

Unit 6

Stata for Graphs *version 16*

Amazing.



Table of Contents

Topic	Page
Learning Objectives	3
Sample Session	4
2. Stata for Graphs	8
2.1 Choose Your Scheme	8
2.2 The Basics	13
(a) Basic Syntax for a Graph	15
(b) Choose Your Plotting <u>Symbol</u> using msymbol()	17
(c) Choose Your Plotting Symbol <u>Size</u> using msize()	18
(d) Choose Your Plotting Symbol <u>Color</u> using mcolor()	19
2.3 Use the Graph Editor to Change How Your Graph Looks	20
2.4 How to Save your Graph	22
3. Discrete Variable Graphs	23
3.1 Bar Chart using histogram, discrete	23
3.2 Grouped Bar Graph using histogram, discrete	25
3.3 Grouped Bar Graph using catplot	27
3.4 Grouped Bar Graph using graph combine	29
4. Continuous Variable Graphs	32
4.1 Dot Plot using Dotplot	32
4.2 Stem and Leaf Plot using Stem or Stemplot	34
4.3 Box and Whisker Plot using Graph Box	36
4.4 Histogram using Histogram	39
4.5 Scatter Plot using Graph Twoway	43
4.6 Matrix Plot using Graph Matrix	46
5. Repeated Measures Graphs	47
5.1 Choose Your Data Layout using Reshape	48
(a) From WIDE to LONG using Reshape long	49
(b) From LONG to WIDE using Reshape wide	51
5.2 (Long) Line Plot or Profile Plot using xtset and xtline	53
5.3 (Long) Line Plot or Profile Plot using xtset and Graph Twoway	54
5.4 (Long) Mean Profile Plot using Collapse and Graph Twoway	56
5.5 (Long) Side-by-side Box and Whisker Plot using Graph box	57
5.5 (Wide) Mean Profile Plot using profileplot	58
Appendices	59
A1. Color palette	59
A2. Symbol palette	60
A3. Marker size palette	61
A4. Line type palette	62

Learning Objectives

When you have finished this unit, you should be able to use Stata to produce selected graphical summaries of :

- single variable distributions, discrete and continuous;
- single variable distributions, discrete and continuous, grouped;
- scatterplots; and
- repeated measures data.

Suggestion –follow along!

These notes have been written so that you can practice the commands as you read and follow along.

Sample Session

How to follow along:

- 1) Download from the course website the data set [relate100obs.dta](#).
- 2) Launch Stata
- 3) From whatever directory is appropriate for you, open relate100obs.dta

Sample session green-comments black-commands blue-results

```
. * ----- Use cd command to set desktop as working directory -----
. cd "/Users/cbigelow/Desktop/"
  /Users/cbigelow/Desktop/
. set more off

. * Input raw data. NOTE – Your command here might be different.
. use "/Users/cbigelow/Desktop/relate100obs.dta"

. * Rename variables to be more meaningful
. rename R3483600 m_praise
. rename R3485200 f_praise
. rename R3828100 age

. * Convert missing value codes to Stata missing values
. mvdecode m_praise,mv(-1=.r\ -2=.d\ -4=.s\ -5=.)
  m_praise: 43 missing values generated

. mvdecode f_praise,mv(-1=.r\ -2=.d\ -4=.s\ -5=.)
  f_praise: 66 missing values generated

. mvdecode age,mv(-1=.r\ -2=.d\ -4=.s\ -5=.)
  age: 4 missing values generated

. * Create variable labels
. label variable m_praise "m_praise: Mother praises R for doing well"
. label variable f_praise "f_praise: Father praises R for doing well"
. label variable age "age: Age of R (years)"
```

Sample session, continued: green-comments black-commands blue-results

```
. * Create value labels
. label define PRAISEF 0 "never" 1 "rarely" 2 "sometimes" 3 "usually" 4 "always"

.* Apply value labels to variables
. label values m_praise PRAISEF
. label values f_praise PRAISEF

* Keep only the variables of interest. Save under new name.
. keep m_praise f_praise age
. save relatenew100, replace

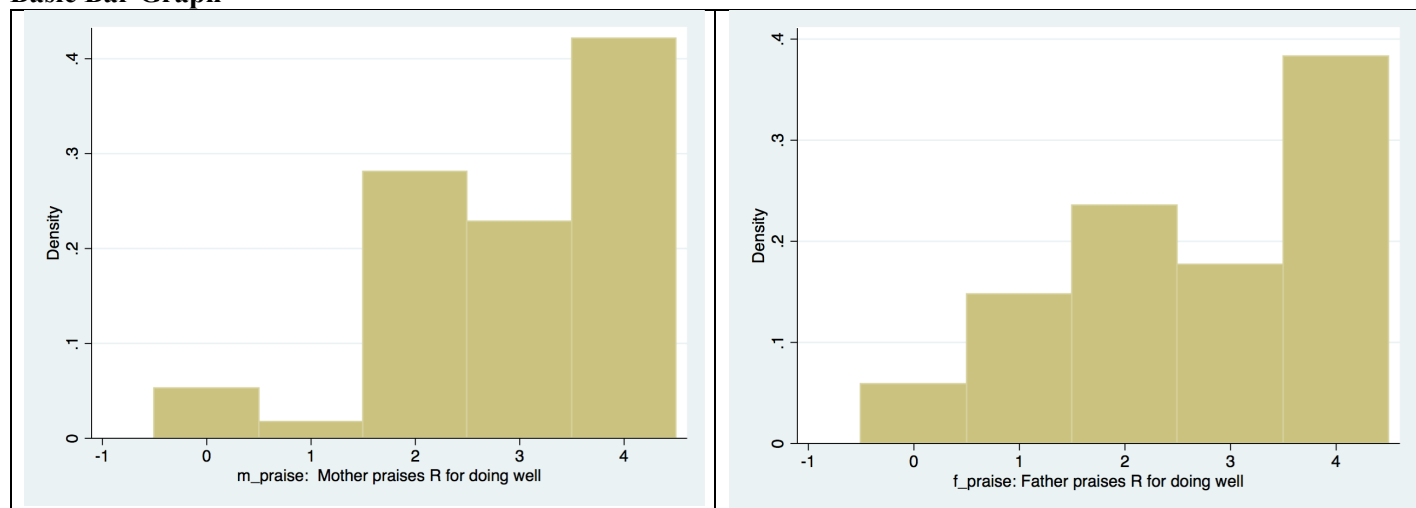
(note: file relatenew100.dta not found)
file relatenew100.dta saved
```

```
. * BAR CHART FOR SINGLE DISCRETE VARIABLE: command histogram with option discrete

* Basic Bar Graph - NO aesthetics
. histogram m_praise, discrete
. histogram f_praise, discrete
```

This yields the following

Basic Bar Graph



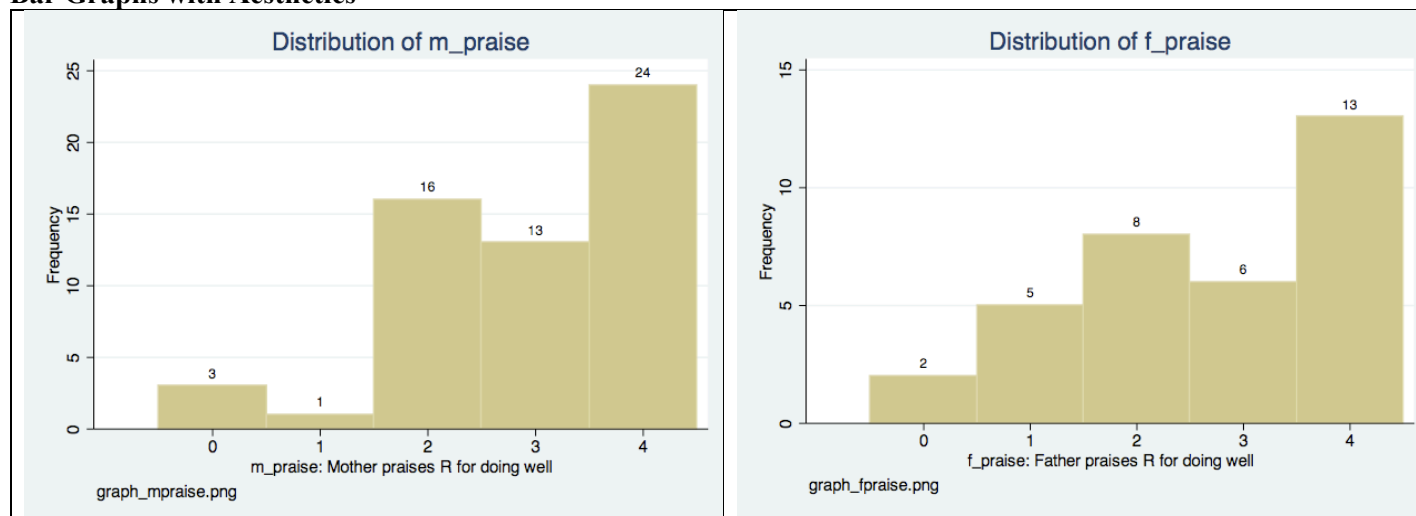
Note – This is okay, but we might like some annotations; e.g. – titles, etc.



Sample session, continued: green-comments black-commands blue-results

```
. * BAR CHART FOR SINGLE DISCRETE VARIABLE: command histogram with option discrete
. * Bar Graphs with frills – Aesthetics added
. histogram m_praise,discrete frequency addlabels ylabel(,grid) xlabel(0(1)4)
. title("Distribution of m_praise") caption("graph_mpraise.png")
. histogram f_praise,discrete frequency addlabels ylabel(,grid) xlabel(0(1)4)
. title("Distribution of f_praise") caption("graph_fpraise.png")
```

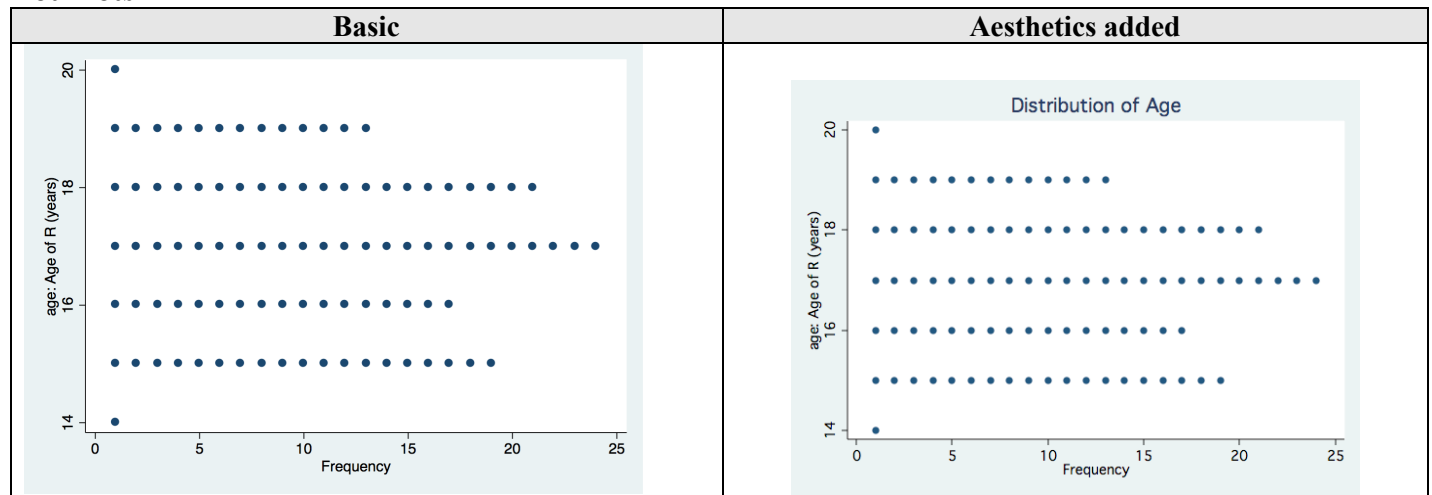
Bar Graphs with Aesthetics



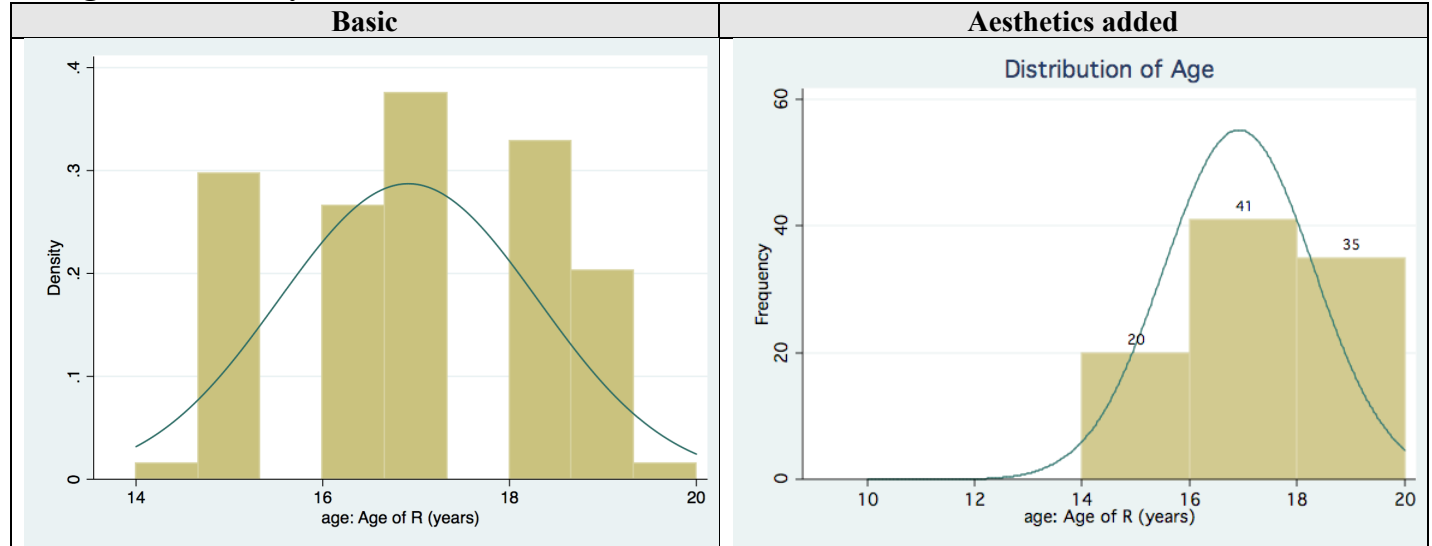
```
. * DOT PLOT FOR SINGLE CONTINUOUS VARIABLE: command dotplot
. * Basic dot plot – NO aesthetics
. dotplot age
. * Dot plot with frills – Aesthetics added
. dotplot age,nogroup title("Distribution of Age")

. * HISTOGRAM w OVERLAY NORMAL FOR SINGLE CONTINUOUS: command histogram with option normal
. * Basic histogram w overlay normal – NO aesthetics
. histogram age, normal
. * Histogram w overlay normal – Aesthetics added
. histogram age, frequency addlabels normal start(10) width(2) title("Distribution of Age")
(bin=5, start=10, width=2)
```

Dot Plots



Histogram w Overlay Normal



2. Stata for Graphs

How to follow along:

This section utilizes a data set that Stata provides - **auto.dta**. You do NOT download it from the course website. Use the **sysuse** command to access:

```
. clear
. sysuse auto
```

2.1 Choose Your Scheme

Stata offers a command called **scheme** that defines the **overall appearance** of your graph. This has to do with whether or not there is a box around your plot, whether or not there is shading, the color of the lines and bars, etc.

The default scheme is **s2color**.

To obtain a listing of the available schemes

Type the following in the command window: **. graph query, schemes**

Note! You may get a slightly smaller set of schemes. This is because I have downloaded some additional schemes.

. graph query, schemes	
Available schemes are	meta
s2color see help scheme_s2color	vg_blue
s2mono see help scheme_s2mono	vg_brite
s2manual see help scheme_s2manual	vg_lgndc
s2gmanual see help scheme_s2gmanual	vg_lgndm
s2gcolor see help scheme_s2gcolor	vg_outc
s1color see help scheme_s1color	vg_outm
s1mono see help scheme_s1mono	vg_palec
s1rcolor see help scheme_s1rcolor	vg_palem
s1manual see help scheme_s1manual	vg_past
sj see help scheme_sj	vg_rose
economist see help scheme_economist	vg_slc
s2color8 see help scheme_s2color8	vg_slm
	vg_s2c
	vg_s2m
	vg_samec
	vg_samem
	vg_size
	vg_teal

There are two ways to set the graph scheme

Method 1: Using the **set scheme** command prior to specifying your graph

. set scheme *schemename*

Example: **. set scheme** lean1

Method 2: Using the graph option **scheme()** within your graph command

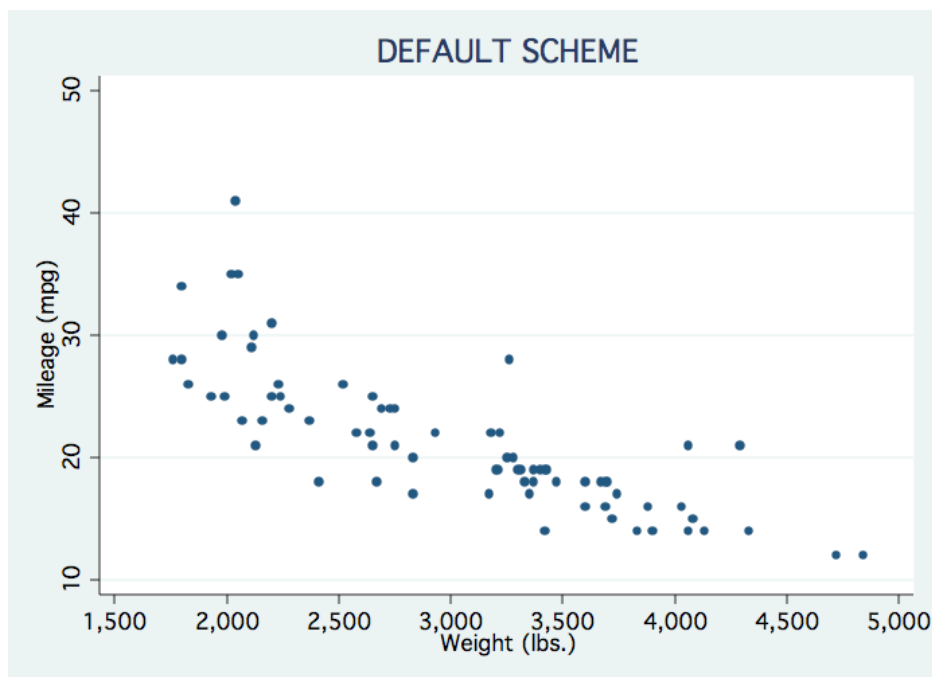
, scheme() *schemename*

Example: **, scheme()** lean1

Illustrations of Selected Graph Schemes

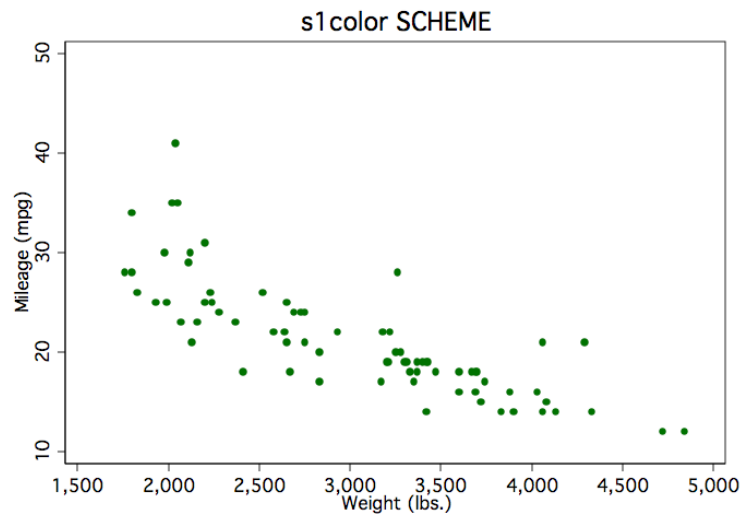
Default is s2color (no changes made yet)

```
. * DEFAULT SCHEME
. scatter mpg weight, title("DEFAULT SCHEME") xlabel(1500(500)5000) ylabel(10(10)50) msymbol(o)
```



