Orthogonality, Finished

Last class we looked at orthogonal bases, with $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m\}$ a basis that have $\mathbf{v}_i \cdot \mathbf{v}_j = 0$ for $i \neq j$. We found that the a coefficients in the expression

$$\mathbf{x} = a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots + a_m\mathbf{v}_m$$

could be calculated using dot products, with $a_i = \frac{\mathbf{x}\cdot\mathbf{v}_i}{\mathbf{v}_i\cdot\mathbf{v}_i}$, so

$$\begin{aligned}\mathbf{x} &= \frac{\mathbf{x}\cdot\mathbf{v}_1}{\mathbf{v}_1\cdot\mathbf{v}_1}\mathbf{v}_1 + \frac{\mathbf{x}\cdot\mathbf{v}_2}{\mathbf{v}_2\cdot\mathbf{v}_2}\mathbf{v}_2 + \cdots + \frac{\mathbf{x}\cdot\mathbf{v}_m}{\mathbf{v}_m\cdot\mathbf{v}_m}\mathbf{v}_m \\ &= \mathrm{Proj}_{\mathbf{v}_1}(\mathbf{x}) + \mathrm{Proj}_{\mathbf{v}_2}(\mathbf{x}) + \cdots + \mathrm{Proj}_{\mathbf{v}_m}(\mathbf{x}).\end{aligned}$$

It turns out that you can actually use a variant of this when the $\mathbf{v}_i$ don't form a basis of the whole space, but rather a basis of a SUBSPACE of whatever $\mathbb{R}^n$ we're working with. What we get is the projection of $\mathbf{x}$ onto a $V \subseteq \mathbb{R}^n$, which is simultaneously three things:

- the element in V closest to $\mathbf{x}$.
- the component of $\mathbf{x}$ that is in V.
- the vector $\mathbf{y}$ such that $\mathbf{x} - \mathbf{y}$ is orthogonal to V.

The actual calculation is

$$\begin{aligned}\mathrm{Proj}_V(\mathbf{x}) &= \frac{\mathbf{x}\cdot\mathbf{v}_1}{\mathbf{v}_1\cdot\mathbf{v}_1}\mathbf{v}_1 + \frac{\mathbf{x}\cdot\mathbf{v}_2}{\mathbf{v}_2\cdot\mathbf{v}_2}\mathbf{v}_2 + \cdots + \frac{\mathbf{x}\cdot\mathbf{v}_m}{\mathbf{v}_m\cdot\mathbf{v}_m}\mathbf{v}_m \\ &= \mathrm{Proj}_{\mathbf{v}_1}(\mathbf{x}) + \mathrm{Proj}_{\mathbf{v}_2}(\mathbf{x}) + \cdots + \mathrm{Proj}_{\mathbf{v}_m}(\mathbf{x}).\end{aligned}$$

This is a repeat of the theorem from the last set.

Theorem: If $\mathbf{x} \in \mathbb{R}^n$ and $V \subseteq \mathbb{R}^n$ has orthogonal basis $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \dots, \mathbf{v}_m\}$ then

$$\mathrm{Proj}_V(\mathbf{x}) = \mathrm{Proj}_{\mathbf{v}_1}(\mathbf{x}) + \mathrm{Proj}_{\mathbf{v}_2}(\mathbf{x}) + \cdots + \mathrm{Proj}_{\mathbf{v}_m}(\mathbf{x}).$$

Proving that this is true is surprisingly easy. Recall the section about projections. The value $\mathbf{z}$ is a projection of $\mathbf{x}$ onto $\mathbf{y}$ if $(\mathbf{x} - \mathbf{z})$ is orthogonal to $\mathbf{y}$ (this means that $\mathbf{z}$ has ALL of the $\mathbf{x}$ component in the $\mathbf{y}$ direction, all that's left is at right angles). We use the same principle. This time we need to confirm that

$$\big(\mathrm{Proj}_{\mathbf{v}_1}(\mathbf{x}) + \mathrm{Proj}_{\mathbf{v}_2}(\mathbf{x}) + \cdots + \mathrm{Proj}_{\mathbf{v}_m}(\mathbf{x}) - \mathbf{x}\big)$$

is orthogonal to all elements in V (which will simultaneously prove we've found the projection onto $V^{\perp}$!).

