save — Save Stata dataset

Description

`save` stores the dataset currently in memory on disk under the name *filename*. If *filename* is not specified, the name under which the data were last known to Stata (`c(filename)`) is used. If *filename* is specified without an extension, `.dta` is used. If your *filename* contains embedded spaces, remember to enclose it in double quotes.

Stata 14 through 17 have the same dataset format so long as the dataset has 32,767 variables or less. Stata 18 and 19 have the same format too, unless the data have alias variables in them; see below. Since Stata/MP 15, Stata/MP has supported more than 32,767 variables and thus has a slightly different dataset format when there are that many variables. If you are using Stata 18 or later, do not have alias variables, and you want to save a dataset so that it may be read by someone using Stata 15, Stata 16, or Stata 17, simply use the `save` command; those older versions will be able to read it. If the dataset has more than 32,767 variables, it can be read by Stata/MP 15, Stata/MP 16, Stata/MP 17, and Stata/MP 18. If you want to save a dataset so that it may be read by someone using Stata 14, again simply use the `save` command; Stata 14 will be able to read it so long as it does not have more than 32,767 variables. Stata 14 supports at most 32,767 variables.

`saveold` saves the dataset currently in memory on disk under the name *filename* in previous `.dta` formats, namely, those for Stata 13, 12, or 11. If you are using Stata 19 and want to save a file so that it may be read by someone using an older version of Stata, use the `saveold` command.

Alias variables, introduced in Stata 18, are variables that reference other variables in a linked frame; see [D] **fralias**. If your dataset does not contain alias variables, then you can use commands `save` and `saveold` as mentioned above. For datasets with alias variables, Stata 18 and later use a new format to accommodate this new variable type. In addition, Stata/MP has a new format to accommodate datasets with alias variables and more than 32,767 variables. If you are using data with alias variables and want to save a dataset so that it may be read by someone using an older version of Stata, then you must either drop the alias variables or use command `frunalias` to change the alias variables into copies of the variables they reference. Then you use commands `save` or `saveold` as mentioned above. The same is true for datasets with alias variables and more than 32,767 variables and older versions of Stata/MP.

Quick start

Save data in memory to `mydata.dta` in the current directory

```
save mydata
```

Same as above, but overwrite `mydata.dta` if it exists

```
save mydata, replace
```

Also save value labels that have not been applied to variables

```
save mydata, replace orphans
```

Save data in Stata 13 format

```
saveold mydata
```

Menu

File > Save as...

Syntax

Save data in memory to file

`save` [*filename*] [, *save_options*]

Save data in memory to file in Stata 13, 12, or 11 format

`saveold` *filename* [, *saveold_options*]

save_options	Description
`nolabel`	omit value labels from the saved dataset
`replace`	overwrite existing dataset
`all`	save `e(sample)` with the dataset; programmer's option
`orphans`	save all value labels
`emptyok`	save dataset even if zero observations and zero variables

saveold_options	Description
`version(#)`	specify version 11 ≤ # ≤ 18; default is `version(13)`, meaning Stata 13 format
`nolabel`	omit value labels from the saved dataset
`replace`	overwrite existing dataset
`all`	save `e(sample)` with the dataset; programmer's option

Options for save

`nolabel` omits value labels from the saved dataset. The associations between variables and value-label names, however, are saved along with the dataset label and the variable labels.

`replace` permits `save` to overwrite an existing dataset.

`all` is for use by programmers. If specified, `e(sample)` will be saved with the dataset. You could run a regression; `save mydata, all`; `drop _all`; `use mydata`; and `predict yhat if e(sample)`.

`orphans` saves all value labels, including those not attached to any variable.

`emptyok` is a programmer's option. It specifies that the dataset be saved, even if it contains zero observations and zero variables. If `emptyok` is not specified and the dataset is empty, `save` responds with the message "no variables defined".

Options for saveold

`version(#)` specifies which previous `.dta` file format is to be used. # may be `18`, `17`, `16`, `15`, `14`, `13`, `12`, or `11`. The default is `version(13)`, meaning Stata 13 format. To save datasets in the modern, Stata 18 and later format, use the `save` command, not `saveold`. Stata 14 through Stata 19 share the same format, provided that there are no alias variables, so you do not have to use `saveold` to save a Stata 14, 15, 16, 17, or 18 dataset; simply use `save`.

nolabel omits value labels from the saved dataset. The associations between variables and value-label names, however, are saved along with the dataset label and the variable labels.

replace permits saveold to overwrite an existing dataset.

all is for use by programmers. If specified, e(sample) will be saved with the dataset. You could run a regression; save mydata, all; drop _all; use mydata; and predict yhat if e(sample).

Remarks and examples

Stata keeps the data on which you are currently working in your computer's memory. You put the data there in the first place; see **[U] 22 Entering and importing data**. Thereafter, you can save the dataset on disk so that you can use it easily in the future. Stata stores your data on disk in a compressed format that only Stata understands. This does not mean, however, that you are locked into using only Stata. Any time you wish, you can export the data to a format other software packages understand; see **[D] export**.

Stata goes to a lot of trouble to keep you from accidentally losing your data. When you attempt to leave Stata by typing exit, Stata checks that your data have been safely stored on disk. If not, Stata refuses to let you leave. (You can tell Stata that you want to leave anyway by typing exit, clear.) Similarly, when you save your data in a disk file, Stata ensures that the disk file does not already exist. If it does exist, Stata refuses to save it. You can use the replace option to tell Stata that it is okay to overwrite an existing file.

