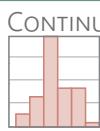


# Data Visualization with Stata

## Cheat Sheet

For more info, see Stata's reference manual ([stata.com](#))

### ONE VARIABLE



**histogram** mpg, width(5) freq kdensity kdenopts(bwidth(5))

**kdensity** mpg, bwidth(3)  
smoothed histogram  
bwidth • kernel(<options>) • normal • normopts(<line options>)

main plot-specific options;  
see help for complete set

### CONTINUOUS



**graph bar** (count), over(foreign, gap(\*0.5)) intensity(\*0.5)  
**bar plot**

**graph hbar** draws horizontal bar charts

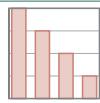
(asis) • (percent) • (count) • over(<variable>, <options: gap(#)> • relabel • descending • reverse) • cw • missing • nofill • allcategories • percentages • stack • bargap(#) • intensity(#\*) • yalternate • xlabelname

**graph bar** (percent), over(rep78) over(foreign)  
**grouped bar plot**

**graph hbar ...**

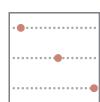
(asis) • (percent) • (count) • over(<variable>, <options: gap(#)> • relabel • descending • reverse) • cw • missing • nofill • allcategories • percentages • stack • bargap(#) • intensity(#\*) • yalternate • xlabelname

### DISCRETE X, CONTINUOUS Y



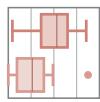
**graph bar** (median) price, over(foreign) **graph hbar ...**

**bar plot** (asis) • (percent) • (count) • (stat: mean median sum min max ...) over(<variable>, <options: gap(#)> • relabel • descending • reverse sort(<variable>)) • cw • missing • nofill • allcategories • percentages stack • bargap(#) • intensity(#\*) • yalternate • xlabelname



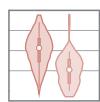
**graph dot** (mean) length headroom, over(foreign) m(1, ms(S))

**dot plot** (asis) • (percent) • (count) • (stat: mean median sum min max ...) over(<variable>, <options: gap(#)> • relabel • descending • reverse sort(<variable>)) • cw • missing • nofill • allcategories • percentages linegap(#) • marker#, <options> • linetype(dot | line | rectangle) dots(<options>) • lines(<options>) • rectangles(<options>) • rwidth



**graph hbox** mpg, over(rep78, descending) by(foreign) **graph box** draws vertical boxplots

over(<variable>, <options: total • gap(#)> • relabel • descending • reverse sort(<variable>)) • missing • allcategories • intensity(#\*) • boxgap(#) medtype(line | line | marker) • medline(<options>) • medmarker(<options>)

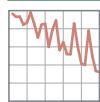


**vioplot** price, over(foreign) **ssc install vioplot**

over(<variable>, <options: total • missing>) • nofill • vertical • horizontal • obs • kernel(<options>) • bwidth(#) • barwidth(#) • dscale(#) • ygap(#) • ogap(#) • density(<options>) bar(<options>) • median(<options>) • obsopts(<options>)

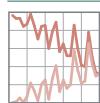
### Plot placement

#### JUXTAPOSE (FACET)



**twoway scatter** mpg price, by(foreign, norescale)  
total • missing • colfirst • rows(#) • cols(#) • holes(<numpList>) compact • [no]edgelabel • [no]rescale • [no]yrescale • [no]xrescale [no]jxaxes • [no]jyaxes • [no]jytitle • [no]jxtitle • [no]ylabel [no]xlabel • [no]ytitle • [no]xtitle • imargin(<options>)

#### SUPERIMPOSE



**graph combine** plot1.gph plot2.gph...

combine two or more saved graphs into a single plot

**scatter** y3 y2 y1 x, msymbol(i o i) mlabel(var3 var2 var1)

plot several y values for a single x value  
**graph twoway scatter** mpg price in 27/74 || scatter mpg price /\* \* if mpg < 15 & price > 12000 in 27/74, mlabel(make) m(i)  
combine twoway plots using ||

BASIC PLOT SYNTAX:

<b>graph</b> <plot type> variables: y first <b>titles</b> <b>title</b> ("title") <b>subtitle</b> ("subtitle") <b>xtitle</b> ("x-axis title") <b>ytitle</b> ("y axis title") <b>xscale</b> (range(low high) log reverse off noline) <b>yscale</b> (<options>)	<b>plot-specific options</b> <b>facet</b> – <b>by(var)</b> <b>xline</b> (xint) <b>yline</b> (yint) <b>text</b> (y x "annotation")	<b>axes</b> <b>custom appearance</b> <b>background options</b>	<b>plot size</b> <b>scheme</b> (s1mono) <b>play</b> (customTheme) <b>xsize</b> (5) <b>ysize</b> (4) <b>save</b>	<b>annotations</b> <b>text</b> (y x "annotation")
--	--	--	--	--

