# 5.5 GRAM–SCHMIDT PROCEDURE

As discussed in §5.4, orthonormal bases possess significant advantages over bases that are not orthonormal. The spaces $\Re^n$ and $\mathcal{C}^n$ clearly possess orthonormal bases (e.g., the standard basis), but what about other spaces? Does every finite-dimensional space possess an orthonormal basis, and, if so, how can one be produced? The ***Gram–Schmidt***[44] orthogonalization procedure developed below answers these questions.

Let $\mathcal{B} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be an arbitrary basis (not necessarily orthonormal) for an $n$-dimensional inner-product space $\mathcal{S}$, and remember that $\|\star\| = \langle \star | \star \rangle^{1/2}$.

**Objective:** Use $\mathcal{B}$ to construct an orthonormal basis $\mathcal{O} = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n\}$ for $\mathcal{S}$.

**Strategy:** Construct $\mathcal{O}$ sequentially so that $\mathcal{O}_k = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k\}$ is an orthonormal basis for $\mathcal{S}_k = span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$ for $k = 1, \ldots, n$.

For $k = 1$, simply take $\mathbf{u}_1 = \mathbf{x}_1 / \|\mathbf{x}_1\|$. It's clear that $\mathcal{O}_1 = \{\mathbf{u}_1\}$ is an orthonormal set whose span agrees with that of $\mathcal{S}_1 = \{\mathbf{x}_1\}$. Now reason inductively. Suppose that $\mathcal{O}_k = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k\}$ is an orthonormal basis for $\mathcal{S}_k = span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$, and consider the problem of finding one additional vector $\mathbf{u}_{k+1}$ such that $\mathcal{O}_{k+1} = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k, \mathbf{u}_{k+1}\}$ is an orthonormal basis for $\mathcal{S}_{k+1} = span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k, \mathbf{x}_{k+1}\}$. For this to hold, the Fourier expansion (p. 299) of $\mathbf{x}_{k+1}$ with respect to $\mathcal{O}_{k+1}$ must be

$$\mathbf{x}_{k+1} = \sum_{i=1}^{k+1} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \mathbf{u}_i,$$

which in turn implies that

$$\mathbf{u}_{k+1} = \frac{\mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \mathbf{u}_i}{\langle \mathbf{u}_{k+1} | \mathbf{x}_{k+1} \rangle}. \tag{5.5.1}$$

Since $\|\mathbf{u}_{k+1}\| = 1$, it follows from (5.5.1) that

$$| \langle \mathbf{u}_{k+1} | \mathbf{x}_{k+1} \rangle | = \left\| \mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \mathbf{u}_i \right\|,$$

[44] Jorgen P. Gram (1850–1916) was a Danish actuary who implicitly presented the essence of orthogonalization procedure in 1883. Gram was apparently unaware that Pierre-Simon Laplace (1749–1827) had earlier used the method. Today, Gram is remembered primarily for his development of this process, but in earlier times his name was also associated with the matrix product $\mathbf{A}^*\mathbf{A}$ that historically was referred to as the *Gram matrix* of $\mathbf{A}$.

Erhard Schmidt (1876–1959) was a student of Hermann Schwarz (of CBS inequality fame) and the great German mathematician David Hilbert. Schmidt explicitly employed the orthogonalization process in 1907 in his study of integral equations, which in turn led to the development of what are now called *Hilbert spaces*. Schmidt made significant use of the orthogonalization process to develop the geometry of Hilbert Spaces, and thus it came to bear Schmidt's name.

so $\langle \mathbf{u}_{k+1} | \mathbf{x}_{k+1} \rangle = e^{i\theta} \left\| \mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \mathbf{u}_i \right\|$ for some $0 \leq \theta < 2\pi$, and

$$\mathbf{u}_{k+1} = \frac{\mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \mathbf{u}_i}{e^{i\theta} \left\| \mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \mathbf{u}_i \right\|}.$$

Since the value of $\theta$ in the scalar $e^{i\theta}$ neither affects $span\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{k+1}\}$ nor the facts that $\|\mathbf{u}_{k+1}\| = 1$ and $\langle \mathbf{u}_{k+1} | \mathbf{u}_i \rangle = 0$ for all $i \leq k$, we can arbitrarily define $\mathbf{u}_{k+1}$ to be the vector corresponding to the $\theta = 0$ or, equivalently, $e^{i\theta} = 1$. For the sake of convenience, let

$$\nu_{k+1} = \left\| \mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \mathbf{u}_i \right\|$$

so that we can write

$$\mathbf{u}_1 = \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|} \quad \text{and} \quad \mathbf{u}_{k+1} = \frac{\mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \mathbf{u}_i}{\nu_{k+1}} \text{ for } k > 0. \tag{5.5.2}$$

This sequence of vectors is called the ***Gram–Schmidt sequence.*** A straightforward induction argument proves that $\mathcal{O}_k = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k\}$ is indeed an orthonormal basis for $span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$ for each $k = 1, 2, \ldots$. Details are called for in Exercise 5.5.7.

The orthogonalization procedure defined by (5.5.2) is valid for any inner-product space, but if we concentrate on subspaces of $\Re^m$ or $\mathcal{C}^m$ with the standard inner product and euclidean norm, then we can formulate (5.5.2) in terms of matrices. Suppose that $\mathcal{B} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ is a basis for an $n$-dimensional subspace $\mathcal{S}$ of $\mathcal{C}^{m\times 1}$ so that the Gram–Schmidt sequence (5.5.2) becomes

$$\mathbf{u}_1 = \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|} \quad \text{and} \quad \mathbf{u}_k = \frac{\mathbf{x}_k - \sum_{i=1}^{k-1} (\mathbf{u}_i^* \mathbf{x}_k) \mathbf{u}_i}{\left\| \mathbf{x}_k - \sum_{i=1}^{k-1} (\mathbf{u}_i^* \mathbf{x}_k) \mathbf{u}_i \right\|} \quad \text{for } k = 2, 3, \ldots, n. \tag{5.5.3}$$

To express this in matrix notation, set

$$\mathbf{U}_1 = \mathbf{0}_{m\times 1} \quad \text{and} \quad \mathbf{U}_k = \big(\mathbf{u}_1 \,|\, \mathbf{u}_2 \,|\, \cdots \,|\, \mathbf{u}_{k-1}\big)_{m\times k-1} \quad \text{for } k > 1,$$

and notice that

$$\mathbf{U}_k^* \mathbf{x}_k = \begin{pmatrix} \mathbf{u}_1^* \mathbf{x}_k \\ \mathbf{u}_2^* \mathbf{x}_k \\ \vdots \\ \mathbf{u}_{k-1}^* \mathbf{x}_k \end{pmatrix} \quad \text{and} \quad \mathbf{U}_k \mathbf{U}_k^* \mathbf{x}_k = \sum_{i=1}^{k-1} \mathbf{u}_i (\mathbf{u}_i^* \mathbf{x}_k) = \sum_{i=1}^{k-1} (\mathbf{u}_i^* \mathbf{x}_k) \mathbf{u}_i.$$

Since

$$\mathbf{x}_k - \sum_{i=1}^{k-1} (\mathbf{u}_i^* \mathbf{x}_k) \mathbf{u}_i = \mathbf{x}_k - \mathbf{U}_k \mathbf{U}_k^* \mathbf{x}_k = (\mathbf{I} - \mathbf{U}_k \mathbf{U}_k^*) \mathbf{x}_k,$$

the vectors in (5.5.3) can be concisely written as

$$\mathbf{u}_k = \frac{(\mathbf{I} - \mathbf{U}_k \mathbf{U}_k^*) \mathbf{x}_k}{\|(\mathbf{I} - \mathbf{U}_k \mathbf{U}_k^*) \mathbf{x}_k\|} \quad \text{for } k = 1, 2, \ldots, n.$$

Below is a summary.

