Handout for SUGUK 2005 invited lecture: *A little bit of Stata programming goes a long way...*

Christopher F Baum
Boston College
baum@bc.edu
http://ideas.repec.org/e/pba1.html

May 2, 2005

Abstract

This tutorial will discuss a number of elementary Stata programming constructs and discuss how they may be used to automate and robustify common data manipulation, estimation and graphics tasks. Those used to the syntax of other statistical packages or programming languages must adopt a different mindset when working with Stata to take full advantage of its capabilities. Some of Stata’s most useful commands for handling repetitive tasks: `forvalues, foreach, egen, local, scalar, estimates` and `matrix` are commonly underutilized by users unacquainted with their power and ease of use. While relatively few users may develop ado–files for circulation to the user community, nearly all will benefit from learning the rudiments of use of the `program, syntax and return` statements when they are faced with the need to perform repetitive analyses. Worked examples making use of these commands will be presented and discussed in the tutorial.

Exhibit 1

```
. local country US UK DE FR

. local ctycode 111 112 136 134

. display "‘country’"
US UK DE FR

. display "‘ctycode’"
111 112 136 134
```

Exhibit 2

```
. local count 0

. local country US UK DE FR

. foreach c of local country {
  2.    local count ‘count’+1
  3.    display "Country ‘count’ : ‘c’"
  4.    }
Country 0+1 : US
Country 0+1+1 : UK
Country 0+1+1+1 : DE
Country 0+1+1+1+1 : FR
```

Exhibit 3

```
. local count 0

. local country US UK DE FR

. foreach c of local country {
  2.    local count = ‘count’+1
  3.    display "Country ‘count’ : ‘c’"
  4.    }
Country 1 : US
Country 2 : UK
Country 3 : DE
Country 4 : FR
```

Exhibit 4

```
. local count 0

. local country US UK DE FR

. foreach c of local country {
  2.    local count = ‘count’+1
  3.    local newlist "‘newlist’ ‘count’ ‘c’"
  4.    }

. display "‘newlist’"
 1 US 2 UK 3 DE 4 FR
```

Exhibit 5

```
. local country US UK DE FR

. foreach c of local country {
  2.        tsline gdp if cty=="‘c’", title("GDP for ‘c’")
  3.        }
```

Exhibit 6

```
. local country US UK DE FR

. foreach c of local country {
  2.        tsline gdp if cty=="‘c’", title("GDP for ‘c’") ///
>           nodraw name(‘c’,replace)
  3.        }

. graph combine ‘country’, ti("Gross Domestic Product, 1971Q1-1995Q4")
```

Exhibit 7

```
. local country US UK DE FR

. local wds: word count ‘country’

. display "There are ‘wds’ countries:"
There are 4 countries:

. forvalues i = 1/‘wds’ {
  2.         local wd: word ‘i’ of ‘country’
  3.         display "Country ‘i’ is ‘wd’"
  4.         }
Country 1 is US
Country 2 is UK
Country 3 is DE
Country 4 is FR
```

Exhibit 8

```
. scalar root2 = sqrt(2.0)

. gen rootGDP = gdp*root2
```

Exhibit 9

```
. forvalues i = 1/4 {
  2.         gen double lngdp‘i’ = log(gdp‘i’)
  3.         summ lngdp‘i’
  4.         }
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     lngdp1 |       400    7.931661      .59451   5.794211   8.768936
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     lngdp2 |       400    7.942132    .5828793   4.892062   8.760156
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     lngdp3 |       400    7.987095     .537941   6.327221   8.736859
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     lngdp4 |       400    7.886774    .5983831   5.665983   8.729272
```

Exhibit 10

```
. forvalues y = 1995(2)1999 {
  2.         forvalues i = 1/4 {
  3.                 summ gdp‘i’_‘y’
  4.                 }
  5.         }
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
   gdp1_1995 |       400    3226.703    1532.497    328.393   6431.328
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
   gdp2_1995 |       400    3242.162    1525.788   133.2281   6375.105
```

```
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp3_1995 |        400    3328.577    1457.716   559.5993   6228.302
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp4_1995 |        400    3093.778    1490.646   288.8719   6181.229
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp1_1997 |        400    3597.038    1686.571   438.5756   7083.191
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp2_1997 |        400    3616.478    1677.353   153.0657   7053.826
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp3_1997 |        400    3710.242     1603.25   667.2679   6948.194
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp4_1997 |        400    3454.322    1639.356   348.2078   6825.981
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp1_1999 |        400    3388.038    1609.122   344.8127   6752.894
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp2_1999 |        400     3404.27    1602.077   139.8895    6693.86
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp3_1999 |        400    3495.006    1530.602   587.5793   6539.717
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
   gdp4_1999 |        400    3248.467    1565.178   303.3155   6490.291
```

