(1) Consider an animal species that has two life stages: juvenile (up to 1 year old) and adult. Suppose that females give birth each year to an average of 1.6 female juveniles. Each year, 30% of the juveniles survive to become adults, and 80% of the adults survive.

(a) Construct a stage-based population model for this species. Develop a matrix T that links the populations in year t, $\mathbf{x}_t$ to the populations in year $t+1$, $\mathbf{x}_{t+1}$ via $\mathbf{x}_{t+1} = T\mathbf{x}_t$.

(b) What is the ratio of adults to juveniles in the long term?

(c) Suppose now that the average number of female offspring an adult bears each year is represented by a parameter k. Use the characteristic equation of the matrix T_k to find the dominant eigenpair of the system. What is the effect of k on the eigenvalues?

(d) What is the long term ratio of adults to in the preceding part? If the distribution depends on k, be sure to clearly indicate how.

(e) Now let ℓ represent the percentage of juveniles that survive to adulthood, and assume as before that each female adult bears on average 1.6 female offspring per year. Use the characteristic equation of the matrix T_ℓ to find the dominant eigenpair of the system. What is the affect of ℓ on the eigenvalues?

(f) What is the long term ratio of adults to in the preceding part? If the distribution depends on ℓ, be sure to clearly indicate how.

(2) Let $(\mathbf{v}, \lambda)$ be an eigenpair of an invertible matrix A. Show that $(\mathbf{v}, 1/\lambda)$ is an eigenpair of A^{-1}.

(3) Show that if λ is an eigenvalue of A, then λ is an eigenvalue of A^T. (Hint: consider how $A - \lambda I$ and $A^T - \lambda I$ are related.)

(a) Show that if λ is an eigenvalue of A^T, then λ is an eigenvalue of A.

(4) We say that a matrix T is *column stochastic* if it has only non-negative entries and its columns each sum to 1. For an example, our transition matrix from the bike rental example

$$T = \begin{bmatrix} 0.8 & 0.4 \\ 0.2 & 0.6 \end{bmatrix}$$

is column stochastic.

(a) Using the characteristic equation, compute the eigenvalues of T.

(b) Using the eigenvalues you found in the previous part, compute the associated eigenvectors of T.

(c) Show that any 2×2 column stochastic matrix T has $\lambda = 1$ as an eigenvalue. (Hint: let $T(1,1) = p$ and $T(2,2) = q$.)

(d) Show that for any column stochastic matrix T, the vector $[1, 1, \ldots, 1]^T$ is an eigenvector of T^T with eigenvalue $\lambda = 1$. Use a previous problem to conclude that $\lambda = 1$ is an eigenvalue of T. Comment on the relevance of this result to the long term distribution of a dynamical system with transition matrix T.

A.10 Studio 3.4

(0) Give the scaling factor r and the rotation angle ϕ for the following matrices:

$$A = \begin{bmatrix} \sqrt{3} & 1 \\ -1 & \sqrt{3} \end{bmatrix}, \quad B = \begin{bmatrix} 4 & -3 \\ 3 & 4 \end{bmatrix}, \quad C = \begin{bmatrix} 40 & -9 \\ 9 & 40 \end{bmatrix}.$$

(1) The population of spotted owls can be broken into three classes: juveniles, subadults and adults. These populations can be related to one another through the dynamical system

$$\begin{bmatrix} j_{t+1} \\ s_{t+1} \\ a_{t+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0.33 \\ 0.18 & 0 & 0 \\ 0 & 0.71 & 0.94 \end{bmatrix} \begin{bmatrix} j_t \\ s_t \\ a_t \end{bmatrix}$$

$$\mathbf{x}_{t+1} = T\mathbf{x}_t.$$

(a) What is the meaning of entry $T(3,2)$ in this model? What is the meaning of entry $T(1,3)$?

(b) What is the long term fate of the population of owls? Is the ratio of the populations constant in the long term?

(c) Now assume that through concerted conservation efforts, the percentage of juveniles surviving to subadulthood has been increased to 50% from the original model. What is the long term fate of the population of owls? Is the ratio of the populations constant in the long term?

(2) One approach to conservation is through so called *stage-based population modeling.* For an example, imagine that American bison females can be divided into calves (up to 1 year old), yearlings (1 to 2 years old), and adults. Suppose on average 42 female calves are born each year per 100 adult females. (Only adults produce offspring.) Each year, about 60% of the calves survive, 75% of the yearlings survive, and 95% of the adults survive. For $t \geq 0$, let $\mathbf{x}_t = [c_t \;\; y_t \;\; a_t]^T$ be the population vector representing the females in the herd.

(a) Construct a matrix A for the herd so that $\mathbf{x}_{t+1} = A\mathbf{x}_t$ for $t \geq 0$.

(b) Show that the buffalo herd is growing, determine the expected growth rate after many years, and give the expected number of calves and yearlings present per 100 adults.

(3) For a 2×2 matrix A, we can find an interesting relationship between the entries A and its eigenvalues. We'll need an additional piece of terminology to complete this formulation. The *trace* of a matrix A is the sum of the entries along the main diagonal, that is, the sum of all entries in positions (i, i). So for an arbitrary 2×2 matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the trace is $\tau = tr(A) = a+d$. Recall that the determinant of A is $\Delta = det(A) = ad - bc$.

(a) Show that the characteristic equation of A is $\lambda - \tau\lambda + \Delta$.

(b) Take a second to think about what this means: to understand the eigenvalues of a 2×2 matrix, we don't need to look at all 4 of the entries; we just need to consider two quantities, τ and Δ, that are related to the entries of the matrix. In essence, this cuts the complexity of the problem in half! (There's nothing to answer here. Just take a second to appreciate this fact.)

(c) Show that A has only real eigenvalues if and only if $\tau^2 \geq 4\Delta$.

(d) Show that if A is the transition matrix of a dynamical system, then the populations go to zero in the long term if $\tau < 0$ and $\Delta > 0$.

(4) Over the course of the next several parts, we'll show that any symmetric matrix has only real eigenvalues. This is known as the *spectral theorem.* This will also give us a chance to practice manipulating complex numbers.

(a) Let $\mathbf{v}$ be a vector with complex entries. Show that $\bar{\mathbf{v}}^T\mathbf{v}$ has only real entries.

(b) Now let $(\mathbf{v}, \lambda)$ be a (possibly complex) eigenpair of a symmetric matrix A. Show that $\bar{\mathbf{v}}^T A\mathbf{v} = \lambda\bar{\mathbf{v}}^T\mathbf{v}$.

(c) Let $(\mathbf{v}, \lambda)$ be the same eigenpair of the symmetric A as in the last part. Show that $\bar{\mathbf{v}}^T A\mathbf{v} = \bar{\lambda}\bar{\mathbf{v}}^T\mathbf{v}$. (Hint: you'll have to use the fact that A is symmetric here.)

(d) Combine the last two parts to show that λ must be real. (Hint: consider the two equivalent ways to write $\bar{\mathbf{v}}^T A\mathbf{v}$ and combine this with fact that $\bar{\mathbf{v}}^T\mathbf{v}$ is real.)

A.11 Studio 4.1

(0) Let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ be vectors in $\mathbb{R}^n$. Show that $span(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k)$ is a subspace of $\mathbb{R}^n$.

(1) Let P_n be the collection of all polynomials of degree n and smaller. Show that P_n is a vector space.

(2) Let $M_{m\times n}$ be the collection of all $m \times n$ matrices. Show that $M_{m\times n}$ is a vector space. (Here, the matrices are the "vectors" of the vector space.)

(3) Define

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

(a) Find a collection of vectors that span $\ker(A)$.

(b) Find a collection of vectors that span $\ker(B)$.

(4) Let H and K be two subspaces of a vector space of V. We define the *intersection* of H and K, denoted $H \cap K$, to be the collection of all vectors from V that are in both H and K.

(a) Show that $H \cap K$ is a subspace of V.

(b) Let's see a concrete example. Define

$$\mathbf{v}_1 = \begin{bmatrix} 5 \\ 3 \\ 8 \end{bmatrix}, \quad \mathbf{v}_2 = \begin{bmatrix} 1 \\ 3 \\ 4 \end{bmatrix}, \quad \mathbf{v}_3 = \begin{bmatrix} 2 \\ -1 \\ 5 \end{bmatrix}, \quad \mathbf{v}_4 = \begin{bmatrix} 0 \\ -12 \\ -28 \end{bmatrix}$$

Let $H = span(\mathbf{v}_1, \mathbf{v}_2)$ and $K = span(\mathbf{v}_3, \mathbf{v}_4)$. Then both H and K are planes in 3-dimensional space, both pass through the origin, and their intersection is a line. Write a short sentence explaining why the preceding statements are true.

(c) Now let's get more quantitative. If the intersection of H and K is a line, then it is the span of a single vector $\mathbf{w}$. Find this vector. (Hint: if $\mathbf{w}$ is in H, then $\mathbf{w} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2$, and if $\mathbf{w}$ is in K, then $\mathbf{w} = c_3\mathbf{v}_3 + c_4\mathbf{v}_4$.

(5) Let H and K be two subspaces of a vector space V. The *union* of H and K, denoted $H \cup K$, is collection of all vectors of V that are in H *or* K. Show that $H \cup K$ is not necessarily a subspace of V by given an example in which H and K are subspaces of $\mathbb{R}^2$.

(6) Let H and K be subspaces of a vector space V. The *sum* of H and K, denoted $H + K$ is the collection of vectors from V that can be written as the sum of a vector in H and a vector in K.

(a) Show that $H + K$ is a subspace of V.

(b) Show that H is a subspace of $H + K$ and K is a subspace of $H + K$.

(7) Let $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_5$ be the columns of matrix A, where

$$A = \begin{bmatrix} 5 & 1 & 2 & 2 & 0 \\ 3 & 3 & 2 & -1 & -12 \\ 8 & 4 & 4 & -5 & 12 \\ 2 & 1 & 1 & 0 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_4 \end{bmatrix}$$

(a) Are $\mathbf{a}_3$ and $\mathbf{a}_5$ in $im(B)$?

(b) Find a collection of vectors that spans $ker(B)$.

A.12 Studio 4.2

(0) Let $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^2$ and c be a scalar.

(a) Show that $\mathbf{u} \circ \mathbf{v} = \mathbf{v} \circ \mathbf{u}$.

(b) Show that $(\mathbf{u} + \mathbf{v}) \circ \mathbf{w} = \mathbf{u} \circ \mathbf{w} + \mathbf{v} \circ \mathbf{w}$.

(c) Show that $(c\mathbf{u}) \circ \mathbf{v} = c(\mathbf{u} \circ \mathbf{v})$.

(1) Define

$$\mathbf{u} = \begin{bmatrix} -3 \\ 7 \\ 4 \\ 0 \end{bmatrix} \quad \mathbf{v} = \begin{bmatrix} 1 \\ -8 \\ 15 \\ -7 \end{bmatrix}.$$

(a) Calculate $\|\mathbf{u}\|$.

(b) Calculate $\|\mathbf{v}\|$.

(c) Calculate $\|\mathbf{u} - \mathbf{v}\|$.

(d) Show that $\|\mathbf{u} - \mathbf{v}\| = \|\mathbf{v} - \mathbf{u}\|$.

(e) Show that $\|\mathbf{u} + \mathbf{v}\|^2 = \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2$ if $\mathbf{u}$ and $\mathbf{v}$ are orthogonal. (This is equivalent to the Pythagorean theorem in higher dimensions.)

(2) Let W be a subspace of $\mathbf{R}^n$ and let $W^\perp$ be the orthogonal complement of W. Show that the only vector in both W and $W^\perp$ is the zero vector.

(3) Let W be a subspace of $\mathbf{R}^n$ and let $W^\perp$ be the orthogonal complement of W. Show that $(W^\perp)^\perp = W$.

(4) Define A as found in `studio12.mat`. Find a collection of vectors that spans $im(A)^{\perp}$.

A.13 Studio 4.3

(0) Define

$$\mathbf{u} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 2 \\ 6 \\ 7 \\ -1 \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} -3 \\ 2 \\ -4 \\ -5 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$

(a) Find the projection of $\mathbf{y}$ onto $U = span(\mathbf{u})$ using both the explicit calculation at the beginning of the section, and the more general method used to complete the regression examples. Confirm that these approaches yield the same result.

(b) Find the projection of $\mathbf{y}$ onto $V = span(\mathbf{u}, \mathbf{v})$. Is the coefficient of $\mathbf{u}$ in this projection the same as the coefficient of $\mathbf{u}$ in the previous part?

(c) Find the projection of $\mathbf{y}$ onto $W = span(\mathbf{u}, \mathbf{v}, \mathbf{w})$.

(d) Find a matrix A such that $im(A) = W$. Confirm that $\mathbf{y} - \hat{\mathbf{y}}$ is a member of $W^{\perp} = \ker A^T$.

(1) Suppose we have (x, y) data points (-2,1), (-1,4), (0,3), (1,7), (2,4).

(a) Find the linear model that minimizes the sum of squared errors. Calculated the sum of squared errors by finding the norm $\|\mathbf{y} - \hat{\mathbf{y}}\|$.

(b) Find the quadratic model that minimizes the sum of squared errors. Calculated the sum of squared errors. Is it larger or smaller than the SSE in the previous part?

(c) Find the cubic model that minimizes the sum of squared errors. Calculated the sum of squared errors. Is it larger or smaller than the SSE in the previous part?

(d) Find the quartic model that minimizes the sum of squared errors. Calculated the sum of squared errors. What does your SSE mean in this case?

(2) Sometimes a single independent variable isn't enough to create a dependable model of a given system. For an example, imagine that you run a small ice cream shop on the coast of Maine. There are two main drivers for your sales: daily temperature and median customer income. Suppose we have a model with two independent variables u, representing the average daily temperature in July in your town, and v, representing the median income of customers who purchased from you in July. You've been collecting data over several years. The results can be seen in Table B.1

Year	**Total Sales**	**Average Temp.**	**Median Income**
2009	27.93	86.92	30.11
2010	28.29	88.51	31.48
2011	29.70	88.01	32.03
2012	31.09	87.05	33.34
2013	33.11	89.15	34.45

Table A.1: Data collected for the total sales (thousands of dollars), average temperate (degrees Fahrenheit), and median household income (thousands of dollars) for July of the indicated year

(a) Find the best multlinear model $s(u, v) = \beta_0 + \beta_1 u + \beta_2 v$ for the given data?

(b) We could also allow for the variables u and v to interact multiplicatively through the model $s(u, v) = \beta_0 + \beta_1 u + \beta_2 v + \beta_3 uv$. Find the best model of this form for the given data.

(c) An even more general model might be $s(u, v) = \beta_0 + \beta_1 u + \beta_2 v + \beta_3 uv + \beta_4 u^2 + \beta_5 v^2$. Find the best model of this form for the given data.

(3) One of the tools used in data mining is logistic regression, which takes a collection of observations about certain probabilities and attempts to construct the underlying cumulative density function. The logistic function in this case is

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x}+1},$$

where β_0 and β_1 are the parameters to be estimated. Notice that $\pi(x)$ is between 0 and 1 for every x. Moreover, we see that $\pi(x) \to 1$ as $x \to \infty$ and $\pi(x) \to 0$ as $x \to -\infty$. So $\pi(x)$ seems like a pretty good candidate for a CDF. Then $\pi(x)$ is the probability that some random variable X has value less than or equal to x.

(a) For an example, let's turn our attention to the grade distribution in a typical core foundation class. The median grade should be a B-, which equations to a 2.6 grade points. In my classes, roughly 15% of students receive an A- or better, which equates to 3.6 grade points or higher. Roughly 5% of students fail the course, which equates to 1 grade point or lower. Find the best logistic model for the underlying cumulative distribution function.

(b) According to your model, what percentage of students earn a C or better?

(c) According to your model, how many grade points should a student earn to be in the top 25% of the class?

A.14 Studio 5.1

(0) Is the collection

$$\left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \end{bmatrix} \right\}$$

a basis for $\mathbb{R}^2$?

(a) Is the collection

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

a basis for $\mathbb{R}^4$?

(1) Define A and B by

$$A = \begin{bmatrix} 1 & 6 & 16 & -40 \\ 2 & 5 & 11 & -31 \\ 3 & 4 & 6 & -22 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -2 \end{bmatrix}.$$

(a) Find a basis for the kernel of A.

(b) Find a basis for the image of A.

(c) What are the dimensions of the kernel and image of A?

(d) Find a basis for the kernel of B.

(e) Find a basis for the image of B.

(f) What are the dimensions of the kernel and image of B?

(2) Let $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ be a set of vectors in $\mathbb{R}^n$.
(a) Argue that if $k > n$, then $\mathcal{B}$ cannot be a basis for $\mathbb{R}^n$.

(b) Argue that if $k < n$, then $\mathcal{B}$ cannot be a basis for $\mathbb{R}^n$.

(c) Conclude that any basis of $\mathbb{R}^n$ must have exactly n elements. (There's nothing to do here other than recognize that the previous two parts directly show this fact.)

(3) The standard basis of $\mathbb{R}^n$ is the collection $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$, where $\mathbf{e}_i$ has a 1 in component i and zero in all other components. We can represent a vector $\mathbf{x}$ as a natural linear combination of these basis elements. For instance, if $\mathbf{x}$ is in $\mathbb{R}^2$, we have

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= x_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= x_1\mathbf{e}_1 + x_2\mathbf{e}_2.\end{aligned}$$

We can extend the idea of the standard basis to other vector spaces.

(a) Let H be the vector space of 2×2 matrices. Come up with a best guess as to the standard basis of H. Show that the collection of elements that you propose is in fact a basis.

(b) Let P_n be the vector space of polynomials of degree at most n. Come up with a best guess as to the standard basis of P_n. Show that the collection of elements that you propose is in fact a basis.

(4) Recall that a $n \times n$ matrix A is symmetric if $A(i,j) = A(j,i)$ for all $1 \leq i, j \leq n$.

(a) Find a basis for the vector space of 2×2 symmetric matrices.

(b) Find a basis for the vector space of 3×3 symmetric matrices.

(c) How would your previous answers generalize to the vector space of $n \times n$ matrices?

A.15 Studio 5.2

(0) Define

$$\mathcal{B} = \left\{ \begin{bmatrix}1\\1\\1\\1\end{bmatrix}, \begin{bmatrix}1\\2\\3\\4\end{bmatrix}, \begin{bmatrix}0\\0\\1\\0\end{bmatrix}, \begin{bmatrix}1\\0\\0\\1\end{bmatrix} \right\}, \quad \mathcal{C} = \left\{ \begin{bmatrix}1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}1\\1\\0\\0\end{bmatrix}, \begin{bmatrix}1\\1\\1\\0\end{bmatrix}, \begin{bmatrix}1\\1\\1\\1\end{bmatrix} \right\}.$$

(a) Verify that $\mathcal{B}$ and $\mathcal{C}$ are bases for $\mathbb{R}^4$.

(b) What are the $\mathcal{B}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$?

(c) Construct a matrix that changes coordinates from the standard basis $\mathcal{E}$ of $\mathbb{R}^4$ to the basis $\mathcal{B}$.

(d) What are the $\mathcal{C}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$?

(e) Construct a matrix that changes coordinates from the standard basis $\mathcal{E}$ of $\mathbb{R}^4$ to the basis $\mathcal{C}$.

(f) Construct a matrix that changes coordinates from the basis $\mathcal{B}$ to basis $\mathcal{C}$.

(g) Verify that the matrix you constructed in the previous subproblem coverts the $\mathcal{B}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$ to the $\mathcal{C}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$.

(h) Construct a matrix that changes coordinates from the basis $\mathcal{C}$ to basis $\mathcal{B}$.

(i) Verify that the matrix you constructed in the previous subproblem coverts the $\mathcal{B}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$ to the $\mathcal{C}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$.

(1) We'll show that the mapping of $\mathbf{x}$ to its coordinates in a given basis $\mathcal{B}$ is a linear transformation. Any linear transformation f satisfies two properties: $f(x+y) = f(x) + f(y)$ and $f(cx) = cf(x)$ for any scalar c.

(a) Let $\mathbf{u}$ and $\mathbf{v}$ be two vectors in a vector space V with basis $\mathcal{B}$. Show that $[\mathbf{u}+\mathbf{v}]_{\mathcal{B}} = [\mathbf{u}]_{\mathcal{B}} + [\mathbf{v}]_{\mathcal{B}}$.

(b) Let $\mathbf{u}$ a vector in a vector space V with basis $\mathcal{B}$. Show that $[c\mathbf{u}]_{\mathcal{B}} = c[\mathbf{u}]_{\mathcal{B}}$.

(2) Define

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}, \quad \mathcal{C} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

Label the vectors of $\mathcal{B}$ in order as $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$, respectively. Similarly, label the vectors of $\mathcal{C}$ in order as $\mathbf{c}_1$, $\mathbf{c}_2$ and $\mathbf{c}_3$, respectively. We can think about the change of coordinates matrix $P_{\mathcal{B}\to\mathcal{C}}$ in a different way than the on presented in the text. In this problem, we'll walk through that process. Define $\mathbf{x} = [4, 5, 6]^T$.

(a) Write $\mathbf{x}$ as a linear combination of the vectors $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$.

(b) Write each vector in $\mathcal{B}$ as a linear combination of the vectors $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$.

(c) Substitute your expressions for $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ in terms of $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$ into your expression for $\mathbf{x}$ in terms of $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$. (You should now have an expression for $\mathbf{x}$ in terms of $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$.) Regroup like terms and confirm that the $\mathcal{C}$-coordinates of $\mathbf{x}$ are the same here as you calculated above.

(d) Take a second to note that we've changed from $\mathcal{B}$-coordinates to $\mathcal{C}$-coordinates by writing each vector of $\mathcal{B}$ in terms of a linear combination of the vectors in $\mathcal{C}$. (There's nothing to do here but make this realization.)

(e) Confirm that numerically

$$P_{\mathcal{B}\to\mathcal{C}} = \begin{bmatrix} [\mathbf{b}_1]_\mathcal{C} & [\mathbf{b}_2]_\mathcal{C} & [\mathbf{b}_3]_\mathcal{C} \end{bmatrix}.$$

(f) Argue why the identity in the preceding part must hold, and generalize this idea.

A.16 Studio 5.3

(0) Define matrices A, B, C and E as found in `studio16.mat`. Determine whether each matrix is diagonalizable, orthogonally diagonalizable or neither. If diagonalizable, confirm that the eigenvectors are linearly independent. If orthogonally diagonalizable, confirm that the eigenvectors are orthogonal.

(1) We say two matrices A and B are *similar* if there exists a matrix P such that $A = PBP^{-1}$.

(a) Show that matrix A from `studio16.mat` is similar to `D = diag([6,5,4])`. (Here D is the diagonal matrix with entries 6, 5, 4 in order along its main diagonal.)

(b) Show that matrix F from `studio16.mat` is similar to `D = diag([6,5,4])`.

(c) Show that matrices A and F are similar.

(2) Let A be a symmetric $n \times n$ matrix.
(a) Show that A^2 is symmetric.

(b) A symmetric matrix A such that $A^2 = A$ is known as a *projection matrix.* Let $\mathbf{y} \in \mathbb{R}^n$, and define $\hat{\mathbf{y}} = A\mathbf{y}$. Show that $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to $\hat{\mathbf{y}}$.

(c) Explain why the previous part shows that any vector in $\mathbb{R}^n$ is a linear combination of a vector in the image of A and a vector in the orthogonal complement of the image of A?

(3) Show that if A is diagonalizable and invertible, then A^{-1} is, too.

(4) Construct a 2×2 matrix that is invertible but not diagonalizable.

(5) Construct a 2×2 matrix that is diagonalizable but not invertible.

(6) Show that if A is invertible and orthogonally diagonalizable, then A^{-1} is, too.

A.17 Studio 5.4

(0) **(a)** Given a matrix X, explain by the Matlab command `B = X - ones(N,1)*mean(X)` returns a matrix B whose columns each have mean zero.

(b) Suppose that $(\mathbf{v}_1, \lambda_1)$ and $(\mathbf{v}_2, \lambda_2)$ are two distinct eigenpairs of a matrix $A^T A$. Show that $A\mathbf{v}_1$ and $A\mathbf{v}_2$ are orthogonal.

(1) Check out the data in `pca_salary.csv`. The Matlab variable `textdata` contains the column headers describing the columns of the matrix `data` which contains observations relating CEO age to CEO pay.

(a) Use PCA to determine the first principal component and second principal components and their associated eigenvalues.

(b) Write an interpretation of the first and second principal components.

(c) How many principal components must you include in order to capture 90% of the total variance?

(2) Check out the data in `pca_temp.csv`. The Matlab variable `textdata` contains the column headers describing the columns of the matrix `data` which contains observations relating latitude, longitude and average January temperature over a 30 year time period.

(a) Use Google to locate the mean longitude and latitude of this data set on a map. Longitude becomes more negative as you move west in this data set, and latitude becomes more positive as you move north in this data set.

(b) Use PCA to determine the first principal component and second principal components and their associated eigenvalues.

(c) Write an interpretation of the first and second principal components.

(d) How many principal components must you include in order to capture 90% of the total variance?

(3) Check out the data in `pca_colleges.csv`. The Matlab variable `textdata` contains the column headers describing the columns of the matrix `data` which contains observations a number of attributes of colleges, including acceptance rate, average SAT score, and cost per year.

(a) Use PCA to determine the first principal component and second principal components and their associated eigenvalues.

(b) Write an interpretation of the first principal component.

(c) How many principal components must you include in order to capture 90% of the total variance?

Appendix B

Studio solutions

B.1 Studio 1.1 solutions

(0) Write the augmented matrix, the row reduced echelon form matrix, and the general solutions of the linear systems below.

