Stata tip 22: Variable name abbreviation

Philip Ryan
University of Adelaide
philip.ryan@adelaide.edu.au

Stata allows users to abbreviate any variable name to the shortest string of characters that uniquely identifies it, given the data currently loaded in memory (see [U] 11.2.3 Variable-name abbreviation). Stata also offers three wildcard characters, *, ~, and ? (see [U] 11.4.1 Lists of existing variables), so users have substantial flexibility in how variables may be referenced.

Stata also allows users to control the values of many of its system parameters using the set command (see [R] set). One of these parameters is varabbrev, which may be toggled on, allowing variable names to be abbreviated, or off, requiring the user to spell out entire variable names.

The default is to allow abbreviations. But this convenience feature can bite. Suppose that in a program we wish to confirm the existence of a variable and that variable does not in fact exist:

```
. clear
. set varabbrev on
. set obs 10
   (obs was 0, now 10)
. generate byte myvar7 = 1
. confirm variable myvar
```

There is no error message here because myvar is an allowed abbreviation for myvar7. A bigger deal is that as myvar7 exists and not myvar, typing drop myvar would drop myvar7, which may or may not have been our intention.

But what if we had wanted to confirm explicitly the existence of variable myvar? There are two ways to do this:

1. Specify the confirm command with the exact option (see [R] confirm):

```
. confirm variable myvar, exact
   variable myvar not found
   r(111);
```

2. Toggle variable abbreviation off:

```
. set varabbrev off
. confirm variable myvar
   variable myvar not found
   r(111);
```
Note that the status of `varabbrev` does not affect the display of variable names. For example,

```
. sysuse auto, clear
(1978 Automobile Data)
. set varabbrev off
. rename weight this_is_a_very_long_varname
. regress price turn length this_is_a_very_long_varname
(output omitted)
```

|       | Coef. | Std. Err. | t     | P>|t|  | [95% Conf. Interval] |
|-------|-------|-----------|-------|------|----------------------|
| price | -318.2055 | 127.1241 | -2.50 | 0.015 | -571.7465  -64.66452 |
| turn  | -66.17856 | 39.87361 | -1.66 | 0.101 | -145.704  13.34684 |
| length| 5.382135 | 1.116756 | 4.82  | 0.000 | 3.154834  7.609435 |
| this_is_a_very_long_varname | 14967.64 | 4541.836 | 3.30  | 0.002 | 5909.228  24026.04 |

In this display, Stata has abbreviated the long variable name, despite the current value of `varabbrev`.

Note that the `list` command has its own option to allow the user partial control of the display; see `[D] list`. As we set `varabbrev off`, we must specify only unabbreviated variable names in a `list` command, but we can override Stata’s default abbreviation in the display using the `abbreviate()` option:

```
. list make turn this_is_a_very_long_varname in 1/4, abb(21)
```

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>make</td>
<td>turn</td>
<td>this_is_a_very_long_varname</td>
</tr>
<tr>
<td>AMC Concord</td>
<td>40</td>
<td>2,930</td>
</tr>
<tr>
<td>AMC Pacer</td>
<td>40</td>
<td>3,350</td>
</tr>
<tr>
<td>AMC Spirit</td>
<td>35</td>
<td>2,640</td>
</tr>
<tr>
<td>Buick Century</td>
<td>40</td>
<td>3,250</td>
</tr>
</tbody>
</table>

The default value for `abbreviate()` is 8, so that otherwise the variable name would have been displayed as `this_is_a_very_long_varname`. 