Exporting STATA results into MS Word using `xml_tab' and `outreg2'

Abstract:

Some user-written programs allow us to post our results in handy ways. Among those, in particular, `xml_tab' and `outreg2' creates publication-ready tables. Yet, enough flexibility is not guaranteed if using only these commands (ado-files). In this lecture, we discuss about the ways to prepare publication-ready tables using `xml_tab' and `outreg2' with some programming in STATA. The approaches should be applicable to the most of cases that a researcher would face when he/she writes academic papers or/and research reports. It will be easier to follow the session if participants are familiar with the following commands/issues in STATA:

xml_tab	Matrix operators
[R] search	[P] matrix rownames
findit	[P] return
net search	ereturn
[R] help	[R] estimates
[D] describe	estimates store
[P] display	estimates table
[P] macro	[R] tabstat
local	[R] summarize
[P] matrix define	[P] forvalues

Instructor:

Chung Choe

Hanyang University **ERICA Campus**

May 30, 2015

STATA User Written Programs

Perhaps the most useful feature of Stata is its extensibility; in STATA users are able to write their own programs to carry out functions which are not supported by Stata. There are a large number of such programs available to the user, and these are readily available for download over the internet.

Steps to install user written commands:

1. Invoke Stata and type `` `findit' `` or `` `net search' `` on the command window.
2. Stata returns a list of packages and programs that match your search criteria.
3. Clicking on a link brings up a more detailed description of the package, which contains links to install the package itself.

Using the `` `findit' `` command you can search for user-written programs and help on any subject you need.

We will use a user-written command `` `xml_tab' `` for today. You should first type the following,

```
help xml_tab
```

If a help file for `` `xml_tab' `` pops up, it means that the command has been already installed. Otherwise, you should install the command following the steps listed above.

SOME USEFUL STATA COMMANDS

[R] HELP: help for STATA command

[D] DESCRIBE: produces a list of variables with format and labels

[P] DISPLAY

`display' -- display strings and values of scalar expressions.

```
display "I was born in Korea"
```

I was born in Korea

```
di "I have worked at C/I since July 2008"
```

I have worked at C/I since July 2008

[P] MACRO

`local' macros are useful when writing STATA programs. In particular, when you need to code the same things repeatedly, macro functions get you to avoid repetitive works and make your programs shorter.

```
local n = 2
local a Jeremy
local b "Jake"
```

The statements store **2**, **Jeremy** and **Jake** in the local `a', `b' and `c' respectively. Then let's see what are displayed by the following two sentences.

```
di "I have n sons: a and b"
```

I have n sons: a and b

```
di "I have `n' sons: `a' and `b'"
```

I have 2 sons: Jeremy and Jake

Note that you should put **``'** to invoke the locals you defined. Otherwise **`di'** will just return the texts inside **" "**.

```
local HYU "ln_wage age grade not_smsa south ttl_exp tenure black"
di "`HYU'"
```

ln_wage age grade not_smsa south ttl_exp tenure black

[P] MATRIX DEFINE

```
matrix A = (1, 2 \ 3, 4 \ ., .)
matlist A
```

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ . & . \end{pmatrix}$$

```
mat B = (100\100\100)
matlist B
```

$$\begin{pmatrix} 100 \\ 100 \\ 100 \end{pmatrix}$$

```
mat C = J(1,3,.)
matlist C
```

$$(\cdot \quad \cdot \quad \cdot)$$

MATRIX OPERATORS

(A , B) add columns of B to the right of A

```
matrix AB = A, B
matlist AB
```

$$\begin{pmatrix}1 & 2\\ 3 & 4\\ . & .\end{pmatrix},\begin{pmatrix}100\\ 100\\ 100\end{pmatrix}=\begin{pmatrix}1 & 2 & 100\\ 3 & 4 & 100\\ . & . & 100\end{pmatrix}$$

AB \ C add rows of C below rows of AB

```
mat ABC = AB \ C
matlist ABC
```

$$\begin{pmatrix}1 & 2 & 100\\ 3 & 4 & 100\\ . & . & 100\end{pmatrix}\backslash(\cdot\quad\cdot\quad\cdot)=\begin{pmatrix}1 & 2 & 100\\ 3 & 4 & 100\\ . & . & 100\\ . & . & .\end{pmatrix}$$

A' transpose the matrix A

```
mat A = A’
matlist A
```

$$A=\begin{pmatrix}1 & 3 & .\\ 2 & 4 & .\end{pmatrix}$$

[P] MATRIX ROWNAMES

`matrix colnames'** and **` matrix rownames' -- name rows and columns

```
matrix colnames ABC = CEPS INSTEAD DIFF LUX
matrix rownames ABC = ROW1 ROW2 ROW3
matlist ABC
```

```
             |      CEPS    INSTEAD       DIFF         LUX
-------------+----------------------------------------------
        ROW1 |         1          3          .           .
        ROW2 |         2          4          .           .
        ROW3 |       100        100        100           .
```

DIFFERENCE BETWEEN `RETURN' AND `ERETURN'

Stata saves results from general and estimation commands.