(a)

$$
\begin{aligned}
3x_1 + 6x_2 &= -3 \\
5x_1 + 7x_2 &= 10
\end{aligned}
$$

The augmented matrix is given by

```
EDU>> [3 6 -3; 5 7 10]

ans =

     3     6    -3
     5     7    10
```

The RREF of the augmented matrix is given by

```
EDU>> rref([3 6 -3; 5 7 10])

ans =

     1     0     9
     0     1    -5
```

Hence, $x_1 = 9$ and $x_2 = -5$ is the unique solution to this linear system.

(b)

$$
\begin{aligned}
x_1 - 5x_2 + 4x_3 &= -3 \\
2x_1 - 7x_2 + 3x_3 &= -2 \\
-2x_1 + x_2 + 7x_3 &= -1
\end{aligned}
$$

The augmented matrix is given by

```
EDU>> [1 -5 4 -3; 2 -7 3 -2; -2 1 7 -1]

ans =

     1    -5     4    -3
     2    -7     3    -2
    -2     1     7    -1
```

The RREF of the augmented matrix is given by

```
EDU>> rref([1 -5 4 -3; 2 -7 3 -2; -2 1 7 -1])
```

```
ans =

    1.0000         0   -4.3333         0
         0    1.0000   -1.6667         0
         0         0         0    1.0000
```

Because the right-most column is a pivot column, the system has no solution. Notice that there is a free variable and yet the system is not consistent.

(c) Do the three lines $2x_1+3x_2=-1$, $6x_1+5x_2=0$ and $2x_1-5x_2=7$ have a common point of intersection? If so, what is it? If not, how are you sure?

These three lines define a system of linear equations who augmented matrix is

```
EDU>> [2 3 -1; 6 5 0; 2 -5 7]

ans =

     2     3    -1
     6     5     0
     2    -5     7
```

An intersection of these three lines is a pair (x_1,x_2) which satisfies all three equations simultaneously, and this is exactly the same as a solution to the linear system whose augmented matrix we just described. We can therefore compute the RREF of the augmented matrix to determine whether solutions exist.

```
EDU>> rref([2 3 -1; 6 5 0; 2 -5 7])

ans =

     1     0     0
     0     1     0
     0     0     1
```

Since the right-most column is a pivot column, the linear system has no solutions. Therefore the three lines do not intersect.

(1) You have 32 coins in denominations of pennies (1 cent each), nickels (5 cents each), and dimes (10 cents each) worth \$1.00 (100 cents) in total.

(a) Write a systems of a linear equations that describes this system.

Let p,n, and d be the number of pennies, nickels and dimes, respectively. Our system of linear equations is given by

$$\begin{aligned} p+n+d &= 32 \\ p+5n+10d &= 100 \end{aligned}$$

(b) Solve the system of linear equations you found in the previous part of this question.

The RREF of the augmented matrix of this system is

```
EDU>> rref([1 1 1 32; 1 5 10 100])

ans =

    1.0000         0   -1.2500   15.0000
         0    1.0000    2.2500   17.0000
```

So d is a free variable in the linear algebraic sense.

(c) How many coins of each type do you have?

Since d represents a physical quantity, we must take into consideration what values d can actually assume. For instance, $d \geq 0$, because we can't have a negative number of dimes. Looking at the first row in the RREF we have $p = 15 + 1.25d$. If $d = 1$, then we must have $p = 16.25$ pennies. But this can't be the case. You can convince yourself that d must be a multiple of 4. Looking at the second row in the RREF, we have $n = 17 - 2.25d$. If $d \geq 8$, then $n < 0$, which is impossible, too. By using our knowledge about the context of the mathematical model we've made, we've determined that the free variable d can only assume two values, $d = 0$ and $d = 4$, in order for our solution to make sense.

(2) Find the interpolating polynomial $p(t) = a_0 + a_1 t + a_2 t^2$ for the data (1,6), (2,15), (3,28). That is, find a_0, a_1 and a_2 such that

$$\begin{aligned} a_0 + a_1(1) + a_2(1)^2 &= 6 \\ a_0 + a_1(2) + a_2(2)^2 &= 15 \\ a_0 + a_1(3) + a_2(3)^2 &= 28 \end{aligned}$$

First, notice that we want to solve for a_0, a_1, a_2. The augmented matrix of this linear system is

```
EDU>> [1 1 1 6; 1 2 4 15; 1 3 9 28]

ans =

     1     1     1     6
     1     2     4    15
     1     3     9    28
```

The RREF of the augmented matrix is

```
EDU>> rref([1 1 1 6; 1 2 4 15; 1 3 9 28])

ans =

     1     0     0     1
     0     1     0     3
     0     0     1     2
```

So the solution to the linear system is $a_0 = 1$, $a_1 = 3$ and $a_2 = 2$. So there is a unique interpolating polynomial $p(t) = 1 + 3t + 2t^2$ that passes through points (1,6), (2,15), (3,28).

(3) **(a)** A system of linear equations with fewer equations than unknowns is sometimes called *underdetermined*. Can such a system have a unique solution?

The number of pivots in a matrix is at most the number of columns and at most the number of rows. If the there are fewer equations than unknowns, then there are fewer rows than columns. Hence, there are fewer pivots than there are columns. Then at least one column must represent a free variable. If the right-most column is a pivot column, then the linear system has no solution. Otherwise, the linear system has infinitely many solutions. So no, such a system cannot have a unique solution.

Example of infinitely many solutions: The augmented matrix

```
EDU>> [1 1 1 1; 2 3 4 5]

ans =

     1     1     1     1
     2     3     4     5
```

represents a linear system of 2 equations in 3 unknowns. The RREF is

```
EDU>> rref([1 1 1 1; 2 3 4 5])

ans =

     1     0    -1    -2
     0     1     2     3
```

Notice the free variable and that the right-most column is not a pivot column.

Example of no solutions: The augmented matrix

```
EDU>> [1 2 3 1; 2 4 6 5]

ans =

     1     2     3     1
     2     4     6     5
```

represents a linear system of 2 equations in 3 unknowns. The RREF is

```
EDU>> rref([1 2 3 1; 2 4 6 5])

ans =

     1     2     3     0
     0     0     0     1
```

Notice the free variables, and also that the right-most column is a pivot column. There is no solution despite the presence of the free variables.

(b) A system of linear equations with more equations than unknowns is sometimes called *overdetermined*. Can such a system be consistent?

The number of pivots in a matrix is at most the number of columns and at most the number of rows. If the there are fewer unknowns than equations, then there are fewer columns than rows. Hence, there are fewer pivots than there are rows. If the right-most column is a pivot column, then there is no solution. But if the right-most column is not a pivot column, then is there is a unique solution. So, depending on the contents of the right side of the linear system, it may be possible for it to be consistent.

Example of no solution: $x_1 + 2x_2 = 3$, $2x_1 + 4x_2 = 10$, $5x_1 + 10x_2 = 30$:

```
EDU>> rref([1 1 3; 2 4 10; 5 10 30])

ans =

     1     0     0
     0     1     0
     0     0     1
```

Example of a unique solution: $x_1+2x_2 = 3$, $2x_1+4x_2 = 10$, $5x_1+10x_2 = 25$:

```
EDU>> rref([1 1 3; 2 4 10; 5 10 25])

ans =

     1     0     1
     0     1     2
     0     0     0
```

(c) Suppose the coefficient matrix of a system of linear equations has a pivot in each row. Is the system consistent? Explain your answer.

Yes, the system is consistent. The only way the system could be inconsistent is if the right-most column of the *augmented* matrix were a pivot column. This would imply that there is a row of all zeros in the RREF of the *coefficient* matrix. Since this is not the case, we can't possibly arrive at the condition $0 = 1$ in the RREF of the augmented matrix. Hence, for any choice of the right-hand side of the equation, we are guaranteed a solution.

(d) Suppose the coefficient matrix of a system of linear equations has a pivot in each column. Is the system consistent? Explain your answer.

The system may or may not be consistent. Imagine a situation in which there are many equations, but only a few unknowns. Then by our work in part (b), it may be the case that there is an inconsistency, even if there is a pivot in every column of the coefficient matrix.

(4) **(a)** Generate a random coefficient matrix with 3 row and 3 columns using the `rand(3,3)` command in Matlab. Does the matrix you generated correspond to linear system with a unique solution? How do you know?

We run verb+rref(rand(3,3))+ and observe the results. The answer is that yes, each of them should correspond to a system with a unique solution. There should be a pivot in every row of the RREF of the coefficient matrix, and so the system must be consistent. There should also be a pivot in every column of the RREF of the coefficient matrix, and so the system has no free variables. These together imply that there is a unique solution.

(b) Repeat the procedure above 10 times. Do you notice a trend? Make a hypothesis about whether a random linear system of 3 equations in 3 unknowns has a unique solution.

All 10 trials should return exactly the same results. This seems to imply that most systems of 3 equations in 3 unknowns have a unique solution.

(c) Is it possible that a linear system with 3 equations in 3 unknowns has no solutions? Stated another way, can a system of 3 linear equations in 3 variables be inconsistent? Does this fit with your hypothesis from above?

Definitely. For an example, try $x_1 + 2x_2 + 3x_3 = 1$, $x_1 + 2x_2 + 3x_3 = 2$, $x_1 + 2x_2 + 3x_3 = 3$. The RREF of the augmented matrix is

```
EDU>> rref([1 2 3 1; 1 2 3 2; 1 2 3 3])

ans =

     1     2     3     0
     0     0     0     1
     0     0     0     0
```

We certainly can't say "all" of these systems have a unique solution, but we might be able to revise it to say "most" of them do.

B.2 Studio 1.2 solutions

(0) **(a)** Does the vector $\mathbf{b} = \begin{bmatrix}1\\2\\3\end{bmatrix}$ lie in the span of the vectors $\mathbf{v}_1 = \begin{bmatrix}2\\2\\2\end{bmatrix}$, $\mathbf{v}_2 = \begin{bmatrix}1\\2\\3\end{bmatrix}$, $\mathbf{v}_3 = \begin{bmatrix}2\\1\\3\end{bmatrix}$?

If $\mathbf{b}$ lies in the span of $\mathbf{v}_1$,$\mathbf{v}_2$, $\mathbf{v}_3$, then there exists weights x_1, x_2, x_3 such that $x_1\mathbf{v}_1 + x_2\mathbf{v}_2 + x_3\mathbf{v}_3 = \mathbf{b}$. In another formulation, $\mathbf{b}$ lies in the span of these vectors if and only if the matrix equation $A\mathbf{x} = \mathbf{b}$, where $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are the columns of A, has a solution. We can determine whether such a solution exists by using Matlab:

```
>> rref([2 1 2 1; 2 2 1 2; 2 3 3 3])

ans =

     1     0     0     0
     0     1     0     1
     0     0     1     0
```

The RREF indicates that the linear system is consistent, and in particular that there is a unique solution, so $\mathbf{b}$ lies in the span of $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$.

(b) Determine whether there is a solution to the matrix equation

$$\begin{bmatrix}72.0 & 56.0 & 8.0\\74.0 & 69.0 & 74.0\\95.0 & 13.0 & 11.0\end{bmatrix}\mathbf{x} = \begin{bmatrix}3.27\\33.67\\90.16\end{bmatrix}.$$

Denote the matrix equation above as $A\mathbf{x} = \mathbf{b}$. Note that the linear system represented by the matrix equation has the augmented matrix $[A|\mathbf{b}]$. After making this observation, the rest is a straightforward Matlab exercise:

```
>> rref([72.0 56.0 8.0 3.27; 74.0 69.0 74.0 33.67; 95.0 13.0 11.0 90.16])

ans =

    1.0000         0         0    1.0596
         0    1.0000         0   -1.4047
         0         0    1.0000    0.7052
```

So the system is consistent, and in particular has a unique solution.

(c) Determine whether the linear system

$$\begin{aligned} 46.0x_1 + 11.0x_2 + 56.0x_3 &= 47.22 \\ 36.0x_1 + 100.0x_2 + 30.0x_3 &= 98.42 \end{aligned}$$

is consistent.

This is similar to the previous part and is essentially an exercise of transferring between different forms of representation. We take the RREF of the augmented matrix:

```
>> rref([46.0 11.0 56.0 47.22; 36.0 100.0 30.0 98.42])

ans =

    1.0000         0    1.2536    0.8657
         0    1.0000   -0.1513    0.6725
```

Hence the system is consistent, because the right-most column is not a pivot column. Alternatively, we could've used our knowledge of underdetermined systems to immediately note that there must be infinitely many solutions to this linear system.

(1) Let A be a $m \times n$ matrix. Prove that if $A\mathbf{x} = \mathbf{b}$, then $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$.

Remember that $A\mathbf{x} = \mathbf{b}$ represents a system of linear equations. Each component of $\mathbf{x}$ represents a variable, and each component of $\mathbf{b}$ represents the righthand side of an equation. Also recall that coefficient matrix A has one column for each variable and one row for each equation. Hence, the number of columns in A must be equal to the number of components in $\mathbf{x}$, and the number of rows in A must be equal to the number of components in $\mathbf{b}$.

(2) Let A be an $m \times n$ matrix. Prove that if the matrix equation $A\mathbf{x} = \mathbf{b}$ has a solution $\mathbf{x} \in \mathbb{R}^n$ for every choice of $\mathbf{b} \in \mathbb{R}^m$, then A must have a pivot position in every row.

Imagine that $rref(A)$ does not have a pivot in every row, so that $rref(A)$ has a row of zeros. We could choose some $\mathbf{d}$ such that $[rref A|\mathbf{d}]$ has a 1 in the rightmost column in the row in which $rref(A)$ is all zeros. Since all the operations necessary to produce the row reduced echelon form are reversible (namely taking linear combinations of rows), we can transform $[rref(A)|\mathbf{d}]$ back into an augmented matrix $[A|\mathbf{b}]$ which represents an inconsistent system.

(Note: this problem is too hard. Sorry! The other direction is easier: if every row of A has a pivot, then $A\mathbf{x} = \mathbf{b}$ is consistent for every $\mathbf{b}$. Give it a shot.)

(3) Let A be a $m \times n$ matrix, and $\mathbf{u}$ and $\mathbf{v}$ be vectors in $\mathbb{R}^n$. Prove that $A(\mathbf{u} + \mathbf{v}) = A\mathbf{u} + A\mathbf{v}$.

Let

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

Then

$$\begin{aligned}
A(\mathbf{u}+\mathbf{v}) = A\begin{bmatrix} u_1+v_1 \\ u_2+v_2 \\ \vdots \\ u_n+v_n \end{bmatrix} &= \begin{bmatrix} a_{11}(u_1+v_1)+a_{12}(u_2+v_2)+\cdots+a_{1n}(u_n+v_n) \\ a_{21}(u_1+v_1)+a_{22}(u_2+v_2)+\cdots+a_{2n}(u_n+v_n) \\ \vdots \\ a_{m1}(u_1+v_1)+a_{m2}(u_2+v_2)+\cdots+a_{mn}(u_n+v_n) \end{bmatrix} \\
&= \begin{bmatrix} a_{11}u_1+a_{12}u_2+\cdots+a_{1n}u_n \\ a_{21}u_1+a_{22}u_2+\cdots+a_{2n}u_n \\ \vdots \\ a_{m1}u_1+a_{m2}u_2+\cdots+a_{mn}u_n \end{bmatrix} + \begin{bmatrix} a_{11}v_1+a_{12}v_2+\cdots a_{1n}v_n \\ a_{21}v_1+a_{22}v_2+\cdots a_{2n}v_n \\ \vdots \\ a_{m1}v_1+a_{m2}v_2+\cdots a_{mn}v_n \end{bmatrix} \\
&= A\mathbf{u}+A\mathbf{v}.
\end{aligned}$$

(4)

$$B = \begin{bmatrix} 1 & 4 & 1 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & 7 \\ 2 & 9 & 5 & -7 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 4 & 1 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & 7 \\ 2 & 9 & 6 & -7 \end{bmatrix},$$

$$D = \begin{bmatrix} 1 & 4 & 1 & 2 \\ 0 & 1 & 3 & -4 \\ 0 & 2 & 6 & 7 \end{bmatrix}, \quad E = \begin{bmatrix} 1 & 4 & 1 \\ 0 & 1 & 3 \\ 0 & 2 & 6 \\ 2 & 9 & 6 \end{bmatrix}$$

(a) Can every vector in $\mathbb{R}^4$ be written as a linear combination of the columns of B?

The columns of B span $\mathbb{R}^4$ if and only if every $\mathbf{b} \in \mathbb{R}^4$ has a solution $\mathbf{x}$ such that $B\mathbf{x} = \mathbf{b}$. We proved that in earlier that if B has a pivot in every row, then every such $\mathbf{b}$ has a solution. So we simply have to test B in Matlab:

```
>> rref([1 4 1 2; 0 1 3 -4; 0 2 6 7; 2 9 5 7])

ans =

     1     0   -11     0
     0     1     3     0
     0     0     0     1
     0     0     0     0
```

So the columns of B do not span $\mathbb{R}^4$. For instance,

$$\begin{bmatrix} 1 \\ 4 \\ 1 \\ 3 \end{bmatrix}$$

is not in the span of the columns because

```
>> rref([1 4 1 2 1; 0 1 3 -4 4 ; 0 2 6 7 1; 2 9 5 7 3])

ans =

     1     0   -11     0     0
     0     1     3     0     0
     0     0     0     1     0
     0     0     0     0     1
```

And so the system with the given $\mathbf{b}$ is consistent.

(b) Do the columns of C span $\mathbb{R}^4$?

The reasoning is the same as the previous part; we need only test C in Matlab:

```
>> rref([1 4 1 2; 0 1 3 -4; 0 2 6 7; 2 9 6 -7])

ans =

     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1
```

So since C has a pivot in every row, there is a solution to $C\mathbf{x} = \mathbf{b}$ for any chosen $\mathbf{b}$.

(c) Does the matrix equation $D\mathbf{x} = \mathbf{b}$ have a solution for every $\mathbf{b} \in \mathbb{R}^3$?

This is equivalent to asking whether the columns of D span $\mathbb{R}^3$. We can test this in Matlab:

```
>> rref([1 4 1 2; 0 1 3 -4; 0 2 6 7])

ans =

     1     0   -11     0
     0     1     3     0
     0     0     0     1
```

So, yes, there is a solution to $D\mathbf{x} = \mathbf{b}$ for any chosen $\mathbf{b} \in \mathbb{R}^3$.

(d) Do the columns of E span $\mathbf{R}^4$?

As we have done with the previous parts, we can test this question with Matlab:

```
>> rref([1 4 1; 0 1 3; 0 2 6; 2 9 6])

ans =

     1     0     0
     0     1     0
     0     0     1
     0     0     0
```

So, no, there is not always a solution to $E\mathbf{x} = \mathbf{b}$. For instance, take

$$\mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

which results in

```
>> rref([1 4 1 1; 0 1 3 1; 0 2 6 1; 2 9 6 1])

ans =

     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1
```

(5) Define

$$A = \begin{bmatrix} 10 & -7 & 1 & 4 & 6 \\ -8 & 4 & -6 & -10 & -3 \\ -7 & 11 & -5 & -1 & -8 \\ 3 & -1 & 10 & 12 & 12 \end{bmatrix}.$$

(a) Do the columns of A span $\mathbb{R}^4$?

This is equivalent to asking whether $A\mathbf{x} = \mathbf{b}$ has a solution $\mathbf{x}$ for every $\mathbf{b} \in \mathbb{R}^4$. From an earlier problem we know this is the case if and only if every row of A has a pivot position. We can test this in Matlab:

```
>> rref([10 -7 1 4 6; -8 4 -6 -10 -3; -7 11 -5 -1 -8; 3 -1 10 12 12])

ans =

     1     0     0     1     0
     0     1     0     1     0
     0     0     1     1     0
     0     0     0     0     1
```

So, yes, the columns of A do span $\mathbb{R}^4$.

(b) Find a column of A that can be deleted and yet have the remaining matrix columns still span $\mathbb{R}^4$.

From the previous part, it appears that if we delete column 4, then we will still have a pivot in every row. We can verify this is true:

```
>> rref([10 -7 1 6; -8 4 -6 -3; -7 11 -5 -8; 3 -1 10 12])

ans =

     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1
```

(c) Can you delete more than one column and yet have the remaining matrix columns still span $\mathbb{R}^4$?

If we were to delete another column, we have a matrix with 4 rows and 3 columns. Since we can have at most one pivot per row, we would have at least 1 row without a pivot. Then by our prior results, the columns of the new matrix would not span $\mathbb{R}^4$.

B.3 Studio 1.3 solutions

(0) Suppose weve been tracking 5 industries over the last several years, and constructed the following matrix describing how individual industries buy the output of other industries.

$$C = \begin{bmatrix} 0.1 & 0.6 & 0.2 & 0 & 0.3 \\ 0.1 & 0.2 & 0 & 0.7 & 0.1 \\ 0.3 & 0 & 0.1 & 0.1 & 0.2 \\ 0.4 & 0.1 & 0.4 & 0.2 & 0.3 \\ 0.1 & 0.1 & 0.3 & 0 & 0.1 \end{bmatrix}$$

As weve seen before with these type of matrices, the matrix entry (i, j) is the percent of the output of industry j which is bought by industry i.

(a) Explain in words why the sums of each column must be equal to 1.

Each column represents the sales of a particular industry. Since we've assumed that each industry either sells or retains 100% of its total output, the column sums must be 1.

(b) Explain in words by the row sums need not be equal to 1.

Each row represents the purchases of a particular industry. Since we haven't placed any constraints on how much a particular industry buys from its fellow industries, we can only assume that some industries will buy more, and some industries will buy less. Therefore, the row sums need not be equal to 1.

(c) Suppose the industry i has a total output p_i, measured in dollars. How much should each industry produce in order for all industries to have their costs exactly balance their revenue?

Remember that the total cost is given by $\mathbf{c} = C\mathbf{p}$ and the total revenue is given by $\mathbf{r} = I\mathbf{p}$, where I here is the 5×5 identity matrix. At the break even point, revenue exactly equals cost. In symbols, $C\mathbf{p} = I\mathbf{p}$. Solving for $\mathbf{p}$ here is equivalent to solving for $\mathbf{p}$ in the homogeneous equation $(C - I)\mathbf{p} = \mathbf{0}$.

```
>> C

C =

    0.1000    0.6000    0.2000         0    0.3000
    0.1000    0.2000         0    0.7000    0.1000
    0.3000         0    0.1000    0.1000    0.2000
    0.4000    0.1000    0.4000    0.2000    0.3000
    0.1000    0.1000    0.3000         0    0.1000

>> rref(C-eye(5))

ans =

    1.0000         0         0         0   -2.4041
         0    1.0000         0         0   -2.6700
         0         0    1.0000         0   -1.3086
         0         0         0    1.0000   -2.5651
         0         0         0         0         0
```

The function `eye` in Matlab generates identity matrices. Type `help eye` to see how it works. We see that the last production level, call it p_5, is a free variable, so there is not a unique solution to the problem. (Notice that I've only taken the RREF of the coefficient matrix, not the augmented matrix. Why is this OK?) Once, we've chosen p_5, the other production levels are set. For instance, $p_1 = 2.4041p_5$. In words, the production of industry 1 is 240% that of industry 5.

(1) You and a friend have opened a boutique cupcake shop, and together you're trying to get a handle on your business total cost as a function of the total number of cupcakes produced. You currently have three data points: you know that the fixed cost of your business hovers around \$700 per week; in the first week you produced 100 cupcakes and your total cost was \$775; in the second week you produced 200 cupcakes and your total cost was \$800.

(a) Your cofounder obviously didn't have a Babson education: he thought that he had found a line that perfectly fit all three data points. Use linear algebraic reasoning to convince him that this can't be the case.

Imagine the linear model is of the form $C(x) = c_1 x + c_0$. We would like to determine the values of c_0 and c_1 that fit our data, if any such coefficients exist. We can substitute data points (0,700), (100,775) and (200,800) into the proposed linear model to produce a system of linear equations.

$$\begin{aligned} 700 &= c_1(0) + c_0 \\ 775 &= c_1(100) + c_0 \\ 800 &= c_1(200) + c_0 \end{aligned}$$

This corresponds to the matrix equation

$$\begin{bmatrix} 1 & 0 \\ 1 & 100 \\ 1 & 200 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = \begin{bmatrix} 700 \\ 775 \\ 800 \end{bmatrix}$$

Notice that this linear system is overdetermined, as covered in Studio 1. We take the RREF of the augmented matrix in Matlab to learn about the solutions.

```
>> rref([1 0 700; 1 100 775; 1 200 800])

ans =

     1     0     0
     0     1     0
     0     0     1
```

We conclude that the linear system is inconsistent. Therefore, there exists no line $y = c_1 x + c_0$ that passes through all three data points.

(b) After giving it another try, your cofounder claims that he has found several quadratic models $C(x) = c_0 + c_1 x + c_2 x^2$ which perfectly match the data. Either prove your cofounder right by producing an infinite family of models that perfectly predict the data, or disprove his claim by showing that there is a unique solution or no solution to this problem. (Not that simply producing a solution is not enough! How do you know there aren't more, for instance?)

This is similar to the last problem, except now we have three coefficients to search for. We'll jump straight to the matrix equation:

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 100 & 100^2 \\ 1 & 200 & 200^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 700 \\ 775 \\ 800 \end{bmatrix}$$

Again, we take the RREF of the augmented matrix in Matlab to learn about the solutions.

```
>> rref([1 0 0 700; 1 100 100^2 775; 1 200 200^2 800])

ans =

    1.0000         0         0  700.0000
         0    1.0000         0    1.0000
         0         0    1.0000   -0.0025
```

So there is a unique parabola $C(x) = -0.0025x^2 + x + 700$ that passes through all three of these data points. Notice that the fixed cost in both the data and the model is 700, which should make you feel a little more confident about the solution.

(c) Motivated by your success with the quadratic modeling exercise, you've recently been wondering if there's a cubic total cost model $C(x) = c_0 + c_1 x + c_2 x^2 + c_3 x^3$ that will even perfectly predict the data. Use linear algebraic reasoning to show how many of these solutions must exist.

