

Title

[M-0] intro — Introduction to the Mata manual
--

Contents

Section	Description
[M-1]	Introduction and advice
[M-2]	Language definition
[M-3]	Commands for controlling Mata
[M-4]	Index and guide to functions
[M-5]	Functions
[M-6]	Mata glossary of common terms

Description

Mata is a matrix programming language that can be used by those who want to perform matrix calculations interactively and by those who want to add new features to Stata.

This entry describes this manual and what has changed since Stata 9.

Remarks

This manual is divided into five sections. Each section is organized alphabetically, but there is an introduction in front that will help you get around.

If you are new to Mata, here is a helpful reading list: start by reading

[M-1] first	Introduction and first session
[M-1] interactive	Using Mata interactively
[M-1] how	How Mata works

You may find other things in section [M-1] that interest you. For a table of contents, see

[M-1] intro	Introduction and advice
--------------------	-------------------------

Whenever you see a term that you are unfamiliar with, see

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

Now that you know the basics, if you are interested, you can look deeper into Mata's programming features:

[M-2] **syntax** Mata language grammar and syntax

[M-2] **syntax** is pretty dense reading, but it summarizes nearly everything. The other entries in [M-2] repeat what is said there but with more explanation; see

[M-2] **intro** Language definition

Along the way, you will eventually be guided to sections [M-4] and [M-5]. [M-5] documents Mata's functions; the alphabetical order makes it easy to find a function if you know its name but makes learning what functions there are hopeless. That is the purpose of [M-4]—to present the functions in logical order. See

[M-4] **intro** Index and guide to functions

Mathematical

[M-4] **matrix** Matrix functions
[M-4] **solvers** Matrix solvers and inverters
[M-4] **scalar** Scalar functions
[M-4] **statistical** Statistical functions
[M-4] **mathematical** Other important functions

Utility and manipulation

[M-4] **standard** Functions to create standard matrices
[M-4] **utility** Matrix utility functions
[M-4] **manipulation** Matrix manipulation functions

Stata interface

[M-4] **stata** Stata interface functions

String, I/O, and programming

[M-4] **string** String manipulation functions
[M-4] **io** I/O functions
[M-4] **programming** Programming functions

What's new

Mata was first released in Stata 9, and thus the release of Stata 10 amounts to Mata's first update. Here is what's changed and what's new:

1. New Stata command `include` is a variation on `do` and `run` and is useful for implementing `#include` for header files in advanced programming situations. See [P] **include** and type `viewsource optimize.mata` for an example of use.
2. Mata now has structures, which will be of special interest to those writing large systems. See [M-2] **struct** and [M-5] **liststruct()**.

3. Mata now engages in more thorough type checking, and produces better code, for those who explicitly declare arguments and variables.
4. Mata inherits all of Stata's formats and functions for dealing with the new date/time variables and values; see [M-5] **date()** and [M-5] **fmtwidth()**.
5. New functions **inbase()** and **frombase()** perform base conversions; see [M-5] **inbase()**.
6. New function **floatround()** returns values rounded to float precision. This is Mata's equivalent of Stata's **float()** function. See [M-5] **floatround()**.
7. New function **nameexternal()** returns the name of an external; see [M-5] **findexternal()**.
8. Concerning matrix manipulation,
 - a. Matrix multiplication is now faster when one of the matrices contains many zeros, as is function **cross()**.
 - b. Appending rows or columns to a matrix using **,** and **** is now faster.
 - c. New function **_diag()** replaces the principal diagonal of a matrix with a specified vector or scalar; see [M-5] **_diag()**.
 - d. New functions **select()** and **st_select()** select rows or columns of a matrix on the basis of a criterion; see [M-5] **select()**.
 - e. Existing functions **rowsum()**, **colsum()**, **sum()**, **quadrowsum()**, **quadcolsum()**, and **quadsum()** now allow an optional second argument that determines how missing values are handled; see [M-5] **sum()**.
 - f. New functions **runningsum()**, **quadrunningsum()**, **_runningsum()**, and **_quadrunningsum()** return the running sum of a vector; see [M-5] **runningsum()**.
 - g. New functions **minindex()** and **maxindex()** return the indices of minimums and maximums (including tied values) of a vector; see [M-5] **minindex()**.
9. Concerning statistics,
 - a. New Mata function **optimize()** performs minimization and maximization. You can code just the function, the function and its first derivatives, or the function and its first and second derivatives. Optimization techniques include Newton–Raphson, Davidon–Fletcher–Powell, Broyden–Fletcher–Goldfarb–Shanno, Berndt–Hall–Hall–Hausman, and the simplex method Nelder–Mead. See [M-5] **optimize()**.
 - b. New function **cvpermute()** forms permutations; see [M-5] **cvpermute()**.
 - c. New function **ghk()** provides the Geweke–Hajivassiliou–Keane multivariate normal simulator; see [M-5] **ghk()**. New function **ghkfast()** is faster but a little more difficult to use; see [M-5] **ghkfast()**.
 - d. New functions **halton()**, **_halton()**, and **ghalton()** compute Halton and Hammersley sets; see [M-5] **halton()**.
 - e. The new density and probability functions available in Stata are also available in Mata, including **binomial()**, **binomialtail()**, **gammaptail()**, **invgammaptail()**, **invbinomialtail()**, **ibetatail()**, **invibetatail()**, **lnnormal()**, and **lnnormalden()**; see [M-5] **normal()**.

Also, as in Stata, convergence and accuracy of many of the cumulatives, reverse cumulatives, and density functions have been improved.

- f. Existing Mata functions `mean()`, `variance()`, `quadvariance()`, `meanvariance()`, `quadmeanvariance()`, `correlation()`, and `quadcorrelation()` now make the weight argument optional. If not specified, unweighted estimates are returned. See [M-5] **mean()**.

10. Concerning string processing,

- a. New function `strtrim()` replaces multiple, consecutive internal spaces with one space; see [M-5] **strtrim()**.
- b. New functions `strtoreal()` and `_strtoreal()` convert strings to numeric values; see [M-5] **strtoreal()**.
- c. New function `_substr()` substitutes a substring into an existing string; see [M-5] **_substr()**.
- d. New function `invtokens()` is the inverse of the existing function `tokens()`; see [M-5] **invtokens()**.
- e. New function `tokenget()` performs advanced parsing; see [M-5] **tokenget()**.

11. Concerning I/O,

- a. New function `adosubdir()` returns the subdirectory in which Stata would search for a file; see [M-5] **adosubdir()**. New function `pathsearchlist(fn)` returns a row vector of full paths/filenames specifying all the locations, in order, where Stata would look for the specified *fn* along the official Stata ado-path; see [M-5] **pathjoin()**.
- b. New function `byteorder()` returns 1 if the computer is HILO and returns 2 if the computer is LOHI; see [M-5] **byteorder()**.
- c. New functions `bufput()` and `bufget()` copy elements into and out of buffers; see [M-5] **bufio()**.
- d. Existing function `cat()` now allows optional second and third arguments that specify the beginning and ending lines of the file to read; see [M-5] **cat()**.

12. New functions `ferrortext()` and `freturncode()` obtain error messages and return codes following an I/O error; see [M-5] **ferrortext()**.

13. Concerning the Stata interface,

- a. New function `stataversion()` returns the version of Stata that you are running, and new function `statasetversion()` allows setting it. See [M-5] **stataversion()**.
- b. New function `setmore()` allows turning `more` on and off. New function `setmoreonexit()` allows restoring `more` to its original setting when execution ends. See [M-5] **more()**.

Also See

[M-1] **first** — Introduction and first session

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