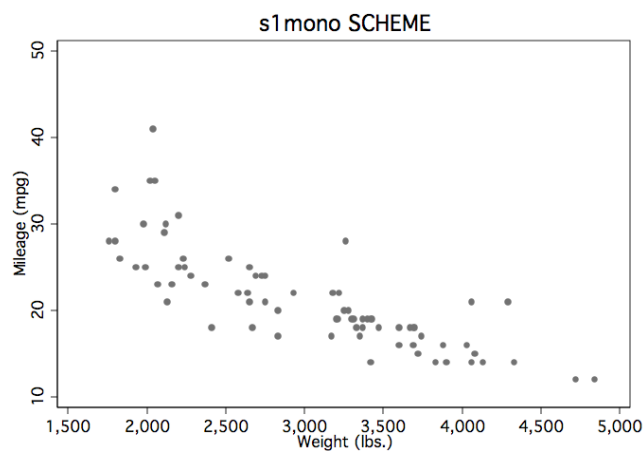
s1color

```
. * s1color SCHEME
. set scheme s1color
. scatter mpg weight, title("s1color SCHEME") xlabel(1500(500)5000) ylabel(10(10)50) msymbol(o)
```



s1mono

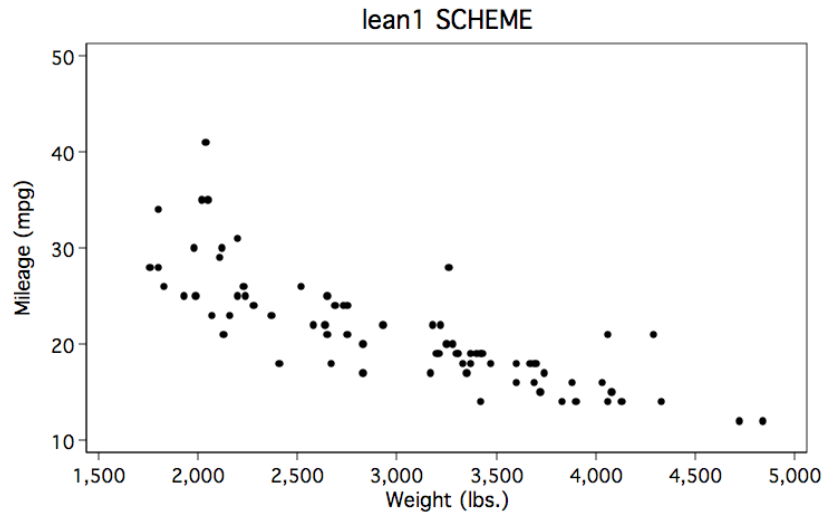
```
. * s1mono
. set scheme s1mono
. scatter mpg weight, title("s1mono SCHEME") xlabel(1500(500)5000) ylabel(10(10)50) msymbol(o)
```



lean1

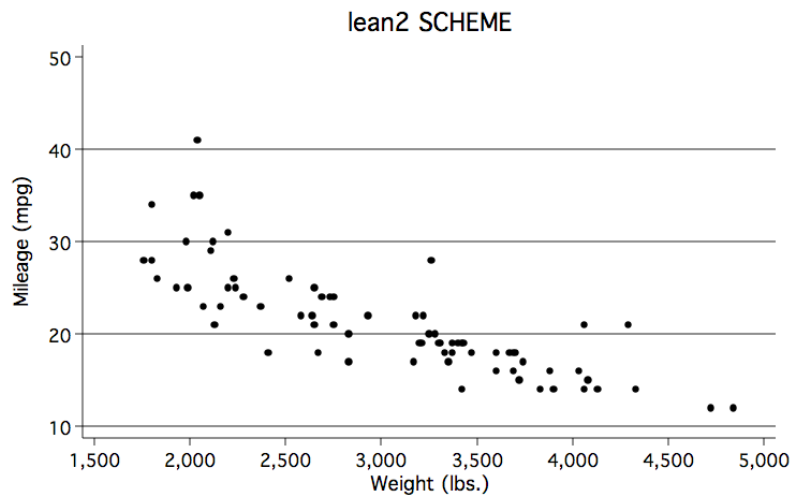


```
. * lean1
. set scheme lean1
. scatter mpg weight, title("lean1 SCHEME") xlabel(1500(500)5000) ylabel(10(10)50) msymbol(o)
```



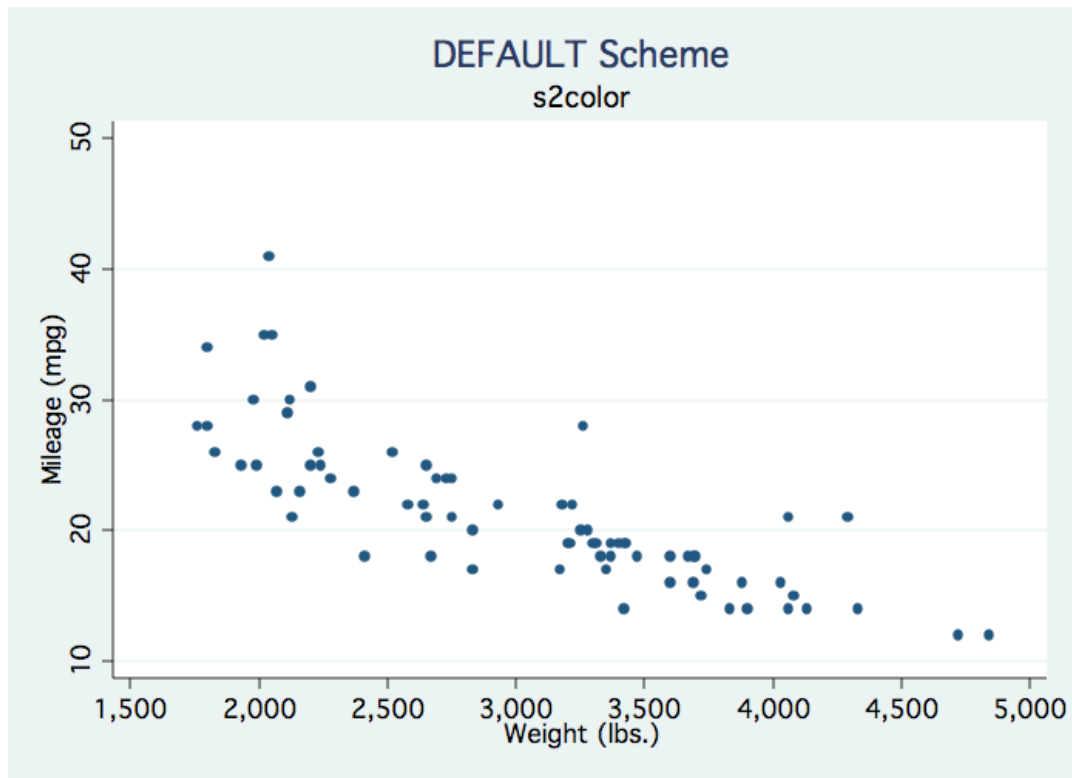
lean2

```
. * lean1
. set scheme lean2
. scatter mpg weight, title("lean2 SCHEME") xlabel(1500(500)5000) ylabel(10(10)50) msymbol(o)
```



Return to Default Scheme

```
. * Return to default scheme
. set scheme s2color
. scatter mpg weight, title("DEFAULT Scheme") subtitle("s2color") xlabel(1500(500)5000)
  ylabel(10(10)50) msymbol(o)
```



Tip! How To Set Your Favorite Scheme as the default

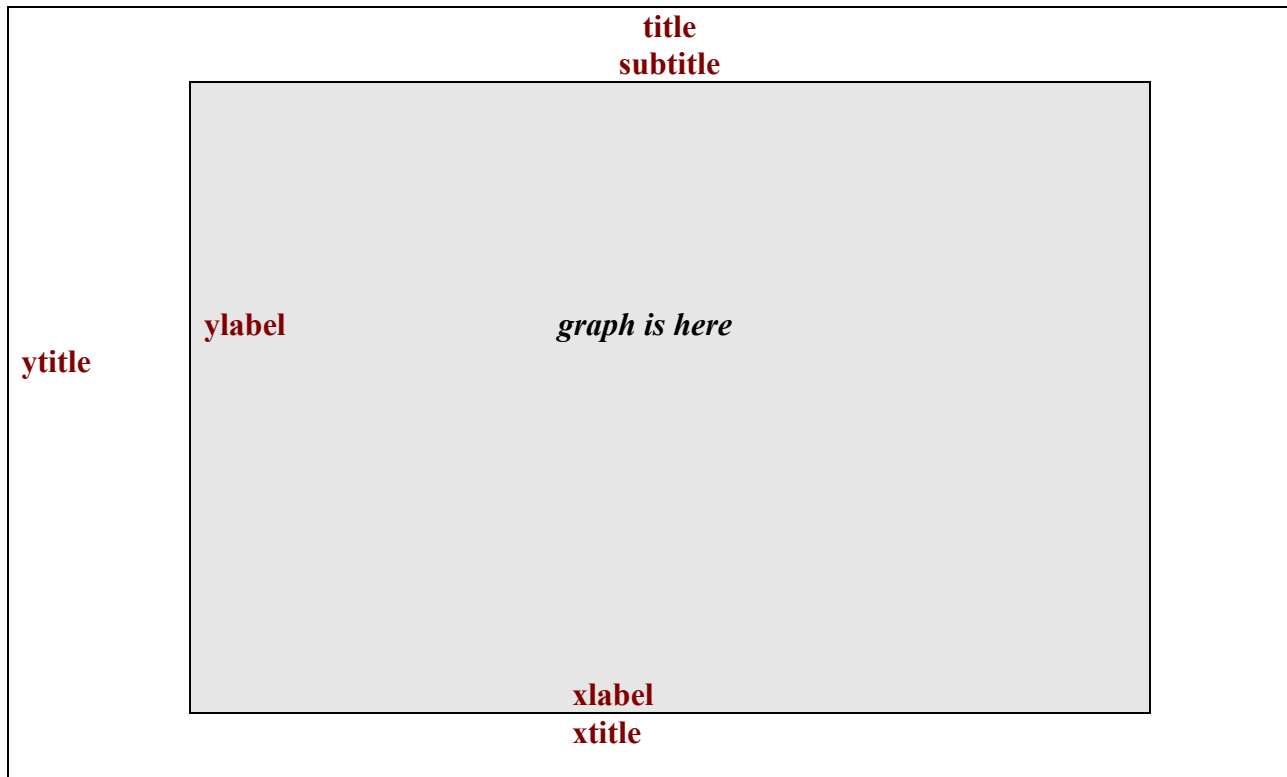
Suppose you want the scheme **lean1** to be the default scheme.
Type the following in the command window

- set scheme lean1, permanently

2.2 The Basics

A Stata graph is comprised of: (1) the actual graph; (2) plot options (**eg – xlabel**) ; and (2) graph options (**eg – title**)

Schematic (*partial*) of Stata Graph Specifications



Tip! Keep this page handy. When you get a little further along and are doing aesthetics (setting titles, labels, etc) this schematic will remind you of the STATA naming conventions.

Example

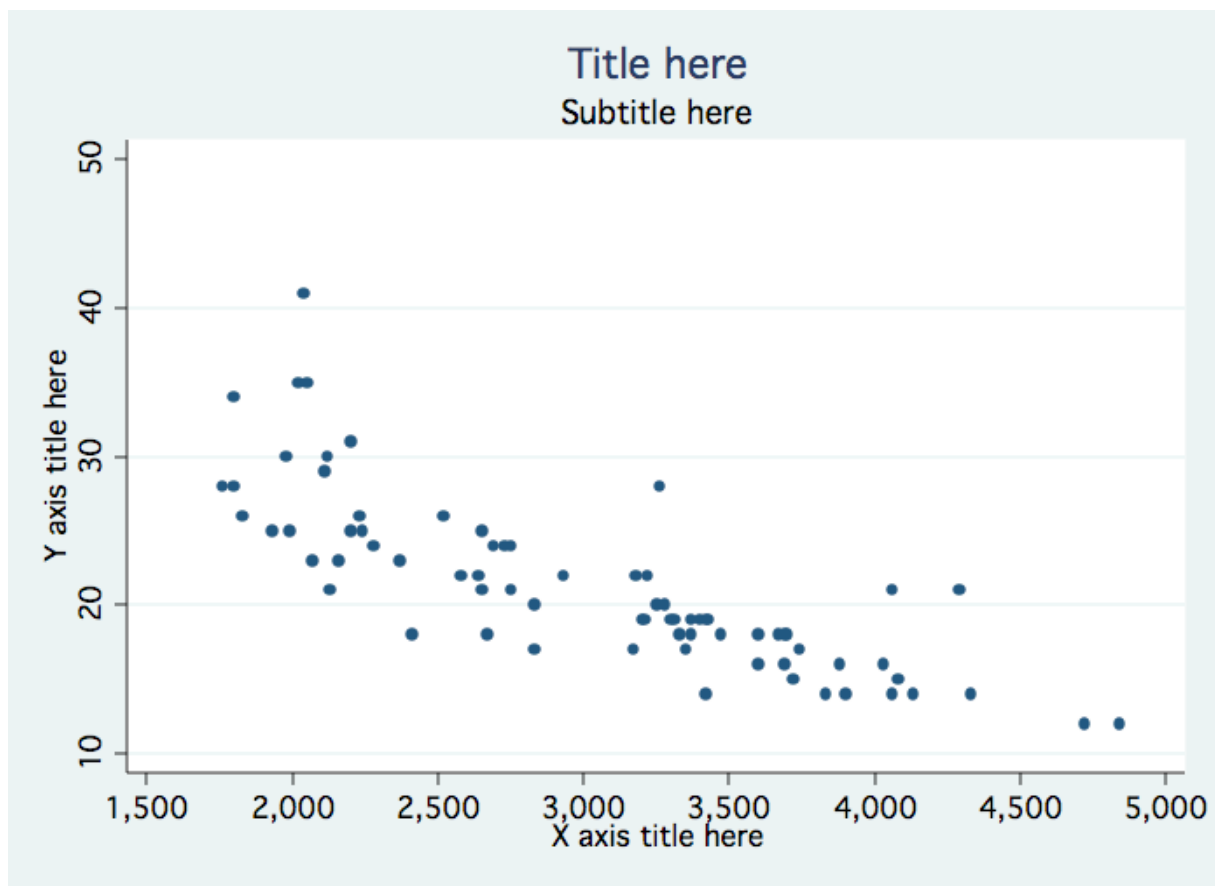
green-comments black-commands blue-results

```
. * Illustration of Scatterplot of X versus Y
. * Y=mpg and X=weight
. * Obtain min and max of each variable so as to set tick marks explicitly

. tabstat mpg weight, stat(min max)
```

stats	mpg	weight
min	12	1760
max	41	4840

```
. * Illustration of scatterplot with specification of titles, axis labels etc.
. scatter mpg weight, title("Title here") subtitle("Subtitle here") xtitle("X axis title here")
  ytitle("Y axis title here") xlabel(1500(500)5000) ylabel(10(10)50) msymbol(o)
```



(a) Basic Syntax for a Graph

```
.graph graphchoice (plot_choice, plot_options) (plot_choice, plot_options), graph_options
```

↑ ↑ ↑
Graph options: Note this comma! Note this comma! Note this comma!

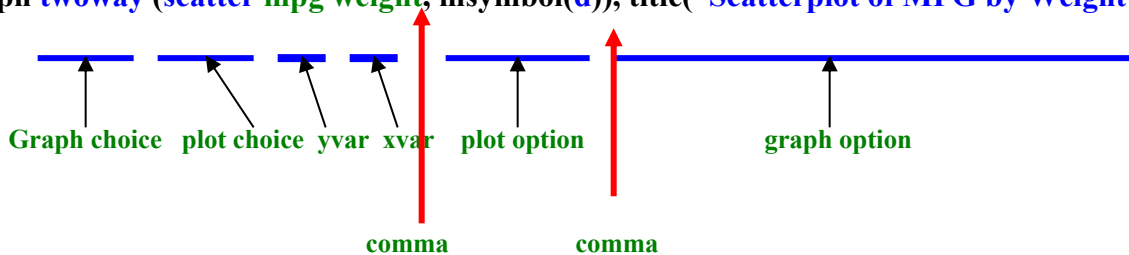
Partial listing ...

- title("title in quotes") - specify title
- subtitle("subtitle in quotes") - specify subtitle
- ytitle("Y-axis title in quotes") - specify Y-axis title
- xtitle("X-axis title in quotes") - specify X-axis title
- legend ("legend in quotes") - specify legend
- caption("caption in quotes") - specify caption
- note("note in quotes") - specify note

Beware! It is not always necessary to type “graph” as the first word in the command line. In fact, sometimes, it is incorrect. See examples below.

Example

```
.graph twoway (scatter mpg weight, msymbol(d)), title("Scatterplot of MPG by Weight")
```

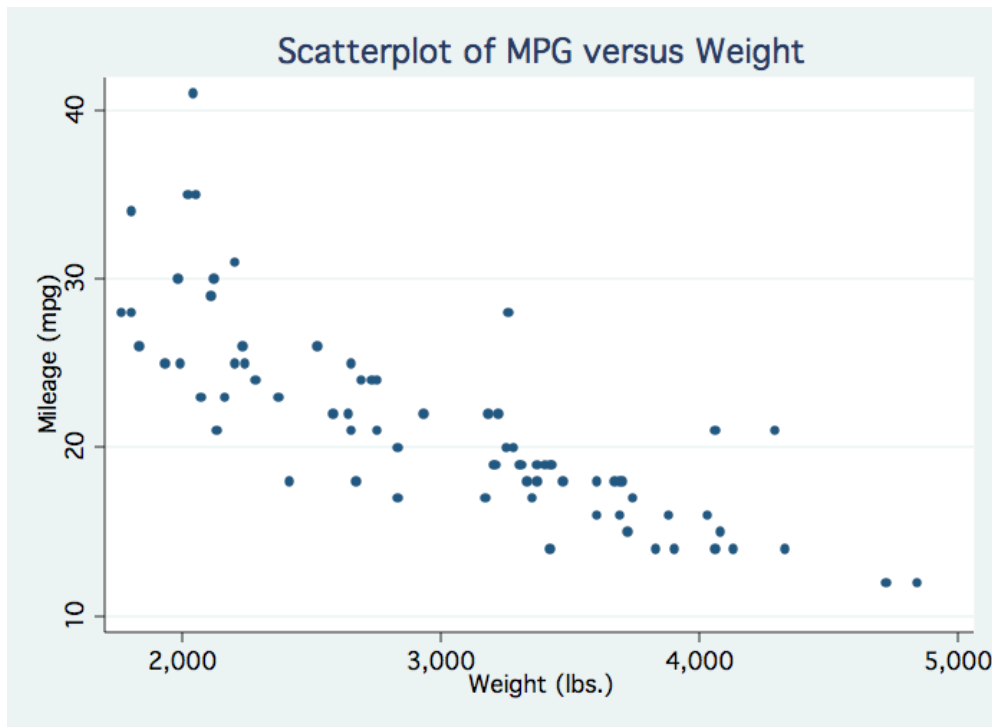


Important Tips to Remember!

