2 A brief description of Stata

Stata is a statistical package for managing, analyzing, and graphing data.

Stata is available for a variety of platforms. Stata may be used either as a point-and-click application or as a command-driven package.

Stata's GUI provides an easy interface for those new to Stata and for experienced Stata users who wish to execute a command that they seldom use.

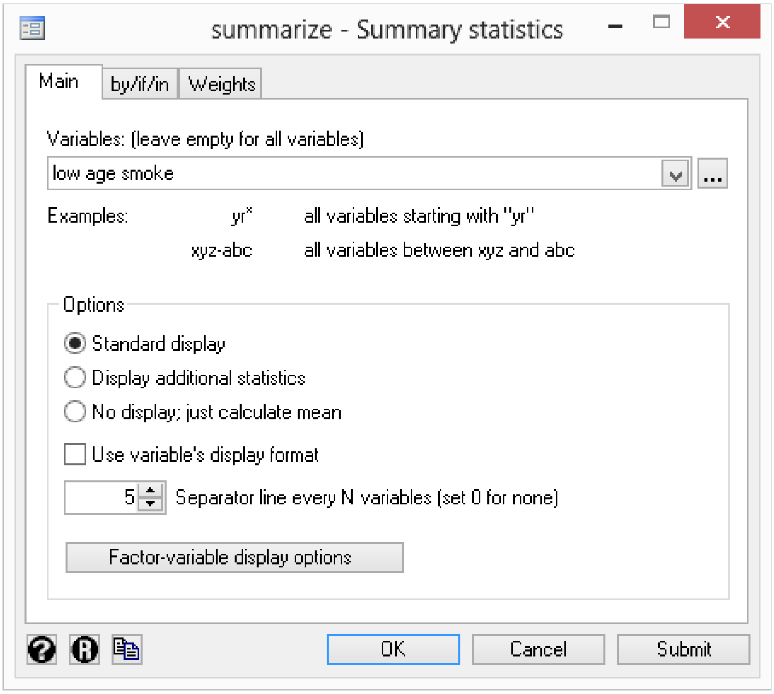
The command language provides a fast way to communicate with Stata and to communicate more complex ideas.

Here is an extract of a Stata session using the GUI:

(Throughout the Stata manuals, we will refer to various datasets. These datasets are all available from http://www.stata-press.com/data/r14/. For easy access to them within Stata, type `webuse` *dataset_name*, or select **File > Example datasets...** and click on *Stata 14 manual datasets*.)

```
. webuse lbw
(Hosmer & Lemeshow data)
```

We select **Data > Describe data > Summary statistics** and choose to summarize variables `low`, `age`, and `smoke`, whose names we obtained from the Variables window. We click on **OK**.


summarize - Summary statistics
Main
by/if/in
Weights

Variables: (leave empty for all variables)
low age smoke

Examples: yr* all variables starting with "yr"
xyz-abc all variables between xyz and abc

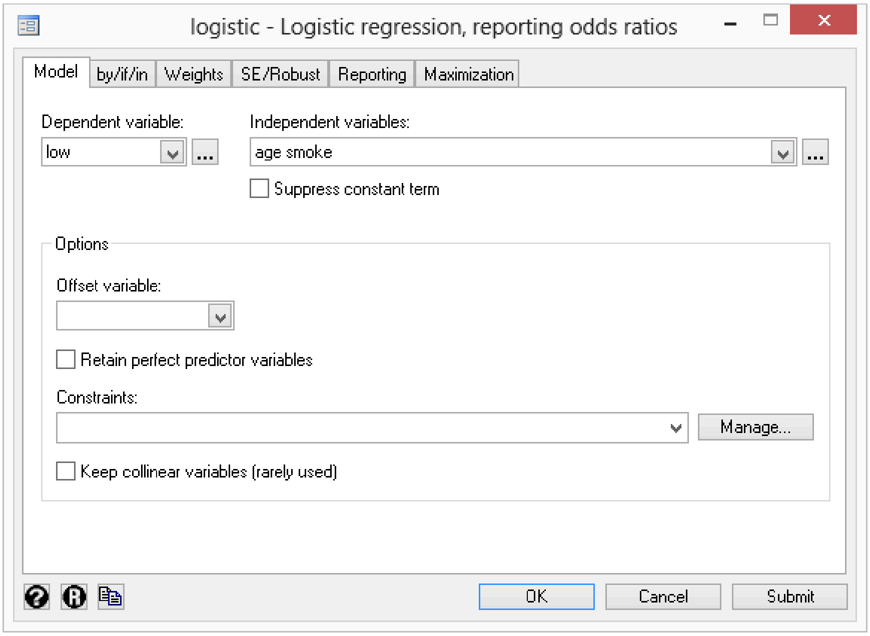
Options
● Standard display
○ Display additional statistics
○ No display; just calculate mean
□ Use variable's display format
5 Separator line every N variables (set 0 for none)

Factor-variable display options
OK
Cancel
Submit

```
. summarize low age smoke

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
         low |        189    .3121693    .4646093          0          1
         age |        189     23.2381    5.298678         14         45
       smoke |        189    .3915344    .4893898          0          1
```

Stata shows us the command that we could have typed in command mode—`summarize low age smoke`—before displaying the results of our request.