Exhibit 11

```
. foreach v of varlist lexp-safewater {
  2.         summ ‘v’
  3.         correlate popgrowth ‘v’
  4.         scatter popgrowth ‘v’
  5.         }

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        lexp |        68    72.27941    4.715315         54         79
(obs=68)

             | popgro~h     lexp
-------------+------------------
   popgrowth |   1.0000
        lexp |  -0.4360   1.0000

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       gnppc |        63    8674.857    10634.68        370      39980
(obs=63)

             | popgro~h    gnppc
-------------+------------------
   popgrowth |   1.0000
       gnppc |  -0.3580   1.0000

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
   safewater |        40        76.1    17.89112         28        100
(obs=40)

             | popgro~h safewa~r
-------------+------------------
   popgrowth |   1.0000
   safewater |  -0.4280   1.0000
```

Exhibit 12

```
. local ctycode 111 112 136 134

. local i 0

. foreach c of local ctycode {
  2.         local ++i
  3.         local rc "‘rc’ (‘i’=‘c’)"
  4.         }

. display "‘rc’"
 (1=111) (2=112) (3=136) (4=134)

. recode cc ‘rc’, gen(newcc)
(400 differences between cc and newcc)

. tab newcc

  RECODE of |
         cc |      Freq.     Percent        Cum.
------------+-----------------------------------
        111 |        100       25.00       25.00
        112 |        100       25.00       50.00
        134 |        100       25.00       75.00
        136 |        100       25.00      100.00
------------+-----------------------------------
```

```
      Total |        400      100.00
```

Exhibit 13

```
. local country US UK DE FR

. local yrlist 1995 1999

. forvalues i = 1/4 {
  2.          local cname: word ‘i’ of ‘country’
  3.          foreach y of local yrlist {
  4.                  rename gdp‘i’_‘y’ gdp‘cname’_‘y’
  5.                  }
  6.          }

. summm gdpUS*
```

```
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
  gdpUS_1995 |       400    3226.703    1532.497    328.393   6431.328
  gdpUS_1999 |       400    3388.038    1609.122   344.8127   6752.894
```

Exhibit 14

```
. webuse abdata,clear

. describe

Contains data from http://www.stata-press.com/data/r8/abdata.dta
  obs:         1,031
 vars:            30                          3 Sep 2002 12:25
 size:       105,162 (99.8% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
c1              str9   %9s
ind             float  %9.0g
year            float  %9.0g
emp             float  %9.0g
wage            float  %9.0g
cap             float  %9.0g
indoutpt        float  %9.0g
n               float  %9.0g
w               float  %9.0g
k               float  %9.0g
ys              float  %9.0g
rec             float  %9.0g
yearm1          float  %9.0g
id              float  %9.0g
nL1             float  %9.0g
nL2             float  %9.0g
wL1             float  %9.0g
kL1             float  %9.0g
kL2             float  %9.0g
ysL1            float  %9.0g
ysL2            float  %9.0g
yr1976          byte   %8.0g                  year==  1976.0000
yr1977          byte   %8.0g                  year==  1977.0000
yr1978          byte   %8.0g                  year==  1978.0000
yr1979          byte   %8.0g                  year==  1979.0000
```

```
yr1980          byte   %8.0g                  year==  1980.0000
yr1981          byte   %8.0g                  year==  1981.0000
yr1982          byte   %8.0g                  year==  1982.0000
yr1983          byte   %8.0g                  year==  1983.0000
yr1984          byte   %8.0g                  year==  1984.0000
-------------------------------------------------------------------------------
Sorted by:  id  year
. return list
scalars:
                  r(N) =  1031
                  r(k) =  30
              r(width) =  98
              r(N_max) =  494610
              r(k_max) =  5000
           r(widthmax) =  50848
            r(changed) =  0
. local sb: sortedby
. di "dataset sorted by : `sb'"
dataset sorted by : id year
```

