Getting Started in Data Analysis using Stata

(ver. 5.0)

Oscar Torres-Reyna
Data Consultant
otorres@princeton.edu

List of topics

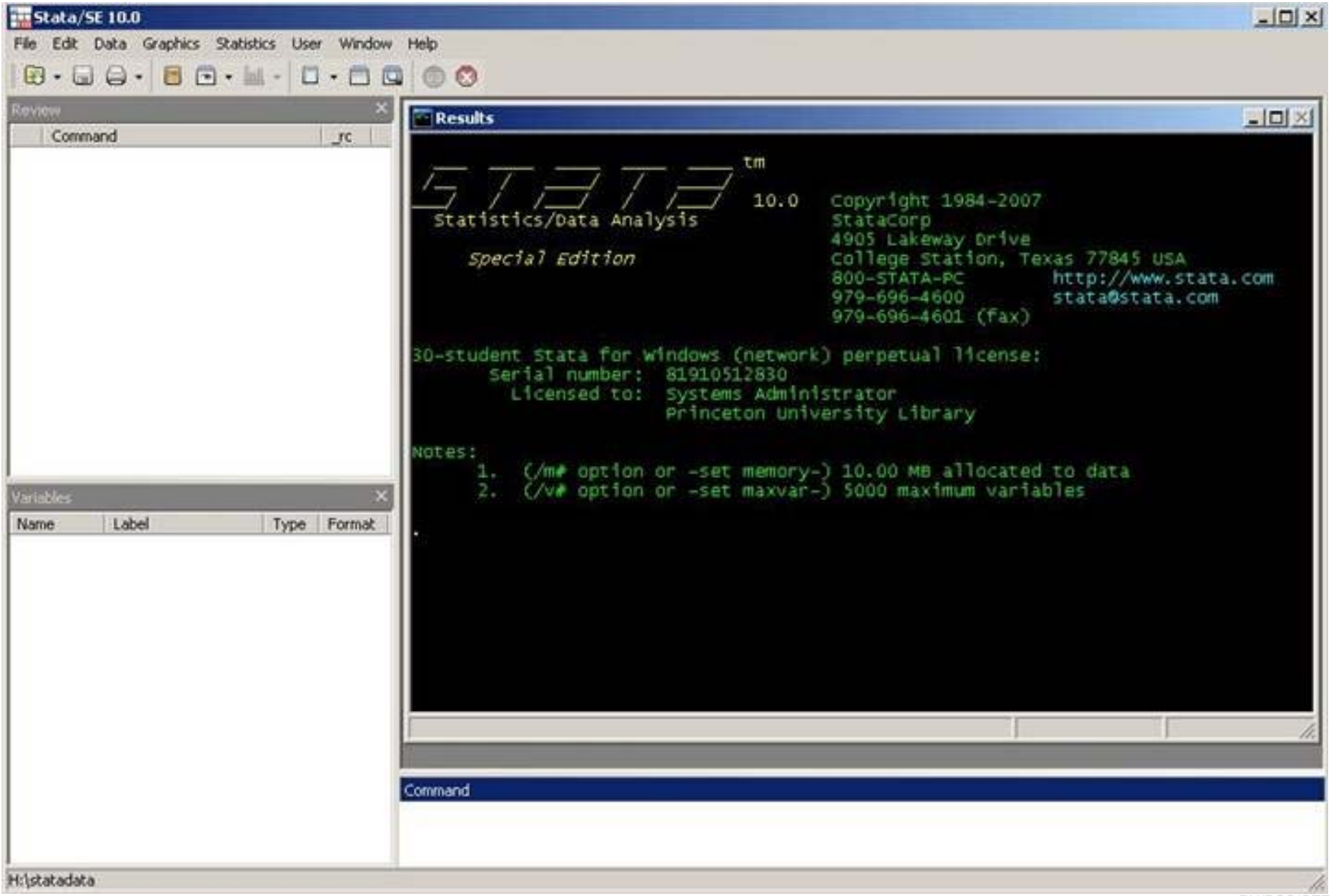
- ❑ What is Stata?
- ❑ Stata screen and general description
- ❑ First steps (log, memory and directory)
- ❑ From SPSS/SAS to Stata
- ❑ Example of a dataset in Excel
- ❑ From *Excel* to *Stata* (copy-and-paste, *.csv)
- ❑ Saving the dataset
- ❑ **Describe** and **summarize** (command, menu)
- ❑ **Rename** and **label variables** (command, menu)
- ❑ Creating new variables (**generate**)
- ❑ Recoding variables (**recode**)
- ❑ Recoding variables using egen
- ❑ Changing values (**replace**)
- ❑ Extracting characters from regular expressions
- ❑ Value labels using the menu
- ❑ Indexing (using _n and _N)
  - ✓ Creating ids and ids by categories
  - ✓ Lags and forward values
  - ✓ Countdown and specific values
- ❑ Sorting
- ❑ Deleting variables (**drop**)
- ❑ **Merge**
- ❑ **Append**
- ❑ Merging fuzzy text (reclink)
- ❑ Frequently used Stata commands
- ❑ Exploring data:
  - ✓ Frequencies (**tab**, **table**)
  - ✓ Crosstabulations (with test for associations)
  - ✓ Descriptive statistics (**tabstat**)
- ❑ Examples of frequencies and crosstabulations
- ❑ Creating dummies
- ❑ Graphs
  - ✓ Scatterplot
  - ✓ Histograms
  - ✓ Catplot (for categorical data)
  - ✓ Bars (graphing mean values)
- ❑ Regression:
  - ✓ Overview and basic setting
  - ✓ Correlation matrix
  - ✓ Output interpretation (what to look for)
  - ✓ Graph matrix
  - ✓ Saving regression coefficients
  - ✓ *F*-test
  - ✓ Testing for linearity
  - ✓ Testing for normality
  - ✓ Testing for homoskedasticity
  - ✓ Testing for omitted-variable bias
  - ✓ Testing for multicolinearity
  - ✓ Robust standard errors
  - ✓ Specification error
  - ✓ Outliers
  - ✓ Summary of influence indicators
  - ✓ Summary of distance measures
  - ✓ Interaction terms
  - ✓ Publishing regression table (**outreg2**)
- ❑ Useful sites (links only)
  - ✓ *Is my model OK?*
  - ✓ *I can't read the output of my model!!!*
  - ✓ Topics in Statistics
  - ✓ Recommended books

What is Stata?

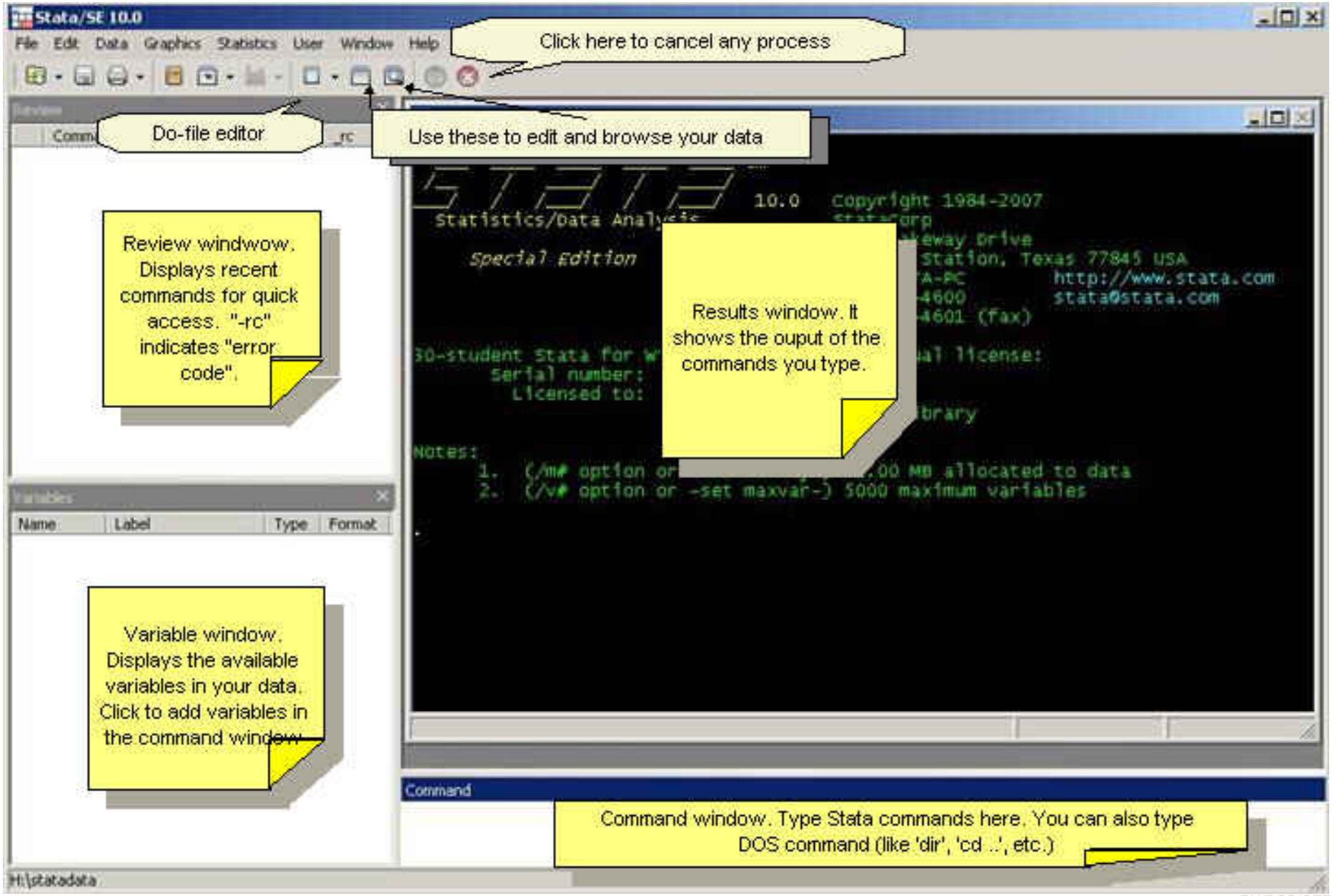
- It is a multi-purpose statistical package to help you explore, summarize and analyze datasets.
- A dataset is a collection of several pieces of information called variables (usually arranged by columns). A variable can have one or several values (information for one or several cases).
- Other statistical packages are SPSS, SAS and R.
- Stata is widely used in social science research and the most used statistical software on campus.

Features	Stata	SPSS	SAS	R
Learning curve	Steep/gradual	Gradual/flat	Pretty steep	Pretty steep
User interface	Programming/point-and-click	Mostly point-and-click	Programming	Programming
Data manipulation	Very strong	Moderate	Very strong	Very strong
Data analysis	Powerful	Powerful	Powerful/versatile	Powerful/versatile
Graphics	Very good	Very good	Good	Good
Cost	Affordable (perpetual licenses, renew only when upgrade)	Expensive (but not need to renew until upgrade, long term licenses)	Expensive (yearly renewal)	Open source

This is the Stata screen…



and here is a brief description …



First steps: Working directory

To check your working directory, type

pwd

```
. pwd
h:\statadata
```

If you want to change it type:

```
cd c:\mydata
```

```
. cd c:\mydata
c:\mydata
```

Use quotes if the new directory has spaces, for example

```
cd “h:\stata and data”
```

```
. cd "h:\stata and data"
h:\stata and data
```

First steps: log file

Create a *log file*, sort of Stata's built-in tape recorder and where you can retrieve the output of your work.

In the command line type

```
log using mylog.log
```

This will create in your working directory a file called '`mylog.log`' which you can read using any word processor (notepad, word).

To close a log file type

```
log close
```

To add more output to an existing log file add the option `append`, type:

```
log using mylog.log, append
```

You can also replace a log file by adding the option `replace`, type:

```
log using mylog.log, replace
```

First steps: set the correct memory allocation

If you get the following error message while opening a datafile

```
no room to add more observations
    An attempt was made to increase the number of observations beyond what is currently possible.  You have the
    following alternatives:

     1.  Store your variables more efficiently; see help compress.  (Think of Stata's data area as the area of a
         rectangle; Stata can trade off width and length.)

     2.  Drop some variables or observations; see help drop.

     3.  Increase the amount of memory allocated to the data area using the set memory command; see help memory.
```

You need to set the *correct memory allocation* for your data. Some big datasets more memory, depending on the size you can type, for example:

```
set mem 700m
```

```
. set mem 700m

Current memory allocation

                    current                                 memory usage
settable            value       description                 (1M = 1024k)
-------------------------------------------------------------------------
set maxvar          5000        max. variables allowed          1.909M
set memory          700M        max. data space               700.000M
set matsize         400         max. RHS vars in models         1.254M
                                                              ---------
                                                              703.163M
```

If this does not work try a bigger number.

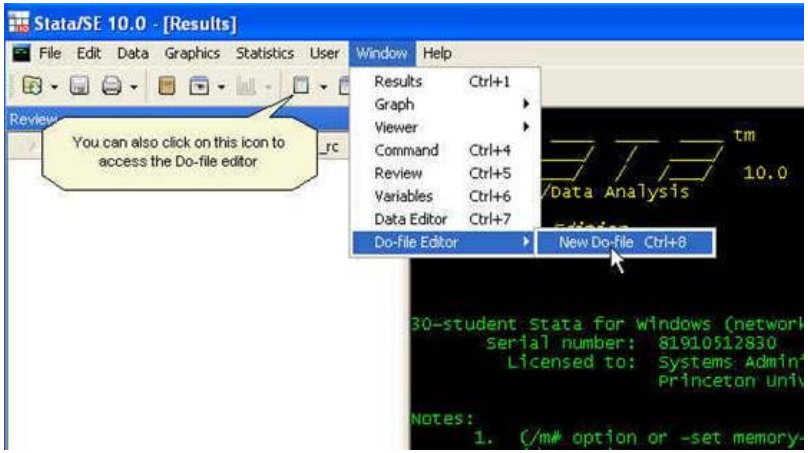
First steps: do-file

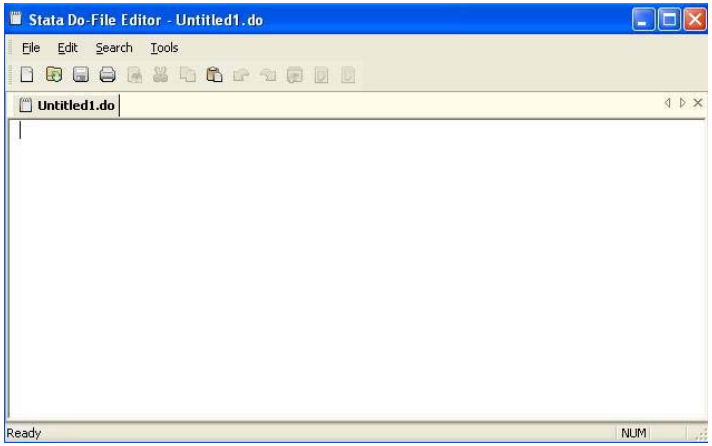
Do-files are ASCII files that contain a sequence of Stata commands to run specific procedures.

You can use do-files to store commands so do you not have to type them again should you need to re-do your work.

You can use any word processor and save the file as ASCII file or you can use Stata's 'do-file editor' with the advantage that you can run the commands from there. Type:

```
doedit
```

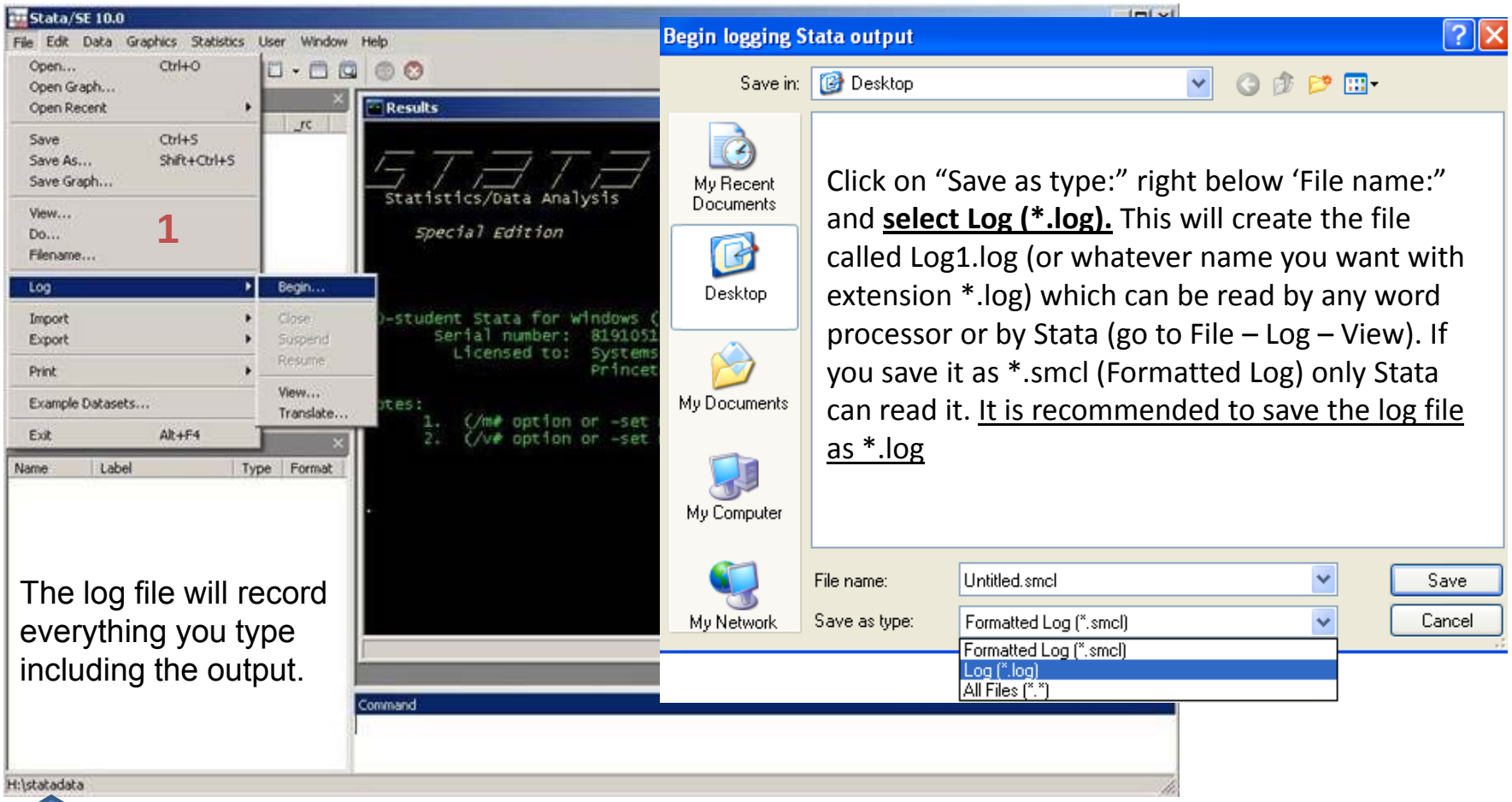




Check the following site for more info: http://www.princeton.edu/~otorres/Stata/

First steps: graphic view

Three basic procedures ***you may want to do first***: create a ***log file*** (sort of Stata's built-in tape recorder and where you can retrieve the output of your work), ***set your working directory***, and set the ***correct memory allocation*** for your data.



The log file will record everything you type including the output.

2

Shows your current working directory. You can change it by typing
`cd c:\mydirectory`

3

When dealing with really big datasets you may want to increase the memory:
`set mem 700m /*You type this in the command window */`
To estimate the size of the file you can use the formula:
Size (in bytes) = (8*Number of cases or rows*(Number of variables + 8))

From SPSS/SAS to Stata

If your data is already in SPSS format (*.sav) or SAS(*.sas7bcat).You can use the command `usespss` to read SPSS files in Stata or the command `usesas` to read SAS files.

If you have a file in SAS XPORT format you can use `fduse` (or go to file-import).

For SPSS and SAS, you may need to install it by typing

```
ssc install usespss
ssc install usesas
```

Once installed just type

```
usespss using “c:\mydata.sav”
usespss using “c:\mydata.sas7bcat”
```

Type `help usespss` or `help usesas` for more details.

For ASCII data please see http://dss.princeton.edu/training/DataPrep101.pdf

PU/DSS/OTR

Example of a dataset in Excel.

