SI231b: Matrix Computations

Lecture 10: QR Factorization

Yue Qiu

qiuyue@shanghaitech.edu.cn

School of Information Science and Technology

ShanghaiTech University

Oct. 15, 2020

Recap: Householder Reflection

- **Problem:** given $\mathbf{x} \in \mathbb{R}^m$, find an orthogonal $\mathbf{H} \in \mathbb{R}^{m\times m}$ such that

$$\mathbf{H}\mathbf{x} = \begin{bmatrix} \beta \\ \mathbf{0} \end{bmatrix} = \beta \mathbf{e}_1, \qquad \text{for some } \beta \in \mathbb{R}.$$

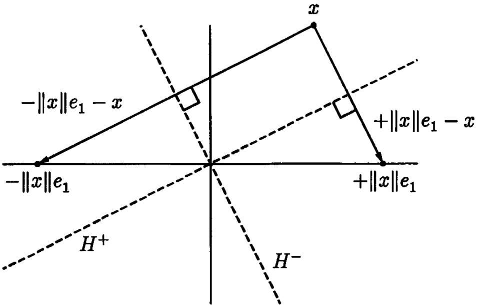

Figure 1: Householder reflection

Outline

- QR Factorization through Householder Reflection
- QR Factorization via Givens Rotation
- Solving Full-rank Least Squares

Householder Reflection

- Householder reflection: let $\mathbf{v} \in \mathbb{R}^m$, $\mathbf{v} \neq \mathbf{0}$. Let

$$\mathbf{H} = \mathbf{I} - \frac{2}{\|\mathbf{v}\|_2^2}\mathbf{v}\mathbf{v}^T,$$

which is a reflection matrix, $\mathbf{H}$ is an orthogonal matrix.

- it can be verified that (try)

$$\mathbf{v} = \mathbf{x} \mp \|\mathbf{x}\|_2\mathbf{e}_1 \quad \Longrightarrow \quad \mathbf{H}\mathbf{x} = \pm\|\mathbf{x}\|_2\mathbf{e}_1;$$

the sign above may be determined to be the one that maximizes $\|\mathbf{v}\|_2$, for the sake of numerical stability (why?)

  - $\mathbf{v} = \mathbf{x} - \|\mathbf{x}\|_2\mathbf{e}_1$ if $x_1 > 0$
  - $\mathbf{v} = \mathbf{x} + \|\mathbf{x}\|_2\mathbf{e}_1$ if $x_1 < 0$

Here, x_1 denotes the first entry of $\mathbf{x}$.

Householder QR

- let $\mathbf{H}_1 \in \mathbb{R}^{m\times m}$ be the Householder reflection w.r.t. $\mathbf{a}_1$. Transform $\mathbf{A}$ as

$$\mathbf{A}^{(1)} = \mathbf{H}_1\mathbf{A} = \begin{bmatrix} \times & \times & \dots & \times \\ 0 & {\color{red}\times} & \dots & \times \\ \vdots & \vdots & & \vdots \\ 0 & {\color{red}\times} & \dots & \times \end{bmatrix}$$

- let $\tilde{\mathbf{H}}_2 \in \mathbb{R}^{(m-1)\times(m-1)}$ be the Householder reflection w.r.t. $\mathbf{A}^{(1)}_{2:m,2}$ (marked red above). Transform $\mathbf{A}^{(1)}$ as

$$\mathbf{A}^{(2)} = \underbrace{\begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{H}}_2 \end{bmatrix}}_{=\mathbf{H}_2} \mathbf{A}^{(1)} = \begin{bmatrix} \times & \times & \dots \times \\ \mathbf{0} & \tilde{\mathbf{H}}_2\mathbf{A}^{(1)}_{2:m,2:n} \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \dots & \times \\ 0 & \times & \times & \dots & \times \\ \vdots & 0 & \times & \dots & \times \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \times & \dots & \times \end{bmatrix}$$

- by repeatedly applying the trick above, we can transform $\mathbf{A}$ as the desired $\mathbf{R}$

$\mathbf{A}^{(0)} = \mathbf{A}$

for $k = 1, \ldots, n$

$\mathbf{A}^{(k)} = \mathbf{H}_k \mathbf{A}^{(k-1)}$, where

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{I}_{k-1} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{H}}_k \end{bmatrix},$$