We'll play the same game that we've been playing the previous parts. Here the matrix equation is

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 100 & 100^2 & 100^3 \\ 1 & 200 & 200^2 & 200^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 700 \\ 775 \\ 800 \end{bmatrix}$$

Notice that this linear system is underdetermined, as covered in Studio 1. We take the RREF of the augmented matrix in Matlab to learn about the solutions.

```
>> rref([1 0 0 0 700; 1 100 100^2 100^3 775; 1 200 200^2 200^3 800])

ans =

           1           0           0           0         700
           0           1           0      -20000           1
           0           0           1         300           0
```

Notice that the coefficient c_3 is a free variable, so there are infinitely many solutions to the system. In context of the problem, there are infinitely many cubic functions $C(x) = c_0 + c_1x + c_2x^2 + c_3x^3$ that pass through all three data points.

(2) An economy has four sectors: agriculture, manufacturing, services and transportation. Agriculture sells 20% of its output to manufacturing, 30% to services, 30% to transportation and retains the rest. Manufacturing sells 35% of its output to agriculture, 35% to services, 20% to transportation and retains the rest. Services sells 10% of its output to agriculture, 20% to manufacturing, 20% to transportation and retains the rest. Transportation sells 20% of its output to agriculture, 30% to manufacturing, 30% to services and retains the rest.

(a) Construct the exchange matrix for this economy.

The columns represent what industries sell, and the rows represent what industries buy. With the columns in order as the industries were given in the problem statement, the exchange matrix is

$$C = \begin{bmatrix} 0.20 & 0.35 & 0.10 & 0.20 \\ 0.20 & 0.10 & 0.20 & 0.30 \\ 0.30 & 0.35 & 0.50 & 0.30 \\ 0.30 & 0.20 & 0.20 & 0.20 \end{bmatrix}.$$

(b) Find a set of equilibrium prices for the economy if the value of transportation is $10.00 per unit.

At the break even point we have $C\mathbf{p} = I\mathbf{p}$, where I here is the 4×4 identity matrix. Solving this system for $\mathbf{p}$ is the same as solving the homogeneous system $(C - I)\mathbf{p}$. We take the RREF of the coefficient matrix in Matlab to learn about the solutions.

```
>> C = [0.2 0.35 0.1 0.2; 0.2 0.1 0.2 0.3; 0.3 0.35 0.5 0.3; 0.3 0.2 0.2 0.2]

C =

    0.2000    0.3500    0.1000    0.2000
    0.2000    0.1000    0.2000    0.3000
    0.3000    0.3500    0.5000    0.3000
    0.3000    0.2000    0.2000    0.2000

>> rref(C - eye(4))

ans =

    1.0000         0         0   -0.8738
         0    1.0000         0   -0.9206
         0         0    1.0000   -1.7687
         0         0         0         0
```

(Notice that we too the RREF of the coefficient matrix here instead of the augmented matrix. Why is that OK?) Then p_t, the total production of the transportation sector, is a free variable, and so there are infinitely many solutions to this problem. If we set $p_t = 10$ as the problem states, then $p_a = 8.73$, $p_m = 9.20$, $p_s = 17.69$ and $p_t = 10$. In class we've thought of the p's as total production value, but you can also think of each of them as a price per one unit of output of each sector. You just have to be careful to specify that we're using the same measure of units of output for each industry.

(c) The services sector launches a successful "eat farm fresh" campaign, and increases its share of the output from the agricultural sector to 40%, where as the share of agricultural production going to manufacturing falls to 10%. Construct the exchange matrix for this new economy.

The exchange matrix becomes

$$C = \begin{bmatrix} 0.20 & 0.35 & 0.10 & 0.20 \\ \mathbf{0.10} & 0.10 & 0.20 & 0.30 \\ \mathbf{0.40} & 0.35 & 0.50 & 0.30 \\ 0.30 & 0.20 & 0.20 & 0.20 \end{bmatrix}.$$

Entries that changed are in bold.

(d) Find a set of equilibrium prices for this new economy if the value of transportation if still \$10.00 per unit. What effect has the service sectors campaign had on the equilibrium prices for the sectors of this economy?

The new equilibrium prices are given through the same process as the earlier solution:

```
>> C = [0.2 0.35 0.1 0.2; 0.1 0.1 0.2 0.3; 0.4 0.35 0.5 0.3; 0.3 0.2 0.2 0.2]

C =

    0.2000    0.3500    0.1000    0.2000
    0.1000    0.1000    0.2000    0.3000
    0.4000    0.3500    0.5000    0.3000
    0.3000    0.2000    0.2000    0.2000

>> rref(C - eye(4))

ans =

    1.0000         0         0   -0.8539
         0    1.0000         0   -0.8447
         0         0    1.0000   -1.8744
         0         0         0         0
```

If we set $p_t = 10$ as the problem states, then $p_a = 8.54$, $p_m = 8.45$, $p_s = 18.74$ and $p_t = 10$. We can write the impact of campaign on the prices as $\Delta p_a = 8.54 - 8.73 = -0.19$, $\Delta p_m = 8.45 - 9.20 = -0.75$, $\Delta p_s = 18.74 - 17.69 = 1.05$. So the campaign increased the price per unit of the services industry, and decreased the price per unit of the agriculture and manufacturing industries.

B.4 Studio 1.4 solutions

(0) Show that the columns of a matrix A are linearly dependent if A has more columns than rows.

A matrix can have at most one pivot per row. Since there are more rows than columns, there is at least one column of A that does not have a pivot. If there is a free variable in A, then the homogeneous equation $A\mathbf{x} = \mathbf{0}$ has a nontrivial solution. So there is a nontrivial linear combination of the columns of A that produces the zero vector. This is exactly the definition of linear dependence. Hence, the columns of A are linearly dependent if A has more columns than rows.

(1) Give an example of a matrix A which has fewer columns than rows and whose columns are linearly dependent.

Consider the matrix

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

Call the columns $\mathbf{v}_1$ and $\mathbf{v}_2$ in order. Then there are nontrivial linear combinations of the columns that produce the zero vector, for instance $2v_1 - 2v_2 = \mathbf{0}$. Hence the columns of the matrix are not linearly independent.

(2) Suppose A is a matrix which has the property that for any $\mathbf{b} \in \mathbb{R}^m$, there exists at *most* one solution $\mathbf{x} \in \mathbb{R}^n$. Explain why the columns of A must be linearly independent.

Remember if $\mathbf{b}$ is in the span of the columns of A, then there exist as many solutions to $A\mathbf{x} = \mathbf{b}$ as there are to $A\mathbf{x} = \mathbf{0}$. So if any $A\mathbf{x} = \mathbf{b}$ has at most one solution, then $A\mathbf{x} = \mathbf{0}$ has *exactly* one solution, namely the trivial solution. But then the only linear combination of the columns of A that produces the zero vector is the trivial combination in which all weights are zero. Hence, the columns are linearly independent.

(3) Let's take another look at the 5 industry economic model we examined in the last studio. The matrix describing how these industries bought from and sold to one another was

$$\mathbf{C} = \begin{bmatrix} 0.1 & 0.6 & 0.2 & 0 & 0.3 \\ 0.1 & 0.2 & 0 & 0.7 & 0.1 \\ 0.3 & 0 & 0.1 & 0.1 & 0.2 \\ 0.4 & 0.1 & 0.4 & 0.2 & 0.3 \\ 0.1 & 0.1 & 0.3 & 0 & 0.1 \end{bmatrix}$$

(a) Are the columns of C linearly independent?

Yes. We examine the RREF of C in order to learn about the solutions of $C\mathbf{x} = \mathbf{0}$.

```
>> C

C =

    0.1000    0.6000    0.2000         0    0.3000
    0.1000    0.2000         0    0.7000    0.1000
    0.3000         0    0.1000    0.1000    0.2000
    0.4000    0.1000    0.4000    0.2000    0.3000
    0.1000    0.1000    0.3000         0    0.1000

>> rref(C)

ans =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

Hence, the only linear combination of the columns of C which produces the zero vector is the trivial combination in which all weights are zero. Therefore the columns of C are linearly independent.

(b) How many, if any, production vectors $\mathbf{p}$ lead to a given total cost vector $\mathbf{c}$?

We can essentially recycle the answer to the last part. Remember that if $\mathbf{c}$ is in the span of the columns of C, then there are as many solutions to the inhomogeneous equation $C\mathbf{p} = \mathbf{c}$ as there are to the homogeneous equation $C\mathbf{p} = \mathbf{0}$. Also recall that if there is a pivot in every row of C, then it is impossible that the system is inconsistent for any choice of $\mathbf{c}$. Then since there is a unique solution to $C\mathbf{p} = \mathbf{0}$, there is a unique solution to $C\mathbf{p} = \mathbf{c}$ for any $\mathbf{c} \in \mathbb{R}^5$.

(c) Last time we were trying to solve the equation $(C - R)\mathbf{p} = \mathbf{0}$ in order to determine the total production levels $\mathbf{p}$ necessary for all industries to break even at the same time. We defined R to be the identity matrix with 5 rows and columns I_5. Are the columns of $C - R$ linearly independent?

No. We examine the RREF of $C - I$, where I here is the 5×5 identity matrix, in order to learn about the solutions of $C\mathbf{x} = \mathbf{0}$.

```
>> rref(C - eye(5))

ans =

    1.0000         0         0         0   -2.4041
         0    1.0000         0         0   -2.6700
         0         0    1.0000         0   -1.3086
         0         0         0    1.0000   -2.5651
         0         0         0         0         0
```

There is a free variable, and so the homogeneous equation has infinitely many solutions. Therefore the columns of $C - I$ are linearly dependent.

(d) How many, if any, production vectors $\mathbf{p}$ lead to a break even scenario?

Infinitely many. This is a direct result of the previous part.

(e) Generate another cost matrix C, say for 3 industries. Are the columns of C linearly independent? Are the columns of $C - R$ linearly independent?

Let

$$C = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

The columns of C are not linearly independent. (Prove it to yourself.) The columns of $C - I_3$ are not linearly independent either:

```
>> C = [1/3 1/3 1/3;1/3 1/3 1/3;1/3 1/3 1/3;]

C =

    0.3333    0.3333    0.3333
    0.3333    0.3333    0.3333
    0.3333    0.3333    0.3333
```

```
>> rref(C - eye(3))

ans =

    1.0000         0   -1.0000
         0    1.0000   -1.0000
         0         0         0
```

The matrix $C - I_3$ has a free variable, and so there are infinitely many solutions to the homogeneous equation.

(f) Can you logically justify (meaning without Matlab) why in general the columns of C would be linearly independent but the columns of $C - R$ would not?

I can't; I was hoping one of you would come up with something clever. From the last part, it's clear that it's not always true, but maybe it's true "most of the time."

B.5 Studio 2.1 solutions

(0) Define matrices

$$A = \begin{bmatrix} 2.0 & 9.0 & 22.0 & 51.0 \\ 12.0 & 96.0 & 47.0 & 45.0 \\ 23.0 & 93.0 & 47.0 & 4.0 \end{bmatrix}, \quad B = \begin{bmatrix} 14.0 & 84.0 & 81.0 \\ 66.0 & 93.0 & 39.0 \\ 31.0 & 3.0 & 58.0 \end{bmatrix}, \quad C = \begin{bmatrix} 76.0 & 96.0 & 13.0 \\ 34.0 & 2.0 & 65.0 \\ 92.0 & 11.0 & 67.0 \\ 17.0 & 66.0 & 27.0 \end{bmatrix}$$

(a) Is AB defined? If so, what is its size?

No; A is 3×4, and B is 3×3. The inner dimensions do not agree, so the matrix product is not defined.

(b) Is BA defined? If so, what is its size?

Yes; B is 3×3, and A is 3×4. So the product BA is 3×4.

(c) Is AC defined? If so, what is its size?

Yes; A is 3×4, and C is 4×3. So the produce AC is 3×3.

(d) Is CA defined? If so, what is its size?

Yes; A is C is 4×3, and A is 3×4. So the product CA is 4×4.

(e) Is ABC defined? If so, what is its size?

No. Here it helps to remember that matrix multiplication is associative, so we can think about ABC as either $A(BC)$ or $(AB)C$. It really comes down to which ever is convenient. Take $(AB)C$. We showed above that AB is not defined, so $(AB)C$ can't be defined.

(f) Is CBA defined? If so, what is its size?

Yes; B is 3×3 and A is 3×4. So BA is 3×4. Then since C is 4×3 and BA is 3×4, the product $C(BA)$ is 4×4.

(1) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using RREF the total production necessary to satisfy the final demands represented by the vector

$$\mathbf{d} = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

Let $\mathbf{x}$ be a vector containing the production of each industry. Recall the intermediate demand is given by $C\mathbf{x}$, so that the total demand is $C\mathbf{x} + \mathbf{d}$. In order for production to exactly meet the total demand, it must be that $\mathbf{x} = C\mathbf{x} + \mathbf{d}$. This implies that $(I_3 - C)\mathbf{x} = \mathbf{d}$.

```
>> C = [0.2 0.7 0.1; 0.4 0.1 0.7; 0.1 0.1 0.1];
>> [eye(3) - C [10; 20; 30]]

ans =

    0.8000   -0.7000   -0.1000   10.0000
   -0.4000    0.9000   -0.7000   20.0000
   -0.1000   -0.1000    0.9000   30.0000

>> rref([eye(3) - C [10; 20; 30]])

ans =

    1.0000         0         0  135.2518
         0    1.0000         0  131.2949
         0         0    1.0000   62.9496
```

So industry 1 should produce 135.25 units, and so on.

(2) If $AB = 0$ but neither A nor B is zero, we call A and B *zero divisors.* Note that there are no zero divisors in the real numbers, so this isn't something we may have encountered so far. Matrices can be zero divisors. For instance, if A and B are

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \end{bmatrix} \quad B = \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & 1 \\ -1 & -1 \end{bmatrix}$$

show that AB is the zero matrix, so that both A and B are zero divisors.

This one is a simple Matlab exercise designed to make you practice actually doing matrix multiplication:

```
>> [1 1 1 1; 1 1 -1 -1] * [1 -1; -1 1; 1 1; -1 -1]

ans =

     0     0
     0     0
```

(3) (a) Unlike the real numbers, matrices do not typically commute, meaning that $AB \neq BA$ in general. To verify this statement, generate two random 3×3 matrices A and B and verify that they do not commute, that is, that $AB \neq BA$.

First we define the matrices.

```
>> A = rand(3,3)

A =

    0.8147    0.9134    0.2785
    0.9058    0.6324    0.5469
    0.1270    0.0975    0.9575

>> B = rand(3,3)

B =

    0.9649    0.9572    0.1419
    0.1576    0.4854    0.4218
    0.9706    0.8003    0.9157
```

Now, if $AB = BA$, then $AB - BA$ should be the zero matrix. But this is true:

```
>> A*B - B*A

ans =

   -0.4707   -0.0544   -0.1722
    0.8828    1.1196    0.1828
   -0.5647   -0.5467   -0.6488
```

(b) Above we said that matrices do not commute *in general.* But some matrices do commute. Come up with an example of matrix that commutes with any square matrix A.

Both the identity matrix and the zero matrix commute with any square matrix.

(4) In the real numbers, if $xy = xz$, then $y = z$, because we can cancel the x on each side. But with matrices, things are a little more complicated.

$$A = \begin{bmatrix} 3 & 4 & 5 & 6 \\ 7 & 8 & 9 & 10 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 3 & 4 & 5 \\ 6 & 7 & 10 & 11 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & 1 \\ -1 & -1 \end{bmatrix}$$

(a) Show that $AC = BC$, and note that $A \neq B$.

We can easily verify the statement in Matlab.

```
>> A = [3 4 5 6; 7 8  9  10];
>> B = [2 3 4 5; 6 7 10 11];
>> C = [1 -1; -1 1; 1 1; -1 -1];
>> A*C - B*C

ans =

     0     0
     0     0
```

Hence $AC = BC$.

(b) Think about how zero divisors and cancellation are related. Use this to come up with another example of matrices A and B such that $AC = BC$ but $A \neq B$.

Suppose we have matrices such that $AC = BC$. Then $(A - B)C = 0$, so that $A - B$ and C are zero divisors. Notice that from above $(A - B)C = 0$. If we can find new matrices D and E such that $D - E = A - B$, then we're all set.A =

B.6 Studio 2.2 solutions

(0) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vector

$$\mathbf{d} = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

```
EDU>> C = [0.2 0.7 0.1; 0.4 0.1 0.7; 0.1 0.1 0.1]

C =

    0.2000    0.7000    0.1000
    0.4000    0.1000    0.7000
    0.1000    0.1000    0.1000

EDU>> (eye(3) - C) \ [10;20;30]

ans =

  135.2518
  131.2950
   62.9496
```

Compare this with the equivalent problem on Studio 5.

(1) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 11 \\ 20 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

```
EDU>> C = [0.2 0.7 0.1; 0.4 0.1 0.7; 0.1 0.1 0.1];
EDU>> x1 = (eye(3) - C) \ [11; 20; 30]

x1 =

  137.9137
  132.8417
   63.4173

EDU>> x2 = (eye(3) - C) \ [10; 20; 30]

x2 =

  135.2518
  131.2950
   62.9496
```

(b) Confirm that the difference of the two production vectors you produced is the first column of the matrix $(I - C)^{-1}$.

```
EDU>> x1 - x2

ans =

    2.6619
    1.5468
    0.4676
```

The inverse is

```
EDU>> inv(eye(3) - C)

ans =

    2.6619    2.3022    2.0863
    1.5468    2.5540    2.1583
    0.4676    0.5396    1.5827
```

So everything checks out.

(c) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 21 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

Keep `C` as in the previous part.

```
EDU>> x1 = (eye(3) - C) \ [10; 21; 30]

x1 =

  137.5540
  133.8489
   63.4892

EDU>> x2 = (eye(3) - C) \ [10; 20; 30]

x2 =

  135.2518
  131.2950
   62.9496
```

(d) Confirm that the difference in the productions you found in the previous part is the second column of $(I - C)^{-1}$.

```
EDU>> x1 - x2

ans =

    2.3022
    2.5540
    0.5396
```

The inverse is

```
EDU>> inv(eye(3) - C)

ans =

    2.6619    2.3022    2.0863
    1.5468    2.5540    2.1583
    0.4676    0.5396    1.5827
```

So everything checks out.

(e) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 20 \\ 31 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

```
EDU>> x1 = (eye(3) - C) \ [10; 20; 31]

x1 =

  137.3381
  133.4532
   64.5324

EDU>> x1 = (eye(3) - C) \ [10; 20; 30]

x1 =

  135.2518
  131.2950
   62.9496
```

(f) Confirm that the difference in the productions you found in the previous part is the third column of $(I - C)^{-1}$.

```
EDU>> x1 - x2

ans =

    2.0863
    2.1583
    1.5827
```

The inverse is

```
EDU>> inv(eye(3) - C)

ans =

    2.6619    2.3022    2.0863
    1.5468    2.5540    2.1583
    0.4676    0.5396    1.5827
```

So everything checks out.

(g) Show that the additional production necessary to satisfy one additional unit of final demand for industry i is exactly the i^{th} column of $(I-C)^{-1}$.

Imagine $\mathbf{d}_1$ and $\mathbf{d}_2$ differ by 1 in component i and are identical in all other components. Then if $\mathbf{x}_1 = C\mathbf{x} + \mathbf{d}_1$, $\mathbf{x}_2 = C\mathbf{x} + \mathbf{d}_2$ and $I - C$ is invertible, then the difference in production necessary to the satisfy the two demands is

$$\begin{aligned}
\Delta\mathbf{x} &= \mathbf{x}_1 - \mathbf{x}_2 \\
&= (I-C)^{-1}\mathbf{d}_1 - (I-C)^{-1}\mathbf{d}_2 \\
&= (I-C)^{-1}(\mathbf{d}_1 - \mathbf{d}_2) \\
&= (I-C)^{-1}\begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix},
\end{aligned}$$

where the 1 in the vector in the last equality is in component i. Interpretting the matrix-vector multiplication as a linear combination of the columns of $(I-C)^{-1}$, we have 0 weight on every column except column i, and weight 1 on column i. Hence, $\Delta\mathbf{x}$ is the i^{th} column of $(I-C)^{-1}$.

(2) **(a)** Consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} k & 0.5 \\ 0.6 & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= C_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse C_k^{-1} exist?

The inverse exists so long as the determinant is nonzero. If you're looking to save time like I am, you type `solve det({{k,0.5},{0.6,0.2}}) = 0` into WolframAlpha and arrive at $\det(C_k) = 0$ if and only if $k = 1.5$.

(b) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & 0.5 \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= B_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse B_k^{-1} exist?

The inverse exists so long as the determinant is nonzero. If you're looking to save time like I am, you type `solve det({{0.1, 0.5},{k, 0.2}}) = 0` into WolframAlpha and arrive at $\det(C_k) = 0$ if and only if $k = 0.04$.

(c) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & \ell \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= A_{\ell,k} \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of ℓ and k does the inverse $A_{\ell,k}^{-1}$ exist?

The inverse exists so long as the determinant is nonzero. Here the determinant is $(0.1)(0.2) - k\ell$, so the inverse exists so long as $k\ell \neq (0.1)(0.2) = 0.02$.

(3) Generally, we can write a polynomial of degree $n-1$ as

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1}x^{n-1}.$$

In order for $p(x)$ to match our n data points exactly, it must be the case that $p(x_i) = y_i$ for every $i = 1, 2, \ldots, n$. But each of these equalities amounts to a linear combination of the coefficients $a_0, a_1, \ldots, a_{n-1}$. We can encode these linear combinations in a matrix equation:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$V\mathbf{a} = \mathbf{y}.$$

We know that in general there is either no solution, a unique solution, or infinitely many solutions to a matrix equation like the one above. But here this idea has a special context: if there is a unique solution to this matrix equation, there is a *unique* polynomial of degree $n-1$ that passes through our n data points. Show that if the $x_1, x_2, \ldots, x_n$ are all distinct, then V is invertible. (Hint: feel free to use the fundamental theorem of algebra which says that a polynomial $p(x)$ of degree k has at most k distinct roots $x_1, x_2, \ldots, x_k$ at which $p(x_i) = 0$.)

Suppose that there is a nontrivial solution to $V\mathbf{a} = \mathbf{0}$. Then there is $p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1}x^{n-1}$ with not all the $a_i = 0$ (*i.e.*, not the zero polynomial $p(x) = 0$) such that $p(x_1) = 0$, $p(x_2) = 0$, $\ldots$, $p(x_n) = 0$. Since all the x_i are distinct, we've found n distinct roots of $p(x)$. But $p(x)$ has degree $n-1$, so by the fundamental theorem of algebra $p(x)$ can only have $n-1$ roots. This is a contradiction, and so the original claim cannot be true. That is to say, there is no nontrivial solution to $V\mathbf{a} = \mathbf{0}$.

It's worth noting what this means in practical terms. If there is no nontrivial solution to $V\mathbf{a} = \mathbf{0}$, then there is a pivot in every column of V. Since V is square, there is also a pivot in every row of V. These together imply that there is a unique solution to any matrix equation $V\mathbf{a} = \mathbf{y}$. But remember, this matrix equation encodes information about the polynomial $p(x)$. To the uniqueness of solutions in matrix language implies in the polynomial language that there is a unique polynomial of degree $n-1$ passing through any n data points, so long as the x coordinates of all the data points are distinct. That's pretty sweet.

B.7 Studio 2.3 solutions

(0) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vector

$$\mathbf{d} = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

```
EDU>> C = [0.2 0.7 0.1; 0.4 0.1 0.7; 0.1 0.1 0.1]

C =

    0.2000    0.7000    0.1000
    0.4000    0.1000    0.7000
    0.1000    0.1000    0.1000

EDU>> (eye(3) - C) \ [10;20;30]

ans =

  135.2518
  131.2950
   62.9496
```

Compare this with the equivalent problem on Studio 5.

(1) Imagine that a Leontief input-output model has the consumption matrix

$$C = \begin{bmatrix} 0.2 & 0.7 & 0.1 \\ 0.4 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}.$$

(a) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 11 \\ 20 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

```
EDU>> C = [0.2 0.7 0.1; 0.4 0.1 0.7; 0.1 0.1 0.1];
EDU>> x1 = (eye(3) - C) \ [11; 20; 30]

x1 =

  137.9137
  132.8417
   63.4173

EDU>> x2 = (eye(3) - C) \ [10; 20; 30]

x2 =

  135.2518
  131.2950
   62.9496
```

(b) Confirm that the difference of the two production vectors you produced is the first column of the matrix $(I - C)^{-1}$.

```
EDU>> x1 - x2

ans =

    2.6619
    1.5468
    0.4676
```

The inverse is

```
EDU>> inv(eye(3) - C)

ans =

    2.6619    2.3022    2.0863
    1.5468    2.5540    2.1583
    0.4676    0.5396    1.5827
```

So everything checks out.

(c) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 21 \\ 30 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

Keep `C` as in the previous part.

```
EDU>> x1 = (eye(3) - C) \ [10; 21; 30]

x1 =

  137.5540
  133.8489
   63.4892

EDU>> x2 = (eye(3) - C) \ [10; 20; 30]

x2 =

  135.2518
  131.2950
   62.9496
```

(d) Confirm that the difference in the productions you found in the previous part is the second column of $(I - C)^{-1}$.