▷ Example 1

We have entered data into Stata for the first time. We have the following data:

```
. describe

Contains data
 Observations:            39
    Variables:             5
───────────────────────────────────────────────────────────────────────────────
Variable      Storage   Display    Value
    name         type    format    label      Variable label
───────────────────────────────────────────────────────────────────────────────
acc_rate        float   %9.0g                 Accident rate
spdlimit        float   %9.0g                 Speed limit
acc_pts         float   %9.0g                 Access points per mile
rate            float   %9.0g      rcat       Accident rate per million vehicle
                                                miles
spdcat          float   %9.0g      scat       Speed limit category
───────────────────────────────────────────────────────────────────────────────
Sorted by:
     Note: Dataset has changed since last saved.
```

We have a dataset containing 39 observations on five variables, and, evidently, we have gone to a lot of trouble to prepare this dataset. We have used the label data command to label the data Minnesota Highway Data, the label variable command to label all the variables, and the label define and label values commands to attach value labels to the last two variables. (See **[U] 12.6.3 Value labels** for information about doing this.)

At the end of the describe, Stata notes that the "dataset has changed since last saved". This is Stata's way of gently reminding us that these data need to be saved. Let's save our data:

```
. save hiway
file hiway.dta saved
```

We type save hiway, and Stata stores the data in a file named hiway.dta. (Stata automatically added the .dta suffix.) Now when we describe our data, we no longer get the warning that our dataset has not been saved; instead, we are told the name of the file in which the data are saved:

```
. describe

Contains data from hiway.dta
 Observations:            39                  Minnesota Highway Data, 1973
    Variables:             5                  21 Jul 2000 11:42
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
acc_rate        float   %9.0g                 Accident rate
spdlimit        float   %9.0g                 Speed limit
acc_pts         float   %9.0g                 Access points per mile
rate            float   %9.0g      rcat       Accident rate per million vehicle
                                                miles
spdcat          float   %9.0g      scat       Speed limit category
-------------------------------------------------------------------------------
Sorted by:
```

Just to prove to you that the data have really been saved, let’s eliminate the copy of the data in memory by typing drop _all:

```
. drop _all

. describe

Contains data
 Observations:             0
    Variables:             0
Sorted by:
```

We now have no data in memory. Because we saved our dataset, we can retrieve it by typing use hiway:

```
. use hiway
(Minnesota Highway Data, 1973)

. describe

Contains data from hiway.dta
 Observations:            39                  Minnesota Highway Data, 1973
    Variables:             5                  21 Jul 2000 11:42
-------------------------------------------------------------------------------
Variable      Storage   Display    Value
    name         type    format    label      Variable label
-------------------------------------------------------------------------------
acc_rate        float   %9.0g                 Accident rate
spdlimit        float   %9.0g                 Speed limit
acc_pts         float   %9.0g                 Access points per mile
rate            float   %9.0g      rcat       Accident rate per million vehicle
                                                miles
spdcat          float   %9.0g      scat       Speed limit category
-------------------------------------------------------------------------------
Sorted by:
```

◁

▷ Example 2

Continuing with our previous example, we have saved our data in the file `hiway.dta`. We continue to work with our data and discover an error; we made a mistake when we typed one of the values for the `spdlimit` variable:

```
. list in 1/3

     ┌─────────────────────────────────────────────────────┐
     │ acc_rate   spdlimit   acc_pts       rate     spdcat │
     ├─────────────────────────────────────────────────────┤
  1. │     1.61         50       2.2    Below 4   Above 60 │
  2. │     1.81         60       6.8    Below 4   55 to 60 │
  3. │     1.84         55        14    Below 4   55 to 60 │
     └─────────────────────────────────────────────────────┘
```

In the first observation, the `spdlimit` variable is 50, whereas the `spdcat` variable indicates that the speed limit is more than 60 miles per hour. We check our original copy of the data and discover that the `spdlimit` variable ought to be 70. We can fix it with the `replace` command:

```
. replace spdlimit=70 in 1
(1 real change made)
```

If we were to `describe` our data now, Stata would warn us that our data have changed since they were last saved:

```
. describe

Contains data from hiway.dta
 Observations:            39                  Minnesota Highway Data, 1973
    Variables:             5                  21 Jul 2000 11:42
────────────────────────────────────────────────────────────────────────────
Variable      Storage   Display    Value
    name         type    format    label      Variable label
────────────────────────────────────────────────────────────────────────────
acc_rate        float   %9.0g                 Accident rate
spdlimit        float   %9.0g                 Speed limit
acc_pts         float   %9.0g                 Access points per mile
rate            float   %9.0g      rcat       Accident rate per million vehicle
                                                miles
spdcat          float   %9.0g      scat       Speed limit category
────────────────────────────────────────────────────────────────────────────
Sorted by:
     Note: Dataset has changed since last saved.
```

We take our cue and attempt to `save` the data again:

```
. save hiway
file hiway.dta already exists
r(602);
```

Stata refuses to honor our request, telling us instead that “file hiway.dta already exists”. Stata will not let us accidentally overwrite an existing dataset. To `replace` the data, we must do so explicitly by typing `save hiway, replace`. If we want to save the file under the same name as it was last known to Stata, we can omit the filename:

```
. save, replace
file hiway.dta saved
```

Now our data are saved.

◁

Also see

[D] **compress** — Compress data in memory

[D] **export** — Overview of exporting data from Stata

[D] **fralias** — Alias variables from linked frames

[D] **frunalias** — Change storage type of alias variables

[D] **import** — Overview of importing data into Stata

[D] **use** — Load Stata dataset

[P] **File formats .dta** — Description of .dta file format

[U] 11.6 Filenaming conventions

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. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on citing Stata documentation.