qrinv( ) — Generalized inverse of matrix via QR decomposition

Description Syntax Remarks and examples Conformability Diagnostics Also see

Description

`qrinv(`*A*`, ...)` returns the inverse or generalized inverse of real or complex matrix A: $m \times n$, $m \geq n$. If optional argument *rank* is specified, the rank of A is returned there.

`_qrinv(`*A*`, ...)` does the same thing except that, rather than returning the result, it overwrites the original matrix A with the result. `_qrinv()` returns the rank of A.

In both cases, optional argument *tol* specifies the tolerance for determining singularity; see *Remarks and examples* below.

Syntax

numeric matrix	`qrinv(numeric matrix` *A*`)`
numeric matrix	`qrinv(numeric matrix` *A*`,` *rank*`)`
numeric matrix	`qrinv(numeric matrix` *A*`,` *rank*`,` *real scalar tol*`)`
real scalar	`_qrinv(numeric matrix` *A*`)`
real scalar	`_qrinv(numeric matrix` *A*`,` *real scalar tol*`)`

where the type of *rank* is irrelevant; the rank of A is returned there.

Remarks and examples

`qrinv()` and `_qrinv()` are most often used on square and possibly rank-deficient matrices but may be used on nonsquare matrices that have more rows than columns. Also see [M-5] **pinv( )** for an alternative. See [M-5] **luinv( )** for a more efficient way to obtain the inverse of full-rank, square matrices, and see [M-5] **invsym( )** for inversion of real, symmetric matrices.

When A is of full rank, the inverse calculated by `qrinv()` is essentially the same as that computed by the faster `luinv()`. When A is singular, `qrinv()` and `_qrinv()` compute a generalized inverse, $A*$, which satisfies

$$A(A*)A = A$$
$$(A*)A(A*) = A*$$

This generalized inverse is also calculated for nonsquare matrices that have more rows than columns and, then returned is a least-squares solution. If A is $m \times n$, $m \geq n$, and if the rank of A is equal to n, then $(A*)A = I$, ignoring roundoff error.

`qrinv(`*A*`)` is implemented as `qrsolve(`*A*`, I(rows(`*A*`)))`; see [M-5] **qrsolve( )** for details and for use of the optional *tol* argument.

Conformability

qrinv(*A*, *rank*, *tol*):

input:

A:	$m \times n$,	$m \geq n$
tol:	1×1	(optional)

output:

rank:	1×1	(optional)
result:	$n \times m$	

_qrinv(*A*, *tol*):

input:

A:	$m \times n$,	$m \geq n$
tol:	1×1	(optional)

output:

A:	$n \times m$	
result:	1×1	(containing rank)

Diagnostics

The inverse returned by these functions is real if *A* is real and is complex if *A* is complex.

qrinv(*A*, ...) and _qrinv(*A*, ...) return a result containing missing values if *A* contains missing values.

_qrinv(*A*, ...) aborts with error if *A* is a view.

See [M-5] **qrsolve()** and [M-1] **Tolerance** for information on the optional *tol* argument.

Also see

[M-5] **cholinv()** — Symmetric, positive-definite matrix inversion

[M-5] **_invmat()** — Inverse and pseudoinverse of a square matrix

[M-5] **invsym()** — Symmetric real matrix inversion

[M-5] **luinv()** — Square matrix inversion

[M-5] **pinv()** — Moore–Penrose pseudoinverse

[M-5] **qrsolve()** — Solve AX=B for X using QR decomposition

[M-5] **solve_tol()** — Tolerance used by solvers and inverters

[M-4] **Matrix** — Matrix functions

[M-4] **Solvers** — Functions to solve AX=B and to obtain A inverse