GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Macros

Stata Self-Learning Course

Macros

- Virtual objects: store strings and numbers
- Use for
  - Creating variable lists
  - Storing results
  - Refer to file paths
- Example: macro
  - local
  - global

Locals and globals

- Locals
  - Store content for duration of dofile or program
  - Called by *`localname'*
- Globals
  - Store content until Stata is closed
  - Called by *$globalname*
- Creating locals

  local *localname content*

  local *localname = expression*

  local *localname: extended_macro_function*
- Creating globals

  replace local by global

Macros: Some hints I

- Only use globals if a local doesn’t do the job
- Call locals or globals with ` ’ and $
- For printing a $-sign, write \$

```
global word hello
display “$14”
display “$word”
display “\$word”
```

- Using macros in file paths on Windows

```
global myfolder user
✖ use C:\Windows\$myfolder\file
```

- Always use / instead of \

Macros: Some hints II

- Locals expand from inside to outside

  Locals defined as x1 = 2, x2 = 5, x3 = 10, and i = 1

  `x`i’’ → expands to `x1’ and then to 2

- Expand globals from inside to outside using curly brackets

  Globals defined as x1 = 2, x2 = 5, x3 = 10, and i = 1

  ${x$i} → expands to $x1 and then to 1

  xi → equivalent to `x’`i’, expands to 1 as local/global x is empty

- Nest globals and locals

  ${x`i’}

Macros: Some hints III

- Use curly brackets if a global is followed by text

local a number	→ defines local
global a number	→ defines global
display “`a’1”	→ reads local a and 1
display “$a1”	→ reads global a1
display “${a}1”	→ reads global a and 1

Macros: Some hints IV

- Macros always contain the content assigned at creation

```
global varlist var1 var2
global newlist $varlist var3
display “$newlist”                → var1 var2 var3
global varlist var1
display “$newlist”                → var1 var2 var3
```

- Creating permanent link between globals:

```
global varlist var1 var2
global newlist \$varlist var3
display “$newlist”                → var1 var2 var3
global varlist var1
display “$newlist”                → var1 var3
```

Macros: Some hints V

- When calling a macro, double quotes at the beginning and end are omitted (but not in the middle)

  local a “Germany”

  local b Germany

  ✖ gen country_number = 1 if country_name == `a’

  ✖ gen country_number = 1 if country_name == `b’

  gen country_number = 1 if country_name == “`a’”

- Compound double quotes `” ”’ avoid confusion

  “A”B”C” vs. `”A`”B”`C”’

- Always use compound double quotes if a macro might contain normal double quotes to avoid unwanted behaviour!

Macros: Some hints VI

- Some commands accept numbers as arguments

  set obs 10

- Problems:
  - Command only works if current number of observations is smaller than 10
  - Unclear how many additional observations are created
- Solution:

  set obs `=_N + 5’ → calculates _N and then executes command