Basics of STATA Software

Introduction to the STATA interface and its functions. Entering and importing data into STATA from economic databases and various data files. Basic data management: opening, viewing, and editing variables. Generating and labeling variables, creating data subsets

Tomáš Oleš
Department of Economic Policy
Faculty of Economics and Finance

January 29, 2025

Agenda

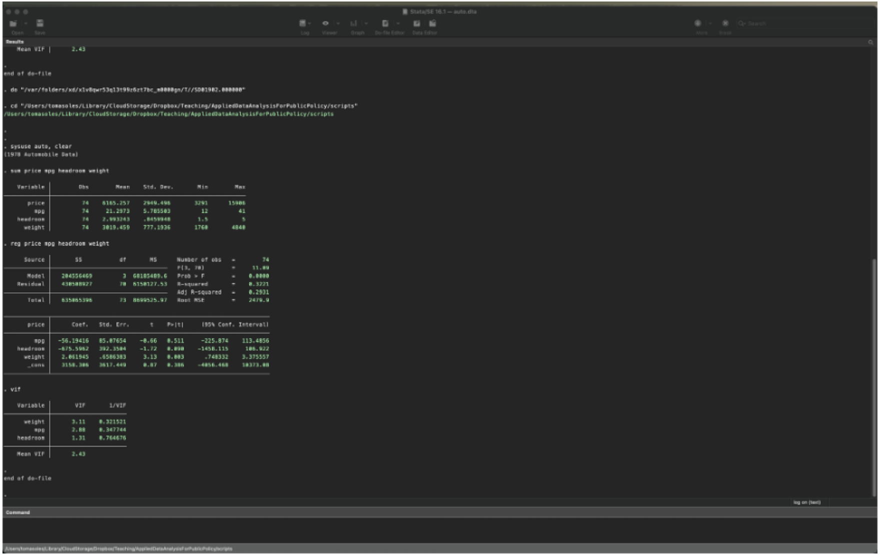
- Get accustomed to the STATA interface.
- Enter and import data into STATA.
- Get comfortable with using the STATA command language.
- Learn common data management commands in STATA.

Why Research Needs to be Reproducible?

Why Research Needs to be Reproducible?

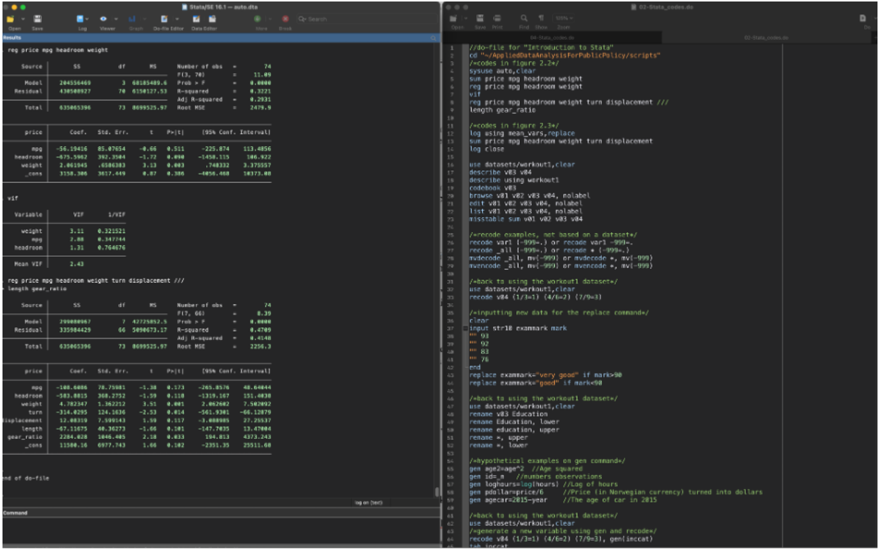
- "The natural (social) scientist is concerned with a particular kind of phenomena . . . he has to confine himself to that which is reproducible . . . I do not claim that the reproducible by itself is more important than the unique. But I do claim that the unique exceeds the treatment by scientific method. Indeed it is the aim of this method to find and test natural laws." — Wolfgang Pauli
- "Kohn's Second Law: An experiment is reproducible until another laboratory tries to repeat it." — Alexander Kohn