```
EDU>> x1 - x2

ans =

    2.3022
    2.5540
    0.5396
```

The inverse is

```
EDU>> inv(eye(3) - C)

ans =

    2.6619    2.3022    2.0863
    1.5468    2.5540    2.1583
    0.4676    0.5396    1.5827
```

So everything checks out.

(e) Calculate using the matrix inverse the total production necessary to satisfy the final demands represented by the vectors

$$\mathbf{d}_1 = \begin{bmatrix} 10 \\ 20 \\ 31 \end{bmatrix}, \quad \mathbf{d}_2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}.$$

```
EDU>> x1 = (eye(3) - C) \ [10; 20; 31]

x1 =

  137.3381
  133.4532
   64.5324

EDU>> x1 = (eye(3) - C) \ [10; 20; 30]

x1 =

  135.2518
  131.2950
   62.9496
```

(f) Confirm that the difference in the productions you found in the previous part is the third column of $(I - C)^{-1}$.

```
EDU>> x1 - x2

ans =

    2.0863
    2.1583
    1.5827
```

The inverse is

```
EDU>> inv(eye(3) - C)

ans =

    2.6619    2.3022    2.0863
    1.5468    2.5540    2.1583
    0.4676    0.5396    1.5827
```

So everything checks out.

(g) Show that the additional production necessary to satisfy one additional unit of final demand for industry i is exactly the i^{th} column of $(I - C)^{-1}$.

Imagine $\mathbf{d}_1$ and $\mathbf{d}_2$ differ by 1 in component i and are identical in all other components. Then if $\mathbf{x}_1 = C\mathbf{x} + \mathbf{d}_1$, $\mathbf{x}_2 = C\mathbf{x} + \mathbf{d}_2$ and $I - C$ is invertible, then the difference in production necessary to the satisfy the two demands is

$$\begin{aligned}
\Delta\mathbf{x} &= \mathbf{x}_1 - \mathbf{x}_2 \\
&= (I - C)^{-1}\mathbf{d}_1 - (I - C)^{-1}\mathbf{d}_2 \\
&= (I - C)^{-1}(\mathbf{d}_1 - \mathbf{d}_2) \\
&= (I - C)^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix},
\end{aligned}$$

where the 1 in the vector in the last equality is in component i. Interpretting the matrix-vector multiplication as a linear combination of the columns of $(I-C)^{-1}$, we have 0 weight on every column except column i, and weight 1 on column i. Hence, $\Delta\mathbf{x}$ is the i^{th} column of $(I - C)^{-1}$.

(2) **(a)** Consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} k & 0.5 \\ 0.6 & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= C_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse C_k^{-1} exist?

The inverse exists so long as the determinant is nonzero. If you're looking to save time like I am, you type `solve det({{k,0.5},{0.6,0.2}}) = 0` into WolframAlpha and arrive at $\det(C_k) = 0$ if and only if $k = 1.5$.

(b) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & 0.5 \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= B_k \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of k does the inverse B_k^{-1} exist?

The inverse exists so long as the determinant is nonzero. If you're looking to save time like I am, you type `solve det({{0.1, 0.5},{k, 0.2}}) = 0` into WolframAlpha and arrive at $\det(C_k) = 0$ if and only if $k = 0.04$.

(c) Now consider the parameterized model

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} 0.1 & \ell \\ k & 0.2 \end{bmatrix} \mathbf{x} + \mathbf{d} \\ &= A_{\ell,k} \mathbf{x} + \mathbf{d}.\end{aligned}$$

For what values of ℓ and k does the inverse $A_{\ell,k}^{-1}$ exist?

The inverse exists so long as the determinant is nonzero. Here the determinant is $(0.1)(0.2) - k\ell$, so the inverse exists so long as $k\ell \neq (0.1)(0.2) = 0.02$.

(3) Generally, we can write a polynomial of degree $n-1$ as

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}.$$

In order for $p(x)$ to match our n data points exactly, it must be the case that $p(x_i) = y_i$ for every $i = 1, 2, \ldots, n$. But each of these equalities amounts to a linear combination of the coefficients $a_0, a_1, \ldots, a_{n-1}$. We can encode these linear combinations in a matrix equation:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$V\mathbf{a} = \mathbf{y}.$$

We know that in general there is either no solution, a unique solution, or infinitely many solutions to a matrix equation like the one above. But here this idea has a special context: if there is a unique solution to this matrix equation, there is a *unique* polynomial of degree $n-1$ that passes through our n data points. Show that if the $x_1, x_2, \ldots, x_n$ are all distinct, then V is invertible. (Hint: feel free to use the fundamental theorem of algebra which says that a polynomial $p(x)$ of degree k has at most k distinct roots $x_1, x_2, \ldots, x_k$ at which $p(x_i) = 0$.)

Suppose that there is a nontrivial solution to $V\mathbf{a} = \mathbf{0}$. Then there is $p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}$ with not all the $a_i = 0$ (*i.e.*, not the zero polynomial $p(x) = 0$) such that $p(x_1) = 0$, $p(x_2) = 0$, $\ldots$, $p(x_n) = 0$. Since all the x_i are distinct, we've found n distinct roots of $p(x)$. But $p(x)$ has degree $n-1$, so by the fundamental theorem of algebra $p(x)$ can only have $n-1$ roots. This is a contradiction, and so the original claim cannot be true. That is to say, there is no nontrivial solution to $V\mathbf{a} = \mathbf{0}$.

It's worth noting what this means in practical terms. If there is no nontrivial solution to $V\mathbf{a} = \mathbf{0}$, then there is a pivot in every column of V. Since V is square, there is also a pivot in every row of V. These together imply that there is a unique solution to any matrix equation $V\mathbf{a} = \mathbf{y}$. But remember, this matrix equation encodes information about the polynomial $p(x)$. To the uniqueness of solutions in matrix language implies in the polynomial language that there is a unique polynomial of degree $n-1$ passing through any n data points, so long as the x coordinates of all the data points are distinct. That's pretty sweet.

B.8 Studio 3.2 solutions

(0) Suppose that your car rental company has 2 locations, L1 and L2, that together house 400 cars. Data indicate that on average 90% of the cars rented at L1 are returned to L1, and 80% of cars rented at L2 are returned to L2. (Thankfully, all cars are returned.)

(a) Write the transition matrix of this dynamical system.

$$T = \begin{bmatrix} 0.9 & 0.2 \\ 0.1 & 0.8 \end{bmatrix}$$

(b) Assume that cars are rented and returned weekly, and that each week every car in both locations is rented. Calculate the distribution of cars after week 1, week 3, and week 10, given an even initial distribution of cars.

```
>> T * [200; 200]

ans =

   220
   180

>> T^3 * [200; 200]

ans =

  243.8000
  156.2000

>> T^10 * [200;200]

ans =

  264.7835
  135.2165
```

Note these are the *average* distribution of cars at each location, so the fractional number of cars indicated by the last two computations is not a problem.

(c) Try several other initial conditions. Is the long term behavior of the car distribution the same?

```
>> T^100 * [200;200]

ans =

  266.6667
  133.3333

>> T^100 * [100;300]

ans =

  266.6667
  133.3333

>> T^100 * [300;100]

ans =

  266.6667
  133.3333
```

Yes, all the initial conditions sampled here have the same long term distribution.

(d) Calculate the eigenpairs of the transition matrix, and use them to predict the long term behavior of the system. What is the ratio of the number of cars at each location?

```
>> [V,D] = eig(T)

V =

    0.8944   -0.7071
    0.4472    0.7071


D =

    1.0000         0
         0    0.7000
```

So the eigenpairs are $(\mathbf{v}_1, \lambda_1) = ([2, 1]^T, 1)$ and $(\mathbf{v}_2, \lambda_2) = ([-1, 1]^T, 0.7)$. The dominant eigenpair is the first one. So in the long term, location 1 will have twice as many cars as location 2.

(e) Suppose now that you have 600 total cars. Is the long term behavior of the system the same as before? What is the ratio of the number of cars at each location?

The long term behavior of the system is the same in the sense that the ratio of the cars at the two locations approaches 2:1. For instance,

```
>> T^100 * [100;500]

ans =

  400.0000
  200.0000
```

Since the transition matrix the same, the eigenpairs are the same, and in particular the dominant eigenpair is the same. So the long term ratio of cars in location 1 to cars in location 2 is 2:1.

(1) In one ecological model, the population of owls o_t and population of rats r_t (in thousands) at time t (in months) is related to the populations at time $t+1$ through the following matrix equation:

$$\begin{bmatrix} o_{t+1} \\ r_{t+1} \end{bmatrix} = \begin{bmatrix} .5 & .4 \\ -p & 1.1 \end{bmatrix} \begin{bmatrix} o_t \\ r_t \end{bmatrix}$$
$$\mathbf{x}_{t+1} = T\mathbf{x}_t.$$

The entry p is known as the *predation rate*.

(a) What is the meaning of each entry of matrix?

Entry (1,1) means that in the absence of rats, only 50% of the owls at time t will survive to time $t+1$. Entry (1,2) indicates that each 1000 rats is enough to sustain 0.4 owls. Entry (2,1) means that every owl leads to p thousand fewer rats in a month. Entry (2,2) indicates that in the absence of predations, there will be 10% more rats at time $t+1$ than there are at time t.

(b) What are the units of the predation rate p? In other words, how is p being measured in this model?

The predation rate is measured in thousands of rats per owl.

(c) Assume that $p = 0.1$. What happens to the populations in the long term?

The eigenpairs of the transition matrix are

```
>> [V,D] = eig([.5 .4; -.1 1.1])

V =

   -0.9822   -0.6071
   -0.1876   -0.7947


D =

    0.5764         0
         0    1.0236
```

The dominant eigenpair is $(\mathbf{v}_2, \lambda_2) = ([0.6071, 0.7947]^T, 1.02)$. Hence the total populations grow together over time. The ratio of rats to owls in the long term is $0.6071/0.7947 = 0.7639$, so there are approximately 0.76 owls for every 1000 rats.

(d) Assume that $p = 0.2$. What happens to the populations in the long term?

The eigenpairs of the transition matrix are

```
>> [V,D] = eig([.5 .4; -.2 1.1])

V =

   -0.8944   -0.7071
   -0.4472   -0.7071


D =

    0.7000         0
         0    0.9000
```

The dominant eigenpair is $(\mathbf{v}_2, \lambda_2) = ([1,1]^T, 0.9)$. Hence the total populations shrink together over time, and both populations eventually go extinct. As they decline, the long term ratio of owls to rats is 1 owl for every 1000 rats.

(2) One approach to conservation is through so called *stage-based population modeling*. Typically in these models, we consider on the female members of the species, because in biological terms males are often cheap; there are many males, and most of them are not going to reproduce anyway. For instance, female orcas have three stages: yearlings, juveniles, and mature. The yearly state transition matrix for the female orca population is

$$T = \begin{bmatrix} 0 & 0.0043 & 0.1132 \\ 0.9775 & 0.9111 & 0 \\ 0 & 0.0736 & 0.9534 \end{bmatrix}$$

(a) Interpret $T(2,1)$, $T(3,3)$, $T(1,3)$ and $T(3,2)$ in terms of the stage-based population model.

Entry $T(2,1)$ indicates that 97.75% of yearlings survive to become juveniles. Entry $T(3,3)$ indicates that each year 95.34% of adults survive. Entry $T(1,3)$ indicates that each year there are roughly 11.32 births of yearlings per 100 adults. Entry $T(3,2)$ indicates that each year roughly 7.36% of the juvenile population becomes adult.

(b) What is the long term behavior of the population of female orcas? What is the ratio of juveniles to adults in the long term? What is the ratio of yearlings to adults in the long term?

The eigenpairs of the transition matrix can be computed in Matlab.

```
>> [V,D] = eig([0 .0043 0.1132; 0.9775 0.9111 0; 0 0.0736 0.9534])

V =

    0.6788   -0.0668    0.0816
   -0.7321    0.8489    0.6972
    0.0568   -0.5243    0.7123


D =

    0.0048         0         0
         0    0.8342         0
         0         0    1.0254
```

So the dominant eigenpair is $(\mathbf{v}_3, \lambda_3) = ([0.0816, 0.6976, 0.7123]^T, 1.0254)$. Since the dominant eigenvalue is greater than 1, the population of females grows over time. In the long term, the population will converge to a scalar multiple of the dominant eigenvector. Therefore, there are roughly $0.6972/0.7123 = 97.88\%$ as many juveniles as there are adults, and there are roughly $0.0816\ /\ 0.7123 = 11.46\%$ as many many yearlings as there are adults.

(3) Imagine that we model our business based on two types of customers: one-time customers and repeat customers. These populations are disjoint, so that every current customer is either a one-time or a repeat, but no customer is both. Naturally (and hopefully), a one-time customer can become a repeat customer. From data you've gathered, you know that each month 40% of your one-time customers remain one-time customers. Around 10% of your repeat customers refer a new customer each month. You also know that on average 95% of repeat customers continue to buy your goods, and that on average 30% of one-time customers convert to repeat customers. (A common metric that I've heard is that if a customer has not bought something from you in 3 months then they are removed from the customer group.)

(a) Write a transition matrix for this model.

Let o_t and r_t be the number of one-time and repeat customers in month t, respectively. Then

$$\begin{bmatrix} o_{t+1} \\ r_{t+1} \end{bmatrix} = \begin{bmatrix} 0.4 & 0.1 \\ 0.3 & 0.95 \end{bmatrix} \begin{bmatrix} o_t \\ r_t \end{bmatrix}.$$

(b) High end industries often decide that repeat customers are the segment on which they want to focus. After all, the pool containing their potential clientele is small, so it makes sense to work hard to keep any customers you have. Cheap products often rely on the fact that they will have a large number of constantly changing one-time customers to support their business. Whatever your strategy, it's important make sure that you know what you're getting yourself into. In the long term, what will the ratio of one-time to repeat customers be for the business in this model?

We can compute the eigenpairs of the transition matrix using Matlab.

```
>> [V,D] = eig([0.4 0.1; 0.3 0.95])

V =

   -0.8944   -0.1644
    0.4472   -0.9864


D =

    0.3500         0
         0    1.0000
```

So the dominant eigenpair is $(\mathbf{v}_2, \lambda_2) = ([0.1644, 0.9864]^T, 1)$. In the long term, the customer populations will approach a scaled version of the dominant eigenvector. Therefore, the ratio of one-time to repeat customers in the long term of this model is 0.1644 / 0.9864 = 16.67%.

(4) Suppose your business has a three tiered customer loyalty program. Every customer opting in to the program is assigned to either the bronze, silver or gold category. Customers do not need to progress through the levels in order; for instance, a customer can go directly from being a bronze category member to being a gold category member. Users can also slip; for instance, a customer can go from being a silver category member to being a bronze category member. Let b_t, s_t and g_t be number of customers in each of these populations in month t. Through data collection and analysis, you have proposed a model for the rates at which customers transition between these class from month to month. You can express your model using the following matrix equation:

$$\begin{bmatrix} b_{t+1} \\ s_{t+1} \\ g_{t+1} \end{bmatrix} = \begin{bmatrix} 0.4 & 0.6 & 0.1 \\ 0.6 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.8 \end{bmatrix} \begin{bmatrix} b_t \\ s_t \\ g_t \end{bmatrix}$$

$$\mathbf{x}_{t+1} = C\mathbf{x}_t$$

(a) Describe in words the meaning of entry (3,1) of C.

Entry (3,1) of C indicates that 10% of the bronze members become gold members each month.

(b) Find the eigenvalues and associated eigenvectors of C.

We can do this easily using Matlab.

```
>> [V,D] = eig([0.4 0.6 0.1; 0.6 0.2 0.2; 0.1 0.2 0.8])

V =

    0.6338    0.5168    0.5755
   -0.7692    0.3431    0.5390
    0.0811   -0.7843    0.6151


D =

   -0.3154         0         0
         0    0.6466         0
         0         0    1.0688
```

(c) In the long term, what percentage of the total number of customers enrolled in your loyalty program do you expect to be in the gold category?

The dominant eigenpair is $(\mathbf{v}_3, \lambda_3) = ([0.5755, 0.5390, .6151]^T, 1.0688)$. Since the dominant eigenvalue is greater than 1, the total number of customers is increasing over time. Moreover, after several months the customer populations will converge to a scaled version of the dominant eigenvector. Therefore, the percentage of the total number of customers that are gold status will be roughly 0.6151 / (0.5755 + 0.5390 + .6151) = 35.56%.

B.9 Studio 3.3 solutions

(0) Define

$$C = \begin{bmatrix} 13 & -4 \\ -4 & 7 \end{bmatrix}.$$

(a) Use the characteristic equation to determine the eigenvalues of C.

The characteristic equation is

$$\begin{aligned} \det(C - \lambda I) &= \det \begin{bmatrix} 13-\lambda & -4 \\ -4 & 7-\lambda \end{bmatrix} \\ &= (13-\lambda)(7-\lambda) - (-4)(-4) \\ &= \lambda^2 - 20\lambda + 75 \\ &= (\lambda - 15)(\lambda - 5). \end{aligned}$$

So the eigenvalues of C are $\lambda_1 = 15$ and $\lambda_2 = 5$.

(b) Using the eigenvalues you found in the previous part, compute the eigenvectors of the matrix.

To find an eigenvector $\mathbf{v}_1 = [v_1, v_2]^T$ corresponding to λ_1, we need to find a nontrivial solution to the homogeneous equation

$$(C - \lambda I)\mathbf{v}_1 = \mathbf{0}.$$

We can use Matlab for this.

```
EDU>> C = [13 -4; -4 7]

C =

    13    -4
    -4     7

EDU>> rref(C - 15*eye(2))

ans =

     1     2
     0     0
```

So, reading off the first line of the row reduced matrix, we have $v_1 = -2v_2$ and so $\mathbf{v}_1 = [v_1, v_2]^T = [-2v_2, v_2]^T = v_2[-2, 1]^T$. Taking $v_2 = 1$, which is valid because v_2 is free, we have $\mathbf{v}_1 = [-2, 1]^T$.

We follow a similar process to find $\mathbf{v}_2 = [v_1, v_2]^T$.

```
EDU>> rref(C - 15*eye(2))

ans =

     1     2
     0     0

EDU>> rref(C - 5*eye(2))

ans =

    1.0000   -0.5000
         0         0
```

So $v_1 = (1/2)v_2$. This implies that $\mathbf{v}_2 = [v_1, v_2]^T = [(1/2)v_2, v_2]^T = v_2[1/2, 1]^T$. Setting $v_2 = 1$ gives the eigenvector $\mathbf{v}_2 = [1/2, 1]^T$.

(1) Consider an animal species that has two life stages: juvenile (up to 1 year old) and adult. Suppose that females give birth each year to an average of 1.6 female juveniles. Each year, 30% of the juveniles survive to become adults, and 80% of the adults survive.

(a) Construct a stage-based population model for this species. Develop a matrix T that links the populations in year t, $\mathbf{x}_t$ to the populations in year $t+1$, $\mathbf{x}_{t+1}$ via $\mathbf{x}_{t+1} = T\mathbf{x}_t$.

Let $\mathbf{x}_t = [j_t, a_t]^T$, where j_t and a_t are the numbers of juveniles and adults present at time t, respectively. Using this ordering, we can write the associated transition matrix.

$$T = \begin{bmatrix} 0 & 1.6 \\ 0.30 & .80 \end{bmatrix}.$$

(b) What is the ratio of adults to juveniles in the long term?

In the long term, the population vector will be approximately equal to a scalar multiple of the dominant eigenvector. Matlab can easily compute the eignpairs of T.

```
EDU>> T = [0 1.6; 0.3 0.8]

T =

         0    1.6000
    0.3000    0.8000

EDU>> [V,D] = eig(T)

V =

   -0.9701   -0.8000
    0.2425   -0.6000


D =

   -0.4000         0
         0    1.2000
```

So the dominant eigenvalue is $\lambda = 1.2$ and has associated dominant eigenvector $[0.8, 0.6]^T$. To the ratio of juveniles to adults in the long term is $0.8/0.6 = 4/3$. So there are approximately 1.3 juveniles for every adult.

(c) Suppose now that the average number of female offspring an adult bears each year is represented by a parameter k. Use the characteristic equation of the matrix T_k to find the dominant eigenpair of the system. What is the effect of k on the eigenvalues?

Our transition matrix now reads

$$T = \begin{bmatrix} 0 & k \\ 0.30 & .80 \end{bmatrix}.$$

The characteristic equation is

$$\begin{aligned} \det(T - \lambda I) &= \det \begin{bmatrix} -\lambda & k \\ 0.30 & .80 - \lambda \end{bmatrix} \\ &= -\lambda(0.8 - \lambda) - (0.3)k \end{aligned}$$

We can find the roots of the characteristic equation using WolframAlpha or some other symbolic calculator.

$$\lambda_{1,2} \approx 0.4 \pm 0.5\sqrt{1.2k + 0.64}.$$

Assuming that k is positive, the argument of the square root function will always be positive, so we never have complex roots. For very small k, both eigenvalues are greater than 1. As k is increased, one eigenvalue increases while the other decreases. The dominant eigenvalue will be $\lambda_1 \approx 0.4 + 0.5\sqrt{1.2k + 0.64}$ for all values of k.

(d) What is the long term ratio of adults to in the preceding part? If the distribution depends on k, be sure to clearly indicate how.

We could always muscle through a problem like this by hand, but it's much easier (and more accurate) to use a symbolic calculator like WolframAlpha. Using the command `Eig[{{0,k},{0.3,0.8}}]` gives a dominant eigenvector

$$\mathbf{v}_1 \approx \begin{bmatrix} -1.33 + 1.66\sqrt{0.64 + 1.2k} \\ 1 \end{bmatrix}$$

When $k = 0$ (i.e., the adults are not reproducing), this eigenvector actually predicts that there will be a *negative* number of juveniles for each adult, and we should interpret this as zero juveniles for each adult. For $k > 0.002$, we have a positive number of juveniles for each 1 adult. This ratio is $-1.33 + 1.66\sqrt{0.64 + 1.2k}$. So the ratio of juveniles to adults grows as $\sqrt{k}$.

(e) Now let ℓ represent the percentage of juveniles that survive to adulthood, and assume as before that each female adult bears on average 1.6 female offspring per year. Use the characteristic equation of the matrix T_ℓ to find the dominant eigenpair of the system. What is the affect of ℓ on the eigenvalues?

Our transition matrix now reads

$$T = \begin{bmatrix} 0 & 1.6 \\ \ell & .80 \end{bmatrix}.$$

The characteristic equation is

$$\begin{aligned}\det(T - \lambda I) &= \det \begin{bmatrix} -\lambda & 1.6 \\ \ell & .80 - \lambda \end{bmatrix} \\ &= -\lambda(0.8 - \lambda) - 1.6\ell.\end{aligned}$$

Using the command `Eig[{{0,1.6},{\ell,0.8}}]` in WolframAlpha gives the eignpairs of the system. The dominant eigenvalue is $\lambda_1 \approx 0.4 + 0.5\sqrt{6.4\ell + 0.64}$. Notice that if ℓ is positive, then the dominant eigenvalue is always positive, so the populations are growing together in the long term.

(f) What is the long term ratio of adults to in the preceding part? If the distribution depends on ℓ, be sure to clearly indicate how.

WolframAlpha also gives us the dominant eigenvector of the transition matrix via the command `Eig[{{0,1.6},{\ell,0.8}}]`.

$$\mathbf{v}_1 \approx \begin{bmatrix} \frac{-0.4+0.5\sqrt{6.4\ell+0.64}}{\ell} \\ 1 \end{bmatrix}.$$

Again, for $\ell = 0$ (i.e., no offspring are surviving to become adults), this prediction doesn't make much sense. But if some offspring are surviving, then we get a more reasonable prediction. Notice that the ratio of juveniles to adults grows in a more complicated way depending on ℓ than in the previous subproblem.

(2) Let $(\mathbf{v}, \lambda)$ be an eigenpair of an invertible matrix A. Show that $(\mathbf{v}, 1/\lambda)$ is an eigenpair of A^{-1}.

We need to confirm that $A^{-1}\mathbf{v} = (1/\lambda)\mathbf{v}$ given that $A\mathbf{v} = \lambda\mathbf{v}$ with A invertible. We know $A^{-1}A = I$ by the definition of the matrix inverse.

$$\begin{aligned} A\mathbf{v} &= \lambda\mathbf{v} \\ A^{-1}A\mathbf{v} &= \lambda A^{-1}\mathbf{v} \\ \mathbf{v} &= \lambda A^{-1}\mathbf{v}. \end{aligned}$$

For this equality to hold, it must be the case that $A^{-1}\mathbf{v} = (1/\lambda)\mathbf{v}$.

(3) Show that if λ is an eigenvalue of A, then λ is an eigenvalue of A^T. (Hint: consider how $A - \lambda I$ and $A^T - \lambda I$ are related.)

An eigenvalue λ of A is a root of the characteristic equation $\det(A - \lambda I)$. Note that $A^T - \lambda I = (A - \lambda I)^T$. Since $\det B = \det B^T$ for any square matrix B, we have that $\det(A - \lambda I) = \det((A - \lambda I)^T) = \det(A^T - \lambda I)$ so that the characteristic equations of A and A^T are the same. Hence, the roots of the characteristic equations of A and A^T are the same.

(a) Show that if λ is an eigenvalue of A^T, then λ is an eigenvalue of A.

To show this, we use the same reasoning as above, only backwards.

(4) We say that a matrix T is *column stochastic* if it has only non-negative entries and its columns each sum to 1. For an example, our transition matrix from the bike rental example

$$T = \begin{bmatrix} 0.8 & 0.4 \\ 0.2 & 0.6 \end{bmatrix}$$

is column stochastic.

(a) Using the characteristic equation, compute the eigenvalues of T.

The characteristic equation T is

$$\begin{aligned} \det(T - \lambda I) &= \det \begin{bmatrix} 0.8 - \lambda & 0.4 \\ 0.2 & 0.6 - \lambda \end{bmatrix} \\ &= (0.8 - \lambda)(0.6 - \lambda) - (0.4)(0.2) \\ &= \lambda^2 - 1.4\lambda + .4 \\ &= (\lambda - 1)(\lambda - 0.4). \end{aligned}$$

So the eigenvalues of T are $\lambda_1 = 1$ and $\lambda_2 = 0.4$.

(b) Using the eigenvalues you found in the previous part, compute the associated eigenvectors of T.

The eigenvector $\mathbf{v}_1$ associated with $\lambda_1 = 1$ is a nontrivial solution to $(T - I)\mathbf{v}_1 = \mathbf{0}$. We can use Matlab to solve this linear system.