Pay attention to spaces:

- (1) There **MUST** be a space between “twoway” and the following parenthesis
- (2) There must **NOT** be a space between “title” and the opening parenthesis that follows.

.graph twoway (scatter mpg weight, msymbol(d)), title("Scatterplot of MPG by Weight")



(b) Choose Your Plotting Symbol using `msymbol()`

`msymbol()` Choices

.* Stata will show you the palette available with the command: `palette symbolpalette`
`.palette symbolpalette`

Symbol palette

● O	○ Oh	● o	○ oh
◆ D	◇ Dh	◆ d	◇ dh
▲ T	△ Th	▲ t	△ th
■ S	□ Sh	■ s	□ sh
+ +		+ smplus	
× X		× x	
		· p	

(symbols shown at larger than default size)

(c) Choose Your Plotting Symbol Size using **msize()**

msize() Choices

.* Stata will show you the marker sizes with the command: **showmarkers, over(msize)**
.showmarkers, over(msize)

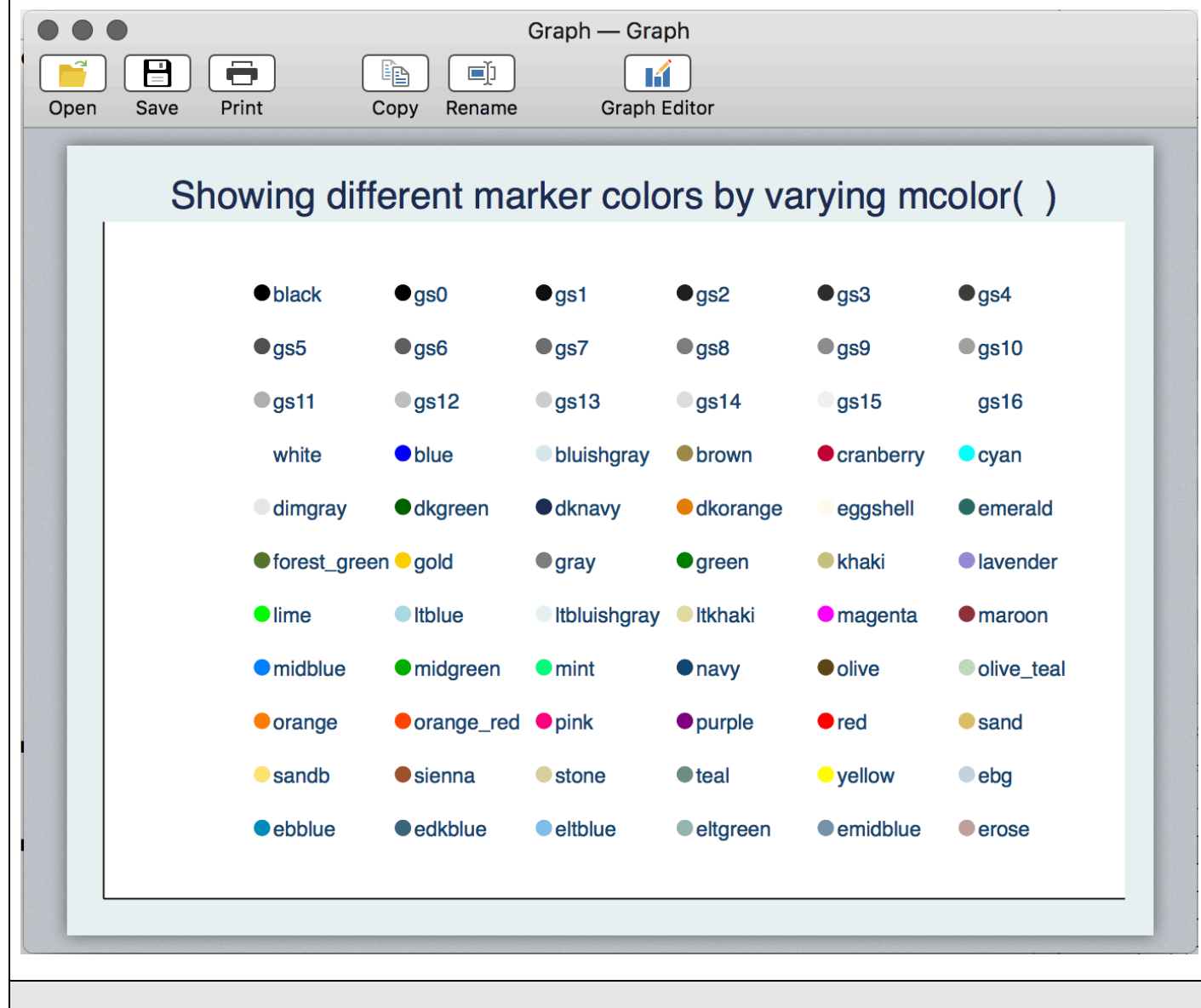
The screenshot shows a Stata Graph window titled "Graph — Graph". The toolbar contains icons for Open, Save, Print, Copy, Rename, and Graph Editor. The main plot area is titled "Showing different symbol sizes by varying msize()". It displays 12 different marker sizes arranged in a 3x4 grid. The markers are labeled as follows:

Row	Column 1	Column 2	Column 3	Column 4
1	vtiny	tiny	vsmall	small
2	medsmall	medium	medlarge	large
3	large	huge	vhuge	ehuge

(d) Choose Your Plotting Symbol Color using `mcolor()`

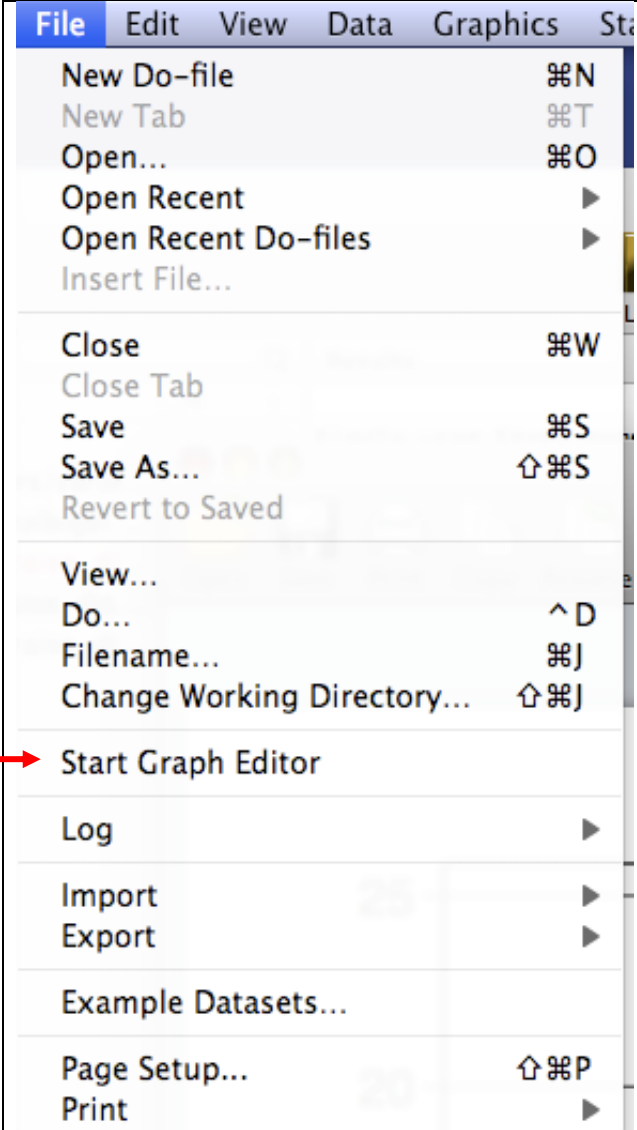
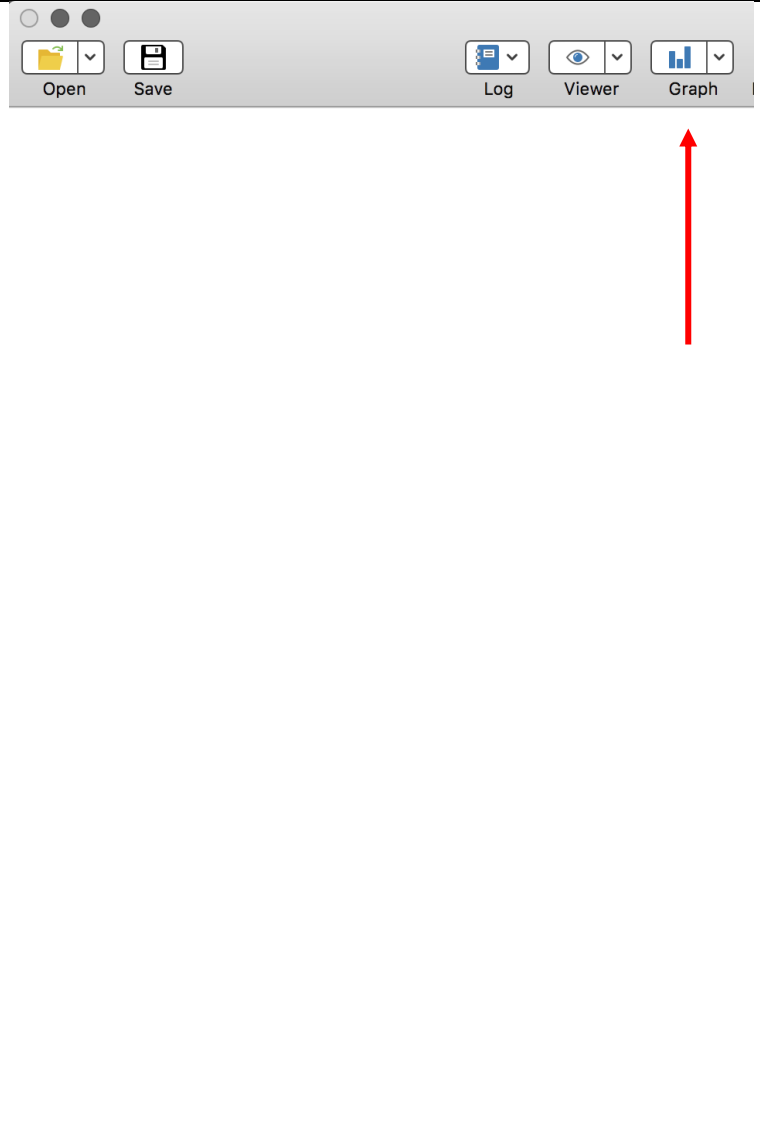
`mcolor()` Choices

.* Stata will show you the plotting colors with the command: `showmarkers, over(mcolor)`
 . `showmarkers, over(mcolor)`








2.3 Use the Graph Editor to Change How Your Graph Looks

There are 2 ways to launch the graph editor

Method #1 - From the main menu bar:	Method #2 - From the Graph Editor Icon in the Graph Itself
 <p>The screenshot shows the 'File' menu in Stata. The menu items are: New Do-file (⌘N), New Tab (⌘T), Open... (⌘O), Open Recent, Open Recent Do-files, Insert File..., Close (⌘W), Close Tab, Save (⌘S), Save As... (⇧⌘S), Revert to Saved, View..., Do... (^D), Filename... (⌘J), Change Working Directory... (⇧⌘J), Start Graph Editor (highlighted with a red arrow), Log, Import, Export, Example Datasets..., Page Setup... (⇧⌘P), and Print.</p>	 <p>The screenshot shows the Stata Graph Editor toolbar. It includes icons for Open, Save, Log, Viewer, and Graph. The 'Graph' icon, which is a bar chart, is highlighted with a red arrow.</p>

Key to Graph Editor Commands and Icons

Located at lower left

	Pointer Tool	Use this to select, drag, or modify the properties of an object. eg – Select your title. Then, holding the left mouse button, drag it to another position on the graph
	Add Text Tool	<u>How to:</u> (1) Select the “add text tool” (2) Click on the spot in your graph where you want to add text (3) A dialog box will appear (4) Type in your text. (5) If need be, use the pointer tool again to move your text to a better location.
	Add Line Tool	<u>How to:</u> (1) Select the “add line tool” (2) Click on the spot in your graph where you want the line to start (3) Holding the left mouse button, drag the line to where you want it to end. (4) Release the mouse.
	Add Marker Tool	Use this to add markers. The “how to” is similar to those for the “add text” and “add line” tools.
	Grid Edit Tool	Stay away from this for now....

Located at right

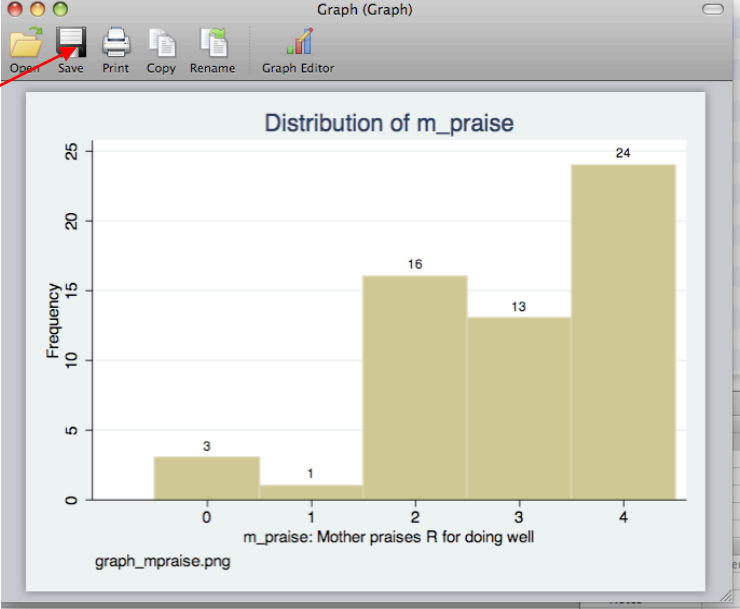
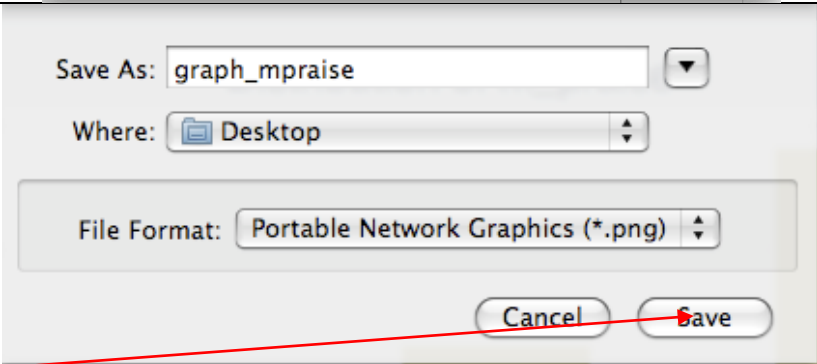
<ul style="list-style-type: none"> ▼ Graph <ul style="list-style-type: none"> ▶ plotregion1 ▶ yaxis1 ▶ xaxis1 ▶ legend ▶ positional titles note caption subtitle title 	This is a series of drop down menus from which you can modify the appearance of your plot region, titles, axes, etc.
---	---

Tip! Use Right-Click!

You can **right click** on any object in your graph. **Try it!** When you do a drop down menu appears. It contains some very handy options, typically: (1) **hide** (2) **show** (3) **lock** (4) **unlock**

2.4 How to Save Your Graph

Tip! Save your graph with the extension “.png”

<p>Step 1 – Click anywhere in the graph to make it active. Click on SAVE icon.</p>	
<p>Step 2 – (1) At SAVE AS: type graph name without the extension, (2) At WHERE: choose directory location, (3) At FILE FORMAT drop down menu, choose “portable network graphics (recommended). Click on SAVE icon</p>	
<p>Step 3 – SAVE</p>	

3. Discrete Variable Graphs

How to follow along:

This section utilizes a data set that Stata provides - **auto.dta**. You do NOT download it from the course website. Use the **sysuse** command to access:

```
. clear
. sysuse auto
```

3.1 Bar Chart Using Histogram, discrete

Graph Desired	Syntax in Stata
Bar chart	. histogram <i>variablename</i> , discrete ↑ note the comma

Use the command **histogram** with the option **discrete** to obtain a bar graph summary of the distribution of a discrete variable.

Beware! Do not use the Stata command **graph bar to obtain a bar graph summary of the distribution of a discrete distribution!** Stata does have a command called **graph bar**, but this does not produce a bar graph summary of a discrete variable distribution. Instead, it produces a bar graph plot of selected statistics of a continuous variable (usually the mean).

Basic Syntax for a Bar Graph Summary of Discrete Data

<p>. histogram <i>discretevariable</i>, discrete <i>graph_options</i></p> <p><u>Graph options:</u> <u>See also the graph options on page 15</u></p> <p>frequency - bar height will be equal to frequency fraction - bar height is proportional to percent; thus sum of bar heights will be 1 percent - bar height is proportional to percent; here, the sum of bar heights will be 100</p> <p>addlabels - display value of bar height; this will be either the frequency, fraction, or percent, depending on what you have chosen</p> <p>gap(<i>percent</i>) - This instructs Stata to reduce the bar width by the specified percent. It makes the bar graph look better</p> <p>xlabel (<i>value "label" value "label"</i>) - This instructs Stata to label the x-axis values with their value code labels. It makes the bar graph look better</p>
--