Exhibit 15

```
. summarize emp, detail
                            emp
-------------------------------------------------------------
      Percentiles      Smallest
 1%         .142           .104
 5%         .431           .122
10%         .665           .123       Obs                1031
25%         1.18           .125       Sum of Wgt.        1031
50%        2.287                      Mean           7.891677
                        Largest       Std. Dev.      15.93492
75%        7.036         101.04
90%       17.919        103.129       Variance       253.9217
95%         32.4        106.565       Skewness       3.922732
99%         89.2        108.562       Kurtosis       19.46982
. return list
scalars:
                  r(N) =  1031
              r(sum_w) =  1031
               r(mean) =  7.891677013539667
                r(Var) =  253.9217371514514
                 r(sd) =  15.93492193741317
           r(skewness) =  3.922731923543386
           r(kurtosis) =  19.46982480250623
                r(sum) =  8136.319000959396
                r(min) =  .1040000021457672
                r(max) =  108.5619964599609
                 r(p1) =  .1420000046491623
                 r(p5) =  .4309999942779541
                r(p10) =  .6650000214576721
                r(p25) =  1.179999947547913
                r(p50) =  2.286999940872192
                r(p75) =  7.035999774932861
                r(p90) =  17.91900062561035
                r(p95) =  32.40000152587891
               r(p99) =  89.19999694824219
. scalar iqr = r(p75) - r(p25)
. di "IQR = " iqr
IQR = 5.8559998
. scalar semean = r(sd)/sqrt(r(N))
. di "Mean = " r(mean) " S.E. = " semean
Mean = 7.891677 S.E. = .49627295
```

Exhibit 16

```
. tsset
       panel variable:  id, 1 to 140
        time variable:  year, 1976 to 1984
. return list
scalars:
              r(tmax) =  1984
              r(tmin) =  1976
              r(imax) =  140
              r(imin) =  1
macros:
          r(panelvar) : "id"
           r(timevar) : "year"
             r(unit1) : "."
             r(tsfmt) : "%9.0g"
             r(tmaxs) : "1984"
             r(tmins) : "1976"
```

Exhibit 17

```
. g lowind = (ind<6)
. ttest emp, by(lowind)
Two-sample t test with equal variances
------------------------------------------------------------------------------
   Group |     Obs        Mean    Std. Err.   Std. Dev.   [95% Conf. Interval]
---------+--------------------------------------------------------------------
       0 |     434    8.955942    .9540405    19.87521    7.080816    10.83107
       1 |     597     7.11799    .5019414    12.26423    6.132201    8.103779
---------+--------------------------------------------------------------------
combined |    1031    7.891677     .496273    15.93492    6.917856    8.865498
---------+--------------------------------------------------------------------
    diff |            1.837952    1.004043               -.1322525    3.808157
------------------------------------------------------------------------------
Degrees of freedom: 1029
                      Ho: mean(0) - mean(1) = diff = 0
     Ha: diff < 0               Ha: diff != 0              Ha: diff > 0
       t =   1.8306               t =   1.8306              t =   1.8306
   P < t =   0.9663          P > |t| =   0.0675          P > t =   0.0337
. return list
scalars:
                r(sd) =  15.93492193741317
              r(sd_2) =  12.26422618476487
              r(sd_1) =  19.87520847697869
```

```
                 r(se) =  1.004042693732077
                r(p_u) =  .0337282628926325
                r(p_l) =  .9662717371073675
                  r(p) =  .067456525785265
                  r(t) =  1.83055206312211
               r(df_t) =  1029
               r(mu_2) =  7.117989959978378
                r(N_2) =  597
               r(mu_1) =  8.955942384452314
                r(N_1) =  434
```

Exhibit 18

```
. regress emp wage cap

      Source |       SS       df       MS              Number of obs =    1031
-------------+------------------------------           F(  2,  1028) = 1160.71
       Model |   181268.08     2    90634.04           Prob > F      =  0.0000
    Residual |  80271.3092  1028  78.0849311           R-squared     =  0.6931
-------------+------------------------------           Adj R-squared =  0.6925
       Total |  261539.389  1030  253.921737           Root MSE      =  8.8366

------------------------------------------------------------------------------
         emp |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        wage |  -.3238453   .0487472    -6.64   0.000    -.4195008   -.2281899
         cap |   2.104883   .0440642    47.77   0.000     2.018417    2.191349
       _cons |   10.35982   1.202309     8.62   0.000     8.000557    12.71908
------------------------------------------------------------------------------

. ereturn list

scalars:
                  e(N) =  1031
               e(df_m) =  2
               e(df_r) =  1028
                  e(F) =  1160.711019312048
                 e(r2) =  .6930813769821942
               e(rmse) =  8.83656783747737
                e(mss) =  181268.0800475577
                e(rss) =  80271.30921843699
               e(r2_a) =  .6924842590385798
                 e(ll) =  -3707.867843699609
               e(ll_0) =  -4316.762338658647

macros:
            e(depvar) : "emp"
               e(cmd) : "regress"
           e(predict) : "regres_p"
             e(model) : "ols"

matrices:
                  e(b) :  1 x 3
                  e(V) :  3 x 3

functions:
            e(sample)

. local regressors: colnames e(b)

. di "Regressors: ‘regressors’"
Regressors: wage cap _cons
```