Stata Interface



```
    Mean VIF |      2.43

.
end of do-file

. do "/var/folders/xd/x1v0qwr53q13t99z6zt7bc_m0000gn/T//SD01902.000000"

. cd "/Users/tomasoles/Library/CloudStorage/Dropbox/Teaching/AppliedDataAnalysisForPublicPolicy/scripts"
/Users/tomasoles/Library/CloudStorage/Dropbox/Teaching/AppliedDataAnalysisForPublicPolicy/scripts

.
.
. sysuse auto, clear
(1978 Automobile Data)

. sum price mpg headroom weight

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
       price |         74    6165.257    2949.496       3291      15906
         mpg |         74     21.2973    5.785503         12         41
    headroom |         74    2.993243    .8459948        1.5          5
      weight |         74    3019.459    777.1936       1760       4840

. reg price mpg headroom weight

      Source |       SS           df       MS      Number of obs   =        74
-------------+----------------------------------   F(3, 70)        =     11.09
       Model |  204556469         3  68185489.6   Prob > F        =    0.0000
    Residual |  430508927        70  6150127.53   R-squared       =    0.3221
-------------+----------------------------------   Adj R-squared   =    0.2931
       Total |  635065396        73  8699525.97   Root MSE        =    2479.9

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         mpg |  -56.19416   85.07654    -0.66   0.511     -225.874    113.4856
    headroom |  -675.5962   392.3504    -1.72   0.090    -1458.115     106.922
      weight |   2.061945   .6586383     3.13   0.003      .748332    3.375557
       _cons |   3158.306   3617.449     0.87   0.386    -4056.468    10373.08
------------------------------------------------------------------------------

. vif

    Variable |       VIF       1/VIF  
-------------+----------------------
      weight |      3.11    0.321521
         mpg |      2.88    0.347744
    headroom |      1.31    0.764676
-------------+----------------------
    Mean VIF |      2.43

.
end of do-file

.
```

