Lecture 6
Householder Reflectors and Givens Rotations

MIT 18.335J / 6.337J
Introduction to Numerical Methods

Per-Olof Persson (persson@mit.edu)
September 26, 2007

After all the steps we get a product of orthogonal matrices

Gram-Schmidt as Triangular Orthogonalization
- Gram-Schmidt multiplies with triangular matrices to make columns orthogonal, for example at the first step:

\[
\begin{bmatrix}
1 & -\frac{v_2}{v_1} & \cdots & -\frac{v_n}{v_1}
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
\vdots \\
v_n
\end{bmatrix}
=
\begin{bmatrix}
q_1 \\
q_2 \\
\vdots \\
q_n
\end{bmatrix}
\]

- After all the steps we get a product of triangular matrices

\[
A R_1 R_2 \cdots R_n = \hat{Q}
\]

- "Triangular orthogonalization"

Householder Triangularization
- The Householder method multiplies by unitary matrices to make columns triangular, for example at the first step:

\[
Q_1 A = \begin{bmatrix}
\tau_{11} & 0 & \cdots & 0 \\
0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
\vdots \\
v_n
\end{bmatrix}
\]

- After all the steps we get a product of orthogonal matrices

\[
Q_n \cdots Q_2 Q_1 A = R
\]

- "Orthogonal triangularization"

Introducing Zeros
- \(Q_k\) introduces zeros below the diagonal in column \(k\)
- Preserves all the zeros previously introduced

Householder Reflectors
- Let \(Q_k\) be of the form

\[
Q_k = \begin{bmatrix}
I & 0 \\
0 & F
\end{bmatrix}
\]

where \(I\) is \((k-1) \times (k-1)\) and \(F\) is \((m-k+1) \times (m-k+1)\)
- Create Householder reflector \(F\) that introduces zeros:

\[
x = \begin{bmatrix}
x \\
x \\
\vdots \\
x
\end{bmatrix},
F x = \begin{bmatrix}
\|x\| \\
0 \\
\vdots \\
0
\end{bmatrix} = \|x\| e_1
\]

Householder Reflectors
- Idea: Reflect across hyperplane \(H\) orthogonal to \(v = \|x\| e_1 - x\), by the unitary matrix

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

- Compare with projector

\[
P_{L_v} = I - \frac{vv^*}{v^*v}
\]
Choice of Reflectors
- We can choose to reflect to any multiple $z$ of $\|x\|e_1$ with $|z| = 1$
- Better numerical properties with large $\|v\|$, for example

$$v = \text{sign}(x_1)\|x\|e_1 + x$$

- Note: $\text{sign}(0) = 1$, but in MATLAB, $\text{sign}(0) = 0$

Applying or Forming $Q$
- Compute $Q^*b = Q_n \cdots Q_2 Q_1 b$ and $Qx = Q_1 Q_2 \cdots Q_n x$ implicitly
- To create $Q$ explicitly, apply to $x = I$

Algorithm: Implicit Calculation of $Q^*b$
for $k = 1$ to $n$
\[ b_{k:m} = b_{k:m} - 2v_k(v_k^*b_{k:m}) \]

Algorithm: Implicit Calculation of $Qx$
for $k = n$ downto 1
\[ x_{k:m} = x_{k:m} - 2v_k(v_k^*x_{k:m}) \]

The Householder Algorithm
- Compute the factor $R$ of a $QR$ factorization of $m \times n$ matrix $A$ ($m \geq n$)
- Leave result in place of $A$, store reflection vectors $v_k$ for later use

Algorithm: Householder QR Factorization
for $k = 1$ to $n$
\[ x = A_{k:m,k} \]
\[ v_k = \text{sign}(x_1)\|x\|e_1 + x \]
\[ v_k = v_k/\|v_k\| \]
\[ A_{k:m,k:n} = A_{k:m,k:n} - 2v_k(v_k^*A_{k:m,k:n}) \]

Operation Count - Householder QR
- Most work done by
\[ A_{k:m,k:n} = A_{k:m,k:n} - 2v_k(v_k^*A_{k:m,k:n}) \]
- Operations per iteration:
  - $2(m-k)(n-k)$ for the dot products $v_k^*A_{k:m,k:n}$
  - $(m-k)(n-k)$ for the outer product $2v_k(\cdots)$
  - $(m-k)(n-k)$ for the subtraction $A_{k:m,k:n} - \cdots$
  - $4(m-k)(n-k)$ total
- Including the outer loop, the total becomes
\[ \sum_{k=1}^{n} 4(m-k)(n-k) = 4 \sum_{k=1}^{n} (mn - k(m+n) + k^2) \]
\[ \approx 4mn^2 - 4(m+n)n^2/2 + 4n^3/3 = 2mn^2 - 2n^3/3 \]

Givens Rotations
- Alternative to Householder reflectors
- A Givens rotation $R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$ rotates $x \in \mathbb{R}^2$ by $\theta$
- To set an element to zero, choose $\cos \theta$ and $\sin \theta$ so that
\[
\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_i \\ x_j \end{bmatrix} = \begin{bmatrix} \sqrt{x_i^2 + x_j^2} \\ 0 \end{bmatrix}
\]
or
\[
\cos \theta = \frac{x_i}{\sqrt{x_i^2 + x_j^2}}, \quad \sin \theta = \frac{-x_j}{\sqrt{x_i^2 + x_j^2}}
\]

Givens QR
- Introduce zeros in column from bottom up

\[
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
0 & \times & \times \\
0 & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\end{bmatrix}
\begin{bmatrix}
(3,4) \\
(2,3) \\
(1,2) \\
(3,4) \\
(2,3) \\
(1,2) \\
(3,4) \\
(2,3) \\
(1,2) \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\times & \times & \times \\
\end{bmatrix}
\]
- Flop count $3mn^2 - n^3$ (or 50% more than Householder QR)