$\mathbf{I}_k$ is the $k \times k$ identity matrix; $\tilde{\mathbf{H}}_k$ is the Householder reflection of $\mathbf{A}^{(k-1)}_{k:m,k}$

end

- $\mathbf{H}_k$ introduces zeros under the diagonal of the k-th column
- the above procedure results in

$$\mathbf{A}^{(n)} = \mathbf{H}_n \mathbf{H}_{n-1} \cdots \mathbf{H}_2 \mathbf{H}_1 \mathbf{A}, \quad \mathbf{A}^{(n)} \text{ taking an upper triangular form}$$

- by letting $\mathbf{R} = \mathbf{A}^{(n)}$, $\mathbf{Q} = (\mathbf{H}_n \mathbf{H}_{n-1} \cdots \mathbf{H}_2 \mathbf{H}_1)^T$, we obtain the full QR
- a popularly used method for QR decomposition

Applying the Householder Matrix: $\mathbf{HA}$

$$\mathbf{HA} = (\mathbf{I} - \beta \mathbf{v}\mathbf{v}^T)\mathbf{A} = \mathbf{A} - (\beta \mathbf{v})(\mathbf{v}^T \mathbf{A})$$

- takes $\mathcal{O}(4mn)$ flops, rather than $\mathcal{O}(m^2 n)$
- only acts on a submatrix of $\mathbf{A}$ as the process goes
- takes $\mathcal{O}(2mn^2 - \frac{2}{3}n^3)$ flops to obtain $\mathbf{R}$ $(m > n)$. What for $m < n$?

Computations of Q

Recall $\mathbf{Q} = (\mathbf{H}_n \cdots \mathbf{H}_2 \mathbf{H}_1)^T = \mathbf{H}_1 \mathbf{H}_2 \cdots \mathbf{H}_n$, with $\mathbf{H}_k = \mathbf{I} - \beta_k \mathbf{v}^{(k)} (\mathbf{v}^{(k)})^T$ and

$$\mathbf{v}^{(k)} = \begin{bmatrix} 0 & \cdots & 0 & v_k^{(k)} & v_{k+1}^{(k)} & \cdots & v_m^{(k)} \end{bmatrix}^T$$

By letting $\mathbf{Q}_{n+1} = \mathbf{I}$, and executing $\mathbf{Q}_k = \mathbf{H}_k \mathbf{Q}_{k+1}$ for $k = n : -1 : 1$, we obtain $\mathbf{Q} = \mathbf{Q}_1$

- efficiently computations by applying Householder matrix
- takes $\mathcal{O}(4mn^2 - 2n^3)$ flops $(m > n)$, what for $m < n$?

Rotation Matrix

- Example: Let

$$\mathbf{J} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}$$

where $c = \cos(\theta), s = \sin(\theta)$ for some θ. Consider $\mathbf{y} = \mathbf{J}\mathbf{x}$:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} cx_1 + sx_2 \\ -sx_1 + cx_2 \end{bmatrix}.$$

It can be verified that

- $\mathbf{J}$ is orthogonal;
- $y_2 = 0$ if $\theta = \arctan(x_2/x_1)$, or if

$$c = \frac{x_1}{\sqrt{x_1^2 + x_2^2}}, \quad s = \frac{x_2}{\sqrt{x_1^2 + x_2^2}}.$$

- Givens rotations:

$$
\mathbf{J}(i,k,\theta) = \begin{array}{c c} & \begin{array}{cccc} & i & & k \end{array} \\ \begin{array}{c} \\ i \\ \\ k \\ \\ \end{array} & \begin{bmatrix} \mathbf{I} & & & & \\ & c & & s & \\ & & \mathbf{I} & & \\ & -s & & c & \\ & & & & \mathbf{I} \end{bmatrix} \end{array}
$$

where $c = \cos(\theta)$, $s = \sin(\theta)$.

- $\mathbf{J}(i,k,\theta)$ is orthogonal
- let $\mathbf{y} = \mathbf{J}(i,k,\theta)\mathbf{x}$. It holds that

$$
y_j = \begin{cases} cx_i + sx_k, & j = i \\ -sx_i + cx_k, & j = k \\ x_j, & j \neq i, k \end{cases}
$$

- y_k is forced to zero if we choose $\theta = \tan^{-1}(x_k/x_i)$.

