Classical vs. Modified GS

From Lecture 9 in Text.

\[
[U, X] = qr(randn(80));
\]

\[
[V, X] = qr(randn(80));
\]

\[
S = \text{diag}(2.\hat{(-1:-1:-80)});
\]

\[
A = U*S*V';
\]
GS

\[ [QC,RC] = \text{cgs}(A); \]

\[ [QM,RM] = \text{mgs}(A); \]

Plot \( r_{jj} = \| P_j a_j \|_2 \) vs. machine epsilon.

Test orthogonality
We need something better!
QR vs. Householder Triangularization

Let $A$ be full rank $m \times n$, $m \geq n$.

Goal: Introduce a sequence of $m \times m$ unitary matrices $Q_k$ so

$$Q_n Q_{n-1} \cdots Q_1 A = R$$

so $Q_k$ introduces zeros below the main diagonal.
Householder Matrices

First, let’s see if we can find $F$ so that

$$
F \begin{bmatrix}
    x_1 \\
    x_2 \\
    \vdots \\
    x_n
\end{bmatrix} = \begin{bmatrix}
    \|x\|_2 \\
    0 \\
    \vdots \\
    0
\end{bmatrix} = \|x\|e_1
$$
Let’s see what this looks like in $\mathbb{R}^2$.

Draw $x$, **reflect** across line (span$\{v\}$) to the horizontal-axis!

$$v = \|x\|e_1 - x.$$
Define a Reflection

- First, project $x$ orthogonally onto span{$v$}.
- If $P = \frac{vv^*}{v^*v}$, then $Px$ is in span{$v$}
- Thus, $x - Px$ is orthogonal to that...
- And $x - 2Px = (I - 2P)x$ is the reflection
Reflection Example

\[ x = \begin{bmatrix} 3 \\ 4 \end{bmatrix}, \quad v = 5 \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \end{bmatrix} \]

\[ F = I - 2\frac{vv^*}{v^*v}, \quad Fx = \begin{bmatrix} 3/5 & 4/5 \\ 4/5 & -3/5 \end{bmatrix} x \]
\[ Fx = \begin{bmatrix} 3/5 & 4/5 \\ 4/5 & -3/5 \end{bmatrix} x = \begin{bmatrix} 5 \\ 0 \end{bmatrix}. \]
Reflection the other direction

Reflect to $-\|x\|e_1$, so then $v = -\|x\|e_1 - x$.

$$F = I - 2\frac{vv^*}{v^*v} \text{ and so } Fx = -\|x\|e_1.$$
Which reflection??

We want our algorithm to be insensitive to rounding errors, so we should reflect to the vector, \( \pm \|x\| e_1 \), that is furthest from \( x \) itself.

Therefore \( v = -\text{sign}(x_1) \|x\| e_1 - x \), so \( v = \text{sign}(x_1) \|x\| e_1 + x \).

\[
F = I - 2 \frac{vv^*}{v^*v}.
\]
Generalization

For $\mathbb{C}^m$, this amounts to reflecting about a hyperplane of dimension $m - 1$. 
Householder QR Example

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

Let \( x = a_1 \).

\[ v = \text{sign}(x_1) \|x_1\| e_1 + x_1 = 3e_1 + a_1 = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \]
Householder QR

\[ F = I - 2 \frac{vv^*}{v^*v} = \begin{bmatrix} -2/3 & -2/3 & -1/3 \\ -2/3 & 11/15 & -2/15 \\ -1/3 & -2/15 & 14/15 \end{bmatrix} \]
Householder QR

\[ Q_1 = F \]

\[ \tilde{A} = Q_1 A = \begin{bmatrix} -3 & 1 \\ 0 & 4/5 \\ 0 & -3/5 \end{bmatrix}. \]

Now, we only want to zero out the last component of the 2nd column.
\[ Q_2 = \begin{bmatrix} 1 & 0 \\ 0 & F \end{bmatrix} \]

where \( F \) is \( 2 \times 2 \).

Note, \( Q_2 \) is orthogonal!
Householder QR

Let $x = \tilde{A}(2 : 3, 2)$. Define $F$ from this.

$$v = \text{sign}(x_1)\|x\|e_1 + x = \begin{bmatrix} 9/5 \\ -3/5 \end{bmatrix}$$

$$F = I - 2\frac{vv^*}{v^*v}.$$
Householder QR

\[ Q_2Q_1A = Q_2\tilde{A} \]

\[
= \begin{bmatrix}
1 & 0 & 0 \\
0 & -\frac{4}{5} & \frac{3}{5} \\
0 & \frac{3}{5} & \frac{4}{5}
\end{bmatrix}
\begin{bmatrix}
-3 & 1 \\
0 & \frac{4}{5} \\
0 & -\frac{3}{5}
\end{bmatrix}
\]

\[
= \begin{bmatrix}
-3 & 1 \\
0 & -1 \\
0 & 0
\end{bmatrix}
\]
Algorithm Efficiencies

- Did we have to compute all the entries in $Q_2$ in order to apply $Q_2$ to $\tilde{A}$?
- Note that $Q_2$ only changes $\tilde{A}$ in rows 2 and below.
- In general, $Q_k = \begin{bmatrix} I & 0 \\ 0 & F_k \end{bmatrix}$ will only affect changes in the current matrix in rows $k$ and below.
- Note that if $v$ has unit length, $F = I - 2vv^*$.  
- Note $Fr = (I - 2vv^*)r = r - 2vv^*r = r - 2v(v^*r)$
Householder QR

Computes $R$ by overwriting $A$

- **for** $k = 1 : n$
  
  $e1 = \text{zeros}(m - k, 1); e1(1) = 1$
  
  $x = A(k : m, k);$
  
  $vk = \text{sign}(x(1)) \times \text{norm}(x) \times e1 + x$
  
  $vk = vk / \text{norm}(vk)$
  
  $A(k : m, k : n) = A(k : m, k : n) - 2vk(vk^*A(k : m, k : n))$

- **end**
Roughly $2mn^2 - 2/3n^3$ flops.
Householder QR

The above code applies the sequence of $Q$ matrices implicitly to obtain $R$. But how do we get $Q$?
\[ Q_n \cdots Q_1 A = R \Rightarrow A = Q_1^* \cdots Q_n^* R = QR \]

why??
Fact: The product of unitary matrices is unitary.
Finding Q

- Often, the reason we want \( Q \) is to form matrix-vector products with \( Q \) or \( Q^* \) to solve linear systems or least squares problems.

- Thus, we don’t need \( Q \) explicitly if we store the \( v_k \) vectors!
Example: House-QR to Solve a System

\[ Ax = b \Rightarrow QRx = b \Rightarrow Rx = Q^*b \]

But \( Q^*b = Q_n \cdots Q_1 b \), so we apply the \( Q_k \)'s to \( b \) the same way we did to \( A \).
Implicitly compute $Q \ast b$

- for $k = 1 : n$

\[ b(k : m) = b(k : m) - 2v_k(v_k \ast b(k : m)) \]

- end

$O(mn)$ flops for this. Requires storage of the $v_k$. How many flops (storage) if the very naive way is used?
Obtaining $q_k$

So how would we get $q_k$ if we wanted it?
What about reduced QR?

This computes (implicitly) a unitary $Q$ and $m \times n \ R$. Where are $\hat{Q}, \hat{R}$?
QR for Least Squares

Assume $A$ is $m \times n$, $m \geq n$ and full rank. Use the Householder QR factorization to determine a solution (algorithm) to

$$\min_x \|Ax - b\|_2$$

• that does not involve matrix inverses
• that does not involve computing the normal equations