1. r-class: general commands such as summarize

```
sum ln_wage age
return list
mat A = r(N)
mat list A
```

```
symmetric A[1,1]
       c1
r1  28091
```

2. e-class: estimation commands such as regress, logistic, etc., that fit statistical models

```
regress ln_wage age
ereturn list
mat B = e(b)'
matlist B
```

```
             |        y1
-------------+-----------
         age |  .0199024
       _cons |  1.098344
```

[R] ESTIMATES -- Estimation Results

`estimates store' -- store estimation results

`estimates table' -- display table of estimation results

```
reg ln_wage age grade
estimates store A
reg ln_wage age age2 grade
est store B
est table A B, star
```

```
----------------------------------------------
    Variable |       A                B
-------------+--------------------------------
         age |   .0145525***     .0558715***
       grade |  .08260302***    .08125002***
        age2 |                 -.00068838***
       _cons |  .21837678***   -.35321247***
----------------------------------------------
      legend: * p<0.05; ** p<0.01; *** p<0.001
```

Table 1: Earning function from OLS				
	Model 1		Model 2	
	coef	se	coef	se
grade	0.065***	0.001	0.063***	0.001
not_smsa	-0.173***	0.005	-0.172***	0.005
south	-0.097***	0.005	-0.099***	0.005
ttl_exp	0.029***	0.001	0.029***	0.001
tenure	0.020***	0.001	0.020***	0.001
black	-0.066***	0.005	-0.068***	0.005
AGE				
age	-0.004***	0.001		
AGE20			0.125***	0.010
AGE30			0.071***	0.012
AGE40			-0.012	0.015
_cons	0.838***	0.018	0.663***	0.015
N	28,091		28,091	
r2	0.363		0.369	
note: *** p<0.01, ** p<0.05, * p<0.1				

/* 1. Exporting results from OLS */

```
local HYU "grade not_smsa south ttl_exp tenure black"

regress ln_wage `HYU' age
estimates store r1, title(Model 1)

regress ln_wage `HYU' AGE20 AGE30 AGE40
estimates store r2, title(Model 2)

xml_tab r1 r2, replace nolabel save(ols.xml) stats(N r2) lines(COL_NAMES 2
LAST_ROW 13 EST_NAMES 2) ///
title("Table 1: Earning function from OLS") rblanks(black "AGE" SCCB0)
format(SCLR2 NCCR3 NCCR3)
```

With a more complete set of options:

Table 1: Earning function from OLS

	Model 1		Model 2	
	coef	se	coef	se
current grade completed	0.065***	0.001	0.063***	0.001
1 if not SMSA	-0.173***	0.005	-0.172***	0.005
1 if south	-0.097***	0.005	-0.099***	0.005
total work experience	0.029***	0.001	0.029***	0.001
job tenure, in years	0.020***	0.001	0.020***	0.001
black	-0.066***	0.005	-0.068***	0.005
AGE				
age in current year	-0.004***	0.001		
AGE20			0.125***	0.010
AGE30			0.071***	0.012
AGE40			-0.012	0.015
_cons	0.838***	0.018	0.663***	0.015
Number of observations	28,091		28,091	
R2	0.363		0.369	

note: *** p<0.01, ** p<0.05, * p<0.1

Complete results available from authors.

