Plotting regression coefficients and other estimates

Ben Jann

University of Bern, ben.jann@soz.unibe.ch

Odense, December 6, 2018
Outline

- Introduction
- The coefplot command
  - Basic usage
  - Labels
  - Confidence intervals
  - The recast option
  - Marker labels
  - The at option
Introduction

- Statistical estimates such as coefficients from regression models are often presented as tables in research articles and presentations.

- However, graphs can be much more effective than tabulation. This is because the . . .

  “. . . reexpression of data in pictorial form capitalizes upon one of the most highly developed human information processing capabilities – the ability to recognize, classify, and remember visual patterns.” (Lewandowsky and Spence 1989:200)

- Graphs do a great job in “revealing patterns, trends, and relative quantities” (Jacoby 1997:7) because they translate differences among numbers into spatial distances, thereby emphasizing the main features of the data.

- Furthermore, pictorial representations seem to be easier to remember than tabular results (Lewandowsky and Spence 1989).
(Lewandowsky and Spence 1989:209)
Introduction

- In many applications, statistics is about estimation based on sample data. Since estimation results are uncertain, standard errors, statistical tests, or confidence intervals are reported.

- Visualizations of results should reflect precision or uncertainty. This is why so called “ropeladder” or “error bar” plots have become increasingly popular. They display, against a common scale,
  - markers for point estimates (e.g. of regression coefficients)
  - and spikes or bars for confidence intervals (“error bars”).

- Such plots are effective because they capitalize on two of the most powerful perceptual capabilities of humans – evaluating the position of points along a common scale and judging the length of lines (Cleveland and McGill 1985). Furthermore, they provide a much better impression of statistical precision than p-values or significance stars in tables.
(Thanks to Nick Cox for pointing me to this and some of the following references.)
To embark on a long series of analyses in order to determine error is always a considerable undertaking and is often impossible owing to the tendency of organic substances to change with time: added to this, unless special precautions are taken, such as were taken in 1905, the operators may, in spite of themselves, be more careful when analysing special samples of this kind, so that the series may not represent a random sample of analytical errors.

\[
\sigma = \frac{1}{\sqrt{n}} \cdot \sqrt{\frac{S \cdot (a - \bar{a})^2}{S(n-1)}},
\]

where  
- \( a \) = Average of a Farm,  
- \( \bar{a} \) = Mean of a Day’s Analyses,  
- \( n \) = Number of Farms analysed in the Day.

Fig. 1. Graph of body lengths in millimeters of several stocks of *Peromyscus maniculatus*. The length of each line represents the extremes of that set of measurements. The middle crossbar represents the mean. Two other crossbars are placed three times the probable error (\( \pm \) two times the standard error) of the mean above and below the mean, respectively, forming a rectangle. If the rectangles of two comparable lines do not overlap in vertical position the two means are indicated to differ by a statistically significant amount. For example, in body length the Linwood stock does not differ significantly from the Sheep Creek stock, but it is significantly shorter than the Summit Springs stock, and is significantly longer than either the Victorville, Grafton, or Buena Vista stocks.
3.78 TWO-TIERED ERROR BARS. The outer error bars are 95% confidence intervals and the inner error bars are 50% confidence intervals. The goal in this method is to show confidence intervals and not standard errors, although for some statistics, confidence intervals happen to be formed from multiples of standard errors.


FIGURE 20.4: Hazard ratios and multilevel confidence bars for effects of predictors in model, using default ranges except for ap.
Creating graphs of point estimates and confidence intervals has been notoriously difficult in Stata (although see Newson 2003).

