

# Table of Contents

## [M-0] Introduction to the Mata manual

intro .....	Introduction to the Mata manual	3
-------------	---------------------------------	---

## [M-1] Introduction and advice

intro .....	Introduction and advice	7
ado .....	Using Mata with ado-files	8
first .....	Introduction and first session	15
help .....	Obtaining online help	30
how .....	How Mata works	31
interactive .....	Using Mata interactively	35
LAPACK .....	The LAPACK linear-algebra routines	43
limits .....	Limits and memory utilization	44
naming .....	Advice on naming functions and variables	46
permutation .....	An aside on permutation matrices and vectors	49
returnedargs .....	Function arguments used to return results	55
source .....	Viewing the source code	58
tolerance .....	Use and specification of tolerances	59

## [M-2] Language definition

intro .....	Language definition	65
break .....	Break out of for, while, or do loop	67
continue .....	Continue with next iteration of for, while, or do loop	68
declarations .....	Declarations and types	70
do .....	do ... while (exp)	79
errors .....	Error codes	80
exp .....	Expressions	85
for .....	for (exp1; exp2; exp3) stmt	91
ftof .....	Passing functions to functions	93
goto .....	goto label	95
if .....	if (exp) ... else ...	96
op_arith .....	Arithmetic operators	98
op_assignment .....	Assignment operator	100
op_colon .....	Colon operators	105
op_conditional .....	Conditional operator	108
op_increment .....	Increment and decrement operators	110
op_join .....	Row- and column-join operators	113
op_kronecker .....	Kronecker direct-product operator	116
op_logical .....	Logical operators	117
op_range .....	Range operators	120
op_transpose .....	Conjugate transpose operator	122
optargs .....	Optional arguments	125
pointers .....	Pointers	130
pragma .....	Suppressing warning messages	140
reswords .....	Reserved words	143
return .....	return and return(exp)	145

semicolons	Use of semicolons	147
subscripts	Use of subscripts	151
syntax	Mata language grammar and syntax	158
version	Version control	166
void	Void matrices	169
while	while (exp) stmt	171

### [M-3] Commands for controlling Mata

intro	Commands for controlling Mata	175
end	Exit Mata and return to Stata	177
mata	Mata invocation command	178
mata clear	Clear Mata's memory	182
mata describe	Describe contents of Mata's memory	183
mata drop	Drop matrix or function	185
mata help	Obtain online help	186
mata matsave	Save and restore matrices	187
mata memory	Report on Mata's memory usage	189
mata mlib	Create function library	190
mata mosave	Save function's compiled code in object file	196
mata rename	Rename matrix or function	198
mata set	Set and display Mata system parameters	199
mata stata	Execute Stata command	202
mata which	Identify function	203
namelists	Specifying matrix and function names	204

### [M-4] Index and guide to functions

intro	Index and guide to functions	209
io	I/O functions	211
manipulation	Matrix manipulation	214
mathematical	Important mathematical functions	216
matrix	Matrix functions	218
programming	Programming functions	221
scalar	Scalar mathematical functions	223
solvers	Functions to solve $AX=B$ and to obtain $A$ inverse	225
standard	Functions to create standard matrices	227
stata	Stata interface functions	229
statistical	Statistical functions	232
string	String manipulation functions	235
utility	Matrix utility functions	237

### [M-5] Mata functions

intro	Mata functions	243
abs()	Absolute value (length)	244
all()	Element comparisons	245
args()	Number of arguments	247
ascii()	Manipulate ASCII codes	248
assert()	Abort execution if false	249
blockdiag()	Block-diagonal matrix	251

