**Example 78.** Find the QR decomposition of $A = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$.

**Solution. (final answer only)** $A = QR$ with $Q = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix}$ and $R = \begin{bmatrix} \sqrt{2} & \frac{1}{\sqrt{2}} & \sqrt{2} \\ 0 & \frac{1}{\sqrt{2}} & \sqrt{2} \\ 0 & 0 & 1 \end{bmatrix}$.

**Example 79.** One practical application of the QR decomposition is solving systems of linear equations.

$$\begin{aligned} A\boldsymbol{x} = \boldsymbol{b} \quad &\Longleftrightarrow \quad QR\boldsymbol{x} = \boldsymbol{b} \qquad \text{(now, multiply with } Q^T \text{ from the left)} \\ &\Longrightarrow \quad R\boldsymbol{x} = Q^T\boldsymbol{b} \end{aligned}$$

The last system is triangular and can be solved by back substitution.

A couple of comments are in order:

- If $A$ is $n \times n$ and invertible, then the "$\Longrightarrow$" is actually a "$\Longleftrightarrow$".
- The equation $R\boldsymbol{x} = Q^T\boldsymbol{b}$ is always consistent! (Can you see that? Recall that $R$ is upper triangular.) Indeed, if $A$ is not $n \times n$ or not invertible, then $R\boldsymbol{x} = Q^T\boldsymbol{b}$ gives the least squares solutions!
  **Why?** $A^TA\hat{\boldsymbol{x}} = A^T\boldsymbol{b} \iff \underbrace{(QR)^TQR}_{=R^TQ^TQR}\hat{\boldsymbol{x}} = (QR)^T\boldsymbol{b} \iff R^TR\hat{\boldsymbol{x}} = R^TQ^T\boldsymbol{b} \iff R\hat{\boldsymbol{x}} = Q^T\boldsymbol{b}$
  [For the last step we need that $R$ is invertible, which is always the case when $A$ is $m \times n$ of rank $n$.]
- So, how does the QR way of solving linear systems compare to our beloved Gaussian elimination (LU)? It turns out that QR is a little slower than LU but makes up for it in "numerical stability".
  **What does that mean?** When computing numerically, we use floating point arithmetic and approximate each number by an expression of the form $0.1234 \cdot 10^{-16}$. A certain (fixed) number of bits is used to store the part $0.1234$ (here, $4$ decimal places of accuracy) as well as the exponent $-16$.
  Now, here is something terrible that can happen in numerical computations: mathematically, the quantities $x$ and $(x+1) - 1$ are exactly the same. However, numerically, they might not. Take, for instance, $0.1234 \cdot 10^{-16}$. Then, to an accuracy of $4$ decimal places, $x + 1 = 0.1000 \cdot 10^1$, so that $(x+1) - 1 = 0.0000$. But $x \neq 0$. We completely lost all the information about $x$.
  To be numerically stable, an algorithm must avoid issues like that.

> $\hat{\boldsymbol{x}}$ is a least squares solution of $A\boldsymbol{x} = \boldsymbol{b}$
> $\iff R\hat{\boldsymbol{x}} = Q^T\boldsymbol{b}$ (where $A = QR$)

**Example 80. (homework)** Suppose $Q$ has orthonormal columns. What is the projection matrix for orthogonally projecting onto $\operatorname{col}(Q)$?

**Solution.** Recall that, to project onto $\operatorname{Col}(A)$, the projection matrix is $P = A(A^TA)^{-1}A^T$.
Since $Q^TQ = I$, to project onto $\operatorname{Col}(Q)$, the projection matrix is $P = QQ^T$.

**Comment.** A familiar special case is when we project onto a unit vector $\boldsymbol{q}$: in that case, the projection of $\boldsymbol{b}$ onto $\boldsymbol{q}$ is $(\boldsymbol{q} \cdot \boldsymbol{b})\boldsymbol{q} = \boldsymbol{q}(\boldsymbol{q}^T\boldsymbol{b}) = (\boldsymbol{q}\boldsymbol{q}^T)\boldsymbol{b}$, so the projection matrix here is $\boldsymbol{q}\boldsymbol{q}^T$.

**Example 81. (homework)** Again, if $P$ is a projection matrix, then what is $P^2$?

**Solution.** We already observed in Example 67 that $P^2 = P$.
If $P$ is the projection onto $W$, we now know that we can always select an orthonormal basis for $W$. Using these basis vectors as the columns of the matrix $Q$, we get $P = QQ^T$.
Since $Q^TQ = I$, we find that $P^2 = (QQ^T)(QQ^T) = Q\underbrace{(Q^TQ)}_{I}Q^T = QQ^T = P$.

Armin Straub
straub@southalabama.edu