1. gather coefficients and variances from the e()-returns
2. compute confidence intervals
3. store results as variables
4. create a variable for the category axis
5. compile labels for coefficients
6. run a lengthy graph command

Things got better with the introduction of marginsplot in Stata 12. With marginsplot it is easily possible to create a ropeladder plot from results left behind by margins.
. sysuse auto, clear
(1978 Automobile Data)
. regress price mpg trunk length turn

<table>
<thead>
<tr>
<th>Source</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>Number of obs = 74</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>159570047</td>
<td>4</td>
<td>39892511.8</td>
<td>F( 4, 69) = 5.79</td>
</tr>
<tr>
<td>Residual</td>
<td>475495349</td>
<td>69</td>
<td>6891236.94</td>
<td>Prob &gt; F = 0.0004</td>
</tr>
<tr>
<td>Total</td>
<td>635065396</td>
<td>73</td>
<td>8699525.97</td>
<td>R-squared = 0.2513</td>
</tr>
</tbody>
</table>

| price | Coef.  | Std. Err. | t   | P>|t| | [95% Conf. Interval] |
|-------|--------|-----------|-----|-----|----------------------|
| mpg   | -186.8417 | 88.17601  | -2.12 | 0.038 | -362.748 to -10.93533 |
| trunk | -12.72642 | 104.8785  | -0.12 | 0.904 | -221.9534 to 196.5005 |
| length| 54.55294  | 35.56248  | 1.53  | 0.130 | -16.39227 to 125.4981 |
| turn  | -200.3248 | 140.0166  | -1.43 | 0.157 | -479.6502 to 79.00066 |
| _cons | 8009.893  | 6205.538  | 1.29  | 0.201 | -4369.817 to 20389.6  |
. margins, dydx(*)

Average marginal effects
Number of obs = 74
Model VCE : OLS
Expression : Linear prediction, predict()
dy/dx w.r.t. : mpg trunk length turn

<table>
<thead>
<tr>
<th></th>
<th>Delta-method</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>dy/dx</td>
<td>Std. Err.</td>
<td>t</td>
<td>P&gt;</td>
<td>t</td>
</tr>
<tr>
<td>mpg</td>
<td>-186.8417</td>
<td>88.17601</td>
<td>-2.12</td>
<td>0.038</td>
<td>-362.748</td>
</tr>
<tr>
<td>trunk</td>
<td>-12.72642</td>
<td>104.8785</td>
<td>-0.12</td>
<td>0.904</td>
<td>-221.9534</td>
</tr>
<tr>
<td>length</td>
<td>54.55294</td>
<td>35.56248</td>
<td>1.53</td>
<td>0.130</td>
<td>-16.39227</td>
</tr>
<tr>
<td>turn</td>
<td>-200.3248</td>
<td>140.0166</td>
<td>-1.43</td>
<td>0.157</td>
<td>-479.6502</td>
</tr>
</tbody>
</table>

Ben Jann (University of Bern)
. marginsplot, horizontal xline(0) yscale(reverse) recast(scatter)

Variables that uniquely identify margins: _deriv

Ben Jann (University of Bern)
The `coefplot` command

- `marginsplot` is a very versatile command that can do much more than what was shown, especially when plotting predictive margins.
- However, `marginsplot` can only deal with results left behind by `margins` and also has various other limitations.
- I therefore wrote a new command called `coefplot`. It is a general tool to graph results from estimation commands in Stata, similar to `outreg` (Gallup 2012) or `estout` (Jann 2007) for tables.
The coefplot command

- Some of coefplot’s functionality overlaps with the possibilities offered by marginsplot, but coefplot goes much beyond:
  - coefplot can be applied to the results of any estimation command that posts its results in `e()` and can also be used to plot results that have been collected manually in matrices.
  - Results from multiple models can be freely combined and arranged in a single graph, including the possibility to distribute results across subgraphs.
  - Given the criticism of a strict interpretation of significance tests and confidence intervals it seems often advisable to display multiple confidence intervals using varying levels. coefplot offers such functionality.
  - Good graphs need good labels. coefplot offers various options to label coefficients, equations, and subgraphs, include labels for groups of estimates, or insert subheadings to structure the display.
## Syntax

```r
coefplot subgraph [ || subgraph || ... ] [, globalopts ]
```

where `subgraph` is defined as

```r
(plot) [ (plot) ... ] [, subgropts ]
```

and `plot` is either `_skip` (to skip a plot) or

```r
model [ \ model \ ... ] [, plotopts ]
```

and `model` is

```r
namelist [, modelopts ]
```

where `namelist` is a list of names of stored models (see help `estimates`; type `.` or leave blank to refer to the active model). The * and ? wildcards are allowed in `namelist`; see *Using wildcards in model names*. Furthermore, `model` may also be

```r
matrix(mspec) [, modelopts ]
```

to plot results from a matrix (see *Plotting results from matrices* below). Parentheses around `plot` can be omitted if `plot` does not contain spaces.

