Plotting regression coefficients and other estimates

Ben Jann

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

Odense, December 6, 2018

Outline

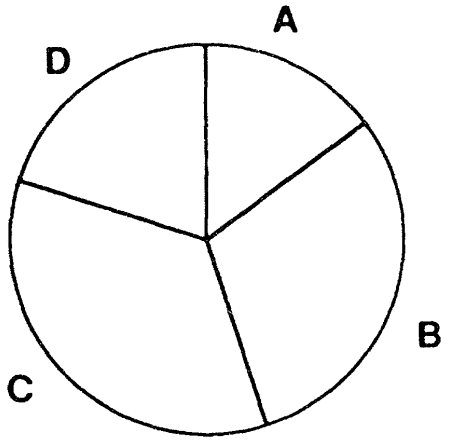
- Introduction
- The coefplot command
  - Basic usage
  - Labels
  - Confidence intervals
  - The recast option
  - Marker labels
  - The at option

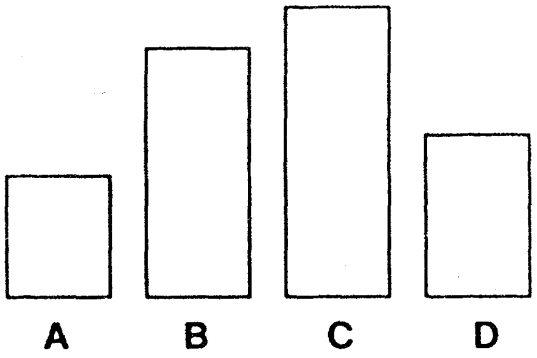
Introduction

- Statistical estimates such as coefficients from regression models are often presented as tables in research articles and presentations.
- However, graphs can me 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 spacial 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).





A	B	C	D
15	30	35	20

(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 perceptional 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.

Chapin (1924)

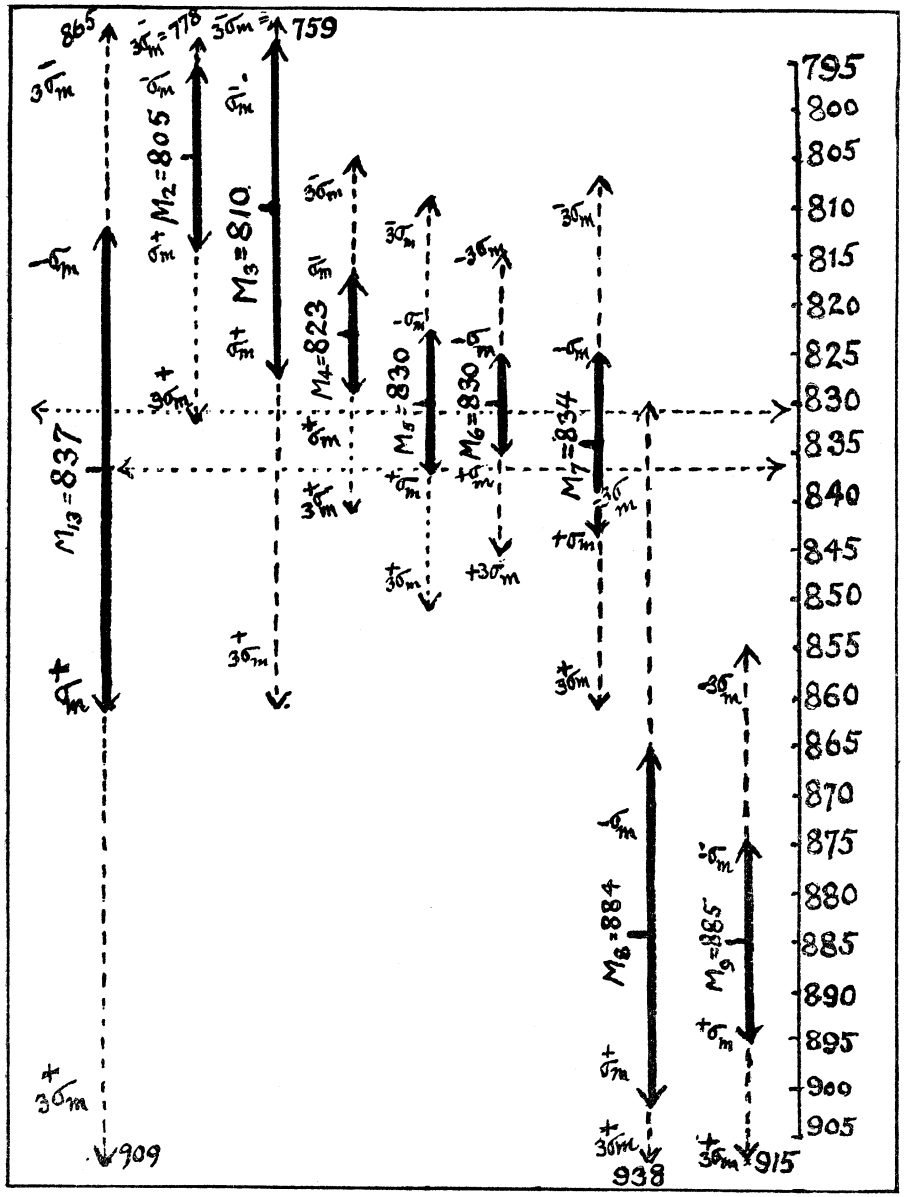


Fig. 3

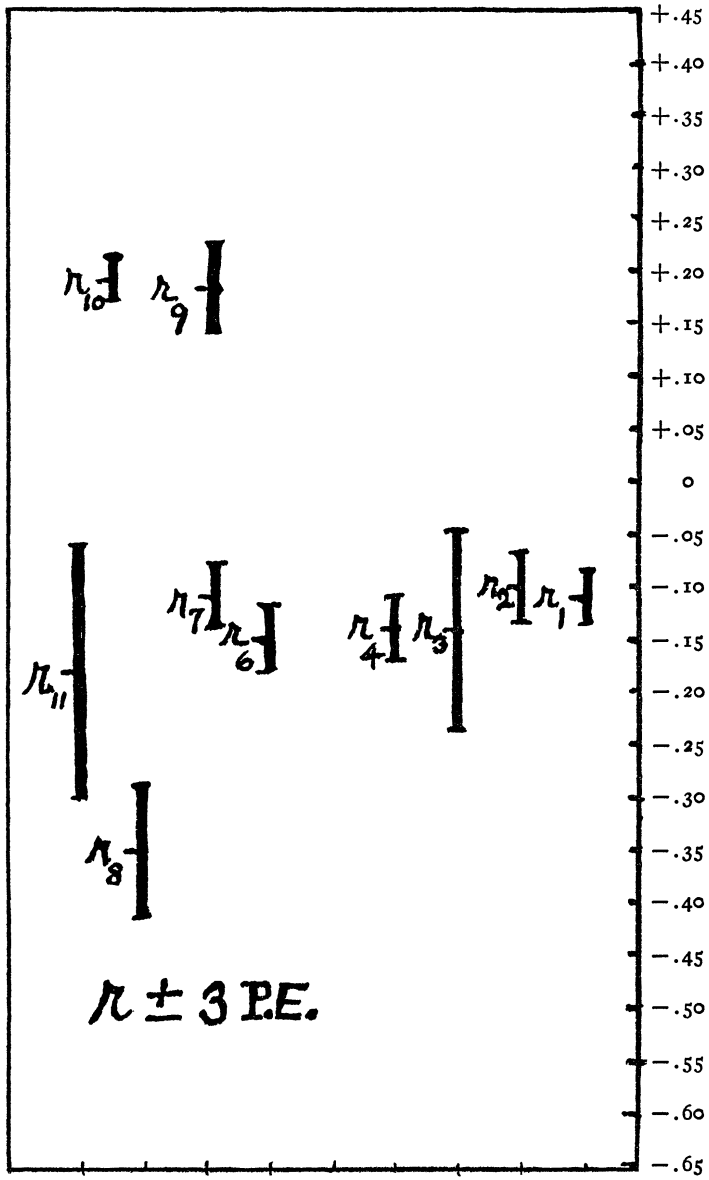


Fig. 4

(Thanks to Nick Cox for pointing me to this and some of the following references.)

Student (1927)

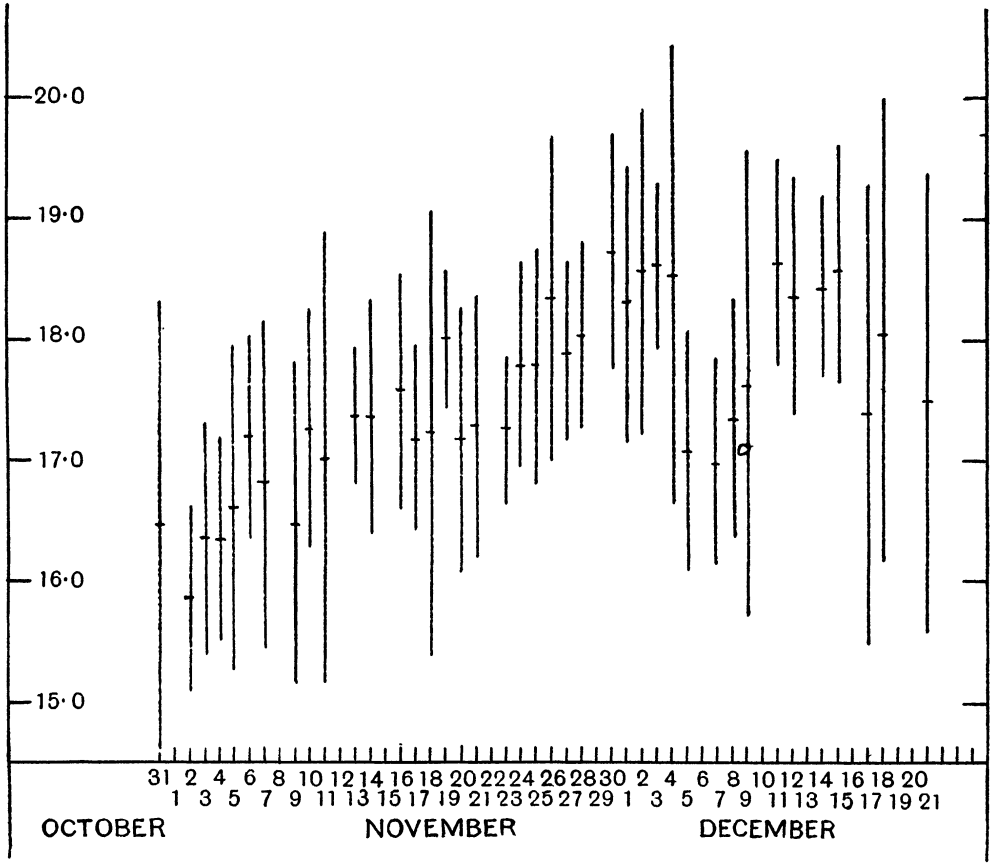


Fig. 3. Means of Daily Analyses with lines showing on each side of the Mean twice the S.D. appropriate to the Number of Analyses made on any given day. The S.D. is derived from the total observations by the formula

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

where a = Average of a Farm,
$\bar{a}$ = Mean of a Day's Analyses,
n = Number of Farms analysed in the Day.

Dice and Leraas (1936)

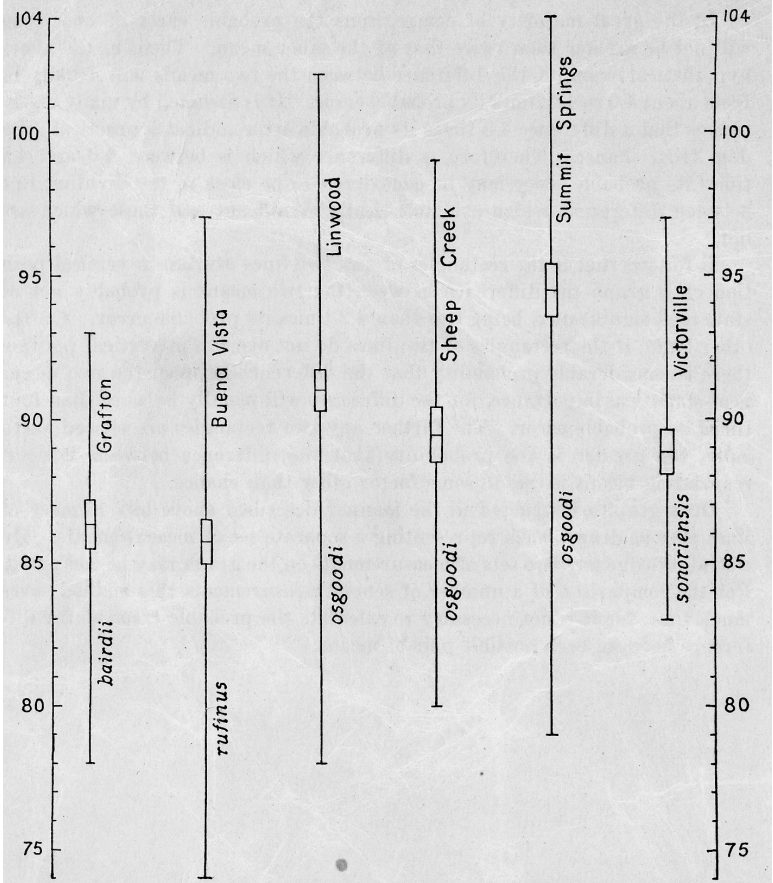
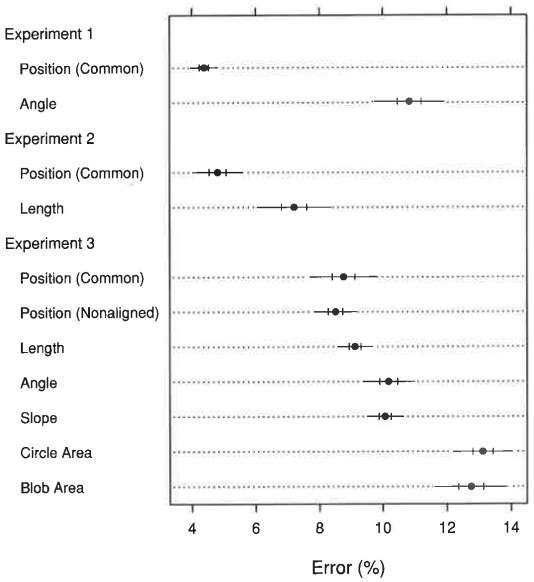


Fig. 1. Graph of body lengths in millimeters of several stocks of *Peromyscus maniculatus*. The length of each line represents the extremes of that set of measurements. The middle crossbar represents the mean. Two other crossbars are placed three times the probable error (= two times the standard error) 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.

Cleveland (1994)



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.

Harrell (2001)

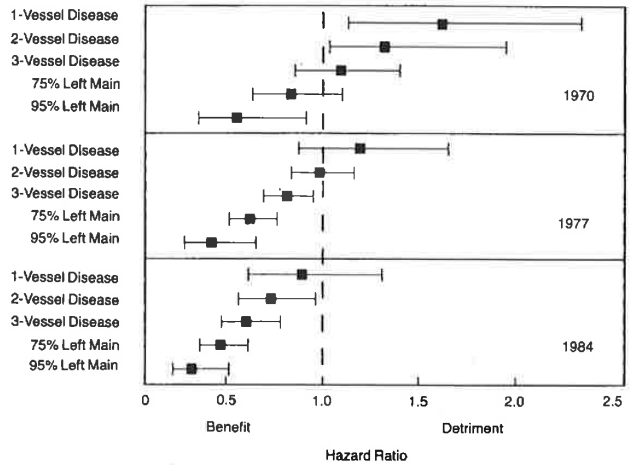


FIGURE 19.12: A display of an interaction between treatment and extent of disease, and between treatment and calendar year of start of treatment. Comparison of medical and surgical average hazard ratios for patients treated in 1970, 1977, and 1984 according to coronary anatomy. Closed squares represent point estimates; bars represent 0.95 confidence limits of average hazard ratios.[61] Reprinted by permission, American Medical Association.

Harrell (2001)

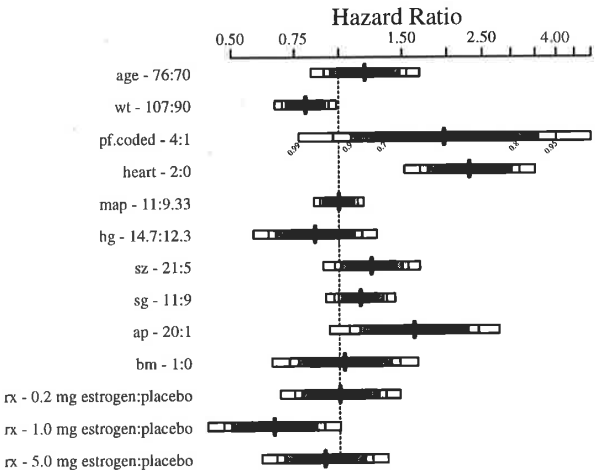


FIGURE 20.4: Hazard ratios and multilevel confidence bars for effects of predictors in model, using default ranges except for ap.

Kastellec and Leoni (2007)

Table 8
Pekkanen, Nyblade and Krauss (2006), table 1: Logit analysis of electoral incentives and LDP post allocation (1996–2003)

Variable	Model 1	Model 2
Block 1: MP Type		
Zombie	0.18 (.22)	0.27 (0.22)
SMD Only	−0.19 (0.22)	−0.19 (0.24)
PR Only	−0.39 (0.18)**	—
Costa Rican in PR	−0.09 (0.29)	—
Block 2: Electoral Strength		
Vote share margin	—	0.005 (0.004)
Margin Squared	—	—
Block 3: Misc Controls		
Urban-Rural Index	0.04 (0.08)	0.04 (0.09)
No Factional Membership	−0.86 (0.26)***	−0.98 (0.31)***
Legal Professional	0.39 (0.29)	−.36 (0.30)
Seniority		
1st Term	−3.76 (0.36)***	−3.66 (0.37)***
2nd Term	−1.61 (0.19)***	−1.59 (0.21)***
4th Term	−0.34 (0.19)**	−0.45 (0.21)***
5th Term	−1.17 (0.22)***	−1.24 (0.24)***
6th Term	−1.15 (0.22)***	−1.04 (0.24)***
7th Term	−1.52 (0.25)***	−1.83 (0.29)***
8th Term	−1.66 (0.28)***	−1.82 (0.32)***
9th Term	−1.34 (0.32)***	−1.21 (0.33)***
10th Term	−2.89 (0.48)***	−2.77 (0.49)***
11th Term	−1.88 (0.43)***	−1.34 (0.46)***
12th Term	−1.08 (0.41)***	−0.94 (0.49)**
Constant	.020 (.20)	0.13 (0.26)
Log-likelihood	−917.24	−764.77
N	1895	1574

Notes: Dependent Variables: 1 if MP holds a post of minister, vice minister, PARC, or HoR Committee Chair.

