Sociology 704: Regression Models for Categorical Data
Instructor: Natasha Sarkisian

Introduction to Stata

Basic syntax of Stata commands:

1. Command – What do you want to do?
2. Names of variables, files, etc. – Which variables or files do you want to use?
3. Qualifier on observations -- Which observations do you want to use?
4. Options – Do you have any other preferences regarding this command?

Obtain help and install user-written commands:

help *command*
search *keyword*
net search *keyword*
net install *pkgname* [, all replace force from(*directory_or_url*)]

Open and close files:

Data files:
use *filename.dta*, clear – opens data file
save *filename.dta*, replace
Log files:
log using *filename.log* [, append replace] – open log file
log close -- close log file (saves automatically)
translate – convert log file types (.log and .smcl) and recover results
cmdlog using *filename* – open command only log file
Do-files:
doedit filename.do – to create or edit a do-file
do *filename.do* – to execute a do-file
Working with directories:
cd *path* – change current working directory
sysdir – list Stata system directories (also allows to change them if necessary; see options in help)
pwd – list current working directory

Add comments:

* comment
// comment
/* comment */

Examine the data:

browse – explore the data
describe – get information on variables and labels
list *varnames* [in *exp*] – list the values of specified variables for specified observations
codebook *varnames* – summarize variables in codebook format
sum *varnames* [, detail] – get summary statistics
tab *varname*, [nolabel missing] – get frequency distribution (options: without value labels, display the missing data)
tab varname varname [, row col cell chi2] – generate a two-way table (Options: get percentages for rows, columns, cells; obtain chi-square test of independence)
tab1 varnames – generate separate frequency distribution for each variable

Basic graphical examination of the data:
dotplot varname – obtain a univariate frequency distribution graph
graph box varname – obtain a univariate boxplot
scatter varname varname – obtain a scatterplot for two variables
graph matrix varnames – obtains all possible scatterplots for a set of variables
graph save filename [,replace] – saves a graph into a .gph file
graph use filename – displays a previously saved graph

Set preferences:
set logtype text – to change the default type of log file to text
set more off [, permanently] – to turn off the feature wherein Stata pauses output with a --more-- in the Results window
set scheme schemename [, permanently]

Conditions:
< less
> more
== equal
<= less or equal
>= more or equal
~= or != not equal
Can connect them with & (and) and | (or).
Can also use parentheses to combine conditions.

Manage the data:
Edit – edit the data
drop [in *range*] [if *exp*] – drop observations
keep [in *range*] [if *exp*] – keep observations
drop *varnames* – drop variables
keep *varnames* – keep variables

Recode variables:
generate *newvarname* = *exp* [in *exp*] [if *exp*] – make a new variable
replace *varname* = *exp* [in *exp*] [if *exp*] – replace values of existing variable
recode *varname* (*rule*) (*rule*) … , generate(*newvarname*) – make a new variable
label variable *varname* "*label*" – create variable label
Create value labels:
label define *labelname label value label value*… -- defines a set of value labels
label values *varname labelname* – applies a set of value labels to a variable

Good resource for learning Stata:
http://www.ats.ucla.edu/stat/stata/

Opening and closing files

Let’s open Stata, rearrange the windows for convenience, then change the working directory:

```
. cd “C:\Documents and Settings\sarkisin\My Documents\”
```

Opening the log file:

```
log using learn_stata.log, replace
```

I choose .log rather than .scml type of file so it can be read in any text editor or word processor.

Note that if you are opening a Stata log file in a Word processor, you should change the font to a fixed width font, such as Courier New (otherwise the output looks misaligned). Courier New 10 or 9 point usually works the best.

You can always convert from one type of log file to another using translate command:

```
translate mylog.smcl mylog.log
```

By the way, you can use translate to recover a log when you have forgotten to start one:

```
translate @Results mylog.txt
```

Using comments in Stata -- everything typed after a star (*) or after // is treated as a comment and not executed; same with any text between /* and */

Opening the data:

```
. use gss2002.dta, clear
```

Examining the data

Describing the dataset:

```
. des
Contains data from C:\Documents and Settings\sarkisin\My Documents\gss2002.dta
  obs:         2,765
 vars:           997                          6 Oct 2004 15:21
 size:     2,961,315 (71.8% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
year            int    %8.0g                  gss year for this respondent
id              int    %8.0g                  respondnt id number
wrkstat         byte   %8.0g       wrkstat    labor frce status
hrs1            byte   %8.0g       hrs1       number of hours worked last week
hrs2            byte   %8.0g       hrs2       number of hours usually work a
                                                week
evwork          byte   %8.0g       evwork     ever work as long as one year
wrkslf          byte   %8.0g       wrkslf     r self-emp or works for somebody
wrkgovt         byte   %8.0g       wrkgovt    govt or private employee
occ80           int    %8.0g       occ80      rs census occupation code (1980)
--Break--
r(1);
```

I used Break button to stop Stata from producing more output.