### modelopts

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main</strong></td>
</tr>
<tr>
<td><strong>omitted</strong></td>
</tr>
<tr>
<td><strong>baselevels</strong></td>
</tr>
<tr>
<td><strong>b(mspec)</strong></td>
</tr>
<tr>
<td><strong>at([spec])</strong></td>
</tr>
<tr>
<td><strong>keep(coeflist)</strong></td>
</tr>
<tr>
<td><strong>drop(coeflist)</strong></td>
</tr>
</tbody>
</table>
Basic usage: Plotting a single model

. quietly sysuse auto, clear
. quietly regress price mpg trunk length turn
. coefplot, drop(_cons) xline(0)
Basic usage: Plotting a single model

Odds ratios from a logit model (eform option)

. quietly logit foreign mpg trunk length turn
. coefplot, drop(_cons) xline(1) eform xtitle(Odds ratio)
Basic usage: Plotting multiple models

. quietly regress price mpg trunk length turn if foreign==0
. estimates store domestic
. quietly regress price mpg trunk length turn if foreign==1
. estimates store foreign
. coefplot (domestic, label(Domestic Cars)) (foreign, label(Foreign Cars)), drop(_cons) xline(0)
Basic usage: Multiple models on separate scales

. quietly regress price mpg trunk length turn
. estimates store Price
. quietly regress weight mpg trunk length turn
. estimates store Weight
. coefplot (Price) (Weight, axis(2)), drop(_cons) xtitle(Price) xtitle(Weight, axis(2))
Basic usage: Appending models

. quietly eststo multivariate: regress price mpg trunk length turn
. foreach var in mpg trunk length turn { // (using -eststo- from -estout- pkg for sake of brevity)
2.   quietly eststo b`var´: regress price `var´
3. }
. coefplot (b_mpg \\ b_trunk \\ b_length \\ b_turn, label(bivariate)) (multivariate) ///
> , drop(_cons) xline(0)
Basic usage: Appending models

Using wildcard syntax to select the models

```
. coefplot (b_*, label(bivariate)) (multivariate), drop(_cons) xline(0)
```
Basic usage: Subgraphs

```stata
. quietly eststo Price_D: regress price mpg trunk length turn if foreign==0
. quietly eststo Price_F: regress price mpg trunk length turn if foreign==1
. quietly eststo Weight_D: regress weight mpg trunk length turn if foreign==0
. quietly eststo Weight_F: regress weight mpg trunk length turn if foreign==1
. coefplot (Price_D) (Price_F), bylabel(Price) ///
>     || (Weight_D) (Weight_F), bylabel(Weight) ///
>     ||, plotlabels(Domestic Foreign) drop(_cons) xline(0) byopts(xrescale)
```

---

**Plotting Estimates**

Ben Jann (University of Bern)
Basic usage: Subgraphs

Using wildcard syntax

```
  . coefplot Price*, bylabel(Price) ///
  >    || Weight*, bylabel(Weight) ///
  >    ||, plotlabels(Domestic Foreign) drop(_cons) xline(0) byopts(xrescale)
```

![Graph showing coefficient plots for Price and Weight with subgraphs for Mileage, Trunk space, Length, and Turn Circle. The graphs are divided by Domestic and Foreign, with various values displayed along the x-axis.]
Basic usage: Subgraphs

Different plot styles across subgraphs

```
. coefplot (Price_D, label(Model 1)) (Price_F, label(Model 2)), bylabel(Price) ///
> || (Weight_D, label(Model 3)) (Weight_F, label(Model 4)), bylabel(Weight) ///
> ||, drop(_cons) xline(0) byopts(xrescale) norecycle legend(rows(1))
```

![Graph showing different plot styles across subgraphs](image)

- Mileage (mpg)
- Trunk space (cu. ft.)
- Length (in.)
- Turn Circle (ft.)

