Orthogonal Matrices and QR.

Learning Goals: learn about orthogonal matrices and their use in simplifying the least squares problem, and the QR factorization and its speed improvements to the least squares problem.

So orthogonal vectors make things much easier. A great example is projecting onto a subspace. Normally, the projection is complicated, but as we have seen when you have an orthogonal basis, all you have to do is add up the projections onto the individual directions. If they are orthonormal, all you have to do to find these projections is a simple dot product. Note that we computed projection matrices by putting a basis into the columns of a matrix. So let’s make the

**Definition:** if the columns of a matrix are orthonormal, the matrix itself is called *orthogonal*. Such matrices are usually denoted by the letter $Q$.

Notice that $Q^TQ = I$. This is true even if $Q$ is not square. In case $Q$ is square, of course this means that $Q^{-1} = Q^T$. But we might be dealing with some subspace, and not need an orthonormal basis for the entire space.

If the columns of $Q$ are merely orthogonal, not unit length, then the matrix would not get the letter $Q$. Instead, we might call it $M$, and $M^TM$ is diagonal (the entries on the diagonal are the squares of the length of the column vectors in $Q$).

Several familiar geometric transformations have orthogonal matrices:

1. Rotation in the plane is affected $v$ getting sent to $Rv$, where
   $$ R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}. $$

2. Permutations. Swapping the coordinates all around (which is a combination of rotations and reflections—see below) produces a permutation matrix, $P$. These have come up before, in the $PA = LU$ factorization. We have already seen that for these matrices $P^{-1}$ and $P^T$ are the same matrix.

3. Reflections. Let $v$ be any vector in $\mathbb{R}^n$ and $V$ the entire hyperplane orthogonal to $v$. Reflection in this hyperplane can be accomplished by breaking a vector $x$ into a part along $v$ and a part in $V$, and then changing the sign of the $v$ part. That is, $x$ gets sent to itself minus twice its projection onto $v$. Thus the matrix is $I - 2P$ where this time $P$ is the projection matrix onto $v$. So we need $I - 2vv^T / (v^Tv)$. If $v$ is a unit vector, the final division is unnecessary and we get $I - 2v^Tv$. Check that this matrix has the same transpose and inverse!

Notice something very important about these geometric transformations: they are all rigid. That is, they do not affect lengths or angles.

**Theorem:** let $Q$ be any (not necessarily square) orthonormal matrix. Then $Q$ preserves dot products. That is, $(Qx) \cdot (Qy) = x \cdot y$. In particular, $\|Qx\| = \|x\|$.

**Proof:**

$$(Qx) \cdot (Qy) = (Qx)^T(Qy) = x^TQ^TQy = x^Ty = x \cdot y. \quad \|Qx\| = \sqrt{(Qx) \cdot (Qx)}$$

so the length property follows from the dot product property.
Now back to the projection and least squares problems. If we have an orthonormal basis for a subspace, and for the projection matrix onto this subspace, we obtain \( Q(Q^T Q)^{-1} Q^T \). But of course \( Q^T Q \) is the identity, so the projection is just \( QQ^T \). The solution to the least squares problem transforms from \( A^T A x = A^T b \) into simply \( x = Q^T b \).

Let’s say we have a matrix \( A \) with independent columns. The system \( A x = b \) has at most one solution for any \( b \), and may be inconsistent and have none at all. That was where least squares came in. We had to project \( b \) onto the column space of \( A \). Let’s orthogonalize the columns of \( A \) (by Gram-Schmidt) and see what this does for us.

Since it is columns we are dealing with, in order to do the column operations we must multiply on the right by elementary matrices. If, say, \( A \) has four columns and we want to subtract twice the second from the fourth, we multiply on the right by

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & -2 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

So we multiply by a bunch of these to find \( AE_1 E_2 \ldots E_n = Q \). Now a slick trick is to multiply this by the inverses to all of the \( E’s: A = Q E_n^{-1} \ldots E_1^{-1} = QR \). The nifty thing to notice is that the order of eliminations makes it possible to read off the \( R \) matrix from the multipliers used—just like when we did \( A = LU \). Here, we call the resulting matrix \( R \) for “right triangular.” Using the example of (1, 1, 1, 1), (1, 1, 1, 2), and (1, 1, 2, 2) we work as follows. Start with \( A \) and \( R = I \). First, set \( w_1 \) to be (1, 1, 1, 1). Then \( v_2 \cdot w_1 / w_1 \cdot w_1 = 5/4 \) and \( v_3 \cdot w_1 / w_1 \cdot w_1 = 6/4 \), so we make our next set of matrices: \( Q \) so far is

\[
\begin{bmatrix}
1 & -1/4 & -1/2 \\
1 & -1/4 & -1/2 \\
1 & -1/4 & 1/2 \\
1 & 3/4 & 1/2
\end{bmatrix}
\]

and \( R \) so far is

\[
\begin{bmatrix}
1 & 5/4 & 3/2 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

Next, set \( w_2 \) equal to this new second column. Then \( w_2 \cdot v_3 / w_2 \cdot w_2 = (1/2) / (3/4) = 2/3 \), so we subtract 2/3 of column 2 from column 3, and put 2/3 into the space in \( R \): \( Q \) so far is

\[
\begin{bmatrix}
1 & -1/4 & -1/3 \\
1 & -1/4 & -1/3 \\
1 & -1/4 & 2/3 \\
1 & 3/4 & 0
\end{bmatrix}
\]

and \( R \) so far is

\[
\begin{bmatrix}
1 & 5/4 & 3/2 \\
0 & 1 & 2/3 \\
0 & 0 & 1
\end{bmatrix}
\]

