The Stata Journal (2007)
7, Number 3, pp. 438–439

Stata tip 50: Efficient use of summarize

Nicholas J. Cox
Department of Geography
Durham University
Durham City, UK
n.j.cox@durham.ac.uk

The `summarize` command must be one of the most commonly used Stata commands. Yet, strangely, one of its options is often not used, even though it can be the best solution to a user's problem. Here I flag this neglected `meanonly` option and speculate briefly on why it is often overlooked.

If you fire up `summarize, meanonly`, no results appear in the Results window for you to examine. This lack is deliberate. The option leaves r-class results in memory. (If you are unclear what that means, start with the online help for `return`.) The user must access those results by typing `return list` to see what they are or by feeding one or more results to something else, such as an explicit `display`, `generate`, or `replace` statement. Accessing the saved results should be done promptly after the `summarize, meanonly` command has finished, because results are ephemeral and will not survive beyond the next r-class command that is issued.

The `meanonly` option leaves in memory the mean, as the name implies, in `r(mean)`. However, contrary to what you might guess, it also leaves behind the count of nonmissing values, the sum, the weighted sum, the minimum, and the maximum in appropriately named results. These results are for the last-named variable. Thus, although invoking `summarize, meanonly` with two or more variables is legal, doing so is utterly pointless because results for all but the last will disappear and machine time will be wasted.

Incidentally, if all you want is a count, the `count` command offers a more direct solution; see [D] **count** and Cox (2007).

The difference between `summarize, meanonly` and `summarize` with no options is that the latter also calculates the variance and its square root, the standard deviation. The reason for the `meanonly` option is that this last calculation can be fairly time consuming in large datasets. Thus, if you need to use only one or more of the results left behind after `summarize, meanonly`, then specifying the option will be sensible. Programs or do-files that will be used repeatedly and/or on large datasets are especially suitable. Budding programmers can entertain themselves by identifying StataCorp programs that passed up opportunities for using `summarize, meanonly`. This issue underscores an old joke that you can always speed up a program that was originally written to run slowly.

As a concrete example, one common task is cycling over a set of categories defined by one or more variables. An easy way to do this is to use `egen, group()` to create a variable with integer values 1 and up (and, optionally, value labels with informative text). When you do not know the number of categories in advance,

```
. summarize group, meanonly
```


produces the maximum of `group`, which is the same as the number of categories present. Thus we can feed `r(max)` to whatever code that needs it, possibly a `forvalues` loop.

A small problem remains of explaining why people often overlook this `meanonly` option. I have three guesses. First, `summarize` is one of the commands that Stata users learn early. Typically, it quickly becomes clear that `summarize` does various things and `summarize, detail` does even more. Thus, users tend to feel that they are familiar with the command and do not study its help carefully. Second, the name `meanonly` is in some ways unfortunate and misleading, because much more than the mean is produced. Perhaps a synonym such as `summarize, short` would be a good idea. (Dropping the `meanonly` name is not likely, given the number of programs and commands that would break.) Third, the explanation of `meanonly` in the manual at [R] **summarize** does not give the complete picture on this option.

Reference

Cox, N. J. 2007. Speaking Stata: Making it count. *Stata Journal* 7: 117–130.