Variables are arranged by columns and cases by rows. Each variable has more than one value

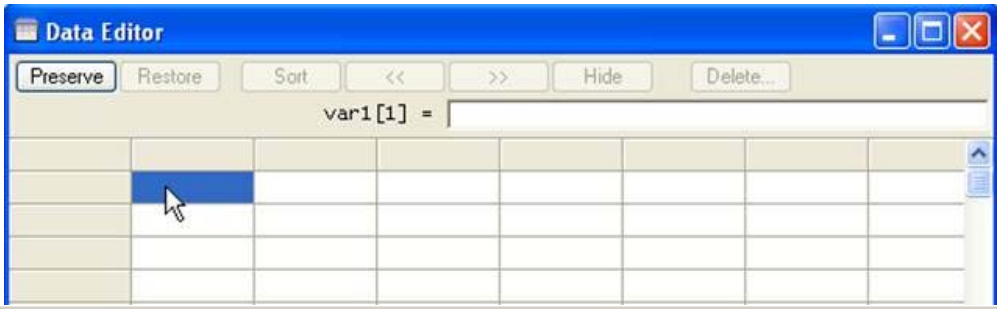
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	ID	Last Name	First Name	City	State	Gender	Student Status	Major	Country	Age	SAT	Average score (grade)	Height (in)	Newspaper readership (times/wk)
2	1	DOE01	JANE01	Los Angeles	California	Female	Graduate	Politics	US	30	2263	67	61	5
3	2	DOE02	JANE02	Sedona	Arizona	Female	Undergraduate	Math	US	19	2006	63	64	7
4	3	DOE01	JOE01	Elmira	New York	Male	Graduate	Math	US	26	2221	78	73	6
5	4	DOE02	JOE02	Lackawana	New York	Male	Graduate	Econ	US	33	1716	78	68	3
6	5	DOE03	JOE03	Defiance	Ohio	Male	Graduate	Econ	US	37	1701	65	71	6
7	6	DOE04	JOE04	Tel Aviv	Israel	Male	Graduate	Econ	Israel	25	1786	69	67	5
8	7	DOE05	JOE05	Cimax	North Carolina	Male	Graduate	Politics	US	39	1577	96	70	5
9	8	DOE03	JANE03	Liberal	Kansas	Female	Undergraduate	Politics	US	21	1842	87	62	5
10	9	DOE04	JANE04	Montreal	Canada	Female	Undergraduate	Math	Canada	18	1813	91	62	6
11	10	DOE05	JANE05	New York	New York	Female	Graduate	Math	US	33	2041	71	66	5
12	11	DOE06	JOE06	Hot Coffe	Mississippi	Male	Undergraduate	Econ	US	18	1787	82	67	3
13	12	DOE06	JANE06	Java	Virginia	Female	Graduate	Math	US	38	1513	79	59	5
14	13	DOE07	JOE07	Varna	Bulgaria	Male	Graduate	Politics	Bulgaria	30	1637	79	63	4
15	14	DOE08	JOE08	Moscow	Russia	Male	Graduate	Politics	Russia	30	1512	70	75	6
16	15	DOE07	JANE07	Drunkard Creek	New York	Female	Undergraduate	Math	US	21	1338	82	64	5
17	16	DOE08	JANE08	Mexican Hat	Utah	Female	Undergraduate	Econ	US	18	1821	80	63	3
18	17	DOE09	JANE09	Amsterdam	Holland	Female	Undergraduate	Math	Holland	19	1494	75	60	3
19	18	DOE10	JANE10	Mexico	Mexico	Female	Graduate	Politics	Mexico	31	2248	95	59	4
20	19	DOE11	JANE11	Caracas	Venezuela	Female	Undergraduate	Math	Venezuela	18	2252	92	68	5
21	20	DOE09	JOE09	San Juan	Puerto Rico	Male	Graduate	Politics	US	33	1923	95	63	7
22	21	DOE12	JANE12	Remote	Oregon	Female	Undergraduate	Econ	US	19	1727	67	62	7
23	22	DOE10	JOE10	New York	New York	Male	Undergraduate	Econ	US	21	1872	82	73	4
24	23	DOE13	JANE13	The X	Massachusetts	Female	Graduate	Politics	US	25	1767	89	68	6
25	24	DOE14	JANE14	Beijing	China	Female	Undergraduate	Math	China	18	1643	79	65	6
26	25	DOE11	JOE11	Stockholm	Sweden	Male	Undergraduate	Politics	Sweden	19	1919	88	64	4
27	26	DOE12	JOE12	Embarrass	Minnesota	Male	Graduate	Econ	US	28	1434	96	71	4
28	27	DOE13	JOE13	Intercourse	Pennsylvania	Male	Undergraduate	Math	US	20	2119	88	71	5
29	28	DOE15	JANE15	Loco	Oklahoma	Female	Undergraduate	Econ	US	20	2309	64	68	6
30	29	DOE14	JOE14	Buenos Aires	Argentina	Male	Graduate	Politics	Argentina	30	2279	85	72	3
31	30	DOE15	JOE15	Acme	Louisiana	Male	Undergraduate	Econ	US	19	1907	79	74	3

Path to the file: http://www.princeton.edu/~otorres/Stata/Students.xls

Excel to Stata (copy-and-paste)

1 - To go **from Excel to Stata** you simply copy-and-paste data into the Stata's "Data editor" which you can open by clicking on the icon that looks like this:

2 - This window will open, is the data editor



3 - Press Ctrl-v to paste the data…

Data Editor

Preserve | Restore | Sort | << | >> | Hide | Delete...

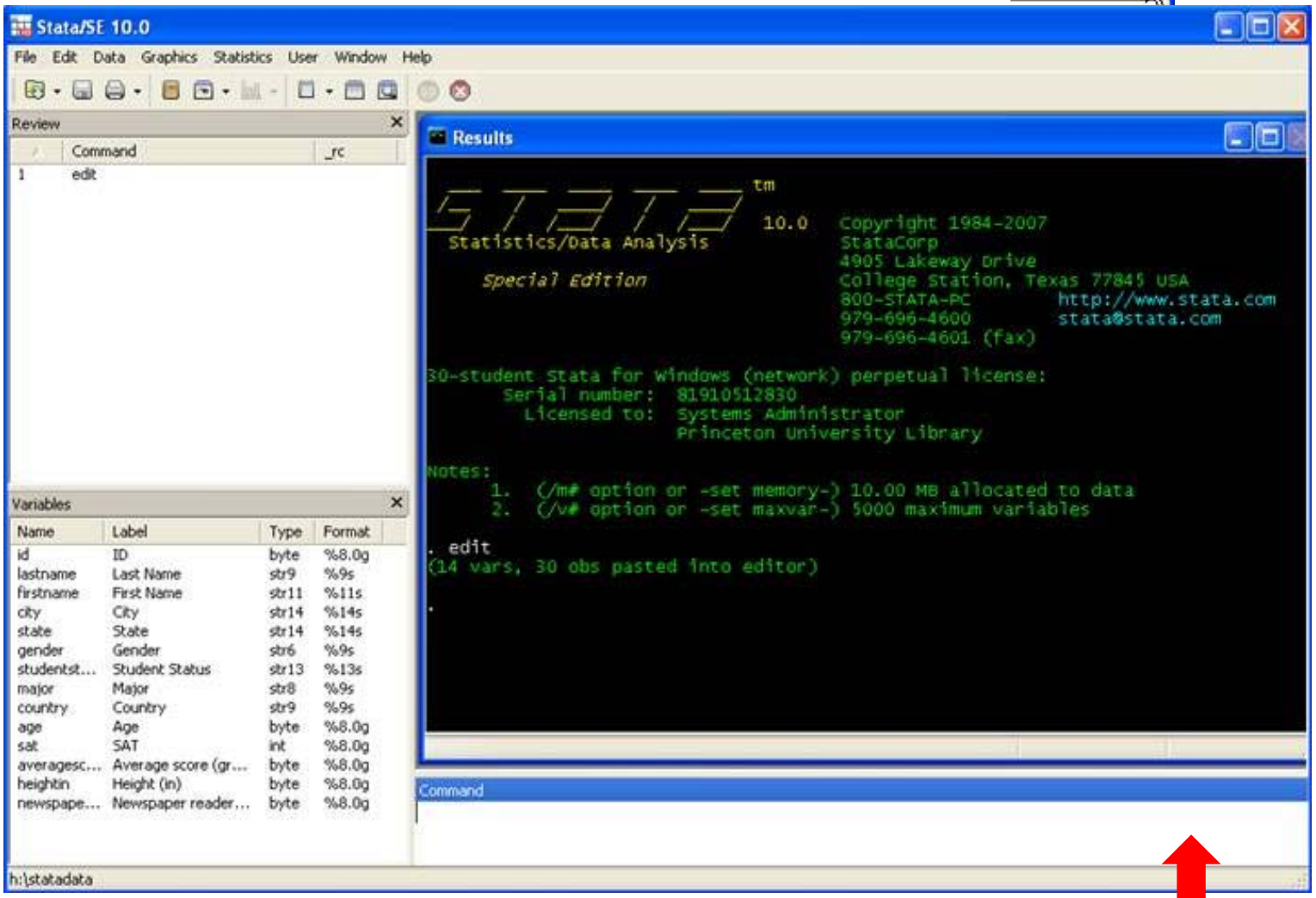
id[1] = 1

	id	lastname	firstname	city	state	gender	studentstatus	major	country	age	sat	averagesco~e	heightin	news
1	1	DOE01	JANE01	Los Angeles	California	Female	Graduate	Politics	US	30	2263	67	61	
2	2	DOE02	JANE02	Sedona	Arizona	Female	Undergraduate	Math	US	19	2006	63	64	
3	3	DOE01	JOE01	Elmira	New York	Male	Graduate	Math	US	26	2221	78	73	
4	4	DOE02	JOE02	Lackawana	New York	Male	Graduate	Econ	US	33	1716	78	68	
5	5	DOE03	JOE03	Defiance	Ohio	Male	Graduate	Econ	US	37	1701	65	71	
6	6	DOE04	JOE04	Tel Aviv	Israel	Male	Graduate	Econ	Israel	25	1786	69	67	
7	7	DOE05	JOE05	Cimax	North Carolina	Male	Graduate	Politics	US	39	1577	96	70	
8	8	DOE03	JANE03	Liberal	Kansas	Female	Undergraduate	Politics	US	21	1842	87	62	
9	9	DOE04	JANE04	Montreal	Canada	Female	Undergraduate	Math	Canada	18	1813	91	62	
10	10	DOE05	JANE05	New York	New York	Female	Graduate	Math	US	33	2041	71	66	
11	11	DOE06	JOE06	Hot Coffe	Mississippi	Male	Undergraduate	Econ	US	18	1787	82	67	
12	12	DOE06	JANE06	Java	Virginia	Female	Graduate	Math	US	38	1513	79	59	
13	13	DOE07	JOE07	Varna	Bulgaria	Male	Graduate	Politics	Bulgaria	30	1637	79	63	
14	14	DOE08	JOE08	Moscow	Russia	Male	Graduate	Politics	Russia	30	1512	70	75	
15	15	DOE07	JANE07	Drunkard Creek	New York	Female	Undergraduate	Math	US	21	1338	82	64	
16	16	DOE08	JANE08	Mexican Hat	Utah	Female	Undergraduate	Econ	US	18	1821	80	63	
17	17	DOE09	JANE09	Amsterdam	Holland	Female	Undergraduate	Math	Holland	19	1494	75	60	
18	18	DOE10	JANE10	Mexico	Mexico	Female	Graduate	Politics	Mexico	31	2248	95	59	
19	19	DOE11	JANE11	Caracas	Venezuela	Female	Undergraduate	Math	Venezuela	18	2252	92	68	
20	20	DOE09	JOE09	San Juan	Puerto Rico	Male	Graduate	Politics	US	33	1923	95	63	
21	21	DOE12	JANE12	Remote	Oregon	Female	Undergraduate	Econ	US	19	1727	67	62	
22	22	DOE10	JOE10	New York	New York	Male	Undergraduate	Econ	US	21	1872	82	73	
23	23	DOE13	JANE13	The X	Massachusetts	Female	Graduate	Politics	US	25	1767	89	68	
24	24	DOE14	JANE14	Beijing	China	Female	Undergraduate	Math	China	18	1643	79	65	
25	25	DOE11	JOE11	Stockholm	Sweden	Male	Undergraduate	Politics	Sweden	19	1919	88	64	
26	26	DOE12	JOE12	Embarrass	Minnesota	Male	Graduate	Econ	US	28	1434	96	71	
27	27	DOE13	JOE13	Intercourse	Pennsylvania	Male	Undergraduate	Math	US	20	2119	88	71	
28	28	DOE15	JANE15	Loco	Oklahoma	Female	Undergraduate	Econ	US	20	2309	64	68	
29	29	DOE14	JOE14	Buenos Aires	Argentina	Male	Graduate	Politics	Argentina	30	2279	85	72	
30	30	DOE15	JOE15	Acme	Louisiana	Male	Undergraduate	Econ	US	19	1907	79	74	

1 - Close the data editor by pressing the "X" button on the upper-right corner of the editor

NOTE: You need to close the data editor or data browser to continue working.

2 - The "Variables" window will show all the variables in your data



3 - Do not forget to save the file, in the command window type --- `save students, replace`

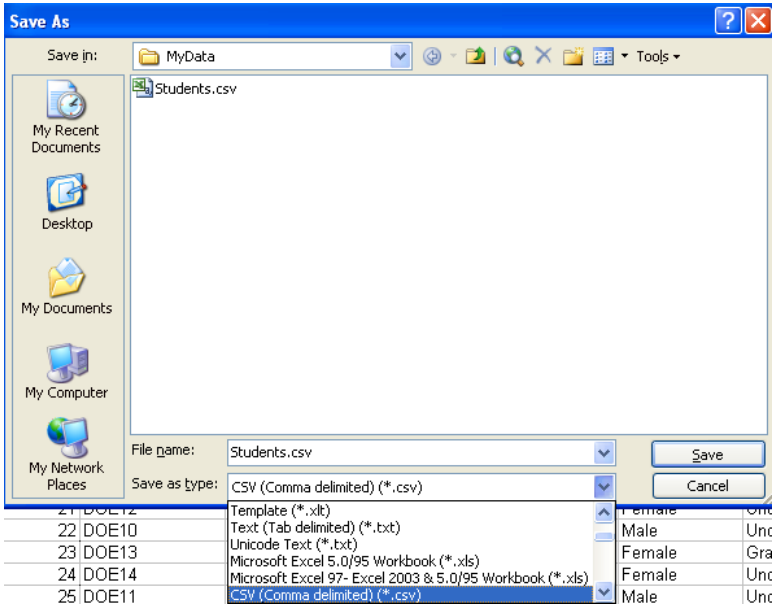
You can also use the menu, go to File – Save As

4 - This is what you will see in the output window, the data has been saved as `students.dta`

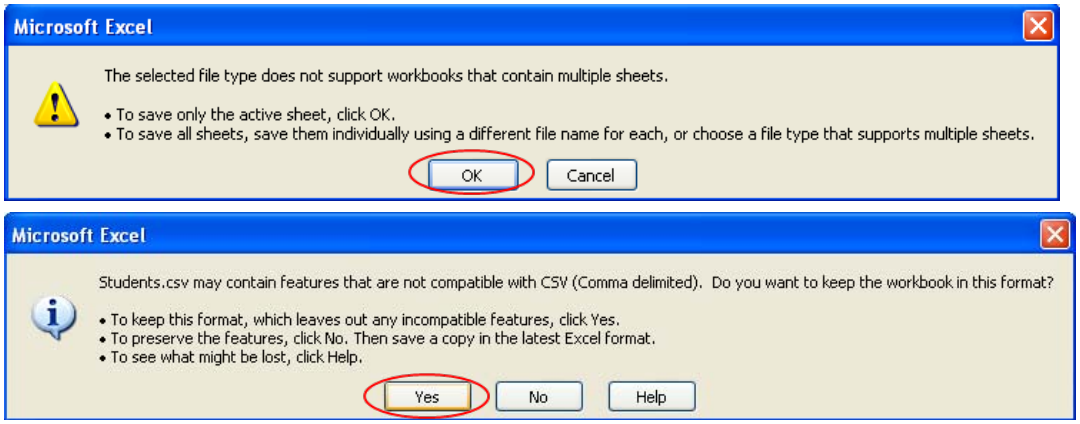
```
. save students, replace
(note: file students.dta not found)
file students.dta saved
```

Excel to Stata (`insheet using *.csv`)

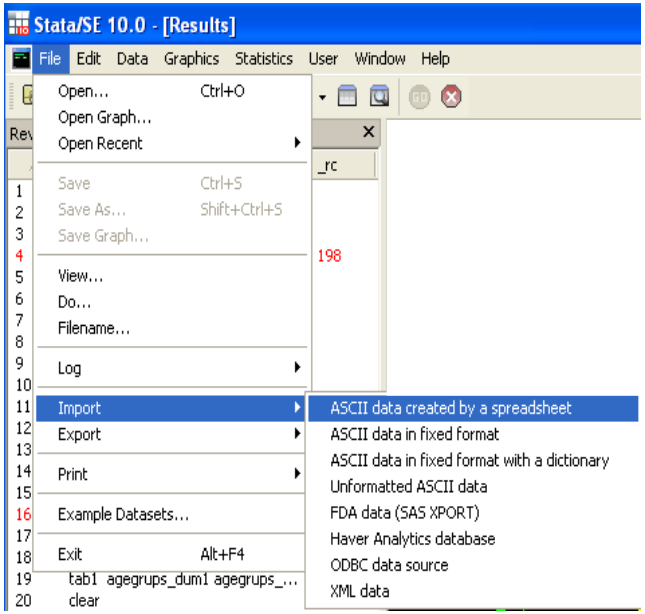
You can also **save the Excel file as *.csv** (comma-separated values) and import it in Stata. **In Excel** go to File-Save as.

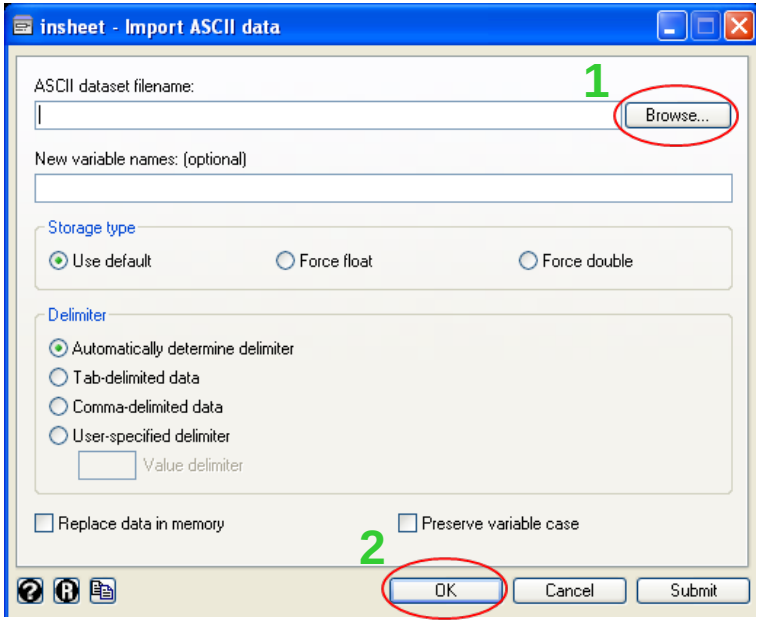


You may get the following messages, click OK and YES…



In Stata go to File-Import-ASCII data created by spreadsheet. Click on 'Browse' to find the file and then OK.





Command: `describe`

If you type `describe` in the command window you will get a general description of the data

```
. describe

Contains data from http://dss.princeton.edu/training/students.dta
  obs:            30
 vars:            14                          29 Sep 2009 17:12
 size:         2,580 (99.9% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
id              byte   %8.0g                  ID
lastname        str5   %9s                    Last Name
firstname       str6   %9s                    First Name
city            str14  %14s                   City
state           str14  %14s                   State
gender          str6   %9s                    Gender
studentstatus   str13  %13s                   Student Status
major           str8   %9s                    Major
country         str9   %9s                    Country
age             byte   %8.0g                  Age
sat             int    %8.0g                  SAT
averagescoreg~e byte   %8.0g                  Average score (grade
heightin        byte   %8.0g                  Height (in)
newspaperread~k byte   %8.0g                  Newspaper readership
-------------------------------------------------------------------------------
```

Type `help describe` in the command window for more information…

Command: summarize

Type summarize to get some basic statistics.

```
. summarize
```

Variable	Obs	Mean	Std. Dev.	Min	Max
id	30	15.5	8.803408	1	30
lastname	0				
firstname	0				
city	0				
state	0				
gender	0				
studentsta~s	0				
major	0				
country	0				
age	30	25.2	6.870226	18	39
sat	30	1848.9	275.1122	1338	2309
averagesco~e	30	80.36667	10.11139	63	96
heightin	30	66.43333	4.658573	59	75
newspaperr~k	30	4.866667	1.279368	3	7

Zeros indicate string variables

Use ‘min’ and ‘max’ values to check for a valid range in each variable. For example, age where no response is usually coded as 99 or 999.

Type help summarize in the command window for more information…

Exploring data: frequencies

Frequency refers to the number of times a value is repeated. Frequencies are used to analyze categorical data. The tables below are *frequency tables*, values are in ascending order. In Stata use the command `tab` (type `help tab` for more details)

variable

```
. tab major
```

Major	Freq.	Percent	Cum.
Econ	10	33.33	33.33
Math	10	33.33	66.67
Politics	10	33.33	100.00
Total	30	100.00	

'Freq.' provides a raw count of each value. In this case 10 students for each major.

'Percent' gives the relative frequency for each value. For example, 33.33% of the students in this group are econ majors.

'Cum.' is the cumulative frequency in ascending order of the values. For example, 66.67% of the students are econ or math majors.

variable

```
. tab readnews
```

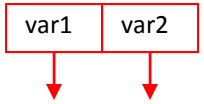
Newspaper readership (times/wk)	Freq.	Percent	Cum.
3	6	20.00	20.00
4	5	16.67	36.67
5	9	30.00	66.67
6	7	23.33	90.00
7	3	10.00	100.00
Total	30	100.00	

'Freq.' Here 6 students read the newspaper 3 days a week, 9 students read it 5 days a week.

'Percent'. Those who read the newspaper 3 days a week represent 20% of the sample, 30% of the students in the sample read the newspaper 5 days a week.

'Cum.' 66.67% of the students read the newspaper 3 to 5 days a week.

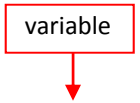
Exploring data: frequencies (using `table`)




```
. table major gender, contents(freq mean  age  mean sat mean score mean readnews)
```

Major	Gender Female	Male
Econ	3	7
	19	25.8571
	1952.33	1743.29
	70.3333	78.7143
	5.33333	4
Math	8	2
	23	23
	1762.5	2170
	79	83
	5.25	5.5
Politics	4	6
	26.75	30.1667
	2030	1807.83
	84.5	85.5
	5	4.83333

`table` is another command to produce frequencies and statistics. For more info and a list of all statistics type `help table`. Here are some examples.




```
. table gender, contents(freq mean age mean score)
```

Gender	Freq.	mean(age)	mean(score)
Female	15	23.2	78.7333
Male	15	27.2	82

The mean age of females is 23 years, for males is 27.
The mean score is 78 for females and 82 for males.

```
. table major, contents(freq mean  age  mean sat mean score mean readnews)
```

Major	Freq.	mean(age)	mean(sat)	mean(score)	mean(read~s)
Econ	10	23.8	1806	76.2	4.4
Math	10	23	1844	79.8	5.3
Politics	10	28.8	1896.7	85.1	4.9

Exploring data: crosstabs

Also known as *contingency tables*, help you to analyze the relationship between two or more variables (mostly categorical). Below is a crosstab between the variable 'ecostatu' and 'gender'. We use the command `tab` (but with two variables to make the crosstab).

Options 'column', 'row' gives you the column and row percentages.

var1 var2

```
. tab ecostatu gender, column row
```

Key
frequency *row percentage* *column percentage*

Status of Nat'l Eco	Gender of Respondent 1	2	Total
Very well	90 60.40 14.33	59 39.60 7.92	149 100.00 10.85
Fairly well	337 50.30 53.66	333 49.70 44.70	670 100.00 48.80
Fairly badly	139 39.94 22.13	209 60.06 28.05	348 100.00 25.35
Very badly	57 29.84 9.08	134 70.16 17.99	191 100.00 13.91
Not sure	2 16.67 0.32	10 83.33 1.34	12 100.00 0.87
Refused	3 100.00 0.48	0 0.00 0.00	3 100.00 0.22
Total	628 45.74 100.00	745 54.26 100.00	1,373 100.00 100.00

The first value in a cell tells you the number of observations for each xtab. In this case, 90 respondents are 'male' and said that the economy is doing 'very well', 59 are 'female' and believe the economy is doing 'very well'

The second value in a cell gives you row percentages for the first variable in the xtab. Out of those who think the economy is doing 'very well', 60.40% are males and 39.60% are females.

The third value in a cell gives you column percentages for the second variable in the xtab. Among males, 14.33% think the economy is doing 'very well' while 7.92% of females have the same opinion.

You can use `tab1` for multiple frequencies or `tab2` to run all possible crosstabs combinations. Type help tab for further details.