Recall that V is spanned by the $\mathbf{v}$ vectors. If we get orthogonality for all of them, we get orthogonality for V. So:

$$\begin{aligned}&\big(\mathrm{Proj}_{\mathbf{v}_1}(\mathbf{x}) + \mathrm{Proj}_{\mathbf{v}_2}(\mathbf{x}) + \cdots + \mathrm{Proj}_{\mathbf{v}_m}(\mathbf{x}) - \mathbf{x}\big)\cdot\mathbf{v}_i \\ &= \mathrm{Proj}_{\mathbf{v}_i}(\mathbf{x})\cdot\mathbf{v}_i - \mathbf{x}\cdot\mathbf{v}_i \\ &= \left(\frac{\mathbf{v}_i\cdot\mathbf{x}}{\mathbf{v}_i\cdot\mathbf{v}_i}\mathbf{v}_i\right)\cdot\mathbf{v}_i - \mathbf{x}\cdot\mathbf{v}_i \\ &= \frac{\mathbf{v}_i\cdot\mathbf{x}}{\mathbf{v}_i\cdot\mathbf{v}_i}\mathbf{v}_i\cdot\mathbf{v}_i - \mathbf{x}\cdot\mathbf{v}_i \\ &= \mathbf{x}\cdot\mathbf{v}_i - \mathbf{x}\cdot\mathbf{v}_i = 0\end{aligned}$$

and done.

We can calculate $\mathrm{Proj}_V(\mathbf{x})$ this way, then calculate $\mathrm{Proj}_{V^\perp}(\mathbf{x}) = \mathbf{x} - \mathrm{Proj}_V(\mathbf{x})$. One implication:

Theorem: For any $\mathbf{x} \in \mathbb{R}^n$ and $V \subseteq \mathbb{R}^n$,

$$\mathbf{x} = \mathrm{Proj}_V(\mathbf{x}) + \mathrm{Proj}_{V^\perp}(\mathbf{x}).$$

Example: Calculate the projection of $\begin{bmatrix} -2 \\ -4 \\ 9 \end{bmatrix}$ onto the subspace $U = \mathrm{Span}\left\{ \begin{bmatrix} -3 \\ -1 \\ 2 \end{bmatrix}, \begin{bmatrix} -2 \\ 4 \\ -1 \end{bmatrix} \right\}$.

Answer I got: $\begin{bmatrix} -4 \\ -6 \\ 5 \end{bmatrix}$, this makes the complement (projection onto $U^\perp$) $= \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix}$. Make sure the complement is orthogonal to U to double check the result.

Example: Calculate the projection of $\begin{bmatrix} -8 \\ 6 \\ -4 \\ 7 \end{bmatrix}$ onto the space $\mathrm{Span}\left\{ \begin{bmatrix} 1 \\ 1 \\ -2 \\ -3 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ -3 \\ 3 \end{bmatrix} \right\}$.

Answer should be $-\begin{bmatrix} 1 \\ 1 \\ -2 \\ -3 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \\ -3 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ -5 \\ 0 \end{bmatrix}$. Check it by finding the complement.

Example: Calculation the projection of $\begin{bmatrix} -1 \\ -1 \\ -1 \\ 1 \end{bmatrix}$ onto the space $\mathrm{Span}\left\{ \begin{bmatrix} 1 \\ 0 \\ -1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 1 \\ 0 \end{bmatrix} \right\}$.

Answer should be $\frac{1}{3}\begin{bmatrix} 1 \\ 0 \\ -1 \\ 1 \end{bmatrix} - \frac{1}{3}\begin{bmatrix} 1 \\ -1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{3} \\ -\frac{2}{3} \\ \frac{1}{3} \end{bmatrix}$. Check it by finding the complement.

Getting Orthogonal Bases: Gram-Schmidt

The Gram-Schmidt Algorithm:

Take a linearly independent set $B = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$. Start with an empty set $C = \{\}$. Repeat until B is empty.