## Gram–Schmidt Orthogonalization Procedure

If $\mathcal{B} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ is a basis for a general inner-product space $\mathcal{S}$, then the ***Gram–Schmidt sequence*** defined by

$$\mathbf{u}_1 = \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|} \quad \text{and} \quad \mathbf{u}_k = \frac{\mathbf{x}_k - \sum_{i=1}^{k-1} \langle \mathbf{u}_i | \mathbf{x}_k \rangle \mathbf{u}_i}{\left\| \mathbf{x}_k - \sum_{i=1}^{k-1} \langle \mathbf{u}_i | \mathbf{x}_k \rangle \mathbf{u}_i \right\|} \text{ for } k = 2, \ldots, n$$

is an orthonormal basis for $\mathcal{S}$. When $\mathcal{S}$ is an $n$-dimensional subspace of $\mathcal{C}^{m\times 1}$, the Gram–Schmidt sequence can be expressed as

$$\mathbf{u}_k = \frac{(\mathbf{I} - \mathbf{U}_k \mathbf{U}_k^*)\,\mathbf{x}_k}{\|(\mathbf{I} - \mathbf{U}_k \mathbf{U}_k^*)\,\mathbf{x}_k\|} \quad \text{for} \quad k = 1, 2, \ldots, n \tag{5.5.4}$$

in which $\mathbf{U}_1 = \mathbf{0}_{m\times 1}$ and $\mathbf{U}_k = \left(\mathbf{u}_1 \,|\, \mathbf{u}_2 \,|\, \cdots \,|\, \mathbf{u}_{k-1}\right)_{m\times k-1}$ for $k > 1$.

## Example 5.5.1

**Classical Gram–Schmidt Algorithm.** The following formal algorithm is the straightforward or "classical" implementation of the Gram–Schmidt procedure. Interpret $\mathbf{a} \leftarrow \mathbf{b}$ to mean that "$\mathbf{a}$ is defined to be (or overwritten by) $\mathbf{b}$."

For $k = 1$:

$$\mathbf{u}_1 \leftarrow \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|}$$

For $k > 1$:

$$\mathbf{u}_k \leftarrow \mathbf{x}_k - \sum_{i=1}^{k-1} (\mathbf{u}_i^* \mathbf{x}_k)\mathbf{u}_i$$

$$\mathbf{u}_k \leftarrow \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}$$

(See Exercise 5.5.10 for other formulations of the Gram–Schmidt algorithm.)

**Problem:** Use the classical formulation of the Gram–Schmidt procedure given above to find an orthonormal basis for the space spanned by the following three linearly independent vectors.

$$\mathbf{x}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{x}_2 = \begin{pmatrix} 1 \\ 2 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{x}_3 = \begin{pmatrix} 3 \\ 1 \\ 1 \\ -1 \end{pmatrix}.$$

**Solution:**

$$k=1: \quad \mathbf{u}_1 \leftarrow \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

$$k=2: \quad \mathbf{u}_2 \leftarrow \mathbf{x}_2 - (\mathbf{u}_1^T\mathbf{x}_2)\mathbf{u}_1 = \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{u}_2 \leftarrow \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$k=3: \quad \mathbf{u}_3 \leftarrow \mathbf{x}_3 - (\mathbf{u}_1^T\mathbf{x}_3)\mathbf{u}_1 - (\mathbf{u}_2^T\mathbf{x}_3)\mathbf{u}_2 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \quad \mathbf{u}_3 \leftarrow \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} = \frac{1}{\sqrt{3}}\begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

Thus

$$\mathbf{u}_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{u}_3 = \frac{1}{\sqrt{3}}\begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

is the desired orthonormal basis.

---

The Gram–Schmidt process frequently appears in the disguised form of a matrix factorization. To see this, let $\mathbf{A}_{m\times n} = \begin{pmatrix}\mathbf{a}_1 \,|\, \mathbf{a}_2 \,|\, \cdots \,|\, \mathbf{a}_n\end{pmatrix}$ be a matrix with linearly independent columns. When Gram–Schmidt is applied to the columns of $\mathbf{A}$, the result is an orthonormal basis $\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n\}$ for $R(\mathbf{A})$, where

$$\mathbf{q}_1 = \frac{\mathbf{a}_1}{\nu_1} \quad \text{and} \quad \mathbf{q}_k = \frac{\mathbf{a}_k - \sum_{i=1}^{k-1}\langle \mathbf{q}_i | \mathbf{a}_k\rangle \mathbf{q}_i}{\nu_k} \quad \text{for } k = 2, 3, \ldots, n,$$

where $\nu_1 = \|\mathbf{a}_1\|$ and $\nu_k = \left\|\mathbf{a}_k - \sum_{i=1}^{k-1}\langle \mathbf{q}_i | \mathbf{a}_k\rangle \mathbf{q}_i\right\|$ for $k > 1$. The above relationships can be rewritten as

$$\mathbf{a}_1 = \nu_1\mathbf{q}_1 \quad \text{and} \quad \mathbf{a}_k = \langle \mathbf{q}_1|\mathbf{a}_k\rangle \mathbf{q}_1 + \cdots + \langle \mathbf{q}_{k-1}|\mathbf{a}_k\rangle \mathbf{q}_{k-1} + \nu_k\mathbf{q}_k \quad \text{for } k > 1,$$

which in turn can be expressed in matrix form by writing

$$\begin{pmatrix}\mathbf{a}_1 \,|\, \mathbf{a}_2 \,|\, \cdots \,|\, \mathbf{a}_n\end{pmatrix} = \begin{pmatrix}\mathbf{q}_1 \,|\, \mathbf{q}_2 \,|\, \cdots \,|\, \mathbf{q}_n\end{pmatrix} \begin{pmatrix} \nu_1 & \langle \mathbf{q}_1|\mathbf{a}_2\rangle & \langle \mathbf{q}_1|\mathbf{a}_3\rangle & \cdots & \langle \mathbf{q}_1|\mathbf{a}_n\rangle \\ 0 & \nu_2 & \langle \mathbf{q}_2|\mathbf{a}_3\rangle & \cdots & \langle \mathbf{q}_2|\mathbf{a}_n\rangle \\ 0 & 0 & \nu_3 & \cdots & \langle \mathbf{q}_3|\mathbf{a}_n\rangle \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \nu_n \end{pmatrix}.$$

This says that it's possible to factor a matrix with independent columns as $\mathbf{A}_{m\times n} = \mathbf{Q}_{m\times n}\mathbf{R}_{n\times n}$, where the columns of $\mathbf{Q}$ are an orthonormal basis for $R(\mathbf{A})$ and $\mathbf{R}$ is an upper-triangular matrix with positive diagonal elements.

The factorization $\mathbf{A} = \mathbf{QR}$ is called the ***QR factorization*** for $\mathbf{A}$, and it is uniquely determined by $\mathbf{A}$ (Exercise 5.5.8). When $\mathbf{A}$ and $\mathbf{Q}$ are not square, some authors emphasize the point by calling $\mathbf{A} = \mathbf{QR}$ the *rectangular* QR factorization—the case when $\mathbf{A}$ and $\mathbf{Q}$ are square is further discussed on p. 345. Below is a summary of the above observations.

## QR Factorization

Every matrix $\mathbf{A}_{m\times n}$ with linearly independent columns can be uniquely factored as $\mathbf{A} = \mathbf{QR}$ in which the columns of $\mathbf{Q}_{m\times n}$ are an orthonormal basis for $R(\mathbf{A})$ and $\mathbf{R}_{n\times n}$ is an upper-triangular matrix with positive diagonal entries.