Base categories: SMD dual-listed, 3rd term. Excluded observations: senior MPs that held no post (> 12 terms, PR-Only MPs in Model 2).

*p < .10, **p < .05, ***p < .001.

Figure 7
Using parallel dot plots with error bars to present two regression models.

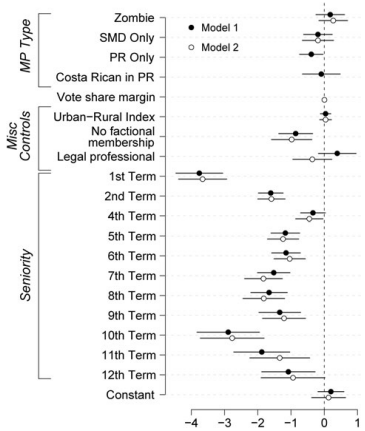


Table 1 from Pekkanen et al. 2006 displays two logistic regression models that examine the allocation of posts in the LDP party in Japan. We turn the table into a graph, and present the two models by plotting parallel lines for each of them grouped by coefficients. We differentiate the models by plotting different symbols for the point estimates: filled (black) circles for Model 1 and empty (white) circles for Model 2.

Introduction

- 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`.

Introduction

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

. regress price mpg trunk length turn

      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  4,    69) =    5.79
       Model |  159570047     4  39892511.8           Prob > F      =  0.0004
    Residual |  475495349    69  6891236.94           R-squared     =  0.2513
-------------+------------------------------           Adj R-squared =  0.2079
       Total |  635065396    73  8699525.97           Root MSE      =  2625.1

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

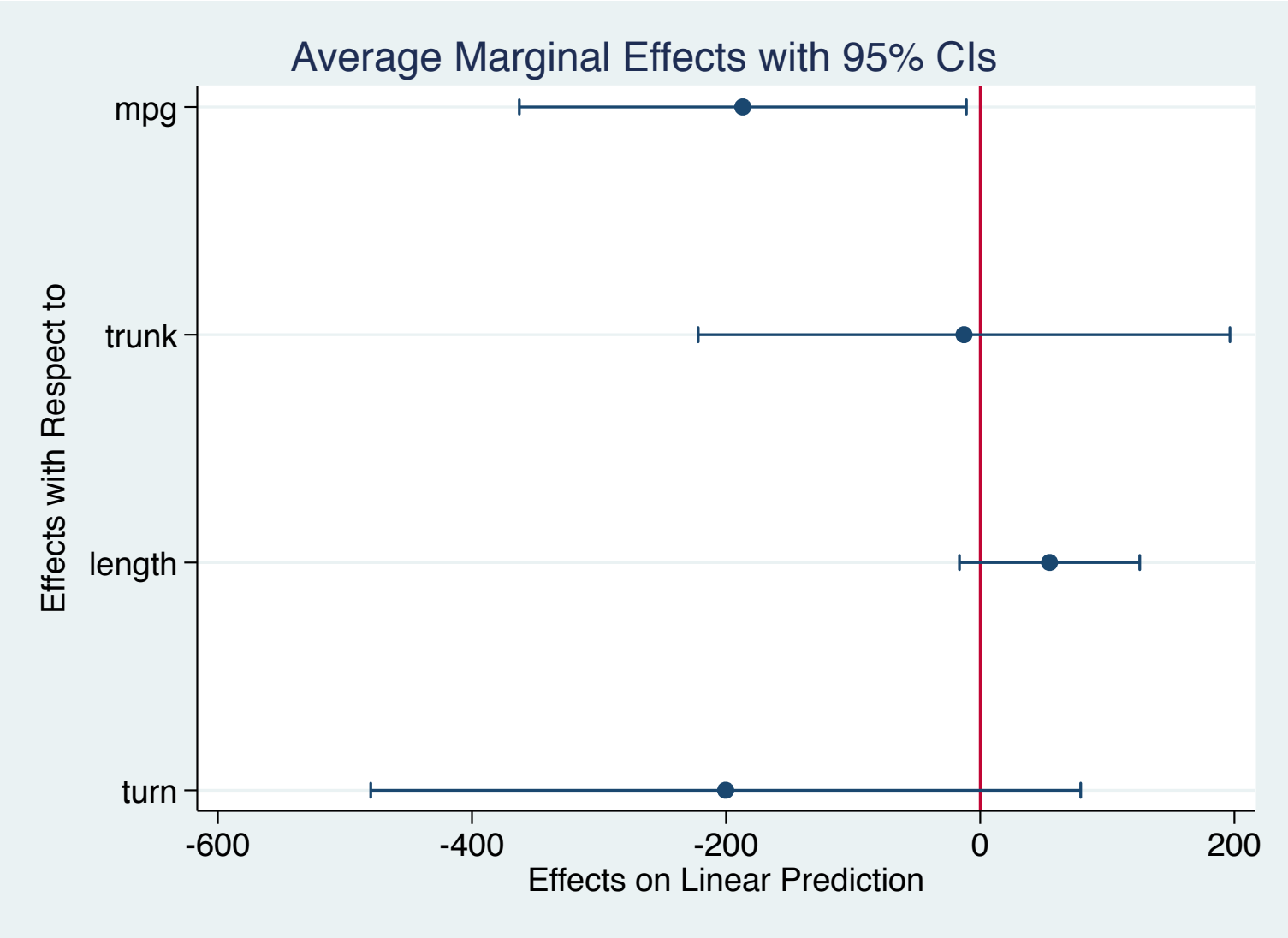
Introduction

```
. 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
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
mpg	-186.8417	88.17601	-2.12	0.038	-362.748	-10.93533
trunk	-12.72642	104.8785	-0.12	0.904	-221.9534	196.5005
length	54.55294	35.56248	1.53	0.130	-16.39227	125.4981
turn	-200.3248	140.0166	-1.43	0.157	-479.6502	79.00066

Introduction

```
. marginsplot, horizontal xline(0) yscale(reverse) recast(scatter)
  Variables that uniquely identify margins: _deriv
```



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

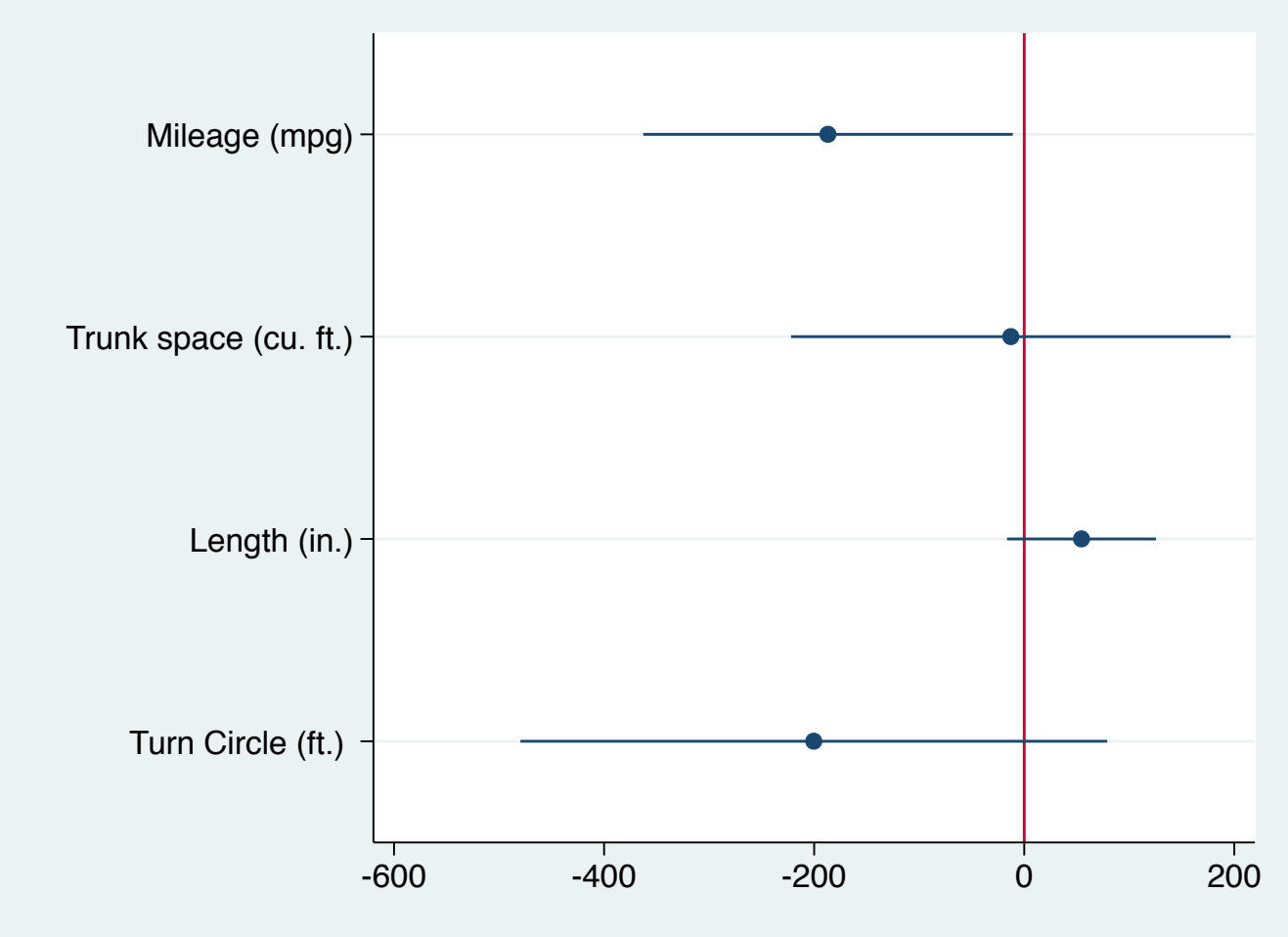
- Some of `coefplot`'s functionality overlaps with the possibilities offered by `marginsplot`, but `coefplot` goes much beyond:
  - `coefplot` can be applied to the results of any estimation command that posts its results in `e()` and can also be used to plot results that have been collected manually 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

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

where subgraph is defined as

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

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

    model [ \ model \ ... ] [, plotopts ]

and model is

    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

    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                  Description
-------------------------------------------------------------------------------------------------------------------
Main
  omitted                  include omitted coefficients
  baselevels               include base levels
  b(mspec)                 specify source to be plotted; default is to plot e(b)
  at[(spec)]               get plot positions from e(at), or as specified by spec
  keep(coeflist)           keep specified coefficients
  drop(coeflist)           drop specified coefficients
```

Basic usage: Plotting a single model

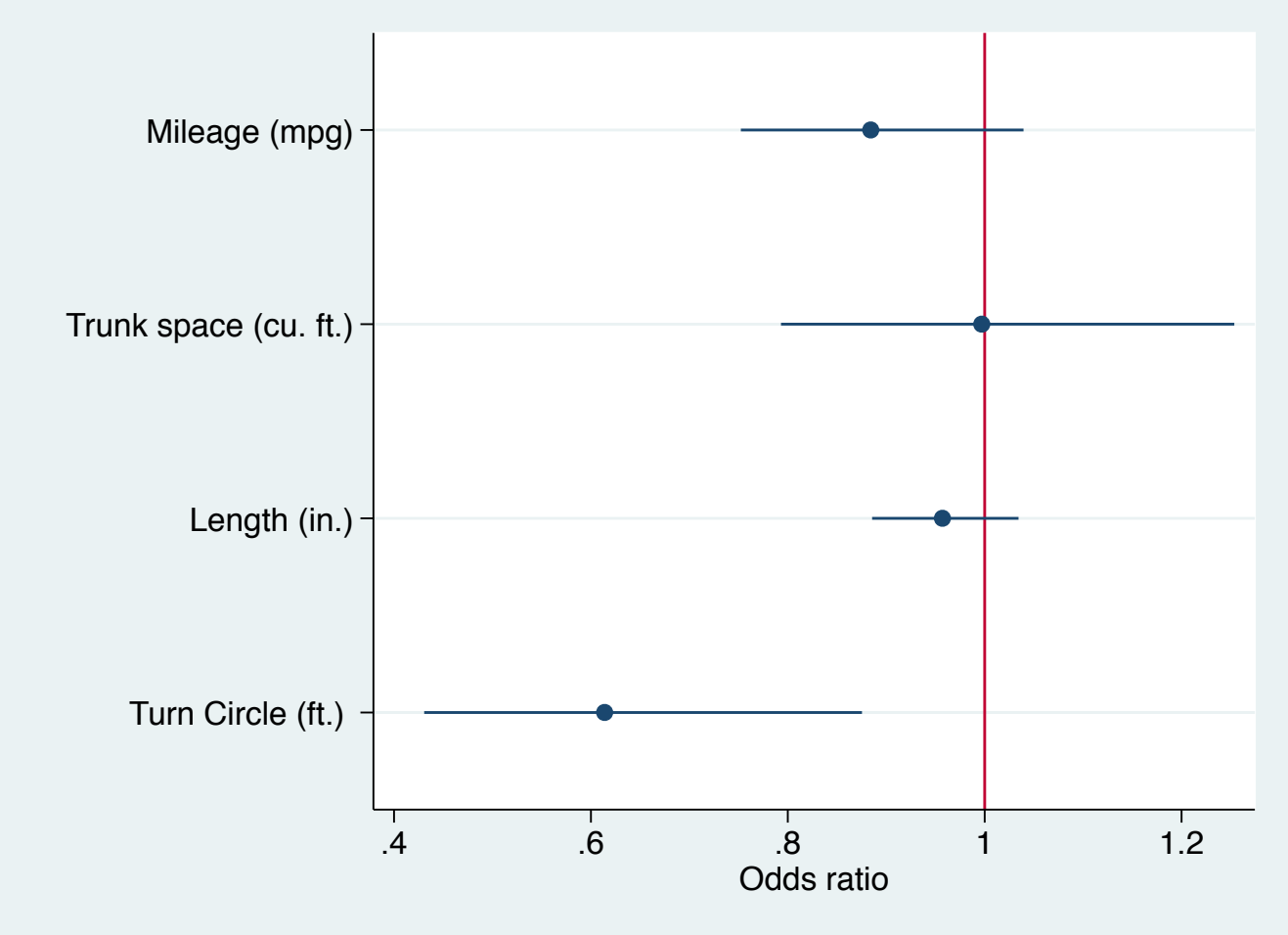
```
. quietly sysuse auto, clear
. quietly regress price mpg trunk length turn
. coefplot, drop(_cons) xline(0)
```



Basic usage: Plotting a single model

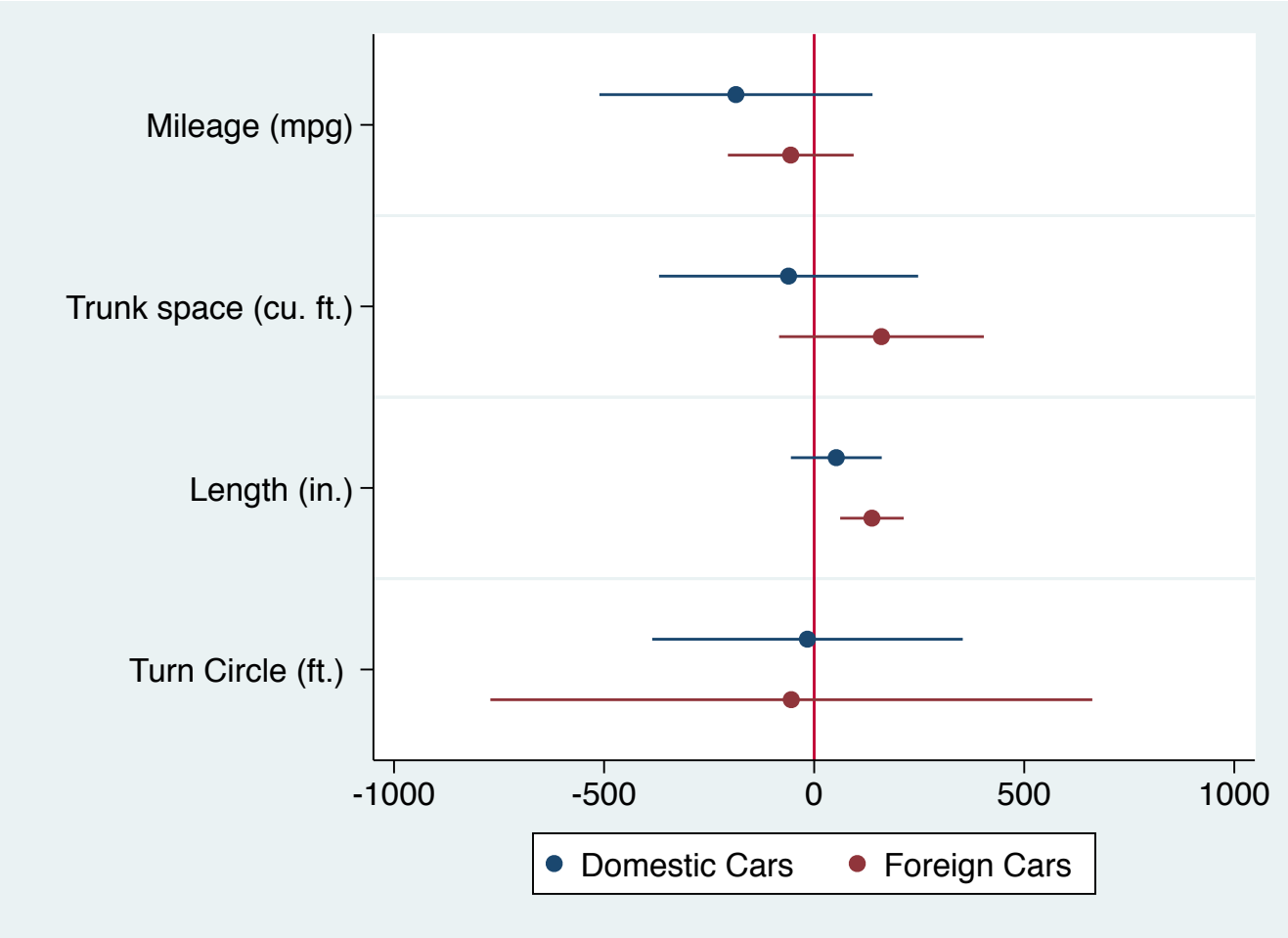
Odds ratios from a logit model (eform option)

```
. quietly logit foreign mpg trunk length turn
. coefplot, drop(_cons) xline(1) eform xtitle(Odds ratio)
```