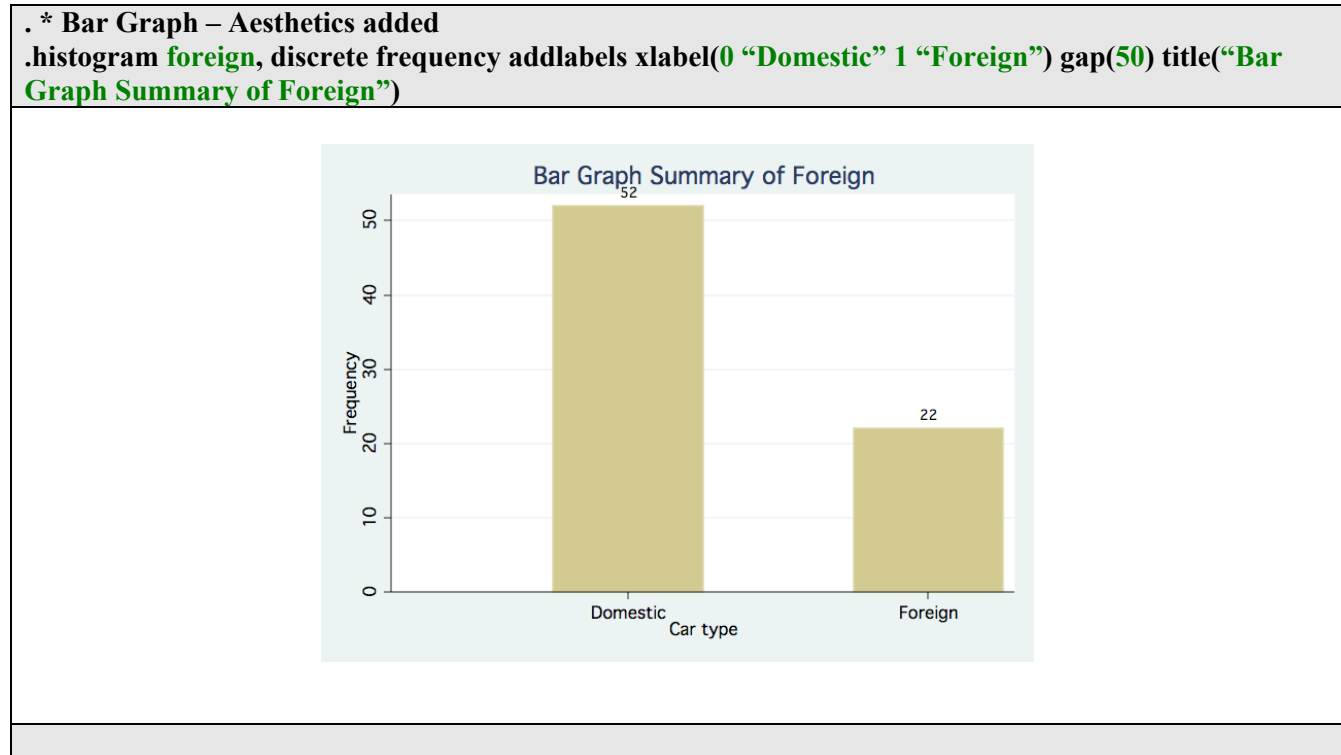
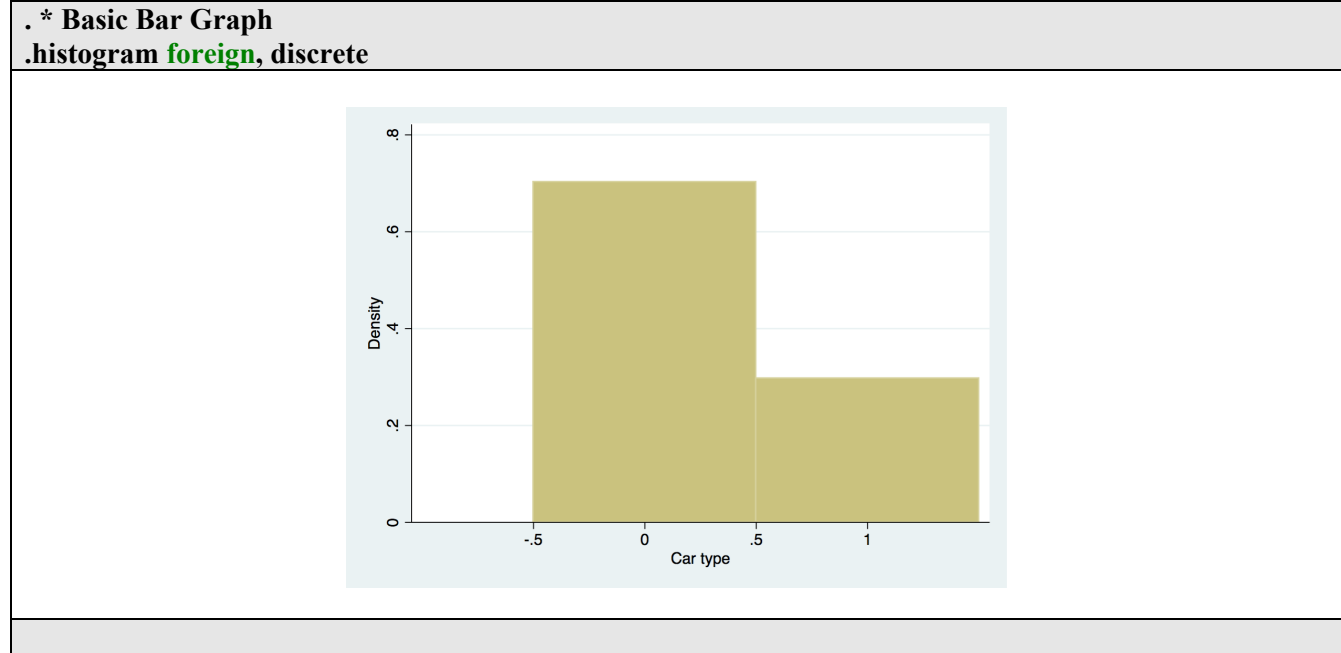
Tip!

Use the option **gap()** to obtain nice spacing of the bars.

Tip!

Use the option **xlabel()** to obtain a nice labeling of the horizontal axis

Example



3.2 Grouped Bar Graph Using histogram, discrete

Graph Desired	Syntax in Stata
Side-by-side or grouped bar graphs, for levels of some grouping variable	<pre>. sort <i>groupingvar</i> . histogram <i>variablename</i>, discrete by(<i>groupingvar</i>)</pre> <p style="text-align: center;">↑ note the comma</p>

You can obtain side by side bar graph summaries for grouped data by using the command **histogram** with the option **discrete** together with the option **by()** to obtain a bar graph summary of the distribution of a discrete variable.

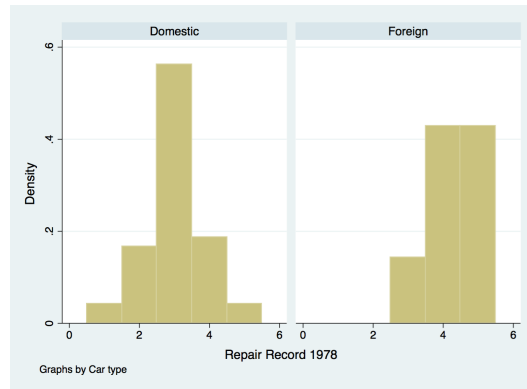
Tip – You must sort the data first by your grouping variable.

Basic Syntax for a Grouped Bar Graph Summary of Discrete Data

<pre>. sort <i>groupingvariable</i> . histogram <i>discretevariable</i>, discrete by(<i>groupingvariable</i>) <i>graph_options</i></pre> <p><u>Graph options for a histogram:</u></p> <p>frequency - bar height will be equal to frequency Tip! This is not a good choice for grouped data fraction - bar height is proportional to percent; thus sum of bar heights will be 1 percent - bar height is proportional to percent; here, the sum of bar heights will be 100</p> <p>addlabels - display value of bar height; this will be either the frequency, fraction, or percent, depending on what you have chosen</p> <p>gap(<i>percent</i>) - This instructs Stata to reduce the bar width by the specified percent.</p> <p>xlabel (<i>value "label" value "label"</i>) - This instructs Stata to label the x-axis values with their value code labels.</p> <p><u>The general graph options introduced on page 15 work here too.</u></p> <p>title(<i>"title in quotes"</i>) - specify title subtitle(<i>"subtitle in quotes"</i>) - specify subtitle ytitle(<i>"Y-axis title in quotes"</i>) - specify Y-axis title xtitle(<i>"X-axis title in quotes"</i>) - specify X-axis title legend (<i>"legend in quotes"</i>) - specify legend caption(<i>"caption in quotes"</i>) - specify caption note(<i>"note in quotes"</i>) - specify note</p>
--

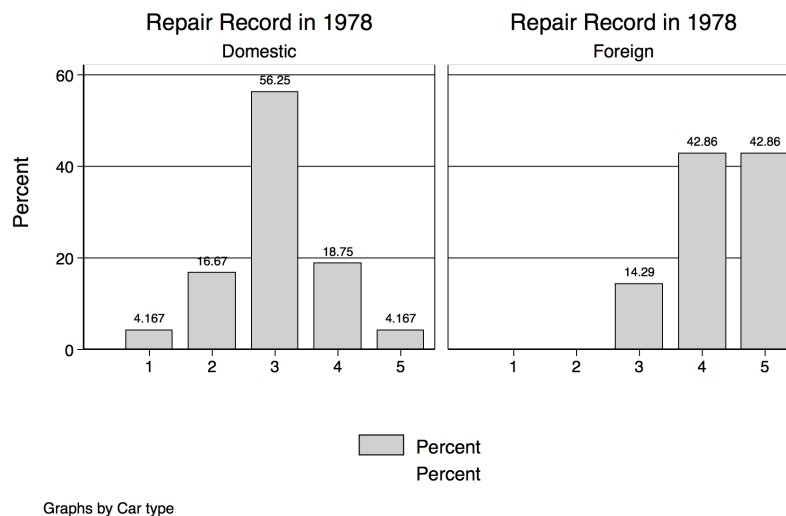
Example

```
. * Basic Bar Graph, by(group)
. sort foreign
. histogram rep78, discrete by(foreign)
```



Tip – Keep your title short as it will appear twice.

```
. * Bar Graph, by(group) – Aesthetics added
. sort foreign
. set scheme lean1
. histogram rep78, discrete by(foreign) percent addlabels xlabel(1 “1” 2 “2” 3 “3” 4 “4” 5 “5”) gap(25)
title(“Repair Record in 1978”) xtitle(“ ”)
```




3.3 Grouped Bar Graph Using catplot

How to follow along:

- 1) Download from the course website the data set [descriptive_gss_small.dta](#).
- 2) clear
- 3) From whatever directory is appropriate for you, open descriptive_gss_small.dta

Preliminary:

Type `help catplot`. If you get an error message saying “help for catplot not found”, this means that you do not have this package of commands. Follow the directions for locating it (findit catplot) and then downloading it to your computer.

Graph Desired	Syntax in Stata
Side-by-side or grouped bar graphs, for levels of some grouping variable	<code>. catplot <i>variablename groupingvar</i>, recast(bar)</code> <div style="text-align: right;">  note the comma </div>

Basic Syntax for a Grouped Bar Graph Using CATPLOT

Tip! -

Check out the help menu for this by entering in the command line:
`.help catplot`

- `. catplot variablename groupingvar`
- `. catplot variablename groupingvar, recast(bar)`
- `. catplot variablename groupingvar, recast(dot)`

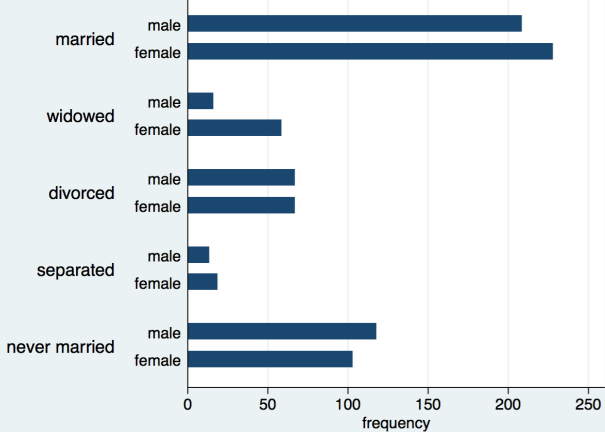
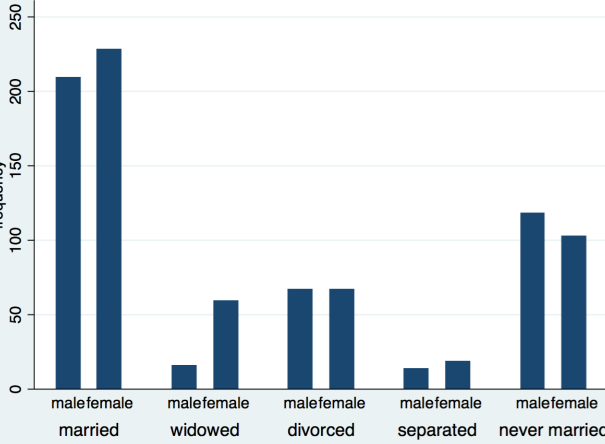
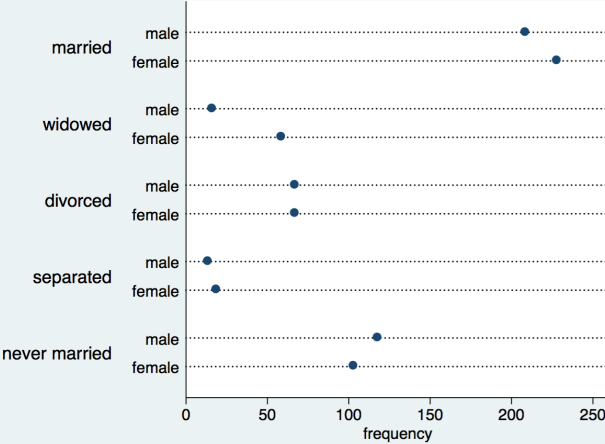
Graph options for catplot

- fraction** - The y-axis will be relative frequencies that total 1.
- percent** - The y-axis will be relative frequencies expressed as percents that total 100.

The general graph options introduced on page 15 work here too.

- `title("title in quotes")` - specify title
- `subtitle("subtitle in quotes")` - specify subtitle
- `ytitle("Y-axis title in quotes")` - specify Y-axis title
- `xtitle("X-axis title in quotes")` - specify X-axis title
- `legend ("legend in quotes")` - specify legend
- `caption("caption in quotes")` - specify caption
- `note("note in quotes")` - specify note

Examples –

<p>. catplot sex marital</p>	
<p>. catplot sex marital, recast(bar)</p>	
<p>. catplot sex marital, recast(dot)</p>	

3.4 Stratified Bar Graph Using **graph combine**

Dear class: A note on terminology –

A “grouped” bar graph is also called a “stratified” bar graph. What we are after is a bar graph for each of a defined set of subgroups. The variable that defines the subgroups might be called the **grouping variable** or it might be called the **stratifying variable**.

How to follow along:

- 1) Download from the course website the data set **depress.dta**.
- 2) clear
- 3) From whatever directory is appropriate for you, open depress.dta

What is **Graph Combine**

It is a nifty (and often, better looking) way to combine several graphs into one final “product”.

How **Graph Combine** Works

Step 1

Create, separately, the individual graphs that will be combined later.
Take care to

- (1) Set the y-axis tick marks to be the same in each of the individual graphs.
- (2) Name each individual graph using the name command: **name(yourchoice, replace)**

Step 2

To combine the graphs, type into the command window

.graph combine *yourchoice1 yourchoice2* etc

Step 3

Save your **combined graph** in its own graph, following the instructions on page 22.

Example - Basic Graph Combine

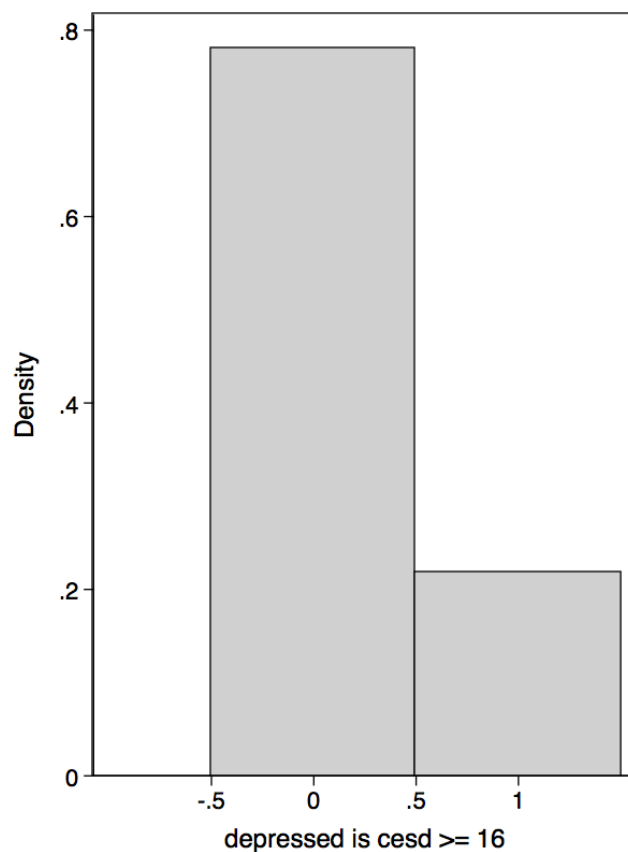
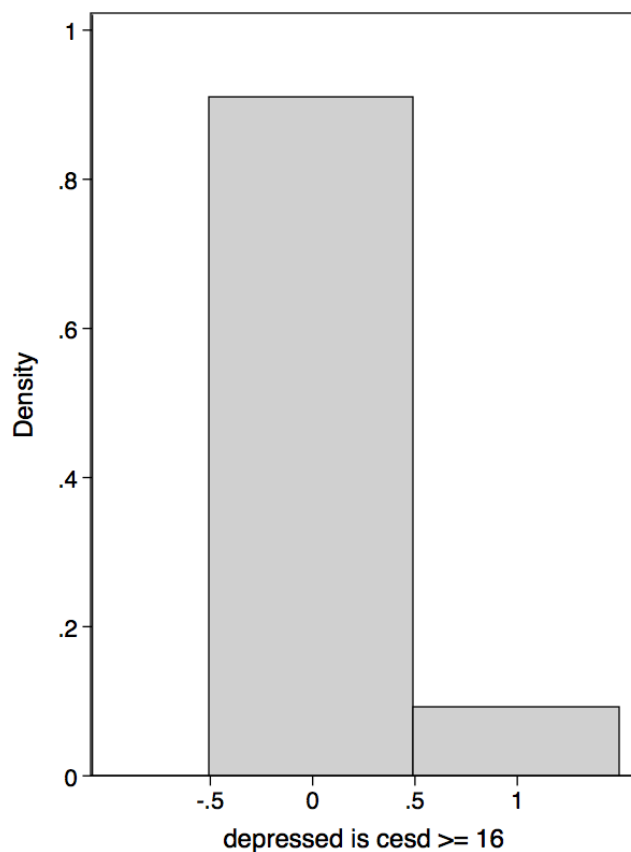
Important! Note that graph produced has the Undesirable feature of different y-axis tick marks. NOT GOOD, EVER!!!

```
. sort sex
. set scheme lean1

.* Create 1st graph. Use name( ) option and give it the name males
. histogram cases if sex==1, discrete name(males, replace)

.* Create 2nd graph. Use name( ) option and give it the name females
. histogram cases if sex==2, discrete name(females, replace)

.* Use command graph combine to combine the two graphs (males and females) into one single graph
.graph combine males females
```



Example – Graph Combine with Aesthetics

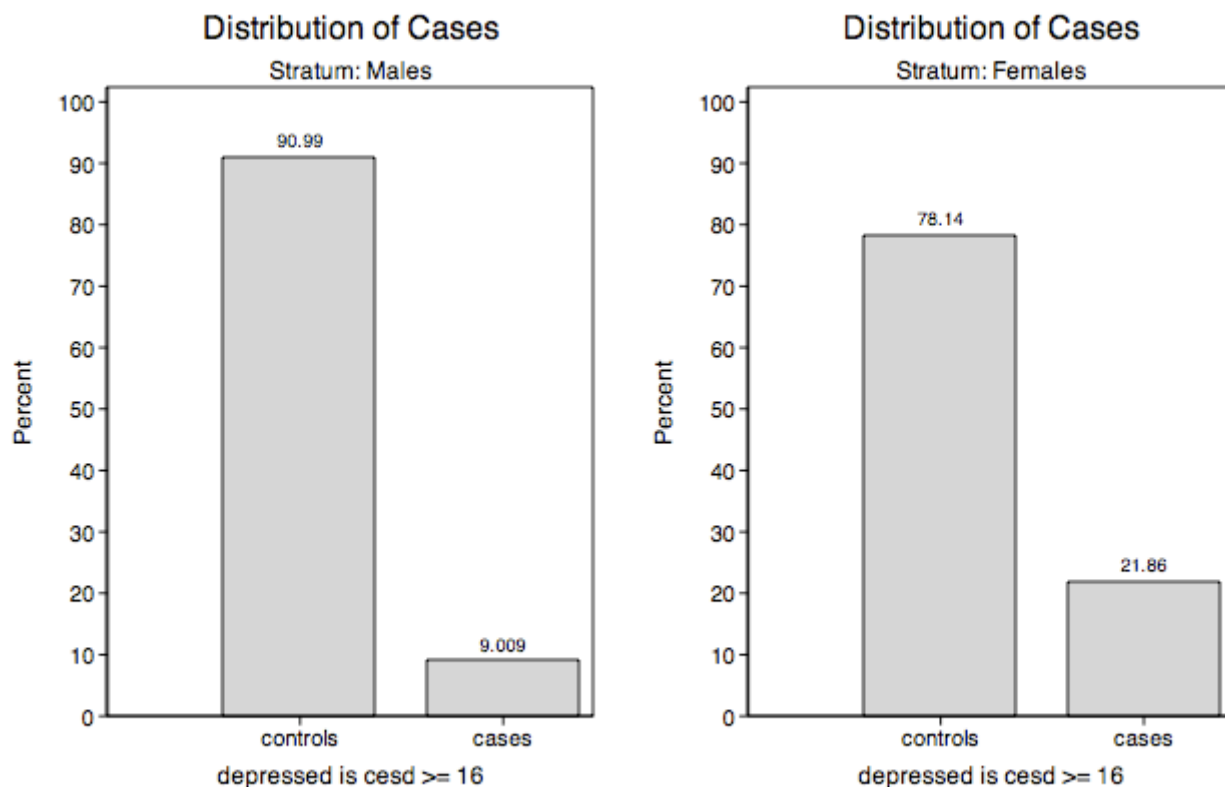
Better – Notice here how the y-axis specifications are the same!

```
. sort sex
. set scheme lean1

.* Create 1st graph. Use name() option and give it the name males
. histogram cases if sex==1, discrete percent gap(25) addlabels title("Distribution of Cases")
  subtitle("Stratum: Males") ylabel(0 (10)100) xlabel(0 "controls" 1 "cases") name(males, replace)

.* Create 2nd graph. Use name() option and give it the name females
. histogram cases if sex==2, discrete percent gap(25) addlabels title("Distribution of Cases")
  subtitle("Stratum: Females") ylabel(0 (10)100) xlabel(0 "controls" 1 "cases") name(females, replace)

.* Use command graph combine to combine the two graphs (males and females) into one single graph
. graph combine males females
```