- The QR factorization is the complete "road map" of the Gram–Schmidt process because the columns of $\mathbf{Q} = \begin{pmatrix}\mathbf{q}_1 \,|\, \mathbf{q}_2 \,|\, \cdots \,|\, \mathbf{q}_n\end{pmatrix}$ are the result of applying the Gram–Schmidt procedure to the columns of $\mathbf{A} = \begin{pmatrix}\mathbf{a}_1 \,|\, \mathbf{a}_2 \,|\, \cdots \,|\, \mathbf{a}_n\end{pmatrix}$ and $\mathbf{R}$ is given by

$$\mathbf{R} = \begin{pmatrix} \nu_1 & \mathbf{q}_1^*\mathbf{a}_2 & \mathbf{q}_1^*\mathbf{a}_3 & \cdots & \mathbf{q}_1^*\mathbf{a}_n \\ 0 & \nu_2 & \mathbf{q}_2^*\mathbf{a}_3 & \cdots & \mathbf{q}_2^*\mathbf{a}_n \\ 0 & 0 & \nu_3 & \cdots & \mathbf{q}_3^*\mathbf{a}_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \nu_n \end{pmatrix},$$

where $\nu_1 = \|\mathbf{a}_1\|$ and $\nu_k = \left\|\mathbf{a}_k - \sum_{i=1}^{k-1} \langle \mathbf{q}_i | \mathbf{a}_k \rangle \, \mathbf{q}_i\right\|$ for $k > 1$.

## Example 5.5.2

**Problem:** Determine the QR factors of

$$\mathbf{A} = \begin{pmatrix} 0 & -20 & -14 \\ 3 & 27 & -4 \\ 4 & 11 & -2 \end{pmatrix}.$$

**Solution:** Using the standard inner product for $\Re^n$, apply the Gram–Schmidt procedure to the columns of $\mathbf{A}$ by setting

$$\mathbf{q}_1 = \frac{\mathbf{a}_1}{\nu_1} \quad \text{and} \quad \mathbf{q}_k = \frac{\mathbf{a}_k - \sum_{i=1}^{k-1} \left(\mathbf{q}_i^T \mathbf{a}_k\right) \mathbf{q}_i}{\nu_k} \quad \text{for } k = 2, 3,$$

where $\nu_1 = \|\mathbf{a}_1\|$ and $\nu_k = \left\|\mathbf{a}_k - \sum_{i=1}^{k-1} \left(\mathbf{q}_i^T \mathbf{a}_k\right) \mathbf{q}_i\right\|$. The computation of these quantities can be organized as follows.

$$k=1\text{:}\quad r_{11} \leftarrow \|\mathbf{a}_1\| = 5 \qquad \text{and} \qquad \mathbf{q}_1 \leftarrow \frac{\mathbf{a}_1}{r_{11}} = \begin{pmatrix} 0 \\ 3/5 \\ 4/5 \end{pmatrix}$$

$$k=2\text{:}\quad r_{12} \leftarrow \mathbf{q}_1^T\mathbf{a}_2 = 25$$

$$\mathbf{q}_2 \leftarrow \mathbf{a}_2 - r_{12}\mathbf{q}_1 = \begin{pmatrix} -20 \\ 12 \\ -9 \end{pmatrix}$$

$$r_{22} \leftarrow \|\mathbf{q}_2\| = 25 \text{ and } \mathbf{q}_2 \leftarrow \frac{\mathbf{q}_2}{r_{22}} = \frac{1}{25}\begin{pmatrix} -20 \\ 12 \\ -9 \end{pmatrix}$$

$$k=3\text{:}\quad r_{13} \leftarrow \mathbf{q}_1^T\mathbf{a}_3 = -4 \text{ and } r_{23} \leftarrow \mathbf{q}_2^T\mathbf{a}_3 = 10$$

$$\mathbf{q}_3 \leftarrow \mathbf{a}_3 - r_{13}\mathbf{q}_1 - r_{23}\mathbf{q}_2 = \frac{2}{5}\begin{pmatrix} -15 \\ -16 \\ 12 \end{pmatrix}$$

$$r_{33} \leftarrow \|\mathbf{q}_3\| = 10 \text{ and } \mathbf{q}_3 \leftarrow \frac{\mathbf{q}_3}{r_{33}} = \frac{1}{25}\begin{pmatrix} -15 \\ -16 \\ 12 \end{pmatrix}$$

Therefore,

$$\mathbf{Q} = \frac{1}{25}\begin{pmatrix} 0 & -20 & -15 \\ 15 & 12 & -16 \\ 20 & -9 & 12 \end{pmatrix} \quad \text{and} \quad \mathbf{R} = \begin{pmatrix} 5 & 25 & -4 \\ 0 & 25 & 10 \\ 0 & 0 & 10 \end{pmatrix}.$$

---

We now have two important matrix factorizations, namely, the LU factorization, discussed in §3.10 on p. 141 and the QR factorization. They are not the same, but some striking analogies exist.

- Each factorization represents a reduction to upper-triangular form—LU by Gaussian elimination, and QR by Gram–Schmidt. In particular, the LU factorization is the complete "road map" of Gaussian elimination applied to a square nonsingular matrix, whereas QR is the complete road map of Gram–Schmidt applied to a matrix with linearly independent columns.
- When they exist, both factorizations $\mathbf{A} = \mathbf{LU}$ and $\mathbf{A} = \mathbf{QR}$ are uniquely determined by $\mathbf{A}$.
- Once the LU factors (assuming they exist) of a nonsingular matrix $\mathbf{A}$ are known, the solution of $\mathbf{Ax} = \mathbf{b}$ is easily computed—solve $\mathbf{Ly} = \mathbf{b}$ by forward substitution, and then solve $\mathbf{Ux} = \mathbf{y}$ by back substitution (see p. 146). The QR factors can be used in a similar manner. If $\mathbf{A} \in \Re^{n\times n}$ is nonsingular, then $\mathbf{Q}^T = \mathbf{Q}^{-1}$ (because $\mathbf{Q}$ has orthonormal columns), so $\mathbf{Ax} = \mathbf{b} \iff \mathbf{QRx} = \mathbf{b} \iff \mathbf{Rx} = \mathbf{Q}^T\mathbf{b}$, which is also a triangular system that is solved by back substitution.

While the LU and QR factors can be used in more or less the same way to solve nonsingular systems, things are different for singular and rectangular cases because $\mathbf{Ax} = \mathbf{b}$ might be inconsistent, in which case a least squares solution as described in §4.6, (p. 223) may be desired. Unfortunately, the LU factors of $\mathbf{A}$ don't exist when $\mathbf{A}$ is rectangular. And even if $\mathbf{A}$ is square and has an LU factorization, the LU factors of $\mathbf{A}$ are not much help in solving the system of normal equations $\mathbf{A}^T\mathbf{Ax} = \mathbf{A}^T\mathbf{b}$ that produces least squares solutions. But the QR factors of $\mathbf{A}_{m\times n}$ always exist as long as $\mathbf{A}$ has linearly independent columns, and, as demonstrated in the following example, the QR factors provide the least squares solution of an inconsistent system in exactly the same way as they provide the solution of a consistent system.