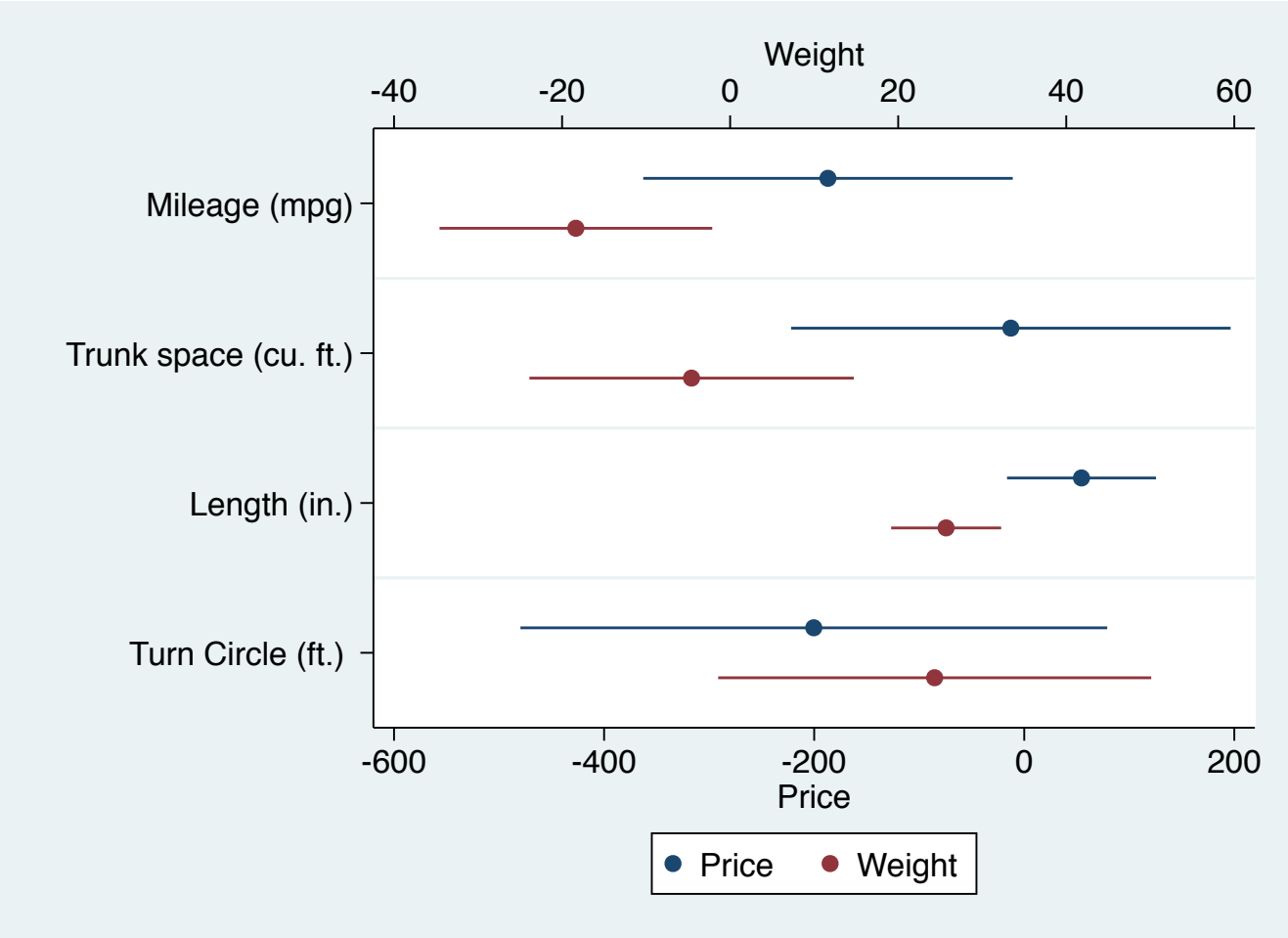
Basic usage: Plotting multiple models

```
. 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)
```



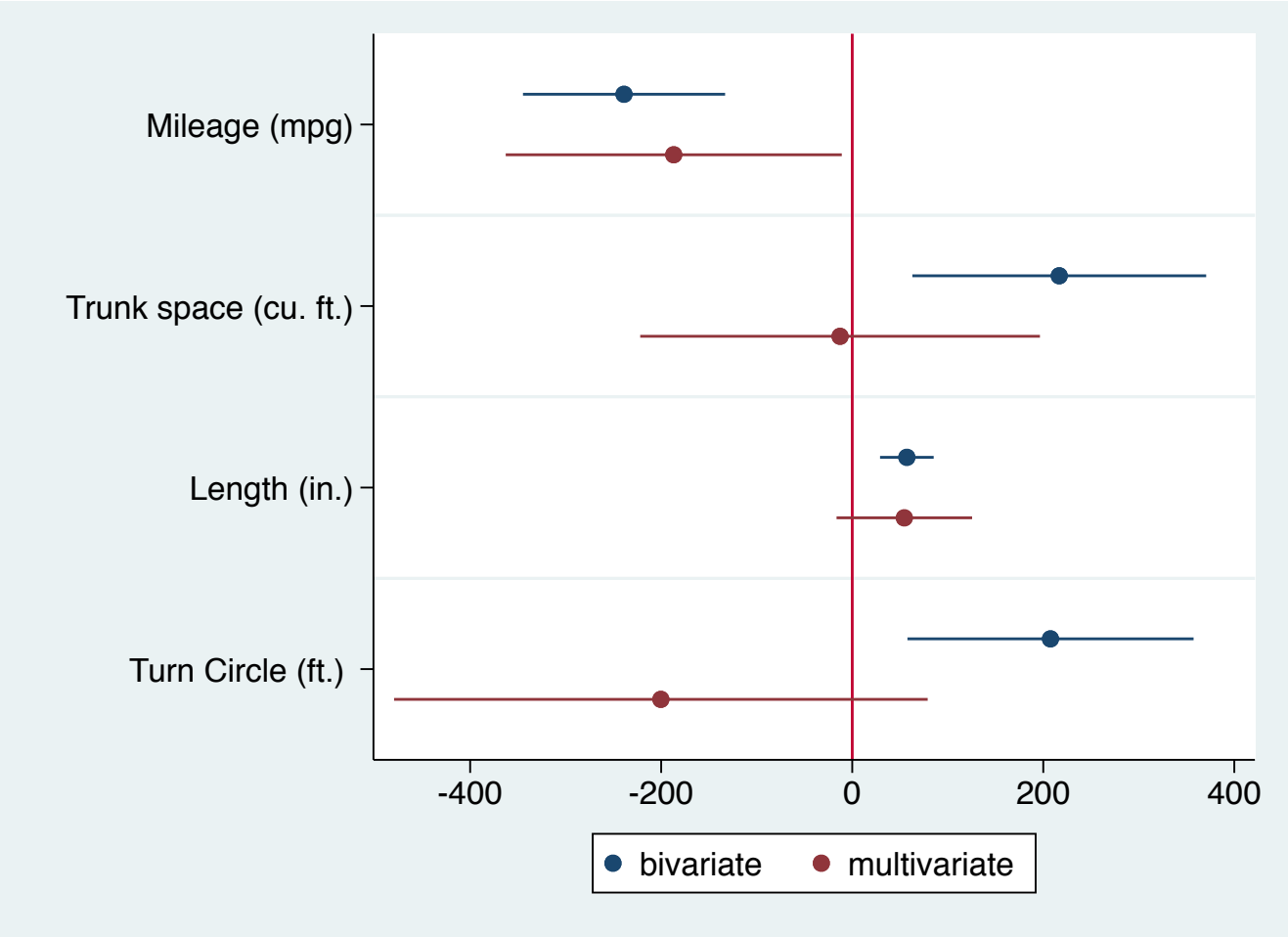
Basic usage: Multiple models on separate scales

```
. 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

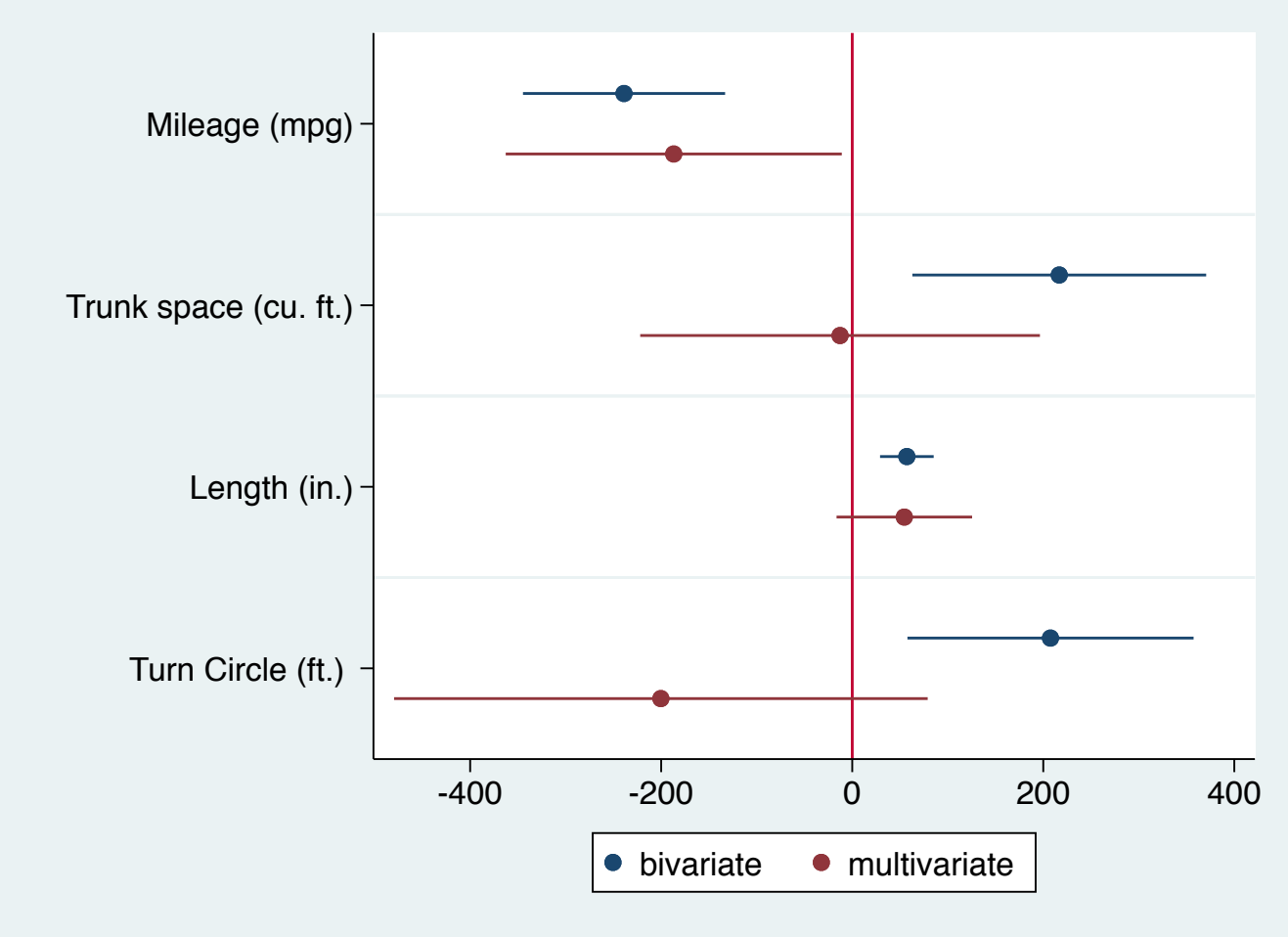
```
. quietly eststo multivariate: regress price mpg trunk length turn
. foreach var in mpg trunk length turn {  // (using -eststo- from -estout- pkg for sake of brevity)
  2.     quietly eststo b_`var´: regress price `var´
  3. }
. coefplot (b_mpg \ b_trunk \ b_length \ b_turn, label(bivariate)) (multivariate)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



Basic usage: Appending models

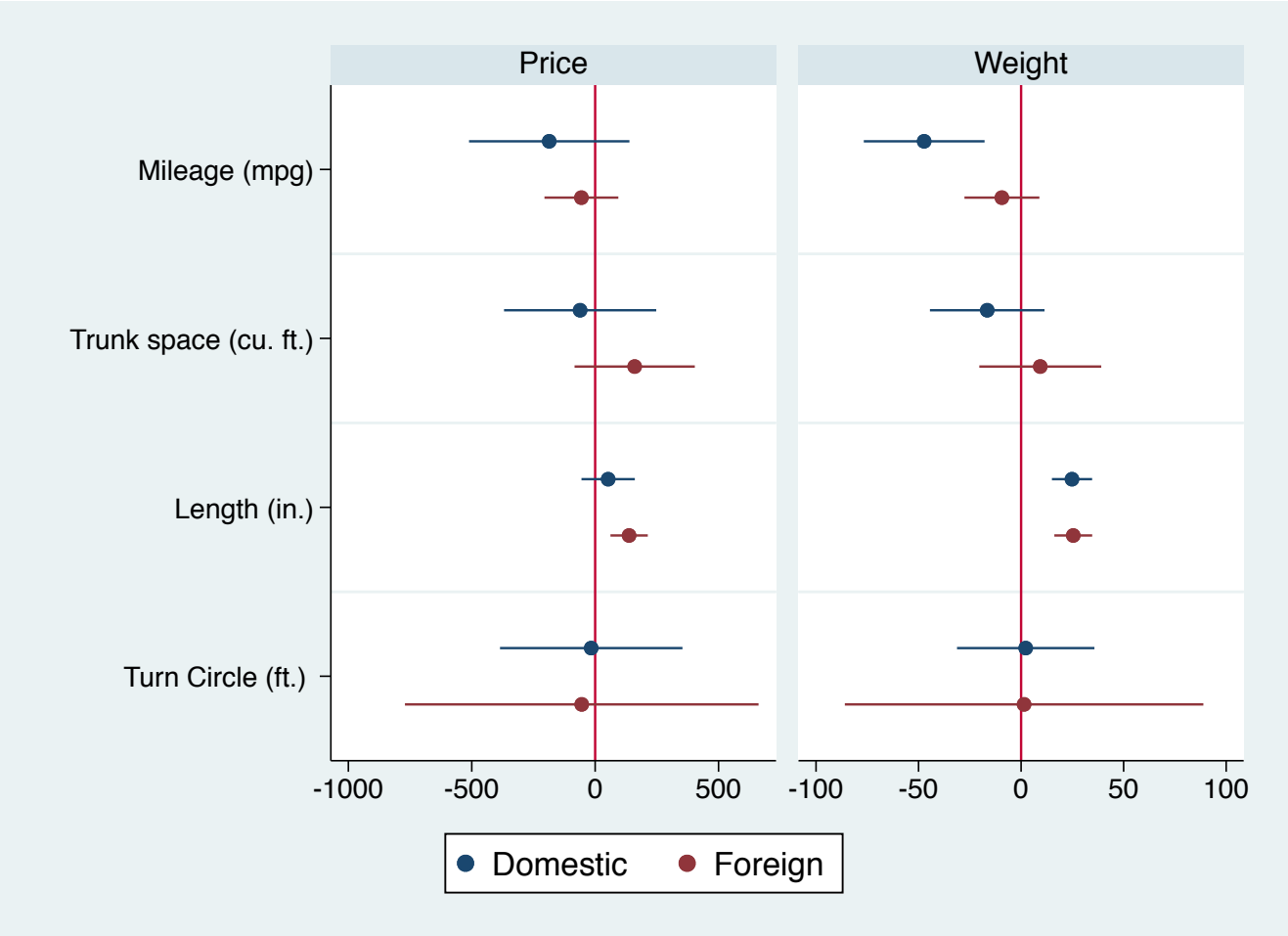
Using wildcard syntax to select the models

```
. coefplot (b_*, label(bivariate)) (multivariate), drop(_cons) xline(0)
```