Exploring data: crosstabs (a closer look)

You can use crosstabs to compare responses among categories in relation to aggregate responses. In the table below we will see whether males and females have opinions similar to the national aggregate.

```
. tab ecostatu gender, column row
```

Key
frequency *row percentage* *column percentage*

Status of Nat'l Eco	Gender of Respondent 1	2	Total
Very well	90 60.40 14.33	59 39.60 7.92	149 100.00 10.85
Fairly well	337 50.30 53.66	333 49.70 44.70	670 100.00 48.80
Fairly badly	139 39.94 22.13	209 60.06 28.05	348 100.00 25.35
Very badly	57 29.84 9.08	134 70.16 17.99	191 100.00 13.91
Not sure	2 16.67 0.32	10 83.33 1.34	12 100.00 0.87
Refused	3 100.00 0.48	0 0.00 0.00	3 100.00 0.22
Total	628 45.74 100.00	745 54.26 100.00	1,373 100.00 100.00

As a rule-of-thumb, a margin of error of ±4 percentage points can be used to indicate a significant difference (some use ±3).

For example, rounding up the percentages, 11% (10.85) answer 'very well' at the national level. With the margin of error, this gives a range roughly between 7% and 15%, anything beyond this range could be considered significantly different (remember this is just an approximation). It does not appear to be a significant bias between males and females for this answer.

In the 'fairly well' category we have 49%, with range between 45% and 53%. The response for males is 54% and for females 45%. We could say here that males tend to be a bit more optimistic on the economy and females tend to be a bit less optimistic.

If we aggregate responses, we could get a better picture. In the table below 68% of males believe the economy is doing well (comparing to 60% at the national level, while 46% of females thing the economy is bad (comparing to 39% aggregate). Males seem to be more optimistic than females.

RECODE of ecostatu (Status of Nat'l Eco)	Gender of Respondent 1	2	Total
Well	427 52.14 67.99	392 47.86 52.62	819 100.00 59.65
Bad	196 36.36 31.21	343 63.64 46.04	539 100.00 39.26
Not sure/ref	5 33.33 0.80	10 66.67 1.34	15 100.00 1.09
Total	628 45.74 100.00	745 54.26 100.00	1,373 100.00 100.00

```
recode ecostatu (1 2 = 1 "Well") (3 4 = 2 "Bad") (5 6=3 "Not sure/ref"), gen(ecostatu1) label(eco)
```

Exploring data: crosstabs (test for associations)

To see whether there is a relationship between two variables you can choose a number of tests. Some apply to nominal variables some others to ordinal. I am running all of them here for presentation purposes.

```
tab ecostatu1 gender, column row nokey chi2 lrchi2 V exact gamma taub
```

Likelihood-ratio χ^2(chi-square) | *Goodman & Kruskal's γ (gamma)* | X^2(chi-square) | Cramer's *V* | Kendall's τ_b (tau-b) | Fisher's exact test

```
. tab ecostatu1 gender, column row nokey chi2 lrchi2 V exact gamma taub

Enumerating sample-space combinations:
stage 3:  enumerations = 1
stage 2:  enumerations = 16
stage 1:  enumerations = 0
```

RECODE of ecostatu (Status of Nat'l Eco)	Gender of Respondent 1	2	Total
Well	427 52.14 67.99	392 47.86 52.62	819 100.00 59.65
Bad	196 36.36 31.21	343 63.64 46.04	539 100.00 39.26
Not sure/ref	5 33.33 0.80	10 66.67 1.34	15 100.00 1.09
Total	628 45.74 100.00	745 54.26 100.00	1,373 100.00 100.00

```
          Pearson chi2(2) =   33.5266   Pr = 0.000
 likelihood-ratio chi2(2) =   33.8162   Pr = 0.000
                Cramér's V =   0.1563
                     gamma =   0.3095   ASE = 0.050
           Kendall's tau-b =   0.1553   ASE = 0.026
            Fisher's exact =                 0.000
```

- For *nominal* data use chi2, lrchi2, V
- For *ordinal* data use gamma and taub
- Use `exact` instead of `chi2` when frequencies are less than 5 across the table.

X^2(chi-square) tests for relationships between variables. The null hypothesis (Ho) is that there is no relationship. To reject this we need a Pr < 0.05 (at 95% confidence). Here both chi2 are significant. Therefore we conclude that there is some relationship between perceptions of the economy and gender

Cramer's *V* is a measure of association between two nominal variables. It goes from 0 to 1 where 1 indicates strong association (for rXc tables). In 2x2 tables, the range is -1 to 1. Here the V is 0.15, which shows a small association.

Gamma and taub are measures of association between two ordinal variables (both have to be in the same direction, i.e. negative to positive, low to high). Both go from -1 to 1. Negative shows inverse relationship, closer to 1 a strong relationship. Gamma is recommended when there are lots of ties in the data. Taub is recommended for square tables.

Fisher's exact test is used when there are very few cases in the cells (usually less than 5). It tests the relationship between two variables. The null is that variables are independent. Here we reject the null and conclude that there is some kind of relationship between variables

Exploring data: descriptive statistics

For continuous data we use descriptive statistics. These statistics are a collection of measurements of two things: *location* and *variability*. Location tells you the central value of your variables (the mean is the most common measure of this) . Variability refers to the spread of the data from the center value (i.e. variance, standard deviation). Statistics is basically the study of what causes such variability. We use the command `tabstat` to get these stats.

```
tabstat age sat score heightin readnews, s(mean median sd var count range min max)
```

```
. tabstat  age sat score heightin readnews, s(mean median sd var count range min max)
```

stats	age	sat	score	heightin	readnews
mean	25.2	1848.9	80.36667	66.43333	4.866667
p50	23	1817	79.5	66.5	5
sd	6.870226	275.1122	10.11139	4.658573	1.279368
variance	47.2	75686.71	102.2402	21.7023	1.636782
N	30	30	30	30	30
range	21	971	33	16	4
min	18	1338	63	59	3
max	39	2309	96	75	7

- The *mean* is the sum of the observations divided by the total number of observations.
- The *median* (p50 in the table above) is the number in the middle . To get the median you have to order the data from lowest to highest. If the number of cases is odd the median is the single value, for an even number of cases the median is the average of the two numbers in the middle.
- The *standard deviation* is the squared root of the variance. Indicates how close the data is to the mean. Assuming a normal distribution, 68% of the values are within 1 sd from the mean, 95% within 2 sd and 99% within 3 sd
- The *variance* measures the dispersion of the data from the mean. It is the simple mean of the squared distance from the mean.
- *Count* (N in the table) refers to the number of observations per variable.
- *Range* is a measure of dispersion. It is the difference between the largest and smallest value, max – min.
- *Min* is the lowest value in the variable.
- *Max* is the largest value in the variable.

Exploring data: descriptive statistics

You could also estimate descriptive statistics by subgroups. For example, by gender below

```
tabstat  age sat score heightin readnews, s(mean median sd var count range min max) by(gender)
```

```
. tabstat  age sat score heightin readnews, s(mean median sd var count range min max) by(gender)

Summary statistics: mean, p50, sd, variance, N, range, min, max
  by categories of: gender (Gender)
```

gender	age	sat	score	heightin	readnews
Female	23.2	1871.8	78.73333	63.4	5.2
	20	1821	79	63	5
	6.581359	307.587	10.66012	3.112188	1.207122
	43.31429	94609.74	113.6381	9.685714	1.457143
	15	15	15	15	15
	20	971	32	9	4
	18	1338	63	59	3
	38	2309	95	68	7
Male	27.2	1826	82	69.46667	4.533333
	28	1787	82	71	4
	6.773899	247.0752	9.613978	3.943651	1.302013
	45.88571	61046.14	92.42857	15.55238	1.695238
	15	15	15	15	15
	21	845	31	12	4
	18	1434	65	63	3
	39	2279	96	75	7
Total	25.2	1848.9	80.36667	66.43333	4.866667
	23	1817	79.5	66.5	5
	6.870226	275.1122	10.11139	4.658573	1.279368
	47.2	75686.71	102.2402	21.7023	1.636782
	30	30	30	30	30
	21	971	33	16	4
	18	1338	63	59	3
	39	2309	96	75	7

Type `help tabstat` for more options.

Examples of frequencies and crosstabulations

Frequencies (`tab` command)

```
. tab gender
```

Gender	Freq.	Percent	Cum.
Female	15	50.00	50.00
Male	15	50.00	100.00
Total	30	100.00	

In this sample we have 15 females and 15 males. Each represents 50% of the total cases.

```
. tab major
```

Major	Freq.	Percent	Cum.
Econ	10	33.33	33.33
Math	10	33.33	66.67
Politics	10	33.33	100.00
Total	30	100.00	

```
. tab  studentstatus
```

Student Status	Freq.	Percent	Cum.
Graduate	15	50.00	50.00
Undergraduate	15	50.00	100.00
Total	30	100.00	

Averarge SAT scores by gender and major

Crosstabulations

```
. tab  gender studentstatus, col row
```

Key

frequency
row percentage
column percentage

Gender	Student Status Graduate	Undergrad	Total
Female	5 33.33 33.33	10 66.67 66.67	15 100.00 50.00
Male	10 66.67 66.67	5 33.33 33.33	15 100.00 50.00
Total	15 50.00 100.00	15 50.00 100.00	30 100.00 100.00

```
. tab gender major, sum(sat)
```

Means, Standard Deviations and Frequencies of SAT

Gender	Major Econ	Math	Politics	Total
Female	1952.3333 312.43773 3	1762.5 317.99326 8	2030 262.25052 4	1871.8 307.58697 15
Male	1743.2857 155.6146 7	2170 72.124892 2	1807.8333 288.99994 6	1826 247.07518 15
Total	1806 219.16559 10	1844 329.76928 10	1896.7 287.20687 10	1848.9 275.11218 30

More examples of frequencies and crosstabulations

```
. tab  studentstatus gender, col row
```

Key

frequency
row percentage
column percentage

Student Status	Gender Female	Male	Total
Graduate	5 33.33 33.33	10 66.67 66.67	15 100.00 50.00
Undergraduate	10 66.67 66.67	5 33.33 33.33	15 100.00 50.00
Total	15 50.00 100.00	15 50.00 100.00	30 100.00 100.00

```
. tab country
```

Country	Freq.	Percent	Cum.
Argentina	1	3.33	3.33
Bulgaria	1	3.33	6.67
Canada	1	3.33	10.00
China	1	3.33	13.33
Holland	1	3.33	16.67
Israel	1	3.33	20.00
Mexico	1	3.33	23.33
Russia	1	3.33	26.67
Sweden	1	3.33	30.00
US	20	66.67	96.67
Venezuela	1	3.33	100.00
Total	30	100.00	

```
. bysort  studentstatus: tab gender major, sum(sat)
```

-> studentstatus = Graduate

Means, Standard Deviations and Frequencies of SAT

Gender	Major Econ	Math	Politics	Total
Female	. . 0	1777 373.35238 2	2092.6667 282.13531 3	1966.4 323.32924 5
Male	1659.25 154.66819 4	2221 0 1	1785.6 317.32286 5	1778.6 284.3086 10
Total	1659.25 154.66819 4	1925 367.97826 3	1900.75 324.8669 8	1841.2 300.38219 15

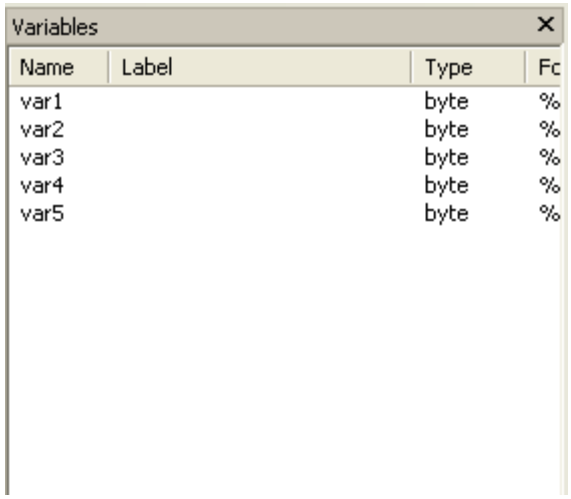
-> studentstatus = Undergraduate

Means, Standard Deviations and Frequencies of SAT

Gender	Major Econ	Math	Politics	Total
Female	1952.3333 312.43773 3	1757.6667 337.01197 6	1842 0 1	1824.5 305.36872 10
Male	1855.3333 61.711695 3	2119 0 1	1919 0 1	1920.8 122.23011 5
Total	1903.8333 208.30979 6	1809.2857 336.59952 7	1880.5 54.447222 2	1856.6 257.72682 15

Average SAT scores by gender and major for graduate and undergraduate students

Before

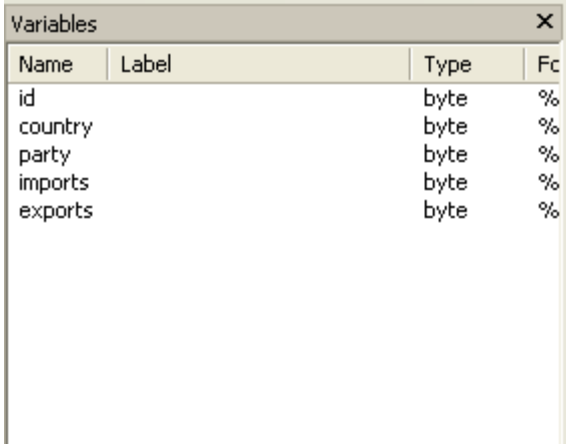


Renaming variables, type:

rename [old name] [new name]

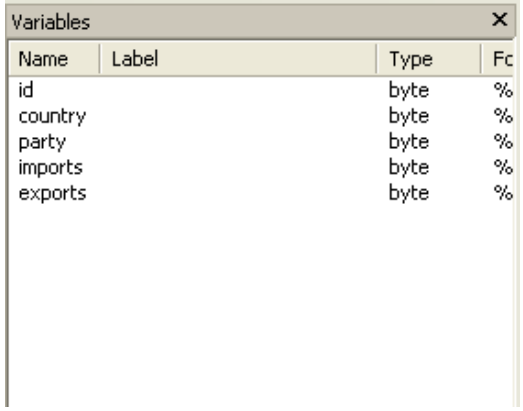
```
rename var1 id
rename var2 country
rename var3 party
rename var4 imports
rename var5 exports
```

After



Adding/changing variable labels, type:

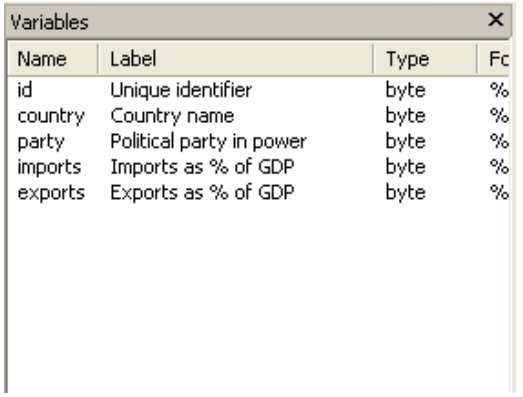
Before



label variable [var name] “Text”

```
label variable id "Unique identifier"
label variable country "Country name"
label variable party "Political party in power"
label variable imports "Imports as % of GDP"
label variable exports "Exports as % of GDP"
```

After



Assigning value labels

Adding labels to each category in a variable is a two step process in Stata.

Step 1: You need to create the labels using `label define`, type:

```
label define label1 1 "Agree" 2 "Disagree" 3 "Do not know"
```

Setp 2: Assign that label to a variable with those categories using `label values`:

```
label values var1 label1
```

If another variable has the same corresponding categories you can use the same label, type

```
label values var2 label1
```

Verify by running frequencies for var1 and var2 (using `tab`)

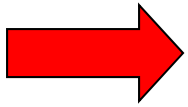
If you type `labelbook` it will list all the labels in the datafile.

NOTE: Defining labels is not the same as creating variables

Creating new variables

To generate a new variable use the command `generate` (gen for short), type
`generate [newvar] = [expression]`

```
generate score2 = score/100
generate readnews2 = readnews*4
```

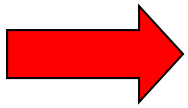


… results for the first five students…

score	height	readnews	score2	readnews2
67	61	5	.67	20
63	64	7	.63	28
78	73	6	.78	24
78	68	3	.78	12
65	71	6	.65	24

You can use `generate` to create constant variables. For example:

```
generate x = 5
generate y = 4*15
generate z = y/x
```

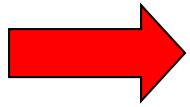


… results for the first five students…

x	y	z
5	60	12
5	60	12
5	60	12
5	60	12
5	60	12

You can also use `generate` with string variables. For example:

```
generate fullname = last + ", " + first
label variable fullname "Student full name"
browse id fullname last first
```



… results for the first five students…

id	fullname	last	first
1	DOE01, JANE01	DOE01	JANE01
2	DOE02, JANE02	DOE02	JANE02
3	DOE01, JOE01	DOE01	JOE01
4	DOE02, JOE02	DOE02	JOE02
5	DOE03, JOE03	DOE03	JOE03

Creating variables from a combination of other variables

To generate a new variable as a conditional from other variables type:

```
generate newvar=(var1==1 & var2==1)
generate newvar=(var1==1 & var2<26)
```

NOTE: & = and, | = or

```
. gen fem_grad=(gender==1 & status==1)

. tab fem_grad
```

fem_grad	Freq.	Percent	Cum.
0	25	83.33	83.33
1	5	16.67	100.00
Total	30	100.00	

```
. tab gender status
```

Gender	Student Status Graduate	Undergrad	Total
Female	5	10	15
Male	10	5	15
Total	15	15	30

```
. gen fem_less25=(gender==1 & age<26)

. tab  fem_less25
```

fem_less25	Freq.	Percent	Cum.
0	19	63.33	63.33
1	11	36.67	100.00
Total	30	100.00	

```
. tab  age gender
```

Age	Gender Female	Male	Total
18	4	1	5
19	3	2	5
20	1	1	2
21	2	1	3
25	1	1	2
26	0	1	1
28	0	1	1
30	1	3	4
31	1	0	1
33	1	2	3
37	0	1	1
38	1	0	1
39	0	1	1
Total	15	15	30

1.- Recoding ‘age’ into three groups.

```
. tab age
```

Age	Freq.	Percent	Cum.
18	5	16.67	16.67
19	5	16.67	33.33
20	2	6.67	40.00
21	3	10.00	50.00
25	2	6.67	56.67
26	1	3.33	60.00
28	1	3.33	63.33
30	4	13.33	76.67
31	1	3.33	80.00
33	3	10.00	90.00
37	1	3.33	93.33
38	1	3.33	96.67
39	1	3.33	100.00
Total	30	100.00	

2.- Use `recode` command, type

```
recode age (18 19 = 1 “18 to 19”) (20/28 = 2 “20 to 29”) (30/39 = 3 “30 to 39”) (else=.),
generate(agegroups) label(agegroups)
```

3.- The new variable is called ‘`agegroups`’:

```
. tab  agegroups
```

RECODE of age (Age)	Freq.	Percent	Cum.
18 to 19	10	33.33	33.33
20 to 29	9	30.00	63.33
30 to 39	11	36.67	100.00
Total	30	100.00	

Recoding variables using **egen**

You can recode variables using the command `egen` and options `cut/group`.

```
egen [newvar] = cut (oldvar), at (break1, break2, break3, etc.)
```

Notice that the breaks show ranges. Below we type four breaks. The first starts at 18 and ends before 20, the second starts at 20 and ends before 30, the third starts at 30 and ends before 40.

```
. egen agegroups2=cut(age), at(18, 20, 30, 40)

. tab agegroups2
```

agegroups2	Freq.	Percent	Cum.
18	10	33.33	33.33
20	9	30.00	63.33
30	11	36.67	100.00
Total	30	100.00	

You could also use the option `group,` which specifies groups with equal frequency (you have to add value labels:

```
egen [newvar] = cut (oldvar), group(number of groups)
```

```
. egen agegroups3=cut(age), group(3)

. tab agegroups3
```

agegroups3	Freq.	Percent	Cum.
0	10	33.33	33.33
1	9	30.00	63.33
2	11	36.67	100.00
Total	30	100.00	

For more details and options type `help egen`

Changing variable values (replace)

Before

```
. tab inc
```

inc	Freq.	Percent	Cum.
1	6	11.76	11.76
2	7	13.73	25.49
3	6	11.76	37.25
4	6	11.76	49.02
5	7	13.73	62.75
6	6	11.76	74.51
7	7	13.73	88.24
99	6	11.76	100.00
Total	51	100.00	

```
replace inc = . If inc==99
```

After

```
. tab inc
```

inc	Freq.	Percent	Cum.
1	6	13.33	13.33
2	7	15.56	28.89
3	6	13.33	42.22
4	6	13.33	55.56
5	7	15.56	71.11
6	6	13.33	84.44
7	7	15.56	100.00
Total	45	100.00	

Before

```
. tab inc
```

inc	Freq.	Percent	Cum.
1	6	11.76	11.76
2	7	13.73	25.49
3	6	11.76	37.25
4	6	11.76	49.02
5	7	13.73	62.75
6	6	11.76	74.51
7	7	13.73	88.24
99	6	11.76	100.00
Total	51	100.00	

```
replace inc = . If inc>5
```

After

```
. tab inc
```

inc	Freq.	Percent	Cum.
1	6	18.75	18.75
2	7	21.88	40.63
3	6	18.75	59.38
4	6	18.75	78.13
5	7	21.88	100.00
Total	32	100.00	

Before

```
. tab inc
```

inc	Freq.	Percent	Cum.
1	6	18.75	18.75
2	7	21.88	40.63
3	6	18.75	59.38
4	6	18.75	78.13
5	7	21.88	100.00
Total	32	100.00	

```
replace inc = 999 If inc==5
```

After

```
. tab inc
```

inc	Freq.	Percent	Cum.
1	6	18.75	18.75
2	7	21.88	40.63
3	6	18.75	59.38
4	6	18.75	78.13
999	7	21.88	100.00
Total	32	100.00	

Extracting characters from regular expressions

To remove strings from var1 below use the following command

```
gen var2=regexr(var1,"[.\}\)\*a-zA-Z]+","")

destring var2, replace
```

```
. list var1 var2
```

	var1	var2
1.	123A33	12333
2.	2144F	2144
3.	2312A	2312
4.	3567754G	3567754
5.	35457S	35457
6.	34234N	34234
7.	234212*	234212
8.	23146}	23146
9.	31231)	31231
10.	AFN.345	345
11.	NYSE.12	12

To extract strings from a combination of strings and numbers

```
gen var2=regexr(var1,"[.0-9]+","")
```

```
. list var1 var2
```

	var1	var2
1.	AFM.123	AFM
2.	ADGT.2345	ADGT
3.	ACDET.1234564	ACDET
4.	CDFGEEGY.596544	CDFGEEGY
5.	ACGETYF.1235	ACGETYF

More info see: http://www.ats.ucla.edu/stat/stata/faq/regex.htm

Indexing is probably one of the most useful characteristics of Stata.