Exhibit 19

```
. generate rooms2 = rooms^2

. qui reg lprice rooms

. est store model1

. qui reg lprice rooms rooms2 ldist

. est store model2

. qui reg lprice ldist stratio lnox

. est store model3

. qui reg lprice lnox ldist rooms stratio

. est store model4

. est table model1 model2 model3 model4, stat(r2_a rmse) ///
>  b(%7.3g) se(%6.3g) p(%4.3f)

----------------------------------------------------
    Variable |  model1    model2    model3    model4
-------------+--------------------------------------
       rooms |    .369     -.821                .255
             |   .0201      .183               .0185
             |   0.000     0.000               0.000
      rooms2 |             .0889
             |              .014
             |             0.000
       ldist |              .237     -.157     -.134
             |             .0255     .0505     .0431
             |             0.000     0.002     0.002
     stratio |                      -.0775    -.0525
             |                       .0066     .0059
             |                       0.000     0.000
        lnox |                       -1.22     -.954
             |                        .135      .117
             |                       0.000     0.000
       _cons |    7.62      11.3      13.6      11.1
             |    .127      .584      .304      .318
             |   0.000     0.000     0.000     0.000
-------------+--------------------------------------
        r2_a |    .399        .5      .424      .581
        rmse |    .317      .289      .311      .265
----------------------------------------------------
                                      legend: b/se/p
```

Exhibit 20

```
.
. estout model1 model2 model3 model4 using ch3.19b_est.tex, ///
> style(tex) replace title("Models of median housing price")  ///
> prehead(\\begin{table}[htbp]\\caption{{\sc @title}}\\centering\\medskip ///
> \begin{tabular}{l*{@M}{r}}) ///
> posthead("\hline") prefoot("\hline") ///
> varlabels(rooms2 "rooms$^2$" _cons "constant") legend ///
> stats(N F r2_a rmse, fmt(%6.0f %6.0f %8.3f %6.3f) ///
> labels("N" "F" "$\bar{R}^2$" "RMS error")) ///
> cells(b(fmt(%8.3f)) se(par format(%6.3f))) ///
> postfoot(\hline\end{tabular}\end{table}) notype
```

Table 1: Models of median housing price

	model1	model2	model3	model4
	b/se	b/se	b/se	b/se
rooms	0.369	-0.821		0.255
	(0.020)	(0.183)		(0.019)
rooms^2		0.089		
		(0.014)		
ldist		0.237	-0.157	-0.134
		(0.026)	(0.050)	(0.043)
stratio			-0.077	-0.052
			(0.007)	(0.006)
lnox			-1.215	-0.954
			(0.135)	(0.117)
constant	7.624	11.263	13.614	11.084
	(0.127)	(0.584)	(0.304)	(0.318)
N	506	506	506	506
F	337	169	125	176
$\bar{R}^2$	0.399	0.500	0.424	0.581
RMS error	0.317	0.289	0.311	0.265

Exhibit 21

```
. capture program drop semean
. *! semean  v1.0.0  CFBaum  16dec2004
. program define semean, rclass
  1. version 8.2
  2. syntax varlist(max=1 numeric)
  3. quietly summarize ‘varlist’
  4. scalar semean = r(sd)/sqrt(r(N))
  5. di _n "Mean of ‘varlist’ = " r(mean) " S.E. = " semean
  6. return scalar semean = semean
  7. return scalar mean = r(mean)
  8. return local var ‘varlist’
  9. end
. semean emp
Mean of emp = 7.891677 S.E. = .49627295
. return list
scalars:
              r(mean) =  7.891677013539667
            r(semean) =  .4962729540865196
macros:
               r(var) : "emp"
```