Using data browser to look at the data and data editor to change data

```
. replace hrs2 = 1 in 7
```

If you are not sure you want to keep your changes, use “preserve” command in the beginning to save a copy of the dataset in Stata memory; restore in the end will return the data to that saved version.

Get summary statistics:

```
. sum  hrs1 hrs2
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        hrs1 |      1729    41.77675    14.62304          1         89
        hrs2 |        50       34.88    15.55719          1         60

. sum  hrs1 hrs2, detail
              number of hours worked last week
-------------------------------------------------------------
      Percentiles      Smallest
 1%            6              1
 5%           16              2
10%           21              2       Obs                1729
25%           36              2       Sum of Wgt.        1729

50%           40                      Mean           41.77675
                        Largest       Std. Dev.      14.62304
75%           50             89
90%           60             89       Variance       213.8332
95%           68             89       Skewness       .2834814
99%           88             89       Kurtosis       4.310339

             number of hours usually work a week
-------------------------------------------------------------
      Percentiles      Smallest
 1%            1              1
 5%            6              3
10%            9              6       Obs                  50
25%           24              7       Sum of Wgt.          50

50%           40                      Mean              34.88
                        Largest       Std. Dev.      15.55719
75%           43             57
90%           53             60       Variance       242.0261
95%           60             60       Skewness      -.5207683
99%           60             60       Kurtosis       2.545694
```

List values of selected variables for each observation:

```
. list  wrkstat hrs1 wrkslf
      +----------------------------+
      |  wrkstat   hrs1     wrkslf |
      |----------------------------|
   1. |  working     40    someone |
   2. |  working     72    someone |
   3. |  working     40    someone |
   4. |  working     60    someone |
   5. |  working     40    someone |
      |----------------------------|
   6. |  working     42    someone |
   7. |  retired      .    someone |
   8. |  keeping      .    someone |
  --Break--
r(1);
```

Same but for observations 100-200:

```
. list  wrkstat hrs1 wrkslf in 100/200
     +----------------------------+
     |  wrkstat   hrs1     wrkslf |
     |----------------------------|
100. |  working     40    someone |
101. |   school      .    someone |
102. |  working     40    someone |
103. |  working     51    someone |
104. |  working     40    someone |
     |----------------------------|
105. |  unempl,      .    someone |
106. |   school      .    someone |
107. |  retired      .    someone |
--Break--
r(1);
```

Get codebook info:

```
. codebook wrkstat
--------------------------------------------------------------------------------
wrkstat
labor frce status
--------------------------------------------------------------------------------
                  type:  numeric (byte)
                 label:  wrkstat

                 range:  [1,8]                        units:  1
         unique values:  8                        missing .:  0/2765

            tabulation:  Freq.   Numeric  Label
                          1432         1  working fulltime
                           312         2  working parttime
                            52         3  temp not working
                           121         4  unempl, laid off
                           414         5  retired
                            78         6  school
                           268         7  keeping house
                            88         8  other
```

Frequency tables -- tabulate command:

```
. tab  wrkstat
      labor frce |
          status |      Freq.     Percent        Cum.
-----------------+-----------------------------------
working fulltime |      1,432       51.79       51.79
working parttime |        312       11.28       63.07
temp not working |         52        1.88       64.95
unempl, laid off |        121        4.38       69.33
         retired |        414       14.97       84.30
          school |         78        2.82       87.12
   keeping house |        268        9.69       96.82
           other |         88        3.18      100.00
-----------------+-----------------------------------
           Total |      2,765      100.00
```

Including missing values:

```
. tab  wrkslf, miss
r self-emp or |
    works for |
     somebody |      Freq.     Percent        Cum.
--------------+-----------------------------------
self-employed |        307       11.10       11.10
 someone else |      2,362       85.42       96.53
            . |         96        3.47      100.00
--------------+-----------------------------------
        Total |      2,765      100.00
```

Note that missing values are in fact stored as very large numbers -- should be careful when doing data management.