Using **_n**, you can create a unique identifier for each case in your data, type

```
. generate idall = _n
. move idall id
. label variable idall "General student ID"
```

Check the results in the data editor, 'idall' is equal to 'id'

	idall	id
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

Using **_N** you can also create a variable with the total number of cases in your dataset:

```
. generate total = _N
. move total idall
. label variable total "Number of students in the sample"
```

Check the results in the data editor:

	total	idall	id
1	30	1	1
2	30	2	2
3	30	3	3
4	30	4	4
5	30	5	5

Indexing: creating ids by categories

We can create id by categories. For example, lets create an id by `major`.

```
. sort major
. by major: gen idmajor = _n
. browse major idmajor
```

Check the results in the data editor:

	major	idmajor
1	Econ	1
2	Econ	2
3	Econ	3
4	Econ	4
5	Econ	5
6	Econ	6
7	Econ	7
8	Econ	8
9	Econ	9
10	Econ	10
11	Math	1
12	Math	2
13	Math	3
14	Math	4
15	Math	5
16	Math	6
17	Math	7
18	Math	8
19	Math	9
20	Math	10
21	Politics	1
22	Politics	2
23	Politics	3
24	Politics	4
25	Politics	5
26	Politics	6
27	Politics	7
28	Politics	8
29	Politics	9
30	Politics	10

First we have to `sort` the data by the variable on which we are basing the id (`major` in this case).

Then we use the command by to tell Stata that we are using `major` as the base variable (notice the colon).

Then we use `browse` to check the two variables.

You can create lagged values with _**n** . Lets rename 'idall' as 'months' (time variable) and will create a lagged variable containing the value of the previous case:

```
. rename idall months

. generate lag_months = months[_n-1]
(1 missing value generated)

. order months lag_months total
```

Check the results in the data editor:

	months	lag_months
1	1	.
2	2	1
3	3	2
4	4	3
5	5	4
6	6	5
7	7	6
8	8	7

If you want to lag more than one period just change [_n-1] to [_n-2] for a lag of two periods, [_n-3] for three, etc.

A more advance alternative to create lags uses the "L" operand within a time series setting (`tset` command must be specified first)

```
. tset months
        time variable:  months, 1 to 30
                delta:  1 unit

. generate lag_months1 = l1.months
(1 missing value generated)
```

You can create forward values with _**n**:

```
. generate for_months = months[_n+1]
(1 missing value generated)

. order months for_months lag_months
```

Check the results in the data editor:

	months	for_months	lag_months
1	1	2	.
2	2	3	1
3	3	4	2
4	4	5	3
5	5	6	4
6	6	7	5
7	7	8	6
8	8	9	7

A more advance alternative uses the "F" operand within a time series setting (`tset` command must be specified first)

```
. tset months
        time variable:  months, 1 to 30
                delta:  1 unit

. generate for_months1 = f1.months
(1 missing value generated)
```

NOTE: Notice the square brackets

Indexing: countdown and specific values

Combining _**n** and _**N** you can create a countdown variable.

```
. generate reverseid = id[_N - _n+1]
. order id reverseid
```

Check the results in the data editor:

	id	reverseid
1	1	30
2	2	29
3	3	28
4	4	27
5	5	26
6	6	25
7	7	24

You can create a variable based on one value of another variable. For example, lets create a variable with the highest SAT value in the sample.

```
. sort sat
. generate highestSAT = sat[_N]
. browse sat highestSAT
```

Check the results in the data editor:

	sat	highestSAT
1	1338	2309
2	1434	2309
3	1494	2309
4	1512	2309
5	1513	2309
25	2221	2309
26	2248	2309
27	2252	2309
28	2263	2309
29	2279	2309
30	2309	2309

NOTE: You could get the same result without sorting by using egen and the max function

```
. egen highestSAT1 = max(sat)
```

Sorting

Before

	last	first	city
1	DOE01	JANE01	Los Angeles
2	DOE02	JANE02	Sedona
3	DOE01	JOE01	Elmira
4	DOE02	JOE02	Lackawana
5	DOE03	JOE03	Defiance
6	DOE04	JOE04	Tel Aviv
7	DOE05	JOE05	Cimax

```
sort var1 var2 …
```

```
. sort city
. browse  last first city
```

After

	last	first	city
1	DOE15	JOE15	Acme
2	DOE09	JANE09	Amsterdam
3	DOE14	JANE14	Beijing
4	DOE14	JOE14	Buenos Aires
5	DOE11	JANE11	Caracas
6	DOE05	JOE05	Cimax
7	DOE03	JOE03	Defiance

`Gsort` is another command to sort data. The difference between `gsort` and `sort` is that with `gsort` you can sort in ascending or descending order, while with `sort` you can sort only in ascending order. Use +/- to indicate whether you want to sort in ascending/descending order. Here are some examples:

```
. gsort -id
. browse  id last first city
```

	id	last	first	city
1	30	DOE15	JOE15	Acme
2	29	DOE14	JOE14	Buenos Aires
3	28	DOE15	JANE15	Loco
4	27	DOE13	JOE13	Intercourse
5	26	DOE12	JOE12	Embarrass
6	25	DOE11	JOE11	Stockholm
7	24	DOE14	JANE14	Beijing

```
. gsort +major -sat
. browse  id last first major sat
```

	id	last	first	major	sat
1	28	DOE15	JANE15	Econ	2309
2	30	DOE15	JOE15	Econ	1907
3	22	DOE10	JOE10	Econ	1872
4	16	DOE08	JANE08	Econ	1821
5	11	DOE06	JOE06	Econ	1787
6	6	DOE04	JOE04	Econ	1786
7	21	DOE12	JANE12	Econ	1727
8	4	DOE02	JOE02	Econ	1716
9	5	DOE03	JOE03	Econ	1701
10	26	DOE12	JOE12	Econ	1434
11	19	DOE11	JANE11	Math	2252
12	3	DOE01	JOE01	Math	2221
13	27	DOE13	JOE13	Math	2119
14	10	DOE05	JANE05	Math	2041
15	2	DOE02	JANE02	Math	2006
16	9	DOE04	JANE04	Math	1813
17	24	DOE14	JANE14	Math	1643
18	12	DOE06	JANE06	Math	1513
19	17	DOE09	JANE09	Math	1494
20	15	DOE07	JANE07	Math	1338
21	29	DOE14	JOE14	Politics	2279
22	1	DOE01	JANE01	Politics	2263
23	18	DOE10	JANE10	Politics	2248
24	20	DOE09	JOE09	Politics	1923
25	25	DOE11	JOE11	Politics	1919
26	8	DOE03	JANE03	Politics	1842
27	23	DOE13	JANE13	Politics	1767
28	13	DOE07	JOE07	Politics	1637
29	7	DOE05	JOE05	Politics	1577
30	14	DOE08	JOE08	Politics	1512

Deleting variables

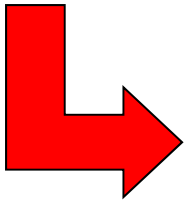
Use `drop` to delete variables and `keep` to keep them

Before

Variables

Name	Label
id	Student ID
reverseid	
months	General student ID
for_months	
lag_months	
total	Number of students in the sample
fullname	Student full name
last	Student last name
first	Student first name
city	City
state	State
gender	Gender
status	Status: grad or undergad
major	Major
country	Country
age	Age
sat	SAT
score	Average score (grade)
height	Height (in)
readnews	Newspaper read / week
score2	Score in decimals
readnews2	Monthly readership
x	
y	
z	
agegroups	Age by groups
agegroups2	
agegroups3	
highestSAT	
highestSAT1	
idmajor	
lag_months1	
for_months1	

```
. drop  reverseid for_months lag_months x y z agegroups2 agegroups3
. drop highestSAT highestSAT1 idmajor lag_months1 for_months1
```

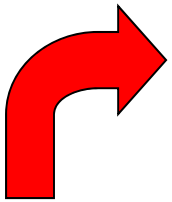


After

Variables

Name	Label
id	Student ID
months	General student ID
total	Number of students in the sample
fullname	Student full name
last	Student last name
first	Student first name
city	City
state	State
gender	Gender
status	Status: grad or undergad
major	Major
country	Country
age	Age
sat	SAT
score	Average score (grade)
height	Height (in)
readnews	Newspaper read / week
score2	Score in decimals
readnews2	Monthly readership
agegroups	Age by groups

Or



```
. keep  id months total-readnews2 agegroups
```

Notice the dash between ‘total’ and ‘readnews2’, you can use this format to indicate a list so you do not have to type in the name of all the variables

Deleting cases (selectively)

You can drop cases selectively using the conditional "if", for example

```
drop if var1==1  /*This will drop observations (rows)
                   where gender =1*/
drop if age>40 /*This will drop observation where
                 age>40*/
```

Alternatively, you can keep options you want

```
keep if var1==1
keep if age<40
keep if country==7 | country==13
keep if state==“New York” | state==“New Jersey”
```

| = "or", & = "and"

For more details type `help keep` or `help drop`.

Merge/Append

MERGE - You merge when you want to add more variables to an existing dataset.

(type `help merge` in the command window for more details)

What you need:

- Both files must be in Stata format
- Both files should have ***at least*** one variable in common (id)

Step 1. You need to sort the data by the id or ids common to both files you want to merge. For both datasets type:

- `sort [id1] [id2] …`
- `save [datafile name], replace`

Step 2. Open the master data (main dataset you want to add more variables to, for example `data1.dta`) and type:

- `merge [id1] [id2] … using [i.e. data2.dta]`

For example, opening a hypothetical data1.dta we type

- `merge lastname firstname using data2.dta`

To verify the merge type

- `tab _merge`

Here are the codes for _merge:

```
_merge==1    obs. from master data
_merge==2    obs. from only one using dataset
_merge==3    obs. from at least two datasets, master or using
```

If you want to keep the observations common to both datasets you can drop the rest by typing:

- `drop if _merge!=3  /*This will drop observations where _merge is not equal to 3 */`

APPEND - You append when you want to add more cases (more rows to your data, type `help append` for more details).

Open the master file (i.e. data1.dta) and type:

- `append using [i.e. data2.dta]`

Merging fuzzy text (reclink)

RECLINK - Matching fuzzy text. Reclink stands for 'record linkage'. It is a program written by Michael Blasnik to merge imperfect string variables. For example

Data1	Data2
Princeton University	Princeton U

Reclink helps you to merge the two databases by using a matching algorithm for these types of variables. Since it is a user created program, you may need to install it by typing `ssc install reclink.` Once installed you can type `help reclink` for details

As in merge, the merging variables must have the same name: state, university, city, name, etc. Both the master and the using files should have an id variable identifying each observation.

Note: the name of ids must be different, for example id1 (id master) and id2 (id using). Sort both files by the matching (merging) variables. The basic sytax is:

```
reclink var1 var2 var3 … using myusingdata, gen(myscore) idm(id1) idu(id2)
```

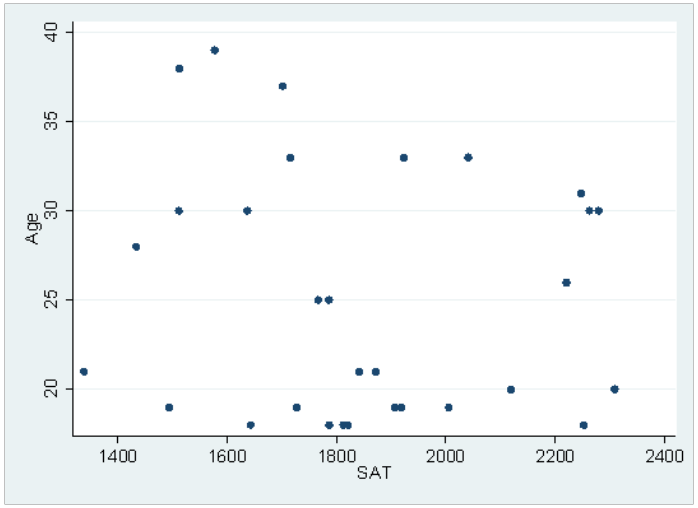
The variable `myscore` indicates the strength of the match; a perfect match will have a score of 1. Description (from reclink help pages):

> "reclink uses record linkage methods to match observations between two datasets where no perfect key fields exist -- essentially a fuzzy merge. reclink allows for user-defined matching and non-matching weights for each variable and employs a bigram string comparator to assess imperfect string matches.
>
> The master and using datasets must each have a variable that uniquely identifies observations. Two new variables are created, one to hold the matching score (scaled 0-1) and one for the merge variable. In addition, all of the matching variables from the using dataset are brought into the master dataset (with newly prefixed names) to allow for manual review of matches."

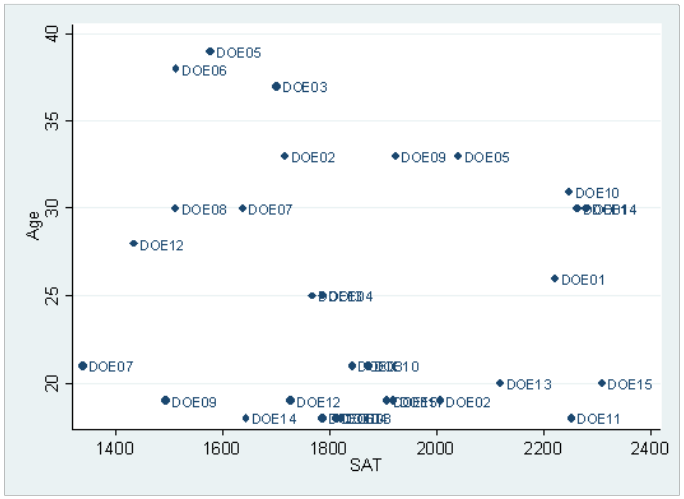
Graphs: scatterplot

Scatterplots are good to explore possible relationships or patterns between variables. Lets see if there is some relationship between age and SAT scores. For many more bells and whistles type `help scatter` in the command window.

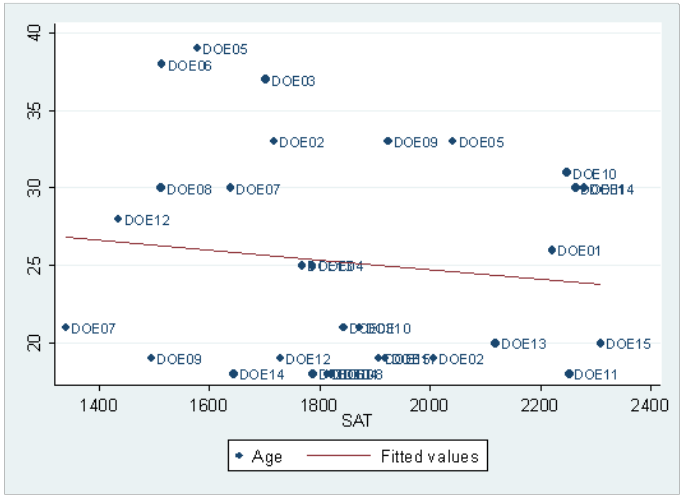
`twoway scatter` *`age sat`*



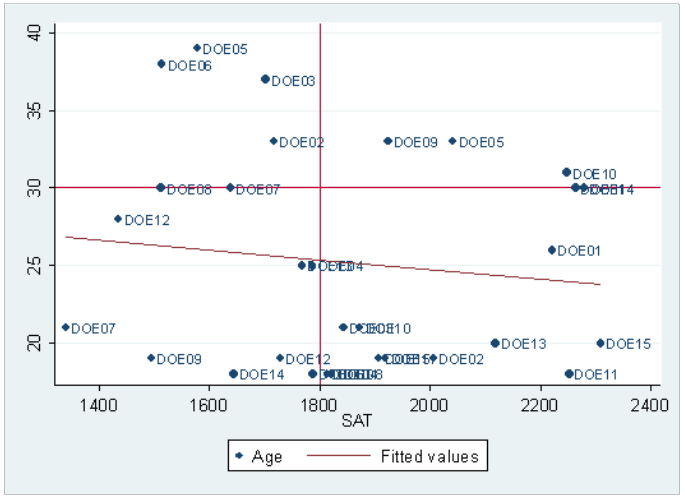
`twoway scatter` *`age sat`*`, mlabel(`*`last`*`)`



`twoway scatter` *`age sat`*`, mlabel(`*`last`*`) ||`
`lfit` *`age sat`*

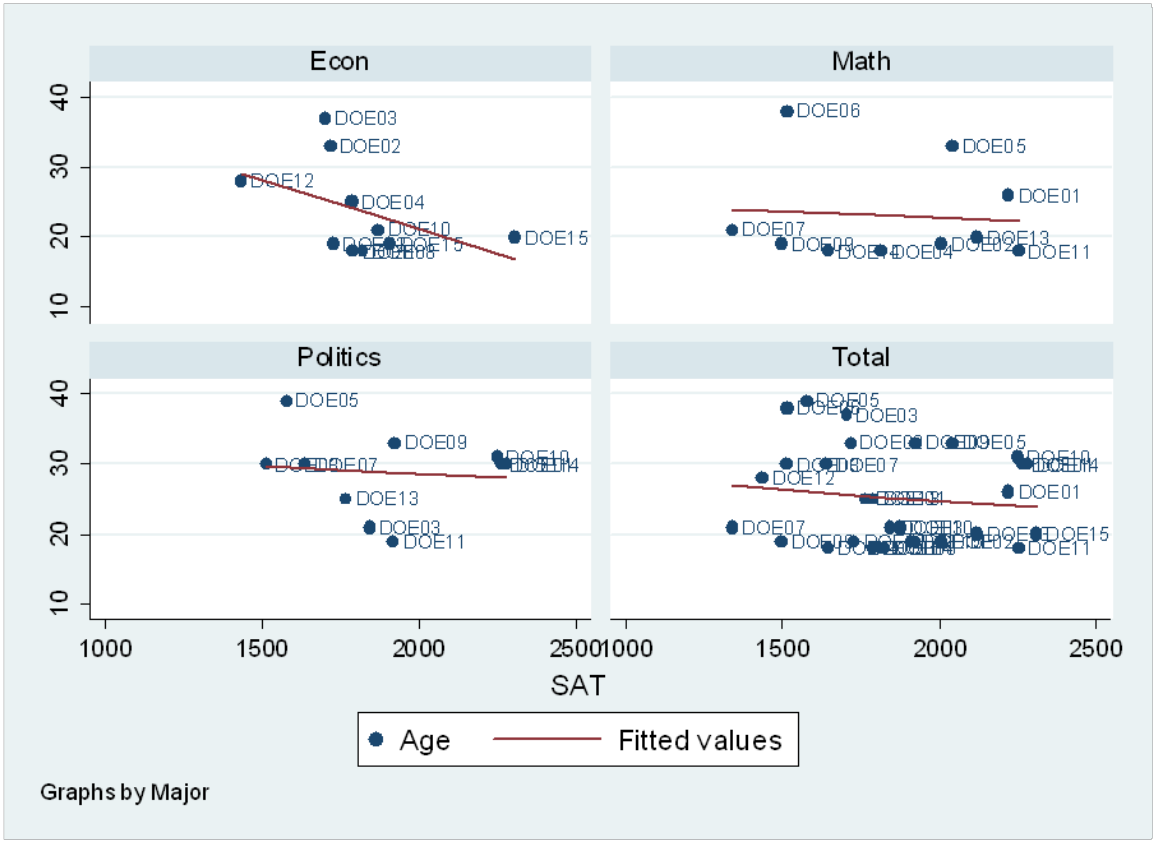


`twoway scatter` *`age sat`*`, mlabel(`*`last`*`) ||`
`lfit` *`age sat`*`, yline(30) xline(1800)`



By categories

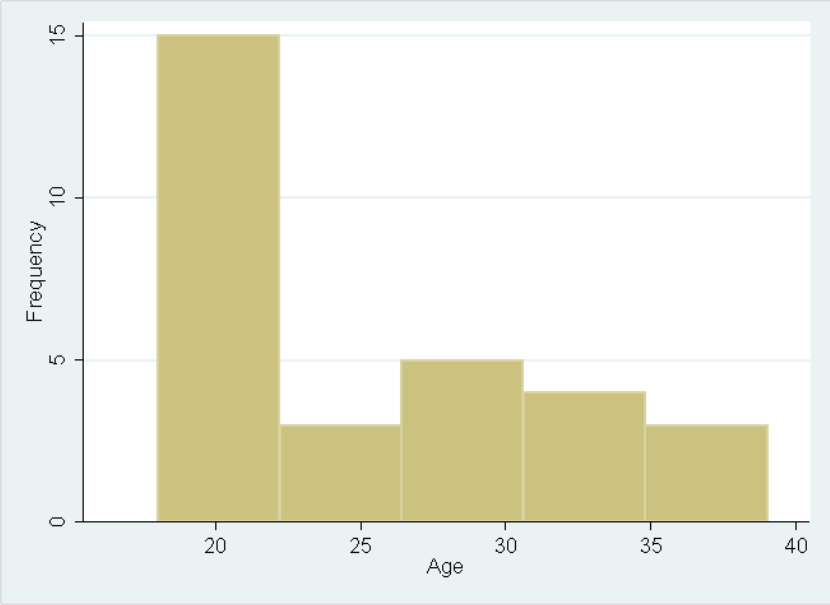
```
twoway scatter age sat, mlabel(last) by(major, total)
```



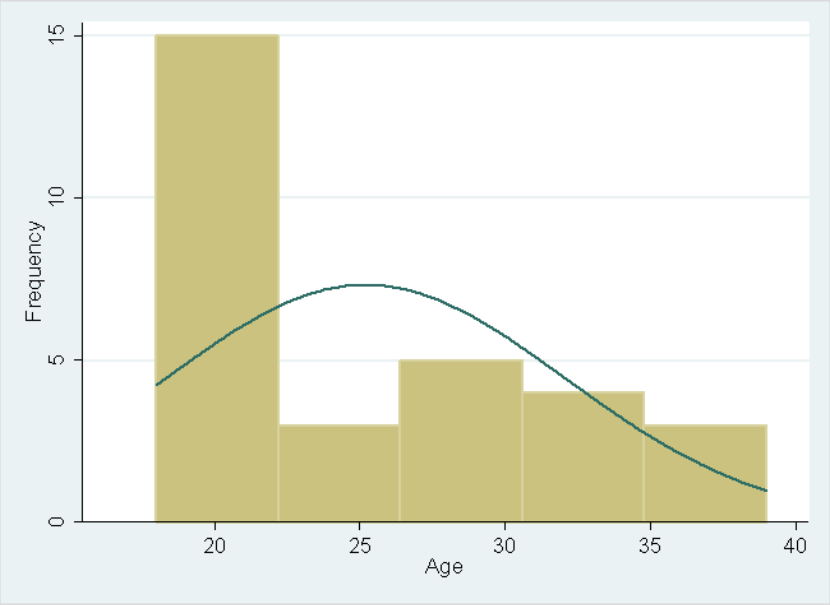
Graphs: histogram

Histograms are another good way to visually explore data, especially to check for a normal distribution; here are some examples (type help histogram in the command window for further details):