4. Continuous Variable Graphs

How to follow along:

This section utilizes the Stata system dataset, **auto.dta**. Again, use the **sysuse** command to access:

```
. clear
. sysuse auto
```

4.1 Dot Plot Using **dotplot**

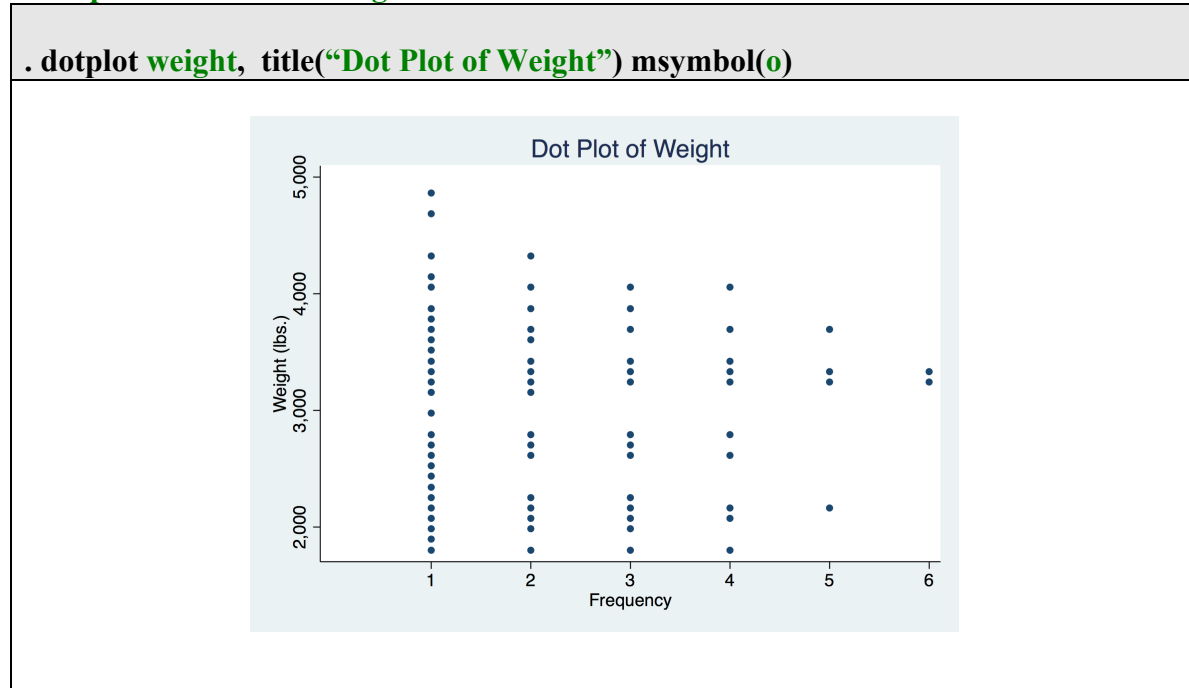
Graph Desired	Syntax in Stata
Dot plot of raw data points	. dotplot <i>variablename</i>
Side-by-side dot plot of raw data points for subgroups of the data sorted.	. sort <i>groupingvar</i> . dotplot <i>variablename</i> , over(<i>groupingvar</i>)

Use **dotplot** to obtain a quick graphical summary of the distribution of values for a variable. You can also use **dotplot** to compare the distribution of values for some variable across groups.

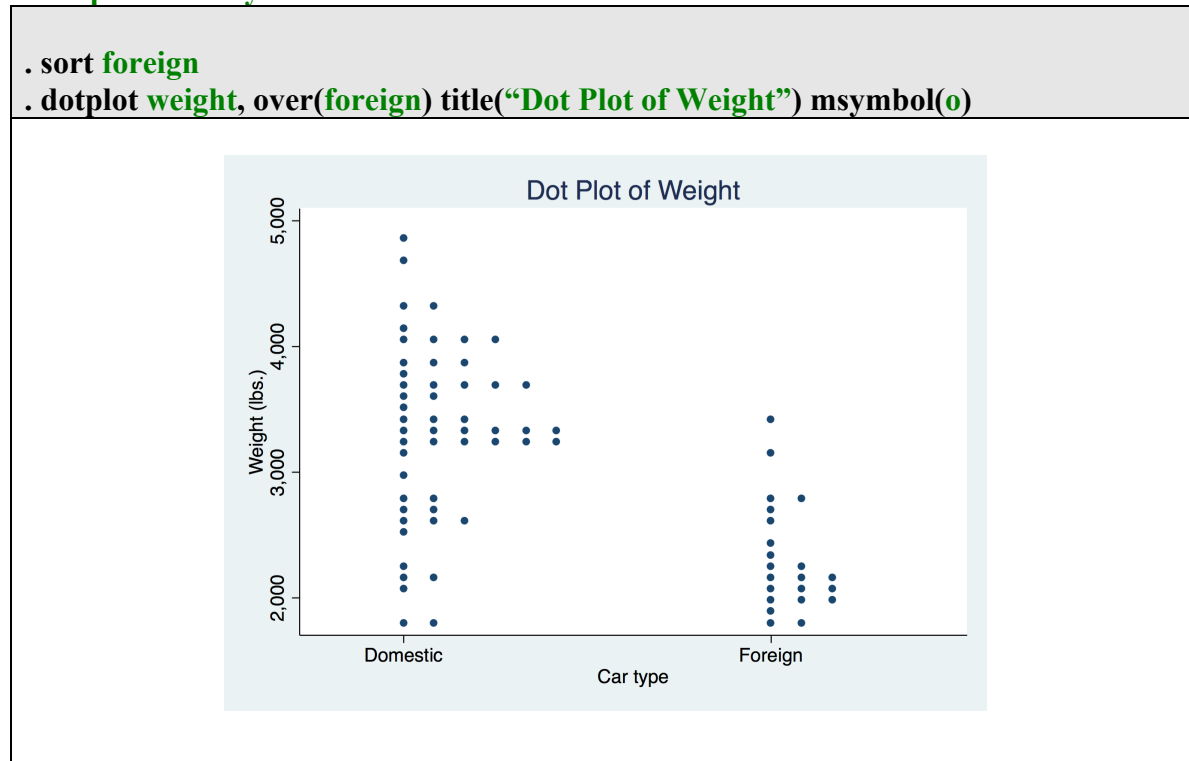
Basic Syntax for a Dot Plot

. dotplot <i>continuousvariable</i> , <i>graph_options</i>
<u>Graph options for dotplot:</u>
over (<i>groupingvariable</i>) - this will produce side by side dot plot for grouped data
nx (#) - sets density of dots. Default is nx(0)
ny (#) - set the number of bins
nogroup - use the actual value of the variable instead of bins
center - center the dots for each column
<u>The general graph options introduced on page 15 work here too.</u>
title ("title in quotes") - specify title
subtitle ("subtitle in quotes") - specify subtitle
ytitle ("Y-axis title in quotes") - specify Y-axis title
xtitle ("X-axis title in quotes") - specify X-axis title
legend ("legend in quotes") - specify legend
caption ("caption in quotes") - specify caption
note ("note in quotes") - specify note

Example - Dot Plot for Single Variable



Example - Side-by-Side Dot Plot



4.2 Stem and Leaf Plot Using **stem** or **stemplot**

Note - The command `stemplot` may not be in your installation. This is another nifty routine developed by a user and that, for you, requires downloading. Issue the command `findit stemplot`) and then follow the instructions to download it to your computer.

Graph Desired	Syntax in Stata
Single group: Stem and Leaf plot for distribution of values of some variable	<code>. stem <i>variablename</i></code>
More than 1 group: Side-by-side stem and leaf plot for distribution of values of some variable separately for groups defined by <code>groupingvar</code> .	<code>. sort <i>groupingvar</i></code> <code>. stemplot <i>variablename</i>, over(<i>groupingvar</i>)</code>

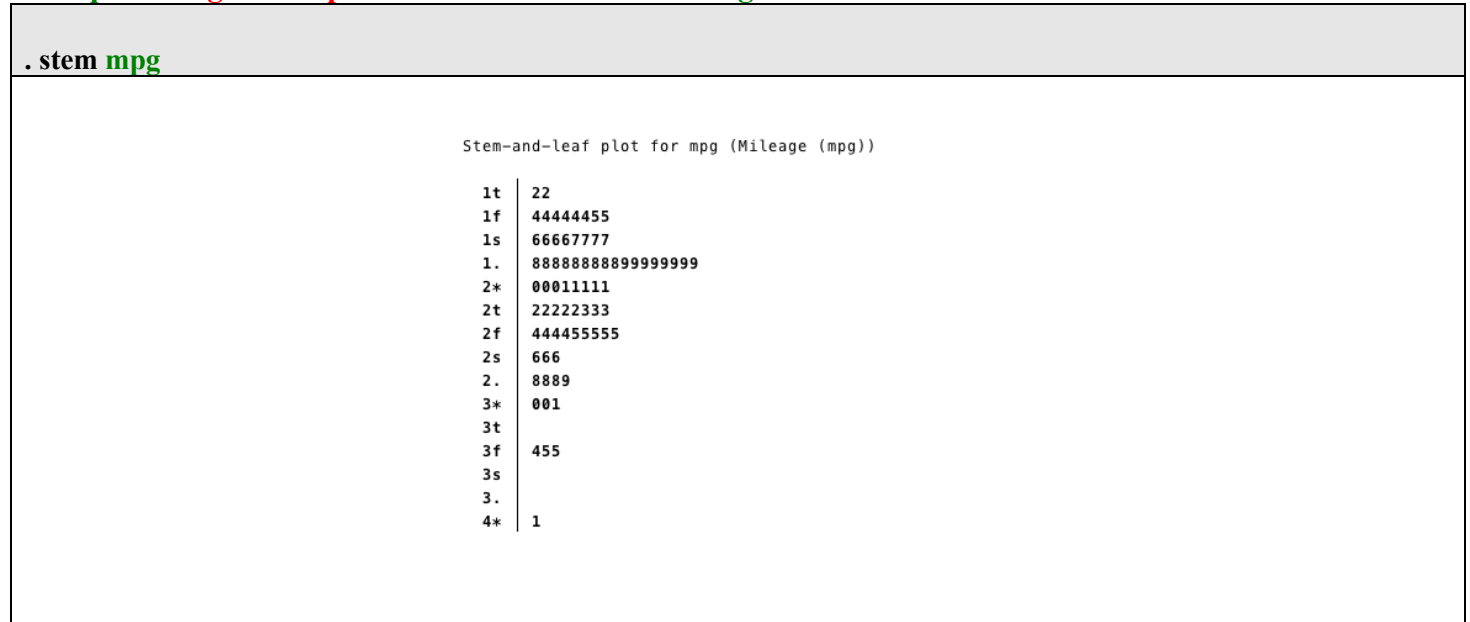
Basic Syntax for **stem**

<code>. stem <i>continuousvariable</i>, <i>graph_options</i></code>
<u>Graph options:</u> <u>Note – This command does not allow certain options such as <code>title()</code></u>
<code>lines(#)</code> - number of stems per interval of 10 digits <code>digits(#)</code> - digits per leaf; the default is 1

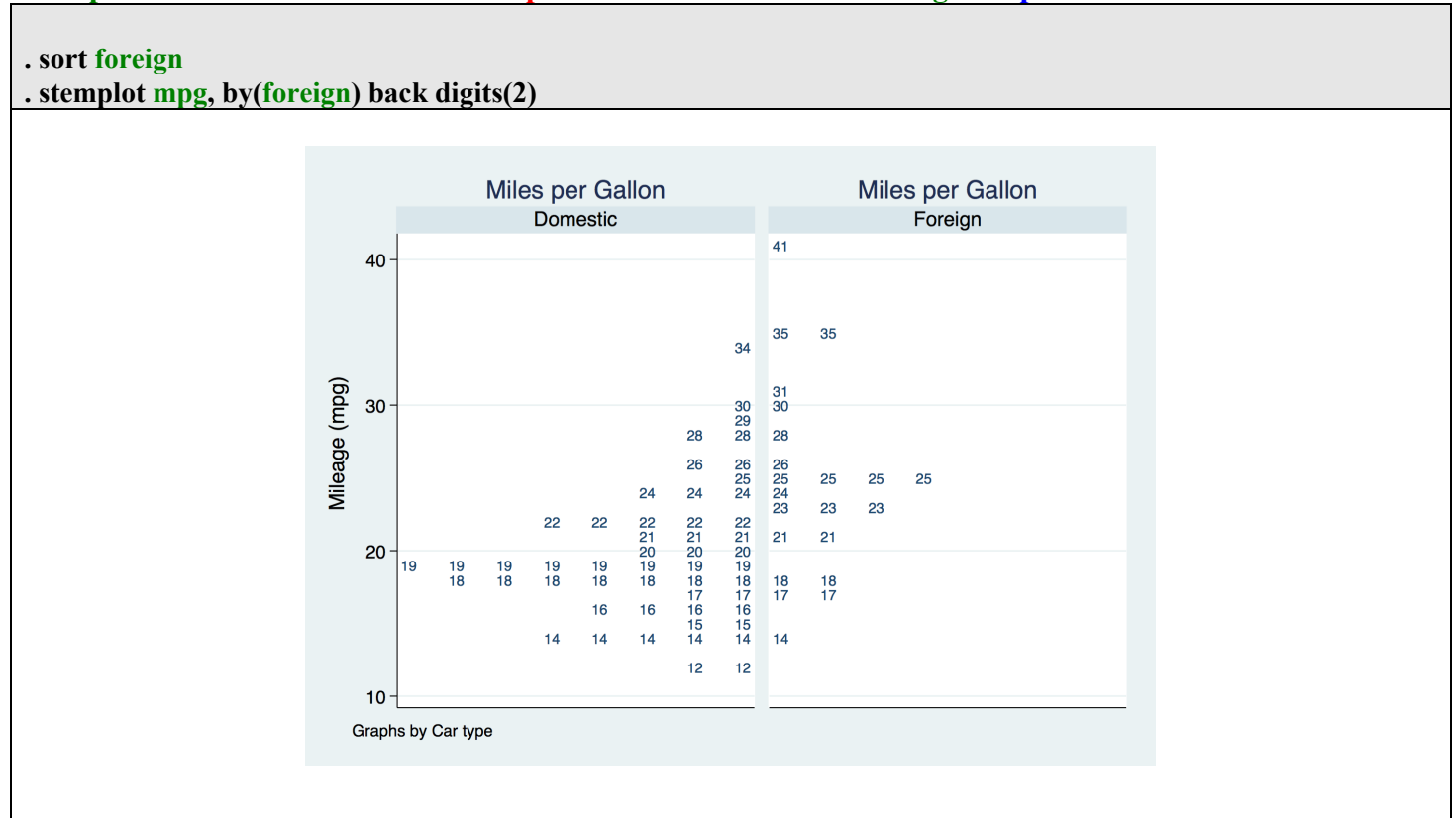
Basic Syntax for **stemplot**

<code>. sort <i>groupingvar</i></code> <code>. stemplot <i>continuousvariable</i>, by(<i>groupingvar</i>) <i>graph_options</i></code>
<u>Graph options for <code>stemplot</code> (but not <code>stem</code>!!):</u> <code>back</code> - produce a “back-to-back” stem and leaf (recommended!) <code>lines(#)</code> - number of stems per interval of 10 digits <code>digits(#)</code> - digits per leaf; the default is 1
<u>The general graph options introduced on page 15 work here too.</u> <code>title(“<i>title in quotes</i>”)</code> - specify title <code>subtitle(“<i>subtitle in quotes</i>”)</code> - specify subtitle <code>ytitle(“<i>Y-axis title in quotes</i>”)</code> - specify Y-axis title <code>xtitle(“<i>X-axis title in quotes</i>”)</code> - specify X-axis title <code>legend(“<i>legend in quotes</i>”)</code> - specify legend <code>caption(“<i>caption in quotes</i>”)</code> - specify caption <code>note(“<i>note in quotes</i>”)</code> - specify note

Example – Single Group Stem and Leaf Dot Plot using `stem`



Example – "Back-to-Back" Two Group Stem and Leaf Dot Plot using `stemplot`



Tip – I used the option `digits(2)` so that the plot would show the actual mileages! How fun is that....