C()	Make complex	252
c()	Access c() value	254
callersversion()	Obtain version number of caller	255
cat()	Load file into string matrix	256
chdir()	Manipulate directories	257
cholesky()	Cholesky square-root decomposition	260
cholinv()	Symmetric, positive-definite matrix inversion	262
cholsolve()	Solve AX=B for X using Cholesky decomposition	264
comb()	Combinatorial function	267
cond()	Condition number	268
conj()	Complex conjugate	270
corr()	Make correlation matrix from variance matrix	271
cross()	Cross products	272
crossdev()	Deviation cross products	280
designmatrix()	Design matrices	283
det()	Determinant of matrix	284
diag()	Create diagonal matrix	286
diag0cnt()	Count zeros on diagonal	288
diagonal()	Extract diagonal into column vector	289
dir()	File list	290
direxists()	Whether directory exists	292
direxternal()	Obtain list of existing external globals	293
display()	Display text interpreting SMCL	294
displayas()	Set display level	296
displayflush()	Flush terminal-output buffer	298
dsign()	FORTRAN-like DSIGN() function	299
e()	Unit vectors	300
editmissing()	Edit matrix for missing values	301
edittoint()	Edit matrix for roundoff error (integers)	302
edittozero()	Edit matrix for roundoff error (zeros)	304
editvalue()	Edit (change) values in matrix	307
eigensystem()	Eigenvalues and eigenvectors	309
eltype()	Element type and organizational type of object	318
epsilon()	unit roundoff error (machine precision)	320
_equilrc()	Row and column equilibration	321
error()	Issue error message	327
errprintf()	Format output and display as error message	331
exit()	Terminate execution	333
exp()	Exponentiation and logarithms	335
factorial()	Factorial and gamma function	336
favorspeed()	Whether speed or space is to be favored	338
fft()	Fourier transform	339
fileexists()	Whether file exists	347
_fillmissing()	Fill matrix with missing values	348
findexternal()	Find, create, and remove external globals	349
findfile()	Find file	353
fopen()	File I/O	354
fullsvd()	Full singular value decomposition	366
Hilbert()	Hilbert matrices	371
I()	Identity matrix	372
indexnot()	Find character not in list	373

<code>invorder()</code>	Permutation vector manipulation	374
<code>invsym()</code>	Symmetric real matrix inversion	376
<code>isdiagonal()</code>	Whether matrix is diagonal	379
<code>isfleeting()</code>	Whether argument is temporary	380
<code>isreal()</code>	Storage type of matrix	383
<code>isrealvalues()</code>	Whether matrix contains only real values	384
<code>issymmetric()</code>	Whether matrix is symmetric (Hermitian)	385
<code>isview()</code>	Whether matrix is view	386
<code>J()</code>	Matrix of constants	387
<code>logit()</code>	Log odds and complementary log-log	389
<code>lowertrianglc()</code>	Extract lower or upper triangle	390
<code>lud()</code>	LU decomposition	393
<code>luinv()</code>	Square matrix inversion	396
<code>lusolve()</code>	Solve $AX=B$ for $X$ using LU decomposition	398
<code>makesymmetric()</code>	Make square matrix symmetric (Hermitian)	402
<code>matexpsym()</code>	Exponentiation and logarithms of symmetric matrices	403
<code>matpowersym()</code>	Powers of a symmetric matrix	405
<code>mean()</code>	Means, variances, and correlations	407
<code>mindouble()</code>	Minimum and maximum nonmissing value	409
<code>minmax()</code>	Minimums and maximums	411
<code>missing()</code>	Count missing and nonmissing values	414
<code>missingof()</code>	Appropriate missing value	416
<code>mod()</code>	Modulus	417
<code>more()</code>	Create <code>-more-</code> condition	418
<code>norm()</code>	Matrix and vector norms	419
<code>normal()</code>	Cumulatives, reverse cumulatives, and densities	421
<code>panelsetup()</code>	Panel-data processing	425
<code>pathjoin()</code>	File path manipulation	433
<code>pinv()</code>	Moore–Penrose pseudoinverse	436
<code>polyeval()</code>	Manipulate and evaluate polynomials	438
<code>printf()</code>	Format output	442
<code>qrd()</code>	QR decomposition	446
<code>qrinv()</code>	Generalized inverse of matrix via QR decomposition	454
<code>qrsolve()</code>	Solve $AX=B$ for $X$ using QR decomposition	456
<code>quadcross()</code>	Quad-precision cross products	459
<code>range()</code>	Vector over specified range	461
<code>rank()</code>	Rank of matrix	463
<code>Re()</code>	Extract real or imaginary part	465
<code>reldif()</code>	Relative/absolute difference	466
<code>rows()</code>	Number of rows and number of columns	468
<code>rowshape()</code>	Reshape matrix	469
<code>setbreakintr()</code>	Break-key processing	471
<code>sign()</code>	Sign and complex quadrant functions	474
<code>sin()</code>	Trigonometric and hyperbolic functions	476
<code>sizeof()</code>	Number of bytes consumed by object	478
<code>solvelower()</code>	Solve $AX=B$ for $X$ , $A$ triangular	479
<code>solve_tol()</code>	Tolerance used by solvers and inverters	482
<code>sort()</code>	Reorder rows of matrix	484
<code>spline3()</code>	Cubic spline interpolation	487
<code>sqrt()</code>	Square root	489
<code>st_addobs()</code>	Add observations to current Stata dataset	490

valofexternal()	Obtain value of external global	592
Vandermonde()	Vandermonde matrices	593
vec()	Stack matrix columns	594

**[M-6] Mata glossary of common terms**

glossary	Mata glossary of common terms	599
----------	-------------------------------	-----