A Key to Reproducibility is to Learn to Code < 3



```
. reg price mpg headroom weight

      Source |       SS           df       MS      Number of obs   =        74
-------------+----------------------------------   F(3, 70)        =     11.09
       Model |   204556469         3  68185489.6   Prob > F        =    0.0000
    Residual |   430508927        70  6150127.53   R-squared       =    0.3221
-------------+----------------------------------   Adj R-squared   =    0.2931
       Total |   635065396        73  8699525.97   Root MSE        =    2479.9

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         mpg |  -56.19416   85.07654    -0.66   0.511     -225.874    113.4856
    headroom |  -675.5962   392.3504    -1.72   0.090    -1458.115     106.922
      weight |   2.061945   .6586383     3.13   0.003      .748332    3.375557
       _cons |   3158.306   3617.449     0.87   0.386    -4056.468    10373.08
------------------------------------------------------------------------------


. vif

    Variable |       VIF       1/VIF
-------------+----------------------
      weight |      3.11    0.321521
         mpg |      2.88    0.347744
    headroom |      1.31    0.764676
-------------+----------------------
    Mean VIF |      2.43

. reg price mpg headroom weight turn displacement ///
> length gear_ratio

      Source |       SS           df       MS      Number of obs   =        74
-------------+----------------------------------   F(7, 66)        =      8.39
       Model |   299080967         7  42725852.5   Prob > F        =    0.0000
    Residual |   335984429        66  5090673.17   R-squared       =    0.4709
-------------+----------------------------------   Adj R-squared   =    0.4148
       Total |   635065396        73  8699525.97   Root MSE        =    2256.3

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         mpg |  -108.6086   78.75081    -1.38   0.173    -265.8576    48.64044
    headroom |  -583.8815   368.2752    -1.59   0.118    -1319.167    151.4038
      weight |   4.782347   1.362212     3.51   0.001     2.062602    7.502092
        turn |  -314.0295   124.1636    -2.53   0.014    -561.9301   -66.12879
displacement |   12.08319   7.599143     1.59   0.117    -3.088985    27.25537
      length |  -67.11675   40.36273    -1.66   0.101    -147.7035    13.47004
  gear_ratio |   2284.028   1046.405     2.18   0.033      194.813    4373.243
       _cons |   11580.16   6977.743     1.66   0.102     -2351.35    25511.68
------------------------------------------------------------------------------

.
end of do-file

.
```

```
//do-file for "Introduction to Stata"
cd "~/AppliedDataAnalysisForPublicPolicy/scripts"
/*codes in figure 2.2*/
sysuse auto,clear
sum price mpg headroom weight
reg price mpg headroom weight
vif
reg price mpg headroom weight turn displacement ///
length gear_ratio

/*codes in figure 2.3*/
log using mean_vars,replace
sum price mpg headroom weight turn displacement
log close

use datasets/workout1,clear
describe v03 v04
describe using workout1
codebook v03
browse v01 v02 v03 v04, nolabel
edit v01 v02 v03 v04, nolabel
list v01 v02 v03 v04, nolabel
misstable sum v01 v02 v03 v04

/*recode examples, not based on a dataset*/
recode var1 (-999=.) or recode var1 -999=.
recode _all (-999=.) or recode * (-999=.)
mvdecode _all, mv(-999) or mvdecode *, mv(-999)
mvencode _all, mv(-999) or mvencode *, mv(-999)

/*back to using the workout1 dataset*/
use datasets/workout1,clear
recode v04 (1/3=1) (4/6=2) (7/9=3)

/*inputting new data for the replace command*/
clear
input str10 exammark mark
"" 93
"" 92
"" 83
"" 76
end
replace exammark="very good" if mark>90
replace exammark="good" if mark<90

/*back to using the workout1 dataset*/
use datasets/workout1,clear
rename v03 Education
rename Education, lower
rename education, upper
rename *, upper
rename *, lower

/*hypothetical examples on gen command*/
gen age2=age^2  //Age squared
gen id=_n   //numbers observations
gen loghours=log(hours) //Log of hours
gen pdollar=price/6     //Price (in Norwegian currency) turned into dollars
gen agecar=2015-year    //The age of car in 2015

/*back to using the workout1 dataset*/
use datasets/workout1,clear
/*generate a new variable using gen and recode*/
recode v04 (1/3=1) (4/6=2) (7/9=3), gen(inccat)
tab inccat
```

Create Your First Code and Run It

```
display("Hello Word!")
```

Using log-file

```
*Load data and create a log-file
log using "~/AppliedDataAnalysisForPublicPolicy/log/my_first_log.log", ///
replace
sum price mpg headroom weight turn displacement
log close
```

Entering data

```
/*inputting new data for the replace command*/
clear
input str10 exammark mark
"" 93
"" 92
"" 83
"" 76
end
```

Importing Data

Default Data Format: Stata primarily uses `.dta` files but supports multiple formats, including:

- `.xls`, `.xlsx` (Excel)
- `.csv` (Comma-separated values)
- `.shp`, `.dbf` (Shapefiles & dBase)
- ...

What if my data is in an unsupported format? Convert it to a Stata-compatible format (e.g., `.csv`) using Python or another tool.
Limitation: Stata can handle only one dataset (table) at a time.

Tasks:

- Import `datasets/highest-points-by-state.csv` into Stata.
- Import `datasets/workout1.dta`.