In addition to missing values specified as ., they can be stored as .a, .b, .c, etc., in order to differentiate between different types of missing values.

To suppress labels:

```
. tab  wrkslf, miss nolabel
 r self-emp |
   or works |
        for |
   somebody |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |        307       11.10       11.10
          2 |      2,362       85.42       96.53
          . |         96        3.47      100.00
------------+-----------------------------------
      Total |      2,765      100.00
```

Cross-tabulation:

```
. tab  wrkslf wrkgovt
r self-emp or |    govt or private
    works for |       employee
     somebody | governmen    private |     Total
--------------+----------------------+----------
self-employed |        13        271 |       284
 someone else |       441      1,914 |     2,355
--------------+----------------------+----------
        Total |       454      2,185 |     2,639
```

With row percentages:

```
. tab  wrkslf wrkgovt, row
+----------------+
| Key            |
|----------------|
|   frequency    |
| row percentage |
+----------------+
r self-emp or |    govt or private
    works for |       employee
     somebody | governmen    private |     Total
--------------+----------------------+----------
self-employed |        13        271 |       284
              |      4.58      95.42 |    100.00
--------------+----------------------+----------
 someone else |       441      1,914 |     2,355
              |     18.73      81.27 |    100.00
--------------+----------------------+----------
        Total |       454      2,185 |     2,639
              |     17.20      82.80 |    100.00
```

With all three types of percentages and a chi-square test:

```
. tab  wrkslf wrkgovt, row col cell chi2
+-------------------+
| Key               |
|-------------------|
|     frequency     |
|  row percentage   |
| column percentage |
|  cell percentage  |
+-------------------+
r self-emp or |    govt or private
    works for |       employee
     somebody | governmen    private |     Total
--------------+----------------------+----------
self-employed |        13        271 |       284
              |      4.58      95.42 |    100.00
              |      2.86      12.40 |     10.76
              |      0.49      10.27 |     10.76
--------------+----------------------+----------
 someone else |       441      1,914 |     2,355
              |     18.73      81.27 |    100.00
              |     97.14      87.60 |     89.24
              |     16.71      72.53 |     89.24
--------------+----------------------+----------
        Total |       454      2,185 |     2,639
              |     17.20      82.80 |    100.00
              |    100.00     100.00 |    100.00
              |     17.20      82.80 |    100.00

          Pearson chi2(1) =  35.6181   Pr = 0.000
```

Multiple univariate tables of frequencies are obtained using tab1 command:

```
. tab1  wrkslf wrkgovt

-> tabulation of wrkslf

r self-emp or |
    works for |
     somebody |      Freq.     Percent        Cum.
--------------+-----------------------------------
self-employed |        307       11.50       11.50
 someone else |      2,362       88.50      100.00
--------------+-----------------------------------
        Total |      2,669      100.00

-> tabulation of wrkgovt

    govt or |
    private |
   employee |      Freq.     Percent        Cum.
------------+-----------------------------------
 government |        454       17.19       17.19
    private |      2,187       82.81      100.00
------------+-----------------------------------
      Total |      2,641      100.00
```

*Using conditions
*Can use:
< less
> more
== equal
<= less or equal
>= more or equal
~= not equal
Can connect them with & (and) and | (or). Can also use parentheses to combine conditions.

```
. codebook marital
-----------------------------------------------------------------------------
marital
marital status
-----------------------------------------------------------------------------
                  type:  numeric (byte)
                 label:  marital

                 range:  [1,5]                        units:  1
         unique values:  5                        missing .:  0/2765

            tabulation:  Freq.   Numeric  Label
                          1269         1  married
                           247         2  widowed
                           445         3  divorced
                            96         4  separated
                           708         5  never married

. sum hrs1 if  wrkslf==1 &  marital==5

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        hrs1 |        35    38.48571    20.74406          8         89
. sum hrs1 if  wrkslf==1 &  marital>1

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        hrs1 |        96    39.48958    20.22609          5         89

. sum hrs1 if  wrkslf==1 &  marital>1 & marital<=5

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        hrs1 |        96    39.48958    20.22609          5         89

. sum hrs1 if  wrkslf==1 &  marital>1 & marital~=.

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        hrs1 |        96    39.48958    20.22609          5         89

. sum hrs1 if  wrkslf==1 &  (marital==1 | marital==2)

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        hrs1 |       137    41.46715    18.42515          3         89
```