```
EDU>> T = [0.8 0.4; 0.2 0.6];
EDU>> rref(T - eye(2))

ans =

     1    -2
     0     0
```

So the eigenvector $\mathbf{v}_1 = [v_1, v_2]^T$ must satisfy $v_1 = 2v_2$. So $\mathbf{v}_1 = [v_1, v_2]^T = [2v_2, v_2]^T = v_2[2, 1]^T$. By setting $v_2 = 1$, which is valid because v_2 is free in this context, we have $\mathbf{v}_1 = [2, 1]^T$.

We can follow a similar process to find the eigenvector $\mathbf{v}_2$ associated with $\lambda = 0.4$.

```
EDU>> rref(T - 0.4*eye(2))

ans =

     1     1
     0     0
```

So the eigenvector $\mathbf{v}_2 = [v_1, v_2]^T$ must satisfy $v_1 = -v_2$. So $\mathbf{v}_2 = [v_1, v_2]^T = [-v_2, v_2]^T = v_2[-1, 1]^T$. By setting $v_2 = 1$, which is valid because v_2 is free in this context, we have $\mathbf{v}_2 = [-1, 1]^T$.

(c) Show that any 2×2 column stochastic matrix T has $\lambda = 1$ as an eigenvalue. (Hint: let $T(1,1) = p$ and $T(2,2) = q$.)

Since the columns must each sum to 1, we have

$$T = \begin{bmatrix} p & 1-q \\ 1-p & q \end{bmatrix}.$$

Then the characteristic equation of T is

$$\begin{aligned} \det(T - lambdaI) &= \det \begin{bmatrix} p-\lambda & 1-q \\ 1-p & q-\lambda \end{bmatrix} \\ &= (p-\lambda)(q-\lambda) - (1-p)(1-q) \\ &= \lambda^2 - (p+q)\lambda + pq - 1 + p + q - pq \end{aligned}$$

If $\lambda = 1$ is a root of this equation, then $\lambda = 1$ is an eigenvalue of T. We can verify this by substitution.

$$\begin{aligned} (\lambda^2 - (p+q)\lambda + pq - 1 + p + q - pq)|_{\lambda=1} &= 1 - p - q + pq - 1 + p + q - pq \\ &= 0 \end{aligned}$$

So $\lambda = 1$ is indeed an eigenvalue of T.

(d) Show that for any column stochastic matrix T, the vector $[1, 1, \ldots, 1]^T$ is an eigenvector of T^T with eigenvalue $\lambda = 1$. Use a previous problem to conclude that $\lambda = 1$ is an eigenvalue of T. Comment on the relevance of this result to the long term distribution of a dynamical system with transition matrix T.

We know from a previous problem that if λ is an eigenvalue of T^T, then λ is an eigenvalue of T, too. Since the columns of T sum to 1, we have

$$T^T \mathbf{1} = \mathbf{1},$$

where $\mathbf{1}$ a vector with 1 in every component. So $\mathbf{1}$ is an eigenvector of T^T with associated eigenvalue $\lambda = 1$. Since $\lambda = 1$ is an eigenvalue of T^T, it is also an eigenvalue of T.

This result almost implies that any dynamical system that has as its transition matrix a column stochastic matrix has a steady state population distribution in the long term. I saw "almost" here because we would have to verify that the eigenvalue $\lambda = 1$ is in fact dominant in T before we can make the full conclusion. We'll see when this last condition holds in a future studio.

B.10 Studio 3.4 solutions

(0) Give the scaling factor r and the rotation angle ϕ for the following matrices:

$$A = \begin{bmatrix} \sqrt{3} & 1 \\ -1 & \sqrt{3} \end{bmatrix}, \quad B = \begin{bmatrix} 4 & -3 \\ 3 & 4 \end{bmatrix}, \quad C = \begin{bmatrix} 40 & -9 \\ 9 & 40 \end{bmatrix}.$$

Recall from class that for matrices of the form

$$\begin{bmatrix} a & b \\ -b & a \end{bmatrix}$$

the scaling factor is $r = |\lambda| = \sqrt{a^2 + b^2}$, and the rotation angle is $\phi = \sin^{-1}(b/r)$

(a) $r_A = \sqrt{\sqrt{3}^2 + 1^2} = \sqrt{4} = 2$ and $\phi_A = \sin^{-1}(1/2) = \pi/6$ radians or $360\phi_A/(2\pi) = 30$ degrees.

(b) $r_B = \sqrt{4^2 + 3^2} = \sqrt{25} = 5$ and $\phi_B = \sin^{-1}(3/5) = 0.6435$ radians or $360\phi_B/(2\pi) \approx 36.8699$ degrees.

(c) $r_C = \sqrt{40^2 + 9^2} = \sqrt{1681} = 41$ and $\phi_C = \sin^{-1}(9/41) = 0.22$ radians or $360\phi_C/(2\pi) \approx 12.68$ degrees.

(1) The population of spotted owls can be broken into three classes: juveniles, subadults and adults. These populations can be related to one another through the dynamical system

$$\begin{bmatrix} j_{t+1} \\ s_{t+1} \\ a_{t+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0.33 \\ 0.18 & 0 & 0 \\ 0 & 0.71 & 0.94 \end{bmatrix} \begin{bmatrix} j_t \\ s_t \\ a_t \end{bmatrix}$$
$$\mathbf{x}_{t+1} = T\mathbf{x}_t.$$

(a) What is the meaning of entry $T(3,2)$ in this model? What is the meaning of entry $T(1,3)$?

Entry $T(3,2)$ denotes the percentage of subadults surviving to become adults each year. Entry $T(1,3)$ denotes the average number of offspring per 1 adult.

(b) What is the long term fate of the population of owls? Is the ratio of the populations constant in the long term?

The long term behavior of the populations is determined by the dominate eigenpair.

```
EDU>> [V,D] = eig([0 0 0.33; 0.18 0 0; 0 0.71 0.94])

V =

   0.6821              0.6821              0.3175
  -0.0624 - 0.5896i   -0.0624 + 0.5896i    0.0581
  -0.0451 + 0.4256i   -0.0451 - 0.4256i    0.9465


D =

  -0.0218 + 0.2059i        0                   0
        0             -0.0218 - 0.2059i        0
        0                  0               0.9836
```

It may be unclear right off the bat whether the complex eigenvalues have norm (absolute value) greater than 1. We can easily test in Matlab.

```
EDU>> abs(-0.0218 + 0.2059i)

ans =

    0.2071
```

So the dominant eigenvalue is $\lambda = 0.9836$, and so the populations go extinct together in the long term. Since the dominant eigenvalue is real, there is no "rotation" in the behavior of the populations. The ratio of populations as the species declines approaches a steady state determined by the dominant eigenvector.

(c) Now assume that through concerted conservation efforts, the percentage of juveniles surviving to subadulthood has been increased to 50% from the original model. What is the long term fate of the population of owls? Is the ratio of the populations constant in the long term?

The new eigenvalues are

```
EDU>> [V,D] = eig([0 0 0.33; 0.5 0 0; 0 0.71 0.94])

V =

  -0.0774 + 0.4782i  -0.0774 - 0.4782i   0.2976
   0.7240             0.7240             0.1421
  -0.4660 - 0.1549i  -0.4660 + 0.1549i   0.9441


D =

  -0.0534 + 0.3302i        0                  0
        0            -0.0534 - 0.3302i        0
        0                  0             1.0469
```

Again, it may be unclear whether the complex eigenvalues dominant. We can verify in Matlab that they do not.

```
EDU>> abs(-0.0534 + 0.3302i)

ans =

    0.3345
```

So the populations grow together over time. Since the dominant eigenvalue is real, there is no "rotation" involved in their growth. In the long term, the ratio of populations approaches a steady state that is determined by the dominant eigenvector.

(2) One approach to conservation is through so called *stage-based population modeling.* For an example, imagine that American bison females can be divided into calves (up to 1 year old), yearlings (1 to 2 years old), and adults. Suppose on average 42 female calves are born each year per 100 adult females. (Only adults produce offspring.) Each year, about 60% of the calves survive, 75% of the yearlings survive, and 95% of the adults survive. For $t \geq 0$, let $\mathbf{x}_t = [c_t \;\; y_t \;\; a_t]^T$ be the population vector representing the females in the herd.

(a) Construct a matrix A for the herd so that $\mathbf{x}_{t+1} = A\mathbf{x}_t$ for $t \geq 0$.

The transition matrix for this dynamical system is

$$A = \begin{bmatrix} 0 & 0 & 0.42 \\ 0.62 & 0 & 0 \\ 0 & 0.75 & 0.95 \end{bmatrix}.$$

(b) Show that the buffalo herd is growing, determine the expected growth rate after many years, and give the expected number of calves and yearlings present per 100 adults.

The eigenpairs of the system are

```
EDU>> [V,D] = eig([0 0 0.42; 0.62 0 0; 0 0.75 0.95])

V =

  -0.0926 + 0.4802i  -0.0926 - 0.4802i   0.3475
   0.7225             0.7225             0.1943
  -0.4537 - 0.1816i  -0.4537 + 0.1816i   0.9173


D =

  -0.0794 + 0.4121i        0                  0
        0            -0.0794 - 0.4121i        0
        0                  0             1.1088
```

It may be unclear whether the complex eigenvalues are dominant. We can verify with Matlab that they are not.

```
EDU>> abs(-0.0794 + 0.4121i)

ans =

    0.4197
```

So the dominant eigenvalue of the matrix is $\lambda = 1.1088$ which indicates that the population of the herd is growing at a rate of 10% per year. In the long term, the ratios of populations in the herd will stabilize to those found in the dominant eigenvector, here $\mathbf{v} = [0.3475, 0.1943, 0.9173]^T$. So the number of calves and yearlings per 100 adults is $100(0.3475 + 0.1943)/0.9173 \approx 59$.

(3) For a 2×2 matrix A, we can find an interesting relationship between the entries A and its eigenvalues. We'll need an additional piece of terminology to complete this formulation. The *trace* of a matrix A is the sum of the entries along the main diagonal, that is, the sum of all entries in positions (i, i). So for an arbitrary 2×2 matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

the trace is $\tau = tr(A) = a+d$. Recall that the determinant of A is $\Delta = det(A) = ad - bc$.

(a) Show that the characteristic equation of A is $\lambda^2 - \tau\lambda + \Delta$.

The characteristic equation of A is

$$\begin{aligned} \det(A - \lambda I) &= \det \begin{bmatrix} a-\lambda & b \\ c & d-\lambda \end{bmatrix} \\ &= (a-\lambda)(d-\lambda) - bc \\ &= \lambda^2 - (a+d)\lambda + ad - bc \\ &= \lambda^2 - \tau\lambda + \Delta \end{aligned}$$

(b) Take a second to think about what this means: to understand the eigenvalues of a 2×2 matrix, we don't need to look at all 4 of the entries; we just need to consider two quantities, τ and Δ, that are related to the entries of the matrix. In essence, this cuts the complexity of the problem in half! (There's nothing to answer here. Just take a second to appreciate this fact.)

(c) Show that A has only real eigenvalues if and only if $\tau^2 \geq 4\Delta$.

Using the quadratic equation, the roots of $\lambda^2 - \tau\lambda + \Delta = 0$ are

$$\lambda_{1,2} = \frac{\tau \pm \sqrt{\tau^2 - 4\Delta}}{2}.$$

The only way the eigenvalues can be complex is if the argument of the square root is negative, but the argument is positive if $\tau^2 \geq 4\Delta$.

(d) Show that if A is the transition matrix of a dynamical system, then the populations go to zero in the long term if $\tau < 0$ and $\Delta > 0$.

If $\Delta > 0$, then $\sqrt{\tau^2 - 4\Delta} < |\tau|$. Then since $\tau < 0$, we have $\tau \pm \sqrt{\tau^2 - 4\Delta} < 0$. This implies that both of the eigenvalues are negative, and therefore all populations go extinct in the long term.

(4) Over the course of the next several parts, we'll show that any symmetric matrix has only real eigenvalues. This is known as the *spectral theorem.* This will also give us a chance to practice manipulating complex numbers.

(a) Let $\mathbf{v}$ be a vector with complex entries. Show that $\bar{\mathbf{v}}^T\mathbf{v}$ has only real entries.

For a more concrete example, we'll deal with a vector $\mathbf{v}$ with three components. The reasoning is exactly the same in the n component case. Expanding the original statement in terms of the components gives

$$\begin{aligned}\bar{\mathbf{v}}^T\mathbf{v} &= \begin{bmatrix}\bar{v}_1 & \bar{v}_2 & \bar{v}_3\end{bmatrix}^T \begin{bmatrix}v_1\\ v_2\\ v_3\end{bmatrix}\\ &= \begin{bmatrix}\bar{v}_1 v_1\\ \bar{v}_2 v_2\\ \bar{v}_3 v_3\end{bmatrix}.\end{aligned}$$

Notice that $\bar{v}_1 v_1 = |v_1|^2$ which is a real number. The same holds for the other components of the product. Thus, the inner product $\bar{\mathbf{v}}^T\mathbf{v}$ has only real entries.

(b) Now let $(\mathbf{v}, \lambda)$ be a (possibly complex) eigenpair of a symmetric matrix A. Show that $\bar{\mathbf{v}}^T A\mathbf{v} = \lambda\bar{\mathbf{v}}^T\mathbf{v}$.

We know that $A\mathbf{v} = \lambda\mathbf{v}$. Using this fact, we arrive at $\bar{\mathbf{v}}^T(A\mathbf{v}) = \bar{\mathbf{v}}^T(\lambda\mathbf{v}) = \lambda\bar{\mathbf{v}}^T\mathbf{v}$ as desired.

(c) Let $(\mathbf{v}, \lambda)$ be the same eigenpair of the symmetric A as in the last part. Show that $\bar{\mathbf{v}}^T A\mathbf{v} = \bar{\lambda}\bar{\mathbf{v}}^T\mathbf{v}$. (Hint: you'll have to use the fact that A is symmetric here.)

Recall that $(ABC)^T = C^T B^T A^T$ for matrices A, B, C. Applied to the quantity we're working with, we find $(\bar{\mathbf{v}}^T A\mathbf{v})^T = \mathbf{v}^T A^T\bar{\mathbf{v}}$. Since A is symmetric, $A^T = A$ so that $(\bar{\mathbf{v}}^T A\mathbf{v})^T = \mathbf{v}^T A\bar{\mathbf{v}}$.

From class we know that if $A\mathbf{v} = \lambda\mathbf{v}$, then $A\bar{\mathbf{v}} = \bar{\lambda}\bar{\mathbf{v}}$, too. Stated another way, eigenpairs appear in complex conjugate pairs. So $(\bar{\mathbf{v}}^T A\bar{\mathbf{v}})^T = \mathbf{v}^T A\bar{\mathbf{v}} = \bar{\lambda}\mathbf{v}^T\bar{\mathbf{v}}$. This implies that $\bar{\mathbf{v}}^T A\bar{\mathbf{v}} = (\bar{\lambda}\mathbf{v}^T\bar{\mathbf{v}})^T = \bar{\lambda}\bar{\mathbf{v}}^T\mathbf{v}$.

(d) Combine the last two parts to show that λ must be real. (Hint: consider the two equivalent ways to write $\bar{\mathbf{v}}^T A\mathbf{v}$ and combine this with fact that $\bar{\mathbf{v}}^T\mathbf{v}$ is real.)

We've show that $\bar{\mathbf{v}}^T A\mathbf{v} = \lambda\bar{\mathbf{v}}^T\mathbf{v}$, $\bar{\mathbf{v}}^T A\mathbf{v} = \bar{\lambda}\bar{\mathbf{v}}^T\mathbf{v}$ and that $\bar{\mathbf{v}}^T\mathbf{v}$ is a real number. Then

$$\begin{aligned}\lambda\bar{\mathbf{v}}^T\mathbf{v} &= \bar{\lambda}\bar{\mathbf{v}}^T\mathbf{v}\\ \lambda &= \bar{\lambda}\end{aligned}$$

But the only way that a complex number can equal its own conjugate is if it is in fact a real number, i.e., a complex number with zero imaginary component.

B.11 Studio 4.1 solutions

(0) Let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ be vectors in $\mathbb{R}^n$. Show that $span(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k)$ is a subspace of $\mathbb{R}^n$.

A vector is in the span if and only if it has the form $\mathbf{b} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_k\mathbf{v}_k$. Define $\mathbf{b}_1 = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_k\mathbf{v}_k$ and $\mathbf{b} = d_1\mathbf{v}_1 + d_2\mathbf{v}_2 + \ldots + d_k\mathbf{v}_k$. We can use the two step subspace test to verify that the span is subspace.

$$\begin{aligned}\mathbf{b}_1 + \mathbf{b}_2 &= (c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_k\mathbf{v}_k) + (d_1\mathbf{v}_1 + d_2\mathbf{v}_2 + \ldots + d_k\mathbf{v}_k) \\ &= (c_1 + d_1)\mathbf{v}_1 + (c_2 + d_2)\mathbf{v}_2 + \ldots + (c_k + d_k)\mathbf{v}_k \\ &= e_1\mathbf{v}_1 + e_2\mathbf{v}_2 + \ldots + e_k\mathbf{v}_k.\end{aligned}$$

So the sum of two vectors in the span is also in the span. We can complete the second step of the test.

$$\begin{aligned}\alpha\mathbf{b}_1 &= \alpha(c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \ldots + c_k\mathbf{v}_k) \\ &= \alpha c_1\mathbf{v}_1 + \alpha c_2\mathbf{v}_2 + \ldots + \alpha c_k\mathbf{v}_k \\ &= f_1\mathbf{v}_1 + f_2\mathbf{v}_2 + \ldots + f_k\mathbf{v}_k.\end{aligned}$$

So scalar multiples of a vector in the span are also in the span.

We conclude that all spans are subspaces.

(1) Let P_n be the collection of all polynomials of degree n and smaller. Show that P_n is a vector space.

A polynomial is in P_n if and only if it has the form $p(x) = c_0 + c_1x + \ldots + c_nx^n$. We can use the two step subspace test to verify that P_n is vector space. Define $p(x) = c_0 + c_1x + \ldots + c_nx^n$ and $q(x) = d_0 + d_1x + \ldots + d_nx^n$.

$$\begin{aligned}p(x) + q(x) &= (c_0 + c_1x + \ldots + c_nx^n) + (d_0 + d_1x + \ldots + d_nx^n) \\ &= (c_0 + d_0) + (c_1 + d_1)x + \ldots + (c_n + d_n)x^n \\ &= e_0 + e_1x + \ldots e_nx^n.\end{aligned}$$

So the sum of two elements from P_n is also in P_n. We can complete the second test of the test.

$$\begin{aligned}\alpha p(x) &= \alpha(c_0 + c_1x + \ldots + c_nx^n) \\ &= \alpha c_0 + \alpha c_1x + \ldots + \alpha c_nx^n \\ &= f_0 + f_1x + \ldots f_nx^n.\end{aligned}$$

So scalar multiples of elements from P_n is also in P_n. We conclude that P_n is a vector space.

(2) Let $M_{m\times n}$ be the collection of all $m \times n$ matrices. Show that $M_{m\times n}$ is a vector space. (Here, the matrices are the "vectors" of the vector space.)

The sum of two matrices M and N of size $m \times n$ is also a matrix of size $m \times n$, since matrix addition is performed component-wise. The scalar multiple αM is also a matrix of size $m \times n$ is also a matrix of size $m \times n$. All of the arithmetic rules associated with a vector space hold. We conclude $M_{m\times n}$ is a vector space.

(3) Define

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

(a) Find a collection of vectors that span $\ker(A)$.

A vector $\mathbf{x}$ is in the kernel of A if and only if $A\mathbf{x} = \mathbf{0}$. To find such vectors, we need only row reduce the coefficient matrix A.

```
>> A = [1 2 4 0; 0 1 3 -2];
>> rref(A)

ans =

     1     0    -2     4
     0     1     3    -2
```

Define $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$. Then the first line reads $x_1 = 2x_3 - 4x_4$ and the second line reads $x_2 = -3x_3 + 2x_4$. Substituting gives an expression for $\mathbf{x}$ in terms of two vectors.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 2x_3 - 4x_4 \\ -3x_3 + 2x_4 \\ x_3 \\ x_4 \end{bmatrix} = x_3 \begin{bmatrix} 2 \\ -3 \\ 1 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -4 \\ 2 \\ 0 \\ 1 \end{bmatrix}.$$

Since both x_3 and x_4 are free, we conclude that the $\ker A = span([2, -3, 1, 0]^T, [-4, 2, 0, 1]^T$.

(b) Find a collection of vectors that span $\ker(B)$.

We follow a similar procedure to the previous part.

```
>> B = [1 3 -4 -3 1; 0 1 -3 1 0; 0 0 0 0 0];
>> rref(B)

ans =

     1     0     5    -6     1
     0     1    -3     1     0
     0     0     0     0     0
```

Define $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5]^T$. Then the first line reads $x_1 = -5x_3 + 6x_4 - x_5$, the second line reads $x_2 = 3x_3 - x_4$, and the third line reads $0 = 0$. Substituting gives an expression for $\mathbf{x}$ in terms of three vectors.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -5x_3 + 6x_4 - x_5 \\ 3x_3 - x_4 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = x_3 \begin{bmatrix} -5 \\ 3 \\ 1 \\ 0 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} 6 \\ -1 \\ 0 \\ 1 \\ 0 \end{bmatrix} + x_5 \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Since x_3, x_4, and x_5 are free, we conclude that $\ker(B) = \mathit{span}([-5, 3, 1, 0, 0]^T, [6, -1, 0, 1, 0]^T, [-1, 0, 0, 0, 1]^T)$.

(4) Let H and K be two subspaces of a vector space of V. We define the *intersection* of H and K, denoted $H \cap K$, to be the collection of all vectors from V that are in both H and K.

(a) Show that $H \cap K$ is a subspace of V.

Let $\mathbf{x}$ and $\mathbf{y}$ both be members of the intersection $H \cap K$. We'll use the two step subspace test to verify that the intersection is a subspace of V. For the first step, we need to verify that $\mathbf{x} + \mathbf{y}$ is a member of $H \cap K$. But since H is a subspace of V, sums of vectors in H are also in H. So $\mathbf{x} + \mathbf{y}$ is also in H. A similar argument shows that $\mathbf{x} + \mathbf{y}$ is in K, too. Then since $\mathbf{x} + \mathbf{y}$ is in both H and K, it is also in the intersection $H \cap K$. An identical argument shows that $\alpha\mathbf{x}$ is in $H \cap K$. Since the intersection passes the two step test, we conclude that all intersections of vectors subspaces are themselves vector subspaces.

(b) Let's see a concrete example. Define

$$\mathbf{v}_1 = \begin{bmatrix} 5 \\ 3 \\ 8 \end{bmatrix}, \quad \mathbf{v}_2 = \begin{bmatrix} 1 \\ 3 \\ 4 \end{bmatrix}, \quad \mathbf{v}_3 = \begin{bmatrix} 2 \\ -1 \\ 5 \end{bmatrix}, \quad \mathbf{v}_4 = \begin{bmatrix} 0 \\ -12 \\ -28 \end{bmatrix}$$

Let $H = \mathit{span}(\mathbf{v}_1, \mathbf{v}_2)$ and $K = \mathit{span}(\mathbf{v}_3, \mathbf{v}_4)$. Then both H and K are planes in 3-dimensional space, both pass through the origin, and their intersection is a line. Write a short sentence explaining why the preceding statements are true.

The span of two vectors in 3-dimensional space is a plane. Two non-identical and non-parallel planes intersect in a line. Since any vector space must include the zero vector, and since both H and K are vector spaces, they cannot be parallel because they both share the zero vector. They are clearly not identical. So their intersection must be a line in the 3-dimensions.

(c) Now let's get more quantitative. If the intersection of H and K is a line, then it is the span of a single vector $\mathbf{w}$. Find this vector. (Hint: if $\mathbf{w}$ is in H, then $\mathbf{w} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2$, and if $\mathbf{w}$ is in K, then $\mathbf{w} = c_3\mathbf{v}_3 + c_4\mathbf{v}_4$.

We can follow our noses here.

$$\begin{aligned} c_1\mathbf{v}_1 + c_2\mathbf{v}_2 &= c_3\mathbf{v}_3 + c_4\mathbf{v}_4 \\ c_1\mathbf{v}_1 + c_2\mathbf{v}_2 - c_3\mathbf{v}_3 - c_4\mathbf{v}_4 &= \mathbf{0} \\ \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & -\mathbf{v}_3 & -\mathbf{v}_4 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} &= \mathbf{0} \\ A\mathbf{c} &= \mathbf{0}. \end{aligned}$$

So the coefficient vector $\mathbf{c}$ is the in of A. Then to find $\mathbf{c}$, it suffices to row reduce the coefficient matrix A. Matlab can help us with the computation.

```
>> A = [5 3 8;  1 3 4; -2 1 -5; 0 12 28]'

A =

     5     1    -2     0
     3     3     1    12
     8     4    -5    28

>> rref(A)

ans =

    1.0000         0         0   -3.3333
         0    1.0000         0    8.6667
         0         0    1.0000   -4.0000
```

So the coefficient vector is has the form

$$\mathbf{c} = \begin{bmatrix} 3.33c_4 \\ -8.66c_4 \\ 4c_4 \\ c_4 \end{bmatrix} = c_4 \begin{bmatrix} 3.33 \\ -8.66 \\ 4 \\ 1 \end{bmatrix}.$$

The fact that c_4 is free here is not a problem; in fact, it's to be expected. Since the intersection $H \cap K$ is the span of a single vector, we can write in an infinite number of different and equivalent ways. Take $c_1 = 1$. Then recall that $\mathbf{w} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2$ and $\mathbf{w} = c_3\mathbf{v}_3 + c_4\mathbf{v}_4$. We can confirm that we actually have found c_1, c_2, c_3, c_4 such that the two formulations of $\mathbf{w}$ are equivalent.