[R] TABSTAT

`tabstat' -- display table of summary statistics

```
tabstat `HYU', s(mean sd n)
```

```
   stats |   ln_wage       age     grade  not_smsa     south   ttl_exp    tenure     black
---------+--------------------------------------------------------------------------------
    mean |  1.677103  29.07988  12.53636  .2826172  .4094194  6.241974  3.123121  .2825816
      sd |  .4778416  6.697371  2.328057  .4502798  .4917355   4.65452  3.750492  .4502627
       N |     28091     28091     28091     28091     28091     28091     28091     28091
--------------------------------------------------------------------------------------------
```

DIFFERENCE BETWEEN `TABSTAT' AND `SUMMARIZE'

```
sum `HYU'
return list
mat A = r(mean)
matlist A
```

```
             |        c1
-------------+-----------
          r1 |  .2825816
```

```
tabstat `HYU', s(mean) save
return list
mat B = r(StatTotal)'
matlist B
```

```
             |      mean
-------------+-----------
     ln_wage |  1.677103
         age |  29.07988
       grade |  12.53636
    not_smsa |  .2826172
       south |  .4094194
     ttl_exp |  6.241974
      tenure |  3.123121
       black |  .2825816
```

Table 1: Summary Statistics 1					
	Total	SMSA	non_SMSA	non_BLACK	BLACK
ln_wage	1.68	1.74	1.51	1.72	1.57
age	29.08	28.99	29.32	29.15	28.89
grade	12.54	12.71	12.09	12.78	11.92
not_smsa	0.28	0.00	1.00	0.30	0.24
south	0.41	0.36	0.55	0.32	0.62
ttl_exp	6.24	6.26	6.20	6.32	6.05
tenure	3.12	3.12	3.14	3.10	3.19
black	0.28	0.30	0.24	0.00	1.00
age2	890.49	884.56	905.55	895.85	876.89
ttl_exp2	60.63	60.60	60.70	61.51	58.39
tenure2	23.82	23.61	24.36	23.43	24.81
Observations	28,091.00	20,152.00	7,939.00	20,153.00	7,938.00
This is the first example table.					

```
/*      2. Summary Statistics `without' STD   */

local HYU "ln_wage age grade not_smsa south ttl_exp tenure black age2
ttl_exp2 tenure2"

tabstat `HYU', by(not_smsa) s(mean) save
mat stotal = r(StatTotal)
mat M1 = r(StatTotal)',r(Stat1)',r(Stat2)'
tabstat `HYU', by(black) s(mean) save
mat M2 = r(Stat1)',r(Stat2)'

tabstat not_smsa, by(not_smsa) s(n) save
mat obs1 = r(StatTotal)',r(Stat1)',r(Stat2)'
tabstat black, by(black) s(n) save
mat obs2 = r(Stat1),r(Stat2)
mat obs = obs1, obs2
mat rown obs = Observations

mat BigM = (M1,M2) \ obs
mat coln BigM = Total SMSA non_SMSA non_BLACK BLACK

xml_tab BigM, save(sumstat1.xml) title(Table 2: Summary Statistics 1) replace
format(SCLR2 NCCR2 NCCR2) ///
lines(SCOL_NAMES 2 COL_NAMES 2 LAST_ROW 13) notes(This is the first example
table.)
```

[P] FORVALUES

`forvalues' -- loop over consecutive values

```
forvalues i = 1/10 {
    di "`i'"
}
1
2
3
4
5
6
7
8
9
10
```

```
mat M1 = J(10,1,.)
mat M2 = (5,4,3,2,1)
forvalues i = 1/5 {
    mat M1[`i',1] = M2[1,`i']
}
matlist M1
             |        c1
-------------+-----------
          r1 |         5
          r2 |         4
          r3 |         3
          r4 |         2
          r5 |         1
          r6 |         .
          r7 |         .
          r8 |         .
          r9 |         .
         r10 |         .
```

[P] FORVALUES - continues

```
mat M1 = J(10,1,.)
mat M2 = (5,4,3,2,1)

forvalues i = 1/5 {
    mat M1[`i'*2,1] = M2[1,`i']
}
```

```
             |        c1
-------------+-----------
          r1 |         .
          r2 |         5
          r3 |         .
          r4 |         4
          r5 |         .
          r6 |         3
          r7 |         .
          r8 |         2
          r9 |         .
         r10 |         1
```

```
/*      3. Summary Statistics `with' STD       */
local HYU "ln_wage age grade not_smsa south ttl_exp tenure black"

tabstat `HYU', by(not_smsa) s(mean sd) save
mat M1 = r(StatTotal)',r(Stat1)',r(Stat2)'
tabstat `HYU', by(AGE) s(mean sd) save
mat M2 = r(Stat1)',r(Stat2)',r(Stat3)',r(Stat4)'

tabstat not_smsa, by(not_smsa) s(n) save
mat obs1 = r(StatTotal)',r(Stat1)',r(Stat2)'
tabstat black, by(AGE) s(n) save
mat obs2 = r(Stat1),r(Stat2),r(Stat3),r(Stat4)
mat obs = obs1, obs2

mat T = J(1,14,.)
mat rown T = Obs
matlist T

forvalues i = 1/7 {
    mat T[1,`i'*2-1] = obs[1,`i']
}

mat BigM = (M1,M2)\ T
matlist BigM
di "Total non-smsa smsa AGE10 AGE20 AGE30 AGE40 "

xml_tab BigM, save(sumstat2.xml) replace title(Table 3: Summary Statistics 2)
format(SCLR2 NCCR2 NCCR2) lines(SCOL_NAMES 2 COL_NAMES 2 LAST_ROW 13)
```

