Lecture 7 - *QR Factorization*

OBJECTIVE:
We introduce the algorithmic idea behind the QR factorization, which is possibly the most important idea in numerical linear algebra.

◇ REDUCED QR FACTORIZATION

For many applications, we are interested in successive column spaces of a matrix $\mathbf{A}$.
i.e., the spaces spanned by the columns $\mathbf{a}_1, \mathbf{a}_2, \ldots$ of $\mathbf{A}$ in succession

$$\langle \mathbf{a}_1 \rangle \subseteq \langle \mathbf{a}_1, \mathbf{a}_2 \rangle \subseteq \langle \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \rangle \subseteq \ldots$$

e.g., $\langle \mathbf{a}_1 \rangle$ is the 1-D subspace spanned by $\mathbf{a}_1$
$\langle \mathbf{a}_1, \mathbf{a}_2 \rangle$ is the 2-D subspace spanned by $\mathbf{a}_1, \mathbf{a}_2$, etc.

The idea of the QR factorization is to successively construct orthonormal vectors $\mathbf{q}_1, \mathbf{q}_2, \ldots$ *that span these successive spaces.*

Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ $(m \geq n)$ have full rank n.
We want to find the sequence $\mathbf{q}_1, \mathbf{q}_2, \ldots$ of orthonormal vectors such that

$$\langle \mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_j \rangle = \langle \mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_j \rangle$$

for $j = 1, 2, \ldots, n$.

From Lecture 1, we can see that this amounts to

$$\begin{aligned} &\left[\; \mathbf{a}_1 \; \middle| \; \mathbf{a}_2 \; \middle| \; \ldots \; \middle| \; \mathbf{a}_n \; \right] \\ &= \left[\; \mathbf{q}_1 \; \middle| \; \mathbf{q}_2 \; \middle| \; \ldots \; \middle| \; \mathbf{q}_n \; \right] \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix} \end{aligned}$$

where $r_{kk} \neq 0$ for $k = 1, 2, \ldots, n$ so that the $\mathbf{a}_k$ can be expressed as a linear combination of the $\mathbf{q}_k$ and vice versa.

Written out

$$\left.\begin{aligned}
\mathbf{a}_1 &= r_{11}\mathbf{q}_1 \\
\mathbf{a}_2 &= r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2 \\
&\vdots \\
\mathbf{a}_n &= r_{1n}\mathbf{q}_1 + r_{2n}\mathbf{q}_2 + \ldots + r_{nn}\mathbf{q}_n
\end{aligned}\right\} (*)$$

or as a matrix equation

$$\mathbf{A} = \hat{\mathbf{Q}}\hat{\mathbf{R}},$$

where $\hat{\mathbf{Q}} \in \mathbb{R}^{m\times n}$ has orthonormal columns
and $\hat{\mathbf{R}} \in \mathbb{R}^{n\times n}$ is upper triangular.
This is called the *reduced QR factorization of* $\mathbf{A}$.

$\diamond$ FULL QR FACTORIZATION

Similar to a full SVD, a full QR factorization of $\mathbf{A} \in \mathbb{R}^{m \times n}$ $(m \geq n)$ appends an additional $(m-n)$ orthonormal columns to $\hat{\mathbf{Q}}$ to make it an $m \times m$ orthogonal matrix.

In the process, $(m-n)$ rows of zeros are appended to $\hat{\mathbf{R}}$, making it an $m \times n$ matrix $\mathbf{R}$

Note 1. *In the full QR, the "silent" columns* $\mathbf{q}_j$, $j > n$ *are orthogonal to* range$(\mathbf{A})$.
If $\mathbf{A}$ *has full rank, they form an orthonormal basis for* range$(\mathbf{A})^{\perp}$ *(the space orthogonal to* range$(\mathbf{A})$*) or equivalently,* null$(\mathbf{A}^T)$.

$\diamond$ GRAM-SCHMIDT ORTHOGONALIZATION

The equations $(*)$ suggest a method for computing reduced QR factorizations:
Given $\mathbf{a}_1, \mathbf{a}_2, \ldots$ we can construct $\mathbf{q}_1, \mathbf{q}_2, \ldots$ and entries r_{ij} by successive orthogonalization.
This is an old idea called *Gram-Schmidt orthogonalization*.

At step j, we want a unit vector $\mathbf{q}_j \in \langle \mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_j \rangle$ that is orthogonal to $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_{j-1}$.
It turns out we know how to do this from Lecture 2: the vector

$$\mathbf{v}_j = \mathbf{a}_j - (\mathbf{q}_1^T \mathbf{a}_j)\mathbf{q}_1 - (\mathbf{q}_2^T \mathbf{a}_j)\mathbf{q}_2 - \ldots - (\mathbf{q}_{j-1}^T \mathbf{a}_j)\mathbf{q}_{j-1}$$

does the trick, except it does not have unit norm.
So, if we divide $\mathbf{v}_j$ by $\|\mathbf{v}_j\|_2$, we get a suitable $\mathbf{q}_j$.

