6 Managing memory

Contents

6.1   Memory-size considerations

Stata works with a copy of data that it loads into memory. To be precise, Stata can work with multiple datasets in memory at the same time. See [D] **frames intro**.

Memory allocation is automatic. Stata automatically sizes itself up and down as your session progresses. Stata obtains memory from the operating system and draws no distinction between real and virtual memory. Virtual memory is memory that resides on disk that operating systems supply when physical memory runs short. Virtual memory is slow but adequate in cases when you have a dataset that is too large to load into real memory. If you wish to limit the maximum amount of memory Stata can use, you can set max_memory; see [D] **memory**. If you use the Linux operating system, we strongly suggest you set max_memory; see *Serious bug in Linux OS* in [D] **memory**.

6.2   Compressing data

Stata stores data in memory. The compress command reduces the amount of memory required to store the data without loss of precision or any other disadvantages; see [D] **compress**. Typing compress every so often is a good idea.

compress works by examining the values you have stored and changing the data types of variables when that can be done without loss of precision. For instance, you may have a variable stored as float but that records only integer values between -127 and 100. compress would change the storage type of that variable to byte and save 3 bytes per observation. If you had 100 variables like that, the savings would be 300 bytes per observation, and if you had 3,000,000 observations, the total savings would be nearly 900 megabytes.

6.3   Setting maxvar

If you get the error message "no room to add more variables", r(900), do not jump to the conclusion that you have exceeded Stata's capacity.

maxvar specifies the maximum number of variables you can use. The default setting depends on whether you are using Stata/MP, Stata/SE, or Stata/BE. To determine the current setting, type query memory at the Stata prompt.

If you use Stata/MP, you can reset this maximum number to 120,000. If you use Stata/SE, you can reset this maximum number to 32,767. Set maxvar to more than you need—at least 20 more than you need but not too much more than you need. Figure that each 10,000 variables consumes roughly 0.5 megabytes of memory.

You reset maxvar using the `set maxvar` command,

set maxvar # [, permanently]

where $2{,}048 \leq \# \leq 120{,}000$, depending on your edition of Stata. You can reset maxvar repeatedly during a session. If you specify the `permanently` option, you change maxvar not only for this session but also for future sessions. Each additional 10,000 variables specified with `set maxvar` requires Stata to set aside roughly 1.3 megabytes of memory for variable names, not including the data stored in those variables.

6.4 The memory command

The `memory` command will show you the major components of Stata's memory footprint.

```
. use https://www.stata-press.com/data/r19/regsmpl
(NLS women 14-26 in 1968)

. memory
Memory usage
```

	Used	Allocated
Data	856,020	67,108,864
strLs	0	0
Data & strLs	856,020	67,108,864
Data & strLs	856,020	67,108,864
Variable names, %fmts, ...	4,644	182,379
Overhead	1,081,344	1,081,744
Stata matrices	0	0
ado-files	34,589	34,589
Stored results	0	0
Mata matrices	0	0
Mata functions	0	0
set maxvar usage	5,281,738	5,281,738
Other	4,066	4,066
Total	7,252,861	73,693,380

See [D] **memory**.

6.5 Setting aside memory for temporary storage of preserved datasets

Stata has a feature to `preserve` and `restore` datasets, allowing you to manipulate the data during an analysis and bring them back without harm. Stata/MP uses memory to make copies of these datasets as fast as possible. Stata/SE and Stata/BE make the copies on disk.

To control the amount of memory Stata/MP will use for these temporary dataset copies before it falls back to slower disk storage, use the `set max_preservemem` setting. See [P] **preserve** for more details.