```
>> 3.33*[5;3;8] - 8.66*[1;3;4]

ans =
```

```
    7.9900
  -15.9900
   -8.0000


>> 4*[2;-1;5]+1*[0;-12;-28]

ans =

     8
   -16
    -8
```

(Note that there is some roundoff error here, because I've only used the first few decimals of some of the weights. The two answers are essentially equivalent.) So $H \cap K = span(\mathbf{w}) = span([8, -16, -8]^T)$.

(5) Let H and K be two subspaces of a vector space V. The *union* of H and K, denoted $H \cup K$, is collection of all vectors of V that are in H *or* K. Show that $H \cup K$ is not necessarily a subspace of V by given an example in which H and K are subspaces of $\mathbb{R}^2$.

Let $H = span([1, 0]^T)$ and $K = span([0, 1]^T)$. Then the union of the subspaces is all vectors of the for $c_1[1, 0]^T$ *or* $c_2[0, 1]^T$. But notice that the sum of two vectors in $H \cup K$ is not necessarily in the subspace. For instance $[1, 0]^T + [0, 1]^T = [1, 1]^T$ is not in H or K. Since the union $H \cup K$ fails the first step of the two step subspace test, it is not a subspace of $\mathbb{R}^2$.

(6) Let H and K be subspaces of a vector space V. The *sum* of H and K, denoted $H + K$ is the collection of vectors from V that can be written as the sum of a vector in H and a vector in K.

(a) Show that $H + K$ is a subspace of V.

Define $\mathbf{x}$ and $\mathbf{y}$ to be two vectors in the sum $H + K$. Then both $\mathbf{x}$ and $\mathbf{y}$ are sums of vectors in H and K. Then it follows directly that the sum of $\mathbf{x}$ and $\mathbf{y}$ are also sums of vectors in H and K. Said differently, $\mathbf{x} + \mathbf{y}$ is in $H + K$. An equivalent argument shows that $\alpha\mathbf{x}$ must be in $H + K$.

(b) Show that H is a subspace of $H + K$ and K is a subspace of $H + K$.

Let $\mathbf{x}$ and $\mathbf{y}$ be two vectors in H. Clearly the sum of the two vectors and scalar multiples of the vectors are in H, too, since H is a vector space in its own right. Since H is entirely include in $H + K$ via $H = H + \mathbf{0}$, where we're thinking of the zero vector as an element of K here, we conclude H is a subspace of $H + K$. An identical argument shows that K is a subspace of $H + K$.

(7) Let $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_5$ be the columns of matrix A, where

$$A = \begin{bmatrix} 5 & 1 & 2 & 2 & 0 \\ 3 & 3 & 2 & -1 & -12 \\ 8 & 4 & 4 & -5 & 12 \\ 2 & 1 & 1 & 0 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_4 \end{bmatrix}$$

(a) Are $\mathbf{a}_3$ and $\mathbf{a}_5$ in $im(B)$?

Determining whether these statements are true takes us way back to the beginning of the semester; these are fundamental questions about solutions to linear systems. Each vector is in the image of B if and only if $B\mathbf{x} = \mathbf{a}_3$ (or $\mathbf{a}_5$) has a valid solution $\mathbf{x}$. Ultimately this is a Matlab exercise.

```
>> B = [ 5 3 8 2; 1 3 4 1; 2 -1 -5 0]'

B =

     5     1     2
     3     3    -1
     8     4    -5
     2     1     0

>> rref([B [2;2;4;1]])

ans =

    1.0000         0         0    0.3333
         0    1.0000         0    0.3333
         0         0    1.0000         0
         0         0         0         0
```

We conclude that $\mathbf{a}_3$ is in $im(B)$, and in particular $\mathbf{a}_3 = 0.33\mathbf{a}_1 + 0.33\mathbf{a}_2$.

For $\mathbf{a}_5$, we perform a similar exercise.

```
>> rref([B [0;-12;12;-2]])

ans =

    1.0000         0         0    3.3333
         0    1.0000         0   -8.6667
         0         0    1.0000   -4.0000
         0         0         0         0
```

We conclude that $\mathbf{a}_5$ is in $im(B)$, and in particular $\mathbf{a}_5 = 3.33\mathbf{a}_1 - 8.66\mathbf{a}_2 - 4\mathbf{a}_4$.

(b) Find a collection of vectors that spans $ker(B)$.

A vector $\mathbf{x}$ is in the kernel of B if and only if $B\mathbf{x} = \mathbf{0}$. We can find solutions to the homogeneous equation by row reducing the coefficient matrix A.

```
>> rref(B)

ans =

     1     0     0
     0     1     0
     0     0     1
     0     0     0
```

Defining $\mathbf{x} = [x_1, x_2, x_3]^T$, we can read off the rows of the RREF as $x_1 = 0$, $x_2 = 0$, $x_3 = 0$. Therefore the only solution to the homogeneous equation is the trivial solution $\mathbf{x} = \mathbf{0}$. Hence, $ker(B) = span(\mathbf{0}) = \mathbf{0}$.

B.12 Studio 4.2 solutions

(0) Let $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^2$ and c be a scalar.

(a) Show that $\mathbf{u} \circ \mathbf{v} = \mathbf{v} \circ \mathbf{u}$.

This is pretty easy to show when the vectors are in $\mathbb{R}^2$.

$$\begin{aligned} \mathbf{u} \circ \mathbf{v} &= u_1 v_1 + u_2 v_2 \\ \mathbf{v} \circ \mathbf{u} &= v_1 u_1 + v_2 u_2. \end{aligned}$$

Since all the quantities are real numbers, multiplication commutes and so the two lines are equal.

Perhaps the easiest way to show this identity when the vectors live in $\mathbb{R}^n$ is to note that $\mathbf{u} \circ \mathbf{v} = \mathbf{u}^T \mathbf{v} = k = k^T = (\mathbf{u}^T \mathbf{v})^T = \mathbf{v}^T \mathbf{u} = \mathbf{v} \circ \mathbf{u}$, where k is a real number.

(b) Show that $(\mathbf{u} + \mathbf{v}) \circ \mathbf{w} = \mathbf{u} \circ \mathbf{w} + \mathbf{v} \circ \mathbf{w}$.

$$\begin{aligned} (\mathbf{u} + \mathbf{v}) \circ \mathbf{w} &= \left(\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \right) \circ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \\ &= \begin{bmatrix} u_1 + v_1 \\ u_2 + v_2 \end{bmatrix} \circ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \\ &= (u_1 + v_1) w_1 + (u_2 + v_2) w_2 \\ &= u_1 w_1 + u_2 w_2 + v_1 w_1 + v_2 w_2 \\ &= \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \circ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \circ \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \\ &= \mathbf{u} \circ \mathbf{w} + \mathbf{v} \circ \mathbf{w}. \end{aligned}$$

(c) Show that $(c\mathbf{u}) \circ \mathbf{v} = c(\mathbf{u} \circ \mathbf{v})$.

$$\begin{aligned} \left(c \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \right) \circ \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} &= \begin{bmatrix} cu_1 \\ cu_2 \end{bmatrix} \circ \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \\ &= cu_1 v_1 + cu_2 v_2 \\ &= c(u_1 v_1 + u_2 v_2) \\ &= c(\mathbf{u} \circ \mathbf{v}). \end{aligned}$$

(1) Define

$$\mathbf{u} = \begin{bmatrix} -3 \\ 7 \\ 4 \\ 0 \end{bmatrix} \qquad \mathbf{v} = \begin{bmatrix} 1 \\ -8 \\ 15 \\ -7 \end{bmatrix}.$$

(a) Calculate $\|\mathbf{u}\|$.

```
>> norm([-3;7;4;0])

ans =

    8.6023
```

(b) Calculate $\|\mathbf{v}\|$.

```
>> norm([1;-8;15;-7])

ans =

   18.4120
```

(c) Calculate $\|\mathbf{u} - \mathbf{v}\|$.

```
>> norm([-3;7;4;0]-[1;-8;15;-7])

ans =

   20.2731
```

(d) Show that $\|\mathbf{u} - \mathbf{v}\| = \|\mathbf{v} - \mathbf{u}\|$.

```
>> norm(-[-3;7;4;0]+[1;-8;15;-7])

ans =

   20.2731
```

(e) Show that $\|\mathbf{u} + \mathbf{v}\|^2 = \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2$ if $\mathbf{u}$ and $\mathbf{v}$ are orthogonal. (This is equivalent to the Pythagorean theorem in higher dimensions.)

$$\begin{aligned}\|\mathbf{u} + \mathbf{v}\|^2 &= (\mathbf{u} + \mathbf{v}) \circ (\mathbf{u} + \mathbf{v}) \\ &= \mathbf{u} \circ \mathbf{u} + 2\mathbf{u} \circ \mathbf{v} + \mathbf{v} \circ \mathbf{v}.\end{aligned}$$

If $\mathbf{u}$ and $\mathbf{v}$ are orthogonal, then $\mathbf{u} \circ \mathbf{v} = 0$, and so we can simplify the preceding statement.

$$\begin{aligned}\|\mathbf{u} + \mathbf{v}\|^2 &= \mathbf{u} \circ \mathbf{u} + \mathbf{v} \circ \mathbf{v} \\ &= \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2.\end{aligned}$$

(2) Let W be a subspace of $\mathbf{R}^n$ and let $W^\perp$ be the orthogonal complement of W. Show that the only vector in both W and $W^\perp$ is the zero vector.

Suppose that $\mathbf{x}$ is in both W and $W^{\perp}$. Then $\mathbf{x}$ is orthogonal to itself. Then by definition, $\mathbf{x} \circ \mathbf{x} = 0$. But by the definition of the inner product, $\mathbf{x} \circ \mathbf{x} = x_1^2 + x_2^2 + \ldots + x_n^2$. So $\mathbf{x} \circ \mathbf{x} = 0$ if and only if each one of the x_i is 0. This implies that $\mathbf{x}$ is the zero vector.

(3) Let W be a subspace of $\mathbf{R}^n$ and let $W^{\perp}$ be the orthogonal complement of W. Show that $(W^{\perp})^{\perp} = W$.

Every vector in $W^{\perp}$ is orthogonal to every vector in W. Similarly, every vector in $(W^{\perp})^{\perp}$ is orthogonal to every vector in $W^{\perp}$. But these are exactly the vectors in W by definition.

(4) Define A as found in `studio12.mat`. Find a collection of vectors that spans $\text{im}(A)^{\perp}$.

We first need to remember that $\text{im}(A)^{\perp} = \ker(A^T)$. Then any vector $\mathbf{x}$ in $\text{im}(A)^{\perp}$ is a solution to the homogeneous equation $A^T\mathbf{x} = \mathbf{0}$. We can find solutions to this equation by row reducing the coefficient matrix A^T.

```
>> load studio12.mat
>> rref(A')

ans =

     1     0     5     0     0
     0     1     1     0     0
     0     0     0     1     0
     0     0     0     0     1
     0     0     0     0     0
```

Defining $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5]^T$, the first line reads, $x_1 = -5x_3$, the second line reads $x_2 = -x_3$, the third line reads $x_4 = 0$, the fourth line reads $x_5 = 0$ and the fifth line reads $0 = 0$. Substituting gives an expression for $\mathbf{x}$ as the linear combination of a single vector.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} -5x_3 \\ -x_3 \\ 0 \\ 0 \\ x_5 \end{bmatrix} = x_3 \begin{bmatrix} -5 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Since x_3 is free, we conclude that $\text{im}(A)^{\perp} = \ker(A^T) = span([-5, -1, 0, 0, 0]^T)$.

B.13 Studio 4.3 solutions

(0) Define

$$\mathbf{u} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 2 \\ 6 \\ 7 \\ -1 \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} -3 \\ 2 \\ -4 \\ -5 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$

(a) Find the projection of $\mathbf{y}$ onto $U = span(\mathbf{u})$ using both the explicit calculation at the beginning of the section, and the more general method used to complete the regression examples. Confirm that these approaches yield the same result.

Using the technology developed at the beginning of the section, we have

$$proj_U\mathbf{y} = \frac{\mathbf{y} \circ \mathbf{u}}{\mathbf{u} \circ \mathbf{u}}\mathbf{u}.$$

We can use Matlab to do the more basic computations here.

```
>> u = [1;2;3;4];
>> y = [1;1;1;1];
>> y'*u

ans =

    10

>> u'*u

ans =

    30
```

So $proj_U\mathbf{y} = (1/3)\mathbf{u}$.

Using the later definition, we have $\hat{\mathbf{x}} = (A^TA)^{-1}A^T\mathbf{y}$, where $A = [\mathbf{u}]$.

```
>> A = [1;2;3;4];
>> xHat = inv(A'*A)*A'*y

xHat =

    0.3333
```

So $\hat{\mathbf{y}} = A\hat{\mathbf{x}} = 0.33\mathbf{u}$. Up to numerical precision, the answers are identical.

(b) Find the projection of $\mathbf{y}$ onto $V = span(\mathbf{u}, \mathbf{v})$. Is the coefficient of $\mathbf{u}$ in this projection the same as the coefficient of $\mathbf{u}$ in the previous part? We can use the formula $\hat{\mathbf{x}} = (A^TA)^{-1}A^T\mathbf{y}$ to find the coefficients of the projection.

```
>> A = [1 2 3 4; 2 6  7 -1]'

A =

     1     2
     2     6
     3     7
     4    -1

>> xHat = inv(A'*A)*A'*y

xHat =

    0.2680
    0.0633
```

So $\hat{\mathbf{y}} = A\hat{\mathbf{x}} = 0.268\mathbf{u}+0.0633\mathbf{v}$. Notice that the coefficient of $\mathbf{u}$ in this projection is not the same as the coefficient of $\mathbf{u}$ in the projection in the previous part.

(c) Find the projection of $\mathbf{y}$ onto $W = span(\mathbf{u}, \mathbf{v}, \mathbf{w})$.

We can use the formula $\hat{\mathbf{x}} = (A^TA)^{-1}A^T\mathbf{y}$ to find the coefficients of the projection.

```
>> A = [1 2 3 4; 2 6  7 -1; -3 2 -4 -5]'

A =

     1     2    -3
     2     6     2
     3     7    -4
     4    -1    -5

>> xHat = inv(A'*A)*A'*y

xHat =

    0.2235
    0.0721
   -0.0342
```

So $\hat{\mathbf{y}} = A\hat{\mathbf{x}} = 0.2235\mathbf{u} + 0.0721\mathbf{v} - 0.0342\mathbf{w}$. Notice that the coefficient of $\mathbf{u}$ in this projection is not the same as the coefficient of $\mathbf{u}$ in the projection in the previous part.

(d) Find a matrix A such that $im(A) = W$. Confirm that $\mathbf{y}-\hat{\mathbf{y}}$ is a member of $W^{\perp} = \ker A^T$.

The desired matrix is

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 2 & 6 & 2 \\ 3 & 7 & -4 \\ 4 & -1 & -5 \end{bmatrix}.$$

We can confirm that $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to W using Matlab.

```
>> A'*(A*xHat - y)

ans =

  1.0e-014 *

   -0.0333
    0.1887
   -0.0777
```

Notice that while the values here are not exactly zero, they are very, very small. We say that up to numerical precision, they are zero.

(1) Suppose we have (x, y) data points (-2,1), (-1,4), (0,3), (1,7), (2,4).

(a) Find the linear model that minimizes the sum of squared errors. Calculated the sum of squared errors by finding the norm $\|\mathbf{y} - \hat{\mathbf{y}}\|$.

Define $\hat{y} = \beta_0 + \beta_1 x$. We can write a matrix equation describing the model.

$$\begin{bmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 3 \\ 7 \\ 4 \end{bmatrix}$$

$$A\beta = \mathbf{y}.$$

We can use the least-squares formula $\beta = (A^T A)^{-1} A^T \mathbf{y}$ to solve for the unknown coefficients.

```
>> A = [1 1 1 1 1; -2 -1 0 1 2]'

A =

     1    -2
     1     -1
     1     0
     1     1
     1     2

>> betaHat = inv(A'*A)*A'*[1;4;3;7;4]

ans =

    3.8000
    0.9000
```

So $\hat{y} = 3.8 + 0.9x$ is the unique linear model that minimizes the sum of squared errors between its predictions and the given observations.

The sum of squared errors is the square of the distance between the original vector and the projection. Mathematically, we want to compute $\|\hat{\mathbf{y}} - \mathbf{y}\| = \|A\hat{\beta} - \mathbf{y}\|$.

```
>> norm([1;4;3;7;4] - A*betaHat)^2

ans =

   10.7000
```

(b) Find the quadratic model that minimizes the sum of squared errors. Calculated the sum of squared errors. Is it larger or smaller than the SSE in the previous part?

```
EDU>> A = [1 1 1 1 1; -2 -1 0 1 2; (-2)^2 (-1)^2 0^2 1^2 2^2]'

A =

     1    -2     4
     1    -1     1
     1     0     0
     1     1     1
     1     2     4

EDU>> betaHat = inv(A'*A)*A'*[1;4;3;7;4]

betaHat =

    4.8000
    0.9000
   -0.5000

EDU>> norm([1;4;3;7;4] - A*betaHat)^2

ans =

    7.2000
```

So $\hat{y} = 4.8 + 0.9x - 0.5x^2$ is the unique quadratic function that minimizes the sum of squared errors between its predictions and the observations. The error here is lower than the error in the previous example. This implies that the quadratic model is a better fit for the data than the linear model.

(c) Find the cubic model that minimizes the sum of squared errors. Calculated the sum of squared errors. Is it larger or smaller than the SSE in the previous part?

```
EDU>> A = [1 1 1 1 1; -2 -1 0 1 2; (-2)^2 (-1)^2 0^2 1^2 2^2; (-2)^3 (-1)^3 0 1 2^3]'

A =

     1    -2     4    -8
     1    -1     1    -1
     1     0     0     0
     1     1     1     1
     1     2     4     8

EDU>> betaHat = inv(A'*A)*A'*[1;4;3;7;4]

betaHat =
```

```
    4.8000
    1.7500
   -0.5000
   -0.2500

EDU>> norm([1;4;3;7;4] - A*betaHat)^2

ans =

    6.3000
```

So $\hat{y} = 4.8 + 1.75 - 0.5^2 - 0.25x^3$ is the unique cubic function that minimizes the sum of squared errors between its predictions and the observations.The error here is lower than the error in the previous example. This implies that the cubic model is a better fit for the data than the quadratic model.

(d) Find the quartic (fourth order) model that minimizes the sum of squared errors. Calculated the sum of squared errors. What does your SSE mean in this case?

```
EDU>> A = [1 1 1 1 1; -2 -1 0 1 2; (-2)^2 (-1)^2 0^2 1^2 2^2; ...
(-2)^3 (-1)^3 0 1 2^3; (-2)^4 (-1)^4 0 1 2^4]'

A =

     1    -2     4    -8    16
     1    -1     1    -1     1
     1     0     0     0     0
     1     1     1     1     1
     1     2     4     8    16

EDU>> betaHat = inv(A'*A)*A'*[1;4;3;7;4]

betaHat =

    3.0000
    1.7500
    3.3750
   -0.2500
   -0.8750

EDU>> norm([1;4;3;7;4] - A*betaHat)^2

ans =

   6.3582e-28
```

So $\hat{y} = 3 + 1.75x + 3.375x^2 - 0.25x^3 - 0.875x^4$ is the unique quadric function that minimizes the sum of squared errors between its predictions and the observations.The error in this case is 0 up to round off error. This means that the function $\hat{y}$ passes directly through all of the observed data.

(2) Sometimes a single independent variable isn't enough to create a dependable model of a given system. For an example, imagine that you run a small ice cream shop on the coast of Maine. There are two main drivers for your sales: daily temperature and median customer income. Suppose we have a model with two independent variables u, representing the average daily temperature in July in your town, and v, representing the median income of customers who purchased from you in July. You've been collecting data over several years. The results can be seen in Table B.1

Year	**Total Sales**	**Average Temp.**	**Median Income**
2009	27.93	86.92	30.11
2010	28.29	88.51	31.48
2011	29.70	88.01	32.03
2012	31.09	87.05	33.34
2013	33.11	89.15	34.45

Table B.1: Data collected for the total sales (thousands of dollars), average temperate (degrees Fahrenheit), and median household income (thousands of dollars) for July of the indicated year

(a) Find the best multlinear model $s(u, v) = \beta_0 + \beta_1 u + \beta_2 v$ for the given data?

The matrix equation for this situation is

$$\begin{aligned} \begin{bmatrix} \mathbf{1} & \mathbf{u} & \mathbf{v} \end{bmatrix} \beta &= \mathbf{s} \\ A\beta &= \mathbf{s}. \end{aligned}$$

Then we can try to directly solve for the coefficients β using the least-squares formula $\beta = (A^T A)^{-1} A^T \mathbf{s}.$,

```
A =

    1.0000   86.9200   30.1100
    1.0000   88.5100   31.4800
    1.0000   88.0100   32.0300
    1.0000   87.0500   33.3400
    1.0000   89.1500   34.4500

>> s = [27.93; 28.29; 29.70; 31.09; 33.11];
>> betaHta = inv(A'*A)*A'*s
```

```
betaHta =

   -0.5805
   -0.1169
    1.2664
```

This implies that $s(u,v) = -0.5805 - 0.1169u + 1.2664v$ is the unique multilinear model that minimizes the sum of squared errors between its predictions and the observations.

(b) We could also allow for the variables u and v to interact multiplicatively through the model $s(u,v) = \beta_0 + \beta_1 u + \beta_2 v + \beta_3 uv$. Find the best model of this form for the given data.

Entering the matrix all again would be a pain. Let's assume we have A in Matlab as defined above. We can grab a particular column of A, for instance, the third.

```
>> A(:,3)

ans =

   30.1100
   31.4800
   32.0300
   33.3400
   34.4500
```

Then uv is equivalent to `A(:,2).*A(:,3)` (Remember that the dot means to do the multiplication component-wise. Without the dot, Matlab will try to do matrix multiplication.)

We can define a new matrix A that is the coefficient matrix in this example.

```
>> A = [A A(:,2).*A(:,3)]

A =

  1.0e+003 *

    0.0010    0.0869    0.0301    2.6172
    0.0010    0.0885    0.0315    2.7863
    0.0010    0.0880    0.0320    2.8190
    0.0010    0.0871    0.0333    2.9022
    0.0010    0.0892    0.0345    3.0712
```

We put the old A next to the new column that represents uv and the assign this new matrix back to the variable named A. Notice that all entries in the matrix are multiplied by $1e3 = 1000$.

We can solve for the unknown coefficients in the usual way.

```
>> betaHta = inv(A'*A)*A'*s

betaHta =

  922.9760
  -10.6391
  -26.8579
    0.3204
```

So $s(u, v) = -922.9760 - 10.6391u - 26.8579v + 0.3204uv$ is the unique model of the given form that minimizes the sum of squared errors between its predictions and the observations. Notice how much introducing the term uv changed the result in this part from the result in the last part.

(c) An even more general model might be $s(u, v) = \beta_0 + \beta_1 u + \beta_2 v + \beta_3 uv + \beta_4 u^2 + \beta_5 v^2$. Find the best model of this form for the given data.

We can use a similar shortcut to the one used in the last problem to create the matrix A. Let's assume we have defined A as in the last problem. Then u^2 corresponds to `A(:,2).*A(:,2) = A(:,2).^2`. Notice that even when you want to square every component in a vector, you need to use `.^`. Otherwise, Matlab will try to do matrix multiplication. So our new matrix is

```
>> A = [A A(:,2).^2 A(:,3).^2]

A =

  1.0e+003 *

    0.0010    0.0869    0.0301    2.6172    7.5551    0.9066
    0.0010    0.0885    0.0315    2.7863    7.8340    0.9910
    0.0010    0.0880    0.0320    2.8190    7.7458    1.0259
    0.0010    0.0871    0.0333    2.9022    7.5777    1.1116
    0.0010    0.0892    0.0345    3.0712    7.9477    1.1868
```

We can try to solve for the unknown coefficients in the usual way.

```
>> betaHta = inv(A'*A)*A'*s
Warning: Matrix is close to singular or badly scaled.
         Results may be inaccurate. RCOND = 1.940349e-024.

betaHta =

  1.0e+003 *

    1.8093
   -0.0249
   -0.0406
```

```
    0.0007
    0.0000
   -0.0002
```

This error matrix makes me nervous. Let's check to make sure that the inverse $(A^TA)^{-1}$ actually exists.

```
>> inv(A'*A)
Warning: Matrix is close to singular or badly scaled.
         Results may be inaccurate. RCOND = 1.940349e-024.

ans =

  1.0e+015 *

    1.1094   -0.0253    0.0003    0.0000    0.0001   -0.0001
   -0.0253    0.0006   -0.0000   -0.0000   -0.0000    0.0000
    0.0003   -0.0000    0.0000    0.0000    0.0000   -0.0000
    0.0000   -0.0000    0.0000    0.0000    0.0000   -0.0000
    0.0001   -0.0000    0.0000    0.0000    0.0000   -0.0000
   -0.0001    0.0000   -0.0000   -0.0000   -0.0000    0.0000
```

Here too Matlab is telling us something is wrong. What's going on here? Let's take a step back and remember what we're trying to do. We want a solution $\hat{\mathbf{s}} = A\hat{\beta}$ that is closest to the observed sales data $\mathbf{s}$. But this is only necessary if $\mathbf{s}$ is not in the image of A to begin with. Note that we have 5 equations and 6 unknowns in this problem. So there are likely infinitely many solutions β for every $\mathbf{s}$. Let's check by RREFing the augmented matrix.