## Example 5.5.3

**Application to the Least Squares Problem.** If $\mathbf{Ax} = \mathbf{b}$ is a possibly inconsistent (real) system, then, as discussed on p. 226, the set of all least squares solutions is the set of solutions to the system of normal equations

$$\mathbf{A}^T\mathbf{Ax} = \mathbf{A}^T\mathbf{b}. \tag{5.5.5}$$

But computing $\mathbf{A}^T\mathbf{A}$ and then performing an LU factorization of $\mathbf{A}^T\mathbf{A}$ to solve (5.5.5) is generally not advisable. First, it's inefficient and, second, as pointed out in Example 4.5.1, computing $\mathbf{A}^T\mathbf{A}$ with floating-point arithmetic can result in a loss of significant information. The QR approach doesn't suffer from either of these objections. Suppose that $rank\,(\mathbf{A}_{m\times n}) = n$ (so that there is a unique least squares solution), and let $\mathbf{A} = \mathbf{QR}$ be the QR factorization. Because the columns of $\mathbf{Q}$ are an orthonormal set, it follows that $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}_n$, so

$$\mathbf{A}^T\mathbf{A} = (\mathbf{QR})^T(\mathbf{QR}) = \mathbf{R}^T\mathbf{Q}^T\mathbf{QR} = \mathbf{R}^T\mathbf{R}. \tag{5.5.6}$$

Consequently, the normal equations (5.5.5) can be written as

$$\mathbf{R}^T\mathbf{Rx} = \mathbf{R}^T\mathbf{Q}^T\mathbf{b}. \tag{5.5.7}$$

But $\mathbf{R}^T$ is nonsingular (it is triangular with positive diagonal entries), so (5.5.7) simplifies to become

$$\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}. \tag{5.5.8}$$

This is just an upper-triangular system that is efficiently solved by back substitution. In other words, most of the work involved in solving the least squares problem is in computing the QR factorization of $\mathbf{A}$. Finally, notice that

$$\mathbf{x} = \mathbf{R}^{-1}\mathbf{Q}^T\mathbf{b} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{b}$$

is the solution of $\mathbf{Ax} = \mathbf{b}$ when the system is consistent as well as the least squares solution when the system is inconsistent (see p. 214). That is, with the QR approach, it makes no difference whether or not $\mathbf{Ax} = \mathbf{b}$ is consistent because in both cases things boil down to solving the same equation—namely, (5.5.8). Below is a formal summary.

## Linear Systems and the QR Factorization

If $rank\,(\mathbf{A}_{m\times n}) = n$, and if $\mathbf{A} = \mathbf{QR}$ is the QR factorization, then the solution of the nonsingular triangular system

$$\mathbf{Rx} = \mathbf{Q}^T\mathbf{b} \tag{5.5.9}$$

is either the solution or the least squares solution of $\mathbf{Ax} = \mathbf{b}$ depending on whether or not $\mathbf{Ax} = \mathbf{b}$ is consistent.

It's worthwhile to reemphasize that the QR approach to the least squares problem obviates the need to explicitly compute the product $\mathbf{A}^T\mathbf{A}$. But if $\mathbf{A}^T\mathbf{A}$ is ever needed, it is retrievable from the factorization $\mathbf{A}^T\mathbf{A} = \mathbf{R}^T\mathbf{R}$. In fact, this is the ***Cholesky factorization*** of $\mathbf{A}^T\mathbf{A}$ as discussed in Example 3.10.7, p. 154.

---

The Gram–Schmidt procedure is a powerful theoretical tool, but it's not a good numerical algorithm when implemented in the straightforward or "classical" sense. When floating-point arithmetic is used, the classical Gram–Schmidt algorithm applied to a set of vectors that is not already close to being an orthogonal set can produce a set of vectors that is far from being an orthogonal set. To see this, consider the following example.

### Example 5.5.4

---

**Problem:** Using 3-digit floating-point arithmetic, apply the classical Gram–Schmidt algorithm to the set

$$\mathbf{x}_1 = \begin{pmatrix} 1 \\ 10^{-3} \\ 10^{-3} \end{pmatrix}, \quad \mathbf{x}_2 = \begin{pmatrix} 1 \\ 10^{-3} \\ 0 \end{pmatrix}, \quad \mathbf{x}_3 = \begin{pmatrix} 1 \\ 0 \\ 10^{-3} \end{pmatrix}.$$

**Solution:**

$k = 1$: $fl\,\|\mathbf{x}_1\| = 1$, so $\mathbf{u}_1 \leftarrow \mathbf{x}_1$.

$k = 2$: $fl\,(\mathbf{u}_1^T\mathbf{x}_2) = 1$, so

$$\mathbf{u}_2 \leftarrow \mathbf{x}_2 - (\mathbf{u}_1^T\mathbf{x}_2)\,\mathbf{u}_1 = \begin{pmatrix} 0 \\ 0 \\ -10^{-3} \end{pmatrix} \quad \text{and} \quad \mathbf{u}_2 \leftarrow fl\left(\frac{\mathbf{u}_2}{\|\mathbf{u}_2\|}\right) = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}.$$

$k = 3$: $fl\,(\mathbf{u}_1^T\mathbf{x}_3) = 1$ and $fl\,(\mathbf{u}_2^T\mathbf{x}_3) = -10^{-3}$, so

$$\mathbf{u}_3 \leftarrow \mathbf{x}_3 - (\mathbf{u}_1^T\mathbf{x}_3)\mathbf{u}_1 - (\mathbf{u}_2^T\mathbf{x}_3)\mathbf{u}_2 = \begin{pmatrix} 0 \\ -10^{-3} \\ -10^{-3} \end{pmatrix} \text{ and } \mathbf{u}_3 \leftarrow fl\left(\frac{\mathbf{u}_3}{\|\mathbf{u}_3\|}\right) = \begin{pmatrix} 0 \\ -.709 \\ -.709 \end{pmatrix}.$$

Therefore, classical Gram–Schmidt with 3-digit arithmetic returns

$$\mathbf{u}_1 = \begin{pmatrix} 1 \\ 10^{-3} \\ 10^{-3} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{u}_3 = \begin{pmatrix} 0 \\ -.709 \\ -.709 \end{pmatrix}, \tag{5.5.10}$$

which is unsatisfactory because $\mathbf{u_2}$ and $\mathbf{u}_3$ are far from being orthogonal.

---

It's possible to improve the numerical stability of the orthogonalization process by rearranging the order of the calculations. Recall from (5.5.4) that

$$\mathbf{u}_k = \frac{(\mathbf{I} - \mathbf{U}_k\mathbf{U}_k^*)\,\mathbf{x}_k}{\|(\mathbf{I} - \mathbf{U}_k\mathbf{U}_k^*)\,\mathbf{x}_k\|}, \quad \text{where} \quad \mathbf{U}_1 = \mathbf{0} \text{ and } \mathbf{U}_k = \big(\mathbf{u}_1 \,|\, \mathbf{u}_2 \,|\, \cdots \,|\, \mathbf{u}_{k-1}\big).$$

If $\mathbf{E}_1 = \mathbf{I}$ and $\mathbf{E}_i = \mathbf{I} - \mathbf{u}_{i-1}\mathbf{u}_{i-1}^*$ for $i > 1$, then the orthogonality of the $\mathbf{u}_i$'s insures that

$$\mathbf{E}_k \cdots \mathbf{E}_2\mathbf{E}_1 = \mathbf{I} - \mathbf{u}_1\mathbf{u}_1^* - \mathbf{u}_2\mathbf{u}_2^* - \cdots - \mathbf{u}_{k-1}\mathbf{u}_{k-1}^* = \mathbf{I} - \mathbf{U}_k\mathbf{U}_k^*,$$

so the Gram–Schmidt sequence can also be expressed as

$$\mathbf{u}_k = \frac{\mathbf{E}_k \cdots \mathbf{E}_2\mathbf{E}_1\mathbf{x}_k}{\|\mathbf{E}_k \cdots \mathbf{E}_2\mathbf{E}_1\mathbf{x}_k\|} \quad \text{for } k = 1, 2, \ldots, n.$$

This means that the Gram–Schmidt sequence can be generated as follows:

$$\begin{aligned}
\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} &\xrightarrow{\text{Normalize 1-st}} \{\mathbf{u}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \\
&\xrightarrow{\text{Apply } \mathbf{E}_2} \{\mathbf{u}_1,\ \mathbf{E}_2\mathbf{x}_2,\ \mathbf{E}_2\mathbf{x}_3,\ \ldots,\ \mathbf{E}_2\mathbf{x}_n\} \\
&\xrightarrow{\text{Normalize 2-nd}} \{\mathbf{u}_1, \mathbf{u}_2,\ \mathbf{E}_2\mathbf{x}_3,\ \ldots,\ \mathbf{E}_2\mathbf{x}_n\} \\
&\xrightarrow{\text{Apply } \mathbf{E}_3} \{\mathbf{u}_1, \mathbf{u}_2,\ \mathbf{E}_3\mathbf{E}_2\mathbf{x}_3,\ \ldots,\ \mathbf{E}_3\mathbf{E}_2\mathbf{x}_n\} \\
&\xrightarrow[\text{etc.}]{\text{Normalize 3-rd}} \{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3,\ \mathbf{E}_3\mathbf{E}_2\mathbf{x}_4, \ldots,\ \mathbf{E}_3\mathbf{E}_2\mathbf{x}_n\},
\end{aligned}$$

While there is no theoretical difference, this "modified" algorithm is numerically more stable than the classical algorithm when floating-point arithmetic is used. The $k^{th}$ step of the classical algorithm alters only the $k^{th}$ vector, but the $k^{th}$ step of the modified algorithm "updates" all vectors from the $k^{th}$ through the last, and conditioning the unorthogonalized tail in this way makes a difference.