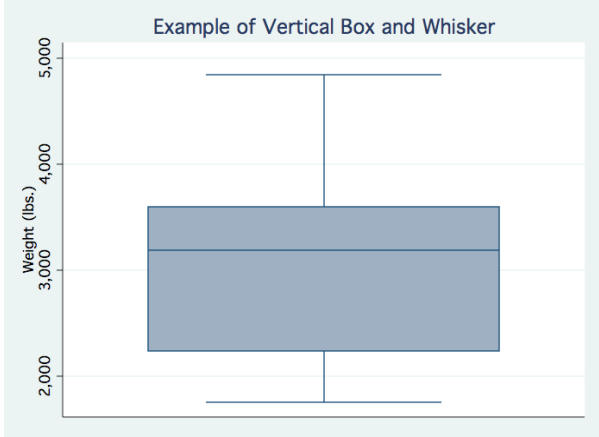
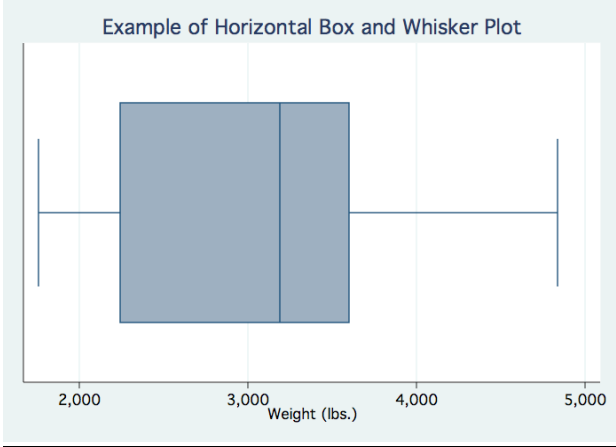
4.3 Box and Whisker Plot Using Graph Box

Graph Desired	Syntax in Stata
Vertical box plot Horizontal box plot	. graph box <i>variablename</i> . graph hbox <i>variablename</i>
Side-by-side vertical box plot Side-by-side horizontal box plot.	. sort <i>groupingvar</i> . graph box <i>variablename</i> , over(<i>groupingvar</i>) . graph hbox <i>variablename</i> , over(<i>groupingvar</i>)

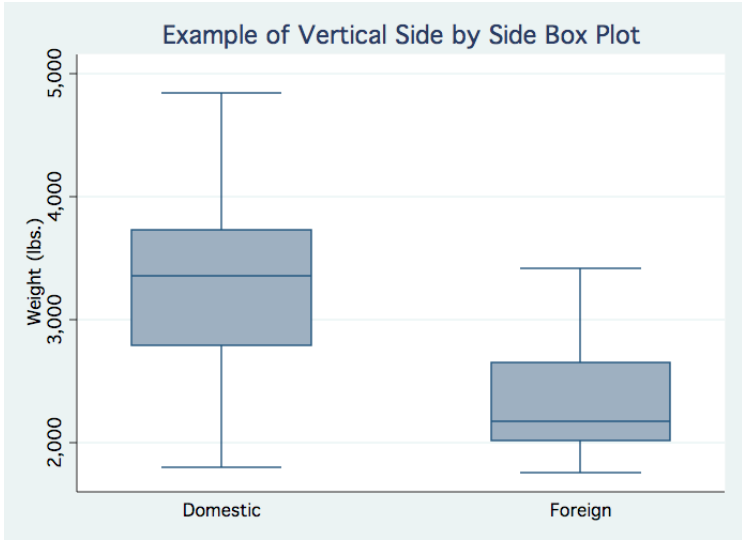
Basic Syntax for a Box and Whisker Plot

<p><u>Vertical</u></p> <p>. graph box <i>continuousvariable</i>, <i>graph_options</i></p> <p><u>Horizontal</u></p> <p>. graph hbox <i>continuousvariable</i>, <i>graph_options</i></p> <p><u>Graph options for box and hbox:</u></p> <p><i>over(groupingvariable)</i> - this will produce side by side dot plot for grouped data</p> <p><i>over(groupingvariable, sort(1))</i> - this will produce side by side dot plot for grouped data and the boxes will be in order of the medians, smallest first</p> <p><i>over(groupingvariable, sort(1) descending)</i> - this will produce side by side dot plot for grouped data and the boxes will be in order of the medians, largest first</p> <p><u>The general graph options introduced on page 15 work here too.</u></p> <p><i>title("title in quotes")</i> - specify title</p> <p><i>subtitle("subtitle in quotes")</i> - specify subtitle</p> <p><i>ytitle("Y-axis title in quotes")</i> - specify Y-axis title</p> <p><i>xtitle("X-axis title in quotes")</i> - specify X-axis title</p> <p><i>legend("legend in quotes")</i> - specify legend</p> <p><i>caption("caption in quotes")</i> - specify caption</p> <p><i>note("note in quotes")</i> - specify note</p>

Example – Basic Box Plot

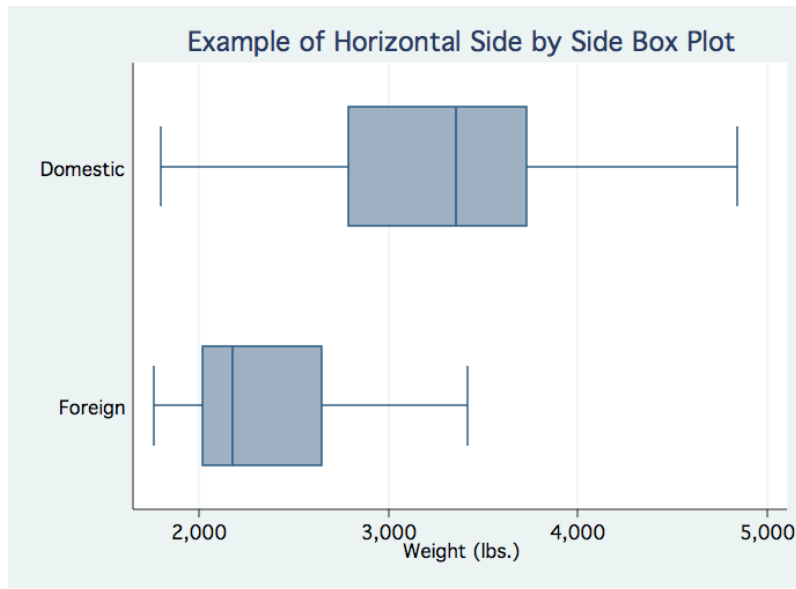
Vertical	Horizontal
<pre>. graph box weight, title("Example of Vertical Box and Whisker")</pre>	<pre>. graph hbox weight, title("Example of Horizontal Box and Whisker")</pre>
	

Example - Side-by-Side Vertical Box and Whisker Plots

<pre>. sort foreign . graph box weight, over(foreign) title("Example of Vertical Side by Side Box Plot")</pre>


Example - Side-by-Side Horizontal Box and Whisker Plots

```
. sort foreign
. graph hbox weight, over(foreign) title("Example of Horizontal Side by Side Box Plot")
```



4.4 Histogram Using Histogram

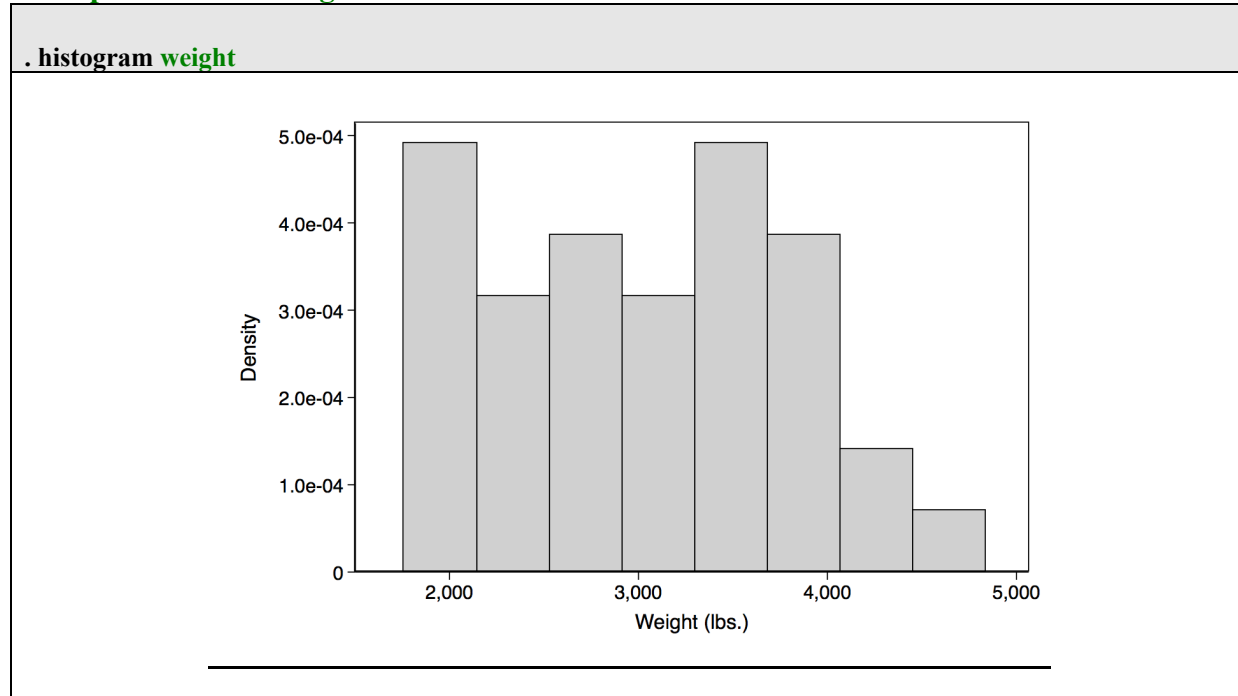
Graph Desired	Syntax in Stata
Histogram	<code>. histogram <i>variablename</i></code>

Tip! Set your own bins explicitly.

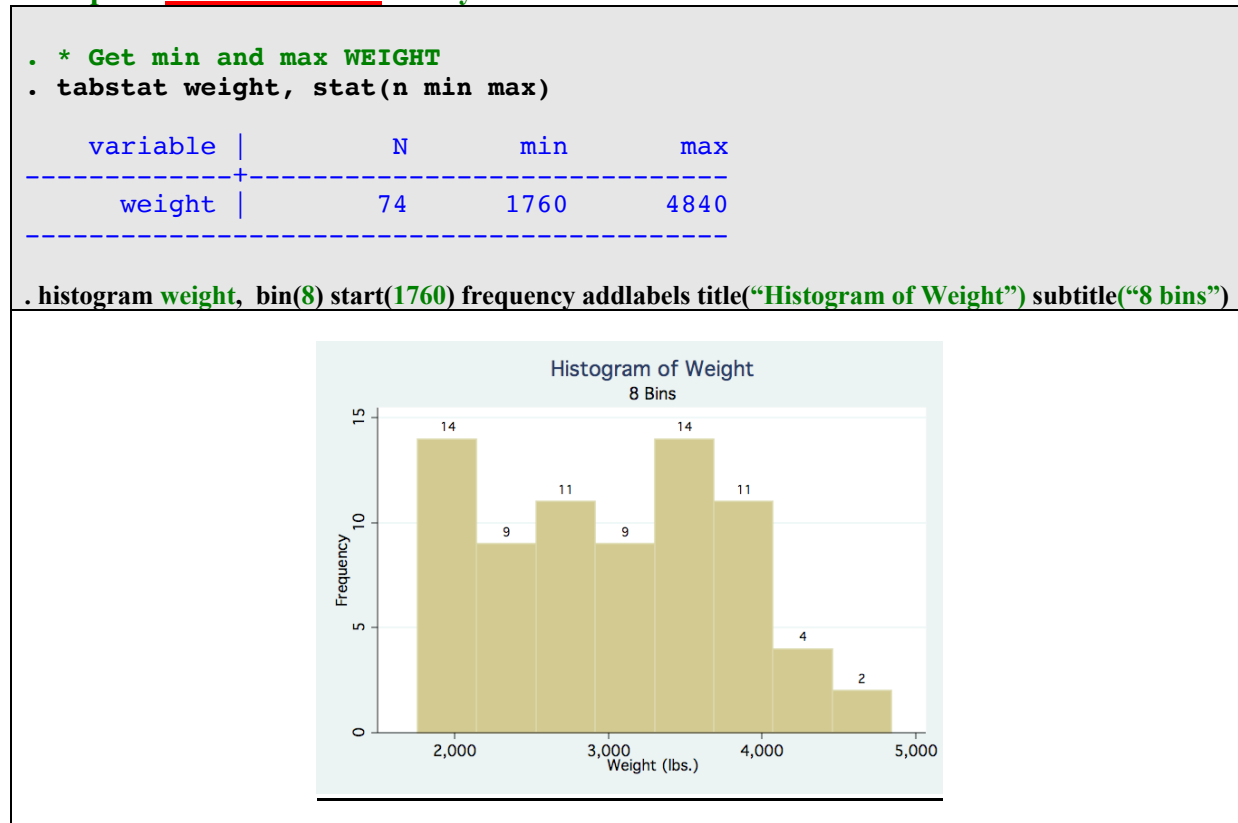
Basic Syntax for a Histogram

<p><code>. histogram <i>continuousvariable</i>, <i>graph_options</i></code></p> <p><u>Graph options for histogram:</u></p> <ul style="list-style-type: none"> <code>start(#)</code> - set value of lower limit of first bin <code>width(#)</code> - width of each bin <code>bin(#)</code> - set number of bins <code>frequency</code> - plot frequencies <code>fraction</code> - plot fractions <code>addlabels</code> - display value of bar height <code>normal</code> - display overlaying normal distribution <p><u>The general graph options introduced on page 15 work here too.</u></p> <ul style="list-style-type: none"> <code>title("title in quotes")</code> - specify title <code>subtitle("subtitle in quotes")</code> - specify subtitle <code>ytitle("Y-axis title in quotes")</code> - specify Y-axis title <code>xtitle("X-axis title in quotes")</code> - specify X-axis title <code>legend ("legend in quotes")</code> - specify legend <code>caption("caption in quotes")</code> - specify caption <code>note("note in quotes")</code> - specify note

Example - Basic Histogram

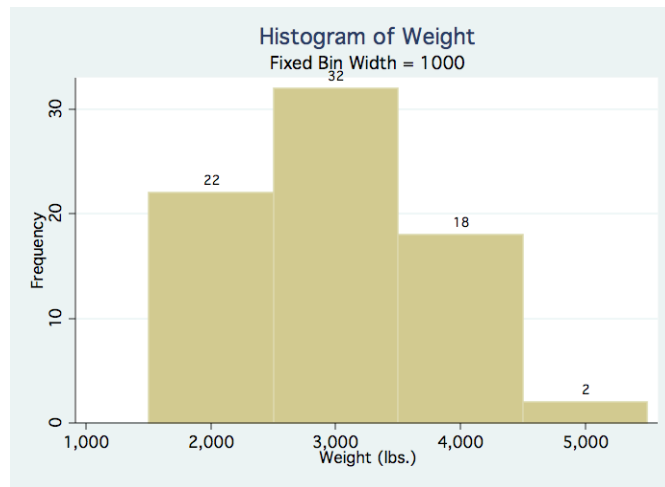


Example - Number of Bins Set by user



Example - Width of Bins Set by User

```
. histogram weight, width(1000) start(1500) frequency addlabels title("Histogram of Weight") subtitle("Fixed Bin Width = 1000")
```



(Fancy!!) Example – Tick Marks at Units of Standard Deviation & Overlay Normal

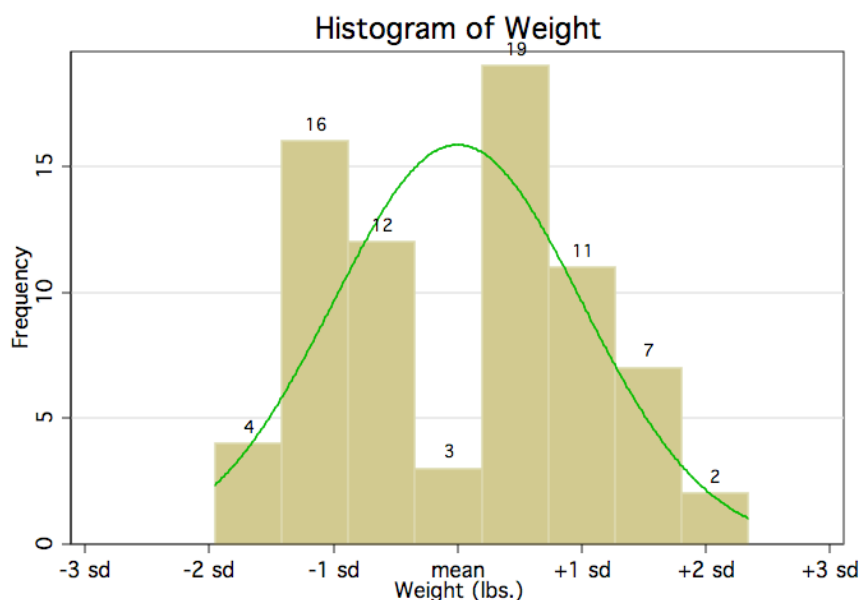
```
. * Preliminary: Obtain mean and sd
. tabstat weight,stat(mean sd)

      variable |      mean      sd
-----+-----
      weight |  3019.459  777.1936

. *Identify tick marks at multiples of sd
. display 3019.459- (1*777.1936)
2242.2654
. display 3019.459- (2*777.1936)
1465.0718
. display 3019.459- (3*777.1936)
687.8782
. display 3019.459+ (1*777.1936)
3796.6526
. display 3019.459+ (2*777.1936)
4573.8462
. display 3019.459+ (3*777.1936)
5351.0398

. set scheme slcolor

. histogram weight, start(1500) bin(8) frequency addlabels normal
ylabel(0(5)15, grid) xlabel(3019.459 "mean" 2242.2654 "-1 sd" 1465.0718 "-2
sd" 687.8782 "-3 sd" 3796.6526 "+1 sd" 4573.8462 "+2 sd" 5351.0398 "+3 sd")
title("Histogram of Weight")
```



4.5 Scatterplot Using Graph Twoway

Basic Syntax for Scatterplot

```
. graph twoway (scatter yvar xvar, plot_options) (plot_choice, plot_options), graph_options
```

Don't forget this comma

It is possible to have multiple plots in one graph
See examples below...

Plot choice:

Partial listing ...

- scatter** - simple x-y scatter plot
- lfit** - linear regression fit
- lfitci** - linear regression fit with 95% confidence band

recommended for repeated measures data only:

- line** - draws a line through all the points **Tip!** Always use with option **sort**
- connected** - draws a line with points shown. **Tip!** Always use with option **sort**

Plot options:

Partial listing ...

- msymbol()** - set choice of plotting symbol (see chart on page 17)
- mcolor()** - set choice of color of plotting symbol. (see chart on page 19)
- xlabel()** - specify labeling of tick marks on the X-axis
- ylabel()** - specify labeling of tick marks on the Y-axis

Tip! For **xlabel()** and **ylabel()**. Use either
xlabel(min(increment)max) or
xlabel(value "label" value "label")

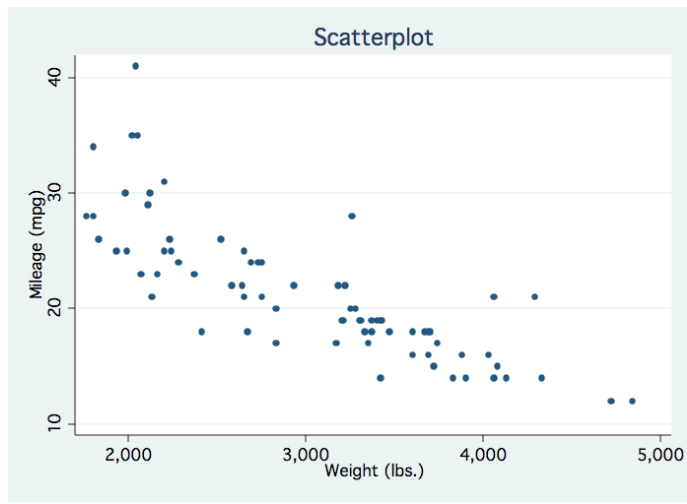
If you want grid lines, add the option grid. Don't forget the comma!
xlabel(min(increment)max, grid)
xlabel(value "label" value "label", grid)

The general graph options introduced on page 15 work here too.