Basic usage: Subgraphs

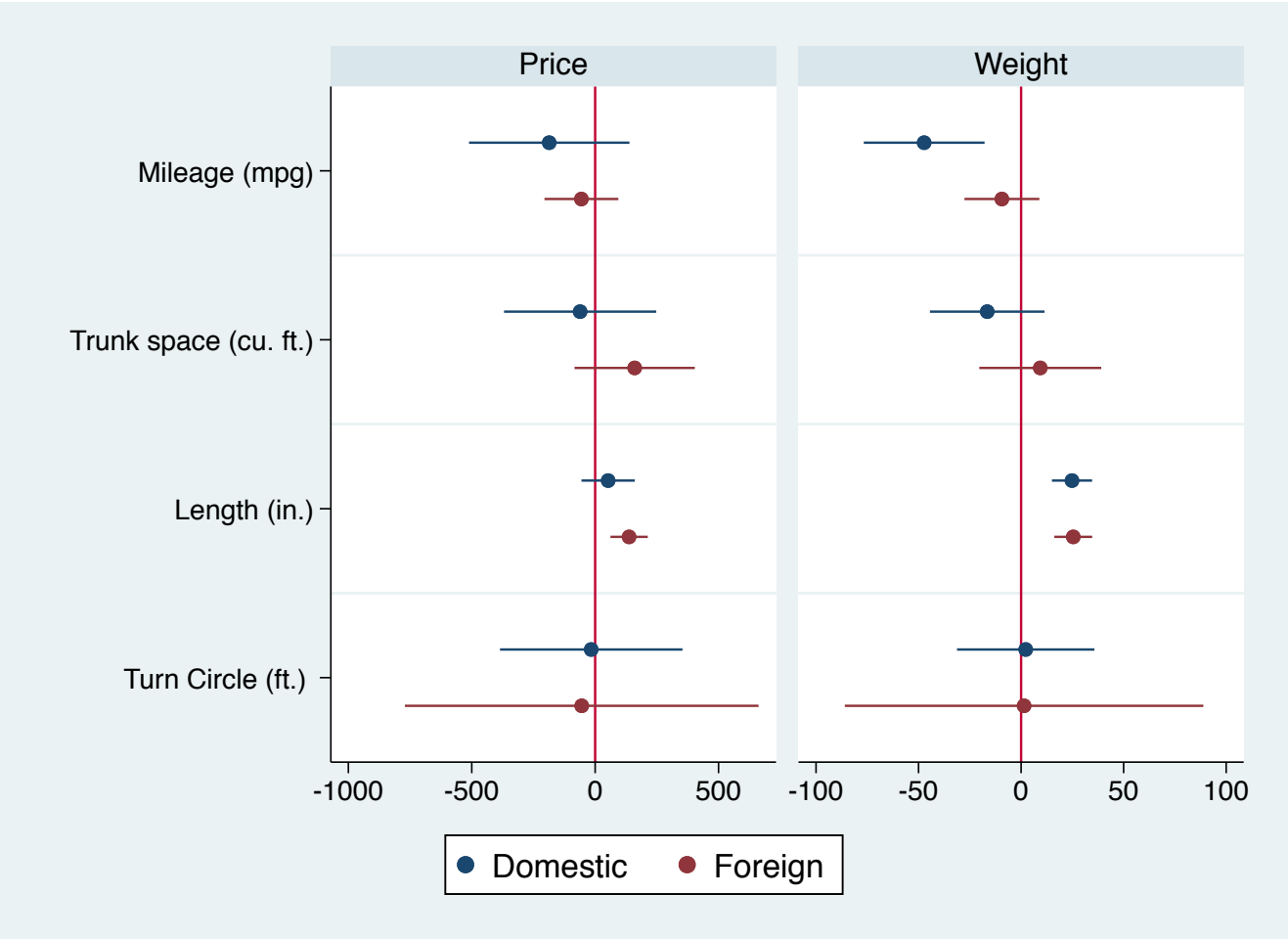
```
. quietly eststo Price_D:  regress price mpg trunk length turn if foreign==0
. quietly eststo Price_F:  regress price mpg trunk length turn if foreign==1
. quietly eststo Weight_D: regress weight mpg trunk length turn if foreign==0
. quietly eststo Weight_F: regress weight mpg trunk length turn if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



Basic usage: Subgraphs

Using wildcard syntax

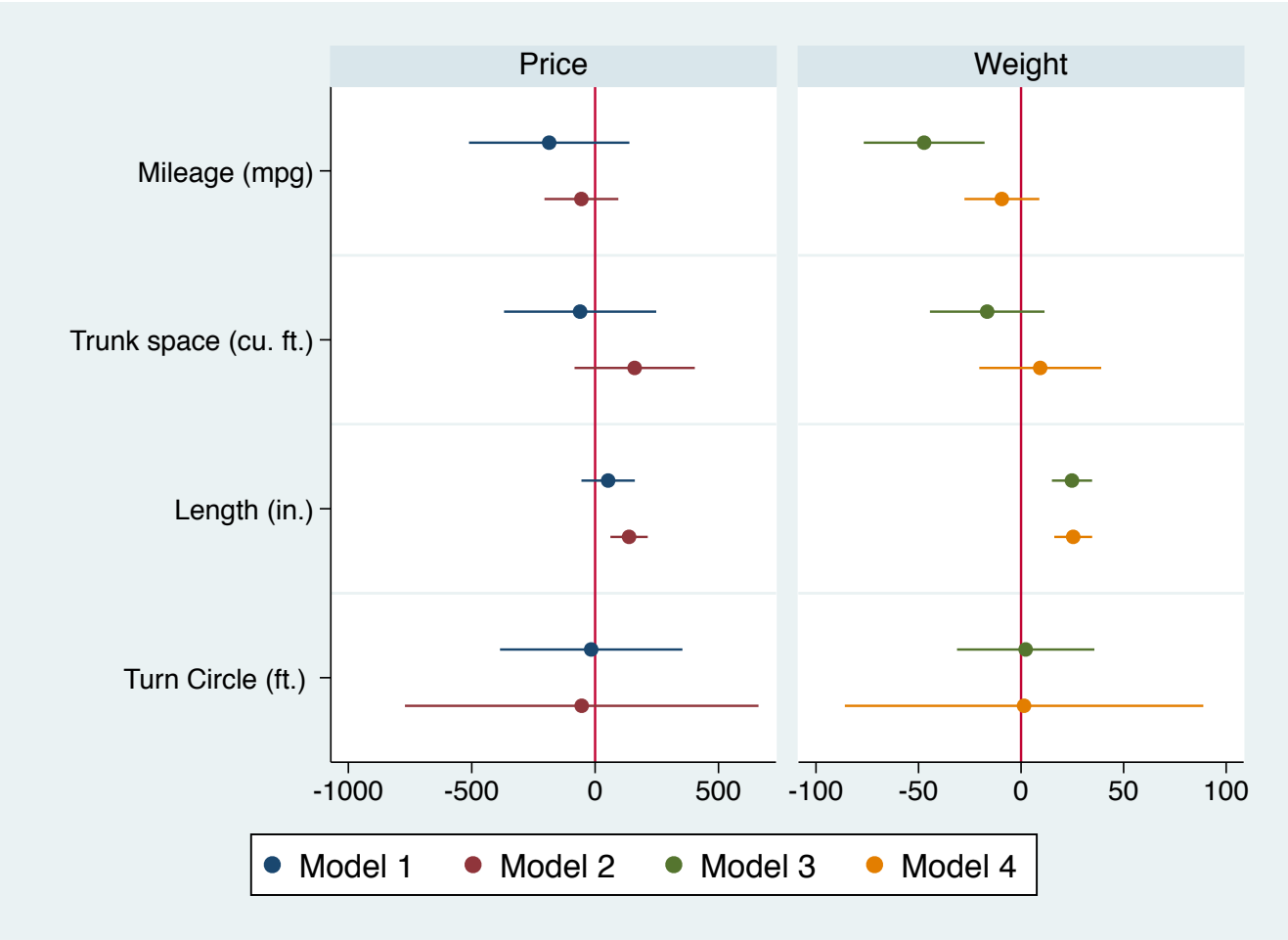
```
. coefplot Price* , bylabel(Price)   ///
>      ||  Weight*, bylabel(Weight)  ///
>      ||, plotlabels(Domestic Foreign) drop(_cons) xline(0) byopts(xrescale)
```



Basic usage: Subgraphs

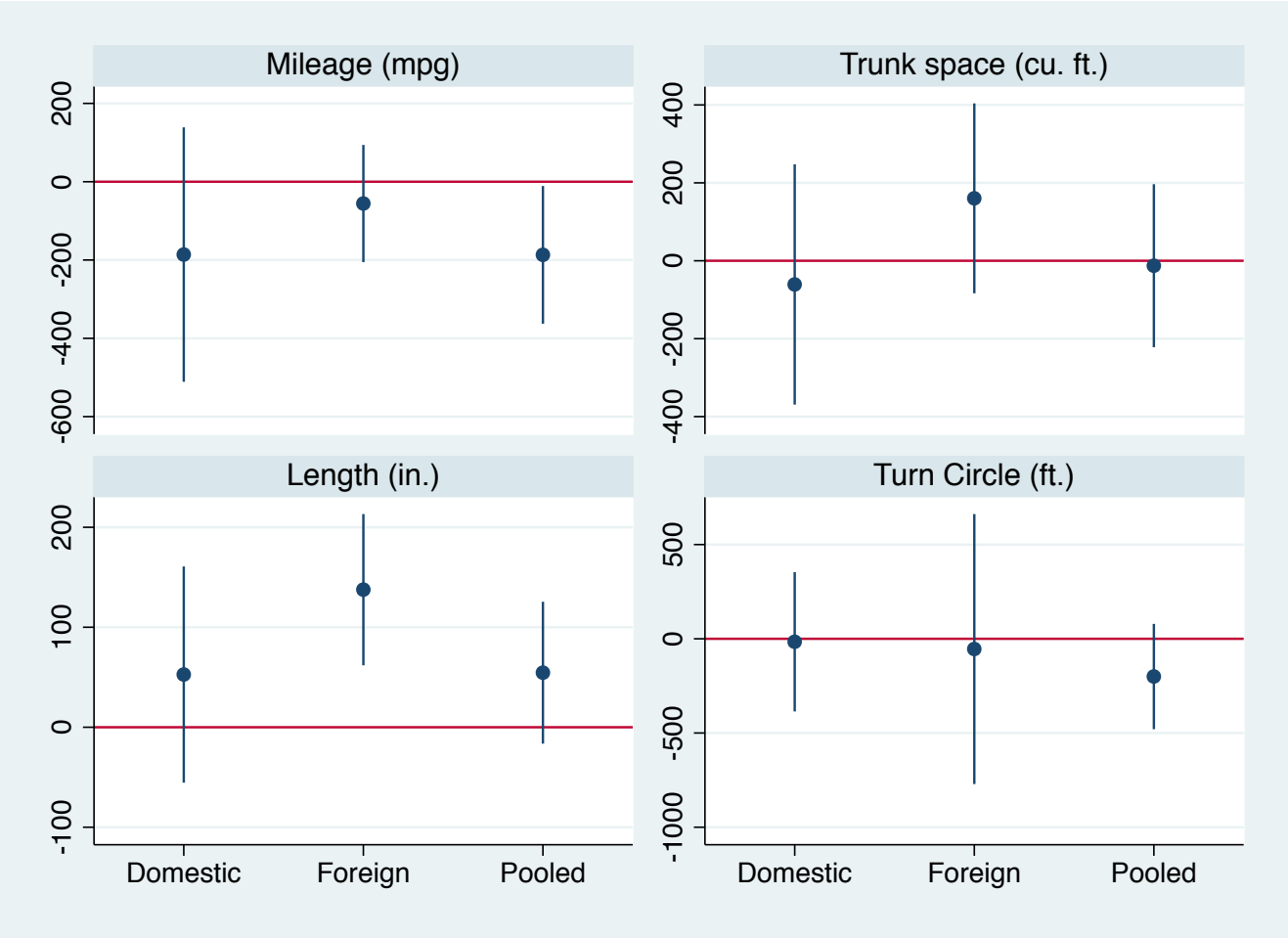
Different plot styles across subgraphs

```
. 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))
```



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)
```

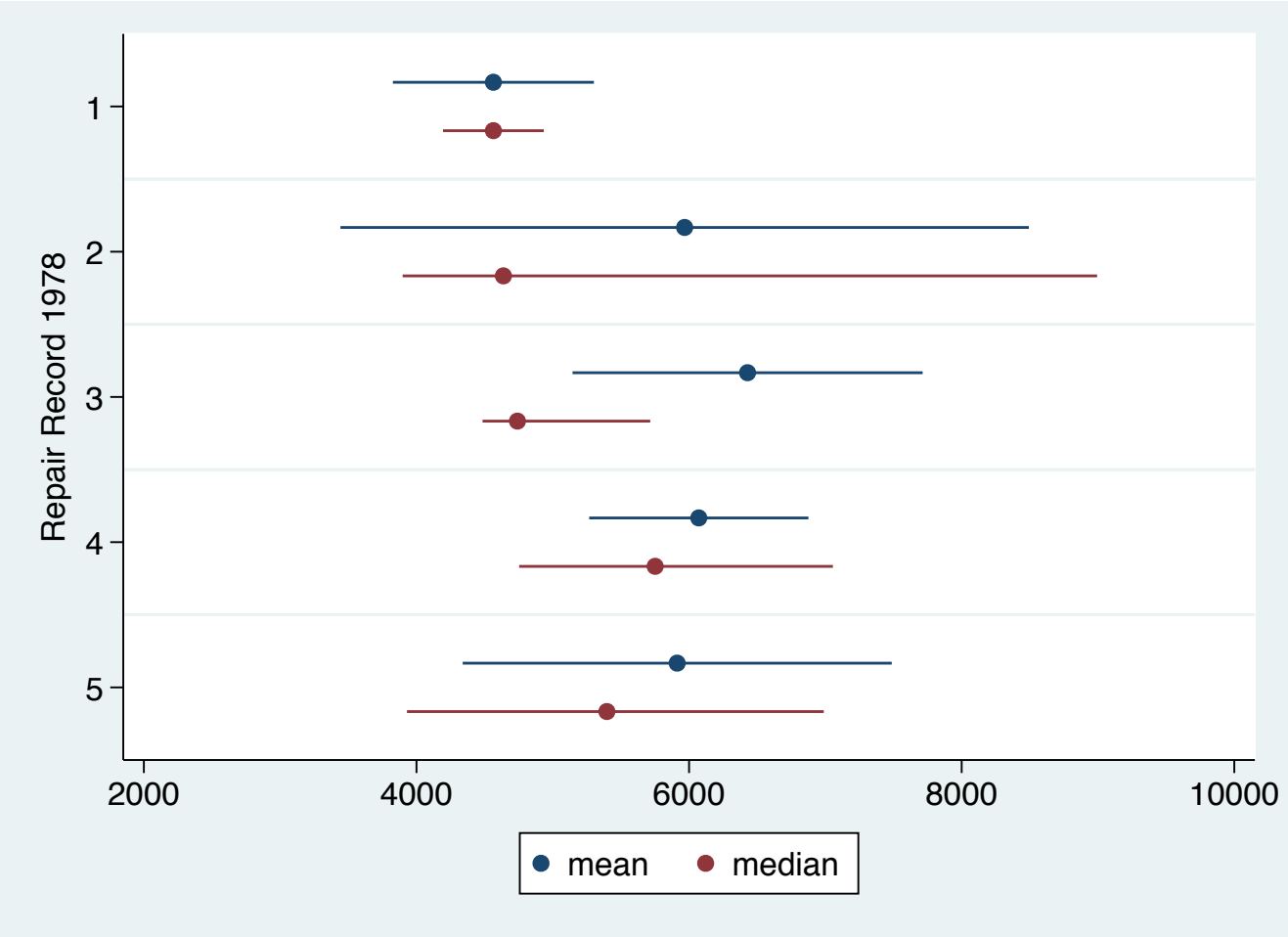


Basic usage: Plotting results from matrix

```
. matrix median = J(5, 3, .)

. matrix coln median = median ll95 ul95

. matrix rown median = 1 2 3 4 5

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

. matrix list median

median[5,3]
      median       ll95       ul95
1     4564.5       4195       4934
2       4638   3898.525    8993.35
3       4741  4484.8407  5714.9172
4     5751.5  4753.4403  7055.1933
5       5397  3930.5673  6988.0509

. eststo mean: mean price, over(rep78)

Mean estimation                     Number of obs    =      69

            1: rep78 = 1
            2: rep78 = 2
            3: rep78 = 3
            4: rep78 = 4
            5: rep78 = 5

--------------------------------------------------------------
        Over |       Mean   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
price        |
           1 |     4564.5      369.5      3827.174    5301.826
           2 |   5967.625   1265.494      3442.372    8492.878
           3 |   6429.233   643.5995       5144.95    7713.516
           4 |     6071.5   402.9585      5267.409    6875.591
           5 |       5913   788.6821      4339.209    7486.791
--------------------------------------------------------------
```

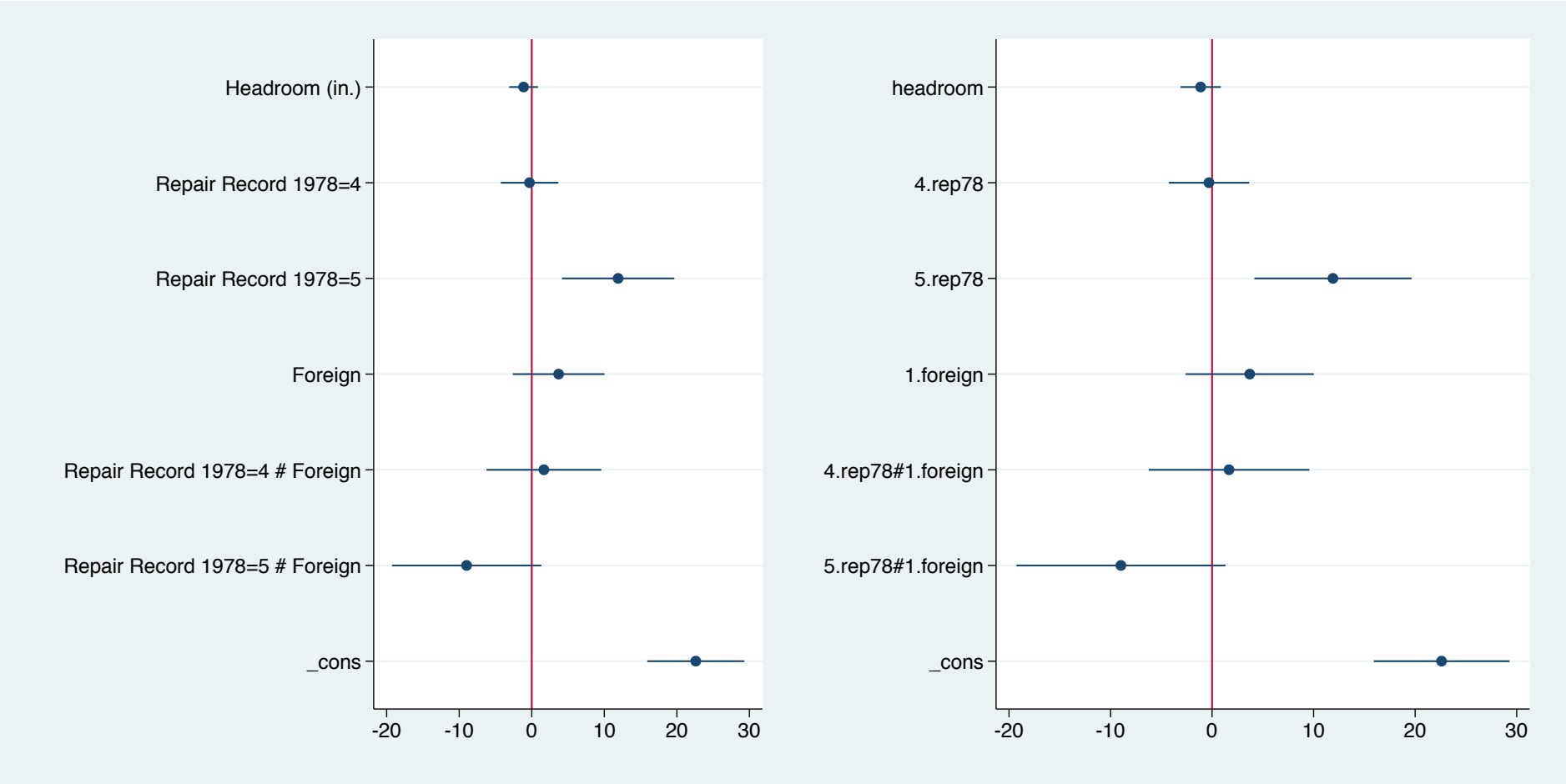
Basic usage: Plotting results from matrix

