SI231b: Matrix Computations

Lecture 12: Computations of QR Factorization

Yue Qiu

qiuyue@shanghaitech.edu.cn

School of Information Science and Technology

ShanghaiTech University

Oct. 26, 2021

One of the Top 10 Algorithms in the 20th Century[1]

Given a rectangular matrix $A \in \mathbb{R}^{m \times n}$, A can be factorized into the form

$$A = QR$$

where

- $Q \in \mathbb{R}^{m \times m}$ is an orthogonal matrix
- $R \in \mathbb{R}^{m \times n}$ is upper-triangular

Reduced QR Factorization

For $m > n$, the reduced QR factorization given by

- $Q \in \mathbb{R}^{m \times n}$ has orthonormal columns
- $R \in \mathbb{R}^{n \times n}$ is upper-triangular
- also called 'economic' QR factorization in some cases

[1] https://doi.ieeecomputersociety.org/10.1109/MCISE.2000.814652

Reflection Matrices

- a matrix $\mathbf{H} \in \mathbb{R}^{m \times m}$ is called a reflection matrix if

$$\mathbf{H} = \mathbf{I} - 2\mathbf{P},$$

where $\mathbf{P}$ is an orthogonal projector.

- interpretation: denote $\mathbf{P}^{\perp} = \mathbf{I} - \mathbf{P}$, and observe

$$\mathbf{x} = \mathbf{P}\mathbf{x} + \mathbf{P}^{\perp}\mathbf{x}, \qquad \mathbf{H}\mathbf{x} = -\mathbf{P}\mathbf{x} + \mathbf{P}^{\perp}\mathbf{x}.$$

The vector $\mathbf{H}\mathbf{x}$ is a reflected version of $\mathbf{x}$, with $\mathcal{R}(\mathbf{P}^{\perp})$ being the "mirror"

- a reflection matrix is orthogonal:

$$\mathbf{H}^T\mathbf{H} = (\mathbf{I} - 2\mathbf{P})(\mathbf{I} - 2\mathbf{P}) = \mathbf{I} - 4\mathbf{P} + 4\mathbf{P}^2 = \mathbf{I} - 4\mathbf{P} + 4\mathbf{P} = \mathbf{I}$$

Householder Reflection

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

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

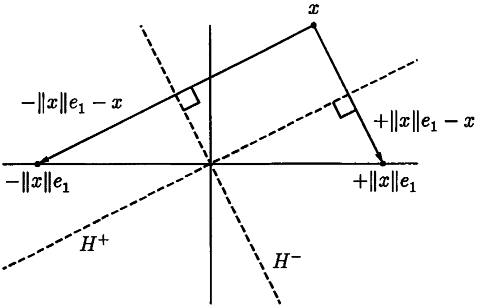

Figure 1: Householder reflection

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

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

which is a reflection matrix with $\mathbf{P} = \mathbf{v}\mathbf{v}^T/\|\mathbf{v}\|_2^2$

- 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 $\mathbf{x}$.

- let $H_1 \in \mathbb{R}^{m\times m}$ be the Householder reflection w.r.t. a_1. Transform A as

$$A^{(1)} = H_1 A = \begin{bmatrix} \times & \times & \dots & \times \\ 0 & \color{red}{\times} & \dots & \times \\ \vdots & \vdots & & \vdots \\ 0 & \color{red}{\times} & \dots & \times \end{bmatrix}$$

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

$$A^{(2)} = \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & \tilde{H}_2 \end{bmatrix}}_{=H_2} A^{(1)} = \begin{bmatrix} \times & \times & \dots \times \\ 0 & \tilde{H}_2 A^{(1)}_{2:m,2:n} \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \dots & \times \\ 0 & \times & \times & \dots & \times \\ \vdots & 0 & \times & \dots & \times \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \times & \dots & \times \end{bmatrix}$$

- by repeatedly applying the trick above, we can transform A as the desired R

$A^{(0)} = A$

for $k = 1, \ldots, n-1$

$A^{(k)} = H_k A^{(k-1)}$, where

$$H_k = \begin{bmatrix} I_{k-1} & 0 \\ 0 & \tilde{H}_k \end{bmatrix},$$

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

end

- H_k introduces zeros under the diagonal of the k-th column
- the above procedure results in

$$A^{(n)} = H_n \cdots H_2 H_1 A, \quad A^{(n)} \text{ taking an upper triangular form}$$

- by letting $R = A^{(n)}$, $Q = (H_n \cdots H_2 H_1)^T$, we obtain the full QR
- a popularly used method for QR decomposition

Applying the Householder Matrix: HA

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

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

Computations of Q

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

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

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

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

Rotation Matrix

- Example: Let

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

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

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

It can be verified that

  - 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:

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

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

- $\mathrm{J}(i,k,\theta)$ is orthogonal
- let $\mathrm{y} = \mathrm{J}(i,k,\theta)\mathrm{x}$. 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.

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

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

where $B \xrightarrow{J} C$ means $B = JC$; $J_{i,k} = J(i, k, \theta)$, with θ chosen to zero out the kth entry in the ith column vector of the matrix transformed by $J_{i,k}$.

Givens QR

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

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

  where R takes the upper triangular form, and Q is orthogonal.

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

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

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

$$x_{LS} = \arg\min \|b - Ax\|_2^2$$

Using orthogonal projection

- solving $Ax = Pb$ to obtain x_{LS}
  - A has orthonormal basis $\{q_1, q_2, \cdots, q_n\}$ (can be computed using QR factorization),
  $$x_{LS} = R^{-1}Q^T b \quad \text{(reduced QR)}$$
  - using $P = A(A^T A)^{-1}A^T$,
  $$(A^T A)x_{LS} = A^T b \quad \text{(normal equation)}$$

Using optimality condition

$$f(x) = \|b - Ax\|_2^2$$

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

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

In the real field $\mathbb{R}$

For $A \in \mathbb{R}^{m\times n}$, the pseudoinverse of A denoted by $A^{+} \in \mathbb{R}^{n\times m}$ satisfying the Moore–Penrose conditions[2]

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

When A has full rank and $m > n$

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

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

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

Readings

You are supposed to read

- Lloyd N. Trefethen and David Bau III. *Numerical Linear Algebra*, SIAM, 1997.

  Lecture 6, 8, 11

- Gene H. Golub and Charles F. Van Loan. *Matrix Computations*, Johns Hopkins University Press, 2013.

  Chapter 5.1 – 5.3