_invmat() — Inverse and pseudoinverse of a square matrix

Description Syntax Remarks and examples Conformability
Diagnostics References Also see

Description

`_invmat(`*A*`)` and `_invmat(`*A*`,` *tol*`)` overwrite the original real or complex, square matrix A with the inverse of A if A has full rank and with the Moore–Penrose pseudoinverse if not. The function returns a real scalar `0` if the inverse is computed and `1` if the pseudoinverse is computed.

The optional argument *tol* specifies the tolerance for determining singularity; see Remarks and examples below.

Syntax

real scalar `_invmat(`*numeric matrix A*`)`

real scalar `_invmat(`*numeric matrix A*`,` *real scalar tol*`)`

Remarks and examples

These routines calculate the inverse of A if A has full rank. The inverse matrix A^{-1} of A satisfies the conditions

$$AA^{-1} = I$$
$$A^{-1}A = I$$

A is required to be square.

However, if A is singular or close to singular, the Moore–Penrose pseudoinverse is computed instead. The Moore–Penrose pseudoinverse is also known as the Moore–Penrose inverse and as the generalized inverse.

The pseudoinverse $A*$ of A satisfies four conditions,

$$\begin{aligned} A(A*)A &= A \\ (A*)A(A*) &= A* \\ (AA*)' &= A(A*) \\ (A*A)' &= (A*)A \end{aligned}$$

where the transpose operator $'$ is understood to mean the conjugate transpose when A is complex. Also, if A is of full rank, then

$$A* = A^{-1}$$

See [M-5] **pinv()** for details about pseudoinverse.

▷ Example 1: Full-rank matrix

If A has full rank, the function returns 0 and computes the inverse. Here we compute the inverse and show that $AA^{-1} = I$.

```
: A
         1     2     3
    +-------------------+
  1 |    1     2     3  |
  2 |  2.5     5     1  |
  3 |    3     2     1  |
    +-------------------+

: rc = _invmat(Ainv = A)

: rc
  0

: Ainv
                  1                2                3
    +------------------------------------------------+
  1 |  -.1153846154    -.1538461538               .5  |
  2 |  -.0192307692     .3076923077             -.25  |
  3 |   .3846153846    -.1538461538                0  |
    +------------------------------------------------+

: Ainv * A
                  1                2                3
    +------------------------------------------------+
  1 |             1    -2.22045e-16     -1.11022e-16  |
  2 |   1.11022e-16               1      2.77556e-17  |
  3 |             0               0                1  |
    +------------------------------------------------+
```

You may verify other conditions for the inverse yourself.

◁

▷ Example 2: Singular matrix

If B does not have full rank, the function returns 1 and computes the pseudoinverse. Here we compute the pseudoinverse and show that $B(B*)B = B$ and $(B*)B(B*) = B*$.

```
: B
       1   2
    +---------+
  1 |  1   2  |
  2 |  0   0  |
    +---------+

: rc = _invmat(Binv = B)

: rc
  1

: Binv
        1    2
    +-----------+
  1 |  .2    0  |
  2 |  .4    0  |
    +-----------+
```

```
: mreldif(B * Binv * B, B), mreldif(Binv * B * Binv, Binv)
                  1              2
    +-----------------------------+
  1 |             0   3.96508e-17 |
    +-----------------------------+
```

You may verify other conditions for the pseudoinverse yourself.

◁

Conformability

`_invmat(`*A*`,` *tol*`)`:

input:

A:	$n \times n$	
tol:	1×1	(optional)

output:

A:	$n \times n$	
result:	1×1	

Diagnostics

The inverse returned by these functions is real if *A* is real and is complex if *A* is complex. The determination of singularity is made relative to *tol*. See *Tolerance* under *Remarks and examples* in [M-5] **lusolve( )** for details.

`_invmat(`*A*`)` returns a matrix containing missing if *A* contains missing values.

`_invmat(`*A*`)` aborts with error if *A* is a view.

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

References

Moore, E. H. 1920. On the reciprocal of the general algebraic matrix. *Bulletin of the American Mathematical Society* 26: 394–395.

Penrose, R. 1955. A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society* 51: 406–413. https://doi.org/10.1017/S0305004100030401.

Also see

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

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

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

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

[M-5] **qrinv( )** — Generalized inverse of matrix via QR decomposition

[M-5] **_solvemat( )** — Solve AX=B for X

[M-4] **Matrix** — Matrix functions

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