1. Pick a vector in B, $\mathbf{v}_i$, remove it from B.
2. Subtract the projections of every vector in C from $\mathbf{v}_i$.
3. Put the resulting vector in C.

Repeating this until B is empty will result in C being an orthogonal set with the same span as the B set.

So, how does this work? Use $\mathbf{y}$ to identify the vectors in C, and $\mathbf{v}$ for the original vectors, in B.

First, $\mathbf{y}_1 = \mathbf{v}_1$.

Next, $\mathbf{y}_2 = \mathbf{v}_2 - \mathrm{Proj}_{\mathbf{y}_1}(\mathbf{v}_2)$. This means that

$$\begin{aligned}\mathbf{y}_2 \cdot \mathbf{y}_1 = \left(\mathbf{v}_2 - \frac{\mathbf{y}_1 \cdot \mathbf{v}_2}{\mathbf{y}_1 \cdot \mathbf{y}_1}\mathbf{y}_1\right) \cdot \mathbf{y}_1 &= \mathbf{v}_2 \cdot \mathbf{y}_1 - \frac{\mathbf{y}_1 \cdot \mathbf{v}_2}{\mathbf{y}_1 \cdot \mathbf{y}_1}\mathbf{y}_1 \cdot \mathbf{y}_1 \\ &= \mathbf{v}_2 \cdot \mathbf{y}_1 - \mathbf{y}_1 \cdot \mathbf{v}_2 \\ &= 0.\end{aligned}$$

So, our new addition to C is orthogonal to the other vector in C. The set C is orthogonal.

$$\mathbf{y}_3 = \mathbf{v}_3 - \mathrm{Proj}_{\mathbf{y}_2}(\mathbf{v}_3) - \mathrm{Proj}_{\mathbf{y}_1}(\mathbf{v}_3)$$

so

$$\begin{aligned}\mathbf{y}_3 \cdot \mathbf{y}_1 &= \mathbf{v}_3 \cdot \mathbf{y}_1 - \mathrm{Proj}_{\mathbf{y}_2}(\mathbf{v}_3) \cdot \mathbf{y}_1 - \mathrm{Proj}_{\mathbf{y}_1}(\mathbf{v}_3) \cdot \mathbf{y}_1 \\ &= \mathbf{v}_3 \cdot \mathbf{y}_1 - \frac{\mathbf{y}_2 \cdot \mathbf{v}_3}{\mathbf{y}_2 \cdot \mathbf{y}_2}\mathbf{y}_2 \cdot \mathbf{y}_1 - \frac{\mathbf{v}_3 \cdot \mathbf{y}_1}{\mathbf{y}_1 \cdot \mathbf{y}_1}\mathbf{y}_1 \cdot \mathbf{y}_1 \\ &= \mathbf{v}_3 \cdot \mathbf{y}_1 - 0 - \mathbf{v}_3 \cdot \mathbf{y}_1 \\ &= 0\end{aligned}$$

and the same result for $\mathbf{y}_3 \cdot \mathbf{y}_2$. The process continues until we're out of elements in B.

Example:

These tend to be lengthy. Make a serious effort to keep them ordered, and don't forget which values you're supposed to use to cancel the projections!

The B set is $\left\{ \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 5 \\ 2 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ -3 \\ 2 \end{bmatrix} \right\}$. Now convert it into an orthogonal basis for that subspace.

$$\text{First transfer:} \qquad C = \left\{ \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} \right\} \qquad B = \left\{ \begin{bmatrix} 0 \\ 5 \\ 2 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ -3 \\ 2 \end{bmatrix} \right\}.$$

The next one will require some manipulation:

$$\mathbf{y}_2 = \mathbf{v}_2 - \mathrm{Proj}_{\mathbf{y}_1}(\mathbf{v}_2) = \begin{bmatrix} 0 \\ 5 \\ 2 \\ 2 \end{bmatrix} - \frac{12}{6} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 5 \\ 2 \\ 2 \end{bmatrix} - 2 \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ 1 \\ 2 \\ 0 \end{bmatrix}.$$

$$\text{Second transfer:} \qquad C = \left\{ \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -2 \\ 1 \\ 2 \\ 0 \end{bmatrix} \right\} \qquad B = \left\{ \begin{bmatrix} 2 \\ 1 \\ -3 \\ 2 \end{bmatrix} \right\}.$$