## Modified Gram–Schmidt Algorithm

For a linearly independent set $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \subset \mathcal{C}^{m\times 1}$, the Gram–Schmidt sequence given on p. 309 can be alternately described as

$$\mathbf{u}_k = \frac{\mathbf{E}_k \cdots \mathbf{E}_2\mathbf{E}_1\mathbf{x}_k}{\|\mathbf{E}_k \cdots \mathbf{E}_2\mathbf{E}_1\mathbf{x}_k\|} \text{ with } \mathbf{E}_1 = \mathbf{I}, \;\; \mathbf{E}_i = \mathbf{I} - \mathbf{u}_{i-1}\mathbf{u}_{i-1}^* \text{ for } i > 1,$$

and this sequence is generated by the following algorithm.

For $k = 1$: $\mathbf{u}_1 \leftarrow \mathbf{x}_1/\|\mathbf{x}_1\|$ and $\mathbf{u}_j \leftarrow \mathbf{x}_j$ for $j = 2, 3, \ldots, n$

For $k > 1$: $\mathbf{u}_j \leftarrow \mathbf{E}_k\mathbf{u}_j = \mathbf{u}_j - \left(\mathbf{u}_{k-1}^*\mathbf{u}_j\right)\mathbf{u}_{k-1}$ for $j = k, k+1, \ldots, n$
$\mathbf{u}_k \leftarrow \mathbf{u}_k/\|\mathbf{u}_k\|$

(An alternate implementation is given in Exercise 5.5.10.)

To see that the modified version of Gram–Schmidt can indeed make a difference when floating-point arithmetic is used, consider the following example.

### Example 5.5.5

**Problem:** Use 3-digit floating-point arithmetic, and apply the modified Gram–Schmidt algorithm to the set given in Example 5.5.4 (p. 314), and then compare the results of the modified algorithm with those of the classical algorithm.

**Solution:** $\mathbf{x}_1 = \begin{pmatrix} 1 \\ 10^{-3} \\ 10^{-3} \end{pmatrix}$, $\mathbf{x}_2 = \begin{pmatrix} 1 \\ 10^{-3} \\ 0 \end{pmatrix}$, $\mathbf{x}_3 = \begin{pmatrix} 1 \\ 0 \\ 10^{-3} \end{pmatrix}$.

$k = 1$: $fl\,\|\mathbf{x}_1\| = 1$, so $\{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3\} \leftarrow \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$.

$k = 2$: $fl\left(\mathbf{u}_1^T\mathbf{u}_2\right) = 1$ and $fl\left(\mathbf{u}_1^T\mathbf{u}_3\right) = 1$, so

$$\mathbf{u}_2 \leftarrow \mathbf{u}_2 - \left(\mathbf{u}_1^T\mathbf{u}_2\right)\mathbf{u}_1 = \begin{pmatrix} 0 \\ 0 \\ -10^{-3} \end{pmatrix}, \quad \mathbf{u}_3 \leftarrow \mathbf{u}_3 - \left(\mathbf{u}_1^T\mathbf{u}_3\right)\mathbf{u}_1 = \begin{pmatrix} 0 \\ -10^{-3} \\ 0 \end{pmatrix},$$

and

$$\mathbf{u}_2 \leftarrow \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}.$$

$k = 3$: $\mathbf{u}_2^T\mathbf{u}_3 = 0$, so

$$\mathbf{u}_3 \leftarrow \mathbf{u}_3 - \left(\mathbf{u}_2^T\mathbf{u}_3\right)\mathbf{u}_2 = \begin{pmatrix} 0 \\ -10^{-3} \\ 0 \end{pmatrix} \quad \text{and} \quad \mathbf{u}_3 \leftarrow \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} = \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}.$$

Thus the modified Gram–Schmidt algorithm produces

$$\mathbf{u}_1 = \begin{pmatrix} 1 \\ 10^{-3} \\ 10^{-3} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{u}_3 = \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}, \tag{5.5.11}$$

which is as good as one can expect using 3-digit arithmetic. Comparing (5.5.11) with the result (5.5.10) obtained in Example 5.5.4 illuminates the advantage possessed by modified Gram–Schmidt algorithm over the classical algorithm.

Below is a summary of some facts concerning the modified Gram–Schmidt algorithm compared with the classical implementation.

## Summary

- When the Gram–Schmidt procedures (classical or modified) are applied to the columns of $\mathbf{A}$ using exact arithmetic, each produces an orthonormal basis for $R(\mathbf{A})$.
- For computing a QR factorization in floating-point arithmetic, the modified algorithm produces results that are at least as good as and often better than the classical algorithm, but the modified algorithm is not unconditionally stable—there are situations in which it fails to produce a set of columns that are nearly orthogonal.
- For solving the least square problem with floating-point arithmetic, the modified procedure is a numerically stable algorithm in the sense that the method described in Example 5.5.3 returns a result that is the exact solution of a nearby least squares problem. However, the Householder method described on p. 346 is just as stable and needs slightly fewer arithmetic operations.

## Exercises for section 5.5

**5.5.1.** Let $\mathcal{S} = span\left\{ \mathbf{x}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{x}_2 = \begin{pmatrix} 2 \\ -1 \\ -1 \\ 1 \end{pmatrix}, \quad \mathbf{x}_3 = \begin{pmatrix} -1 \\ 2 \\ 2 \\ 1 \end{pmatrix} \right\}$.

(a) Use the classical Gram–Schmidt algorithm (with exact arithmetic) to determine an orthonormal basis for $\mathcal{S}$.

(b) Verify directly that the Gram–Schmidt sequence produced in part (a) is indeed an orthonormal basis for $\mathcal{S}$.

(c) Repeat part (a) using the modified Gram–Schmidt algorithm, and compare the results.

**5.5.2.** Use the Gram–Schmidt procedure to find an orthonormal basis for the four fundamental subspaces of $\mathbf{A} = \begin{pmatrix} 1 & -2 & 3 & -1 \\ 2 & -4 & 6 & -2 \\ 3 & -6 & 9 & -3 \end{pmatrix}$.

**5.5.3.** Apply the Gram–Schmidt procedure with the standard inner product for $\mathcal{C}^3$ to $\left\{ \begin{pmatrix} \mathrm{i} \\ \mathrm{i} \\ \mathrm{i} \end{pmatrix}, \begin{pmatrix} 0 \\ \mathrm{i} \\ \mathrm{i} \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \mathrm{i} \end{pmatrix} \right\}$.

**5.5.4.** Explain what happens when the Gram–Schmidt process is applied to an orthonormal set of vectors.

**5.5.5.** Explain what happens when the Gram–Schmidt process is applied to a linearly dependent set of vectors.

**5.5.6.** Let $\mathbf{A} = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 2 & 1 \\ 1 & 1 & -3 \\ 0 & 1 & 1 \end{pmatrix}$ and $\mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$.

(a) Determine the rectangular QR factorization of $\mathbf{A}$.

(b) Use the QR factors from part (a) to determine the least squares solution to $\mathbf{Ax} = \mathbf{b}$.