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



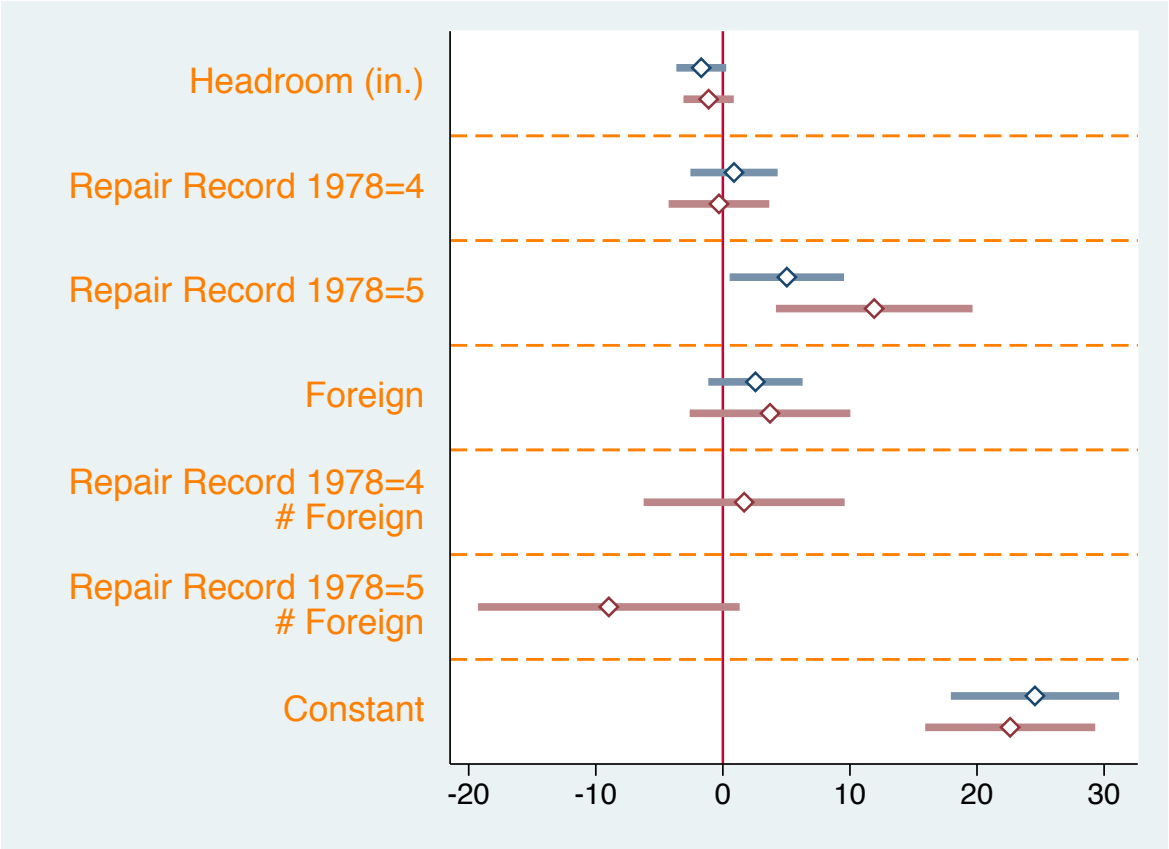
Labels: Factor variables and interactions

```
. 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)
```



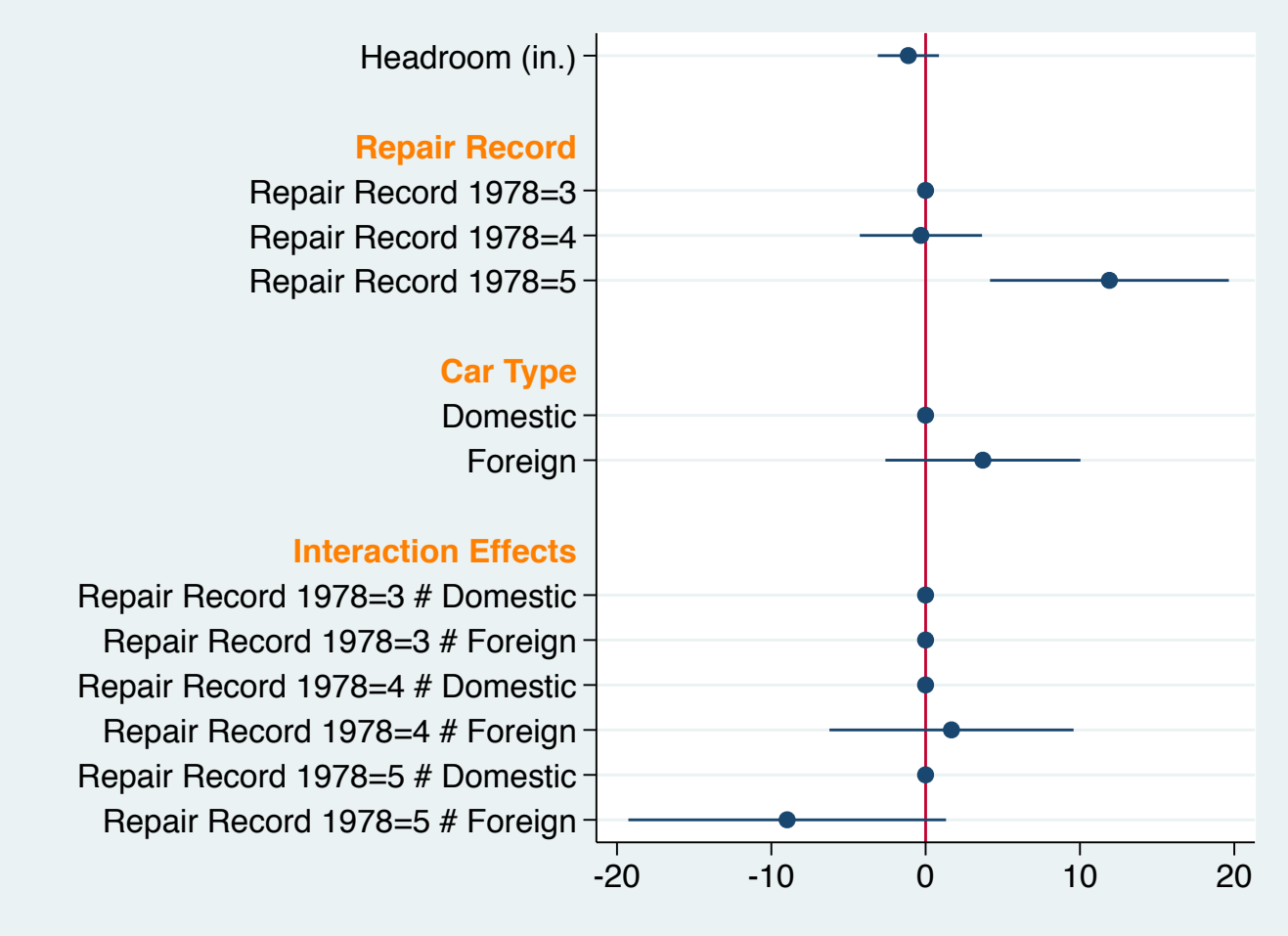
Custom labels, wrapping, grid, offsets, styling

```
. 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))
```



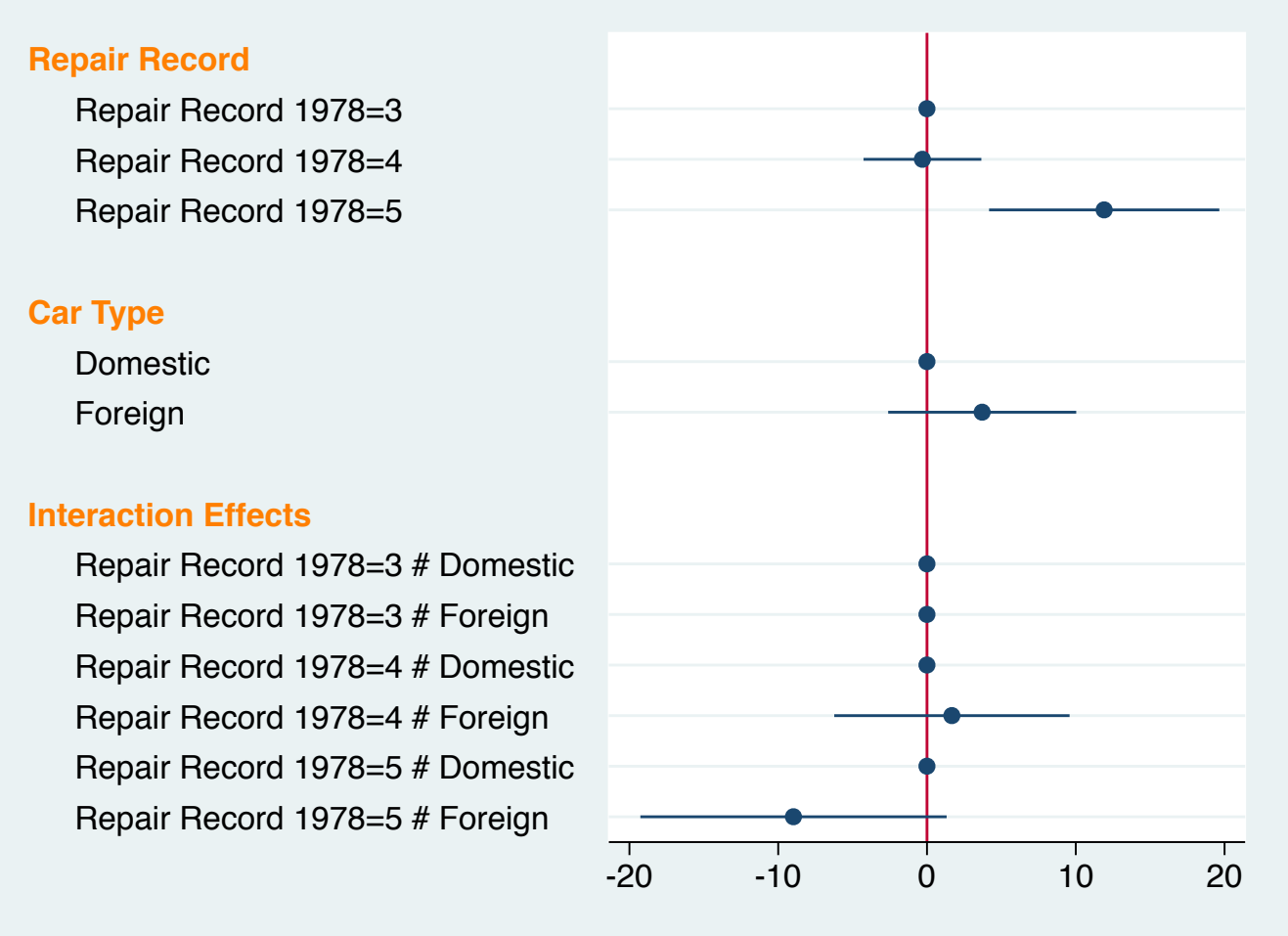
Headings

```
. 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))
```



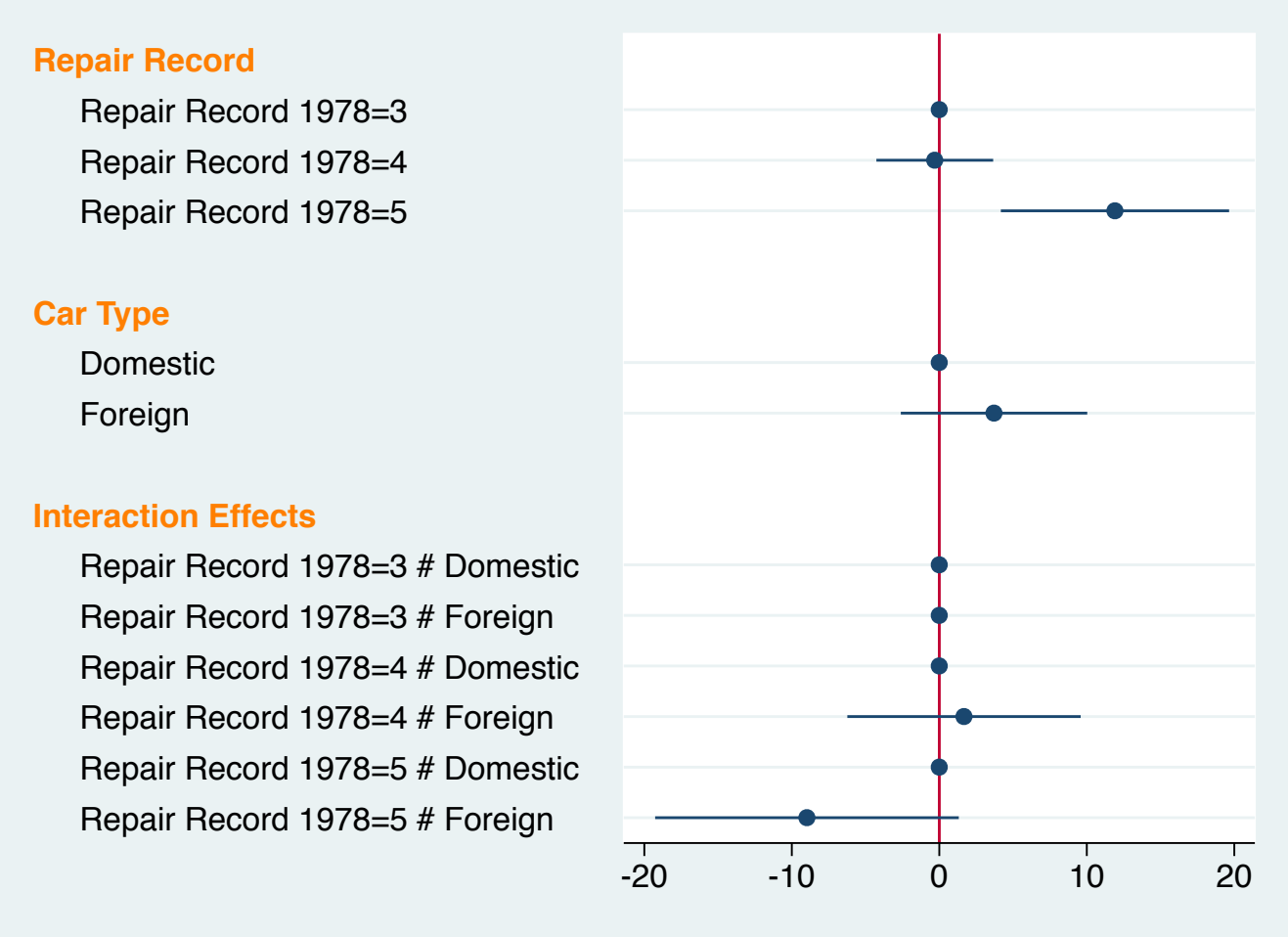
Left-aligned labels (possible, but a bit complicated)

```
. 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))
```



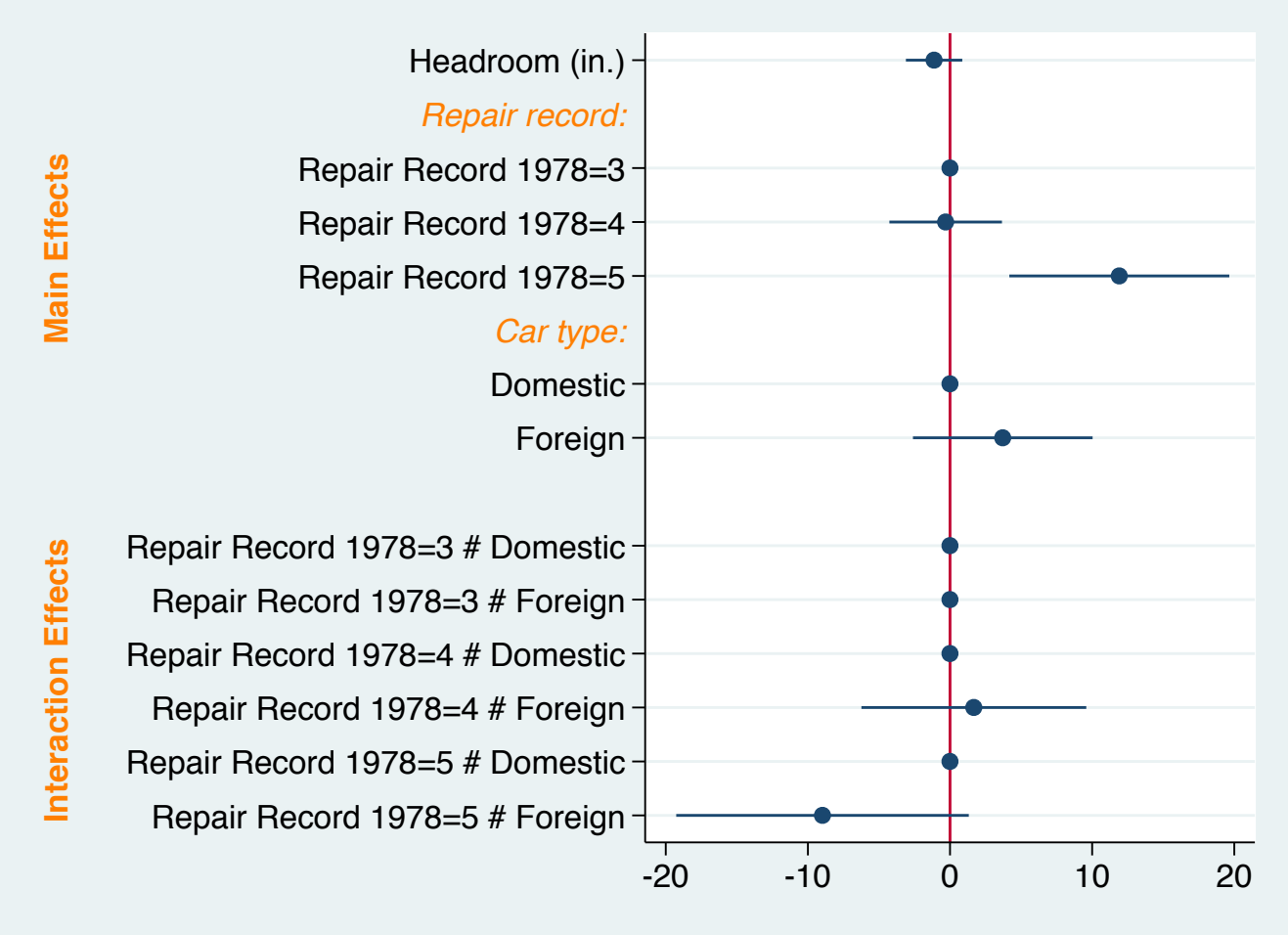
Left-aligned labels (using the graph editor)

```
. 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
```