-1000 -500 0 500

-100 -50 0 50 100

- Model 1
- Model 2
- Model 3
- Model 4

Ben Jann (University of Bern) Plotting Estimates Odense, 06.12.2018 24
Basic usage: Subgraphs by coefficients

quietly eststo Domestic: regress price mpg trunk length turn if foreign==0
quietly eststo Foreign: regress price mpg trunk length turn if foreign==1
quietly eststo Pooled: regress price mpg trunk length turn
coefplot Domestic || Foreign || Pooled, drop(_cons) yline(0) vertical bycoefs byopts(yrescale)
Basic usage: Plotting results from matrix

```
matrix median = J(5, 3, .)
matrix coln median = median ll95 ul95
matrix rown median = 1 2 3 4 5
forv i = 1/5 {
    quietly centile price if rep78==`i´
    matrix median[`i´,1] = r(c_1), r(lb_1), r(ub_1)
}
matrix list median
```

```
mean: mean price, over(rep78)
```

```
Mean estimation
```

<table>
<thead>
<tr>
<th>Over</th>
<th>Mean</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>price</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4564.5</td>
<td>369.5</td>
<td>3827.174 5301.826</td>
</tr>
<tr>
<td>2</td>
<td>5967.625</td>
<td>1265.494</td>
<td>3442.372 8492.878</td>
</tr>
<tr>
<td>3</td>
<td>6429.233</td>
<td>643.5995</td>
<td>5144.95 7713.516</td>
</tr>
<tr>
<td>4</td>
<td>6071.5</td>
<td>402.9585</td>
<td>5267.409 6875.591</td>
</tr>
<tr>
<td>5</td>
<td>5913</td>
<td>788.6821</td>
<td>4339.209 7486.791</td>
</tr>
</tbody>
</table>
Basic usage: Plotting results from matrix

```plaintext
.coeplot (mean) (matrix(median[,1]), ci((median[,2] median[,3]))), ytitle(Repair Record 1978)
```

![Graph showing mean and median repair records for 1978](image)
. quietly sysuse auto, clear
. quietly keep if rep78>=3
. quietly regress mpg headroom i.rep##i.foreign
. coefplot, xline(0) name(a)
. coefplot, xline(0) name(b) nolabels
. graph combine a b, xsize(8)
Custom labels, wrapping, grid, offsets, styling

. quietly sysuse auto, clear
. quietly keep if rep78>=3
. quietly eststo m1: regress mpg headroom i.rep i.foreign
. quietly eststo m2: regress mpg headroom i.rep##i.foreign
. coefplot (m1, offset(.15)) (m2, drop(*##*) offset(-.15)) (m2, keep(*##*) pstyle(p2)), ///
> xline(0) legend(off) msymbol(D) mfcolor(white) ciopts(lwidth(*3) lcolor(*.6)) ///
> grid(between glcolor(orange) glpattern(dash)) ///
> coeflabels(_cons = "Constant", wrap(20) notick labcolor(orange) labsize(medlarge) labgap(3))
. coefplot, xline(0) drop(_cons) omitted baselevels ///
>      headings(3.rep78 = "{bf:Repair Record}" 0.foreign = "{bf:Car Type}" ///
>      3.rep78#0.foreign = "{bf:Interaction Effects}", labcolor(orange))
 `. coefplot, xline(0) drop(headroom _cons) omitted baselevels
> graphregion(margin(l=65)) yscale(alt noline) coeflabels(labgap(-125) notick)
> headings(3.rep78 = "{bf:Repair Record}" 0.foreign = "{bf:Car Type}"
> 3.rep78#0.foreign = "{bf:Interaction Effects}", labcolor(orange) labgap(-130))