- Example: consider a 4×3 matrix.

$$\mathbf{A} = \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} \xrightarrow{\mathbf{J}_{1,2}} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} \xrightarrow{\mathbf{J}_{1,3}} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \\ \times & \times & \times \end{bmatrix} \xrightarrow{\mathbf{J}_{1,4}} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \end{bmatrix} \xrightarrow{\mathbf{J}_{2,3}}$$

$$\begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \\ 0 & \times & \times \end{bmatrix} \xrightarrow{\mathbf{J}_{2,4}} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \\ 0 & 0 & \times \end{bmatrix} \xrightarrow{\mathbf{J}_{3,4}} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \\ 0 & 0 & 0 \end{bmatrix} = \mathbf{R}$$

where $\mathbf{B} \xrightarrow{\mathbf{J}} \mathbf{C}$ means $\mathbf{B} = \mathbf{J}\mathbf{C}$; $\mathbf{J}_{i,k} = \mathbf{J}(i, k, \theta)$, with θ chosen to zero out the (i, k)th entry of the matrix transformed by $\mathbf{J}_{i,k}$.

Givens QR

- **Givens QR:** assume $m \geq n$. Perform a sequence of Givens rotations to annihilate the lower triangular parts of $\mathbf{A}$ to obtain

$$\underbrace{(\mathbf{J}_{m,n} \dots \mathbf{J}_{n+2,n}\mathbf{J}_{n+1,n}) \dots (\mathbf{J}_{2m} \dots \mathbf{J}_{24}\mathbf{J}_{23})(\mathbf{J}_{1m} \dots \mathbf{J}_{13}\mathbf{J}_{12})}_{\mathbf{Q}^T} \mathbf{A} = \mathbf{R}$$

  where $\mathbf{R}$ takes the upper triangular form, and $\mathbf{Q}$ is orthogonal.

- applying Givens rotations $\mathbf{J}_{i,k}\mathbf{A}$ only updates the i, k row of $\mathbf{A}$, i.e.,

$$\mathbf{A}([i,j],:) = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \mathbf{A}([i,j],:)$$

- takes $\mathcal{O}(3mn^2 - n^3)$ flops to get $\mathbf{R}$, what for $\mathbf{Q}$?
- can be faster than Householder QR if $\mathbf{A}$ has certain sparse structures and we exploit them

$$\mathbf{x}_{LS} = \arg\min \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2$$

Using orthogonal projection

- solving $\mathbf{A}\mathbf{x} = \mathbf{P}\mathbf{b}$ to obtain $\mathbf{x}_{LS}$
  - $\mathbf{A}$ has orthonormal basis $\{\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_n\}$ (can be computed using QR factorization),
    $$\mathbf{x}_{LS} = \mathbf{R}^{-1}\mathbf{Q}^T\mathbf{b} \quad \text{(reduced QR)}$$
  - using $\mathbf{P} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$,
    $$(\mathbf{A}^T\mathbf{A})\mathbf{x}_{LS} = \mathbf{A}^T\mathbf{b} \quad \text{(normal equation)}$$

Using optimality condition

$$f(\mathbf{x}) = \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2$$

$$\nabla f(\mathbf{x}) = 0 \Longrightarrow \mathbf{x}_{LS} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b},$$

Rank-deficient LS, cf. [Golub-van Loan 13]

In the real field $\mathbb{R}$

For $\mathbf{A} \in \mathbb{R}^{m\times n}$, the pseudoinverse of $\mathbf{A}$ denoted by $\mathbf{A}^{+} \in \mathbb{R}^{n\times m}$ satisfying the Moore–Penrose conditions[1]

1. $\mathbf{A}\mathbf{A}^{\dagger}\mathbf{A} = \mathbf{A}$
2. $\mathbf{A}^{\dagger}\mathbf{A}\mathbf{A}^{\dagger} = \mathbf{A}^{\dagger}$
3. $(\mathbf{A}\mathbf{A}^{\dagger})^{T} = \mathbf{A}\mathbf{A}^{\dagger}$
4. $(\mathbf{A}^{\dagger}\mathbf{A})^{T} = \mathbf{A}^{\dagger}\mathbf{A}$

When $\mathbf{A}$ has full rank and $m > n$

- $\mathbf{A}^{\dagger} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T}$
  - In terms of reduced QR factorization of $\mathbf{A}$

$$\mathbf{A}^{\dagger} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T} = \mathbf{R}^{-1}\mathbf{Q}^{T}$$

[1] R. Penrose, A Generalized Inverse for Matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51(3), 1955