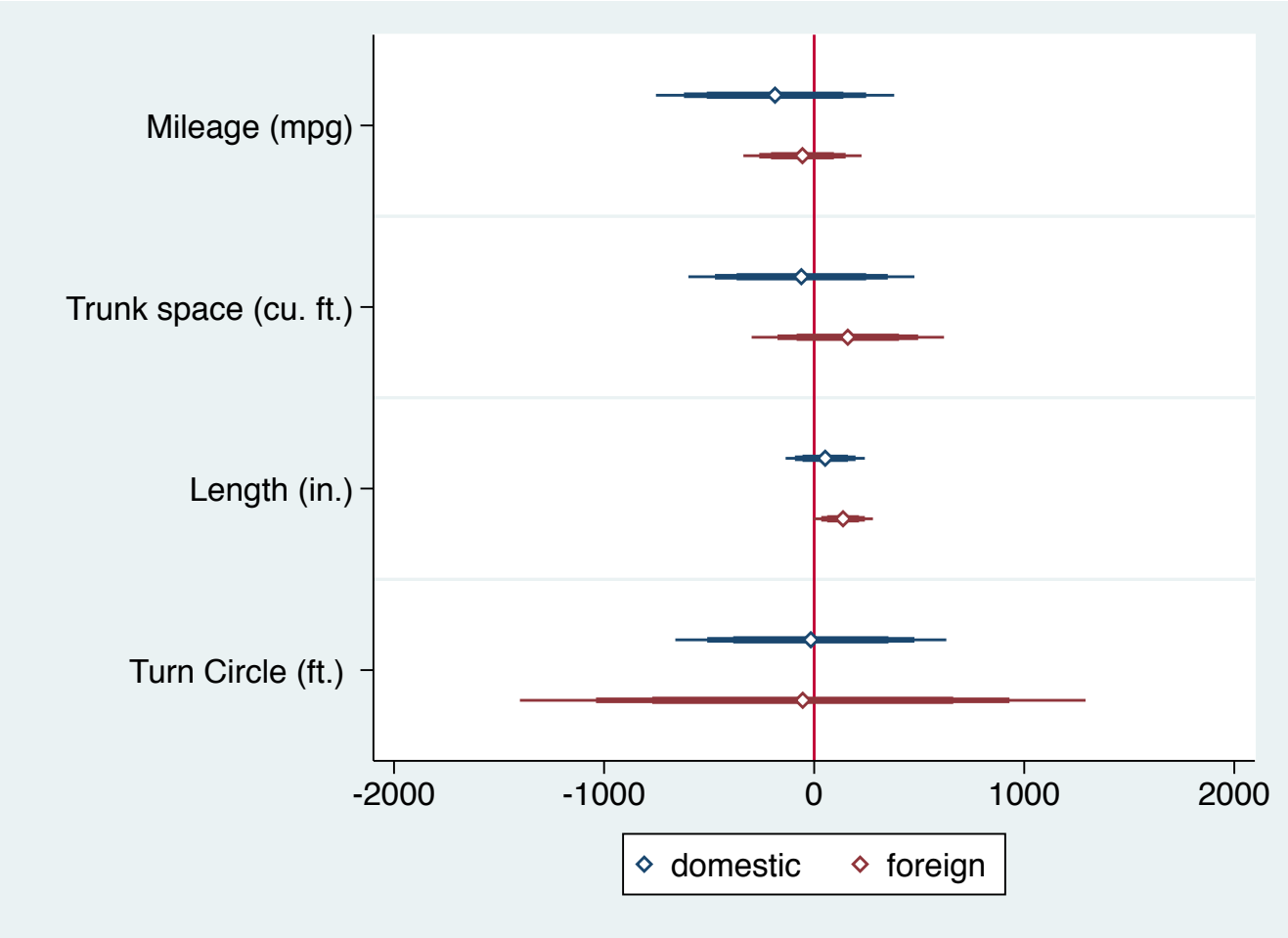
Headings and groups

```
. 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))
```



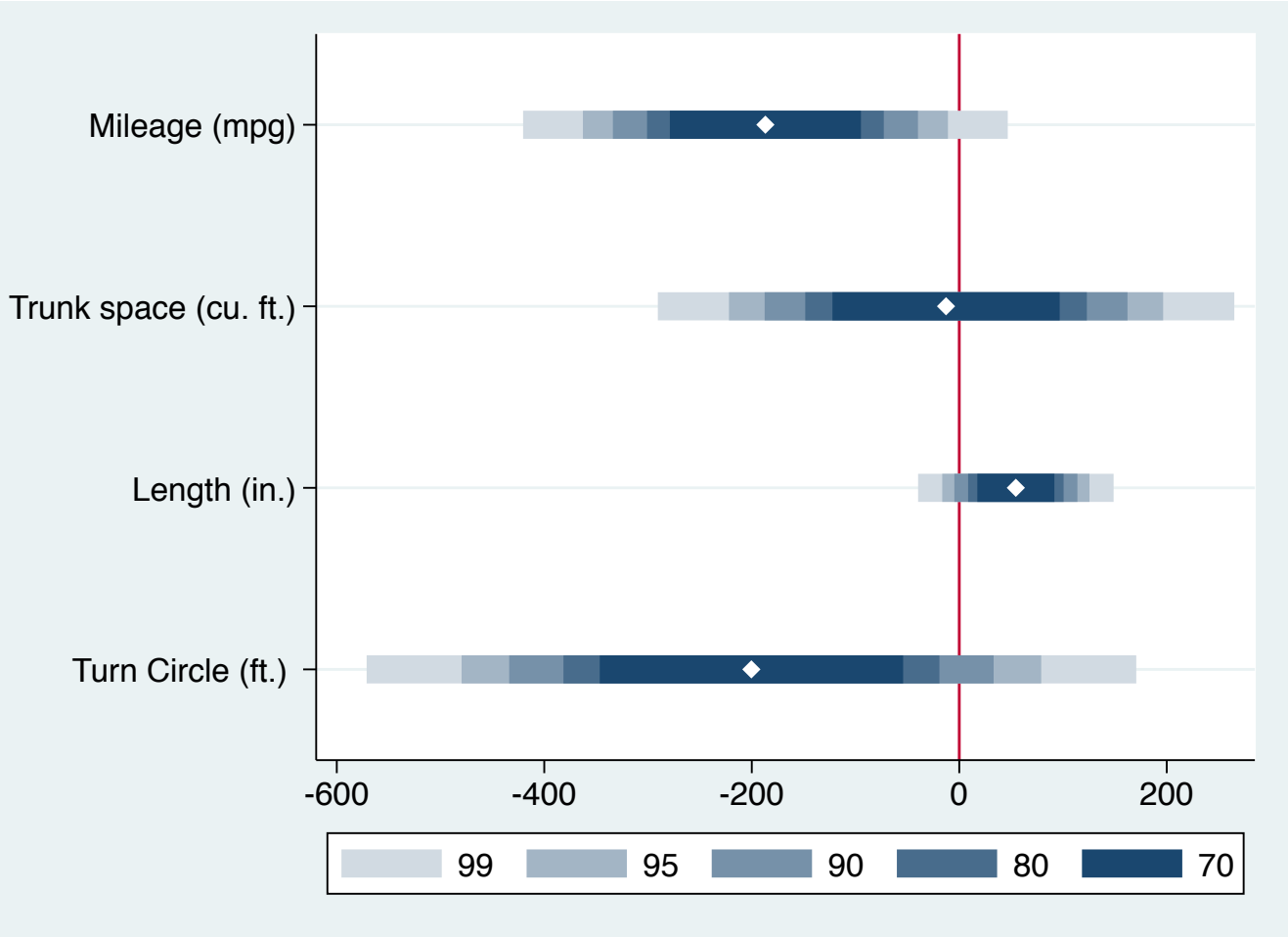
Confidence intervals: Multiple levels

```
. 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)
```



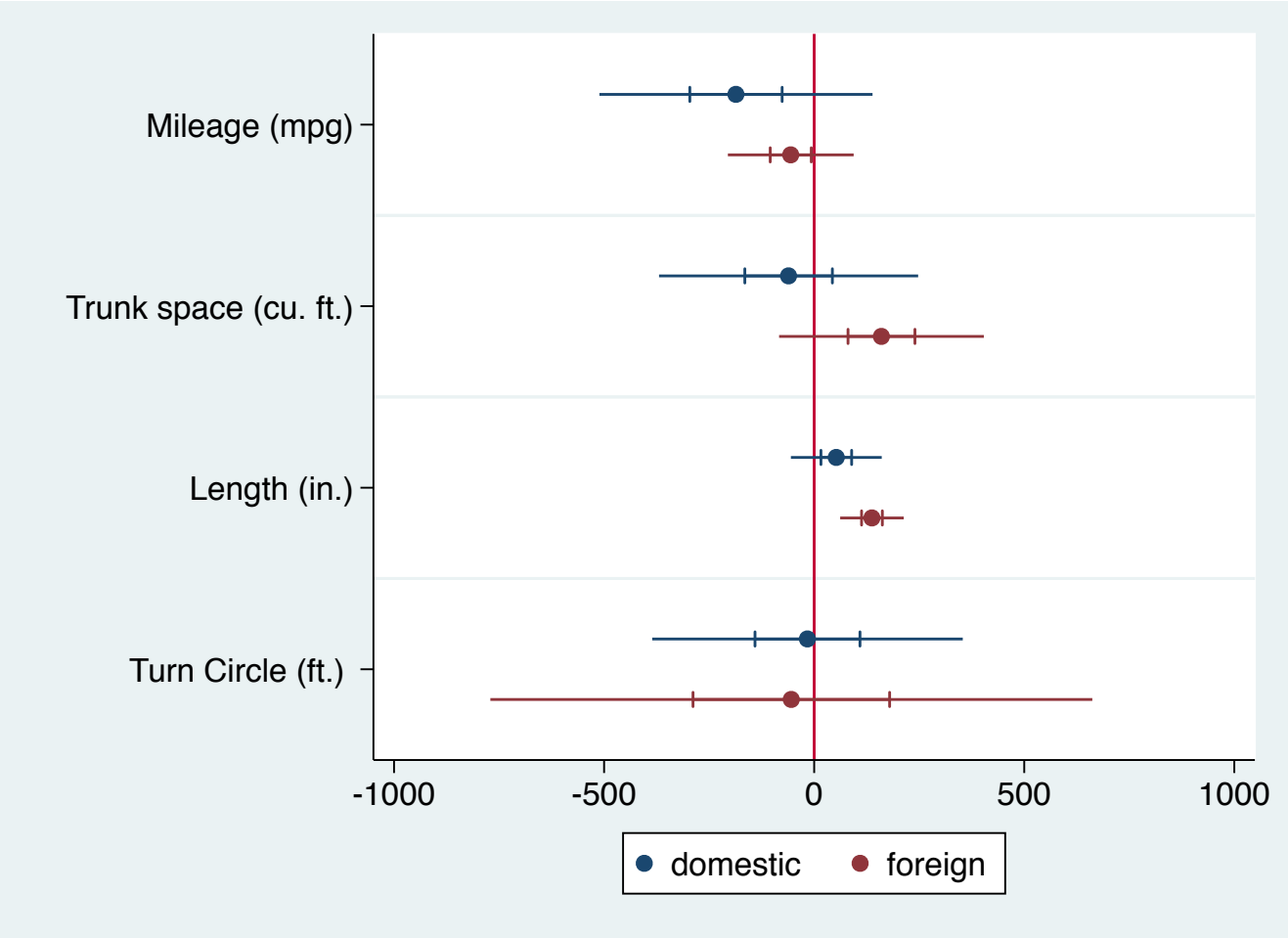
Confidence intervals: Harrell style

```
. 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))
```



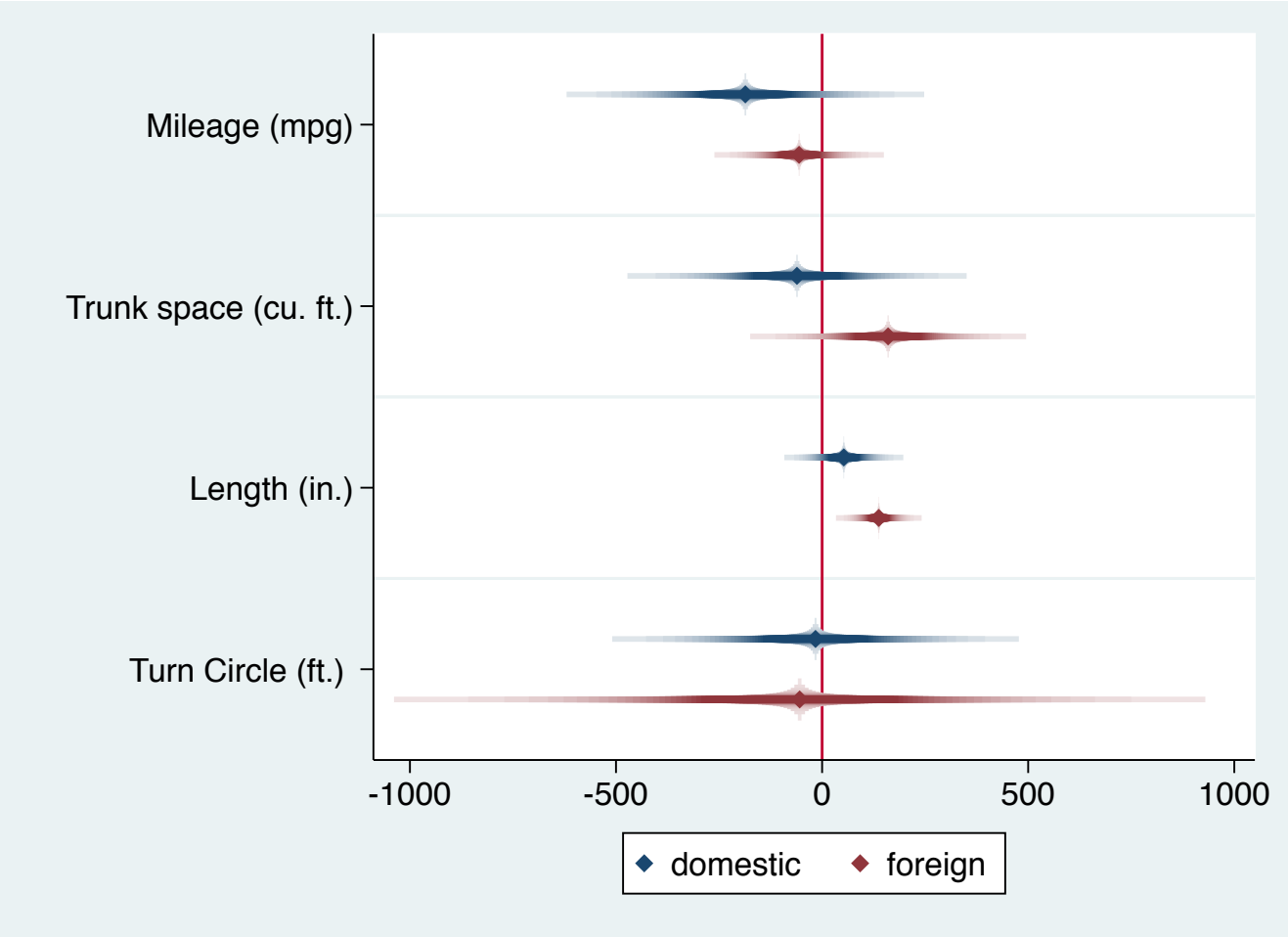
Confidence intervals: Cleveland style

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



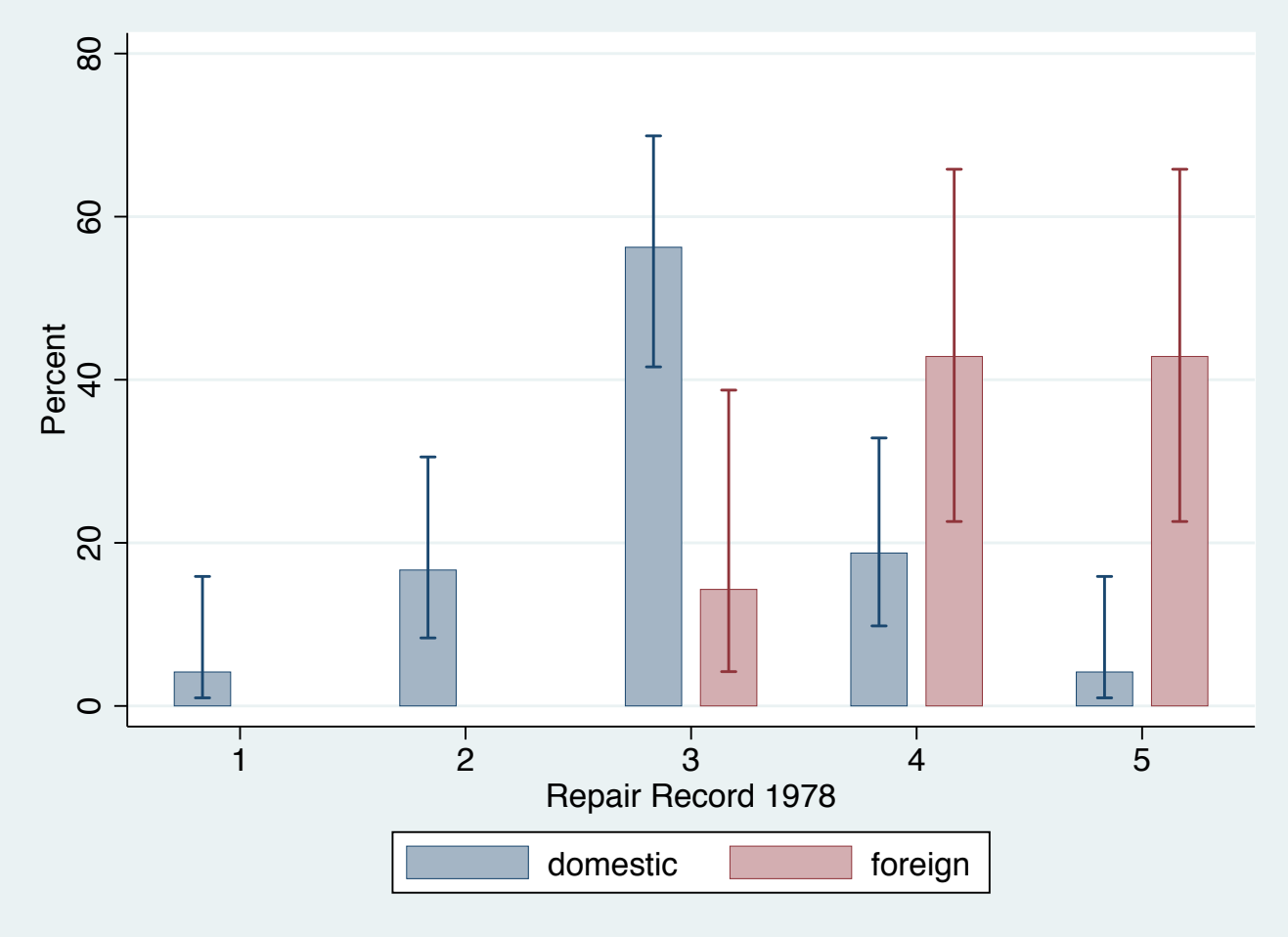
Confidence intervals: Smoothed

```
. coefplot domestic foreign, drop(_cons) xline(0) msymbol(d) cismooth
```