Help and installation

Help in Stata – help and search commands:

```
. help tabulate

. search logistic

Keyword search

        Keywords:  logistic
          Search:  (1) Official help files, FAQs, Examples, SJs, and STBs

Search of official help files, FAQs, Examples, SJs, and STBs


[U]     Chapter 26  . . . . . . . . . .  Overview of Stata estimation commands
        (help estcom)

[R]     clogit  . . . . . . .  Conditional (fixed-effects) logistic regression
        (help clogit)

[R]     cloglog . . . . . . . . . . . . . . . . Complementary log-log regression
        (help cloglog)

[R]     constraint  . . . . . . . . . . . . . . .  Define and list constraints
        (help constraint)

[R]     fracpoly  . . . . . . . . . . . . . . Fractional polynomial regression
        (help fracpoly)

[R]     glogit  . . . . . . . . . . . . . .  Logit and probit for grouped data
        (help glogit)

[R]     logistic  . . . . . . . . . Logistic regression, reporting odds ratios
        (help logistic)

[R]     logistic postestimation . . . . . .  Postestimation tools for logistic
        (help logistic postestimation)

[R]     logit . . . . . . . . . .  logistic regression, reporting coefficients
        (help logit)

[R]     logit postestimation  . . . . . . . . . Postestimation tools for logit
        (help logit postestimation)

[R]     mfp . . . . . . . . . . . Multivariable fractional polynomial models
        (help mfp)

[R]     mlogit  . . . . . . . . . Multinomial (polytomous) logistic regression
        (help mlogit)

[R]     nlogit  . . . . . . . . . . . . . . . . . . . .  Nested logit regression
        (help nlogit)

[R]     ologit  . . . . . . . . . . . . . . . . . .  Ordered logistic regression
        (help ologit)
--Break--
r(1);
```

