

The QR decomposition of a matrix

Matrices > Basics | Matrix products | Special matrices | QR | Matrix inverses | Linear maps | Matrix norms | Applications

- Basic idea
- Case when the matrix has linearly independent columns
- General case
- Full QR decomposition

Basic idea

The basic goal of the QR decomposition is to *factor* a matrix as a product of two matrices (traditionally called Q, R , hence the name of this factorization). Each matrix has a simple structure which can be further exploited in dealing with, say, [linear equations](#).

The QR decomposition is nothing else than the [Gram-Schmidt procedure](#) applied to the columns of the matrix, and with the result expressed in matrix form. Consider a $m \times n$ matrix $A = (a_1, \dots, a_n)$, with each $a_i \in \mathbf{R}^m$ a column of A .

Case when A is full column rank

Assume first that the a_i 's (the columns of A) are linearly independent. Each step of the G-S procedure can be written as

$$a_i = (a_i^T q_1)q_1 + \dots + (a_i^T q_{i-1})q_{i-1} + \|\tilde{q}_i\|_2 q_i, \quad i = 1, \dots, n.$$

We write this as

$$a_i = r_{i1}q_1 + \dots + r_{i,i-1}q_{i-1} + r_{ii}q_i, \quad i = 1, \dots, n,$$

where $r_{ij} = (a_i^T q_j)$ ($1 \leq j \leq i-1$) and $r_{ii} = \|\tilde{q}_i\|_2$.

Since the q_i 's are unit-length and normalized, the matrix $Q = (q_1, \dots, q_n)$ satisfies $Q^T Q = I_n$. The QR decomposition of a $m \times n$ matrix A thus allows to write the matrix in *factored* form:

$$A = QR, \quad Q = \begin{pmatrix} q_1 & \dots & q_n \end{pmatrix}, \quad R = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ 0 & r_{22} & & r_{2n} \\ \vdots & & \ddots & \vdots \\ 0 & & 0 & r_{nn} \end{pmatrix}$$

where Q is a $m \times n$ matrix with $Q^T Q = I_n$, and R is $n \times n$, upper-triangular.

Matlab syntax

```
>> [Q,R] = qr(A,0); % A is a mxn matrix, Q is mxn orthogonal, R is nxn upper triangular
```

Example: QR decomposition of a 4x6 matrix.

Case when the columns are not independent

When the columns of A are not independent, at some step of the G-S procedure we encounter a zero vector \tilde{q}_j , which means a_j is a linear combination of a_{j-1}, \dots, a_1 . The **modified Gram-Schmidt procedure** then simply skips to the next vector and continues.

In matrix form, we obtain $A = QR$, with $Q \in \mathbf{R}^{m \times r}$, $r = \mathbf{Rank}(A)$, and R has an upper staircase form, for example:

$$R = \begin{pmatrix} * & * & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * \end{pmatrix}.$$

(This is simply an upper triangular matrix with some rows deleted. It is still upper triangular.)

We can permute the columns of R to bring forward the first non-zero elements in each row:

$$R = \begin{pmatrix} R_1 & R_2 \end{pmatrix} P^T, \quad \begin{pmatrix} R_1 & | & R_2 \end{pmatrix} := \left(\begin{array}{ccc|ccc} * & * & * & * & * & * \\ 0 & * & 0 & * & * & * \\ 0 & 0 & * & 0 & 0 & * \end{array} \right),$$

where P is a **permutation matrix** (that is, its columns are the unit vectors in some order), whose effect is to permute columns. (Since P is orthogonal, $P^{-1} = P^T$.) Now, R_1 is square, upper triangular, and *invertible*, since none of its diagonal elements is zero.

The QR decomposition can be written

$$AP = Q \begin{pmatrix} R_1 & R_2 \end{pmatrix},$$

where

1. $Q \in \mathbf{R}^{m \times r}$, $Q^T Q = I_r$;
2. r is the *rank* of A ;
3. R_1 is $r \times r$ upper triangular, invertible matrix;
4. R_2 is a $r \times (n - r)$ matrix;
5. P is a $m \times m$ permutation matrix.

Matlab syntax

```
>> [Q,R,inds] = qr(A,0); % here inds is a permutation vector such that A(:,inds) = Q*R
```

Full QR decomposition

The *full QR decomposition* allows to write $A = QR$ where $Q \in \mathbf{R}^{m \times m}$ is *square* and orthogonal ($Q^T Q = Q Q^T = I_m$). In other words, the columns of Q are an orthonormal basis for the whole output space \mathbf{R}^m , not just for the range of A .

We obtain the full decomposition by appending an $m \times m$ identity matrix to the columns of A : $A \rightarrow [A, I_m]$. The QR decomposition of the augmented matrix allows to write

$$AP = QR = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix} \begin{pmatrix} R_1 & R_2 \\ 0 & 0 \end{pmatrix},$$

where the columns of the $m \times m$ matrix $Q = [Q_1, Q_2]$ are orthogonal, and R_1 is upper triangular and invertible. (As before, P is a permutation matrix.) In the G-S procedure, the columns of Q_1 are obtained from those of A , while the columns of Q_2 come from the extra columns added to A .

The full QR decomposition reveals the rank of A : we simply look at the elements on the diagonal of R that are not zero, that is, the size of R_1 .

Matlab syntax

```
>> [Q,R] = qr(A); % A is a mxn matrix, Q is mxm orthogonal, R is mxn upper triangular
```

Example: [QR decomposition of a 4x6 matrix.](#)