ENGG5781 Matrix Analysis and Computations
Lecture 8: QR Decomposition

Wing-Kin (Ken) Ma

2022-23 First Term

Department of Electronic Engineering
The Chinese University of Hong Kong

Lecture 8: QR Decomposition

- QR decomposition
- Gram-Schmidt QR
- Householder QR
- Givens QR
- the QR algorithm for computing eigenvalues

Summary

QR decomposition: Any $\mathbf{A} \in \mathbb{R}^{m \times n}$ admits a decomposition

$$\mathbf{A} = \mathbf{QR},$$

where $\mathbf{Q} \in \mathbb{R}^{m \times m}$ is orthogonal, $\mathbf{R} \in \mathbb{R}^{m \times n}$ takes an upper triangular form.

- efficient to compute
  - done algorithmically by either Gram-Schmidt, Householder reflections, or Givens rotations
- can be used to compute
  - a basis for $\mathcal{R}(\mathbf{A})$ or for $\mathcal{R}(\mathbf{A})^\perp$;
  - LS solutions.
- a building block for the QR algorithm—a popular numerical method for solving the eigenvalue problem (all eigenvalues)

Thin QR for Tall or Square $\mathbf{A}$

- suppose that QR decomposition exists; we will prove that later

- for $m \geq n$,

$$\mathbf{A} = \mathbf{QR} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1 \mathbf{R}_1,$$

  where $\mathbf{Q}_1 \in \mathbb{R}^{m \times n}$, $\mathbf{Q}_2 \in \mathbb{R}^{m \times (m-n)}$, $\mathbf{R}_1 \in \mathbb{R}^{n \times n}$ which is upper triangular

- the decomposition $\mathbf{A} = \mathbf{Q}_1 \mathbf{R}_1$ is called the thin QR decomposition of $\mathbf{A}$; $(\mathbf{Q}_1, \mathbf{R}_1)$ is called a thin QR factor of $\mathbf{A}$

- properties under thin QR and $m \geq n$:
  - $\mathbf{A}$ has full column rank if and only if $r_{ii} \neq 0$ for all i;
  - if $\mathbf{A}$ has full column rank,

$$\mathcal{R}(\mathbf{A}) = \mathcal{R}(\mathbf{Q}_1), \qquad \mathcal{R}(\mathbf{A})^{\perp} = \mathcal{R}(\mathbf{Q}_2)$$

LS via QR

- Problem: compute the solution to

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2,$$

  with $\mathbf{A}$ being of full column rank

- observe

$$\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 = \|\mathbf{Q}^T\mathbf{y} - \mathbf{R}\mathbf{x}\|_2^2 = \left\| \begin{bmatrix} \mathbf{Q}_1^T\mathbf{y} \\ \mathbf{Q}_2^T\mathbf{y} \end{bmatrix} - \begin{bmatrix} \mathbf{R}_1\mathbf{x} \\ \mathbf{0} \end{bmatrix} \right\|_2^2$$
$$= \|\mathbf{Q}_1^T\mathbf{y} - \mathbf{R}_1\mathbf{x}\|_2^2 + \|\mathbf{Q}_2^T\mathbf{y}\|_2^2$$

- Solution (computational): compute the thin QR factor $(\mathbf{Q}_1, \mathbf{R}_1)$ of $\mathbf{A}$; then solve

$$\mathbf{R}_1\mathbf{x} = \mathbf{Q}_1^T\mathbf{y}$$

  via backward substitution.

Existence of QR Decomposition for Full Column-Rank Matrices

Theorem 8.1. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a full column-rank matrix. Then $\mathbf{A}$ admits a decomposition

$$\mathbf{A} = \mathbf{Q}_1 \mathbf{R}_1,$$

where $\mathbf{Q}_1 \in \mathbb{R}^{m \times n}$ is semi-orthogonal; $\mathbf{R}_1 \in \mathbb{R}^{n \times n}$ is upper triangular. If we restrict $r_{ii} > 0$ for all i, then $(\mathbf{Q}_1, \mathbf{R}_1)$ is unique.

- Proof:
  1. let $\mathbf{C} = \mathbf{A}^T \mathbf{A}$, which is PD if $\mathbf{A}$ has full column rank
  2. since $\mathbf{C}$ is PD, it admits the Cholesky decomposition $\mathbf{C} = \mathbf{R}_1^T \mathbf{R}_1$
  3. $\mathbf{R}_1$, as the Cholesky factor, is unique (cf. Theorem 7.3)
  4. let $\mathbf{Q}_1 = \mathbf{A}\mathbf{R}_1^{-1}$. It can be verified that $\mathbf{Q}_1^T \mathbf{Q}_1 = \mathbf{I}, \mathbf{Q}_1 \mathbf{R}_1 = \mathbf{A}$

- Remark: the proof above reveals that thin QR may be computed via Cholesky decomposition, but this is not what we usually do in practice

Gram-Schmidt for Thin QR

Recall the Gram-Schmidt procedure in Lecture 1:

Algorithm: Gram-Schmidt
input: a collection of linearly independent vectors $\mathbf{a}_1, \ldots, \mathbf{a}_n$
$\tilde{\mathbf{q}}_1 = \mathbf{a}_1$, $\mathbf{q}_1 = \tilde{\mathbf{q}}_1 / \|\tilde{\mathbf{q}}_1\|_2$
for $i = 2, \ldots, n$
    $\tilde{\mathbf{q}}_i = \mathbf{a}_i - \sum_{j=1}^{i-1} (\mathbf{q}_j^T \mathbf{a}_i) \mathbf{q}_j$
    $\mathbf{q}_i = \tilde{\mathbf{q}}_i / \|\tilde{\mathbf{q}}_i\|_2$
end
output: $\mathbf{q}_1, \ldots, \mathbf{q}_n$

- let $r_{ii} = \|\tilde{\mathbf{q}}_i\|_2$, $r_{ij} = \mathbf{q}_j^T \mathbf{a}_i$ for $j = 1, \ldots, i-1$
- we see that $\mathbf{a}_i = \sum_{j=1}^{i} r_{ij} \mathbf{q}_i$ for all i, or, equivalently,

$$\mathbf{A} = \mathbf{Q}_1 \mathbf{R}_1,$$

where $\mathbf{Q}_1 = [\ \mathbf{q}_1, \ldots, \mathbf{q}_n\ ]$; $\mathbf{R}_1$ is upper triangular with $[\mathbf{R}_1]_{ij} = r_{ij}$ for $j \leq i$

Gram-Schmidt for Thin QR

- complexity of Gram-Schmidt: $\mathcal{O}(mn^2)$
- there are several variants with the Gram-Schmidt procedure, and they were usually proposed for improving numerical stability
  - say, what if $\tilde{\mathbf{q}}_i$ is close to $\mathbf{0}$?
- Gram-Schmidt tells us how we may compute the thin QR, but not the full QR

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 Reflections

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

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

- 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 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

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

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

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

Householder QR

- assume $m \geq n$, without loss of generality (why?)

$\mathbf{A}^{(0)} = \mathbf{A}$
for $k = 1, \ldots, n-1$
    $\mathbf{A}^{(k)} = \mathbf{H}_k \mathbf{A}^{(k-1)}$, 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

- the above procedure results in

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

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

Householder QR

$\mathbf{A}^{(0)} = \mathbf{A}$
for $k = 1, \ldots, n-1$
$\mathbf{A}^{(k)} = \mathbf{H}_k \mathbf{A}^{(k-1)}$, 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

- the complexity (for $m \geq n$):
  - $\mathcal{O}(n^2(m - n/3))$ for $\mathbf{R}$ only
    - $*$ a direct implementation of the above Householder pseudo-code does not lead us to this complexity; the structures of $\mathbf{H}_k$ are exploited in the implementations to lead to this complexity
  - $\mathcal{O}(m^2 n - mn^2 + n^3/3)$ if $\mathbf{Q}$ is also wanted

Givens Rotations

- 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 $\theta = \tan^{-1}(x_2/x_1)$, 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

- Givens rotations:

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

  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 $\theta = \tan^{-1}(x_k/x_i)$.

Givens QR

- 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.

- complexity (for $m \geq n$): $\mathcal{O}(n^2(m - n/3))$ for $\mathbf{R}$ only

- not as efficient as Householder QR for general (and dense) $\mathbf{A}$'s
  - the flop count for Householder QR is $2n^2(m - n/3)$ (for $\mathbf{R}$ and for $m \geq n$)
  - the flop count for Givens QR is $3n^2(m - n/3)$

- can be faster than Householder QR if $\mathbf{A}$ has certain sparse structures and we exploit them

The QR Algorithm for Computing Eigenvalues

input: $\mathbf{A} \in \mathbb{C}^{n \times n}$
$\mathbf{A}^{(0)} = \mathbf{A}$
for $k = 1, 2, \ldots$
    $\mathbf{Q}^{(k)}\mathbf{R}^{(k)} = \mathbf{A}^{(k-1)}$ (perform QR decomposition)
    $\mathbf{A}^{(k)} = \mathbf{R}^{(k)}\mathbf{Q}^{(k)}$
end

- denote the Schur decomposition of $\mathbf{A}$ by $\mathbf{A} = \mathbf{U}\mathbf{T}\mathbf{U}^H$
- under some mild assumptions, $\mathbf{A}^{(k)}$ converges to $\mathbf{T}$
  - if our problem is to compute all the eigenvalues of $\mathbf{A}$, picking the diagonal elements of $\mathbf{A}^{(k)}$ for a sufficiently large k would do
  - no simple way to understand why it works...
- the most popular method for computing all eigenvalues of a general $\mathbf{A}$
  - the practical QR algorithm used in modern software is more sophisticated, although the main idea is the same as that of the above basic QR algorithm