```
>> rref([A s])

ans =

  1.0e+004 *

    0.0001         0         0         0         0    1.9856   -0.3068
         0    0.0001         0         0         0   -0.0453    0.0086
         0         0    0.0001         0         0    0.0005   -0.0042
         0         0         0    0.0001         0    0.0001    0.0000
         0         0         0         0    0.0001    0.0002   -0.0001
```

Notice that the coefficient β_5 is free. So there are in fact infinitely many models of the form $s(u,v) = \beta_0 + \beta_1 u + \beta_2 v + \beta_3 uv + \beta_4 u^2 + \beta_5 v^2$ that perfectly fit the data!

(3) One of the tools used in data mining is logistic regression, which takes a collection of observations about certain probabilities and attempts to construct the underlying cumulative density function. The logistic function in this case is

$$\pi(x) = \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x}+1},$$

where β_0 and β_1 are the parameters to be estimated. Notice that $\pi(x)$ is between 0 and 1 for every x. Moreover, we see that $\pi(x) \to 1$ as $x \to \infty$ and $\pi(x) \to 0$ as $x \to -\infty$. So $\pi(x)$ seems like a pretty good candidate for a CDF. Then $\pi(x)$ is the probability that some random variable X has value less than or equal to x.

(a) For an example, let's turn our attention to the grade distribution in a typical core foundation class. The median grade should be a B-, which equations to a 2.6 grade points. In my classes, roughly 15% of students receive an A- or better, which equates to 3.6 grade points or higher. Roughly 5% of students fail the course, which equates to 1 grade point or lower. Find the best logistic model for the underlying cumulative distribution function.

Our $(x, \pi(x)$, where x is a grade point and $\pi(x)$ is the percentage of students that make less than that grade point, data points here are $(2.6, 0.5)$, $(3.6, 0.85)$ and $(1, 0.05)$. Recall that we can turn the logistic function into a linear combination of the unknown coefficients.

$$\begin{aligned}
\pi(x) &= \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1 x}+1} \\
&= \frac{1}{1+e^{-\beta_0-\beta_1 x}} \\
\pi(x) + \pi(x)e^{-\beta_0-\beta_1 x} &= 1 \\
e^{-\beta_0-\beta_1 x} &= \frac{1-\pi(x)}{\pi(x)} \\
\beta_0 + \beta_1 x &= \ln\left(\frac{\pi(x)}{1-\pi(x)}\right).
\end{aligned}$$

Then the linear system associated with this model and the given data points are

$$\begin{bmatrix} 1 & 2.6 \\ 1 & 3.6 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \ln\left(\frac{0.5}{1-0.5}\right) \\ \ln\left(\frac{0.85}{1-0.85}\right) \\ \ln\left(\frac{0.05}{1-0.05}\right) \end{bmatrix}$$
$$A\beta = \mathbf{y}.$$

We can use Matlab to attempt to solve directly for the unknown coefficients using $\hat{\beta} = (A^T A)^{-1} A^T \mathbf{y}$.

```
>> A = [1 1 1; 2.6 3.6 1]'
```

```
A =

    1.0000    2.6000
    1.0000    3.6000
    1.0000    1.0000

>> y = [log(0.5/(1-0.5)) log(0.85/(1-0.85)) log(0.05/(1-0.05))]'

y =

         0
    1.7346
   -2.9444

>> betaHat = inv(A'*A)*A'*y

betaHat =

   -4.7315
    1.8034
```

(b) According to your model, what percentage of students earn a C or better?

A C is 2 grade points. We can substitute $x = 2$ in order to determine the probability.

$$\begin{aligned}\pi(2) &= \frac{e^{-4.7315+1.8034(2)}}{e^{-4.7315+1.8034(2)}+1} \\ &\approx 0.245\end{aligned}$$

So approximately 25% of the class will make less than a C.

(c) According to your model, how many grade points should a student earn to be in the top 25% of the class?

This implies that $\pi(x) = 0.75$. We can substitute into our equation and solve for the desired grad point x.

$$\begin{aligned}-4.7315 + 1.8034x &= \ln\left(\frac{0.75}{1-0.75}\right) \\ x &= \frac{\ln\left(\frac{0.75}{1-0.75}\right) + 4.7315}{1.8034} \\ &\approx 3.23\end{aligned}$$

So a student must earn roughly a B+ to be in the top 25% of the class.

B.14 Studio 5.1 solutions

(0) Is the collection

$$\left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \end{bmatrix} \right\}$$

a basis for $\mathbb{R}^2$?

No, the vectors are not linearly independent. We can confirm this by identifying a free variable in the RREF of the matrix whose columns are the basis vectors.

```
EDU>> A = [1 1 2; 1 2 3];
EDU>> rref(A)

ans =

     1     0     1
     0     1     1
```

(a) Is the collection

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

a basis for $\mathbb{R}^4$?

Yes, we can confirm using the RREF of the matrix whose columns are the basis vectors that there are no inconsistencies (so the vectors span $\mathbb{R}^4$) and there are no free variables (so the vectors are linearly independent).

```
EDU>> B = [1 1 1 1; 1 2 3 4; 0 0 1 0; 1 0 0 1]'

B =

     1     1     0     1
     1     2     0     0
     1     3     1     0
     1     4     0     1

EDU>> rref(B)

ans =
```

```
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1
```

(1) Define A and B by

$$A = \begin{bmatrix} 1 & 6 & 16 & -40 \\ 2 & 5 & 11 & -31 \\ 3 & 4 & 6 & -22 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -2 \end{bmatrix}.$$

(a) Find a basis for the kernel of A.

We can solve the homogeneous equation $A\mathbf{x} = \mathbf{0}$ by row reducing the coefficient matrix A.

```
EDU>> A = [1 6 16 -40; 2 5 11 -31; 3 4 6 -22];
EDU>> rref(A)

ans =

     1     0    -2     2
     0     1     3    -7
     0     0     0     0
```

Defining $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$, the first line implies $x_1 = 2x_3 - 2x_4$, and the second line implies $x_2 = -3x_3 + 7x_4$.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 2x_3 - 2x_4 \\ -3x_3 + 7x_4 \\ x_3 \\ x_4 \end{bmatrix} = x_3 \begin{bmatrix} 2 \\ -3 \\ 1 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -2 \\ 7 \\ 0 \\ 1 \end{bmatrix}.$$

Since x_3 and x_4 are free, we can conclude that

$$\ker(A) = \mathit{span}\left(\begin{bmatrix} 2 \\ -3 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -2 \\ 7 \\ 0 \\ 1 \end{bmatrix} \right),$$

and so

$$\mathcal{B}_{ker} = \left\{ \begin{bmatrix} 2 \\ -3 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -2 \\ 7 \\ 0 \\ 1 \end{bmatrix} \right\}.$$

Notice that the kernel is a subspace of $\mathbb{R}^4$ and not $\mathbb{R}^3$ because the kernel is a subspace of the domain, not the range.

(b) Find a basis for the image of A.

A basis of the image of A is simply the pivot columns of A. Hence,

$$\mathcal{B}_{im} = \left\{ \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 6 \\ 5 \\ 4 \end{bmatrix} \right\}.$$

Notice that the image is a subspace of $\mathbb{R}^3$ and not $\mathbb{R}^4$ because the image is a subspace of the range, not the domain.

(c) What are the dimensions of the kernel and image of A?

Both the image and the kernel are 2-dimensional.

(d) Find a basis for the kernel of B.

We can solve the homogeneous equation $B\mathbf{x} = \mathbf{0}$ by row reducing the coefficient matrix B.

```
EDU>> B = [1 1 1 1 0; 0 1 1 0 0; 0 0 1 -1 -2]

B =

     1     1     1     1     0
     0     1     1     0     0
     0     0     1    -1    -2

EDU>> rref(B)

ans =

     1     0     0     1     0
     0     1     0     1     2
     0     0     1    -1    -2
```

Defining $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5]^T$, the first line implies $x_1 = -x_4$, and the second line implies $x_2 = -x_4 - 2x_5$, and the third line implies $x_3 = x_4 + 2x_5$.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} -x_4 \\ -x_4 - 2x_5 \\ x_4 + 2x_5 \\ x_4 \\ x_5 \end{bmatrix} = x_4 \begin{bmatrix} -1 \\ -1 \\ 1 \\ 1 \\ 0 \end{bmatrix} + x_5 \begin{bmatrix} 0 \\ -2 \\ 2 \\ 0 \\ 1 \end{bmatrix}.$$

Since x_3 and x_4 are free, we can conclude that

$$\ker(A) = span\left(\begin{bmatrix}-1\\-1\\1\\1\\0\end{bmatrix}, \begin{bmatrix}0\\-2\\2\\0\\1\end{bmatrix}\right),$$

and so

$$\mathcal{B}_{ker} = \left\{\begin{bmatrix}-1\\-1\\1\\1\\0\end{bmatrix}, \begin{bmatrix}0\\-2\\2\\0\\1\end{bmatrix}\right\}.$$

Notice that the kernel is a subspace of $\mathbb{R}^5$ and not $\mathbb{R}^3$ because the kernel is a subspace of the domain, not the range.

(e) Find a basis for the image of B.

A basis of the image of B is simply the pivot columns of B. Hence,

$$\mathcal{B}_{im} = \left\{\begin{bmatrix}1\\0\\0\end{bmatrix}, \begin{bmatrix}1\\1\\0\end{bmatrix}, \begin{bmatrix}1\\1\\1\end{bmatrix}\right\}.$$

Notice that the image is a subspace of $\mathbb{R}^3$ and not $\mathbb{R}^5$ because the image is a subspace of the range, not the domain.

(f) What are the dimensions of the kernel and image of B?

The kernel of B is 2-dimensional, while the image of B is 3-dimensional.

(2) Let $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ be a set of vectors in $\mathbb{R}^n$.
(a) Argue that if $k > n$, then $\mathcal{B}$ cannot be a basis for $\mathbb{R}^n$.

If $k > n$, then there are more vectors than there are components in each vector. If we stack the vectors of $\mathcal{B}$ up as the columns of a matrix A, we could say that A has more columns than it has rows. The number of pivots in a matrix is at most its number of rows and at most its number of columns. (Here the row restriction is tighter.) We can combine the last two ideas in order to claim that some column of A does not contain a pivot and is therefore a free variable. A free variable then implies that the columns of A, that is, the vectors of $\mathcal{B}$ are not linearly independent.

(b) Argue that if $k < n$, then $\mathcal{B}$ cannot be a basis for $\mathbb{R}^n$.

If $k < n$, then there are more components in each vector than there are vectors. If we stack the vectors of $\mathcal{B}$ up as the columns of a matrix A, we could say that A has more rows than it has columns. The number of pivots in a matrix is at most its number of rows and at most its number of columns. (Here the column restriction is tighter.) We can combine the last two ideas in order to claim that some row of A does not contain a pivot. Then there is some $\mathbf{b}$ for which the matrix equation $A\mathbf{x} = \mathbf{b}$ does not have a solution, that is, the vectors of $\mathcal{B}$ do not span $\mathbb{R}^n$.

(c) Conclude that any basis of $\mathbb{R}^n$ must have exactly n elements. (There's nothing to do here other than recognize that the previous two parts directly show this fact.)

(3) The standard basis of $\mathbb{R}^n$ is the collection $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$, where $\mathbf{e}_i$ has a 1 in component i and zero in all other components. We can represent a vector $\mathbf{x}$ as a natural linear combination of these basis elements. For instance, if $\mathbf{x}$ is in $\mathbb{R}^2$, we have

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= x_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= x_1\mathbf{e}_1 + x_2\mathbf{e}_2. \end{aligned}$$

We can extend the idea of the standard basis to other vector spaces.

(a) Let H be the vector space of 2×2 matrices. Come up with a best guess as to the standard basis of H. Show that the collection of elements that you propose is in fact a basis.

In the standard basis for $\mathbb{R}^n$ we have a basis that might seem natural because the weights of the basis vectors in description of a vector $\mathbf{x}$ are just the components of $\mathbf{x}$ itself.

Imagine we have a general matrix A.

$$\begin{aligned} A &= \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \\ &= a_1 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + a_2 \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + a_3 \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + a_4 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \end{aligned}$$

This seems like the equivalent to the standard basis in $\mathbb{R}^n$.

Let's define

$$\mathcal{B}_{2\times 2} = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \right\}.$$

We have shown above that these elements span the vector space of 2×2, because an arbitrary 2×2 matrix A can be written as a linear combination of the elements. We can also verify (either by inspection or through a few computations) that the elements are linearly independent, because no one element is a linear combination of the others.

(b) Let P_n be the vector space of polynomials of degree at most n. Come up with a best guess as to the standard basis of P_n. Show that the collection of elements that you propose is in fact a basis.

An arbitrary element in P_n has the form $p(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$. Again, a "natural" way to break this along the components. Define

$$\mathcal{B}_{P_n} = \left\{1, x, x^2, \ldots, x^n\right\}.$$

We have shown above that these elements span the vector space P_n, because an arbitrary polynomial $p(x)$ of degree at most n can be written as a linear combination of the elements. We can also verify (either by inspection or through a few computations) that the elements are linearly independent, because no one element is a linear combination of the others.

(4) Recall that a $n \times n$ matrix A is symmetric if $A(i,j) = A(j,i)$ for all $1 \leq i,j \leq n$.

(a) Find a basis for the vector space of 2×2 symmetric matrices.

We want a collection $\mathcal{B}$ of 2×2 symmetric matrices such that any 2×2 matrix A can be written as a linear combination of elements in $\mathcal{B}$. Any symmetric 2×2 matrix has the form

$$A = \begin{bmatrix} a_1 & a_2 \\ a_2 & a_3 \end{bmatrix}.$$

We can separate A into a linear combination of symmetric matrices

$$\begin{aligned} A &= \begin{bmatrix} a_1 & a_2 \\ a_2 & a_3 \end{bmatrix} \\ &= a_1 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + a_2 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + a_3 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \end{aligned}$$

Let's propose a basis according to these elements we've found.

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \right\}.$$

First, the elements of $\mathcal{B}$ are clearly in the space, that is, they are symmetric. The elements in $\mathcal{B}$ certainly span the space in question, exactly because we have written an arbitrary matrix A in the space as a linear combination of them. They're also linearly independent by inspection. We conclude that $\mathcal{B}$ is indeed a basis of the space. Notice that this implies that the space of 2×2 symmetric matrices is 3-dimensional.

(b) Find a basis for the vector space of 3×3 symmetric matrices. We want a collection $\mathcal{B}$ of 3×3 symmetric matrices such that any 3×3 matrix A can be written as a linear combination of elements in $\mathcal{B}$. Any symmetric 3×3 matrix has the form

$$A = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_2 & a_4 & a_5 \\ a_3 & a_5 & a_6 \end{bmatrix}$$

We can break this matrix up exactly as we did above and subsequently identify the constituent basis elements.

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right\}$$

First, the elements of $\mathcal{B}$ are clearly in the space, that is, they are symmetric. The elements in $\mathcal{B}$ certainly span the space in question, exactly because we have written an arbitrary matrix A in the space as a linear combination of them. They're also linearly independent by inspection. We conclude that $\mathcal{B}$ is indeed a basis of the space. Notice that this implies that the space of 3×3 symmetric matrices is 6-dimensional.

(c) How would your previous answers generalize to the vector space of $n \times n$ matrices?

The process we've followed above easily generalizes to $n \times n$, though it is harder to think about a succinct way to describe the basis vectors. There are n basis elements that come from entries along the main diagonal and $(n^2 - n)/2$ basis elements that off from the off-diagonal pairs. This gives a total of $(n^2+n)/2$ basis elements. We conclude that the space of symmetric $n \times n$ matrices has dimension $(n^2 + n)/2$.

B.15 Studio 5.2 solutions

(0) Define

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\}, \quad \mathcal{C} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

(a) Verify that $\mathcal{B}$ and $\mathcal{C}$ are bases for $\mathbb{R}^4$.

Perhaps the easiest way to verify that these collections of vectors are bases of $\mathbb{R}^4$ is to row reduce the B that has columns take from $\mathcal{C}$, and similar for a matrix C.

```
EDU>> B = [1 1 0 1; 1 2 0 0; 1 3 1 0; 1 4 0 1]
EDU>> rref(B)

ans =

     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1

EDU>> C = [1 1 1 1; 0 1 1 1; 0 0 1 1; 0 0 0 1];
EDU>> rref(C)

ans =

     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1
```

In both cases the RREF is the identity matrix. Since there is a pivot in every row, the vectors of $\mathcal{B}$ span $\mathbb{R}^4$. Since there is a pivot in every column, the vectors of $\mathcal{B}$ are linearly independent. Thus $\mathcal{B}$ is a basis of $\mathbb{R}^4$. Equivalent statements hold for $\mathcal{C}$.

(b) What are the $\mathcal{B}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$?

First note that the change of coordinates matrix $P_{\mathcal{E}\leftarrow\mathcal{B}}$ is just the matrix B we defined above. For shorthand, we'll just refer to the change of basis as B in what follows.

We're looking for a solution $[\mathbf{x}]_{\mathcal{B}}$ to $B[\mathbf{x}]_{\mathcal{B}} = \mathbf{x}$. Since the columns of B form a basis of $\mathbf{R}^4$, the matrix inverse B^{-1} exists. Therefore $[\mathbf{x}]_{\mathcal{B}} = B^{-1}\mathbf{x}$. Matlab will do the actual computation for us.

```
>> Bcoords = inv(B)*[5;6;7;8]

Bcoords =

    4.0000
    1.0000
   -0.0000
    0.0000
```

So the $\mathcal{B}$-coordinates of $\mathbf{x}$ are $[\mathbf{x}]_{\mathcal{B}} = [4, -1, 0, 0]^T$.

(c) Construct a matrix that changes coordinates from the standard basis $\mathcal{E}$ of $\mathbb{R}^4$ to the basis $\mathcal{B}$.

This matrix is in fact just the inverse B^{-1} we constructed in the last part, since $[\mathbf{x}]_{\mathcal{B}} = B^{-1}\mathbf{x} = B^{-1}[\mathbf{x}]_{\mathcal{E}}$.

(d) What are the $\mathcal{C}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$?

First note that the change of coordinates matrix $P_{\mathcal{E}\leftarrow\mathcal{C}}$ is just the matrix C we defined above. For shorthand, we'll just refer to the change of basis as C in what follows.

We're looking for a solution $[\mathbf{x}]_{\mathcal{C}}$ to $C[\mathbf{x}]_{\mathcal{C}} = \mathbf{x}$. Since the columns of C form a basis of $\mathbf{R}^4$, the matrix inverse C^{-1} exists. Therefore $[\mathbf{x}]_{\mathcal{C}} = C^{-1}\mathbf{x}$. Matlab will do the actual computation for us.

```
EDU>> Ccoords = inv(C)*[5;6;7;8]

ans =

    -1
    -1
    -1
     8
```

So the $\mathcal{C}$-coordinates of $\mathbf{x}$ are $[\mathbf{x}]_{\mathcal{C}} = [-1, -1, -1, 8]^T$.

(e) Construct a matrix that changes coordinates from the standard basis $\mathcal{E}$ of $\mathbb{R}^4$ to the basis $\mathcal{C}$.

This matrix is in fact just the inverse C^{-1} we constructed in the last part, since $[\mathbf{x}]_{\mathcal{C}} = C^{-1}\mathbf{x} = C^{-1}[\mathbf{x}]_{\mathcal{E}}$.

(f) Construct a matrix that changes coordinates from the basis $\mathcal{B}$ to basis $\mathcal{C}$.

Recall that $C[\mathbf{x}]_{\mathcal{C}} = \mathbf{x} = B[\mathbf{x}]_{\mathcal{B}}$. Rearranging gives us a convenient expression.

$$[\mathbf{x}]_{\mathcal{C}} = C^{-1}B[\mathbf{x}]_{\mathcal{B}}.$$

So the matrix $C^{-1}B$ is the one we're looking for. Note that in the expanded change of coordinate notation, we have $C^{-1}B = P^{-1}_{\mathcal{E}\leftarrow\mathcal{C}} P_{\mathcal{E}\leftarrow\mathcal{B}} = P_{\mathcal{C}\leftarrow\mathcal{E}} P_{\mathcal{E}\leftarrow\mathcal{B}} = P_{\mathcal{C}\leftarrow\mathcal{B}}$.

Matlab gives us a numerical expression for $C^{-1}B$.

```
>> inv(C)*B

ans =

     0    -1     0     1
     0    -1    -1     0
     0    -1     1    -1
     1     4     0     1
```

(g) Verify that the matrix you constructed in the previous subproblem coverts the $\mathcal{B}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$ to the $\mathcal{C}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$.

With $\mathbf{x} = [5, 6, 7, 8]^T$, we've shown $[\mathbf{x}]_{\mathcal{B}} = [4, 1, 0, 0]^T$ and $[\mathbf{x}]_{\mathcal{C}} = [-1, -1, -1, 8]^T$. Matlab confirms that the change of coordinate matrix $C^{-1}B$ does indeed convert $\mathcal{B}$-coordinates into $\mathcal{C}$-coordinates.

```
>> inv(C)*B*[4;1;0;0]

ans =

    -1
    -1
    -1
     8
```

(h) Construct a matrix that changes coordinates from the basis $\mathcal{C}$ to basis $\mathcal{B}$. Recall that $C[\mathbf{x}]_{\mathcal{C}} = \mathbf{x} = B[\mathbf{x}]_{\mathcal{B}}$. Rearranging gives us a convenient expression.

$$[\mathbf{x}]_{\mathcal{B}} = B^{-1}C[\mathbf{x}]_{\mathcal{C}}.$$

So the matrix $B^{-1}C$ is the one we're looking for. Note that in the expanded change of coordinate notation, we have $B^{-1}C = P^{-1}_{\mathcal{E}\leftarrow\mathcal{B}}P_{\mathcal{E}\leftarrow\mathcal{C}} = P_{\mathcal{B}\leftarrow\mathcal{E}}P_{\mathcal{E}\leftarrow\mathcal{C}} = P_{\mathcal{B}\leftarrow\mathcal{C}}$.

(i) Verify that the matrix you constructed in the previous subproblem coverts the $\mathcal{C}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$ to the $\mathcal{B}$-coordinates of $\mathbf{x} = [5, 6, 7, 8]^T$.

With $\mathbf{x} = [5, 6, 7, 8]^T$, we've shown $[\mathbf{x}]_{\mathcal{B}} = [4, 1, 0, 0]^T$ and $[\mathbf{x}]_{\mathcal{C}} = [-1, -1, -1, 8]^T$. Matlab confirms that the change of coordinate matrix $B^{-1}C$ does indeed convert $\mathcal{C}$-coordinates into $\mathcal{B}$-coordinates.

```
>> inv(B)*C*[-1; -1; -1; 8]

ans =

    4.0000
    1.0000
   -0.0000
    0.0000
```

(1) We'll show that the mapping of $\mathbf{x}$ to its coordinates in a given basis $\mathcal{B}$ is a linear transformation. Any linear transformation f satisfies two properties: $f(x+y) = f(x) + f(y)$ and $f(cx) = cf(x)$ for any scalar c.

(a) Let $\mathbf{u}$ and $\mathbf{v}$ be two vectors in a vector space V with basis $\mathcal{B} = \{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n\}$. Show that $[\mathbf{u}+\mathbf{v}]_\mathcal{B} = [\mathbf{u}]_\mathcal{B} + [\mathbf{v}]_\mathcal{B}$.

Since $\mathcal{B}$ is a basis of V, we can write both $\mathbf{u}$ and $\mathbf{v}$ in terms of the basis vectors of $\mathcal{B}$.

$$\begin{aligned}\mathbf{u} &= u_1\mathbf{b}_1 + u_2\mathbf{b}_2 + \ldots + u_n\mathbf{b}_n \\ \mathbf{v} &= v_1\mathbf{b}_1 + v_2\mathbf{b}_2 + \ldots + v_n\mathbf{b}_n.\end{aligned}$$

Using these expressions, we can write the sum of the $\mathcal{B}$-coordinates of the two vectors.

$$\begin{aligned}[\mathbf{u}]_\mathcal{B} + [\mathbf{v}]_\mathcal{B} &= \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \\ &= \begin{bmatrix} u_1+v_1 \\ u_2+v_2 \\ \vdots \\ u_n+v_n \end{bmatrix}.\end{aligned}$$

We can also write the sum of the vectors.

$$\begin{aligned}\mathbf{u}+\mathbf{v} &= (u_1\mathbf{b}_1 + u_2\mathbf{b}_2 + \ldots + u_n\mathbf{b}_n) + (v_1\mathbf{b}_1 + v_2\mathbf{b}_2 + \ldots + v_n\mathbf{b}_n) \\ &= (u_1+v_1)\mathbf{b}_1 + (u_2+v_2)\mathbf{b}_2 + \ldots + (u_n+v_n)\mathbf{b}_n.\end{aligned}$$

But then notice that the $\mathcal{B}$-coordinates of the sum of the vectors matches the sum of the $\mathcal{B}$-coordinates of the vectors.

$$[\mathbf{u}+\mathbf{v}]_\mathcal{B} = \begin{bmatrix} u_1+v_1 \\ u_2+v_2 \\ \vdots \\ u_n+v_n \end{bmatrix}.$$

(b) Let $\mathbf{u}$ a vector in a vector space V with basis $\mathcal{B}$. Show that $[c\mathbf{u}]_\mathcal{B} = c[\mathbf{u}]_\mathcal{B}$.

Since $\mathcal{B}$ is a basis of V, we can write $\mathbf{u}$ in terms of the basis vectors of $\mathcal{B}$.

$$\mathbf{u} = u_1\mathbf{b}_1 + u_2\mathbf{b}_2 + \ldots + u_n\mathbf{b}_n.$$

Note that multiplying by a constant c just distributes to each of the basis terms.

$$\begin{aligned}c\mathbf{u} &= c(u_1\mathbf{b}_1 + u_2\mathbf{b}_2 + \ldots + u_n\mathbf{b}_n) \\ &= cu_1\mathbf{b}_1 + cu_2\mathbf{b}_2 + \ldots + cu_n\mathbf{b}_n.\end{aligned}$$

This directly implies that $[c\mathbf{u}]_\mathcal{B} = c[\mathbf{u}]_\mathcal{B}$.

(2) Define

$$\mathcal{B} = \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}, \quad \mathcal{C} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

Label the vectors of $\mathcal{B}$ in order as $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$, respectively. Similarly, label the vectors of $\mathcal{C}$ in order as $\mathbf{c}_1$, $\mathbf{c}_2$ and $\mathbf{c}_3$, respectively. We can think about the change of coordinates matrix $P_{\mathcal{C}\leftarrow\mathcal{B}}$ in a different way than the on presented in the text. In this problem, we'll walk through that process. Define $\mathbf{x} = [4, 5, 6]^T$.

(a) Write $\mathbf{x}$ as a linear combination of the vectors $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$.

Define the matrix B to have the vectors from $\mathcal{B}$ as its columns. Then $\mathcal{B}$-coordinates of $\mathbf{x}$ are $[\mathbf{x}]_{\mathcal{B}} = B^{-1}\mathbf{x}$. We can use Matlab to compute these coordinates.