Table 3: Summary Statistics 2												
	mean	sd	mean	sd	mean	sd	mean	sd	mean	sd	mean	sd
ln_wage	1.68	0.48	1.74	0.47	1.51	0.45	1.31	0.38	1.61	0.41	1.79	0.51
age	29.08	6.70	28.99	6.66	29.32	6.79	18.48	0.71	24.48	2.74	34.19	2.75
grade	12.54	2.33	12.71	2.30	12.09	2.33	11.17	1.55	12.34	2.13	12.91	2.50
not_smsa	0.28	0.45	0.00	0.00	1.00	0.00	0.30	0.46	0.27	0.44	0.30	0.46
south	0.41	0.49	0.36	0.48	0.55	0.50	0.36	0.48	0.40	0.49	0.42	0.49
ttl_exp	6.24	4.65	6.26	4.63	6.20	4.72	1.30	1.08	3.63	2.34	9.18	4.12
tenure	3.12	3.75	3.12	3.73	3.14	3.81	0.66	0.79	1.94	2.04	4.47	4.34
black	0.28	0.45	0.30	0.46	0.24	0.42	0.28	0.45	0.29	0.45	0.29	0.45
Obs	28,091.00	.	20,152.00	.	7,939.00	.	1,616.00	.	13,712.00	.	10,750.00	.

/* 4. Preparing Tables with `OUTREG2' */

```
label var age "나이"

matlist BigM

local HYU "ln_wage age grade not_smsa south ttl_exp tenure black"

mat rown BigM = `HYU'

matlist BigM

xml_tab BigM, save(sumstat_prac.xml) replace
```

xml_tab을 사용해서 표를 만들 경우, 라벨이 영어인 경우는 문제가 되지 않으나, 한글인 경우는 최종결과를 보고하는 표를 확인해 보면 글자가 깨어져 나온다.

이런 경우 outreg2라는 명령어를 사용할 수 있는데, 문제는 outreg2 matrix를 바로 excel로 보낼 수 없고, regession 뒤에 관련된 정보들을 외부 파일을 생성해 리포트 할 수 있다.

```
reg `HYU'
outreg2 using sumstat_prac, sum replace word label ctitle(신성장기반)  ///
addnote("자료: 한국기업데이터(2002-2009), 중소기업진흥공단자료(2002-2009)") ///
title(<표 1> 기초통계량)
```

<표 1> 기초통계량

VARIABLES	(1) 신성장기반
ln(wage/GNP deflator)	1.677
	(0.478)
나이	29.08
	(6.697)
current grade completed	12.54
	(2.328)
1 if not SMSA	0.283
	(0.450)
1 if south	0.409
	(0.492)
total work experience	6.242
	(4.655)
job tenure, in years	3.123
	(3.750)
black	0.283
	(0.450)
Observations	28,091

자료: 한국기업데이터(2002-2009), 중소기업진흥공단자료(2002-2009)

Automatic Update of tables in MS Word (2007)

Tables in Excel, created by `xml_tab', can be linked with MS Word to create dynamically updated documents.

Detailed steps you should take are following:

1. Place tables of results with `xml_tab' into an XML workbook.
2. Open **both** the resulting workbook in Excel **and** your working MS word file.
3. Move the cursor in Word where you want you to put your tables.

   Insert
   Object
   Create from file
   Browse the file to be linked
   Choose the option `link to file'

4. Maybe the tables appeared in Word are not looking same with those in Excel.

   Right-click on a table in Word
   Linked worksheet object
   Links
   Click update now

5. If you are still unable to make a successful update, double-click the table in Word. This opens the table with Excel. Then repeat the step 4.

Note: if you close the Word doc and re-open it later after changing a table, do **NOT** forget first open all the linked files. Otherwise, you will lose the links.

REFERENCE

1. Lokshin, Michael and Zurab Sajaia (2008), Creating print-ready tables in Stata, ***The Stata Journal***, 8(3): 374-389.
2. Stata 10 manuals and online help (via the command `help')