**5.5.7.** Given a linearly independent set of vectors $\mathcal{S} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in an inner-product space, let $\mathcal{S}_k = span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$ for $k = 1, 2, \ldots, n$. Give an induction argument to prove that if $\mathcal{O}_k = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k\}$ is the Gram–Schmidt sequence defined in (5.5.2), then $\mathcal{O}_k$ is indeed an orthonormal basis for $\mathcal{S}_k = span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$ for each $k = 1, 2, \ldots, n$.

**5.5.8.** Prove that if $rank(\mathbf{A}_{m\times n}) = n$, then the rectangular QR factorization of $\mathbf{A}$ is unique. That is, if $\mathbf{A} = \mathbf{QR}$, where $\mathbf{Q}_{m\times n}$ has orthonormal columns and $\mathbf{R}_{n\times n}$ is upper triangular with positive diagonal entries, then $\mathbf{Q}$ and $\mathbf{R}$ are unique. **Hint:** Recall Example 3.10.7, p. 154.

**5.5.9.** (a) Apply classical Gram–Schmidt with 3-digit floating-point arithmetic to $\left\{ \mathbf{x}_1 = \begin{pmatrix} 1 \\ 0 \\ 10^{-3} \end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix} 1 \\ 10^{-3} \\ 0 \end{pmatrix} \right\}$. You may assume that $fl\left(\sqrt{2}\right) = 1.41$.

(b) Again using 3-digit floating-point arithmetic, apply the modified Gram–Schmidt algorithm to $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$, and compare the result with that of part (a).

**5.5.10.** Depending on how the inner products $r_{ij}$ are defined, verify that the following code implements both the classical and modified Gram–Schmidt algorithms applied to a set of vectors $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$.

*For* $j = 1$ *to* $n$
$\mathbf{u}_j \longleftarrow \mathbf{x}_j$
*For* $i = 1$ *to* $j - 1$
$r_{ij} \longleftarrow \begin{cases} \langle \mathbf{u}_i | \mathbf{x}_j \rangle & \text{(classical Gram–Schmidt)} \\ \langle \mathbf{u}_i | \mathbf{u}_j \rangle & \text{(modified Gram–Schmidt)} \end{cases}$
$\mathbf{u}_j \longleftarrow \mathbf{u}_j - r_{ij}\mathbf{u}_i$
*End*
$r_{jj} \longleftarrow \|\mathbf{u}_j\|$
*If* $r_{jj} = 0$
*quit* (because $\mathbf{x}_j \in span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{j-1}\}$)
*Else* $\mathbf{u}_j \longleftarrow \mathbf{u}_j / r_{jj}$
*End*

If exact arithmetic is used, will the inner products $r_{ij}$ be the same for both implementations?

**5.5.11.** Let $\mathcal{V}$ be the inner-product space of real-valued continuous functions defined on the interval $[-1, 1]$, where the inner product is defined by

$$\langle f | g \rangle = \int_{-1}^{1} f(x)g(x)dx,$$

and let $\mathcal{S}$ be the subspace of $\mathcal{V}$ that is spanned by the three linearly independent polynomials $q_0 = 1, \quad q_1 = x, \quad q_2 = x^2$.

(a) Use the Gram–Schmidt process to determine an orthonormal set of polynomials $\{p_0, p_1, p_2\}$ that spans $\mathcal{S}$. These polynomials are the first three normalized ***Legendre***[45] ***polynomials.***

(b) Verify that $p_n$ satisfies ***Legendre's differential equation***

$$(1 - x^2)y'' - 2xy' + n(n + 1)y = 0$$

for $n = 0, 1, 2$. This equation and its solutions are of considerable importance in applied mathematics.

[45] Adrien–Marie Legendre (1752–1833) was one of the most eminent French mathematicians of the eighteenth century. His primary work in higher mathematics concerned number theory and the study of elliptic functions. But he was also instrumental in the development of the theory of least squares, and some people believe that Legendre should receive the credit that is often afforded to Gauss for the introduction of the method of least squares. Like Gauss and many other successful mathematicians, Legendre spent substantial time engaged in diligent and painstaking computation. It is reported that in 1824 Legendre refused to vote for the government's candidate for Institut National, so his pension was stopped, and he died in poverty.

**5.4.17.** Choose any unit vector $\mathbf{e}_i$ for $\mathbf{y}$. The angle between $\mathbf{e}$ and $\mathbf{e}_i$ approaches $\pi/2$ as $n \to \infty$, but $\mathbf{e}^T\mathbf{e}_i = 1$ for all $n$.

**5.4.18.** If $\mathbf{y}$ is negatively correlated to $\mathbf{x}$, then $\mathbf{z_x} = -\mathbf{z_y}$, but $\|\mathbf{z_x} - \mathbf{z_y}\|_2 = 2\sqrt{n}$ gives no indication of the fact that $\mathbf{z_x}$ and $\mathbf{z_y}$ are on the same line. Continuity therefore dictates that when $\mathbf{y} \approx \beta_0\mathbf{e} + \beta_1\mathbf{x}$ with $\beta_1 < 0$, then $\mathbf{z_x} \approx -\mathbf{z_y}$, but $\|\mathbf{z_x} - \mathbf{z_y}\|_2 \approx 2\sqrt{n}$ gives no hint that $\mathbf{z_x}$ and $\mathbf{z_y}$ are almost on the same line. If we want to use norms to gauge linear correlation, we should use

$$\min\left\{\|\mathbf{z_x} - \mathbf{z_y}\|_2\,,\ \|\mathbf{z_x} + \mathbf{z_y}\|_2\right\}.$$

**5.4.19.** (a) $\cos\theta = 1 \implies \langle \mathbf{x}|\mathbf{y}\rangle = \|\mathbf{x}\|\ \|\mathbf{y}\| > 0$, and the straightforward extension of Exercise 5.1.9 guarantees that

$$\mathbf{y} = \frac{\langle \mathbf{x}|\mathbf{y}\rangle}{\|\mathbf{x}\|^2}\mathbf{x}, \quad \text{and clearly} \quad \frac{\langle \mathbf{x}|\mathbf{y}\rangle}{\|\mathbf{x}\|^2} > 0.$$

Conversely, if $\mathbf{y} = \alpha\mathbf{x}$ for $\alpha > 0$, then $\langle \mathbf{x}|\mathbf{y}\rangle = \alpha\|\mathbf{x}\|^2 \implies \cos\theta = 1$.

(b) $\cos\theta = -1 \implies \langle \mathbf{x}|\mathbf{y}\rangle = -\|\mathbf{x}\|\ \|\mathbf{y}\| < 0$, so the generalized version of Exercise 5.1.9 guarantees that

$$\mathbf{y} = \frac{\langle \mathbf{x}|\mathbf{y}\rangle}{\|\mathbf{x}\|^2}\mathbf{x}, \quad \text{and in this case} \quad \frac{\langle \mathbf{x}|\mathbf{y}\rangle}{\|\mathbf{x}\|^2} < 0.$$

Conversely, if $\mathbf{y} = \alpha\mathbf{x}$ for $\alpha < 0$, then $\langle \mathbf{x}|\mathbf{y}\rangle = \alpha\|\mathbf{x}\|^2$, so

$$\cos\theta = \frac{\alpha\|\mathbf{x}\|^2}{|\alpha|\ \|\mathbf{x}\|^2} = -1.$$

**5.4.20.** $F(t) = \sum_n^\infty (-1)^n \frac{2}{n}\sin nt$.

## Solutions for exercises in section 5. 5

**5.5.1.** (a)

$$\mathbf{u}_1 = \frac{1}{2}\begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{u}_2 = \frac{1}{2\sqrt{3}}\begin{pmatrix} 3 \\ -1 \\ -1 \\ 1 \end{pmatrix}, \quad \mathbf{u}_3 = \frac{1}{\sqrt{6}}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 2 \end{pmatrix}$$

(b) First verify this is an orthonormal set by showing $\mathbf{u}_i^T\mathbf{u}_j = \begin{cases} 1 & \text{when } i = j, \\ 0 & \text{when } i \neq j. \end{cases}$
To show that the $\mathbf{x}_i$'s and the $\mathbf{u}_i$'s span the same space, place the $\mathbf{x}_i$'s as rows in a matrix $\mathbf{A}$, and place the $\mathbf{u}_i$'s as rows in a matrix $\mathbf{B}$, and then verify that $\mathbf{E_A} = \mathbf{E_B}$—recall Example 4.2.2.