Solution: Importing Data into Stata

Importing a CSV File:

```
import delimited "datasets/highest-points-by-state.csv", clear
```

Importing a Stata (.dta) File:

```
use "datasets/workout1.dta", clear
```

Converting Unsupported Formats: Use Python (e.g., Pandas) to convert files:

```
import pandas as pd
df = pd.read_json("data.json")
df.to_csv("data.csv", index=False)
```

Examining the Data

```
use datasets/workout1,clear
describe v03 v04
describe using workout1
codebook v03
browse v01 v02 v03 v04, nolabel
edit v01 v02 v03 v04, nolabel
list v01 v02 v03 v04, nolabel
misstable sum v01 v02 v03 v04
```

Making Changes to Variables

```
/*recode examples, not based on a dataset*/
recode var1 (-999=.) or recode var1 -999=.
recode _all (-999=.) or recode * (-999=.)
mvdecode _all, mv(-999) or mvdecode *, mv(-999)
mvencode _all, mv(-999) or mvencode *, mv(-999)

/*back to using the workout1 dataset*/
use datasets/workout1,clear
recode v04 (1/3=1) (4/6=2) (7/9=3)
```

Replacing the Data

```
/*inputting new data for the replace command*/
clear
input str10 exammark mark
"" 93
"" 92
"" 83
"" 76
end
replace exammark="very good" if mark>90
replace exammark="good" if mark<90
/*back to using the workout1 dataset*/
use datasets/workout1,clear
rename v03 Education
rename Education, lower
rename *, upper
```

Mathematical Operators in Stata

Arithmetic		**Logical**		**Relational**	
$+$	addition	$\&$	and	$>$	greater than
$-$	subtraction	$\vert$	or	$<$	less than
$*$	multiplication	$\neg$	not	$\geq$	greater or equal
$/$	division	$\sim$	not	$\leq$	less or equal
^	power			$==$	equal
$-$	negation			$\neq$	not equal
$+$	string concatenation			$\sim=$	not equal

Table: The three types of mathematical operators used in Stata

Generating variables: workout1.dta

```
/*hypothetical examples on gen command*/

gen age2=age^2  //Age squared
gen id=_n //numbers observations
gen loghours=log(hours) //Log of hours
gen pdollar=price/6  //Price (in Norwegian currency) to dollars
gen agecar=2015-year //The age of car in 2015
```

```
*back to using the workout1 dataset*/
use datasets/workout1,clear
/*generate a new variable using gen and recode*/
recode v04 (1/3=1) (4/6=2) (7/9=3), gen(inccat)
tab inccat

/*generate a new variable using gen and replace*/
gen inccat2=.
replace inccat2=1 if (v04<=3)
replace inccat2=2 if (v04>=4) & (v04<=6)
replace inccat2=3 if (v04>=7) & (v04<.)
tab inccat2

/*hypothetical example showing labelling values of several variables */
label define lablikert 1"disagree" 6"agree"
label values var1-var5 lablikert
```

Appending Data

```
/*inputting new data manually*/
clear
input id data var1 var2
1 1 3 2
2 1 4 3
3 1 5 1
end
save dataset1,replace
clear
input id data var1 var2
4 2 3 1
5 2 5 3
6 2 5 4
end
save dataset2,replace
clear
```

```
/*appending data*/
append using dataset1 dataset2,gen(dataset3)
save dataset3
list,sep(0)
```

Merging Data

```
/*inputting new data manually*/
clear
input id v1_14 v2_14
1 3 5
2 4 5
3 2 3
4 1 2
5 1 2
end
save data14,replace
```

```
clear
input id v1_15 v2_15
1 4 5
2 5 5
3 3 4
4 2 3
5 2 3
end
save data15,replace
clear
```

```
/*merging data*/
use data14,clear
merge 1:1 id using data15
save data1415,replace
drop _merge // drops this variable
list, sep(0)
```

Reshaping Data

```
/*reshape from wide to long*/
use data1415,clear //we use the data from above since it is in
                     a wide format
drop _merge
list
reshape long v1_ v2_ , i(id) j(year)
list
```