You can also use “net search” command that will search Stata resources online in addition to local resources:

```
. net search spost
(contacting http://www.stata.com)

14 packages found (Stata Journal and STB listed first)
------------------------------------------------------

st0094 from http://www.stata-journal.com/software/sj5-4
    SJ5-4 st0094.  Confidence intervals for predicted outcomes... / Confidence
    intervals for predicted outcomes in regression / models for categorical
    outcomes / by Jun Xu and J. Scott Long, Indiana University / Support:
    spostsup@indiana.edu / After installation, type help prvalue and prgen

spost9_ado from http://www.indiana.edu/~jslsoc/stata
    spost9_ado | Stata 9-13 commands for the post-estimation interpretation /
    Distribution-date: 05Aug2013 / of regression models. Use package
    spostado.pkg for Stata 8. / Based on Long & Freese - Regression Models for
    Categorical Dependent / Variables Using Stata. Second Edition. / Support

spost9_do from http://www.indiana.edu/~jslsoc/stata
    spost9_do | SPost9 example do files. / Distribution-date: 27Jul2005 / Long
    & Freese 2005 Regression for Categorical Dependent Variables / using
    Stata. Second Edition. Stata Version 9. / Support
    www.indiana.edu/~jslsoc/spost.htm / Scott Long & Jeremy Freese

spostado from http://www.indiana.edu/~jslsoc/stata
    spostado: Stata 8 commands for the post-estimation interpretation of /
    regression models. Based on Long's Regression Models for Categorical / and
    Limited Dependent Variables. / Support: www.indiana.edu/~jslsoc/spost.htm
    / Scott Long & Jeremy Freese (spostsup@indiana.edu)

spostrm7 from http://www.indiana.edu/~jslsoc/stata
    spostrm7: Stata 7 do & data files to reproduce RM4CLDVs results using
    SPost. / Files correspond to chapters of Long: Regression Models for
    Categorical / & Limited Dependent Variables. / Support:
    www.indiana.edu/~jslsoc/spost.htm / Scott Long & Jeremy Freese

spostst8 from http://www.indiana.edu/~jslsoc/stata
    spostst8: Stata 8 do & data files to reproduce RM4STATA results using
    SPost. / Files correspond to chapters of Long & Freese-Regression Models
    for Categorical / Dependent Variables Using Stata (Stata 8 Revised
    Edition). / Support: www.indiana.edu/~jslsoc/spost.htm / Scott Long &

test9_legacy from http://www.indiana.edu/~jslsoc/stata
    test9_legacy | SPost9 commands not included in test13_ado. / Support
    www.indiana.edu/~jslsoc/spost.htm / Scott Long & Jeremy Freese
    (jslong@indiana.edu)

difd from http://fmwww.bc.edu/RePEc/bocode/d
    'DIFD': module to evaluate test items for differential item functioning
    (DIF) / DIF detection is a first step in assessing bias in test items.  /
    difd detects DIF in test items between groups, conditional on / the trait
    that the test is measuring, using logistic / regression.  The criteria for

difwithpar from http://fmwww.bc.edu/RePEc/bocode/d
    'DIFWITHPAR': module for detection of and adjustment for differential item
    functioning (DIF) / Identifies differential item functioning, creates /
    dummy/virtual items to be used to adjust ability (trait) / estimates in
    PARSCALE, writes the code and data file needed to / process the updated

grcompare from http://fmwww.bc.edu/RePEc/bocode/g
    'GRCOMPARE': module to make group comparisons in binary regression models
    / This is a Stata module to make group comparisons in binary / regression
    models using alternative measures, including gradip: / average difference
    in predicted probabilities over a range; / grdiame:difference in group

prepar from http://fmwww.bc.edu/RePEc/bocode/p
    'PREPAR': module to write code and data file needed to process variables
    in PARSCALE / This program writes the input code and data file for
    PARSCALE, / which is a real time-saver if you aren't familiar with /
    PARSCALE.  / KW: PARSCALE / Requires: Stata version 8.2, PARSCALE and

runparscale from http://fmwww.bc.edu/RePEc/bocode/r
    'RUNPARSCALE': module to run PARSCALE from Stata / Builds a PARSCALE data
    file and command file, executes the / command file, displays the PARSCALE
    log file in Stata results / window, and merges the PARSCALE theta
    estimates and their / standard errors back into the original data set.  /

scottlong from http://www.indiana.edu/~jslsoc/stata
    scottlong | Temporary files... / Distribution-date: 11Aug2013

test13_ado from http://www.indiana.edu/~jslsoc/stata
    test13_ado | FOR TESTING ONLY. Report problems to (jslong@indiana.edu) /
    Scott Long & Jeremy Freese (jslong@indiana.edu)


1 reference found in tables of contents
---------------------------------------

http://www.indiana.edu/~jslsoc/stata/
    SPost:  Interpreting regression models. Scott Long & Jeremy Freese / WF:
    Workflow of data analysis. Scott Long / Teaching:  Teaching files. Scott
    Long / Research:  Research examples and commands. Scott Long / Support:
    www.indiana.edu/~jslsoc/spost.htm / www.indiana.edu/~jslsoc/workflow.htm /
```

Note that some of the things we found are user-written programs that implement user-written commands that can be quite helpful; to install, click on the package and click to install, or type

```
. net install spost9_ado, from(http://www.indiana.edu/~jslsoc/stata)
```

Also, do not forget to do Stata updates on a regular basis, including updating all installed programs (ado files).

```
. update all
```

Using do-files

Open do-file editor, create and save your file (.do).

You can execute that file from the do-file editor or using the command line:

```
. do mydofile.do
```

But be careful to specify the location of your file or make sure it is in the working directory specified in the last "cd" command.

It is often convenient to create and edit do-files in another text editor – I prefer TextPad: http://www.textpad.com

You can also keep the log of just the commands:
cmdlog using filename
Then you can use that log as a do-file.

And if you want to save all commands you've done so far, just right click on the command window and select "Save Review Contents." If some of your commands had errors (highlighted in red), you can right click on each of them and delete them from the Review window before copying your commands.

You should keep a do-file with all your data management steps, and in most cases it's a good idea to have one with your analysis steps as well – that way, if you make a mistake, you can easily rerun things. To have that, we can save all the commands that we did interactively into a do-file, or we can right away write a do-file and then execute it.