- title("title in quotes")** - specify title
- subtitle("subtitle in quotes")** - specify subtitle
- ytitle("Y-axis title in quotes")** - specify Y-axis title
- xtitle("X-axis title in quotes")** - specify X-axis title
- legend("legend in quotes")** - specify legend
- caption("caption in quotes")** - specify caption
- note("note in quotes")** - specify note

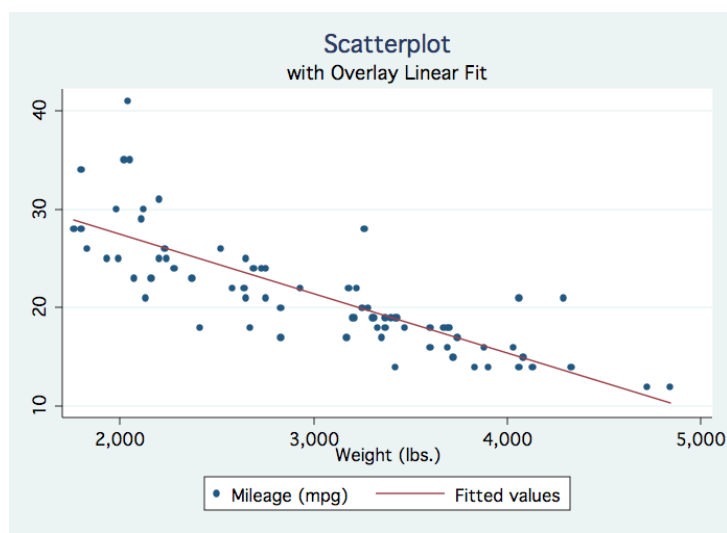
Example - Scatterplot

```
. graph twoway (scatter mpg weight, msymbol(o)), title("Scatterplot")
```



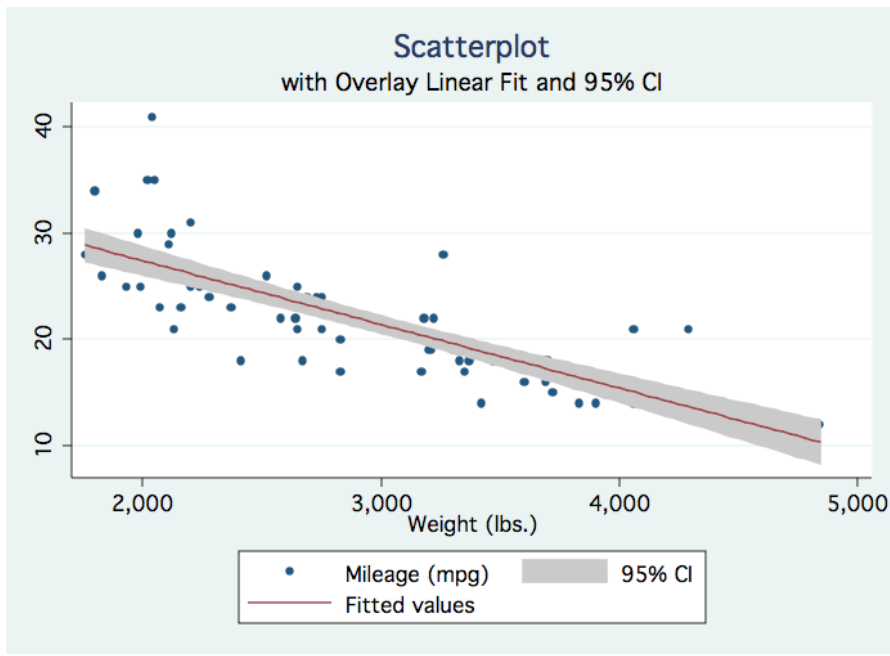
Example - Scatterplot with Linear Fit

```
. graph twoway (scatter mpg weight, msymbol(o)) (lfit mpg weight),  
title("Scatterplot") subtitle("with Overlay Linear Fit")
```



Example - Scatterplot with Linear Fit and 95% Confidence Band

```
. graph twoway (scatter mpg weight, msymbol(o)) (lfitci mpg weight),  
title("Scatterplot") subtitle("with Overlay Linear Fit and 95% CI")
```



4.6 Matrix Plot Using Graph Matrix

Graph Desired	Syntax in Stata
Pairwise X-Y scatter plot for all the variables in a data set (eg – a preliminary in regression)	<p>.* For full matrix . graph matrix <i>variable1 variable2 variable3</i></p> <p>.* For half matrix . graph matrix <i>variable1 variable2 variable3</i>, half</p>

Basic Syntax for a Matrix Plot

. graph matrix *variable1 variable2 variable3*, *graph_options*

Graph options:

See also the graph options on [pages 15-22](#)

half – show lower diagonal only **Note-** Because I find it easier to look at upper diagonal, I don't use this option.

by(*groupingvariable*) - repeat for values of grouping variable. **Tip!** Sort by grouping var first!

jitter(#) - Jitter the points for greater resolution **Tip!** Good for large sample sizes

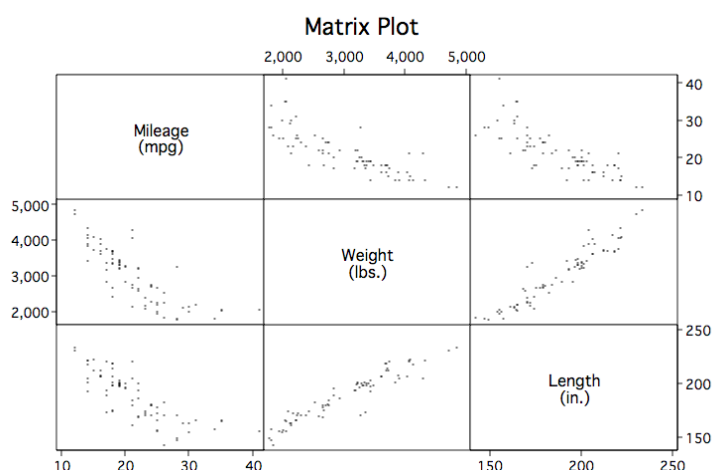
msize(*choice*) - set size of marker symbol.

Choices are: vtiny, tiny, small, medsmall, medium, medlarge, large
vlarge, huge, vhuge, and ehuge

Tip! Use option **msymbol(*p*)** when the sample size is large

Example -

```
. set scheme lean1
. graph matrix mpg weight length, msymbol(p) title("Matrix Plot")
```



5. Repeated Measures Graphs

Following along?

Download from the course website the following two data sets: [roy_long.dta](#) and [roy_wide.dta](#).

Repeated measurements data arise in many fields of study so that there are a variety of terminologies.

Definition -

Our definition of repeated measurements is measurements of the same characteristic on the same individual unit on more than one occasion (Crowder and Hand, p. 1)

Overview of Some Graphical Summaries

Side-by-side box and whisker plots -

While side-by-side box plots do not summarize the interrelationships among the repeated measures themselves, this plot is still useful. It provides a sense of equality of means and equality of variances.

Line plots or profile plots -

These give a feel for the raw data profiles and can reveal unusual individuals.

Mean response profiles, with or without associated standard error (SE) or confidence bands -

These give a feel for the functional form of the response profile. In settings where there are multiple groups, these also give a feel for group differences.

Matrix Scatterplot -

This gives you a visual feel for the autocorrelation structure.

See, Section 4.6 Matrix Plot using **graph matrix** (page 46)



5.1 Choose Your Data Layout Using **Reshape**

There are two kinds of data layouts, univariate (**long**) and multivariate (**wide**).

Example – (Source: Pottoff RR and Roy SN. 1964 *A generalized multivariate analysis of variance model useful especially for growth curve problems. Biometrika Vol 51 pp 313-326*)

Pottoff and Roy investigated the growth curves of some dental measurement over time in two groups of children: 11 girls and 16 boys. Each child was measured on 4 occasions of age, in years: 8 10, 12 and 14.

- Layout is Univariate/Long

<u>id</u>	<u>group</u>	<u>age</u>	<u>growth</u>
1	1	8	21.0
1	1	10	20.0
1	1	12	21.5
1	1	14	23.0
.... Rows omitted ...			
27	2	8	22.0
27	2	10	21.5
27	2	12	23.5
27	2	14	25.0

- Typically, the long format is used for univariate repeated measurements analyses

- Layout is Multivariate/Wide

<u>id</u>	<u>group</u>	<u>Y₁</u>	<u>Y₂</u>	<u>Y₃</u>	<u>Y₄</u>
1	1	21.0	20.0	21.5	23.0
.... Rows omitted ...					
27	2	22.0	21.5	23.5	25.0



- Typically the wide format is used for multivariate repeated measurements analysis approaches
- **Tip!** Notice the absence of recorded information on the actual occasions of measurements in the multivariate/wide layout. Thus, this layout is meaningful only when the occasions of measurement are the same for all subjects.

(a) Reshaping from WIDE to LONG Using **Reshape long**

How to follow along:

- 1) Download from the course website the data set **roy_wide.dta**.
- 2) clear
- 3) From whatever directory is appropriate for you, open **roy_wide.dta**

Step1 – Begin with a layout of data in WIDE format, as in the following example:

IMPORTANT! Your repeated measures variables must have the same prefix. Here it is “**growth**”

	id	growth8	growth10	growth12	growth14	x
1.	1	21	20	21.5	23	1
2.	2	21	21.5	24	25.5	1
3.	3	20.5	24	25.5	26	1
4.	4	23.5	24.5	25	26.5	1
5.	5	21.5	23	22.5	23.5	1

Step2 – Specify two variable names that will be used in your LONG data format

- (1) A new one that you choose: For your repeated measures occasion: Example – **age**
- (2) This must be the prefix: For your repeated measures outcome: Example – **growth**



Step3 – Note the variable (or variables plural!) that identifies each unique repeated measures profile:

Example – **id**

Reshaping from WIDE to LONG Using Reshape long

. reshape long newoutcome, i(identifiedvariable) j(newoccasionvariable)

Note - The i and the j are required

Example -

. use "http://people.umass.edu/biep691f/data/roy_wide.dta", clear

. reshape long growth, i(id) j(age)

(note: j = 8 10 12 14)

Data	wide	->	long
Number of obs.	27	->	108
Number of variables	6	->	4
j variable (4 values)		->	age
xij variables:			
	growth8 growth10 ... growth14	->	growth

Check: list id age growth in 1/8, sepby(id)

	id	age	growth
1.	1	1	21
2.	1	2	20
3.	1	3	21.5
4.	1	4	23
5.	2	1	21
6.	2	2	21.5
7.	2	3	24
8.	2	4	25.5

(b) Reshaping from LONG to WIDE Using Reshape wide

How to follow along:

- 1) Download from the course website the data set `roy_long.dta`.
- 2) clear
- 3) From whatever directory is appropriate for you, open `roy_long.dta`

Step1 – Begin with a layout of data in LONG format, as in the following partial listing:

`. list id group age growth in 1/12, sepby(id)`

	id	group	age	growth
1.	1	1	8	21
2.	1	1	10	20
3.	1	1	12	21.5
4.	1	1	14	23
5.	2	1	8	21
6.	2	1	10	21.5
7.	2	1	12	24
8.	2	1	14	25.5
9.	3	1	8	20.5
10.	3	1	10	24
11.	3	1	12	25.5
12.	3	1	14	26

Step2 – Identify the following

- (1) The long format variable outcome that is going to be made wide Example – **growth**
- (2) The variable (or variables plural) that identifies each profile Example – **id**

(3) The variable whose values will be the suffix of the outcome variable names

Example – **age**

Reshaping from LONG to WIDE Using Reshape wide

. reshape wide *longoutcome*, i(*identifyingvariable*) j(*longvariable_ containingsuffix*)

Note - The i and the j are required

Example -

. reshape wide growth, i(id) j(age)

(note: j = 8 10 12 14)

Data	long	->	wide
Number of obs.	108	->	27
Number of variables	4	->	6
j variable (4 values)	age	->	(dropped)
xij variables:	growth	->	growth8 growth10 ... growth14

Check:

.list id group growth8 growth10 growth12 growth 14 in 1/5

	id	group	growth8	growth10	growth12	growth14
1.	1	1	21	20	21.5	23
2.	2	1	21	21.5	24	25.5
3.	3	1	20.5	24	25.5	26
4.	4	1	23.5	24.5	25	26.5
5.	5	1	21.5	23	22.5	23.5

5.2 Line Plot or Profile Plot Using **xtset** and **xtline**. Data are in **LONG** format.

Graph Desired	Syntax in Stata
Plot of individual profiles (indexed by some identifying variable such as id) over some panel dimension (eg over time).	<pre>. xtset <i>IDvariable</i> <i>panelvariable</i> . xtline <i>variablename</i>, t(<i>panelvariable</i>) i(<i>IDvariable</i>)</pre>

```
. * Data are in long format.
. clear
. use "/Users/cbigelow/Desktop/roy_long.dta", clear
. set scheme slmono

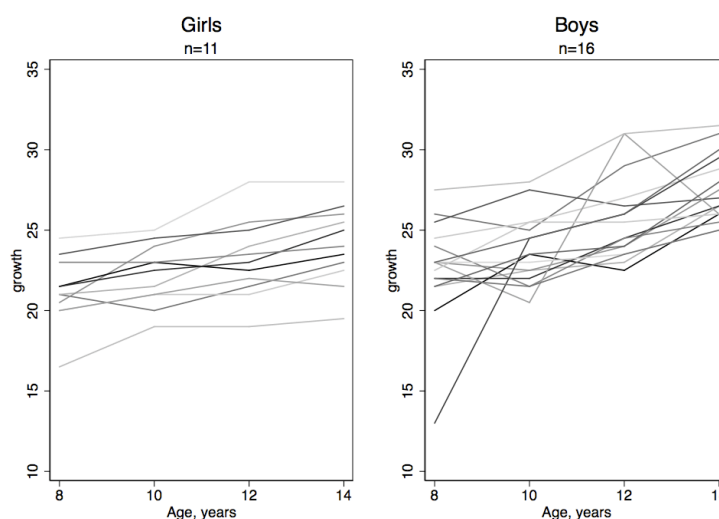
. * For illustration - We'll clear the xtset so that we can do it again.
. xtset, clear

. * Declare data to be panel data using command xtset
. xtset id age

. * Tip - Make sure Y-axis is the same for Girls and Boys
. xtline growth if group==1, t(age) i(id) overlay title("Girls") subtitle("n=11") ylabel(10 (5) 35)
legend(off) name(girls, replace)

. xtline growth if group==2, t(age) i(id) overlay title("Boys") subtitle("n=16") ylabel(10 (5) 35)
legend(off) name(boys, replace)

. graph combine girls boys
```



5.3 Line Plot or Profile Plot Using **xtset** and **graph twoway**. Data are in **LONG** format.

Graph Desired	Syntax in Stata
Overlay plot of individual profiles (indexed by some identifying variable such as id) over some panel dimension (eg over time).	<code>. xtset <i>IDvariable</i> <i>panelvariable</i></code> <code>. graph twoway (connected <i>panelvariable</i> <i>timevariable</i>, connect(L) symbol(d))</code>

```

.* Data are in long format.
. clear
. use "/Users/cbigelow/Desktop/roy_long.dta", clear
. set scheme slmono

. * For illustration - We'll clear the xtset so that we can do it again.
. xtset, clear

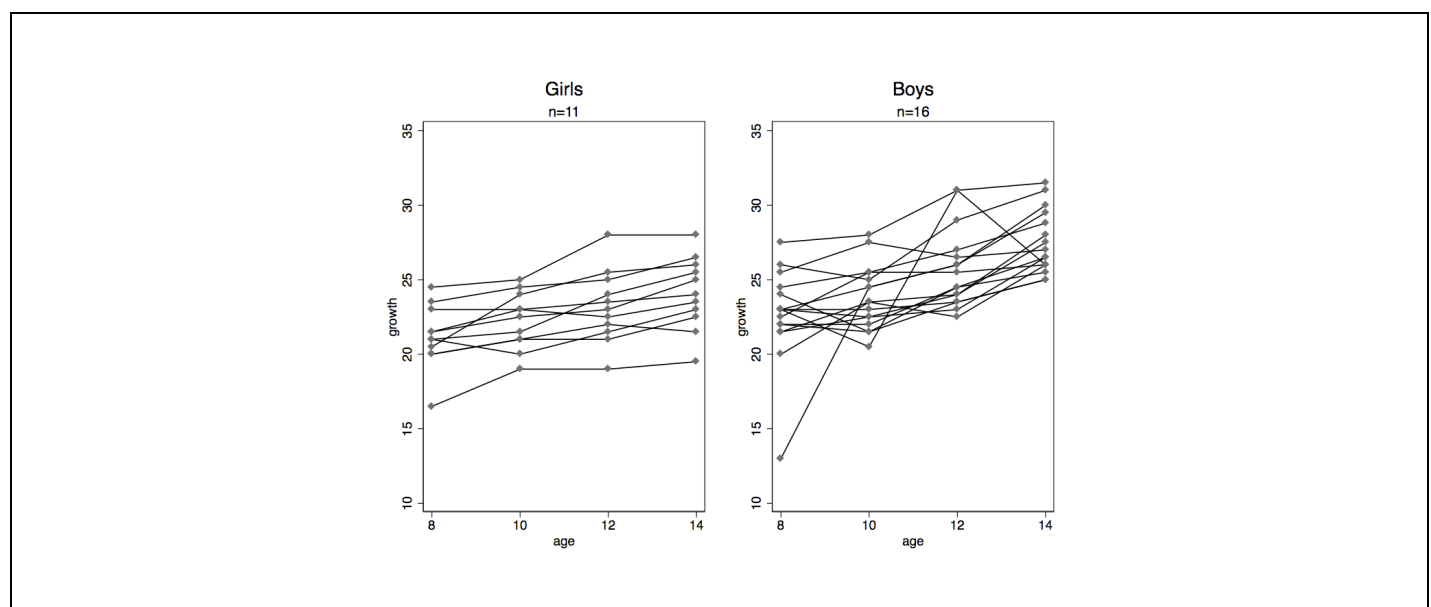
. * Declare data to be panel data using command xtset
. xtset id age

.* Tip - Use option ylabel( ) to make sure Y-axis is the same for Girls and Boys
. graph twoway (connected growth age, connect(L) symbol(d)) if group==1, title("Girls")
  subtitle("n=11") ylabel(10(5)35) xlabel(8(2)14) name(girls, replace)

. graph twoway (connected growth age, connect(L) symbol(d)) if group==2, title("Boys")
  subtitle("n=16") ylabel(10(5)35) xlabel(8(2)14) name(boys, replace)

. graph combine girls boys

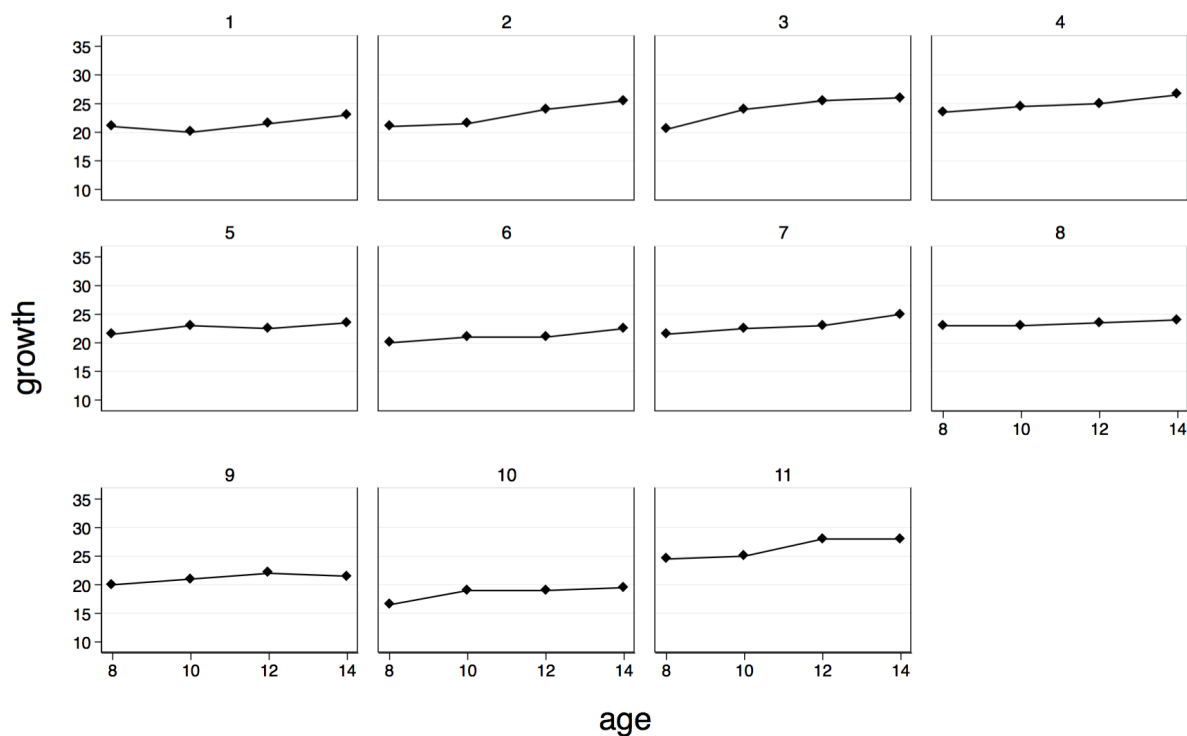
```