Check that \( QR = A \). As a final stage, we pull the lengths out of the columns of \( Q \) and put them in \( R \):

\[
\begin{bmatrix}
1 & -1/4 & -1/3 \\
1 & -1/4 & -1/3 \\
1 & -1/4 & 2/3 \\
1 & 3/4 & 0
\end{bmatrix}
\]
\[
\begin{bmatrix}
1/2 & -1/2\sqrt{3} & -1/\sqrt{6} \\
1/2 & -1/2\sqrt{3} & -1/\sqrt{6} \\
1/2 & -1/2\sqrt{3} & 2/\sqrt{6} \\
1/2 & \sqrt{3}/2 & 0
\end{bmatrix}
\begin{bmatrix}
2 & \sqrt{3}/2 \\
0 & \sqrt{3}/2 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
1 & 5/4 & 3/2 \\
0 & 1 & 2/3 \\
0 & 0 & 1
\end{bmatrix}
= QR. \text{ Notice that the multiples of the columns of } Q \text{ went into the rows of } R. \text{ Check that } QR = A \text{ and that } Q \text{ is orthogonal.}
\]

Now why would we have gone to all this work just to find another factorization of a matrix, especially one that requires independent columns? Let’s look at \( Ax = b \). If we are going to need to solve the least squares problem, we will need to solve \( A^T Ax = A^T b \). Now \( A = QR \). So we rewrite as \( R^T Q^T Ax = R^T Q^T b \). Now the matrix \( R^T \) is triangular, and hence \( invertible \), and so we multiply by its inverse. Also, \( Q^T Q = I \). This leaves \( Rx = Q^T b \).

And that’s why we want this factorization. If we need to solve several least squares problems involving the same coefficients, we can just multiply each \( b \) by \( Q^T \) (which simply projects each \( b \) onto the column space of \( A! \)). Then solve \( Rx = Q^T b \) by \( back \ substitution \)!

Example: Find the \( QR \) factorization of \( A = \begin{bmatrix}
1 & 3 & 3 \\
1 & 2 & 2 \\
1 & 3 & 4 \\
1 & 2 & 3
\end{bmatrix} \) and use it to find least squares solutions to \( Ax = b \) for the two right-hand sides \( b_1 = \begin{bmatrix}
10 \\
7 \\
11 \\
8
\end{bmatrix} \) and \( b_2 = \begin{bmatrix}
0 \\
0 \\
0 \\
1
\end{bmatrix} \). Note that \( b_1 \) is actually in the column space of \( A \) and has unique solution \( x = (1, 2, 1) \), while \( b_2 \) is not in the column space.

First, we will solve by the normal equations, so that we know we are getting the right answers. \( A^T A = \begin{bmatrix}
4 & 10 & 12 \\
10 & 26 & 31 \\
12 & 31 & 38
\end{bmatrix} \), \( A^T b_1 = \begin{bmatrix}
36 \\
93 \\
112
\end{bmatrix} \) and \( A^T b_2 = \begin{bmatrix}
1 \\
2 \\
3
\end{bmatrix} \). We will brute force solve \( A^T Ax = A^T b \) by doing a quick row reduction on the augmented matrix:
and now the rest is easy by back substitution: we get \( x_1 = (1, 2, 1) \) and \( x_2 = (5/4, -1, \frac{1}{2}) \). Now that we have baseline solutions, let’s do the \( QR \) work and make sure we get the same answers!

To start, the first column has length 2, so we divide by 2 (and place this multiplier into the \( R \) matrix as we create \( QR \)): 

\[
\begin{bmatrix}
1/2 & 1/2 & 0 \\
1/2 & -1/2 & -1 \\
1/2 & 1/2 & 1 \\
1/2 & -1/2 & 0
\end{bmatrix}
\begin{bmatrix}
2 & 5 & 6 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}.
\]

The second column is already unit length. Projecting the third column onto it, we find we need to subtract the second from the third, yielding our final \( QR \) factorization

\[
\begin{bmatrix}
1/2 & 1/2 & -1/2 \\
1/2 & -1/2 & -1/2 \\
1/2 & 1/2 & 1/2 \\
1/2 & -1/2 & 1/2
\end{bmatrix}
\begin{bmatrix}
2 & 5 & 6 \\
0 & 1 & 1 \\
0 & 0 & 1
\end{bmatrix}.
\]

Now we find \( Q^T b_1 = (18, 3, 1) \) and \( Q^T b_2 = (\frac{1}{2}, -\frac{1}{2}, \frac{1}{2}) \). We can easily solve \( Rx = \) these two by back substitution, to get \( x_1 = (1, 2, 1) \) and \( x_2 = (5/4, -1, \frac{1}{2}) \).

Reading: 4.4
Problems: 4.4: 1, 3, 4, 5, 6, 7, 10, 11, 13, 14, 15, 16, 18, 21, 22, 23, 24, 30, 31,

Carefully find the \( QR \) factorization \( A = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 1 & 2 \\ 2 & 1 & -1 \\ 2 & 1 & 0 \end{bmatrix} \).