**Repair Record**
- Repair Record 1978=3
- Repair Record 1978=4
- Repair Record 1978=5

**Car Type**
- Domestic
- Foreign

**Interaction Effects**
- Repair Record 1978=3 # Domestic
- Repair Record 1978=3 # Foreign
- Repair Record 1978=4 # Domestic
- Repair Record 1978=4 # Foreign
- Repair Record 1978=5 # Domestic
- Repair Record 1978=5 # Foreign
Left-aligned labels (using the graph editor)

```
.coefplot, xline(0) drop(headroom _cons) omitted baselevels
> yscale(alt noline) coeflabels(, notick labgap(5))
> headings(3.rep78 = "{bf:Repair Record}" 0.foreign = "{bf:Car Type}"
> 3.rep78#0.foreign = "{bf:Interaction Effects}", labcolor(orange) labgap(0))
.gr_edit .move yaxis1 leftof 8 5
```

**Repair Record**
- Repair Record 1978=3
- Repair Record 1978=4
- Repair Record 1978=5

**Car Type**
- Domestic
- Foreign

**Interaction Effects**
- Repair Record 1978=3 # Domestic
- Repair Record 1978=3 # Foreign
- Repair Record 1978=4 # Domestic
- Repair Record 1978=4 # Foreign
- Repair Record 1978=5 # Domestic
- Repair Record 1978=5 # Foreign
. coefplot, xline(0) drop(_cons) omitted base ///
> headings(3.rep78 = "{it:Repair record:}" 0.foreign = "{it:Car type:}" ///
> , nogap labcolor(orange)) ///
> groups(headroom 1.foreign = "{bf:Main Effects}" ///
> ?.rep78#.foreign = "{bf:Interaction Effects}"), labcolor(orange))

---

**Headings and groups**

```
. coefplot, xline(0) drop(_cons) omitted base ///
> headings(3.rep78 = "{it:Repair record:}" 0.foreign = "{it:Car type:}" ///
> , nogap labcolor(orange)) ///
> groups(headroom 1.foreign = "{bf:Main Effects}" ///
> ?.rep78#.foreign = "{bf:Interaction Effects}"), labcolor(orange))
```
Confidence intervals: Multiple levels

. quietly sysuse auto, clear
. quietly eststo domestic: regress price mpg trunk length turn if foreign==0
. quietly eststo foreign: regress price mpg trunk length turn if foreign==1
. coefplot domestic foreign, drop(_cons) xline(0) msymbol(d) mfcolor(white) levels(99.9 99 95)
Confidence intervals: Harrell style

. quietly regress price mpg trunk length turn
. coefplot, drop(_cons) xline(0) msymbol(d) mcolor(white) ///
> levels(99 95 90 80 70) ciopts(lwidth(3 ..) lcolor(*.2 *.4 *.6 *.8 *1)) ///
> legend(order(1 "99" 2 "95" 3 "90" 4 "80" 5 "70") row(1))
Confidence intervals: Cleveland style

.coefplot domestic foreign, drop(_cons) xline(0) levels(95 50) ciopts(recast(. rcap))

Mileage (mpg)

Trunk space (cu. ft.)

Length (in.)

Turn Circle (ft.)

-1000 -500 0 500 1000
domestic foreign

Ben Jann (University of Bern)

Odense, 06.12.2018
Confidence intervals: Smoothed

.coefplot domestic foreign, drop(_cons) xline(0) msymbol(d) cissmooth
The recast option: Extremely useful!