(c) The result should be the same as in part (a).

**5.5.2.** First reduce $\mathbf{A}$ to $\mathbf{E_A}$ to determine a "regular" basis for each space.

$$R(\mathbf{A}) = span\left\{\begin{pmatrix}1\\2\\3\end{pmatrix}\right\} \qquad N\left(\mathbf{A}^T\right) = span\left\{\begin{pmatrix}-2\\1\\0\end{pmatrix}, \begin{pmatrix}-3\\0\\1\end{pmatrix}\right\}$$

$$R\left(\mathbf{A}^T\right) = span\left\{\begin{pmatrix}1\\-2\\3\\-1\end{pmatrix}\right\} \qquad N(\mathbf{A}) = span\left\{\begin{pmatrix}2\\1\\0\\0\end{pmatrix}, \begin{pmatrix}-3\\0\\1\\0\end{pmatrix}, \begin{pmatrix}1\\0\\0\\1\end{pmatrix}\right\}$$

Now apply Gram–Schmidt to each of these.

$$R(\mathbf{A}) = span\left\{\frac{1}{\sqrt{14}}\begin{pmatrix}1\\2\\3\end{pmatrix}\right\} \quad N\left(\mathbf{A}^T\right) = span\left\{\frac{1}{\sqrt{5}}\begin{pmatrix}-2\\1\\0\end{pmatrix}, \frac{1}{\sqrt{70}}\begin{pmatrix}-3\\-6\\5\end{pmatrix}\right\}$$

$$R\left(\mathbf{A}^T\right) = span\left\{\frac{1}{\sqrt{15}}\begin{pmatrix}1\\-2\\3\\-1\end{pmatrix}\right\}$$

$$N(\mathbf{A}) = span\left\{\frac{1}{\sqrt{5}}\begin{pmatrix}2\\1\\0\\0\end{pmatrix}, \frac{1}{\sqrt{70}}\begin{pmatrix}-3\\6\\5\\0\end{pmatrix}, \frac{1}{\sqrt{210}}\begin{pmatrix}1\\-2\\3\\14\end{pmatrix}\right\}$$

**5.5.3.**

$$\mathbf{u}_1 = \frac{1}{\sqrt{3}}\begin{pmatrix}\mathrm{i}\\\mathrm{i}\\\mathrm{i}\end{pmatrix}, \quad \mathbf{u}_2 = \frac{1}{\sqrt{6}}\begin{pmatrix}-2\mathrm{i}\\\mathrm{i}\\\mathrm{i}\end{pmatrix}, \quad \mathbf{u}_3 = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\-\mathrm{i}\\\mathrm{i}\end{pmatrix}$$

**5.5.4.** Nothing! The resulting orthonormal set is the same as the original.

**5.5.5.** It breaks down at the first vector such that $\mathbf{x}_k \in span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k-1}\}$ because if

$$\mathbf{x}_k \in span\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k-1}\} = span\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{k-1}\},$$

then the Fourier expansion of $\mathbf{x}_k$ with respect to $span\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{k-1}\}$ is

$$\mathbf{x}_k = \sum_{i=1}^{k-1} \langle \mathbf{u}_i | \mathbf{x}_k \rangle \, \mathbf{u}_i,$$

and therefore

$$\mathbf{u}_k = \frac{\left(\mathbf{x}_k - \sum_{i=1}^{k-1} \langle \mathbf{u}_i | \mathbf{x}_k \rangle \, \mathbf{u}_i\right)}{\left\|\left(\mathbf{x}_k - \sum_{i=1}^{k-1} \langle \mathbf{u}_i | \mathbf{x}_k \rangle \, \mathbf{u}_i\right)\right\|} = \frac{\mathbf{0}}{\|\mathbf{0}\|}$$

is not defined.

**5.5.6.** (a) The rectangular QR factors are

$$\mathbf{Q} = \begin{pmatrix} 1/\sqrt{3} & -1/\sqrt{3} & 1/\sqrt{6} \\ 1/\sqrt{3} & 1/\sqrt{3} & 1/\sqrt{6} \\ 1/\sqrt{3} & 0 & -2/\sqrt{6} \\ 0 & 1/\sqrt{3} & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{R} = \begin{pmatrix} \sqrt{3} & \sqrt{3} & -\sqrt{3} \\ 0 & \sqrt{3} & \sqrt{3} \\ 0 & 0 & \sqrt{6} \end{pmatrix}.$$

(b) Following Example 5.5.3, solve $\mathbf{R}\mathbf{x} = \mathbf{Q}^T\mathbf{b}$ to get $\mathbf{x} = \begin{pmatrix} 2/3 \\ 1/3 \\ 0 \end{pmatrix}$.

**5.5.7.** For $k = 1$, there is nothing to prove. For $k > 1$, assume that $\mathcal{O}_k$ is an orthonormal basis for $\mathcal{S}_k$. First establish that $\mathcal{O}_{k+1}$ must be an orthonormal set. Orthogonality follows because for each $j < k+1$,

$$\begin{aligned} \langle \mathbf{u}_j | \mathbf{u}_{k+1} \rangle &= \left\langle \mathbf{u}_j \middle| \frac{1}{\nu_{k+1}} \left( \mathbf{x}_{k+1} - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \mathbf{u}_i \right) \right\rangle \\ &= \frac{1}{\nu_{k+1}} \left( \langle \mathbf{u}_j | \mathbf{x}_{k+1} \rangle - \left\langle \mathbf{u}_j \middle| \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \mathbf{u}_i \right\rangle \right) \\ &= \frac{1}{\nu_{k+1}} \left( \langle \mathbf{u}_j | \mathbf{x}_{k+1} \rangle - \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \langle \mathbf{u}_j | \mathbf{u}_i \rangle \right) \\ &= \frac{1}{\nu_{k+1}} \left( \langle \mathbf{u}_j | \mathbf{x}_{k+1} \rangle - \langle \mathbf{u}_j | \mathbf{x}_{k+1} \rangle \right) = 0. \end{aligned}$$

This together with the fact that each $\mathbf{u}_i$ has unit norm means that $\mathcal{O}_{k+1}$ is an orthonormal set. Now assume $\mathcal{O}_k$ is a basis for $\mathcal{S}_k$, and prove that $\mathcal{O}_{k+1}$ is a basis for $\mathcal{S}_{k+1}$. If $\mathbf{x} \in \mathcal{S}_{k+1}$, then $\mathbf{x}$ can be written as a combination

$$\mathbf{x} = \sum_{i=1}^{k+1} \alpha_i \mathbf{x}_i = \left( \sum_{i=1}^{k} \alpha_i \mathbf{x}_i \right) + \alpha_{k+1} \mathbf{x}_{k+1},$$

where $\sum_{i=1}^{k} \alpha_i \mathbf{x}_i \in \mathcal{S}_k = span\,(\mathcal{O}_k) \subset span\,(\mathcal{O}_{k+1})$. Couple this together with the fact that

$$\mathbf{x}_{k+1} = \nu_{k+1}\mathbf{u}_{k+1} + \sum_{i=1}^{k} \langle \mathbf{u}_i | \mathbf{x}_{k+1} \rangle \, \mathbf{u}_i \in span\,(\mathcal{O}_{k+1})$$

to conclude that $\mathbf{x} \in span\,(\mathcal{O}_{k+1})$. Consequently, $\mathcal{O}_{k+1}$ spans $\mathcal{S}_{k+1}$, and therefore $\mathcal{O}_{k+1}$ is a basis for $\mathcal{S}_{k+1}$ because orthonormal sets are always linearly independent.