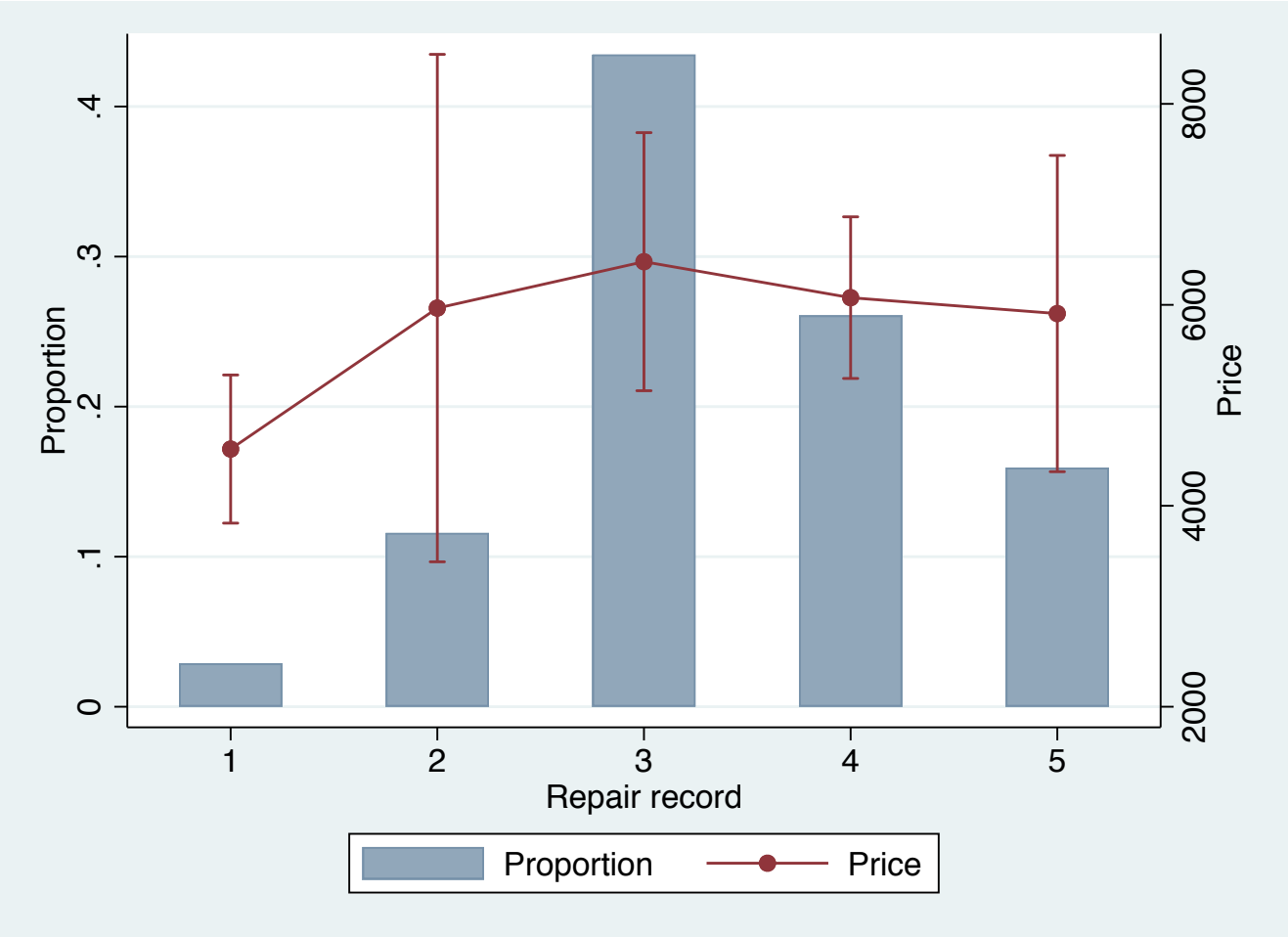
The recast option: Extremely useful!

```
. 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))
```

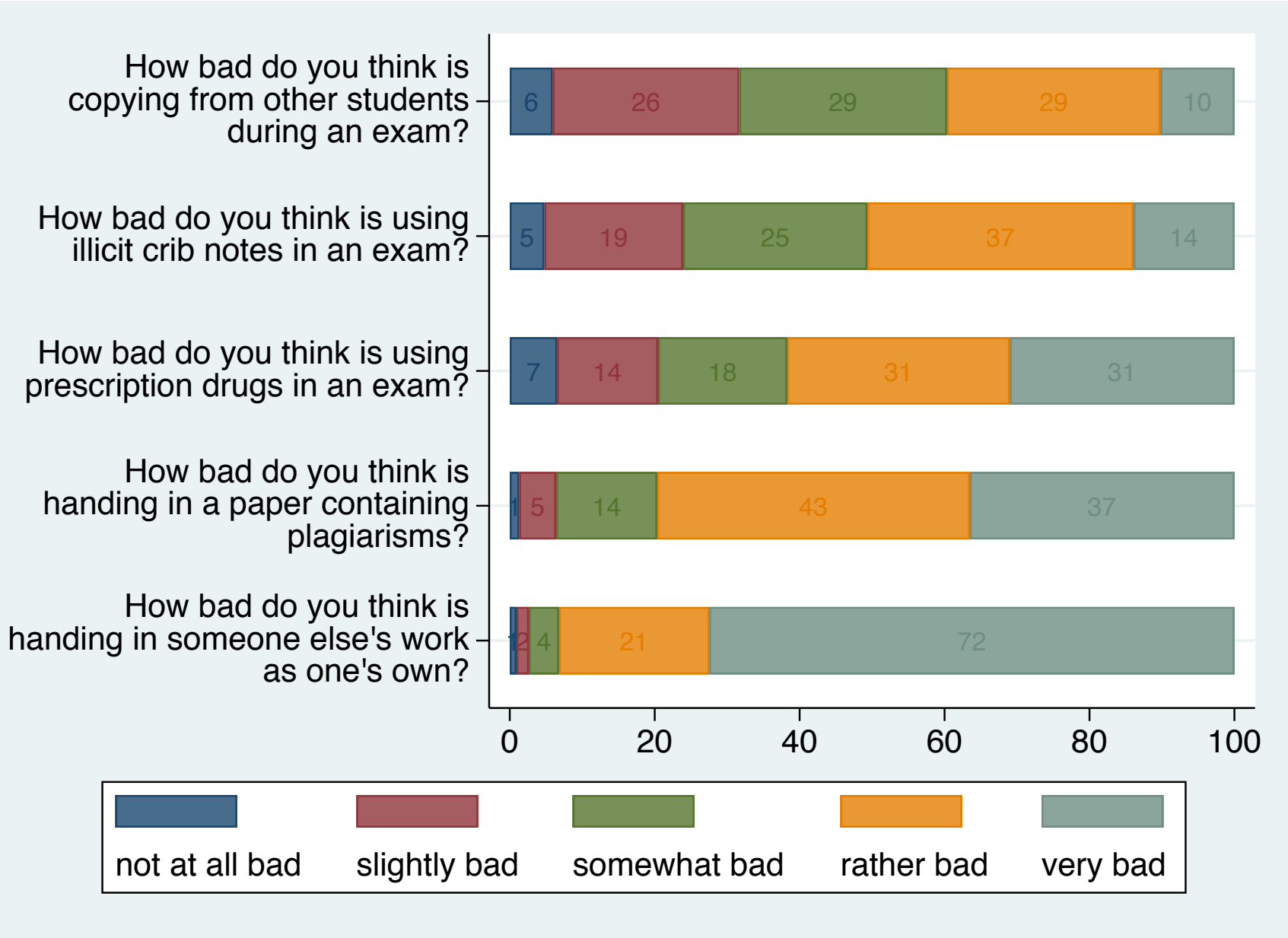


The recast option: A more complicated bar chart

```
. use "http://repec.sowi.unibe.ch/files/wp8/ASQ-ETHBE-2011.dta", clear
(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)
```

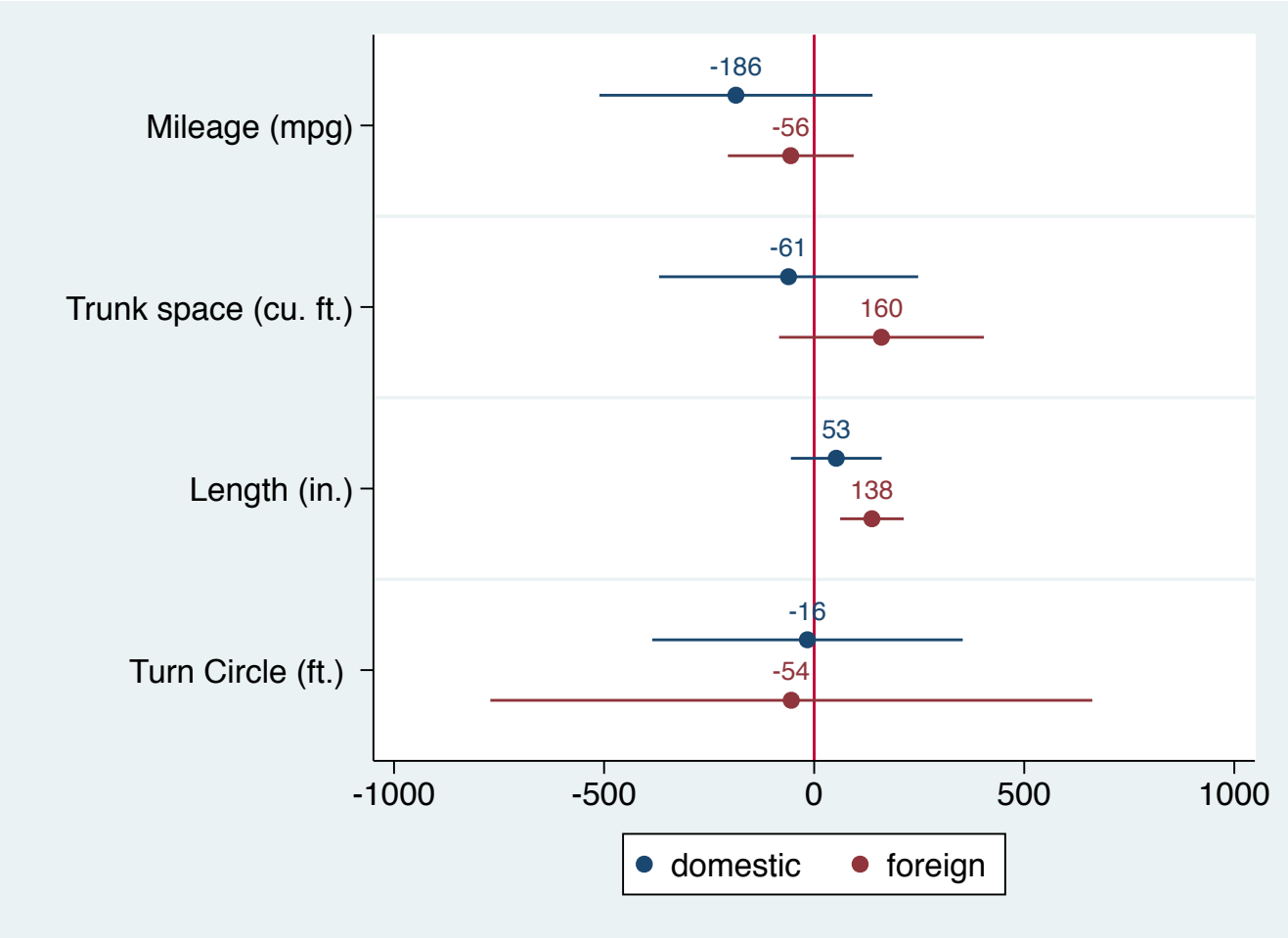
(OK, probably too complicated. I guess I should provide a wrapper for that.)

The recast option: A more complicated bar chart



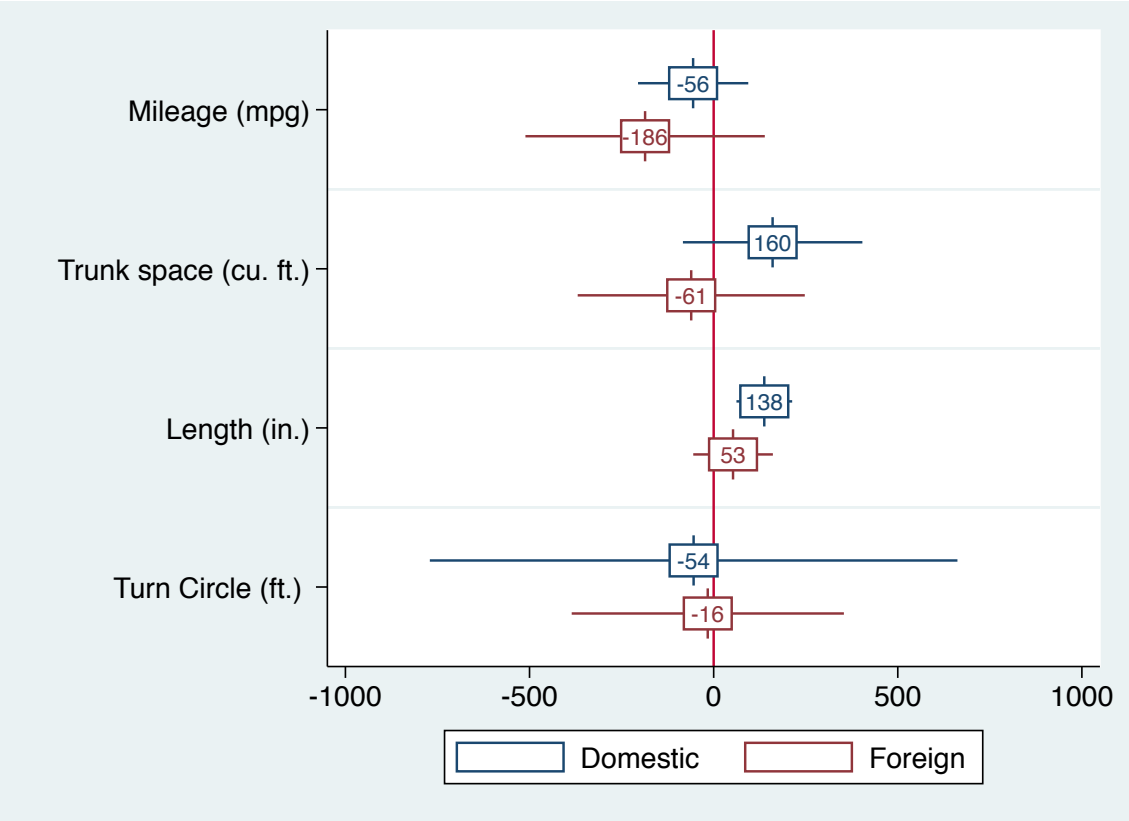
Marker labels

```
. 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)
```