. quietly eststo domestic: proportion rep if foreign==0
. quietly eststo foreign: proportion rep if foreign==1
. coefplot domestic foreign, vertical xtitle(Repair Record 1978) rescale(100) ytitle(Percent) ///
  recast(bar) barwidth(0.25) fcolor(*.5) ciopts(recast(rcap)) citop citype(logit)
The recast option: Combining different plot types

. quietly eststo prop: proportion rep78
. quietly eststo mean: mean price, over(rep78)
. coefplot (prop, recast(bar) noci barwidth(0.5) color(*.6)) ///
>   (mean, recast(connected) ciopts(recast(rcap)) axis(2)) ///
>   , vertical nooffsets plotlabels("Proportion" "Price") ///
>   xtitle("Repair record") ytitle("Proportion") ytitle("Price", axis(2))
The recast option: A more complicated bar chart

(Online Survey on "Exams and Written assignments" 2011)
. local vars q21_1 q21_2 q21_3 q21_4 q21_5
. local lblname q21_
. local levels 1 2 3 4 5
. local nvars: list sizeof vars
. local nlevels: list sizeof levels
. matrix p = J(`nlevels´, `nvars´, .)
. matrix colnames p = `vars´
. matrix rownames p = `levels´
. local i 0
. foreach v of local vars {
2. local ++i
3. quietly proportion `v´
4. matrix p[1,`i´] = e(b)´ * 100
5. }
. matrix r = p
. mata: st_replacematrix("r", mm_colrunsum(st_matrix("p")))
. mata: st_matrix("l", (J(1,`nvars´,0) \ st_matrix("r")[1::`nlevels´-1,]))
. matrix m = r
. mata: st_replacematrix("m", (st_matrix("l") :+ st_matrix("r"))/2)
. local plots
. local i 0
. foreach l of local levels {
2. local ++i
3. local lbl: lab `lblname´ `l´
4. local plots `plots´ (matrix(m[`i´]), ci((l[`i´] r[`i´])) aux(p[`i´]) ///
> key(ci) label(`lbl´))
5. }
. coefplot `plots´, nooffset ms(i) mlabel(@aux) mlabpos(0) format(%9.0f) ///
> coeflabels(), wrap(30)) ciopts(recast(rbar) barwidth(0.5)) ///
> legend(rows(1) span stack)

(OK, probably too complicated. I guess I should provide a wrapper for that.)
The recast option: A more complicated bar chart

How bad do you think is copying from other students during an exam?

How bad do you think is using illicit crib notes in an exam?

How bad do you think is using prescription drugs in an exam?

How bad do you think is handing in a paper containing plagiarisms?

How bad do you think is handing in someone else's work as one's own?

- not at all bad
- slightly bad
- somewhat bad
- rather bad
- very bad

Ben Jann (University of Bern)
Marker labels

. quietly sysuse auto, clear
. quietly eststo domestic: regress price mpg trunk length turn if foreign==0
. quietly eststo foreign: regress price mpg trunk length turn if foreign==1
. coefplot domestic foreign, drop(_cons) xline(0) mlabel format(%9.0f) mlabposition(12) mlabgap(*2)
Marker labels: A slightly involved example

. quietly regress price mpg trunk length turn if foreign==0
. mata: st_matrix("e(box)"), (st_matrix("e(b)") :- 65 \ st_matrix("e(b)") :+ 65))
. mata: st_matrix("e(spike)"), (st_matrix("e(b)") :- 1e-9 \ st_matrix("e(b)") :+ 1e-9 ))
. estimates store foreign
. quietly regress price mpg trunk length turn if foreign==1
. mata: st_matrix("e(box)"), (st_matrix("e(b)") :- 65 \ st_matrix("e(b)") :+ 65))
. mata: st_matrix("e(spike)"), (st_matrix("e(b)") :- 1e-9 \ st_matrix("e(b)") :+ 1e-9 ))
. estimates store domestic
. coefplot domestic foreign, drop(_cons) xline(0) legend(order(3 "Domestic" 7 "Foreign") ///
> msymb(i) mlabel format(%.9f) mlabposition(0) ci(95 spike box) ///
> ciopts(recast(. rbar rbar) barwidth(. 0.3 0.2) fcolor(. . white) lwidth(. medium medium))

Ben Jann (University of Bern)  Plotting Estimates  Odense, 06.12.2018 43
Marker labels: Custom labels

```stata
.coefplot (domestic, mlabels(length = 1 "+" * = 11 "0")) ///
  (foreign, mlabels(trunk length = 1 "+" * = 11 "0")) ///
  , drop(_cons) xline(0) ///
  subti("Hypotheses: 0 no effect, + positive effect, - negative effect", size(small))
```

Hypotheses: 0 no effect, + positive effect, - negative effect

- Mileage (mpg)
- Trunk space (cu. ft.)
- Length (in.)
- Turn Circle (ft.)