Graphics in Stata

```
. scatter hrs1 prestg80

. graph matrix hrs1 hrs2 prestg80 sphrs1 sppres80

. histogram hrs1
(bin=32, start=1, width=2.75)
```

We can save graphs for future use:

```
graph save mygraph.gph
```

To then display that graph, we type:

```
graph use mygraph.gph
```

You can also export them into different, non-Stata formats:

```
. graph export mygraph.wmf
```

The output format is determined by the suffix of the file name (see help graph export):

```
                    Implied
    suffix          option          Output format
    ----------------------------------------------------------------------
    .ps             as(ps)          PostScript
    .eps            as(eps)         EPS (Encapsulated PostScript)
    .wmf            as(wmf)         Windows Metafile
    .emf            as(emf)         Windows Enhanced Metafile
    .pdf            as(pdf)         PDF
    .png            as(png)         PNG (Portable Network Graphics)
    .tif            as(tif)         TIFF
```

Or you can just copy graphs and paste them into your word processor

To further explore the options available for graphics, use:

```
. help graph
```

Stata versions and settings

Be aware that there are different versions of Stata: Variable number limits are 2,047 for Stata/IC, and 99 for Small Stata. When using Stata/MP and Stata/SE, the maximum number of variables in your dataset can be changed by using “set maxvar” command. The default value of maxvar is 5,000 for Stata/MP and Stata/SE.

Here, we are using Stata/IC; the version on the apps server is Stata/SE.

Besides set maxvar, to make it easier for you to work with Stata, you can change some of other default settings using “set” command. Moreover, if you want to execute certain settings commands automatically every time you start Stata, you can put these commands into a file named profile.do, which is a do-file that Stata executes every time that it starts. Once started, Stata looks for this file and executes every command in the file before you begin entering commands.

An example profile.do file is:

```
        set logtype text
        set more off
```

Stata looks for profile.do first in the directory where Stata is installed, then in the current directory, then along your path, then in your home directory as defined by Windows’ USERPROFILE, and finally along the ado-path. It is recommended that you put profile.do in the default working directory that you set when you installed Stata. If you are not sure what your default working directory is, type pwd in the Command window immediately after starting Stata (without running a cd command). If you want to know where other Stata system directories are located, use sysdir:

```
. help sysdir

. sysdir
   STATA:  C:\Program Files (x86)\Stata13\
    BASE:  C:\Program Files (x86)\Stata13\ado\base\
    SITE:  C:\Program Files (x86)\Stata13\ado\site\
    PLUS:  c:\ado\plus\
PERSONAL:  c:\ado\personal\
OLDPLACE:  c:\ado\

. pwd
C:\Documents and Settings\sarkisin\My Documents
```

Some Stata settings can be made “permanent” instead of placing them into profile.do. For example, if you want Stata to never pause output with a --more-- in the Results window, you could type

```
. set more off, permanently
```

Another useful set command that you will likely encounter once you start running statistical models on large data is “set matsize” (can also be used with ”permanently” option). set matsize sets the maximum number of variables that can be included in any of Stata's estimation commands.

For Stata/IC, the initial value is 400, but it may be changed upward or downward. The upper limit is 800. For Stata/MP and Stata/SE, the default value is 400, but it may be changed upward or downward. The upper limit is 11,000. This command may not be used with Small Stata; matsize is permanently frozen at 100.

Another useful set command has to do with graphs.

```
.  set scheme schemename [, permanently]
```

set scheme allows you to set the graphics scheme to be used. The default setting is s2color. You can use point and click to explore graphics schemes.

