

SOC 5050: Week 10 Commands Quick Reference

Christopher Prener, Ph.D.

October 23rd, 2016

Macros

Local Macros

```
local macroName "string"  
local macroName = val  
display 'macroName'
```

Global Macros

```
global macroName "string"  
global macroName = val  
display $macroName
```

Scalars

```
return list1
```

¹ This command must be executed after a statistical test that stores scalars.

MarkDoc

```
txt "The average of the variable 'x' is " ///  
    %10.3f 'mpgMean' " ." 2
```

² This can only be run during a MarkDoc session. The text %10.3f rounds the output to the thousandths place.

if and in Qualifiers

if

This qualifier can be used to set a condition on almost any command in Stata. The `if` qualifier should appear *before* any command options but after the main body of the command. The `if` qualifier must be paired with an expression that Stata can evaluate. The expression should use relational and (optionally) logical operators.

Relational Operators

> - greater then
< - less then
>= - greater then or equal
<= - less then or equal
== - equal³
!= - not equal

³ This is a common point of confusion for new programmers. Stata (and other languages) use a single equals sign (=) for the assignment of a value to an object. In order to differentiate the evaluation of an expression from this assignment, evaluations use the double equals sign (==).

Relational Operators

& - and

| - or

! - not

Combining Operators`summarize var1 if var2 <= 10``summarize var1 if var3 == 4``summarize var1 if var2 <= 10 & var3 == 4`*in*

This qualifier allows you to limit the scope of a command to a particular set of observations.⁴

`list var1 in 1``list var1 in 1/10`

⁴ The `list` command will display data for the given variables and (optionally) observations. A forward slash (/) can be used to specify a range of values.

The `in` qualifier is used far less frequently than the `if` qualifier. It is used principally for data exploration and occasionally (if narrowly applied) for data manipulation.

*T-Tests**One Sample T-Test*`ttest yVar = mu`*Levene's Test*`sdtest yVar, by(xVar)`*Independent Samples T-Test*`ttest yVar, by(xVar) [unequal welch]`*Dependent Samples T-Test*`ttest g1Var==g2Var`*Cohen's d*`esize twosample yVar, by(xVar) cohensd [unequal welch]`

Graphing Means

Collapsing Data

For graphing single means as well as means from independent samples, data must be “collapsed” or summarized using the

`collapse` command:

```
collapse (mean) mVar = yVar (sd) sVar = yVar (count) nVar=yVar [, by(xVar)]
```

Standard Error Bar Chart

```
serrbar mVar sVar xVar, scale(val)
```

This chart typically requires significant alternation to make it useable.

The following twoway graph options are *strongly recommended* for all graphs:

```
ytitle("string") - y-axis title
```

```
xlabel({xVal "string" xVal "string" | none}) - x-axis labels5
```

```
xtitle({"string" | ""}) - x-axis title
```

```
title("title text")
```

```
subtitle("subtitle text")
```

```
note("note text")
```

```
caption("caption text")
```

⁵ Pairs of `xVal` and *strings* can be repeated as necessary. For these graphs, you will need two pairs.

The following twoway graph options are *strongly recommended* for one sample t-test graphs *only* to visualize μ :

```
yline(yVal, lcolor(color)) - draw horizontal line representing  $\mu$ 
```

```
text(yVal 1 "&mu", place(n)) - label line as  $\mu$ 
```

The following twoway graph options *may* be needed depending on the distribution you are working with:

```
yscale(r(lowerVal upperVal)) - y-axis scale
```

```
ylabel(lowerVal(steps)upperVal, grid) - y-axis labels
```

```
xscale(r(-.2 1.2)) - x-axis scale6
```

⁶ This can be used as-is for both independent and dependent sample plots.

Typographic Details

```
{&mu} - add  $\mu$  letter
```

```
{it:string} - italicize text
```

Power Analysis

```
power twomeans mean1 mean2, sd(val) power(beta)
```

Document Details

Document produced by [Christopher Prener, Ph.D.](#) for the Saint Louis University course SOC 5050 - QUANTITATIVE ANALYSIS: APPLIED INFERENTIAL STATISTICS. See the [course wiki](#) and the repository [README.md](#) file for additional details.



This work is licensed under a [Creative Commons Attribution 4.0 International License](#).