`histogram age, frequency`



`histogram age, frequency normal`



`Catplot` is used to graph categorical data. Since it is a user defined program you may have to install it typing: `ssc install catplot`

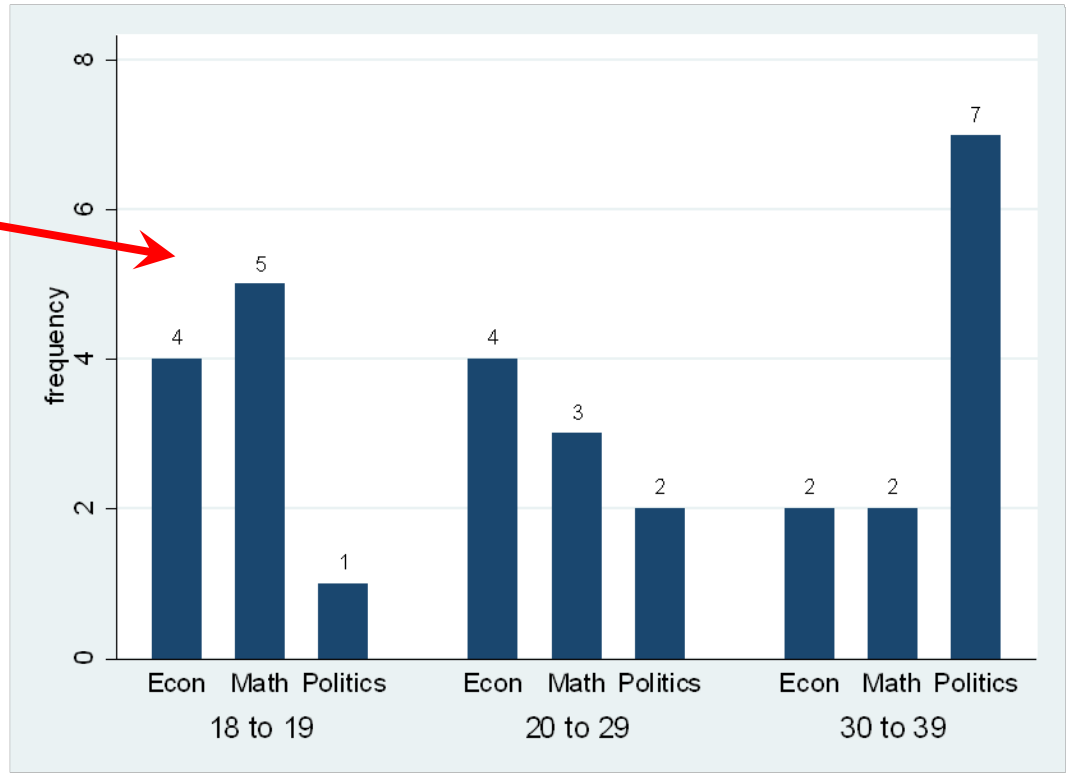
Now, type

`tab agegroups major, col row cell`

`catplot bar major agegroups, blabel(bar)`

```
. tab agegroups major, col row cell

Key

    frequency
  row percentage
  column percentage
  cell percentage
```

Age by groups	Major Econ	Math	Politics	Total
18 to 19	4	5	1	10
	40.00	50.00	10.00	100.00
	40.00	50.00	10.00	33.33
	13.33	16.67	3.33	33.33
20 to 29	4	3	2	9
	44.44	33.33	22.22	100.00
	40.00	30.00	20.00	30.00
	13.33	10.00	6.67	30.00
30 to 39	2	2	7	11
	18.18	18.18	63.64	100.00
	20.00	20.00	70.00	36.67
	6.67	6.67	23.33	36.67
Total	10	10	10	30
	33.33	33.33	33.33	100.00
	100.00	100.00	100.00	100.00
	33.33	33.33	33.33	100.00



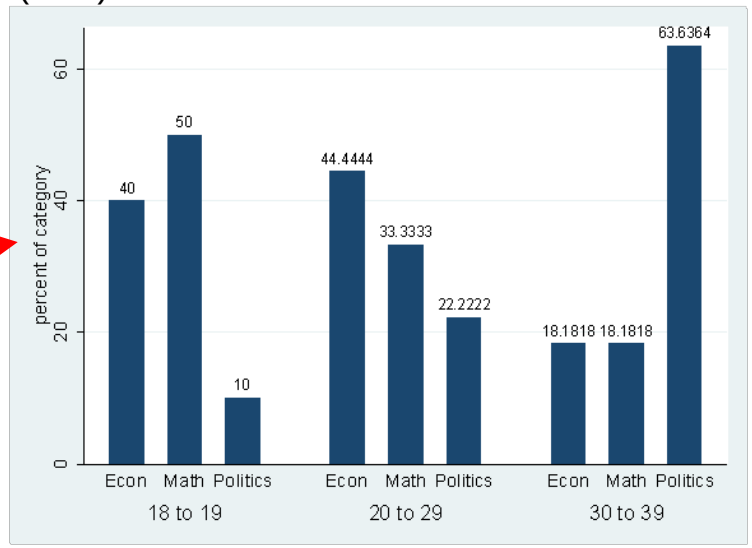
Note: Numbers correspond to the frequencies in the table.

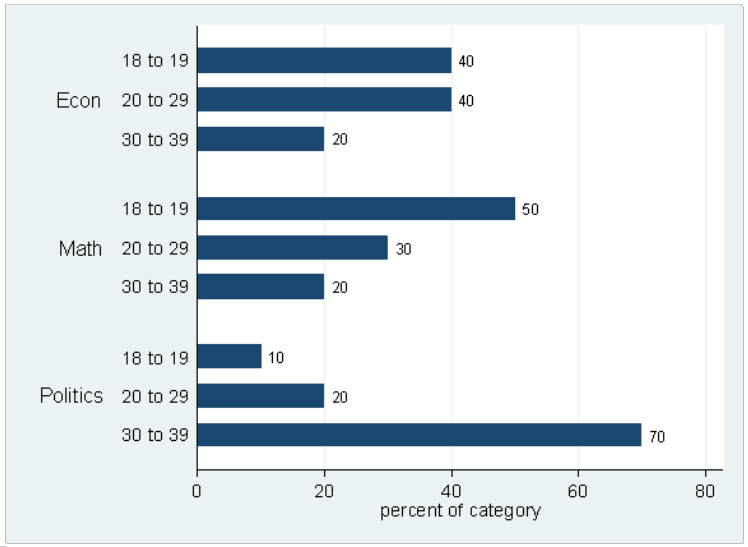
Graphs: catplot

```
catplot  bar major agegroups, percent(agegroups)  blabel(bar)
```

```
. tab agegroups major, col row cell

Key

    frequency
  row percentage
column percentage
  cell percentage
```

Age by groups	Major Econ	Math	Politics	Total
18 to 19	4	5	1	10
	40.00	50.00	10.00	100.00
	40.00	50.00	10.00	33.33
	13.33	16.67	3.33	33.33
20 to 29	4	3	2	9
	44.44	33.33	22.22	100.00
	40.00	30.00	20.00	30.00
	13.33	10.00	6.67	30.00
30 to 39	2	2	7	11
	18.18	18.18	63.64	100.00
	20.00	20.00	70.00	36.67
	6.67	6.67	23.33	36.67
Total	10	10	10	30
	33.33	33.33	33.33	100.00
	100.00	100.00	100.00	100.00
	33.33	33.33	33.33	100.00






```
catplot  hbar agegroups major, percent(major)  blabel(bar)
```

Graphs: catplot

catplot hbar *major agegroups*, percent(*major sex*) blabel(bar) by(*sex*)

Raw counts by major and sex →

```
. bysort sex: tab agegroups major, col nokey
```

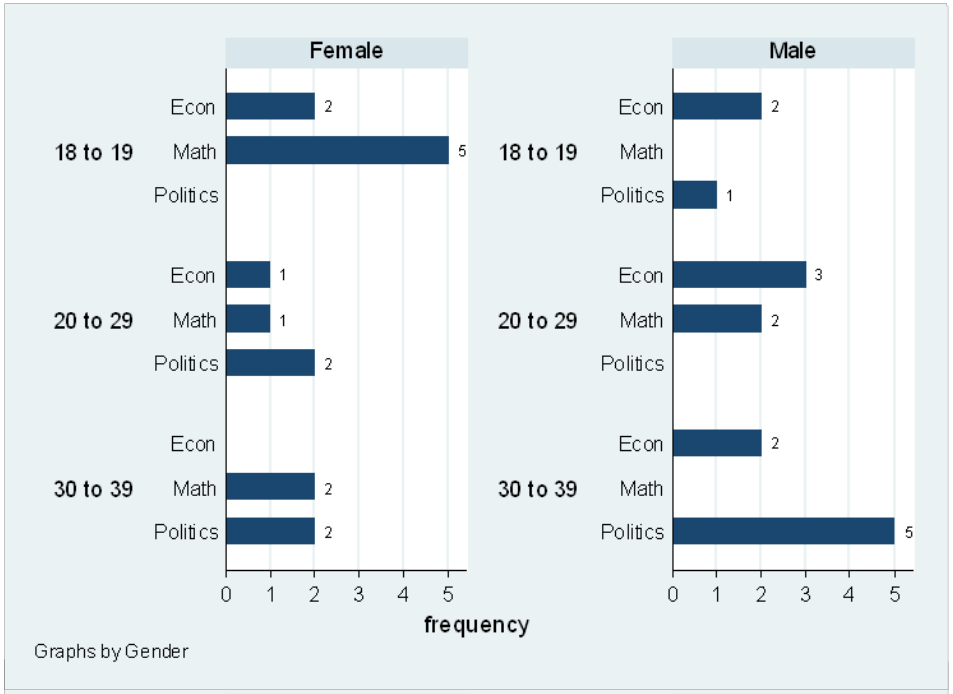
-> sex = Female

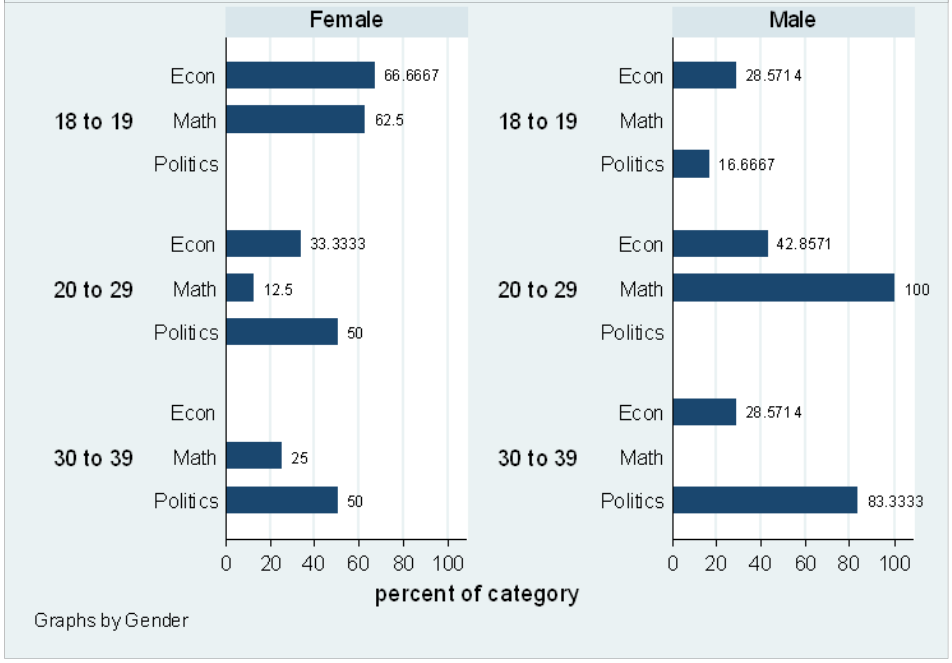
Age by groups	Major Econ	Math	Politics	Total
18 to 19	2 66.67	5 62.50	0 0.00	7 46.67
20 to 29	1 33.33	1 12.50	2 50.00	4 26.67
30 to 39	0 0.00	2 25.00	2 50.00	4 26.67
Total	3 100.00	8 100.00	4 100.00	15 100.00

-> sex = Male

Age by groups	Major Econ	Math	Politics	Total
18 to 19	2 28.57	0 0.00	1 16.67	3 20.00
20 to 29	3 42.86	2 100.00	0 0.00	5 33.33
30 to 39	2 28.57	0 0.00	5 83.33	7 46.67
Total	7 100.00	2 100.00	6 100.00	15 100.00

Percentages by major and sex →

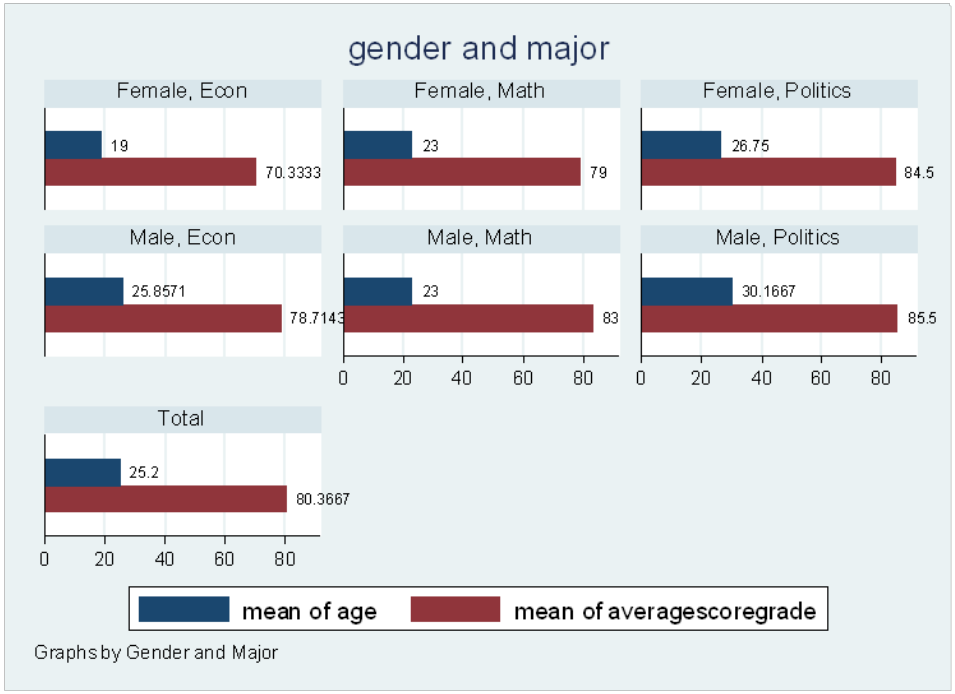




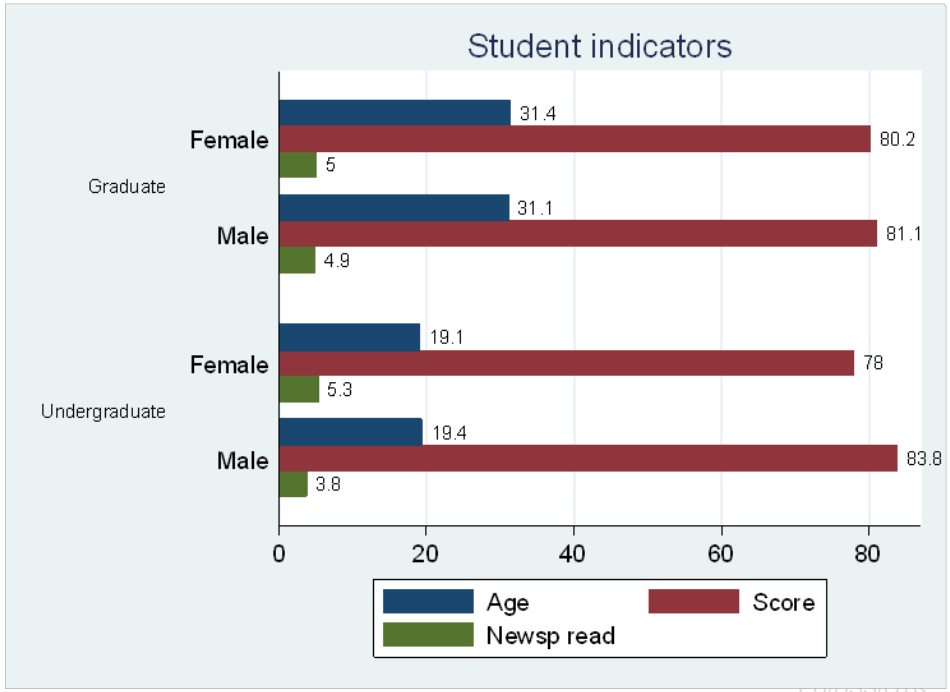
Graphs: means

Stata can also help to visually present summaries of data. If you do not want to type you can go to ‘graphics’ in the menu.

```
graph hbar (mean) age (mean) averagescoregrade,
blabel(bar) by(, title(gender and major)) by(gender
major, total)
```




```
graph hbar (mean) age  averagescoregrade
newspaperreadershiptimeswk, over(gender)
over(studentstatus, label(labsize(small))) blabel(bar)
title(Student indicators) legend(label(1 "Age")
label(2 "Score") label(3 "Newsp read"))
```



In this section we will explore some basics of regression analysis.
We will run a multivariate regression and some diagnostics :

- General setting and output interpretation (what to look for)
- Normality
- Linearity/functional form
- Homoskedasticity/heteroskedasticiy
- Robust standard errors
- Omitted variable bias/specification error
- Outliers
- *F*-test
- Interaction terms

The main references/sources for this section are:

- Stock, James and Mark Watson, *Introduction to Econometrics*, 2003
- Hamilton, Lawrence, *Statistics with Stata (updated for version 9)*, 2006
- The UCLA online tutorial http://www.ats.ucla.edu/stat/stata/

We use regression to estimate the unknown effect of changing one variable over another (Stock and Watson, 2003, ch. 4)

When we run a regression we assume a linear relationship between two variables (i.e. *X* and *Y*). Technically, it estimates how much *Y* changes when *X* changes one unit.

In Stata we use the command `regress`, type:

```
regress [dependent variable] [independent variable(s)]
```

```
regress y x
```

In a multivariate setting we type:

```
regress y x1 x2 x3 …
```

Before running a regression it is recommended to have a clear idea of what you are trying to estimate (i.e. which are your dependent and independent variables).

A regression makes sense only if there is a sound theory behind it.

Data and examples for this section come from the book *Statistics with Stata (updated for version 9)* by Lawrence C. Hamilton (chapter 6). Click here to download the data or search for it at http://www.duxbury.com/highered/. Use the file `states.dta` (educational data for the U.S.).

```
. use states
(U.S. states data 1990-91)

. describe

Contains data from states.dta
  obs:            51                          U.S. states data 1990-91
 vars:            21                          14 Sep 2003 18:34
 size:         4,386 (99.9% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
state           str20  %20s                   State
region          byte   %9.0g       region     Geographical region
pop             float  %9.0g                  1990 population
area            float  %9.0g                  Land area, square miles
density         float  %7.2f                  People per square mile
metro           float  %5.1f                  Metropolitan area population, %
waste           float  %5.2f                  Per capita solid waste, tons
energy          int    %8.0g                  Per capita energy consumed, Btu
miles           float  %8.0g                  Per capita miles/year, 1,000
toxic           float  %5.2f                  Per capita toxics released, lbs
green           float  %5.2f                  Per capita greenhouse gas, tons
house           byte   %8.0g                  House '91 environ. voting, %
senate          byte   %8.0g                  Senate '91 environ. voting, %
csat            int    %9.0g                  Mean composite SAT score
vsat            int    %8.0g                  Mean verbal SAT score
msat            int    %8.0g                  Mean math SAT score
percent         byte   %9.0g                  % HS graduates taking SAT
expense         int    %9.0g                  Per pupil expenditures prim&sec
income          double %10.0g                 Median household income, $1,000
high            float  %9.0g                  % adults HS diploma
college         float  %9.0g                  % adults college degree
-------------------------------------------------------------------------------
Sorted by:  state
```

Starting question: *Are SAT scores higher in states that spend more money on education controlling by other factors?*

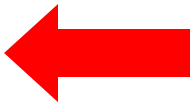
- Dependent (or predicted, Y) variable – SAT scores, variable `csat` in dataset
- Independent (or predictor, X) variable(s) – Expenditures on education, variable `expense` in dataset. Other variables `percent, income, high, college.`

Here is a general description of the variables in the model

```
. describe csat expense percent income high college

              storage  display     value
variable name   type   format      label      variable label
------------------------------------------------------------------------------
csat            int    %9.0g                  Mean composite SAT score
expense         int    %9.0g                  Per pupil expenditures prim&sec
percent         byte   %9.0g                  % HS graduates taking SAT
income          double %10.0g                 Median household income, $1,000
high            float  %9.0g                  % adults HS diploma
college         float  %9.0g                  % adults college degree
```

```
. summarize csat expense percent income high college

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        csat |         51     944.098    66.93497        832       1093
     expense |         51    5235.961    1401.155       2960       9259
     percent |         51    35.76471    26.19281          4         81
      income |         51    33.95657    6.423134     23.465     48.618
        high |         51    76.26078    5.588741       64.3       86.6
-------------+---------------------------------------------------------
     college |         51    20.02157     4.16578       12.3       33.3
```

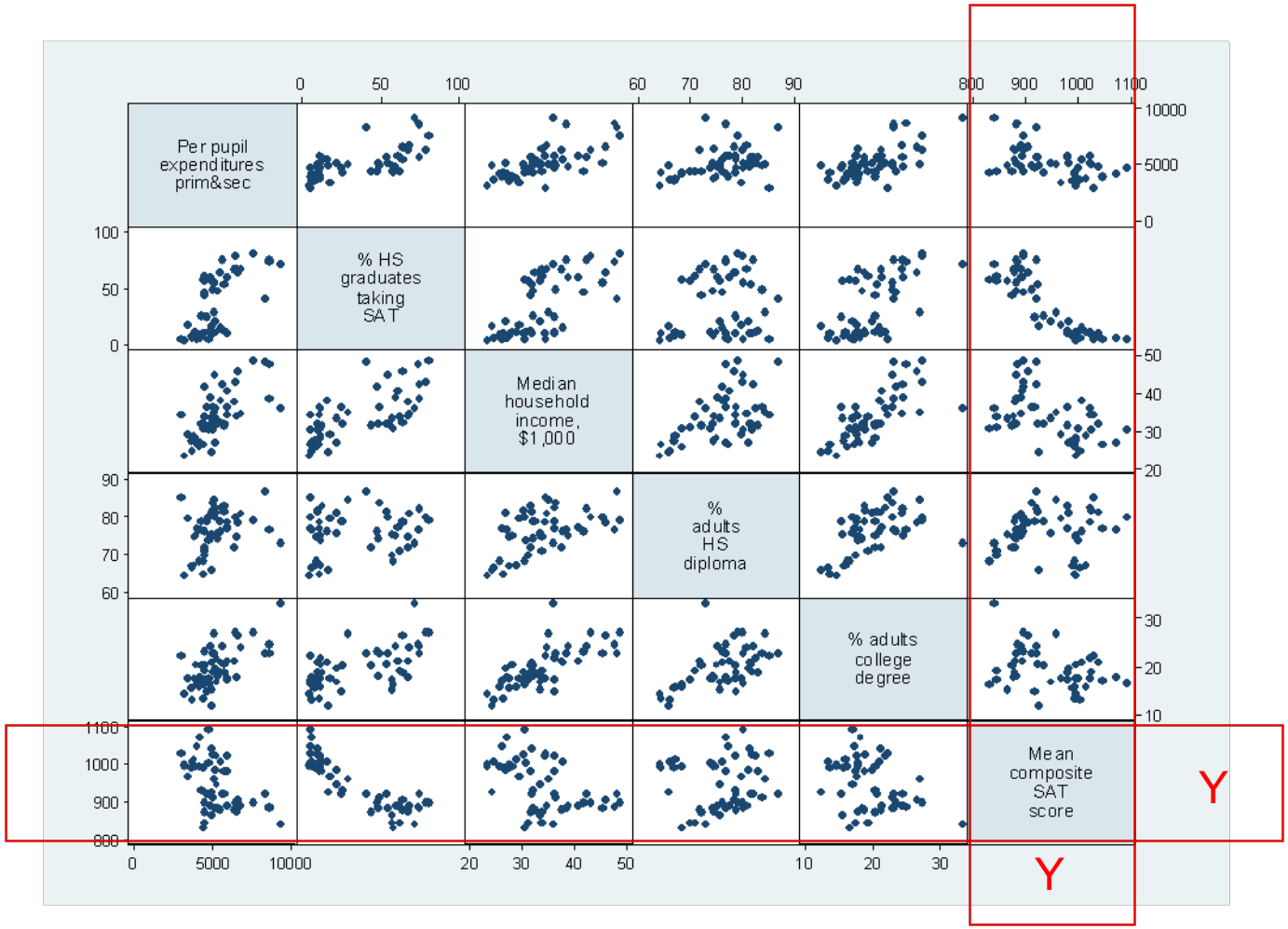
```
. corr csat expense percent income high college
(obs=51)

             |     csat  expense  percent   income     high  college
-------------+------------------------------------------------------
        csat |   1.0000
     expense |  -0.4663   1.0000
     percent |  -0.8758   0.6509   1.0000
      income |  -0.4713   0.6784   0.6733   1.0000
        high |   0.0858   0.3133   0.1413   0.5099   1.0000
     college |  -0.3729   0.6400   0.6091   0.7234   0.5319   1.0000
```



This is a correlation matrix for all variables in the model. Numbers are Pearson correlation coefficients, go from -1 to 1. Closer to 1 means strong correlation. A negative value indicates an inverse relationship (roughly, when one goes up the other goes down).

