

CS 450: Numerical Analysis

Lecture 7

Chapter 3 – Linear Least Squares

Stability and Efficiency of Linear Least Squares Algorithms

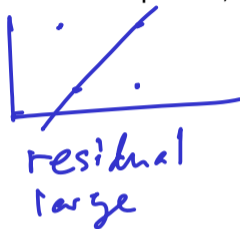
Edgar Solomonik

Department of Computer Science
University of Illinois at Urbana-Champaign

February 7, 2018

Conditioning of Linear Least Squares

- ▶ Consider fitting a line to a collection of points, then perturbing the points:



- ▶ Linear least squares is ill-posed for any A , unless we consider solving for a particular b

if residuals is small

$\kappa(A)$ condition number of A
is condition of LLS

$\kappa(A) \neq \|A\| \|A^{-1}\|$ if A is rectangular

$$\kappa(A) = \frac{\sigma_{\max}}{\sigma_{\min}}$$

Stability of Normal Equations

- ▶ Normal equations solve $A^T A x = A b$:

$$\kappa(A^T A) = \kappa(A)^2$$

y

$$S x = y$$

Why is $\frac{\|Ax\|}{\|b\|} \leq 1$

$$R x = Q^T b$$

norm
terms

- ▶ We can often improve the solution to the normal equations by performing them again on the reduced Q :

chol $(A^T A) = L L^T \Rightarrow A^T A = L L^T$

$$A = Q R \Rightarrow A^T A = R^T R$$

$\Delta \nabla$

Chol on $y - QR$

Chol on $Q^T b$

so $A \approx \hat{Q} \hat{R} = Q \bar{R} \bar{R}^T$

$\hat{Q} = Q \bar{R}$

\bar{R}

Gram-Schmidt Orthogonalization

- ▶ Classical and Modified Gram-Schmidt process for QR:
The i th column of Q is $q_i = b_i / \|b_i\|_2$ where for CGS,

$$b_i = a_i - \sum_{j=1}^{i-1} \frac{\langle a_j, a_i \rangle}{\langle a_j, a_j \rangle} a_j,$$

while for MGS

$$b_i = a_i - \sum_{j=1}^{i-1} \langle q_j, a_i \rangle a_j.$$




$$\begin{aligned} \text{span}([a_1 \dots a_{i-1}]) \\ = \text{span}([q_1 \dots q_{i-1}]) \end{aligned}$$

- ▶ The cost of Gram-Schmidt is $2mn^2$ to leading order if $A \in \mathbb{R}^{m \times n}$.
inner product $\langle q_j, a_i \rangle$ $2m$ $\sum_{i=1}^n (n-i)4m$
adds and mult $i=1$ $= 2mn^2$

Error in MGS Orthogonalization


- ▶ MGS can be expressed in terms of projection matrices $P_i = I - q_i q_i^T$



A diagram showing a vector a_j and its orthogonal projection onto a line defined by vector q_i . The projection is labeled $q_i q_i^T a_j$. The resulting orthogonal component is labeled $a_j - q_i q_i^T a_j = a_j - q_i \langle q_i, a_j \rangle$.

$$P_i a_j = a_j - q_i q_i^T a_j = a_j - q_i \langle q_i, a_j \rangle$$

- ▶ The error in q_n due to a perturbation in a_n is amplified by $\prod_{i=1}^{n-1} \kappa(P_i)$



A diagram showing a vector q_n and a rectangular region representing a loss of orthogonality. The text explains that this loss is due to error in updating (applying each P_i).

loss of orthogonality due to error
in updating (applying each P_i)

Householder QR: Eliminating Error in MGS

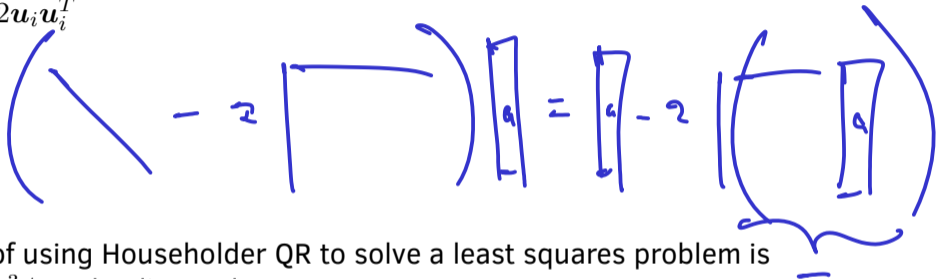
- ▶ Householder QR eliminates error amplification by using orthogonal Householder matrices (reflectors) rather than projection matrices

$$Q_i = I - 2u_i u_i^T$$

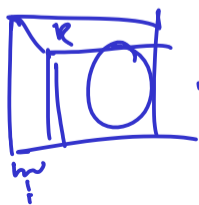
$$v = a \pm |a|e$$

$$a \lfloor 0 \rfloor > 0$$

then \pm



- ▶ The cost of using Householder QR to solve a least squares problem is $2mn^2 - 2n^3/3$ to leading order



$$\sum_{i=1}^n 2(m-i)(n-i) = 2mn^2 - 2n^3/3 \quad O(mn)$$

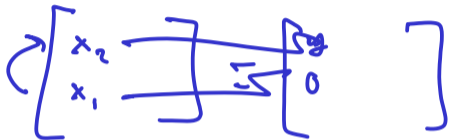
$$H_2 H_1 R = A$$

$$A \approx b$$

$$R x \approx \overbrace{H_1 H_2}^b b$$

Givens Rotations

- ▶ *Givens rotations* eliminate one element at a time $G = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \|z\| \\ 0 \end{bmatrix}$



- ▶ Givens rotations can be advantageous when working with *sparse* matrices



Solving Rank-Deficient Least Squares Problems

- ▶ The *pseudoinverse* is defined by $A^\dagger = V \Sigma^\dagger U^T$ where Σ^\dagger inverts only nonzero elements in Σ , it satisfies $AA^\dagger A = A$

$$A^\dagger = \underbrace{(A^T A)^{-1}}_{A \text{ is full rank}} A = \underbrace{\begin{bmatrix} \square & \begin{bmatrix} \uparrow & \uparrow & \uparrow \\ \square & \square & \square \end{bmatrix} & \square \end{bmatrix}}_{A \text{ arbitrary, full}} \text{ resp. nonzero d. equal entries}$$

- ▶ Given a least squares problem $Ax \cong b$, where A is rank-deficient, we can solve it via the pseudoinverse

$$x = A^\dagger b$$

QR with Column Pivoting

- ▶ An effective way to solve rank-deficient least squares problems without the SVD, is using QR with column pivoting $AP = QR$

A hand-drawn diagram illustrating the equation $AP = QR$. On the left, a vertical rectangle represents matrix A . To its right is a square bracket containing three dots, representing the permutation matrix P . A small dot is placed between A and P . To the right of this is another vertical rectangle representing matrix Q . To its right is a trapezoidal shape representing the upper triangular matrix R . A small dot is placed between Q and R .

- ▶ The rank-deficient least squares problem $Ax \cong b$ can be solved by QR with column pivoting

Aggregation of Transformations

- ▶ Householder transformations can be *aggregated* in the form $I - \mathbf{Y}\mathbf{T}\mathbf{Y}^T$ where \mathbf{Y} is lower-trapezoidal and \mathbf{T} is upper-triangular

- ▶ Given an arbitrary orthogonal matrix \mathbf{Q} , we can compute \mathbf{Y} via LU factorization of $I - \mathbf{Q}$