```
EDU>> B = [1 1 0; 1 2 0; 1 3 1];
EDU>> Bcoords = inv(B)*[4;5;6]

Bcoords =

     3
     1
     0
```

This implies that $\mathbf{x} = 3\mathbf{b}_1 + \mathbf{b}_2$.

(b) Write each vector in $\mathcal{B}$ as a linear combination of the vectors $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$.

Define the matrix C to have the vectors from $\mathcal{C}$ as its columns. Then $\mathcal{C}$-coordinates of $\mathbf{b}_1$ are $[\mathbf{b}_1]_{\mathcal{C}} = C^{-1}\mathbf{b}_1$. We can use Matlab to compute these coordinates.

```
EDU>> C = [1 1 1; 0 1 1; 0 0 1];
EDU>> b1_C = inv(C) * [1;1;1]

b1_C =

     0
     0
     1
```

Notice that this is an easy one because $\mathbf{b}_1 = \mathbf{c}_3$, so $[\mathbf{b}_1]_{\mathcal{C}} = [0, 0, 1]^T$.

We can follow a similar procedure for the other two.

```
EDU>> b2_C = inv(C) * [1;2;3]

b2_C =
```

```
    -1
    -1
     3

EDU>> b3_C = inv(C) * [0;0;1]

b3_C =

     0
    -1
     1
```

We conclude that $\mathbf{b}_1 = \mathbf{c}_3$, $\mathbf{b}_2 = -\mathbf{c}_1 - \mathbf{c}_2 + 3\mathbf{c}_3$, $\mathbf{b}_3 = -\mathbf{c}_2 + \mathbf{c}_3$.

(c) Substitute your expressions for $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ in terms of $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$ into your expression for $\mathbf{x}$ in terms of $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$. (You should now have an expression for $\mathbf{x}$ in terms of $\mathbf{c}_1$, $\mathbf{c}_2$, and $\mathbf{c}_3$.) Regroup like terms and confirm that the $\mathcal{C}$-coordinates of $\mathbf{x}$ are the same here as you calculated above.

Earlier we found that $\mathbf{x} = 3\mathbf{b}_1 + \mathbf{b}_2$, so that the $\mathcal{B}$-coordinates of $\mathbf{x}$ are $[\mathbf{x}]_{\mathcal{B}} = [3, 1, 0]^T$. Then substituting in our new expression gives a formulation for $\mathbf{x}$ in terms of the vectors of $\mathcal{B}$.

$$\begin{aligned}\mathbf{x} &= 3\mathbf{b}_1 + \mathbf{b}_2\\ &= 3(\mathbf{c}_3) + (-\mathbf{c}_1 - \mathbf{c}_2 + 3\mathbf{c}_3)\\ &= -\mathbf{c}_1 - \mathbf{c}_2 + 6\mathbf{c}_3.\end{aligned}$$

We can conclude that the $\mathcal{C}$-coordinates of $\mathbf{x}$ are $[\mathbf{x}]_{\mathcal{C}} = [-1-1, 6]$.

(d) Take a second to note that we've changed from $\mathcal{B}$-coordinates to $\mathcal{C}$-coordinates by writing each vector of $\mathcal{B}$ in terms of a linear combination of the vectors in $\mathcal{C}$. (There's nothing to do here but make this realization.)

Noted.

(e) Confirm that numerically

$$P_{\mathcal{C}\leftarrow\mathcal{B}} = \begin{bmatrix} [\mathbf{b}_1]_{\mathcal{C}} & [\mathbf{b}_2]_{\mathcal{C}} & [\mathbf{b}_3]_{\mathcal{C}} \end{bmatrix}.$$

Since $C[\mathbf{x}]_{\mathcal{C}} = \mathbf{x} = B[\mathbf{x}]_{\mathcal{B}}$, we have $[\mathbf{x}]_{\mathcal{C}} = C^{-1}B[\mathbf{x}]_{\mathcal{B}}$, and so $P_{\mathcal{C}\leftarrow\mathcal{B}} = C^{-1}B$. Matlab gives a numerical expression for the change of coordinates matrix.

```
EDU>> inv(C)*B

ans =

     0    -1     0
     0    -1    -1
     1     3     1
```

Notice that the first column matches our computation of $[\mathbf{b}_1]_{\mathcal{C}}$, the second columns matches our computation of $[\mathbf{b}_2]_{\mathcal{C}}$, and the third column matches our computation of $[\mathbf{b}_3]_{\mathcal{C}}$.

(f) Argue why the identity in the preceding part must hold, and generalize this idea. (Hint: use the fact that coordinate mapping is a linear transformation.)

We'll argue that $[\mathbf{x}]_{\mathcal{C}} = \begin{bmatrix} [\mathbf{b}_1]_{\mathcal{C}} & [\mathbf{b}_2]_{\mathcal{C}} & [\mathbf{b}_3]_{\mathcal{C}} \end{bmatrix} [\mathbf{x}]_{\mathcal{B}}$. Let $[\mathbf{x}]_{\mathcal{B}} = [b_1, b_2, b_3]^T$, so that $\mathbf{x} = b_1\mathbf{b}_1 + b_2\mathbf{b}_2 + b_3\mathbf{b}_3$. Then we can use the fact that coordinate mapping is a linear transformation (which you showed in the previous problem) to derive the desired result.

$$\begin{aligned}
[\mathbf{x}]_{\mathcal{C}} &= [b_1\mathbf{b}_1 + b_2\mathbf{b}_2 + b_3\mathbf{b}_3]_{\mathcal{C}} \\
&= [b_1\mathbf{b}_1]_{\mathcal{C}} + [b_2\mathbf{b}_2]_{\mathcal{C}} + [b_3\mathbf{b}_3]_{\mathcal{C}} \\
&= b_1[\mathbf{b}_1]_{\mathcal{C}} + b_2[\mathbf{b}_2]_{\mathcal{C}} + b_3[\mathbf{b}_3]_{\mathcal{C}} \\
&= \begin{bmatrix} [\mathbf{b}_1]_{\mathcal{C}} & [\mathbf{b}_2]_{\mathcal{C}} & [\mathbf{b}_3]_{\mathcal{C}} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \\
&= \begin{bmatrix} [\mathbf{b}_1]_{\mathcal{C}} & [\mathbf{b}_2]_{\mathcal{C}} & [\mathbf{b}_3]_{\mathcal{C}} \end{bmatrix} [\mathbf{x}]_{\mathcal{B}}
\end{aligned}$$

This same approach generalizes to any pair of bases for $\mathbb{R}^n$.

B.16 Studio 5.3 solutions

(0) Define matrices A, B and E as found in `studio16.mat`. Determine whether each matrix is diagonalizable, orthogonally diagonalizable or neither. If diagonalizable, confirm that the eigenvectors are linearly independent. If orthogonally diagonalizable, confirm that the eigenvectors are orthogonal.

An $n \times n$ matrix A is diagonalizable if and only if its eigenvectors form a basis of $\mathbb{R}^n$. If P is a matrix whose columns are the eigenvectors of A, then A is diagonalizable if and only if the RREF of P is I_n.

An $n\times n$ matrix A is orthogonally diagonalizable if and only if it is symmetric. (Perhaps the quickest way to verify in Matlab that a matrix is symmetric is to compute $A^T - A$. If this matrix is the zeros matrix, then A is symmetric.) To verify that the eigenvectors are orthogonal, we verify that $P^TP = I_n$. Note that this also verifies that Matlab has scaled each eigenvector to have norm 1.

After loading `studio16.mat`, we have

```
>> A' - A

ans =

  1.0e-015 *

         0    0.2220   -0.4441
   -0.2220         0   -0.1110
    0.4441    0.1110         0
```

So A is orthogonally diagonalizable. We can verify that the eigenvectors of A are orthogonal.

```
>> [P,D] = eig(A);
>> P'*P

ans =

    1.0000    0.0000   -0.0000
    0.0000    1.0000    0.0000
   -0.0000    0.0000    1.0000
```

The negative signs here are due to roundoff error.

For B, we have

```
>> B

B =

    0.5190   -5.2964    3.6414
    0.2773    1.4586   -0.3316
   -0.6234   -4.3733    4.0224
```

The matrix B is clearly not symmetric. But it is diagonalizable:

```
>> [P,D] = eig(B)

P =

   -0.8120    0.8481    0.0571
   -0.0206    0.1180    0.5621
   -0.5832    0.5165    0.8251


D =

    3.0000         0         0
         0    2.0000         0
         0         0    1.0000

>> rref(P)

ans =

     1     0     0
     0     1     0
     0     0     1
```

For B, we have

```
>> E

E =

     3     0     0
     0     2     0
     1     0     3
```

The matrix E is clearly not symmetric. It is also not diagonalizable.

```
>> [P,D] = eig(E)

P =

         0    0.0000         0
         0         0    1.0000
    1.0000   -1.0000         0


D =
```

```
     3     0     0
     0     3     0
     0     0     2

>> rref(P)

ans =

     1    -1     0
     0     0     1
     0     0     0
```

Notice the free variable which implies the eigenvectors of E are not linearly independent.

(1) We say two matrices A and B are *similar* if there exists a matrix P such that $A = PBP^{-1}$.

(a) Show that matrix A from `studio16.mat` is similar to `D = diag([6,5,4])`. (Here D is the diagonal matrix with entries 6, 5, 4 in order along its main diagonal.)

We want a matrix P that diagonalizes A into D. But the only matrix that diagonalizes A is the matrix P whose columns are the eigenvectors of A. Notice that if $A = PDP^{-1}$ if and only if $P^{-1}AP = D$. Let's check that we get the correct D.

```
>> [P,D] = eig(A);
>> inv(P) * A * P

ans =

    6.0000    0.0000         0
   -0.0000    5.0000   -0.0000
    0.0000   -0.0000    4.0000
```

So A is similar to D.

(b) Show that matrix F from `studio16.mat` is similar to `D = diag([6,5,4])`.

We follow the same procedure as the first part.

```
>> [Q,D] = eig(F);
>> inv(Q)*F*Q

ans =

    6.0000    0.0000   -0.0000
    0.0000    4.0000   -0.0000
    0.0000         0    5.0000
```

Notice that A is similar to D and F is similar to D, but A is not equal to F.

(c) Show that matrices A and F are similar.
We want a matrix R such that $A = RFR^{-1}$. Notice that

$$P^{-1}AP = D = Q^{-1}FQ.$$

This implies that

$$A = PQ^{-1}FQP^{-1}.$$

Using the identity

$$(AB)^{-1} = B^{-1}A^{-1}$$

if A and B are appropriately sized and both invertible, we can define $R = QP^{-1}$ and claim

$$A = R^{-1}FR.$$

So A and F are similar.

(2) Let A be a symmetric $n \times n$ matrix.
(a) Show that A^2 is symmetric.

We need to verify that $(A^2)^T = A^2$. Recall that $(AB)^T = B^T A^T$. Here,

$$(A^2)^T = (AA)^T = A^T A^T = AA = A^2,$$

where $A^T = A$ comes from the assumption that A is symmetric. So A^2 is symmetric as well.

(b) A symmetric matrix A such that $A^2 = A$ is known as a *projection matrix*. Let $\mathbf{y} \in \mathbb{R}^n$, and define $\hat{\mathbf{y}} = A\mathbf{y}$. Show that $\mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to $\hat{\mathbf{y}}$.

We need to verify that $(\mathbf{y} - \hat{\mathbf{y}}) \circ \hat{\mathbf{y}} = 0$. Substituting and rearranging gives

$$\begin{aligned}(\mathbf{y} - \hat{\mathbf{y}}) \circ \hat{\mathbf{y}} &= (\mathbf{y} - A\mathbf{y}) \circ A\mathbf{y} \\ &= \mathbf{y} \circ A\mathbf{y} - A\mathbf{y} \circ A\mathbf{y}.\end{aligned}$$

Using the definition $\mathbf{v} \circ \mathbf{w} = \mathbf{v}^T\mathbf{w}$, we have

$$\begin{aligned}(\mathbf{y} - \hat{\mathbf{y}}) \circ \hat{\mathbf{y}} &= \mathbf{y}^T A\mathbf{y} - (A\mathbf{y})^T A\mathbf{y} \\ &= \mathbf{y}^T A\mathbf{y} - \mathbf{y}^T A^T A\mathbf{y}.\end{aligned}$$

Since A is symmetric $A^T = A$, and since A is a projection matrix $A^2 = A$. Combining these in sequence gives

$$\begin{aligned}(\mathbf{y} - \hat{\mathbf{y}}) \circ \hat{\mathbf{y}} &= \mathbf{y}^T A\mathbf{y} - \mathbf{y}^T A^T A\mathbf{y} \\ &= \mathbf{y}^T A\mathbf{y} - \mathbf{y}^T AA\mathbf{y} \\ &= \mathbf{y}^T A\mathbf{y} - \mathbf{y}^T A\mathbf{y} \\ &= 0.\end{aligned}$$

We conclude that $\hat{\mathbf{y}} = A\mathbf{y}$ is orthogonal to $\mathbf{y} - \hat{\mathbf{y}}$. Compare this to our work on projections to conclude that the name *projection matrix* is appropriate here.

(c) Explain why the previous part shows that any vector in $\mathbb{R}^n$ is a linear combination of a vector in the image of A and a vector in the orthogonal complement of the image of A.

Let $\mathbf{z} = \mathbf{y} - \hat{\mathbf{y}}$. Then $\mathbf{y} = \mathbf{z} + \hat{\mathbf{y}}$. The vector $\hat{\mathbf{y}}$ is in the image of A by definition, and we have just shown that $\mathbf{z}$ one vector in the image of A. But I don't think is quite enough to actually prove that $\mathbf{z}$ is orthogonal to *everything* in $im(A)$.

Instead, note that $im(A)^{\perp} = ker(A^T)$. We have

$$\begin{aligned} A^T(\mathbf{y} - \hat{\mathbf{y}}) &= A^T\mathbf{y} - A^TA\mathbf{y} \\ &= A\mathbf{y} - A^2\mathbf{y} \\ &= A\mathbf{y} - A\mathbf{y} \\ &= 0. \end{aligned}$$

Since $\mathbf{z} = \mathbf{y} - \hat{\mathbf{y}} \in ker(A^T)$, we can conclude that it is orthogonal to the image of A.

(3) Show that if A is diagonalizable and invertible, then A^{-1} is, too.

If A is diagonalizable, then $A = PDP^{-1}$ for some matrix P and diagonal matrix D. Recalling that $(ABC)^{-1} = C^{-1}B^{-1}A^{-1}$, we have

$$\begin{aligned} A^{-1} &= (PDP^{-1})^{-1} \\ &= (P^{-1})^{-1}D^{-1}P^{-1} \\ &= PD^{-1}P^{-1}. \end{aligned}$$

So A^{-1} is diagonalizable, and in fact, its associated matrices are P and D^{-1}.

(4) Construct a 2×2 matrix that is invertible but not diagonalizable.

Consider the matrix

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

We can confirm that A is invertible by verifying that $rref(A) = I_2$, and we confirm that A is not diagonalizable by verifying that $rref(P) \neq I_2$, where P is the matrix whose columns are the eigenvectors of A.

```
>> A = [1 0; -1 1]

A =

     1     0
    -1     1
```

```
>> rref(A)

ans =

     1     0
     0     1

>> [P,D] = eig(A);
>> rref(P)

ans =

     1     1
     0     0
```

(5) Construct a 2×2 matrix that is diagonalizable but not invertible.

Consider the matrix

$$B = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}.$$

We can confirm that B is not invertible by verifying that $rref(B) \neq I_2$, and we confirm that B is diagonalizable by verifying that $rref(P) = I_2$, where P is a matrix whose columns are the eigenvectors of B.

```
>> B = [1 2; 2 4]

B =

     1     2
     2     4

>> rref(B)

ans =

     1     2
     0     0

>> [P,D] = eig(B);
>> rref(P)

ans =
```

```
1      0
0      1
```

(6) Show that if A is invertible and orthogonally diagonalizable, then A^{-1} is, too.

We showed earlier that if A is invertible and diagonalizable, then A^{-1} is, too. So here we need only verify that the eigenvectors of A^{-1} are mutually orthogonal if the eigenvectors of A were. But notice that we showed

$$A^{-1} = PD^{-1}P^{-1},$$

which shows the eigenvectors of A and A^{-1} are the same, so that if the eigenvectors of A are mutually orthogonal, then so are the eigenvectors of A^{-1}.

B.17 Studio 5.4 solutions

(0) (a) Given a matrix X, explain by the Matlab command `B = X - ones(N,1)*mean(X)` returns a matrix B whose columns each have mean zero.

A good place to start here is to learn about the `mean` function in Matlab:

```
>> help mean
 MEAN   Average or mean value.
    For vectors, MEAN(X) is the mean value of the elements in X. For
    matrices, MEAN(X) is a row vector containing the mean value of
    each column.
```

Say for the sake of argument that X is $N \times 2$, so that X has only two columns, and suppose in addition the first column has mean 1 and the second column has mean 2. Then `mean(X) = [1,2]`.

Notice that

$$M = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} [1\ 2] = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ \vdots & \vdots \\ 1 & 2 \end{bmatrix}.$$

So each column in $B = X - M$ has zero mean exactly because we've subtracted the mean from every entry in each respective column. Similar logic applies to the general case.

(b) Suppose that $(\mathbf{v}_1, \lambda_1)$ and $(\mathbf{v}_2, \lambda_2)$ are two distinct eigenpairs of a matrix $A^T A$. Show that $A\mathbf{v}_1$ and $A\mathbf{v}_2$ are orthogonal.

We need to verify that $A\mathbf{v}_1 \circ A\mathbf{v}_2 = 0$.

$$\begin{aligned} A\mathbf{v}_1 \circ A\mathbf{v}_2 &= (A\mathbf{v}_1)^T A\mathbf{v}_2 \\ &= \mathbf{v}_1^T A^T A\mathbf{v}_2 \\ &= \mathbf{v}_1^T(\lambda_2\mathbf{v}_2) \\ &= \lambda_2(\mathbf{v}_1 \circ \mathbf{v}_2) \\ &= 0. \end{aligned}$$

We conclude that $A\mathbf{v}_1$ and $A\mathbf{v}_2$ are orthogonal.

(1) Check out the data in `pca_salary.csv`. The Matlab variable `textdata` contains the column headers describing the columns of the matrix `data` which contains observations relating CEO age to CEO pay.

(a) Use PCA to determine the first principal component and second principal components and their associated eigenvalues.

Define X to be the data matrix found in the file provided.

```
>> N = length(X(:,1));
>> B = X - ones(N,1)*mean(X);
>> S = 1/(N-1)*B'*B

S =

  1.0e+004 *

    0.0081    0.0253
    0.0253    4.8635

>> [P,D] = eig(S)

P =

   -1.0000    0.0052
    0.0052    1.0000


D =

  1.0e+004 *

    0.0079         0
         0    4.8636
```

So the first principal component is $[0.0052, 1.0]^T$ which has associated eigenvalue $\lambda_1 \approx 48,636$, and the second principal component is $[-1.00, 0.0052]^T$ which has associated eigenvalue $\lambda_2 \approx 79$.

(b) Write an interpretation of the first and second principal components.

The first principal component represents an increasing trend of salary with respect to age. Quantitatively, if a CEO's age increases by 1 year, the dominant trend predicts that her salary will increase $1/0.0052 \approx 192$ thousand dollars.

The second principal component represents a decreasing trend of salary with respect to age. Quantitatively, if a CEO's age increases by 1 year, the second principal component predicts that her salary will decrease by .0052 thousand dollars (that is, 5.2 dollars).

(c) How many principal components must you include in order to capture 90% of the total variance?

The first principal component is tremendously dominant in this data set, representing $\lambda_1/tr(S) = 99.84\%$ of the total variance of the data set.

(2) Check out the data in `pca_temp.csv`. The Matlab variable `textdata` contains the column headers describing the columns of the matrix `data` which contains observations relating latitude, longitude and average January temperature over a 30 year time period.

(a) Use Google to locate the mean longitude and latitude of this data set on a map. Longitude becomes more negative as you move west in this data set, and latitude becomes more positive as you move north in this data set.

Let X be the data matrix. The longitude and latitude are represented by the second and third columns of X, respectively.

```
>> mean(X)

ans =

   26.5179   38.9696  -90.9625
```

The mean position of the data is therefore (38.9696, -90.9625), which is located in Troy, Missouri. This is very near the geographic center of the country.

(b) Use PCA to determine the first principal component and second principal components and their associated eigenvalues.

```
>> N = length(X(:,1));
>> B = X - ones(N,1)*mean(X);
>> S = 1/(N-1)*B'*B

S =

  179.0179  -61.0276   -4.7943
  -61.0276   28.9287  -11.6538
   -4.7943  -11.6538  224.0020

>> [P,D]= eig(S)

P =

   -0.3344    0.9414    0.0434
   -0.9407   -0.3362    0.0459
   -0.0578    0.0255   -0.9980


D =

    6.5181         0         0
         0  200.6843         0
         0         0  224.7462
```

(c) Write an interpretation of the first and second principal components.

The first principal component is $[0.0434, 0.0459, -0.9980]^T$. This reflects a trend in which increasing latitude (that is, moving north) by 0.0459 degrees and decreasing longitude (that is, moving west) by 0.9980 degrees results in a small temperature increase, roughly 0.0434 degrees Fahrenheit. Since the movement west is much, much larger than the movement north, we can essentially conclude that there is a small increasing temperature trend as we move westward. This makes sense, as much of the west coast of the United States is relatively warm in January compared to the rest of the country.

The second principal component is $[0.9414, -0.3362, 0.0255]^T$. This reflects a trend in which decreasing latitude (that is, moving south) by 0.3362 degrees and increasing longitude (that is, moving east) by 0.0255 degrees results in a rather large temperature increase, roughly 0.9414 degrees Fahrenheit. Since the change the latitude is much larger than the change in latitude, we can conclude that there is a large increasing temperature trend as we move southward. This makes sense, because temperature generally increases as we move towards the equator.

(d) How many principal components must you include in order to capture 90% of the total variance?

We must include the first two principal components. The first captures only approximately $224/trace(S) = 0.5186$ percent of the variance.

(3) Check out the data in `pca_colleges.csv`. The Matlab variable `textdata` contains the column headers describing the columns of the matrix `data` which contains observations a number of attributes of colleges, including acceptance rate, average SAT score, and cost per year.

(a) Use PCA to determine the first principal component and second principal components and their associated eigenvalues.

```
>> X = data;
>> N = length(X(:,1))

N =

    50

>> B = X - ones(N,1)*mean(X);
>> S = 1/(N-1)*B'*B;
>> [P,D] = eig(S)

P =

    0.0024    0.9811   -0.1546    0.0528    0.1036    0.0011
   -0.0003   -0.1478   -0.6198   -0.3552    0.6540    0.2004
    1.0000   -0.0024   -0.0002   -0.0001   -0.0003   -0.0001
    0.0005    0.0752    0.7122   -0.1799    0.4367    0.5139
    0.0002    0.0128    0.2899   -0.2029    0.4229   -0.8341
    0.0000    0.0987    0.0280   -0.8930   -0.4381    0.0064


D =

  1.0e+008 *

    2.3306         0         0         0         0         0
         0    0.0000         0         0         0         0
         0         0    0.0000         0         0         0
         0         0         0    0.0000         0         0
         0         0         0         0    0.0000         0
         0         0         0         0         0    0.0000
```

(b) Write an interpretation of the first principal component.

There's a lot of information back in here. To make the numbers more tractable, let's multiply the first principal component by 10,000:

$$\mathbf{p}_1 = \begin{bmatrix} 23.6572 \\ -2.583 \\ 10,000 \\ 4.5146 \\ 1.5509 \\ 0.2528 \end{bmatrix}.$$

Since the third component of the principal vector is represents the price per year, let's interpret the other components of the principal component vector as "what we get for spending 10,000." For instance, for 10,000 you get a school whose admission rate is roughly 2.583% smaller. The average student scores roughly 23.6572 points higher on their SAT. Roughly 4.5% more of the students will come from the top 10% of their class. You see a modest increase of 1.55% of faculty members with PhDs. And perhaps most interesting, you see only a 0.2528% (don't multiple by 100 here. It's already taken into account) increase in graduation rate.

These numbers are pretty interesting. They show that there is little dependence of faculty profile or graduation chances for more expensive schools. The quality of students does increase, fairly substantially in fact. And more expensive schools tend to be more selective, though not all that much more.

(c) How many principal components must you include in order to capture 90% of the total variance?

The first principal component dominates the data by an extremely wide margin.