This allows us to rewrite $(*)$ as

$$\left.\begin{aligned}
\mathbf{q}_1 &= \tfrac{\mathbf{a}_1}{r_{11}} \\
\mathbf{q}_2 &= \tfrac{\mathbf{a}_2 - r_{12}\mathbf{q}_1}{r_{22}} \\
\mathbf{q}_3 &= \tfrac{\mathbf{a}_3 - r_{13}\mathbf{q}_1 - r_{23}\mathbf{q}_2}{r_{33}} \\
&\vdots \\
\mathbf{q}_n &= \tfrac{\mathbf{a}_n - \sum_{i=1}^{n-1} r_{in}\mathbf{q}_i}{r_{nn}}
\end{aligned}\right\} (**)$$

It is clear the appropriate coefficients r_{ij} in $(**)$ are

$$r_{jj} = \mathbf{q}_i^T \mathbf{a}_j \qquad i \neq j$$

The coefficients r_{jj} are chosen for normalization:

$$r_{jj} = \|\mathbf{a}_j - \sum_{i=1}^{j-1} r_{ij}\mathbf{q}_i\|_2$$

Note that the sign of r_{jj} is not determined!
By convention, we choose $r_{jj} > 0$.

This leads to a factorization $\mathbf{A} = \hat{\mathbf{Q}}\hat{\mathbf{R}}$ where $\hat{\mathbf{R}}$ has positive diagonal entries.

Formally, the algorithm just described is called the *classical Gram-Schmidt iteration*.
The reason for "classical" is because this was the way it was originally envisioned and implemented.
However, it turns out to be *UNSTABLE* numerically due to roundoff errors on the computer.
Next lecture, we will cover the *modified Gram-Schmidt iteration* which does the same job but is numerically stable.

ALGORITHM 7.1: CLASSICAL GRAM-SCHMIDT (UNSTABLE)

for $j = 1$ to n **do**
  $\mathbf{v}_j = \mathbf{a}_j$
  for $i = 1$to $j - 1$ **do**
    $r_{ij} = \mathbf{q}_i^T \mathbf{a}_j$
    $\mathbf{v}_j = \mathbf{v}_j - r_{ij}\mathbf{q}_i$
  end for
  $r_{jj} = \|\mathbf{v}_j\|_2$
  $\mathbf{q}_j = \frac{\mathbf{v}_j}{r_{jj}}$
end for

$\diamond$ EXISTENCE AND UNIQUENESS OF THE QR FACTORIZATION

Bottom line:
All matrices have QR factorizations.
And, if we impose the restriction $r_{jj} > 0$, then the QR factorization is unique.

Theorem 1. *Every* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *(*$m \geq n$*) has a full QR factorization (and hence also a reduced QR factorization).*

Theorem 2. *Every* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *(*$m \geq n$*) of full rank has a unique reduced QR factorization* $\mathbf{A} = \hat{\mathbf{Q}}\hat{\mathbf{R}}$ *with* $r_{jj} > 0$*.*

$\diamond$ SOLUTION OF $\mathbf{Ax} = \mathbf{b}$ BY QR FACTORIZATION

Suppose we wish to solve $\mathbf{Ax} = \mathbf{b}$ for $\mathbf{x}$, where $\mathbf{A} \in \mathbb{R}^{m \times m}$ is nonsingular.
If $\mathbf{A} = \mathbf{QR}$ is a QR factorization, then

$$\mathbf{QRx} = \mathbf{b}$$

or

$$\mathbf{R}\mathbf{x} = \mathbf{Q}^T\mathbf{b}$$

The right-hand side is easy to compute once we know $\mathbf{Q}$.
Then the solution $\mathbf{x}$ can be easily solved by the method of back substitution because $\mathbf{R}$ is upper triangular (solve for $\mathbf{x}_m$ directly, then use it to find $\mathbf{x}_{m-1}$, etc.).

In summary, to solve $\mathbf{A}\mathbf{x} = \mathbf{b}$

1. Compute a QR factorization $\mathbf{A} = \mathbf{Q}\mathbf{R}$

2. Compute $\mathbf{y} = \mathbf{Q}^T\mathbf{b}$

3. Solve $\mathbf{R}\mathbf{x} = \mathbf{y}$ for $\mathbf{x}$ by back substitution

This is an excellent method for solving linear systems, but, it is not the standard:
Gaussian elimination (or LU factorization) is used in practice because it is twice as fast.
However, we will find a good use for QR soon!