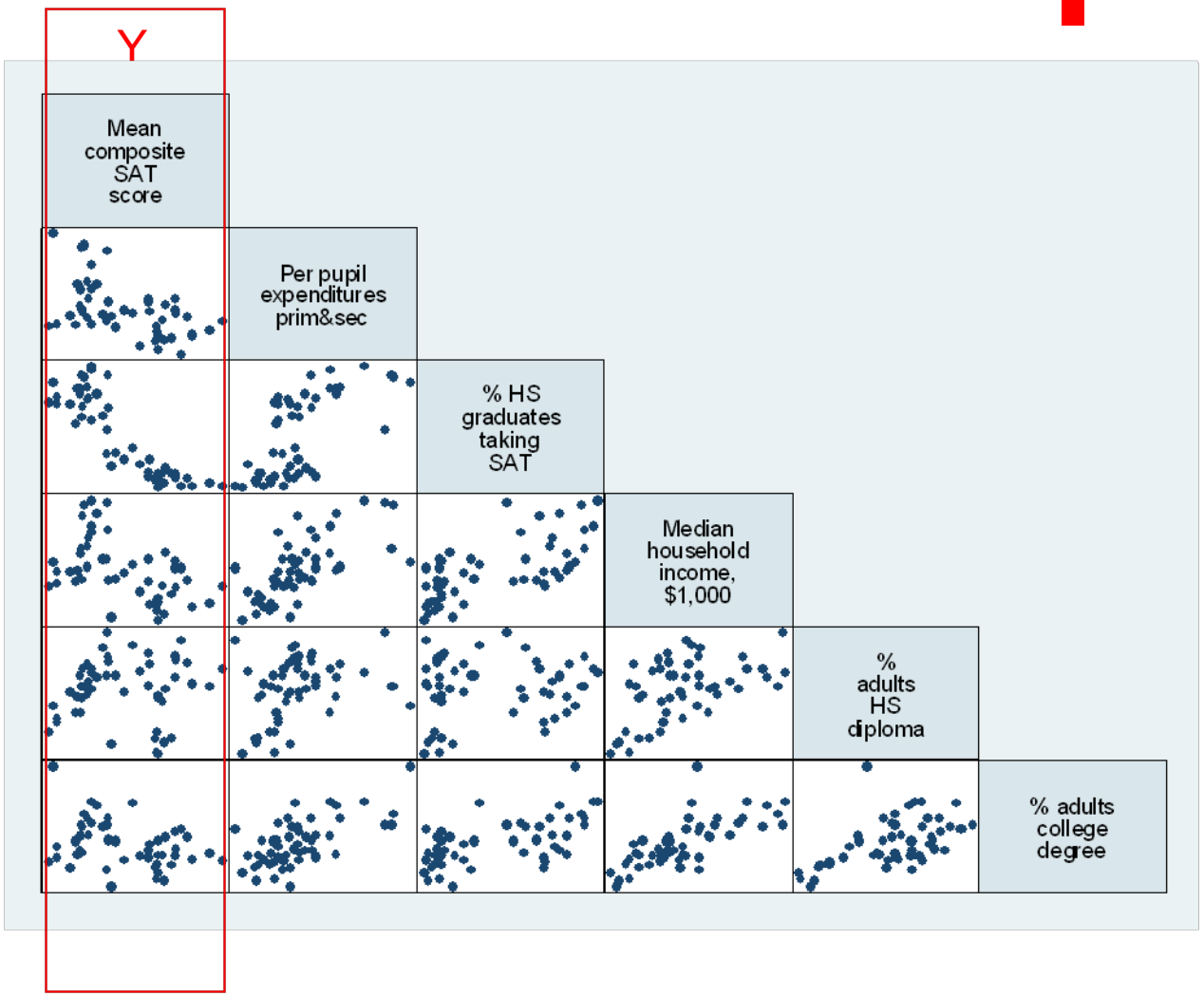
Before running a regression is always recommended to graph dependent and independent variables to explore their relationship. Command `graph matrix` produces a series of scatterplots for all variables. Type:

```
graph matrix expense percent income high college csat
```



Here is another option for the graph.

```
graph matrix csat expense percent income high college, half
maxis(ylabel(none) xlabel(none))
```



Regression: what to look for

Lets run the regression:

`regress csat expense percent income high college, robust`

Robust standard errors (to control for heteroskedasticity)

Dependent variable (Y)

Independent variables (X)

```
. regress csat expense percent income high college, robust

Linear regression                               Number of obs =      51
                                                F(  5,    45) =   50.90
                                                Prob > F      =  0.0000
                                                R-squared     =  0.8243
                                                Root MSE      =  29.571

------------------------------------------------------------------------------
             |               Robust
        csat |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     expense |   .0033528    .004781     0.70   0.487    -.0062766    .0129823
     percent |  -2.618177   .2288594   -11.44   0.000    -3.079123   -2.15723
      income |   .1055853   1.207246     0.09   0.931    -2.325933   2.537104
        high |   1.630841    .943318     1.73   0.091    -.2690989   3.530781
     college |   2.030894   2.113792     0.96   0.342    -2.226502    6.28829
       _cons |   851.5649   57.28743    14.86   0.000     736.1821   966.9477
------------------------------------------------------------------------------
```

1

This is the p-value of the model. It indicates the reliability of X to predict Y. Usually we need a p-value lower than 0.05 to show a statistically significant relationship between X and Y.

2

R-square shows the amount of variance of Y explained by X. In this case the model explains 82.43% of the variance in SAT scores.

3

Adj R^2 (not shown here) shows the same as R^2 but adjusted by the # of cases and # of variables. When the # of variables is small and the # of cases is very large then Adj R^2 is closer to R^2. This provides a more honest association between X and Y.

4

Two-tail p-values test the hypothesis that each coefficient is different from 0. To reject this, the p-value has to be lower than 0.05 (you could choose also an alpha of 0.10). In this case, `expense`, `income`, and `college` are not statistically significant in explaining SAT; `high` is almost significant at 0.10. `Percent` is the only variable that has some significant impact on SAT (its coefficient is different from 0)

5

The t-values test the hypothesis that the coefficient is different from 0. To reject this, you need a t-value greater than 1.96 (at 0.05 confidence). You can get the t-values by dividing the coefficient by its standard error. The t-values also show the importance of a variable in the model. In this case, `percent` is the most important.

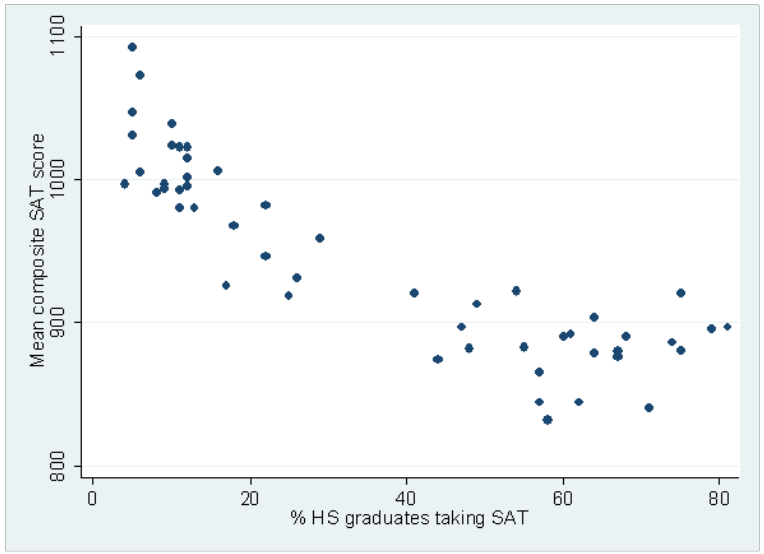
6

csat = 851.56 + 0.003*expense – 2.62*percent + 0.11*income + 1.63*high + 2.03*college

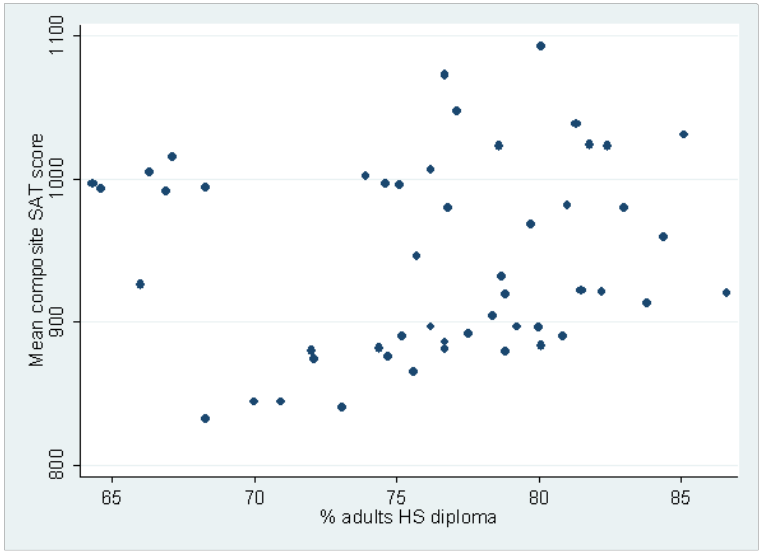
Regression: exploring relationships

Given the previous results, we need to do some adjustments since only one variable was significant. Lets explore further the relationship between `csat` and `percent,` and `high.`

`scatter csat percent`



`scatter csat high`



There seem to be a curvilinear relationship between `csat` and `percent,` and slightly linear between `csat` and `high.` Whenever we find polynomial relationships (curves) we need to add a square (or some other higher power) version of the variable, in this case `percent square` will suffice .

```
generate percent2 = percent^2
```

Now the model will look like this

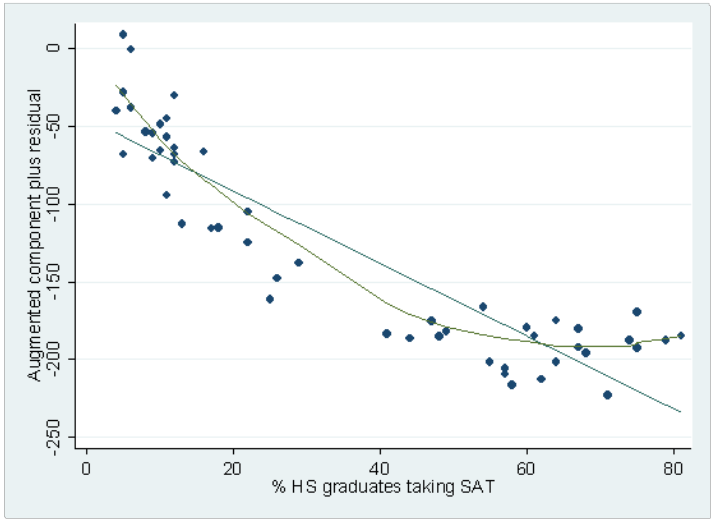
```
regress csat percent percent2 high
```

Regression: functional form/linearity

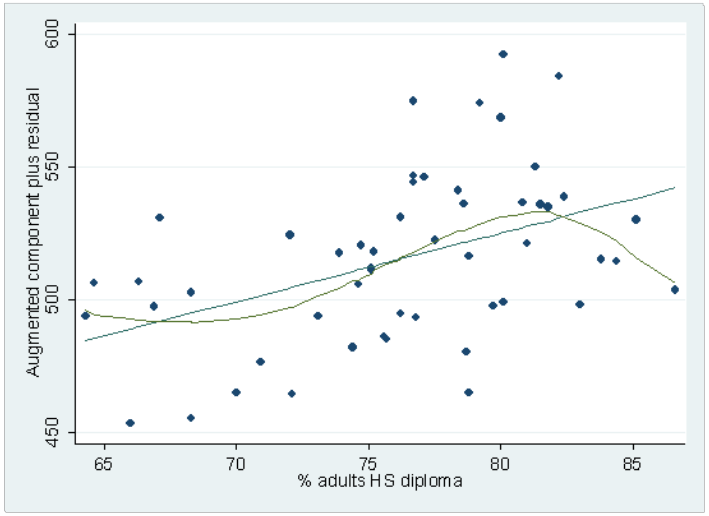
As a footnote, another graphical way to explore a possible linear relationship between variables or to detect nonlinearity to define a functional form is by using the command `acprplot` (augmented component-plus-residual plot). Right after running a the regression:

```
regress csat percent high /* Notice we do not include percent2 */
```

```
acprplot percent, lowess
```




```
acprplot high, lowess
```



The option `lowess` (locally weighted scatterplot smoothing) draw the observed pattern in the data to help identify nonlinearities. `Percent` shows a quadratic relation, it makes sense to add a square version of it. `High` shows a polynomial pattern as well but goes around the regression line (except on the right). We could keep it as is for now.

Form more details see http://www.ats.ucla.edu/stat/stata/webbooks/reg/chapter2/statareg2.htm, and/or type `help acprplot` and `help lowess`.

Regression: F-test

Before we continue lets take another look at the original regression and run some individual tests on its coefficients.

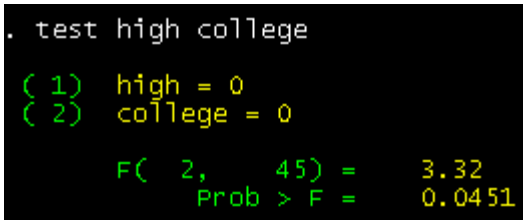
We have two types of tests with the regression model: *F*-test, which tests the overall fit of the model (all coefficients different from 0) and *t*-test (individual coefficients different from 0). You can customize your tests to check for other possible situations, like two coefficients jointly different from 0. In the regression, two variables related to educational attainment were not significant. We could, however, test whether these two have no effect on SAT scores (see Hamilton, 2006, p.175). Lets run the original regression again:

```
quietly regress csat expense percent income high college
```

Note 'quietly' suppress the regression output

To test the null hypothesis that *both* coefficients do not have any effect on `csat`, type:

```
test high college
```



The p-value is 0.0451, under the 0.05 usual threshold (95% confidence) so we conclude that *both* variables have indeed some effect on SAT. In a way, this is saying that both have similar effect or measuring the same thing (which could suggest multicolinearity). We could keep `high` since it was borderline significant.

Some other possible tests are (see Hamilton, 2006, p.176):

```
test income = 1
test high = college
test income = (high + college)/100
```

Note: Not to be confused with `ttest`. Type `help test` and `help ttest` for more details

Regression: output

Lets try the new model. It has now a higher R-squared (0.92) and all the variables are significant.

```
. regress csat percent percent2 high

      Source |       SS       df       MS              Number of obs =      51
-------------+------------------------------           F(  3,    47) =  193.37
       Model |  207225.103     3  69075.0343            Prob > F      =  0.0000
    Residual |  16789.4069    47  357.221424            R-squared     =  0.9251
-------------+------------------------------           Adj R-squared =  0.9203
       Total |  224014.51     50   4480.2902            Root MSE      =    18.9

------------------------------------------------------------------------------
        csat |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     percent |  -6.520312   .5095805   -12.80   0.000    -7.545455   -5.495168
    percent2 |   .0536555   .0063678     8.43   0.000     .0408452    .0664659
        high |   2.986509   .4857502     6.15   0.000     2.009305    3.963712
       _cons |   844.8207   36.63387    23.06   0.000     771.1228    918.5185
------------------------------------------------------------------------------
```

The new equation is:

`csat = 844.82 – 6.52*percent + 0.05*percent2 + 2.98*high`

`Percent's` coefficient is -6.52. So, if `percent` increases by one unit, `csat` will decrease by 6.52 units. With a statistically significant p-value of 0.000 (which means that -6.52 is statistically different from 0), `percent` has an important impact on `csat` controlling by other variables (holding them constant). You could read `percent2` (which explains the upward effect) the same way. The net effect of `percent` is the difference between both coefficients (which is still negative).

`High's` coefficient is 2.98. So, if `high` increases by one unit, `csat` will increase by 2.98 units.

The constant 844.82 means that if all variables are 0, the average `csat` score would be 844.82. It is where the regression line crosses the Y axis.

Regression: saving regression coefficients/getting predicted values

Stata temporarily stores the coefficients as `_b[varname]`, so if you type:

```
. display _b[percent]
-6.5203116

. display _b[percent2]
.05365555

. display _b[high]
2.9865088

. display _b[_cons]
844.82067
```

You can save the coefficients as variables by typing:

```
gen percent_coeff = _b[percent]
gen percent_coeff = _b[percent2]
gen high_coeff = _b[high]
gen constant_coeff = _b[_cons]
```

How good the model is will depend on how well it predicts Y and on the validity of the tests.

There are two ways to generate the *predicted values of Y* (usually called *Yhat*) given the model:

Option A, using `generate` after running the regression:

```
generate csat_predict = _b[_cons] + _b[percent]*percent + _b[percent2]*percent2 + _b[high]*high
```

Option B, using `predict` immediately after running the regression:

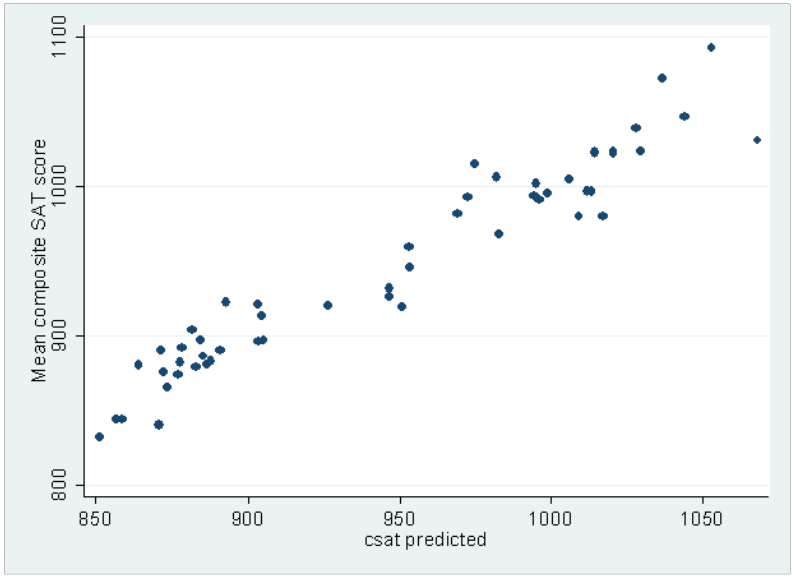
```
predict csat_predict
label variable csat_predict "csat predicted"
```

```
. predict csat_predict
(option xb assumed; fitted values)

. label variable  csat_predict "csat predicted"
```

Regression: observed vs. predicted values

Now lets see how well we did, type

```
scatter csat csat_predict
```



We should expect a 45 degree pattern in the data. Y-axis is the observed data and x-axis the predicted data (Yhat).
In this case the model seems to be doing a good job in predicting `csat`

Regression: testing for normality

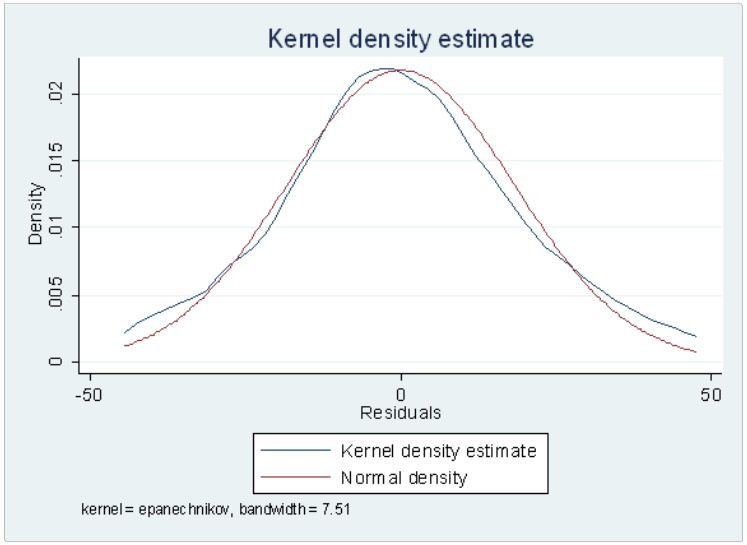
A main assumption of the regression model (OLS) that guarantee the validity of all tests (p, t and F) is that residuals behave 'normal'. Residuals (here indicated by the letter "e") are the difference between the observed values (Y) and the predicted values (Yhat): `e = Y – Yhat`.