Next we fit a logistic regression model of `low` on `age` and `smoke`. We select **Statistics > Binary outcomes > Logistic regression (reporting odds ratios)**, fill in the fields, and click on **OK**.



logistic - Logistic regression, reporting odds ratios

Dependent variable: low

Independent variables: age smoke

□ Suppress constant term

Options

Offset variable:

□ Retain perfect predictor variables

Constraints:

□ Keep collinear variables (rarely used)

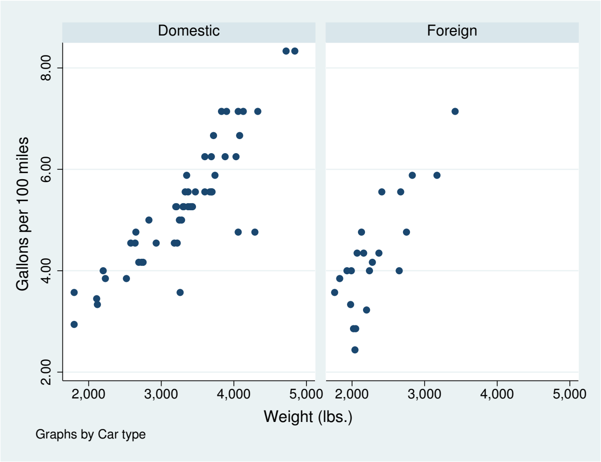
```
. logistic low age smoke

Logistic regression                             Number of obs     =        189
                                                LR chi2(2)        =       7.40
                                                Prob > chi2       =     0.0248
Log likelihood = -113.63815                     Pseudo R2         =     0.0315

------------------------------------------------------------------------------
         low | Odds Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   .9514394   .0304194    -1.56   0.119     .8936482    1.012968
       smoke |   1.997405    .642777     2.15   0.032     1.063027    3.753081
       _cons |   1.062798   .8048781     0.08   0.936     .2408901    4.689025
------------------------------------------------------------------------------
```

Here is an extract of a Stata session using the command language:

```
. use http://www.stata-press.com/data/r14/auto
(1978 Automobile Data)

. summarize mpg weight

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
         mpg |         74     21.2973    5.785503         12         41
      weight |         74    3019.459    777.1936       1760       4840
```

The user typed `summarize mpg weight` and Stata responded with a table of summary statistics. Other commands would produce different results:

```
. generate gp100m = 100/mpg

. label var gp100m "Gallons per 100 miles"

. format gp100m %5.2f

. correlate gp100m weight
(obs=74)

             |   gp100m   weight
-------------+------------------
      gp100m |   1.0000
      weight |   0.8544   1.0000

. regress gp100m weight gear_ratio

      Source |       SS           df       MS      Number of obs   =        74
-------------+----------------------------------   F(2, 71)        =     96.65
       Model |  87.4543721         2  43.7271861   Prob > F        =    0.0000
    Residual |  32.1218886        71  .452420967   R-squared       =    0.7314
-------------+----------------------------------   Adj R-squared   =    0.7238
       Total |  119.576261        73  1.63803097   Root MSE        =    .67262

------------------------------------------------------------------------------
      gp100m |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |   .0014769   .0001556     9.49   0.000     .0011665    .0017872
  gear_ratio |   .1566091   .2651131     0.59   0.557    -.3720115    .6852297
       _cons |   .0878243   1.198434     0.07   0.942    -2.301786    2.477435
------------------------------------------------------------------------------

. scatter gp100m weight, by(foreign)
```



The user-interface model is type a little, get a little, etc., so that the user is always in control.

Stata's model for a dataset is that of a table—the rows are the observations and the columns are the variables:

```
. list mpg weight gp100m in 1/10

     +-------------------------+
     | mpg   weight   gp100m |
     |-------------------------|
  1. |  22    2,930     4.55 |
  2. |  17    3,350     5.88 |
  3. |  22    2,640     4.55 |
  4. |  20    3,250     5.00 |
  5. |  15    4,080     6.67 |
     |-------------------------|
  6. |  18    3,670     5.56 |
  7. |  26    2,230     3.85 |
  8. |  20    3,280     5.00 |
  9. |  16    3,880     6.25 |
 10. |  19    3,400     5.26 |
     +-------------------------+
```

Observations are numbered; variables are named.

Stata is fast. That speed is due partly to careful programming, and partly because Stata keeps the data in memory. Stata's file model is that of a word processor: a dataset may exist on disk, but the dataset in memory is a copy. Datasets are loaded into memory, where they are worked on, analyzed, changed, and then perhaps stored back on disk.

Working on a copy of the data in memory makes Stata safe for interactive use. The only way to harm the permanent copy of your data on disk is if you explicitly save over it.

Having the data in memory means that the dataset size is limited by the amount of computer memory. Stata stores the data in memory in an efficient format—you will be surprised how much data can fit. Nevertheless, if you work with extremely large datasets, you may run into memory constraints. You will want to learn how to store your data as efficiently as possible; see [D] **compress**.