

checksum — Calculate checksum of file

Syntax

Description

Options

Remarks and examples

Stored results

Also see

Syntax

```
checksum filename [ , options ]

set checksum { on | off } [ , permanently ]
```

<i>options</i>	Description
save	save output to <i>filename</i> .sum; default is to display a report
replace	may overwrite <i>filename</i> .sum; use with save
saving(<i>filename2</i> [, replace])	save output to <i>filename2</i> ; alternative to save

Description

checksum creates *filename*.sum files for later use by Stata when it reads files over a network. These optional files are used to reduce the chances of corrupted files going undetected. Whenever Stata reads file *filename*.*suffix* over a network, whether by use, net, update, etc., it also looks for *filename*.sum. If Stata finds that file, Stata reads it and uses its contents to verify that the first file was received without error. If there are errors, Stata informs the user that the file could not be read.

set checksum on tells Stata to verify that files downloaded over a network have been received without error.

set checksum off, which is the default, tells Stata to bypass the file verification.

□ Technical note

checksum calculates a CRC checksum following the POSIX 1003.2 specification and displays the file size in bytes. checksum produces the same results as the Unix cksum command. Comparing the checksum of the original file with the received file guarantees the integrity of the received file.

When comparing Stata’s checksum results with those of Unix, do not confuse Unix’s sum and cksum commands. Unix’s cksum and Stata’s checksum use a more robust algorithm than that used by Unix’s sum. In some Unix operating systems, there is no cksum command, and the more robust algorithm is obtained by specifying an option with sum.



Options

save saves the output of the **checksum** command to the text file *filename.sum*. The default is to display a report but not create a file.

replace is for use with **save**; it permits Stata to overwrite an existing *filename.sum* file.

saving(filename2 [, replace]) is an alternative to **save**. It saves the output in the specified filename. You must supply a file extension if you want one, because none is assumed.

permanently specifies that, in addition to making the change right now, the **checksum** setting be remembered and become the default setting when you invoke Stata.

Remarks and examples

[stata.com](http://www.stata.com)

► Example 1

Say that you wish to put a dataset on your homepage so that colleagues can use it over the Internet by typing

```
. use http://www.myuni.edu/department/~joe/mydata
```

mydata.dta is important, and even though the chances of the file *mydata.dta* being corrupted by the Internet are small, you wish to guard against that. The solution is to create the checksum file named *mydata.sum* and place that on your homepage. Your colleagues need type nothing different, but now Stata will verify that all goes well. When they use the file, they will see either

```
. use http://www.myuni.edu/department/~joe/mydata
(important data from joe)
```

or

```
. use http://www.myuni.edu/department/~joe/mydata
file transmission error (checksums do not match)
http://www.myuni.edu/department/~joe/mydata.dta not downloaded
r(639);
```

To make the checksum file, change to the directory where the file is located and type

```
. checksum mydata.dta, save
Checksum for mydata.dta = 263508742, size = 4052
file mydata.sum saved
```

► Example 2

Let's use **checksum** on the **auto** dataset that is shipped with Stata. We will load the dataset and save it to our current directory.

```
. use http://www.stata-press.com/data/r13/auto
(1978 Automobile Data)

. save auto
file auto.dta saved

. checksum auto.dta
Checksum for auto.dta = 2694850408, size = 6442
```

We see the report produced by `checksum`, but we decide to save this information to a file.

```
. checksum auto.dta, save
. type auto.sum
1 6442 2694850408
```

The first number is the version number (possibly used for future releases). The second number is the file's size in bytes, which can be used with the checksum value to ensure that the file transferred without corruption. The third number is the checksum value. Although two different files can have the same checksum value, two files with the same checksum value almost certainly could not have the same file size.

This example is admittedly artificial. Typically, you would use `checksum` to verify that no file transmission error occurred during a web download. If you want to verify that your own data are unchanged, using `datasignature` is better; see [D] [datasignature](#).

◀

Stored results

`checksum` stores the following in `r()`:

Scalars

<code>r(version)</code>	checksum version number
<code>r(filelen)</code>	length of file in bytes
<code>r(checksum)</code>	checksum value

Also see

[D] [use](#) — Load Stata dataset

[R] [net](#) — Install and manage user-written additions from the Internet

[D] [datasignature](#) — Determine whether data have changed