In Stata you type: `predict e, resid`

It will generate a variable called "e" (residuals).

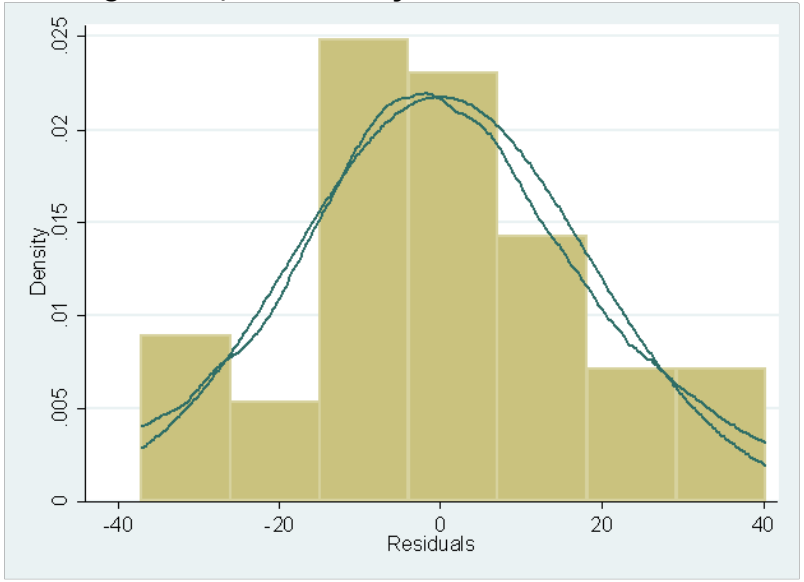
Three graphs will help us check for normality in the residuals: `kdensity, pnorm` and `qnorm`.

`kdensity e, normal`



A kernel density plot produces a kind of histogram for the residuals, the option `normal` overlays a normal distribution to compare. Here residuals seem to follow a normal distribution. Below is an example using `histogram`.

`histogram e, kdensity normal`

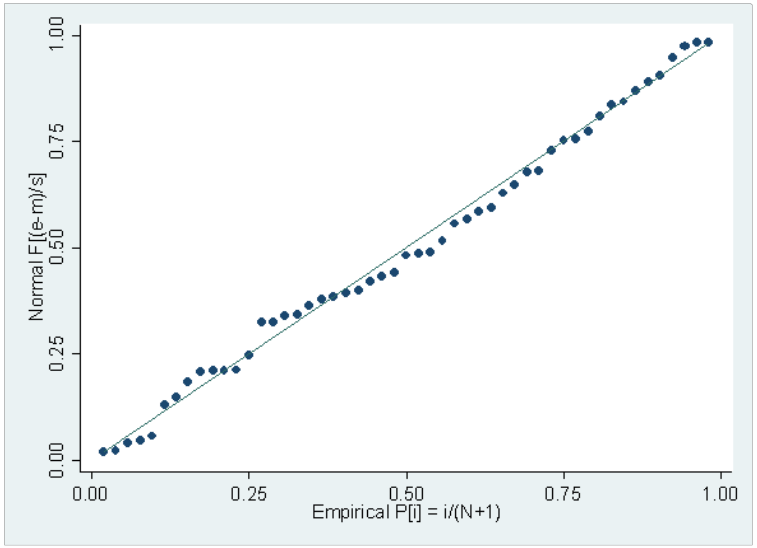


If residuals do not follow a 'normal' pattern then you should check for omitted variables, model specification, linearity, functional forms. In sum, you may need to reassess your model/theory. In practice normality does not represent much of a problem when dealing with really big samples.

Regression: testing for normality

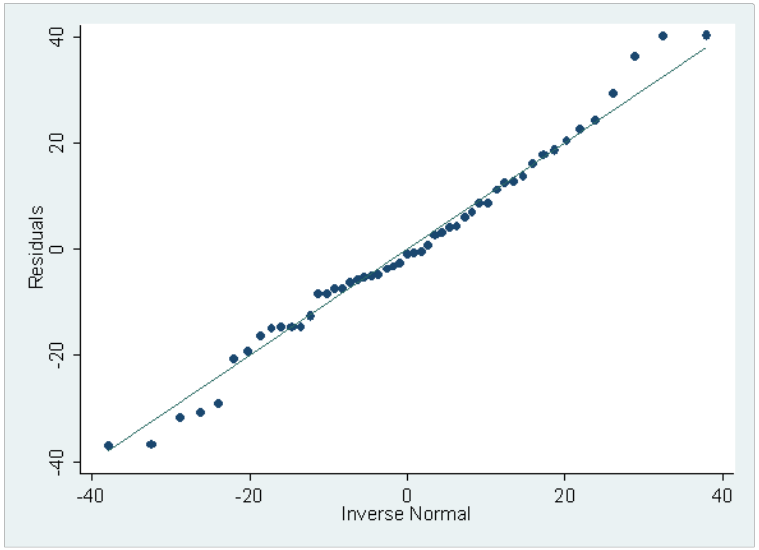
Standardize normal probability plot (`pnorm`) checks for non-normality in the middle range of residuals. Again, slightly off the line but looks ok.

`pnorm e`



Quintile-normal plots (`qnorm`) check for non-non-normality in the extremes of the data (tails). It plots quintiles of residuals vs quintiles of a normal distribution. Tails are a bit off the normal.

`qnorm e`



A non-graphical test is the Shapiro-Wilk test for normality. It tests the hypothesis that the distribution is normal, in this case the null hypothesis is that the distribution of the residuals is normal. Type

`swilk e`

```
. swilk e

                    Shapiro-Wilk W test for normal data
    Variable |   Obs        W          V          z       Prob>z
-------------+--------------------------------------------------
           e |    51     0.98238      0.842     -0.368    0.64349
```

The null hypothesis is that the distribution of the residuals is normal, here the p-value is 0.64 (way over the usual 0.05 threshold) therefore we failed to reject the null. We conclude then that residuals are normally distributed.

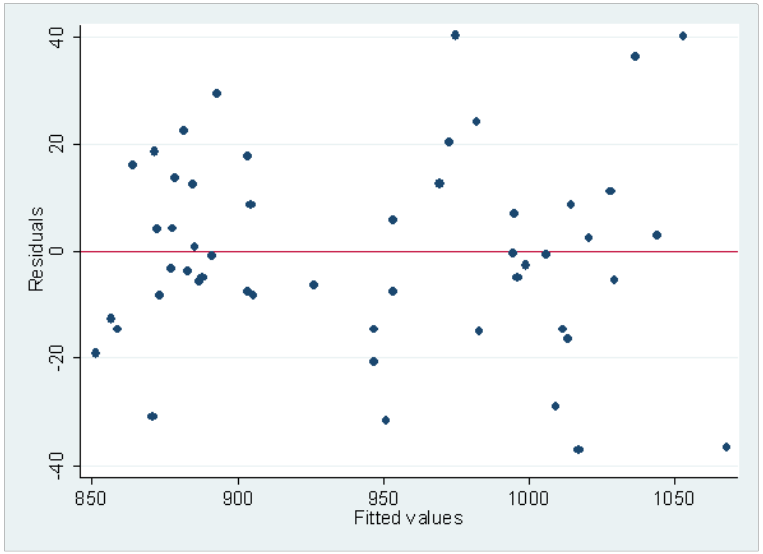
Regression: testing for homoskedasticity

Another important assumption is that the variance in the residuals has to be homoskedastic, which means constant. Residuals cannot varied for lower of higher values of X (i.e. fitted values of Y since Y=Xb). A definition:

> "The error term [e] is homoskedastic if the variance of the conditional distribution of [e_i] given X_i [var(e_i|X_i)], is constant for i=1…n, and in particular does not depend on x; otherwise, the error term is heteroskedastic" (Stock and Watson, 2003, p.126)

When plotting residuals vs. predicted values (Yhat) we *should not observe* any pattern at all. In Stata we do this using `rvfplot` right after running the regression, it will automatically draw a scatterplot between residuals and predicted values; and `hettest` to produce a non-graphical test.

`rvfplot, yline(0)`



Residuals seem to slightly expand at higher levels of Yhat.

`estat hettest`

```
. estat hettest

Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
         Ho: Constant variance
         Variables: fitted values of csat

         chi2(1)      =     4.86
         Prob > chi2  =   0.0274
```

This is the Breusch-Pagan test for heteroskedasticity. The null hypothesis is that residuals are homoskedastic. Here we reject the null and concluded that residuals are heteroskedastic.

These two tests suggest the presence of heteroskedasticity in our model. The problem with this is that we may have the wrong estimates of the standard errors for the coefficients and therefore their t-values.
By default Stata assumes homoskedastic standard errors, so we need to adjust our model to account for heteroskedasticity. To do this we use the option `robust` in the `regress` command.

```
regress csat percent percent2 high, robust
```

See the next slide for results

Regression: robust standard errors

To run a regression with robust standard erros type:

```
regress csat percent percent2 high, robust
```

Notice the difference in the standard errors and the t-values. Following Stock and Watson, as a rule-of-thumb, you should always assume heteroskedasticiy in your model and use robust standard errors by adding the option `robust` (or `r` for short) to the regression command (see Stock and Watson, 2003, chapter 4)

```
. regress csat percent percent2 high, robust

Linear regression                               Number of obs =      51
                                                F(  3,    47) =  160.90
                                                Prob > F      =  0.0000
                                                R-squared     =  0.9251
                                                Root MSE      =   18.9

------------------------------------------------------------------------------
             |               Robust
        csat |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     percent |  -6.520312   .4934097   -13.21   0.000    -7.512924   -5.527699
    percent2 |   .0536555   .0056491     9.50   0.000      .042291    .0650201
        high |   2.986509    .54564     5.47   0.000     1.888823    4.084195
       _cons |   844.8207   38.8214    21.76   0.000     766.7221    922.9192
------------------------------------------------------------------------------
```

Regression: omitted-variable test

How do we know we have included all variables we need to explain Y?

Testing for omitted variable bias is important for our model since it is related to the assumption that the error term and the independent variables in the model are not correlated (E(e|X) = 0)

If we are missing one variable in our model and "[(1)] is correlated with the included regressor; and [(2)] the omitted variable is a determinant of the dependent variable" (Stock and Watson, 2003, p.144), then our regression coefficients are inconsistent.

In Stata we test for omitted-variable bias using the `ovtest` command. After running the regression type:

```
ovtest
```

```
. ovtest

Ramsey RESET test using powers of the fitted values of csat
       Ho:  model has no omitted variables
                   F(3, 44) =      1.48
                   Prob > F =      0.2319
```

The null hypothesis is that the model does not have omitted-variables bias, the p-value is 0.2319 higher that the usual threshold of 0.05, so we fail to reject the null and conclude that we do not need more variables.

Regression: specification error

Another command to test model specification is `linktest`. It basically checks whether we need more variables in our model by running a new regression with the observed Y (`csat`) against Yhat (`csat_predicted`) and `Yhat-squared` as independent variables[1].

The thing to look for here is the significance of `_hatsq`. The null hypothesis is that there is no specification error. If the p-value of `_hatsq` is <u>not significant</u> then we fail to reject the null and conclude that our model is correctly specified.

```
. linktest
```

Source	SS	df	MS
Model	207270.449	2	103635.225
Residual	16744.0604	48	348.834592
Total	224014.51	50	4480.2902

Number of obs	= 51
F(2, 48)	= 297.09
Prob > F	= 0.0000
R-squared	= 0.9253
Adj R-squared	= 0.9221
Root MSE	= 18.677

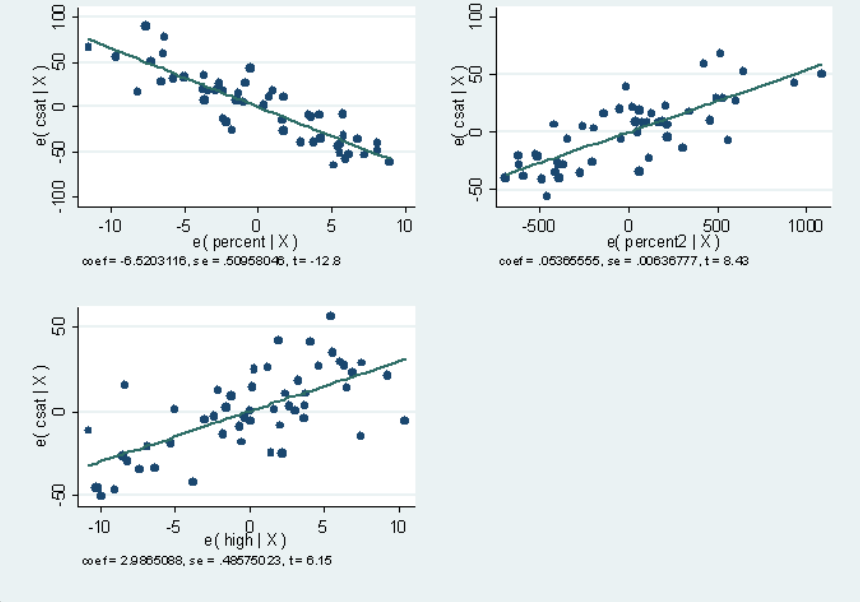
csat	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
_hat	1.588861	1.633699	0.97	0.336	-1.69591	4.873632
_hatsq	-.00031	.0008597	-0.36	0.720	-.0020384	.0014185
_cons	-278.4089	773.132	-0.36	0.720	-1832.895	1276.077

[1] For more details see http://www.ats.ucla.edu/stat/stata/webbooks/reg/chapter2/statareg2.htm, and/or type `help linktest`.

Regression: outliers

To check for outliers we use the `avplots` command (added-variable plots). Outliers are data points with extreme values that could have a negative effect on our estimators. After running the regression type:

```
avplots
```



These plots regress each variable against all others, notice the coefficients on each. All data points seem to be in range, no outliers observed.

For more details and tests on this and influential and leverage variables please check
http://www.ats.ucla.edu/stat/stata/webbooks/reg/chapter2/statareg2.htm

Also type `help diagplots` in the Stata command window.

Regression: summary of influence indicators

DfBeta	Measures the influence (in standard errors terms) of each observation on the *coefficient* of a particular independent variable (for example, x1)	A case is an influential outlier if
		
		|DfBeta|> 2/SQRT(N)
		
		Where N is the sample size.
		
		Note: Stata estimates standardized DfBetas.
			In Stata after running the regression type:
			
			`reg y x1 x2 x3`
			
			`dfbeta x1`
			
			Note: you could also type:
			
			`predict DFx1, dfbeta(x1)`
			
			To estimate the dfbetas for all predictors just type:
			
			`dfbeta`
			
			To flag the cutoff
			
			`gen cutoffdfbeta = abs(DFx1) > 2/sqrt(e(N)) & e(sample)`
				In SPSS: Analyze-Regression-Linear; click Save. Select under "Influence Statistics" to add as a new variable (DFB1_1) or in syntax type
				`REGRESSION`
				`  /MISSING LISTWISE`
				`  /STATISTICS COEFF OUTS R ANOVA`
				`  /CRITERIA=PIN(.05) POUT(.10)`
				`  /NOORIGIN`
				`  /DEPENDENT  Y`
				`  /METHOD=ENTER X1 X2 X3`
				`  /CASEWISE PLOT(ZRESID) OUTLIERS(3) DEFAULTS DFBETA`
				`  /SAVE MAHAL COOK LEVER DFBETA SDBETA DFFIT SDFIT COVRATIO .`

DfFit	It is a summary measure of leverage and high residuals.
	
	Measures how much an observation influences the regression model as a whole.
	
	How much the predicted values change as a result of including and excluding a particular observation.
		High influence if
		
		|DfFIT| >2*SQRT(k/N)
		
		Where k is the number of parameters (including the intercept) and N is the sample size.
			After running the regression type
			
			`predict dfits if e(sample), dfits`
			
			To generate the flag for the cutoff type:
			
			`gen cutoffdfit= abs(dfits)>2*sqrt((e(df_m) +1)/e(N)) & e(sample)`
				Same as DfBeta above (DFF_1)

Covariance ratio	Measures the impact of an observation on the standard errors	High impact if |COVRATIO-1| ≥ 3*k/N Where k is the number of parameters (including the intercept) and N is the sample size.	In Stata after running the regression type `predict covratio if e(sample), covratio`	Same as DfBeta above (COV_1)

Regression: summary of distance measures

Cook's distance	Measures how much an observation influences the overall model or predicted values. It is a summary measure of leverage and high residuals. .	High influence if D > 4/N Where N is the sample size. A D>1 indicates big outlier problem	In Stata after running the regression type: `predict D, cooksd`	In SPSS: Analyze-Regression-Linear; click Save. Select under "Distances" to add as a new variable (COO_1) or in syntax type REGRESSION /MISSING LISTWISE /STATISTICS COEFF OUTS R ANOVA /CRITERIA=PIN(.05) POUT(.10) /NOORIGIN /DEPENDENT Y /METHOD=ENTER X1 X2 X3 /CASEWISE PLOT(ZRESID) OUTLIERS(3) DEFAULTS DFBETA /SAVE MAHAL COOK LEVER DFBETA SDBETA DFFIT SDFIT COVRATIO.
Leverage	Measures how much an observation influences regression coefficients.	High influence if leverage h > 2*k/N Where k is the number of parameters (including the intercept) and N is the sample size. A rule-of-thumb: Leverage goes from 0 to 1. A value closer to 1 or over 0.5 may indicate problems.	In Stata after running the regression type: `predict lev, leverage`	Same as above (LEV_1)
Mahalanobis distance	It is rescaled measure of leverage. M = leverage*(N-1) Where N is sample size.	Higher levels indicate higher distance from average values. The M-distance follows a Chi-square distribution with k-1 df and alpha=0.001 (where k is the number of independent variables). Any value over this Chi-square value may indicate problems.	Not available	Same as above (MAH_1)

Sources for the summary tables: influence indicators and distance measures

- Statnotes: http://faculty.chass.ncsu.edu/garson/PA765/regress.htm#outlier2
- *An Introduction to Econometrics Using Stata*/Christopher F. Baum, Stata Press, 2006
- *Statistics with Stata (updated for version 9) /* Lawrence Hamilton, Thomson Books/Cole, 2006

Regression: multicollinearity

An important assumption for the multiple regression model is that independent variables are *not perfectly multicolinear*. This is, one regressor should not be a linear function of another. When multicollinearity is present, Stata will drop one of the variables to avoid a division by zero in the OLS procedure (see Stock and Watson, 2003, chapter 5). A mayor problem with multicollinearity is that *standand errors may be inflated*. The Stata command to check for multicollinearity is `vif` (variance inflation factor). Right after running the regression type:

```
. vif
```

Variable	VIF	1/VIF
percent	24.94	0.040103
percent2	24.78	0.040354
high	1.03	0.969423
Mean VIF	16.92	

A `vif > 10` or a `1/vif < 0.10` indicates trouble. We know that `percent` and `percent2` are related since one is the square of the other. They are ok since `percent` has a quadratic relationship with Y. `High` has a `vif` of 1.03 and `1/vif` of 0.96 so we are ok here.

Lets run another regression and get the `vif`.

```
quietly regress csat expense percent income high college
```

```
. vif
```

Variable	VIF	1/VIF
income	3.21	0.311756
college	2.73	0.365683
percent	2.53	0.395603
expense	2.24	0.445673
high	1.76	0.568732
Mean VIF	2.49	

We do not observe multicollinearity problems here. All `vifs` are under 10 .

Form more details see http://www.ats.ucla.edu/stat/stata/webbooks/reg/chapter2/statareg2.htm, and/or type `help vif`.

Regression: publishing regression output (outreg2)

The command outreg2 gives you the type of presentation you see in published papers. If outreg2 is not available you need to install it by typing

```
ssc install outreg2
```

Let's say the regression is `regress csat percent percent2 high, robust`

The basic syntax for outreg2 is: `outreg2 using [pick a name], [type either word or excel]`

After the regression type the following if you want to export the **results to excel***

```
outreg2 using results, excel
```

```
. outreg2 using results, excel
"results.xml"
seeout
```

Click here to see the file

Or this if you want to **export to word**

```
outreg2 using results, word
```

```
. outreg2 using results, word
"results.rtf"
seeout
```

Click here to see the file

In excel

	A	B
1	**v1**	**v2**
2		(3)
3	COEFFICIENT	csat
4		
5	percent	-6.520***
6		(0.49)
7	percent2	0.0537***
8		(0.0056)
9	high	2.987***
10		(0.55)
11	Constant	844.8***
12		(38.8)
13	Observations	51
14	R-squared	0.93
15	Robust standard errors in parentheses	
16	*** p<0.01, ** p<0.05, * p<0.10	

In word

COEFFICIENT	csat
percent	-6.520***
	(0.49)
percent2	0.0537***
	(0.0056)
high	2.987***
	(0.55)
Constant	844.8***
	(38.8)
Observations	51
R-squared	0.93

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.10

*See the following document for some additional info/tips http://www.fiu.edu/~tardanic/brianne.pdf

Regression: publishing regression output (outreg2)

You can add more models to compare. Lets say you want to add another model without percent2:

```
regress csat percent high, robust
```

Now type to export the results to excel (**notice** we add the append option)

```
outreg2 using results, word append
```

In excel

3	COEFFICIENT	csat	csat
4			
5	percent	-6.520***	-2.315***
6		(0.49)	(0.17)
7	percent2	0.0537***	
8		(0.0056)	
9	high	2.987***	2.561***
10		(0.55)	(0.72)
11	Constant	844.8***	831.6***
12		(38.8)	(53.5)
13	Observations	51	51
14	R-squared	0.93	0.81
15	Robust standard errors in parentheses		
16	*** p<0.01, ** p<0.05, * p<0.1		

In word

COEFFICIENT	csat	csat
percent	-6.520***	-2.315***
	(0.49)	(0.17)
percent2	0.0537***	
	(0.0056)	
high	2.987***	2.561***
	(0.55)	(0.72)
Constant	844.8***	831.6***
	(38.8)	(53.5)
Observations	51	51
R-squared	0.93	0.81

Robust standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1

NOTE: If you run logit/probit regression with odds ratios you need to add the option eform to export the odd ratios

Type help outreg2 for more details. If you do not see outreg2, you may have to install it by typing ssc install outreg2. If this does not work type findit outreg2, select from the list and click "install".