The last one will require even more manipulation:

$$\begin{aligned} \mathbf{y}_3 = \mathbf{v}_3 - \mathrm{Proj}_{\mathbf{y}_1}(\mathbf{v}_3) - \mathrm{Proj}_{\mathbf{y}2}(\mathbf{v}_3) &= \begin{bmatrix} 2 \\ 1 \\ -3 \\ 2 \end{bmatrix} - \frac{6}{6} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} - \frac{-9}{9} \begin{bmatrix} -2 \\ 1 \\ 2 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} 2 \\ 1 \\ -3 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} -2 \\ 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ -1 \\ 1 \end{bmatrix}. \end{aligned}$$

$$\text{Third transfer:} \qquad C = \left\{ \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -2 \\ 1 \\ 2 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ -1 \\ 1 \end{bmatrix} \right\} \qquad B = \{\},$$

and done. Double check that the C set is orthogonal.

So, what happens if the set is not linearly independent? Here's what happens:

Example: Reduce the set $\left\{ \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} \begin{bmatrix} 3 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -2 \\ 0 \\ -2 \end{bmatrix} \right\}$ using Gram-Schmidt.

The result is a zero vector at one point. This is always the result of using Gram-Schmidt on a linearly dependent set.

An Application: Linear Regression

This is a particular use of the concept of projections, to statistics. The setup is somewhat cumbersome. We'll see what we can make of it.

The idea is to find a best fit for a set of data. We'll start with finding a linear best fit, a line that is as close to as many of the data points as possible. So, if we have the points

$$(x,y) = \quad (1,3), \quad (2,4), \quad (3,4)$$

our objective is to find a line $y = a_0 + a_1 x$ that is as close as possible to those points. If we were to try to solve for a SINGLE line that matched those points, the system would have

$$a_0 + a_1(1) = 3 \qquad a_0 + a_1(2) = 4 \qquad a_0 + a_2(3) = 4$$

which forms the system

$$[D|\mathbf{y}] = \left[\begin{array}{cc|c} 1 & 1 & 3 \\ 1 & 2 & 4 \\ 1 & 3 & 4 \end{array}\right] \longrightarrow \text{Linear Reduction} \longrightarrow \left[\begin{array}{cc|c} 1 & 1 & 2 \\ 0 & 1 & 1 \\ 0 & 0 & -1 \end{array}\right]$$

so no single solution. What we want then, is to get the $\mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}$ such that we minimize the error from the fit:

$$D\mathbf{a} - \mathbf{y} \quad = \quad \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} - \begin{bmatrix} 2 \\ 4 \\ 3 \end{bmatrix}.$$

So, how do we do that? Look at the error's expression. It is the difference between two terms. The first is a multiplication of a matrix with a column vector, so an element of the column space of the matrix. The second is just a single vector. So, what we're looking for is *minimizing the distance between a subspace of* $\mathbb{R}^3$ *and a vector*. The point in the column space of D closest to $\mathbf{y}$ will be $\text{Proj}_{\text{Col}(D)}(\mathbf{y})$. So, $D\mathbf{a}$ will be equal to the projection of $\mathbf{y}$ onto the column space of D. Unfortunately, the column vectors of A are not orthogonal. The procedure is:

- Apply Gram-Schmidt to the columns of D.
- Calculate the projection of $\mathbf{y}$ onto $\text{Col}(D)$.
- Get the solution to $D\mathbf{a} = \text{Proj}_{\text{Col}(D)}(\mathbf{y})$.

Example Questions:

Section 4.5: 7.bd)

Section 4.6: 1.b), 3.bd), 4.b) (use the projection method, not the indirect one, also, pay attention to the numbers. This one is not in the form we've seen)