

Utility — Matrix utility functions

[Contents](#)[Description](#)[Remarks and examples](#)[Also see](#)**Contents**

[M-5] Manual entry	Function	Purpose
Complex		
Re()	Re() Im()	real part imaginary part
C()	C()	make complex
Shape & type		
rows()	rows() cols() length()	number of rows number of columns number of elements of vector
eltype()	eltype() orgtype() classname() structname()	element type of object organizational type of object class name of a Mata class scalar struct name of a Mata struct scalar
isreal()	isreal() iscomplex() isstring() ispointer()	object is real matrix object is complex matrix object is string matrix object is pointer matrix
isrealvalues()	isrealvalues()	whether matrix contains only real values
isview()	isview()	whether matrix is view

Properties

issymmetric()	issymmetric() issymmetriconly()	whether matrix is symmetric (Hermitian) whether matrix is mechanically symmetric
isdiagonal()	isdiagonal()	whether matrix is diagonal
diag0cnt()	diag0cnt()	count 0s on diagonal

Selection

select()	<code>select()</code> <code>st_select()</code> <code>selectindex()</code>	select rows or columns select rows or columns of view select indices
-----------------	---------------------------------------------------------------------------------	----------------------------------------------------------------------------

Missing values

missing()	<code>missing()</code> <code>rowmissing()</code> <code>colmissing()</code> <code>nonmissing()</code> <code>rownonmissing()</code> <code>colnonmissing()</code> <code>hasmissing()</code>	count of missing values count of missing values, by row count of missing values, by column count of nonmissing values count of nonmissing values, by row count of nonmissing values, by column whether matrix has missing values
missingof()	<code>missingof()</code>	appropriate missing value

Range, sums, & cross products

minmax()	<code>rowmin()</code> <code>colmin()</code> <code>min()</code> <code>rowmax()</code> <code>colmax()</code> <code>max()</code> <code>rowminmax()</code> <code>colminmax()</code> <code>minmax()</code> <code>rowmaxabs()</code> <code>colmaxabs()</code>	minimum, by row minimum, by column minimum, overall maximum, by row maximum, by column maximum, overall minimum and maximum, by row minimum and maximum, by column minimum and maximum, overall <code>rowmax(abs())</code> <code>colmax(abs())</code>
minindex()	<code>minindex()</code> <code>maxindex()</code>	indices of minimums indices of maximums
sum()	<code>rowsum()</code> <code>colsum()</code> <code>sum()</code> <code>quadrowsum()</code> <code>quadcolsum()</code> <code>quadsum()</code>	sum of each row sum of each column overall sum quad-precision sum of each row quad-precision sum of each column quad-precision overall sum

 Range, sums, & cross products, *continued*

runningsum()	runningsum() quadrunningsum()	running sum of vector quad-precision runningsum()
panelsum()	panelsum()	within-panel sum of each column
cross()	cross()	$X'X$, $X'Z$, etc.
crossdev()	crossdev()	$(X: -x)'(X: -x)$, $(X: -x)'(Z: -z)$, etc.
quadcross()	quadcross() quadcrossdev()	quad-precision cross() quad-precision crossdev()

 Programming

reldif()	reldif() mreldif() mreldifsym() mreldifre()	relative difference max. relative difference between matrices max. relative difference from symmetry max. relative difference from real
all()	all() any() allof() anyof()	$\text{sum}(!L)==0$ $\text{sum}(L)!=0$ $\text{all}(P==s)$ $\text{any}(P==s)$
panelsetup()	panelsetup() panelstats() panelsubmatrix() panelsubview()	initialize panel-data processing summary statistics on panels obtain matrix for panel i obtain view matrix for panel i
_negate()	_negate()	fast negation of matrix

 Constants & tolerances

mindouble()	mindouble() maxdouble() smallestdouble()	minimum nonmissing value maximum nonmissing value smallest $e > 0$
epsilon()	epsilon()	unit roundoff error
floatround()	floatround()	round to float precision
solve_tol()	solve_tol()	tolerance used by solvers and inverters

Description

Matrix utility functions tell you something about the matrix, such as the number of rows or whether it is diagonal.

Remarks and examples

[stata.com](https://www.stata.com)

There is a thin line between utility and manipulation; also see

[M-4] [Manipulation](#) Matrix manipulation functions

Also see

[M-4] [Intro](#) — Categorical guide to Mata functions