

Mathematical — Important mathematical functions

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

Contents

[M-5]

Manual entry	Function	Purpose
Basics (also see [M-4] Scalar)		
sum()	rowsum() colsum() sum() quadrowsum() quadcolsum() quadsum()	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
runningsum()	runningsum() quadruningsum()	running sum of vector quad-precision runningsum()
minmax()	rowmin() colmin() min() rowmax() colmax() max() rowminmax() colminmax() minmax() rowmaxabs() colmaxabs()	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 rowmax(abs()) colmax(abs())
deriv()	deriv() deriv_init() deriv_init_*(*) deriv() deriv_result_*(*) deriv_query()	numerical derivatives begin derivatives set details compute derivatives access results report settings
optimize()	optimize() optimize_init() optimize_init_*(*) optimize() optimize_result_*(*) optimize_query()	function maximization and minimization begin optimization set details perform optimization access results report settings

Basics, <i>continued</i>

moptimize()	<code>moptimize()</code> <code>moptimize_ado_cleanup()</code> <code>moptimize_evaluate()</code> <code>moptimize_init()</code> <code>moptimize_init_*</code> <code>moptimize_result_*</code> <code>moptimize_query()</code> <code>moptimize_util_*</code>	function optimization perform cleanup after ado evaluate function at initial values begin setup of optimization problem set details access <code>moptimize()</code> results report settings utility functions for writing evaluators and processing results
solvenl()	<code>solvenl_init()</code> <code>solvenl_init_*</code> <code>solvenl_solve()</code> <code>solvenl_result_*</code> <code>solvenl_dump()</code>	begin solver set details solve equations access results report detailed settings
LinearProgram()	<code>LinearProgram()</code>	linear programming
Quadrature()	<code>Quadrature()</code> <code>QuadratureVec()</code>	numerical integration vector of numerical integration

Fourier transform

fft()	<code>fft()</code> <code>invfft()</code> <code>convolve()</code> <code>deconvolve()</code> <code>Corr()</code> <code>ftperiodogram()</code> <code>ftpad()</code> <code>ftwrap()</code> <code>ftunwrap()</code> <code>ftretime()</code> <code>ftfreqs()</code>	fast Fourier transform inverse fast Fourier transform convolution inverse of <code>convolve()</code> correlation power spectrum pad to power-of-2 length convert to frequency-wraparound order convert from frequency-wraparound order change time scale of signal frequencies of transform
--------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Cubic splines

spline3()	<code>spline3()</code> <code>spline3eval()</code>	fit cubic spline evaluate cubic spline
------------------	------------------------------------------------------	-------------------------------------------

Polynomials

polyeval()	<code>polyeval()</code>	evaluate polynomial
	<code>polysolve()</code>	solve for polynomial
	<code>polytrim()</code>	trim polynomial
	<code>polyderiv()</code>	derivative of polynomial
	<code>polyinteg()</code>	integral of polynomial
	<code>polyadd()</code>	add polynomials
	<code>polymult()</code>	multiply polynomials
	<code>polydiv()</code>	divide polynomials
	<code>polyroots()</code>	find roots of polynomial

Number-theoretic point sets

halton()	<code>halton()</code>	generate a Halton or Hammersley set
	<code>ghalton()</code>	generate a generalized Halton sequence

Base conversion

inbase()	<code>inbase()</code>	convert to specified base
	<code>frombase()</code>	convert from specified base

Description

The above functions are important mathematical functions that most people would not call either matrix functions or scalar functions, but that use matrices and scalars.

Remarks and examples

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

For other mathematical functions, see

[M-4] Matrix	Matrix mathematical functions
[M-4] Scalar	Scalar mathematical functions
[M-4] Statistical	Statistical functions

Also see

[M-4] **Intro** — Categorical guide to Mata functions

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.