Note: If you get the following error message (when you use the option append or replace it means that you need to close the excel/word window.

```
file results.rtf is read-only; cannot be modified or erased
```

Regression: publishing regression output (outreg2) continue

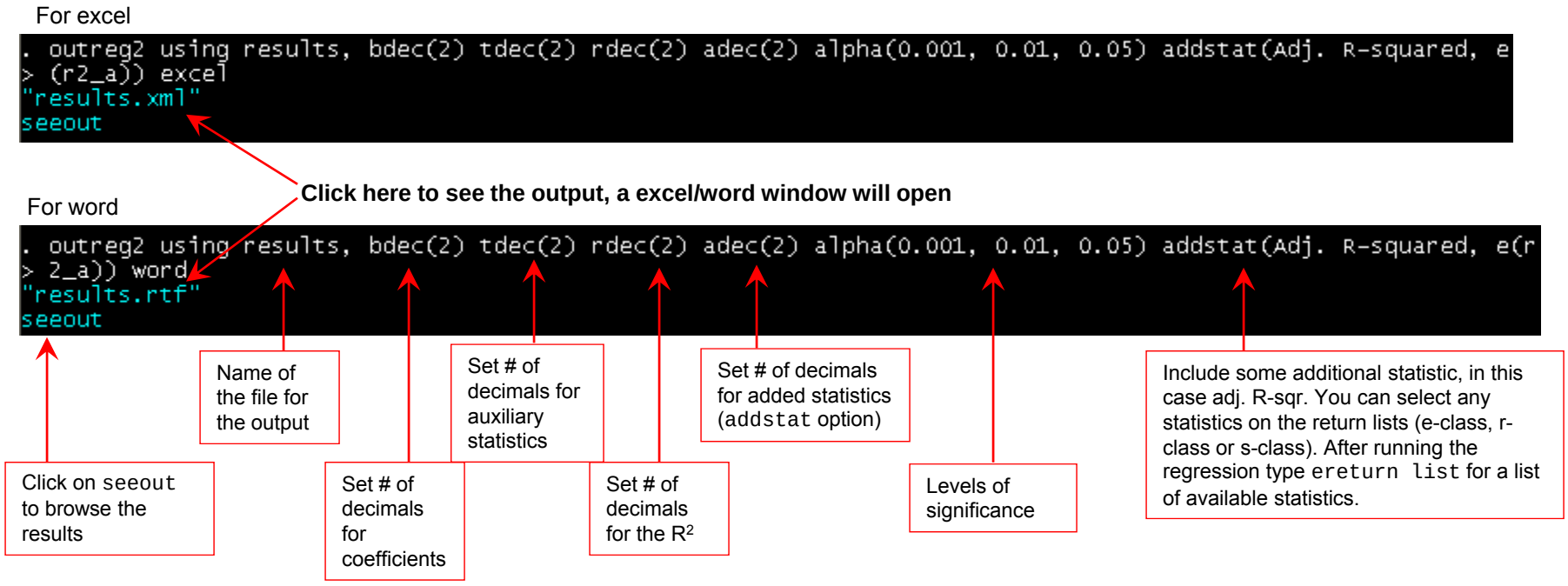
For a customized look, here are some options:

*** Excel

```
outreg2 using results, bdec(2) tdec(2) rdec(2) adec(2) alpha (0.01, 0.05, 0.10)
addstat(Adj. R-squared, e(r2_a)) excel
```

*** Word

```
outreg2 using results, bdec(2) tdec(2) rdec(2) adec(2) alpha (0.01, 0.05, 0.10)
addstat(Adj. R-squared, e(r2_a)) excel
```



Regression: interaction between dummies

Interaction terms are needed whenever there is reason to believe that the effect of one independent variable depends on the value of another independent variable. We will explore here the interaction between two dummy (binary) variables. In the example below there could be the case that the effect of student-teacher ratio on test scores may depend on the percent of English learners in the district*.

- Dependent variable (Y) – Average test score, variable `testscr` in dataset.
- Independent variables (X)
  - Binary `hi_str`, where '0' if student-teacher ratio (`str`) is lower than 20, '1' equal to 20 or higher.
    - In Stata, first `generate hi_str = 0 if str<20`. Then `replace hi_str=1 if str>=20`.
  - Binary `hi_el`, where '0' if English learners (`el_pct`) is lower than 10%, '1' equal to 10% or higher
    - In Stata, first `generate hi_el = 0 if el_pct<10`. Then `replace hi_el=1 if el_pct>=10`.
  - Interaction term `str_el = hi_str * hi_el`. In Stata: `generate str_el = hi_str*hi_el`

We run the regression

```
regress testscr hi_el hi_str str_el, robust
```

```
. regress  testscr  hi_el hi_str str_el, robust

Linear regression                               Number of obs =     420
                                                F(  3,   416) =   60.20
                                                Prob > F      =  0.0000
                                                R-squared     =  0.2956
                                                Root MSE      =  16.049

------------------------------------------------------------------------------
             |               Robust
    testscr  |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       hi_el |  -18.16295   2.345952    -7.74   0.000    -22.77435   -13.55155
      hi_str |  -1.907842   1.932215    -0.99   0.324    -5.705964    1.890279
      str_el |  -3.494335   3.121226    -1.12   0.264    -9.629677    2.641006
       _cons |   664.1433   1.388089   478.46   0.000     661.4147    666.8718
------------------------------------------------------------------------------
```

The equation is `testscr_hat = 664.1 – 18.1*hi_el – 1.9*hi_str – 3.5*str_el`

The effect of `hi_str` on the tests scores is -1.9 but given the interaction term (and assuming all coefficients are significant), the net effect is `-1.9 -3.5*hi_el`. If `hi_el` is 0 then the effect is -1.9 (which is `hi_str` coefficient), but if `hi_el` is 1 then the effect is -1.9 -3.5 = -5.4. In this case, the effect of student-teacher ratio is more negative in districts where the percent of English learners is higher.

See the next slide for more detailed computations.

*The data used in this section is the "California Test Score" data set (`caschool.dta`) from chapter 6 of the book *Introduction to Econometrics* from Stock and Watson, 2003. Data can be downloaded from http://wps.aw.com/aw_stock_ie_2/50/13016/3332253.cw/index.html.For a detailed discussion please refer to the respective section in the book.

Regression: interaction between dummies (cont.)

You can compute the expected values of test scores given different values of `hi_str` and `hi_el`. To see the effect of `hi_str` given `hi_el` type the following right after running the regression in the previous slide.

```
. predict yhat1 if hi_str==0 & hi_el==0
(option xb assumed; fitted values)
(271 missing values generated)

. predict yhat2 if hi_str==1 & hi_el==0
(option xb assumed; fitted values)
(341 missing values generated)

. predict yhat3 if hi_str==0 & hi_el==1
(option xb assumed; fitted values)
(331 missing values generated)

. predict yhat4 if hi_str==1 & hi_el==1
(option xb assumed; fitted values)
(317 missing values generated)
```

These are different scenarios holding constant `hi_el` and varying `hi_str`. Below we add some labels

```
. label variable yhat1 "Low str/Low el"

. label variable yhat2 "High str/Low el"

. label variable yhat3 "Low str/High el"

. label variable yhat4 "High str/High el"
```

We then obtain the average of the estimations for the test scores (for all four scenarios, notice same values for all cases).

```
. summarize yhat1 yhat2 yhat3 yhat4
```

Variable	Obs	Mean	Std. Dev.	Min	Max
yhat1	149	664.1433	0	664.1433	664.1433
yhat2	79	662.2355	0	662.2355	662.2355
yhat3	89	645.9803	0	645.9803	645.9803
yhat4	103	640.5782	0	640.5782	640.5782

```
. display 664.1 - 662.2
1.9

. display 645.9 - 640.5
5.4

. display 5.4 - 1.9
3.5
```

Here we estimate the net effect of low/high student-teacher ratio holding constant the percent of English learners. When `hi_el` is 0 the effect of going from low to high student-teacher ratio goes from a score of 664.2 to 662.2, a difference of 1.9. From a policy perspective you could argue that moving from high str to low str improve test scores by 1.9 in low English learners districts.

When `hi_el` is 1, the effect of going from low to high student-teacher ratio goes from a score of 645.9 down to 640.5, a decline of 5.4 points (1.9+3.5). From a policy perspective you could say that reducing the str in districts with high percentage of English learners could improve test scores by 5.4 points.

*The data used in this section is the "California Test Score" data set (`caschool.dta`) from chapter 6 of the book *Introduction to Econometrics* from Stock and Watson, 2003. Data can be downloaded from http://wps.aw.com/aw_stock_ie_2/50/13016/3332253.cw/index.html.For a detailed discussion please refer to the respective section in the book.

Regression: interaction between a dummy and a continuous variable

Lets explore the same interaction as before but we keep student-teacher ratio continuous and the English learners variable as binary. The question remains the same*.

- Dependent variable (Y) – Average test score, variable `testscr` in dataset.
- Independent variables (X)
  - Continuous `str`, student-teacher ratio.
  - Binary `hi_el`, where '0' if English learners (`el_pct`) is lower than 10%, '1' equal to 10% or higher
  - Interaction term `str_el2 = str * hi_el.` In Stata: `generate str_el2 = str*hi_el`

We will run the regression

```
regress testscr str hi_el str_el2, robust
```

```
. regress testscr str hi_el str_el2, robust

Linear regression                               Number of obs =     420
                                                F(  3,   416) =   63.67
                                                Prob > F      =  0.0000
                                                R-squared     =  0.3103
                                                Root MSE      =   15.88

------------------------------------------------------------------------------
             |               Robust
    testscr |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         str |  -.9684601   .5891016    -1.64   0.101    -2.126447    .1895268
       hi_el |   5.639141   19.51456     0.29   0.773    -32.72029    43.99857
     str_el2 |  -1.276613   .9669194    -1.32   0.187     -3.17727    .6240436
       _cons |   682.2458   11.86781    57.49   0.000     658.9175    705.5742
------------------------------------------------------------------------------
```

The equation is `testscr_hat = 682.2 – 0.97*str + 5.6*hi_el – 1.28*str_el2`

The effect of `str` on `testscr` will be mediated by `hi_el`.

- If `hi_el` is 0 (low) then the effect of `str` is 682.2 – 0.97*str.
- If `hi_el` is 1 (high) then the effect of `str` is 682.2 – 0.97*str + 5.6 – 1.28*str = 687.8 – 2.25*str

Notice that how `hi_el` changes both the intercept and the slope of `str`. Reducing `str` by one in low EL districts will increase test scores by 0.97 points, but it will have a higher impact (2.25 points) in high EL districts. The difference between these two effects is 1.28 which is the coefficient of the interaction (Stock and Watson, 2003, p.223).

*The data used in this section is the "California Test Score" data set (`caschool.dta`) from chapter 6 of the book *Introduction to Econometrics* from Stock and Watson, 2003. Data can be downloaded from http://wps.aw.com/aw_stock_ie_2/50/13016/3332253.cw/index.html.For a detailed discussion please refer to the respective section in the book.

Regression: interaction between two continuous variables

Lets keep now both variables continuous. The question remains the same*.

- Dependent variable (Y) – Average test score, variable `testscr` in dataset.
- Independent variables (X)
  - Continuous `str`, student-teacher ratio.
  - Continuous `el_pct`, percent of English learners.
  - Interaction term `str_el3 = str * el_pct.` In Stata: `generate str_el3 = str*el_pct`

We will run the regression

```
regress testscr str el_pct str_el3, robust
```

```
. regress testscr str el_pct str_el3, robust

Linear regression                               Number of obs =     420
                                                F(  3,   416) =  155.05
                                                Prob > F      =  0.0000
                                                R-squared     =  0.4264
                                                Root MSE      =  14.482

------------------------------------------------------------------------------
             |               Robust
    testscr |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         str |  -1.117018   .5875135    -1.90   0.058    -2.271884    .0378468
      el_pct |  -.6729116   .3741231    -1.80   0.073    -1.408319    .0624958
      str_el3 |   .0011618   .0185357     0.06   0.950    -.0352736    .0375971
       _cons |   686.3385   11.75935    58.37   0.000     663.2234    709.4537
------------------------------------------------------------------------------
```

The equation is `testscr_hat = 686.3 – 1.12*str - 0.67*el_pct + 0.0012*str_el3`

The effect of the interaction term is very small. Following Stock and Watson (2003, p.229), algebraically the slope of `str` is `–1.12 + 0.0012*el_pct` (remember that `str_el3` is equal to `str*el_pct`). So:

- If el_pct = 10, the slope of str is -1.108
- If el_pct = 20, the slope of str is -1.096. A difference in effect of 0.012 points.

In the continuous case there is an effect but is very small (and not significant). See Stock and Watson, 2003, for further details.

*The data used in this section is the "California Test Score" data set (`caschool.dta`) from chapter 6 of the book *Introduction to Econometrics* from Stock and Watson, 2003. Data can be downloaded from http://wps.aw.com/aw_stock_ie_2/50/13016/3332253.cw/index.html.For a detailed discussion please refer to the respective section in the book.

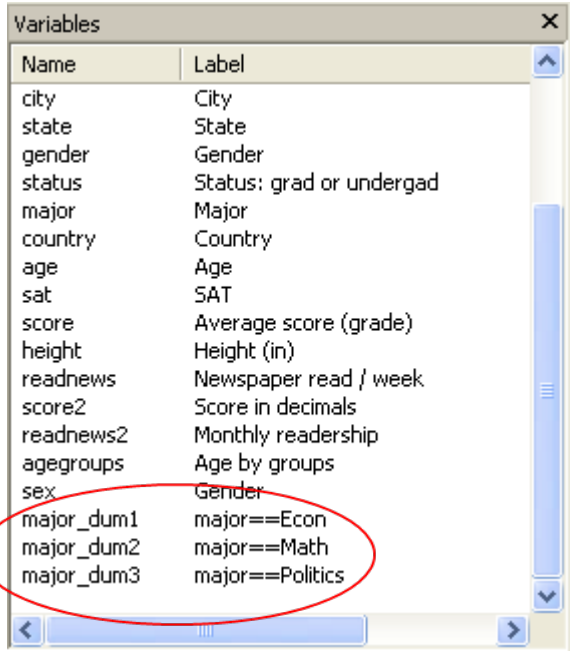
Creating dummies

You can create dummy variables by either using `recode` or using a combination of `tab/gen` commands:

`tab major, generate(major_dum)`

```
. tab major, generate(major_dum)

      Major |      Freq.     Percent        Cum.
------------+-----------------------------------
      Econ |         10       33.33       33.33
      Math |         10       33.33       66.67
  Politics |         10       33.33      100.00
------------+-----------------------------------
      Total |         30      100.00
```

Check the 'variables' window, at the end you will see three new variables. Using `tab1` (for multiple frequencies) you can check that they are all 0 and 1 values




```
. tab1  major_dum1 major_dum2 major_dum3

-> tabulation of major_dum1

major==Econ |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         20       66.67       66.67
          1 |         10       33.33      100.00
------------+-----------------------------------
      Total |         30      100.00

-> tabulation of major_dum2

major==Math |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         20       66.67       66.67
          1 |         10       33.33      100.00
------------+-----------------------------------
      Total |         30      100.00

-> tabulation of major_dum3

major==Poli |
       tics |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         20       66.67       66.67
          1 |         10       33.33      100.00
------------+-----------------------------------
      Total |         30      100.00
```

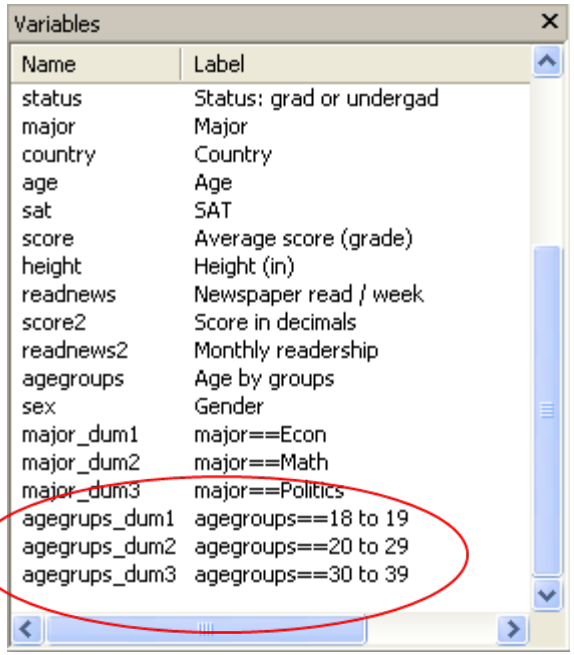
Creating dummies (cont.)

Here is another example:

`tab agregroups, generate(agegroups_dum)`

```
. tab agegroups, generate(agegrups_dum)

    Age by |
    groups |      Freq.     Percent        Cum.
-----------+-----------------------------------
 18 to 19  |         10       33.33       33.33
 20 to 29  |          9       30.00       63.33
 30 to 39  |         11       36.67      100.00
-----------+-----------------------------------
     Total |         30      100.00
```

Check the 'variables' window, at the end you will see three new variables. Using `tab1` (for multiple frequencies) you can check that they are all 0 and 1 values




```
. tab1  agegrups_dum1 agegrups_dum2 agegrups_dum3

-> tabulation of agegrups_dum1

agegroups== |
  18 to 19  |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         20       66.67       66.67
          1 |         10       33.33      100.00
------------+-----------------------------------
      Total |         30      100.00

-> tabulation of agegrups_dum2

agegroups== |
  20 to 29  |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         21       70.00       70.00
          1 |          9       30.00      100.00
------------+-----------------------------------
      Total |         30      100.00

-> tabulation of agegrups_dum3

agegroups== |
  30 to 39  |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         19       63.33       63.33
          1 |         11       36.67      100.00
------------+-----------------------------------
      Total |         30      100.00
```

Frequently used Stata commands

Category	Stata commands
Getting on-line help	**help** **search**
Operating-system interface	**pwd** **cd** **sysdir** **mkdir** **dir / ls** **erase** **copy** **type**
Using and saving data from disk	**use** **clear** **save** **append** **merge** **compress**
Inputting data into Stata	**input** **edit** **infile** **infix** **insheet**
The Internet and Updating Stata	**update** **net** **ado** **news**
Basic data reporting	**describe** **codebook** **inspect** **list** **browse** **count** **assert** **summarize** **Table (tab)** **tabulate**
Data manipulation	**generate** **replace** **egen** **recode** **rename** **drop** **keep** **sort** **encode** **decode** **order** **by** **reshape**
Formatting	**format** **label**
Keeping track of your work	**log** **notes**
Convenience	**display**

Type `help [command name]` in the windows command for details

Source: http://www.ats.ucla.edu/stat/stata/notes2/commands.htm

Is my model OK? (links)

Regression diagnostics: A checklist
http://www.ats.ucla.edu/stat/stata/webbooks/reg/chapter2/statareg2.htm

Logistic regression diagnostics: A checklist
http://www.ats.ucla.edu/stat/stata/webbooks/logistic/chapter3/statalog3.htm

Times series diagnostics: A checklist (pdf)
http://homepages.nyu.edu/~mrg217/timeseries.pdf

Times series: dfueller test for unit roots (for R and Stata)
http://www.econ.uiuc.edu/~econ472/tutorial9.html

Panel data tests: heteroskedasticity and autocorrelation

- http://www.stata.com/support/faqs/stat/panel.html
- http://www.stata.com/support/faqs/stat/xtreg.html
- http://www.stata.com/support/faqs/stat/xt.html
- http://dss.princeton.edu/online_help/analysis/panel.htm

I can't read the output of my model!!! (links)

Data Analysis: Annotated Output
http://www.ats.ucla.edu/stat/AnnotatedOutput/default.htm

Data Analysis Examples
http://www.ats.ucla.edu/stat/dae/

Regression with Stata
http://www.ats.ucla.edu/STAT/stata/webbooks/reg/default.htm

Regression
http://www.ats.ucla.edu/stat/stata/topics/regression.htm

How to interpret dummy variables in a regression
http://www.ats.ucla.edu/stat/Stata/webbooks/reg/chapter3/statareg3.htm

How to create dummies
http://www.stata.com/support/faqs/data/dummy.html
http://www.ats.ucla.edu/stat/stata/faq/dummy.htm

Logit output: what are the odds ratios?
http://www.ats.ucla.edu/stat/stata/library/odds_ratio_logistic.htm

Topics in Statistics (links)

What statistical analysis should I use?
http://www.ats.ucla.edu/stat/mult_pkg/whatstat/default.htm

***Statnotes: Topics in Multivariate Analysis*, by G. David Garson**
http://www2.chass.ncsu.edu/garson/pa765/statnote.htm

Elementary Concepts in Statistics
http://www.statsoft.com/textbook/stathome.html

Introductory Statistics: Concepts, Models, and Applications
http://www.psychstat.missouristate.edu/introbook/sbk00.htm

Statistical Data Analysis
http://math.nicholls.edu/badie/statdataanalysis.html

Stata Library. Graph Examples (some may not work with STATA 10)
http://www.ats.ucla.edu/STAT/stata/library/GraphExamples/default.htm

Comparing Group Means: The T-test and One-way ANOVA Using STATA, SAS, and SPSS
http://www.indiana.edu/~statmath/stat/all/ttest/

Useful links / Recommended books

- DSS Online Training Section http://dss.princeton.edu/training/
- UCLA Resources to learn and use STATA http://www.ats.ucla.edu/stat/stata/
- DSS help-sheets for STATA http://dss/online_help/stats_packages/stata/stata.htm
- *Introduction to Stata* (PDF), Christopher F. Baum, Boston College, USA. "A 67-page description of Stata, its key features and benefits, and other useful information." http://fmwww.bc.edu/GStat/docs/StataIntro.pdf
- STATA FAQ website http://stata.com/support/faqs/
- Princeton DSS Libguides http://libguides.princeton.edu/dss

Books

- *Introduction to econometrics* / James H. Stock, Mark W. Watson. 2nd ed., Boston: Pearson Addison Wesley, 2007.
- *Data analysis using regression and multilevel/hierarchical models* / Andrew Gelman, Jennifer Hill. Cambridge ; New York : Cambridge University Press, 2007.
- *Econometric analysis* / William H. Greene. 6th ed., Upper Saddle River, N.J. : Prentice Hall, 2008.
- *Designing Social Inquiry: Scientific Inference in Qualitative Research* / *Gary King, Robert O. Keohane, Sidney Verba, Princeton University Press*, 1994.
- *Unifying Political Methodology: The Likelihood Theory of Statistical Inference* / Gary King, Cambridge University Press, 1989
- *Statistical Analysis: an interdisciplinary introduction to univariate & multivariate methods* / Sam Kachigan, New York : Radius Press, c1986
- *Statistics with Stata (updated for version 9)* / Lawrence Hamilton, Thomson Books/Cole, 2006