---

**Ben Jann (University of Bern)**

**Plotting Estimates**

**Odense, 06.12.2018**
Marker labels: P-Values

```stata
.coefplot domestic foreign, drop(_cons) xline(0) ///
>   mlabposition(1) mlabgap(*2) ///
>   mlabel("{it:p} = " + string(@pval,"%9.3f"))
```

![Diagram showing estimates with p-values]

- Mileage (mpg)
- Trunk space (cu. ft.)
- Length (in.)
- Turn Circle (ft.)

- Domestic: 
  - Mileage: p = 0.443
  - Trunk space: p = 0.693
  - Length: p = 0.332
  - Turn Circle: p = 0.931

- Foreign: 
  - Mileage: p = 0.255
  - Trunk space: p = 0.183
  - Length: p = 0.001
  - Turn Circle: p = 0.875

Ben Jann (University of Bern)
Plotting Estimates
Odense, 06.12.2018
. quietly regress mpg headroom i.rep##i.foreign if rep78==3
. coefplot, xline(0) mlabposition(1) ///
    mlabel(cond(@pval<.001, "***", ///
        cond(@pval<.01, "**", ///
            cond(@pval<.05, ",*", ///
                cond(@pval<.1, "+", ","))))))) ///
    note("+ p < .1, * p < .05, ** p < .01, *** p < .001")
By the way:

Here is how to draw significant and nonsignificant results in different colors

```
.. coefplot (., if(@pval>=.1) color(*.3) ciopts(lcolor(*.3))) ///
> (., if(@pval<.1 & @pval>=.05) pstyle(p1) color(*.6) ciopts(lcolor(*.6))) ///
> (., if(@pval<.05) pstyle(p1)) ///
> , plotlabels("n.s." "p < .1" "p < .05") legen(rows(1)) nooffset xline(0)
```
The at option

. quietly logit foreign mpg
. quietly eststo bivariate: margins, at(mpg=(10(2)40)) post
. quietly logit foreign mpg turn price
. quietly eststo multivariate: margins, at(mpg=(10(2)40)) post
. coefplot bivariate multivariate, at ytitle(Pr(foreign=1)) xtitle(Miles per Gallon) ///
   recast(line) lwidth(*2) ciopts(recast(rline) lpattern(dash))
The addplot command

- Due to the limitations of Stata’s graph system, some things are difficult to achieve within a single graph command. For example, options for customizing subgraphs are quite limited.
- In such cases, a solution can be to modify the graph manually using the graph editor.
- An alternative is the addplot command that can be applied to a graph after it has been created (type `ssc install addplot` to install the command; see Jann 2015 for a short paper on addplot).
addplot: Subgraph-specific xlines

. quietly logit foreign mpg trunk length turn
. coefplot ., bylabel(Log odds) || ., bylabel(Odds ratios) eform ///
> | |, drop(_cons) byopts(xrescale)
. addplot 1: , xline(0) norescaling
. addplot 2: , xline(1) norescaling
For more example see Jann (2014) or:

http://repec.sowi.unibe.ch/stata/coefplot
Some examples from my applied work

Schweizer
Ausländer
Ausweis C
In der Schweiz geboren
Im Ausland geboren
Ausweis B

Betäubungsmittelgesetz (BetmG)

Männer
Frauen

Besuch/Sicherstellung
Konsum
Anbau/Herstellung, Handel, Schmuggel
Europa
Afrika
Amerika
Asien

Prevalence estimate in %

Difference to DQ

copying from other students in exam
using crib notes in exam
taking drugs to enhance exam performance
including plagiarism in paper
handing in someone else's paper

Ben Jann (University of Bern)
Plotting Estimates

Odense, 06.12.2018
References I

References II