Basics of Data Management in Stata

```
*To sort all variables in the dataset, use order command to specify a certain
order and aorder command to sort alphabetically.
. order wrkstat marital sibs childs
. aorder

*To keep only a subselection of variables in the dataset, use drop and keep
. drop spwrksta- spind80
. keep wrkstat marital sibs childs

*Can also use if and in qualifiers with drop and keep commands:
. drop if wrkstat==2
. keep in 1/100

*to return to the original dataset without saving the modified one:
. use "C:\Documents and Settings\sarkisin\My Documents\gss2002.dta", clear

*Creating new variables
. gen hrs40=.
(2765 missing values generated)
. replace hrs40 = 0 if hrs1<40
(490 real changes made)
. replace hrs40 = 1 if hrs1>=40 & hrs1~=.
(1239 real changes made)

. tab hrs40, missing
      hrs40 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |        490       17.72       17.72
          1 |      1,239       44.81       62.53
          . |      1,036       37.47      100.00
------------+-----------------------------------
      Total |      2,765      100.00

*label the variable
. label variable hrs40 "R works 40 hours a week or more"
*label its values: two steps, first define a set of labels
. label define hrs40label 0 "less than 40" 1 "40 or more"
*next, apply this set to the new variable
. label values hrs40 hrs40label

. tab hrs40, missing
  R works 40 |
hours a week |
     or more |      Freq.     Percent        Cum.
-------------+-----------------------------------
less than 40 |        490       17.72       17.72
```

```
  40 or more |      1,239       44.81       62.53
           . |      1,036       37.47      100.00
-------------+-----------------------------------
       Total |      2,765      100.00

. codebook hrs40
------------------------------------------------------------------------
hrs40                                          R works 40 hours a week or more
------------------------------------------------------------------------
                 type:  numeric (float)
                label:  hrs40label

                range:  [0,1]                        units:  1
        unique values:  2                        missing .:  1036/2765

           tabulation:  Freq.   Numeric  Label
                          490         0  less than 40
                         1239         1  40 or more
                         1036         .

*To rename a variable, use the rename command:
.rename hrs40 hours40

*generate a dummy variable indicating married respondents
. codebook marital
---------------------------------------------------------------------------------
marital                                                            marital status
---------------------------------------------------------------------------------
                 type:  numeric (byte)
                label:  marital

                range:  [1,5]                        units:  1
        unique values:  5                        missing .:  0/2765

           tabulation:  Freq.   Numeric  Label
                         1269         1  married
                          247         2  widowed
                          445         3  divorced
                           96         4  separated
                          708         5  never married
. gen married=(marital==1)
. tab married
    married |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      1,496       54.10       54.10
          1 |      1,269       45.90      100.00
------------+-----------------------------------
      Total |      2,765      100.00
. replace married=. if marital==.
(0 real changes made)
*another way to generate such a dummy variable
. gen married2=0
. replace married2=1 if marital==1
(1269 real changes made)

. tab married2
   married2 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      1,496       54.10       54.10
          1 |      1,269       45.90      100.00
```

```
------------+-----------------------------------
      Total |      2,765      100.00

*generate age squared variable
. gen age2=age^2
(14 missing values generated)

*generate square root of age variable
. gen age2sqrt=sqrt(age2)
(14 missing values generated)

*generate log of age variable
. gen agelg=log(age)
(14 missing values generated)

*generate marital status with 3 categories
. recode marital (1=1) (2=2) (3=2) (4=2) (5=3), gen(married3)
(1249 differences between marital and married3)

*or, we can do the same but a bit shorter:
. recode marital (1=1) (2/4=2) (5=3), gen(marital3)
(1249 differences between marital and marital3)

. tab marital3
  RECODE of |
    marital |
   (marital |
    status) |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |      1,269       45.90       45.90
          2 |        788       28.50       74.39
          3 |        708       25.61      100.00
------------+-----------------------------------
      Total |      2,765      100.00

*label the new variable
. label variable marital3 "marital status 3 categories"
. tab marital3
    marital |
   status 3 |
 categories |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |      1,269       45.90       45.90
          2 |        788       28.50       74.39
          3 |        708       25.61      100.00
------------+-----------------------------------
      Total |      2,765      100.00

*label values of the new variable
. label define marital3label 1"married" 2 "previously married" 3 "never married"
. label values marital3 marital3label

*check the results
. codebook marital3
------------------------------------------------------------------------------
marital3                                           marital status 3 categories
------------------------------------------------------------------------------
                  type:  numeric (byte)
                 label:  marital3label
                 range:  [1,3]                        units:  1
         unique values:  3                        missing .:  0/2765

            tabulation:  Freq.   Numeric  Label
                          1269         1  married
                           788         2  previously married
                           708         3  never married

*Saving the dataset with newly created variable
. save "C:\Documents and Settings\My Documents\gss2002changed.dta"
file C:\Documents and Settings\My Documents\gss2002changed.dta saved
```

You should keep a do-file with all your data management steps, and in most cases it's a good idea to have one with your analysis steps as well – that way, if you make a mistake, you can easily rerun things. To have that, we can save all the commands that we did interactively into a do-file, or we can right away write a do-file and then execute it.

Note that if you are opening a Stata log file in a Word processor, you should change the font to a fixed width font, such as Courier New (otherwise the output looks misaligned). Courier New 10 point usually works the best.

```
*exiting Stata
. exit, clear
```