Tip! Sometimes, especially in data screening of small to modest sample size data sets, it is possible to produce the repeated measures profiles in separate plots. To do this, add the option **by(*IDvariable*)**

Graph Desired	Syntax in Stata
Separate plots of individual profiles (indexed by some identifying variable such as id) over some panel dimension (eg over time).	. xtset <i>IDvariable</i> <i>panelvariable</i> . graph twoway (connected <i>panelvariable</i> <i>timevariable</i>, connect(L) symbol(d)), by(<i>IDvariable</i>)

```
.* For illustration (only so that the plot is not too busy): Girls only
. graph twoway (connected growth age, connect(L) symbol(d)) if group==1, by(id) ylabel(10(5)35)
  xlabel(8(2)14)
```



5.4 Mean Profile Using Collapse and Graph Twoway Data are in LONG format.

HIGHLY RECOMMENDED! Tip – To obtain the same graph much more simply (albeit perhaps with less bells and whistles) consider jumping ahead to Section 5.6 Mean Profile Using profileplot.

```
. * Use command collapse to create a data set with mean, std, and n. Check.
. collapse (mean) growth (sd) sdgrowth=growth (count) n=growth, by(group age)
. list

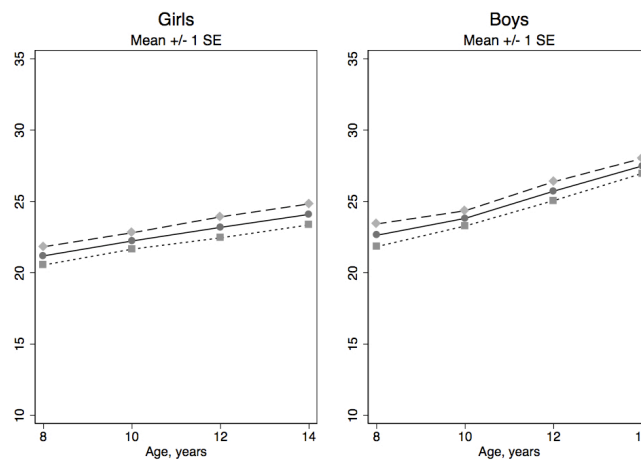
      age      group      growth      sdgrowth      n
1.       8         1     21.18182     2.124532     11
2.      10         1     22.22727     1.902152     11
3.      12         1     23.18182     2.442056     11
4.      14         1     24.09091     2.437398     11
5.       8         2     22.625      3.185906     16
6.      10         2     23.8125     2.136001     16
7.      12         2     25.71875     2.651847     16
8.      14         2     27.4875     2.096624     16

. * Tell STATA to produce +/- 1 se bars
. generate high=growth+1*sdgrowth/sqrt(n)
. generate low=growth-1*sdgrowth/sqrt(n)

. * mean +/- SE plot for Girls
. graph twoway (connected growth age) (connected high age) (connected low age) if group==1,
ylabel(10 (5) 35) title("Girls") subtitle("Mean +/- 1 SE") legend(off) name(girls, replace)

. * mean +/- SE plot for Boys
. graph twoway (connected growth age) (connected high age) (connected low age) if group==2,
ylabel(10 (5) 35) title("Boys") subtitle("Mean +/- 1 SE") legend(off) name(boys, replace)

.graph combine girls boys
```



5.5 Side-by-Side Box and Whisker Plot Using Graph Box Data are in LONG format

Reference-

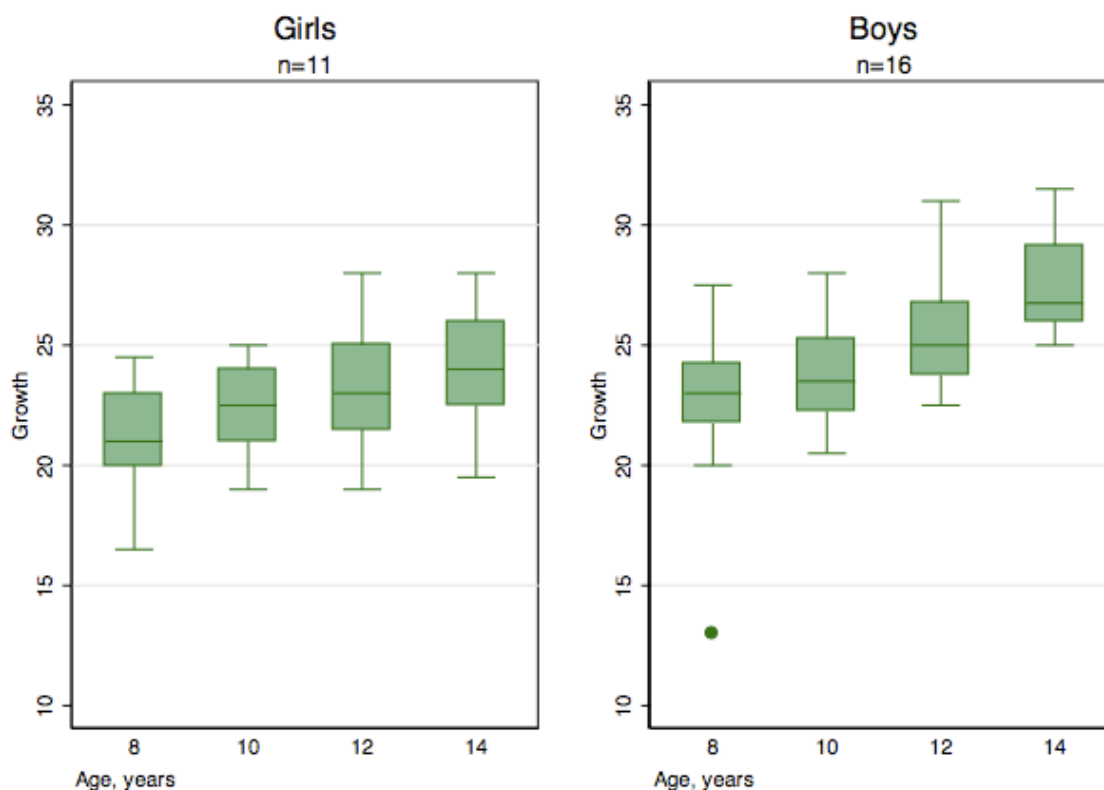
Section 4.2, *Box and Whisker Plot using graph box*, p. 36.

```
.* Data are in long format.
. clear
. use "http://people.umass.edu/biep691f/data/roy_long.dta", clear
. sort age

. * Tip - Make sure Y-axis is the same for Girls and Boys
. graph box growth if group==1, over(age) title("Girls") subtitle("n=11") ytitle("Growth")
ylabel(10 (5) 35) caption("Age, years") name(girls, replace)

. graph box growth if group==2, over(age) title("Boys") subtitle("n=16") ytitle("Growth")
ylabel(10 (5) 35) caption("Age, years") name(boys, replace)

.graph combine girls boys
```



5.6 Mean Profile Using `profileplot` Data are in **WIDE** format

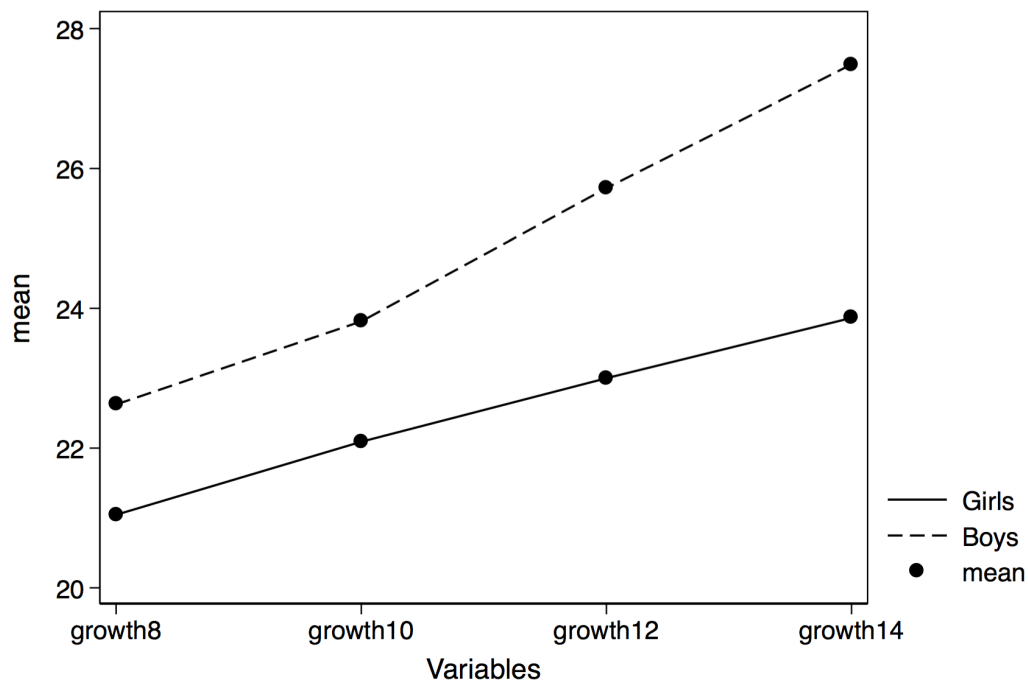
Preliminary - The command `profileplot` is not in your installation. Issue the command `findit profileplot` and then follow the instructions to download it to your computer.

How to follow along:

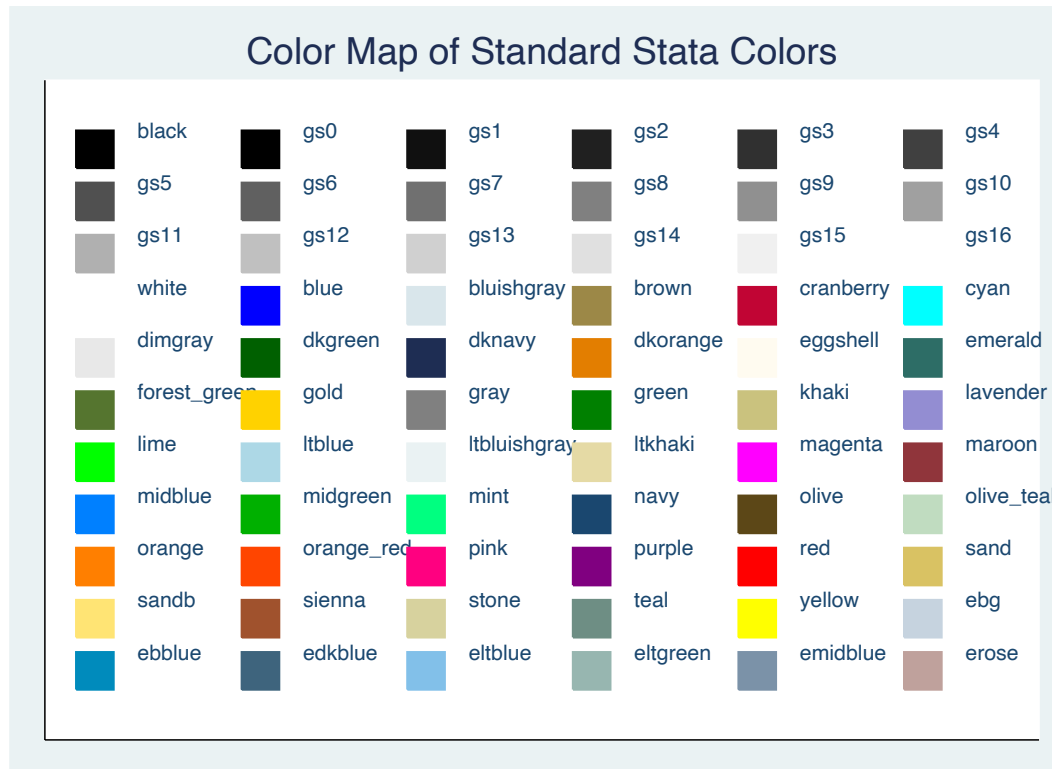
- 1) Download from the course website the data set `roy_wide.dta`.
- 2) `clear`
- 3) From whatever directory is appropriate for you, open `roy_wide.dta`

```
.* Data are in wide format.
. clear
. use "/Users/cbigelow/Desktop/roy_wide.dta", clear

. sort x
. profileplot growth8 growth10 growth12 growth14
```

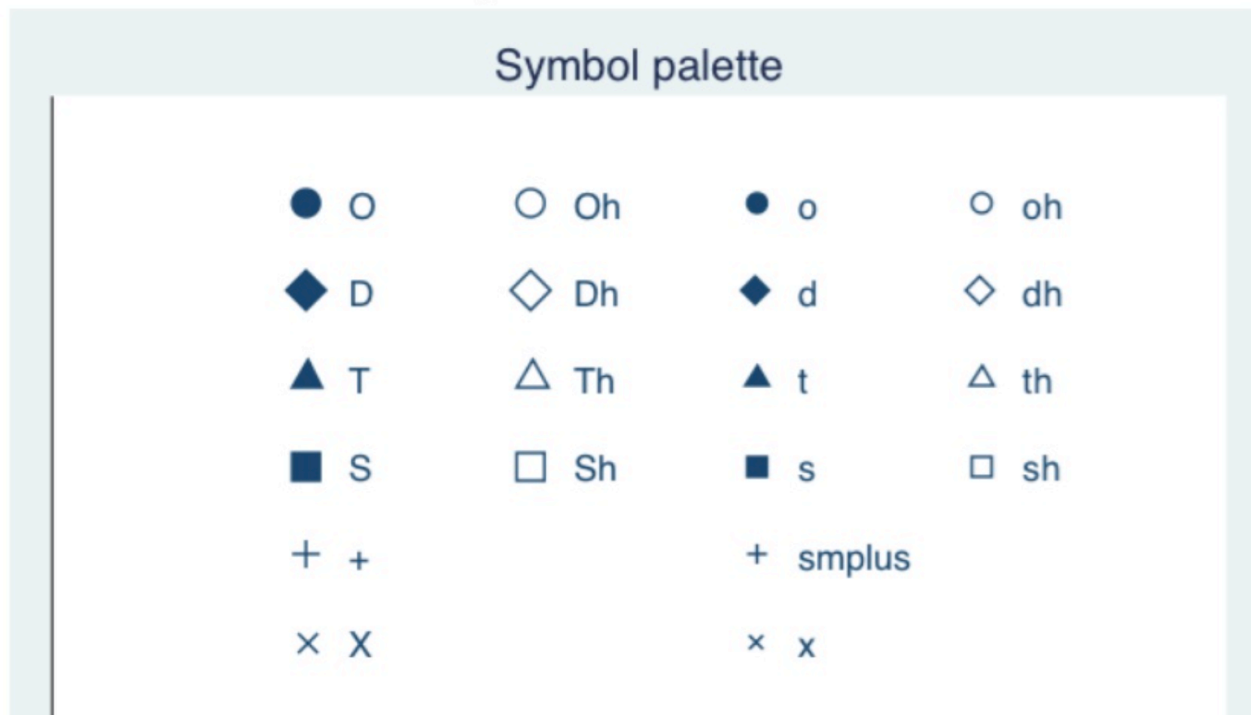


Appendix A1 Color Palette



Appendix A2 Symbol Palette

- A variety of symbol shapes are available: use `palette` `symbolpalette` to see them and `msymbol()` to set them



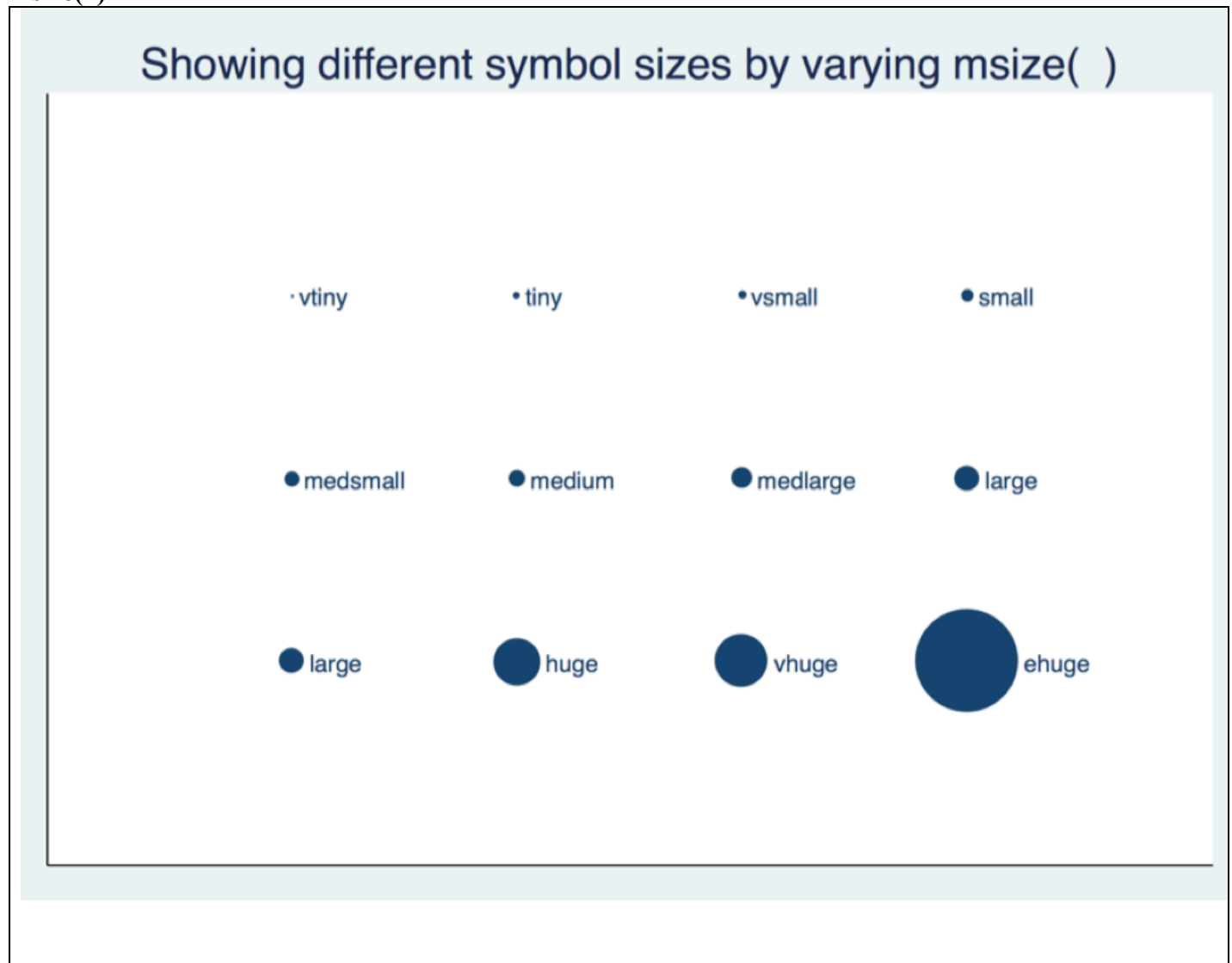
Appendix A3 Marker Size Palette

.* To see your marker size choices

.showmarkers, over(msize)

.* To set your your marker size, include in your graph options

msize()



Appendix A4 Line Type Palette

.* To see your line type choices

`.palette linepalette`

.* To set your line type, include in your graph options

`linetype()`

`palette linepalette`

Line pattern palette

	solid
	dash
	longdash_dot
	dot
	longdash
	dash_dot
	shortdash
	shortdash_dot
	blank