### TWO+ CONTINUOUS VARIABLES



**graph matrix** mpg price weight, half  
scatterplot of each combination of variables  
half • jitter(#) • jitterseed(#) • diagonal • [aweights(<variable>)]



**twoway scatter** mpg weight, jitter(7)  
scatterplot  
jitter(#) • jitterseed(#) • sort • cmissing(yes | no) • connect(<options>) • [aweight(<variable>)]



**twoway scatter** mpg weight, mlabel(mpg)  
scatterplot with labelled values  
jitter(#) • jitterseed(#) • sort • cmissing(yes | no) • connect(<options>) • [aweight(<variable>)]



**twoway connected** mpg price, sort(price)  
scatterplot with connected lines and symbols  
jitter(#) • jitterseed(#) • sort • see also line • connect(<options>) • cmissing(yes | no)



**twoway area** mpg price, sort(price)  
line plot with area shading  
sort • cmissing(yes | no) • vertical • horizontal base(#)



**twoway bar** price rep78  
bar plot  
vertical • horizontal • base(#) • barwidth(#)



**twoway dot** mpg rep78  
dot plot  
vertical • horizontal • base(#) • ndots(#) • dcolor(<color>) • dfcolor(<color>) • dlcolor(<color>) • dsizel(<marker size>) • dsymbol(<marker type>) • dlwidth(<stroke size>) • dottedrend(yes | no)



**twoway dropline** mpg price in 1/5  
dropped line plot  
vertical • horizontal • base(#)



**twoway rcapsym** length headroom price  
range plot (y1 ÷ y2) with capped lines  
vertical • horizontal

see also rcap



**twoway rarea** length headroom price, sort  
range plot (y1 ÷ y2) with area shading  
vertical • horizontal • sort  
cmissing(yes | no)



**twoway rbar** length headroom price  
range plot (y1 ÷ y2) with bars  
vertical • horizontal • barwidth(#) • mwidth  
msize(<marker size>)



**twoway pcspike** wage68 ttl\_exp68 wage88 ttl\_exp88  
Parallel coordinates plot  
vertical • horizontal



**twoway pccapsym** wage68 ttl\_exp68 wage88 ttl\_exp88  
Slope/bump plot  
vertical • horizontal • headlabel

### THREE VARIABLES



**twoway contour** mpg price weight, level(20) crule(intensity)  
3D contour plot  
ccuts(#s) • levels(#s) • minmax • crule(hue | chue | intensity | linear) • scolor(<color>) • ecolor(<color>) • colors(<colorlist>) • heatmap  
interp(thinplatespline | shepard | none)



**regress** price mpg trunk weight length turn, nocons  
matrix regmat = e(V)  
**plotmatrix**, mat(regmat) color(green)  
heatmap  
mat(<variable>) • split(<options>) • color(<color>) • freq

### SUMMARY PLOTS



**twoway mband** mpg weight || scatter mpg weight  
plot median of the y values  
bands(#)



**binscatter** weight mpg, line(None) **ssc install binscatter**  
plot a single value (mean or median) for each x value  
medians • nquantiles(#s) • discrete • controls(<variables>) • linetype(lfit | qfit | connect | none) • aweight(<variable>)

### FITTING RESULTS



**twoway lfitci** mpg weight || scatter mpg weight  
calculate and plot linear fit to data with confidence intervals  
level(#s) • stdp • stdf • nofit • fitplot(<plottype>) • ciplot(<plottype>)  
range(# | #) • n(#s) • atobs • estopts(<options>) • predots(<options>)



**twoway lowess** mpg weight || scatter mpg weight  
calculate and plot lowess smoothing  
bwwidth(#) • mean • noweight • logit • adjust



**twoway qfici** mpg weight, alwidth(None) || scatter mpg weight  
calculate and plot quadratic fit to data with confidence intervals  
level(#s) • stdp • stdf • nofit • fitplot(<plottype>) • ciplot(<plottype>)  
range(# | #) • n(#s) • atobs • estopts(<options>) • predots(<options>)

### REGRESSION RESULTS



**regress** price mpg headroom trunk length turn  
**coefplot**, drop(cons) xline(0) **ssc install coefplot**  
Plot regression coefficients  
baselevels • b(<options>) • at(<options>) • noci • levels(#s)  
keep(<variables>) • drop(<variables>) • rename(<list>)  
horizontal • vertical • generate(<variable>)



**regress** mpg weight length turn  
margins, eyex(weight) at(weight = (1800(200)4800))  
**marginsplot**, noci  
Plot marginal effects of regression  
horizontal • noci