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()`	sum of each row
	`colsum()`	sum of each column
	`sum()`	overall sum
	`quadrowsum()`	quad-precision sum of each row
	`quadcolsum()`	quad-precision sum of each column
	`quadsum()`	quad-precision overall sum
runningsum()	`runningsum()`	running sum of vector
	`quadrunningsum()`	quad-precision `runningsum()`
minmax()	`rowmin()`	minimum, by row
	`colmin()`	minimum, by column
	`min()`	minimum, overall
	`rowmax()`	maximum, by row
	`colmax()`	maximum, by column
	`max()`	maximum, overall
	`rowminmax()`	minimum and maximum, by row
	`colminmax()`	minimum and maximum, by column
	`minmax()`	minimum and maximum, overall
	`rowmaxabs()`	`rowmax(abs())`
	`colmaxabs()`	`colmax(abs())`
deriv()	`deriv()`	numerical derivatives
	`deriv_init()`	begin derivatives
	`deriv_init_*()`	set details
	`deriv()`	compute derivatives
	`deriv_result_*()`	access results
	`deriv_query()`	report settings
optimize()	`optimize()`	function maximization and minimization
	`optimize_init()`	begin optimization
	`optimize_init_*()`	set details
	`optimize()`	perform optimization
	`optimize_result_*()`	access results
	`optimize_query()`	report settings

Basics, *continued*

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

Fourier transform

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

Cubic splines

spline3( )	`spline3()`	fit cubic spline
	`spline3eval()`	evaluate cubic spline

Polynomials

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

Number-theoretic point sets

halton( )	`halton()`	generate a Halton or Hammersley set
	`ghalton()`	generate a generalized Halton sequence

Base conversion

inbase( )	`inbase()`	convert to specified base
	`frombase()`	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

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