**5.5.8.** If $\mathbf{A} = \mathbf{Q}_1\mathbf{R}_1 = \mathbf{Q}_2\mathbf{R}_2$ are two rectangular QR factorizations, then (5.5.6) implies $\mathbf{A}^T\mathbf{A} = \mathbf{R}_1^T\mathbf{R}_1 = \mathbf{R}_2^T\mathbf{R}_2$. It follows from Example 3.10.7 that $\mathbf{A}^T\mathbf{A}$ is positive definite, and $\mathbf{R}_1 = \mathbf{R}_2$ because the Cholesky factorization of a positive definite matrix is unique. Therefore, $\mathbf{Q}_1 = \mathbf{A}\mathbf{R}_1^{-1} = \mathbf{A}\mathbf{R}_2^{-1} = \mathbf{Q}_2$.

**5.5.9.** (a) **Step 1:** $fl\,\|\mathbf{x}_1\| = 1$, so $\mathbf{u}_1 \leftarrow \mathbf{x}_1$.

**Step 2:** $\mathbf{u}_1^T\mathbf{x}_2 = 1$, so

$$\mathbf{u}_2 \leftarrow \mathbf{x}_2 - \left(\mathbf{u}_1^T\mathbf{x}_2\right)\mathbf{u}_1 = \begin{pmatrix} 0 \\ 0 \\ -10^{-3} \end{pmatrix} \quad \text{and} \quad \mathbf{u}_2 \leftarrow \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}.$$

**Step 3:** $\mathbf{u}_1^T\mathbf{x}_3 = 1$ and $\mathbf{u}_2^T\mathbf{x}_3 = 0$, so

$$\mathbf{u}_3 \leftarrow \mathbf{x}_3 - \left(\mathbf{u}_1^T\mathbf{x}_3\right)\mathbf{u}_1 - \left(\mathbf{u}_2^T\mathbf{x}_3\right)\mathbf{u}_2 = \begin{pmatrix} 0 \\ 10^{-3} \\ -10^{-3} \end{pmatrix} \text{ and } \mathbf{u}_3 \leftarrow \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} = \begin{pmatrix} 0 \\ .709 \\ -.709 \end{pmatrix}.$$

Therefore, the result of the classical Gram–Schmidt algorithm using 3-digit arithmetic is

$$\mathbf{u}_1 = \begin{pmatrix} 1 \\ 0 \\ 10^{-3} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{u}_3 = \begin{pmatrix} 0 \\ .709 \\ -.709 \end{pmatrix},$$

which is not very good because $\mathbf{u_2}$ and $\mathbf{u}_3$ are not even close to being orthogonal.

(b) **Step 1:** $fl\,\|\mathbf{x}_1\| = 1$, so

$$\{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3\} \leftarrow \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}.$$

**Step 2:** $\mathbf{u}_1^T\mathbf{u}_2 = 1$ and $\mathbf{u}_1^T\mathbf{u}_3 = 1$, so

$$\mathbf{u}_2 \leftarrow \mathbf{u}_2 - \left(\mathbf{u}_1^T\mathbf{u}_2\right)\mathbf{u}_1 = \begin{pmatrix} 0 \\ 0 \\ -10^{-3} \end{pmatrix}, \quad \mathbf{u}_3 \leftarrow \mathbf{u}_3 - \left(\mathbf{u}_1^T\mathbf{u}_3\right)\mathbf{u}_1 = \begin{pmatrix} 0 \\ 10^{-3} \\ -10^{-3} \end{pmatrix},$$

and then

$$\mathbf{u}_2 \leftarrow \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}.$$

**Step 3:** $\mathbf{u}_2^T\mathbf{u}_3 = 10^{-3}$, so

$$\mathbf{u}_3 \leftarrow \mathbf{u}_3 - \left(\mathbf{u}_2^T\mathbf{u}_3\right)\mathbf{u}_2 = \begin{pmatrix} 0 \\ 10^{-3} \\ 0 \end{pmatrix} \quad \text{and} \quad \mathbf{u}_3 \leftarrow \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}.$$

Thus the modified Gram–Schmidt algorithm produces

$$\mathbf{u}_1 = \begin{pmatrix} 1 \\ 0 \\ 10^{-3} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}, \quad \mathbf{u}_3 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix},$$

which is as close to being an orthonormal set as one could reasonably hope to obtain by using 3-digit arithmetic.

**5.5.10.** Yes. In both cases $r_{ij}$ is the $(i,j)$-entry in the upper-triangular matrix R in the QR factorization.

**5.5.11.** $p_0(x) = 1/\sqrt{2}, \quad p_1(x) = \sqrt{3/2}\,x, \quad p_2(x) = \sqrt{5/8}\,(3x^2 - 1)$

## Solutions for exercises in section 5. 6

**5.6.1.** (a), (c), and (d).

**5.6.2.** Yes, because $\mathbf{U}^*\mathbf{U} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

**5.6.3.** (a) Eight: $\mathbf{D} = \begin{pmatrix} \pm 1 & 0 & 0 \\ 0 & \pm 1 & 0 \\ 0 & 0 & \pm 1 \end{pmatrix}$ (b) $2^n$ : $\mathbf{D} = \begin{pmatrix} \pm 1 & 0 & \cdots & 0 \\ 0 & \pm 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \pm 1 \end{pmatrix}$

(c) There are infinitely many because each diagonal entry can be any point on the unit circle in the complex plane—these matrices have the form given in part (d) of Exercise 5.6.1.

**5.6.4.** (a) When $\alpha^2 + \beta^2 = 1/2$. (b) When $\alpha^2 + \beta^2 = 1$.

**5.6.5.** (a) $(\mathbf{UV})^*(\mathbf{UV}) = \mathbf{V}^*\mathbf{U}^*\mathbf{UV} = \mathbf{V}^*\mathbf{V} = \mathbf{I}$.

(b) Consider $\mathbf{I} + (-\mathbf{I}) = \mathbf{0}$.

(c)

$$\begin{aligned} \begin{pmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{pmatrix}^* \begin{pmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{pmatrix} &= \begin{pmatrix} \mathbf{U}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{V}^* \end{pmatrix} \begin{pmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{U}^*\mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}^*\mathbf{V} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}. \end{aligned}$$

**5.6.6.** Recall from (3.7.8) or (4.2.10) that $(\mathbf{I}+\mathbf{A})^{-1}$ exists if and only if $N(\mathbf{I}+\mathbf{A}) = \mathbf{0}$, and write $\mathbf{x} \in N(\mathbf{I}+\mathbf{A}) \implies \mathbf{x} = -\mathbf{Ax} \implies \mathbf{x}^*\mathbf{x} = -\mathbf{x}^*\mathbf{Ax}$. But taking the conjugate transpose of both sides yields $\mathbf{x}^*\mathbf{x} = -\mathbf{x}^*\mathbf{A}^*\mathbf{x} = \mathbf{x}^*\mathbf{Ax}$, so $\mathbf{x}^*\mathbf{x} = 0$, and thus $\mathbf{x} = \mathbf{0}$. Replacing $\mathbf{A}$ by $-\mathbf{A}$ in Exercise 3.7.6 gives $\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1} = (\mathbf{I}+\mathbf{A})^{-1}\mathbf{A}$, so

$$\begin{aligned} (\mathbf{I}-\mathbf{A})(\mathbf{I}+\mathbf{A})^{-1} &= (\mathbf{I}+\mathbf{A})^{-1} - \mathbf{A}(\mathbf{I}+\mathbf{A})^{-1} \\ &= (\mathbf{I}+\mathbf{A})^{-1} - (\mathbf{I}+\mathbf{A})^{-1}\mathbf{A} = (\mathbf{I}+\mathbf{A})^{-1}(\mathbf{I}-\mathbf{A}). \end{aligned}$$