Exhibit 22

```
. capture program drop semean
. *! semean  v1.0.1  CFBaum  16dec2004
. program define semean, rclass
  1. version 8.2
  2. syntax varlist(max=1 numeric) [if] [in] [, noPRInt]
  3. marksample touse
  4. quietly summarize ‘varlist’ if ‘touse’
  5. scalar semean = r(sd)/sqrt(r(N))
  6. if ("‘print’" ~= "noprint") {
  7.         di _n "Mean of ‘varlist’ = " r(mean) " S.E. = " semean
  8.         }
  9. return scalar semean = semean
 10. return scalar mean = r(mean)
 11. return scalar N = r(N)
 12. return local var ‘varlist’
 13. end
. semean emp if year < 1982, noprint
. return list
scalars:
                 r(N) =  778
              r(mean) =  8.579679950573757
            r(semean) =  .6023535944792725
macros:
               r(var) : "emp"
```

Exhibit 23

```
. capture program drop semean

. *! semean  v1.0.2  CFBaum  16dec2004
. program define semean, rclass byable(recall)
  version 8.2
  syntax varlist(max=1 ts numeric) [if] [in] [, noPRInt]
  marksample touse
  quietly summarize ‘varlist’ if ‘touse’
  scalar semean = r(sd)/sqrt(r(N))
  if ("‘print’" ~= "noprint") {
          di _n "Mean of ‘varlist’ = " r(mean) " S.E. = " semean
          }
  return scalar semean = semean
  return scalar mean = r(mean)
  return scalar N = r(N)
  return local var ‘varlist’
  end

. semean D.emp if year == 1982

Mean of D.emp = -.79091424 S.E. = .17187137

. bysort year: semean emp
-------------------------------------------------------------------------------

-> year = 1976

Mean of emp = 9.8449251 S.E. = 2.1021706
-------------------------------------------------------------------------------

-> year = 1977

Mean of emp = 8.5351159 S.E. = 1.393463
-------------------------------------------------------------------------------

-> year = 1978

Mean of emp = 8.6443428 S.E. = 1.3930028
-------------------------------------------------------------------------------

-> year = 1979

Mean of emp = 8.7162357 S.E. = 1.4311206
-------------------------------------------------------------------------------

-> year = 1980

Mean of emp = 8.5576715 S.E. = 1.4611882
-------------------------------------------------------------------------------

-> year = 1981

Mean of emp = 7.7214 S.E. = 1.3467025
-------------------------------------------------------------------------------

-> year = 1982

Mean of emp = 6.9304857 S.E. = 1.2245105
-------------------------------------------------------------------------------

-> year = 1983

Mean of emp = 5.2992564 S.E. = 1.3286027
-------------------------------------------------------------------------------

-> year = 1984

Mean of emp = 2.2205143 S.E. = .48380791
```

Exhibit 24

```
. capture program drop semean
. *! semean  v1.1.0  CFBaum  16dec2004
. program define semean, rclass byable(recall)
  version 8.2
  syntax varlist(max=1 ts numeric) [if] [in] [, noPRInt FUNCtion(string)]
  marksample touse
  tempvar target
  if "‘function’" == "" {
          local tgt "‘varlist’"
          }
  else {
          local tgt "‘function’(‘varlist’)"
          }
  capture tsset
  capture gen double ‘target’ = ‘tgt’ if ‘touse’
  if _rc > 0 {
          di as err "Error: bad function ‘tgt’"
          error 198
          }
  quietly summarize ‘target’
  scalar semean = r(sd)/sqrt(r(N))
  if ("‘print’" ~= "noprint") {
          di _n "Mean of ‘tgt’ = " r(mean) " S.E. = " semean
          }
  return scalar semean = semean
  return scalar mean = r(mean)
  return scalar N = r(N)
  return local var ‘tgt’
  end
. semean emp
Mean of emp = 7.891677 S.E. = .49627295
. semean emp, func(sqrt)
Mean of sqrt(emp) = 2.1652401 S.E. = .05576835
. semean emp if year==1982, func(log)
Mean of log(emp) = .92474464 S.E. = .11333991
. return list
scalars:
                 r(N) =  140
              r(mean) =  .9247446421128256
            r(semean) =  .1133399069800714
macros:
               r(var) : "log(emp)"
. semean D.emp if year==1982, func(log)
Mean of log(D.emp) = -2.7743942 S.E. = .39944652
. return list
scalars:
                 r(N) =  22
              r(mean) =  -2.774394169773632
            r(semean) =  .3994465211383764
macros:
               r(var) : "log(D.emp)"
```