Stata Test

We'll look at the dataset they seem to use for all examples in the Stata manual:

```
. sysuse auto
. summ mpg disp
```

```
(1978 Automobile Data)

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
         mpg |        74     21.2973    5.785503         12         41
displacement |        74    197.2973    91.83722         79        425
```

The maximum price is 15906.

Let's generate new variables and fit a regression model

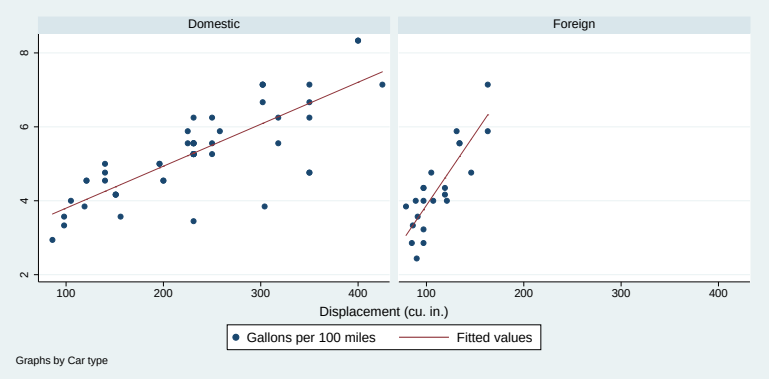
```
. gen gp100m = 100/mpg
. gen for_disp = foreign*displacement
. label var gp100m "Gallons per 100 miles"
. regr gp100m foreign disp for_disp
```

```
      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  3,    70) =   52.98
       Model |  83.0134814     3  27.6711605           Prob > F      =  0.0000
    Residual |  36.5627794    70   .52232542           R-squared     =  0.6942
-------------+------------------------------           Adj R-squared =  0.6811
       Total |  119.576261    73  1.63803097           Root MSE      =  .72272

------------------------------------------------------------------------------
      gp100m |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     foreign |  -2.672303   .7796252    -3.43   0.001    -4.227216   -1.117389
displacement |    .011352   .0011869     9.56   0.000     .0089847    .0137192
    for_disp |   .0274881   .0064489     4.26   0.000     .0146263      .04035
       _cons |   2.665073   .2949491     9.04   0.000     2.076815     3.25333
------------------------------------------------------------------------------
```

And here is a nice plot of the results

```
. predict g_hat
. twoway (scatter gp100m disp) (line g_hat disp, sort), by(foreign)
```



Using Mata

Let's try some matrix stuff, in case it Matas to you.

```
. mata
. pr = st_data(.,("price"))
. printf("Price in thousands")
. pr' / 1000
```

```
Price in thousands
            1        2        3        4        5        6        7        8
    +-------------------------------------------------------------------------+
  1 |   4.099    4.749    3.799    4.816    7.827    5.788    4.453    5.189  |
    +-------------------------------------------------------------------------+
            9       10       11       12       13       14       15       16
    -------------------------------------------------------------------------
  1    10.372    4.082   11.385     14.5   15.906    3.299    5.705    4.504
    -------------------------------------------------------------------------
           17       18       19       20       21       22       23       24
    -------------------------------------------------------------------------
  1     5.104    3.667    3.955    3.984     4.01    5.886    6.342    4.389
    -------------------------------------------------------------------------
           25       26       27       28       29       30       31       32
    -------------------------------------------------------------------------
  1     4.187   11.497   13.594   13.466    3.829    5.379    6.165    4.516
    -------------------------------------------------------------------------
           33       34       35       36       37       38       39       40
    -------------------------------------------------------------------------
  1     6.303    3.291    8.814    5.172    4.733     4.89    4.181    4.195
    -------------------------------------------------------------------------
           41       42       43       44       45       46       47       48
    -------------------------------------------------------------------------
  1    10.371    4.647    4.425    4.482    6.486     4.06    5.798    4.934
    -------------------------------------------------------------------------
           49       50       51       52       53       54       55       56
    -------------------------------------------------------------------------
  1     5.222    4.723    4.424    4.172     9.69    6.295    9.735    6.229
    -------------------------------------------------------------------------
           57       58       59       60       61       62       63       64
    -------------------------------------------------------------------------
  1     4.589    5.079    8.129    4.296    5.799    4.499    3.995    12.99
    -------------------------------------------------------------------------
           65       66       67       68       69       70       71       72
    -------------------------------------------------------------------------
  1     3.895    3.798    5.899    3.748    5.719     7.14    5.397    4.697
    -------------------------------------------------------------------------
           73       74
    -------------------+
  1      6.85   11.995  |
    -------------------+
```

The variable pr has 74 elements and its maximum value is 15906, just as we computed before.

```
. end
. /* Now we're back in interactive mode */
. disp price/1000
```

```
4.099
```

Testing some post-processing issues

Want to make sure that very long strings and programs are handled right.

```
. disp "This is an extremely extremely extremely extremely extremely extremely extremely extremely extre
. disp " ...followed by a short line."
. /* I have a silly program */
. prog stuff
.   disp "This is a test"
.   disp "This is another line"
.   disp "If I'm lucky, the program code will not show up in the output listing"
. end
. stuff
```

```
This is an extremely extremely extremely extremely extremely extremely extremel
> y extremely extremely extremely extremely extremely extremely extremely extre
> mely extremely extremely long line

 ...followed by a short line.

This is a test
This is another line
If I'm lucky, the program code will not show up in the output listing
```