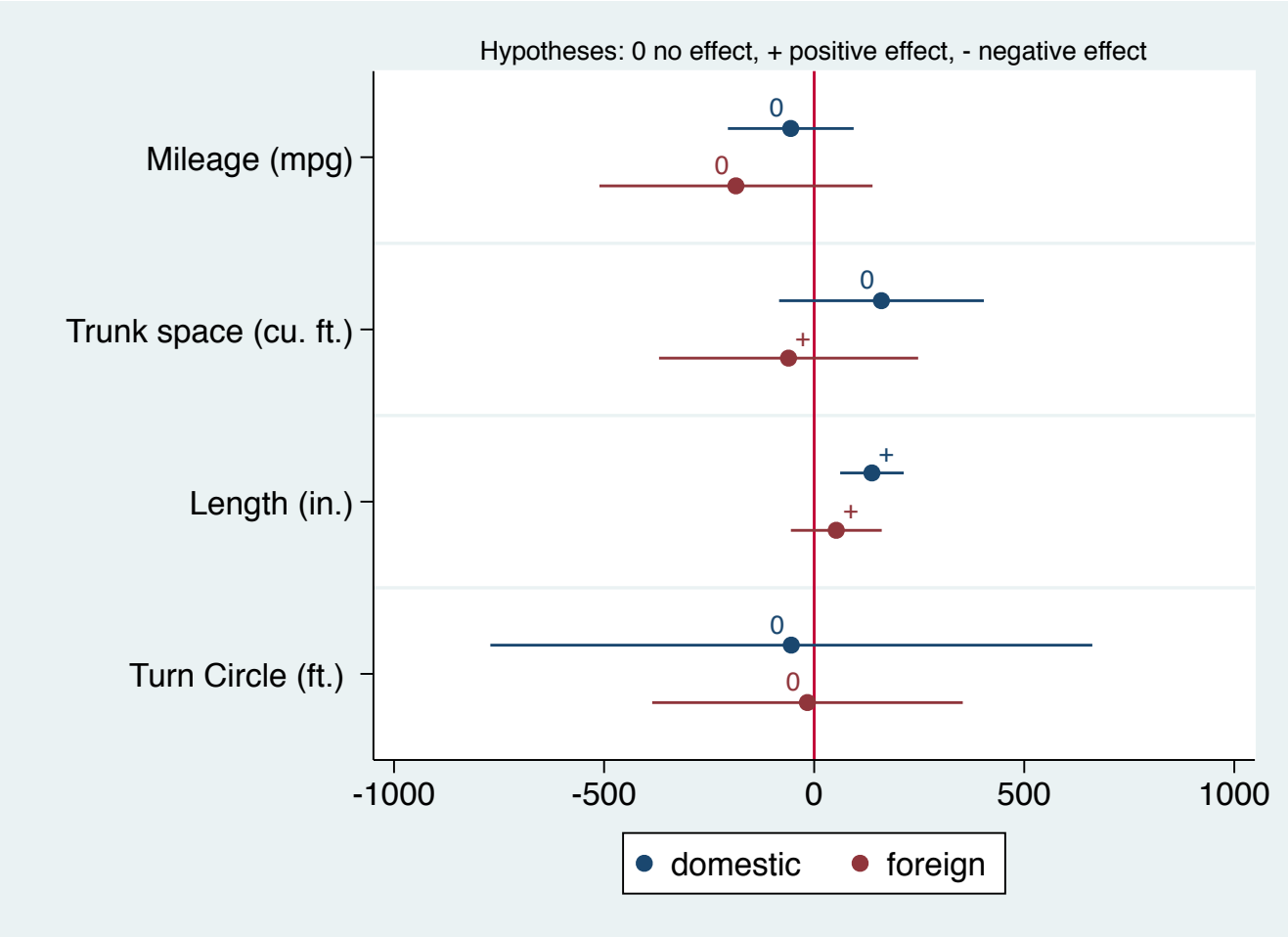
Marker labels: A slightly involved example

```
. 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(%9.0f) mlabposition(0) ci(95 spike box)                     ///
>     ciopts(recast(. rbar rbar) barwidth(. 0.3 0.2) fcolor(. . white) lwidth(. medium medium))
```



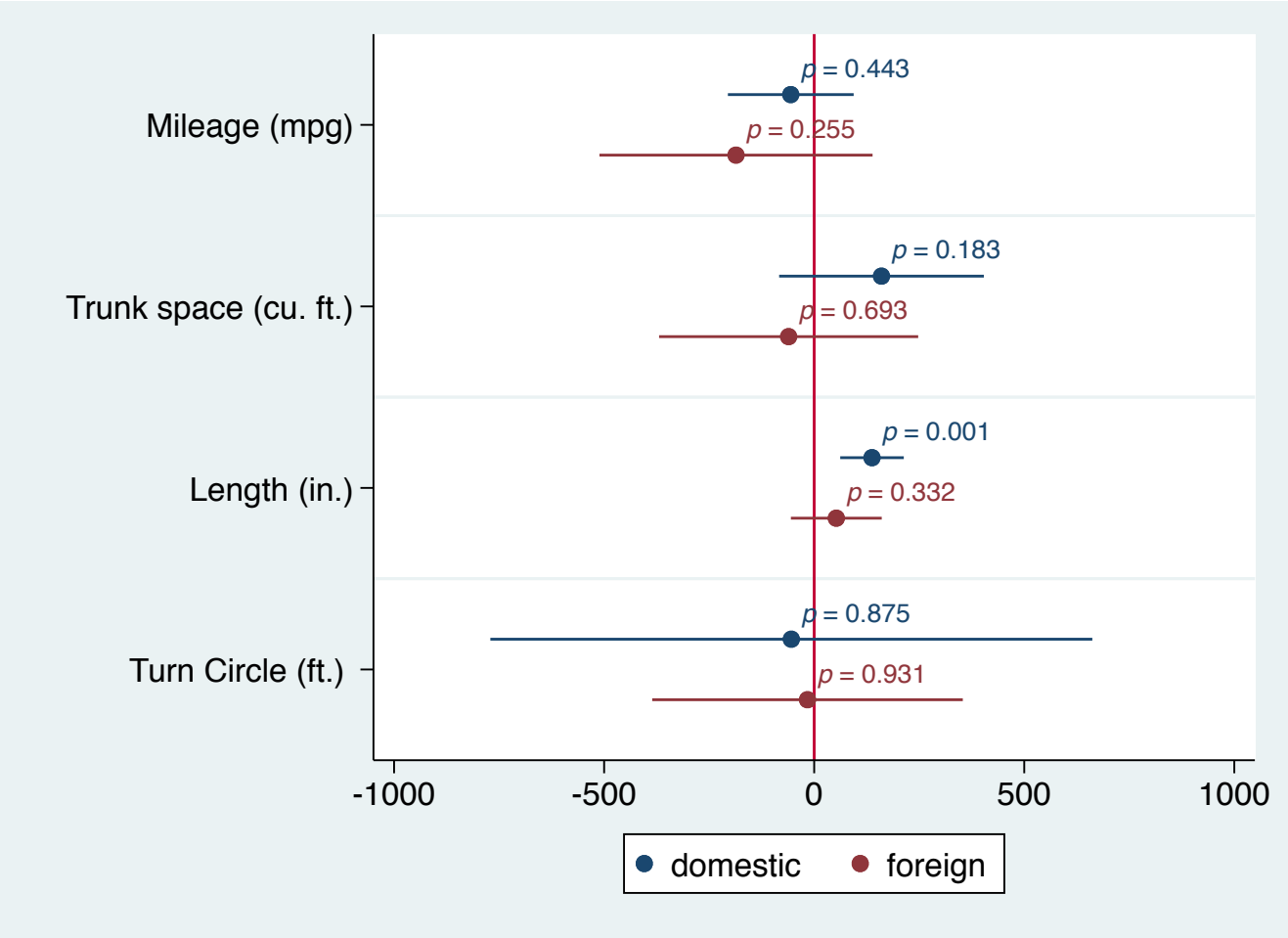
Marker labels: Custom labels

```
. 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))
```



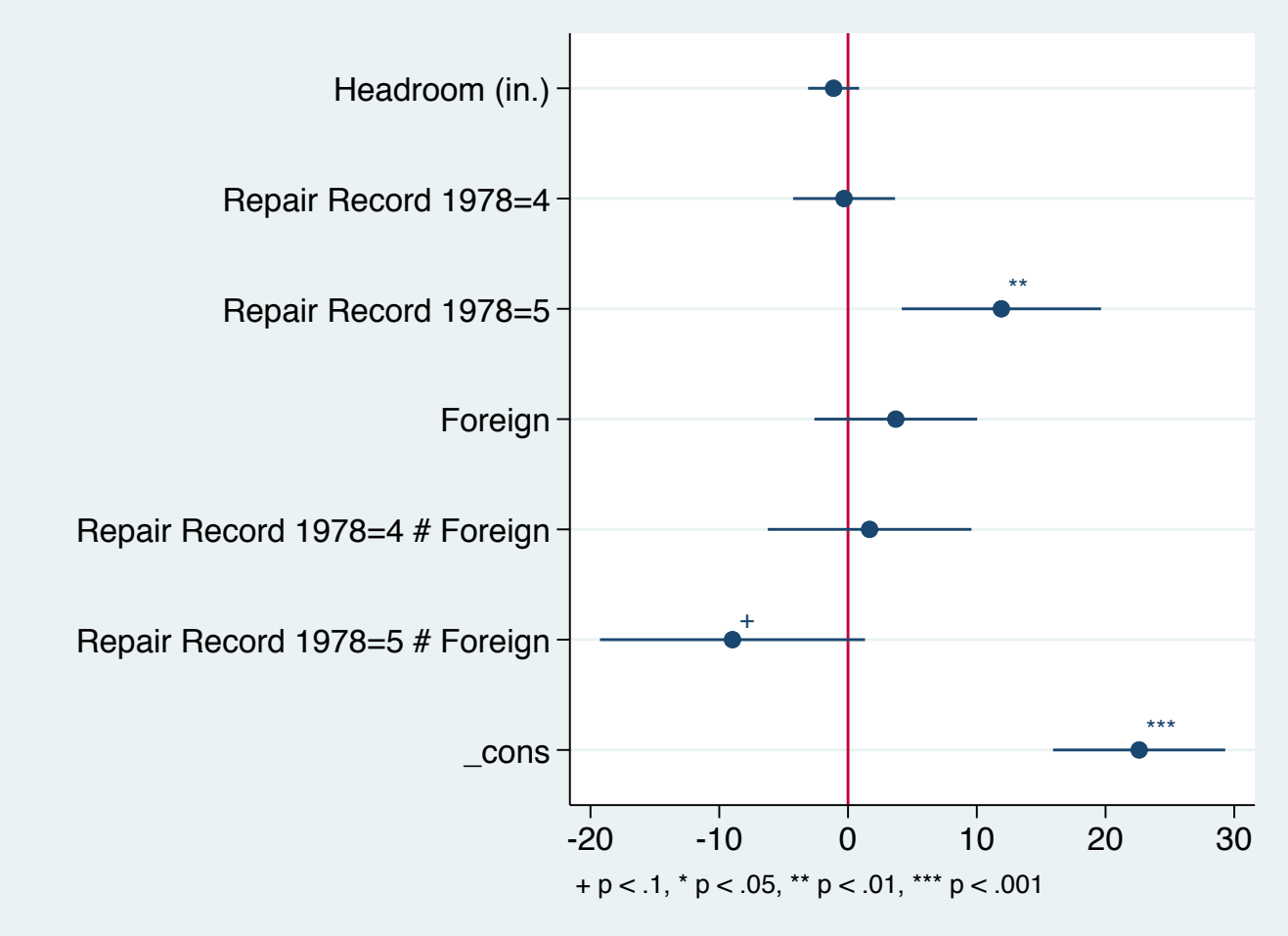
Marker labels: P-Values

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



Marker labels: Significance Stars

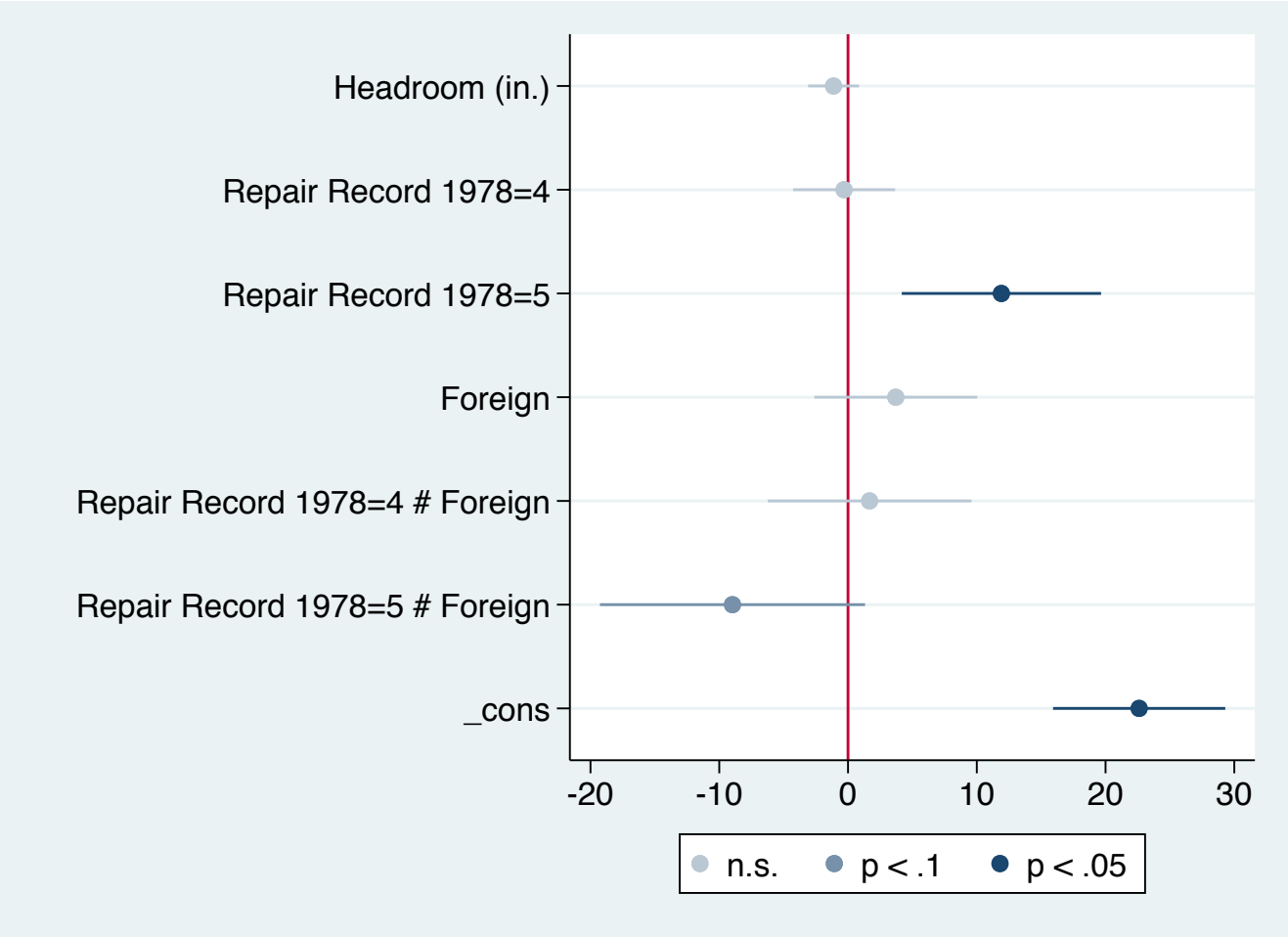
```
. quietly regress mpg headroom i.rep##i.foreign if rep78>=3

. 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:

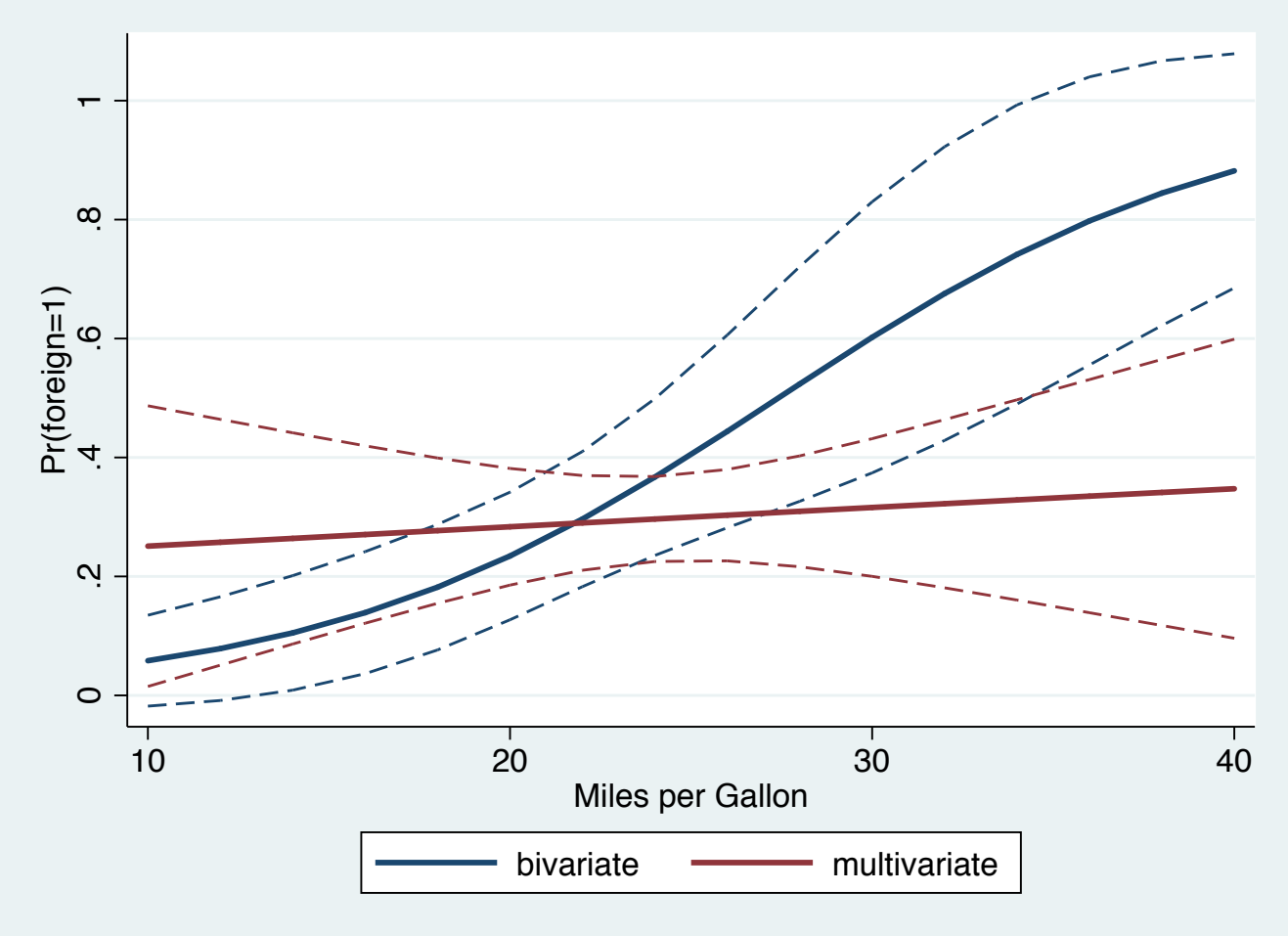
Here is how to draw significant and nonsignificant results in different colors

```
. 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))
```

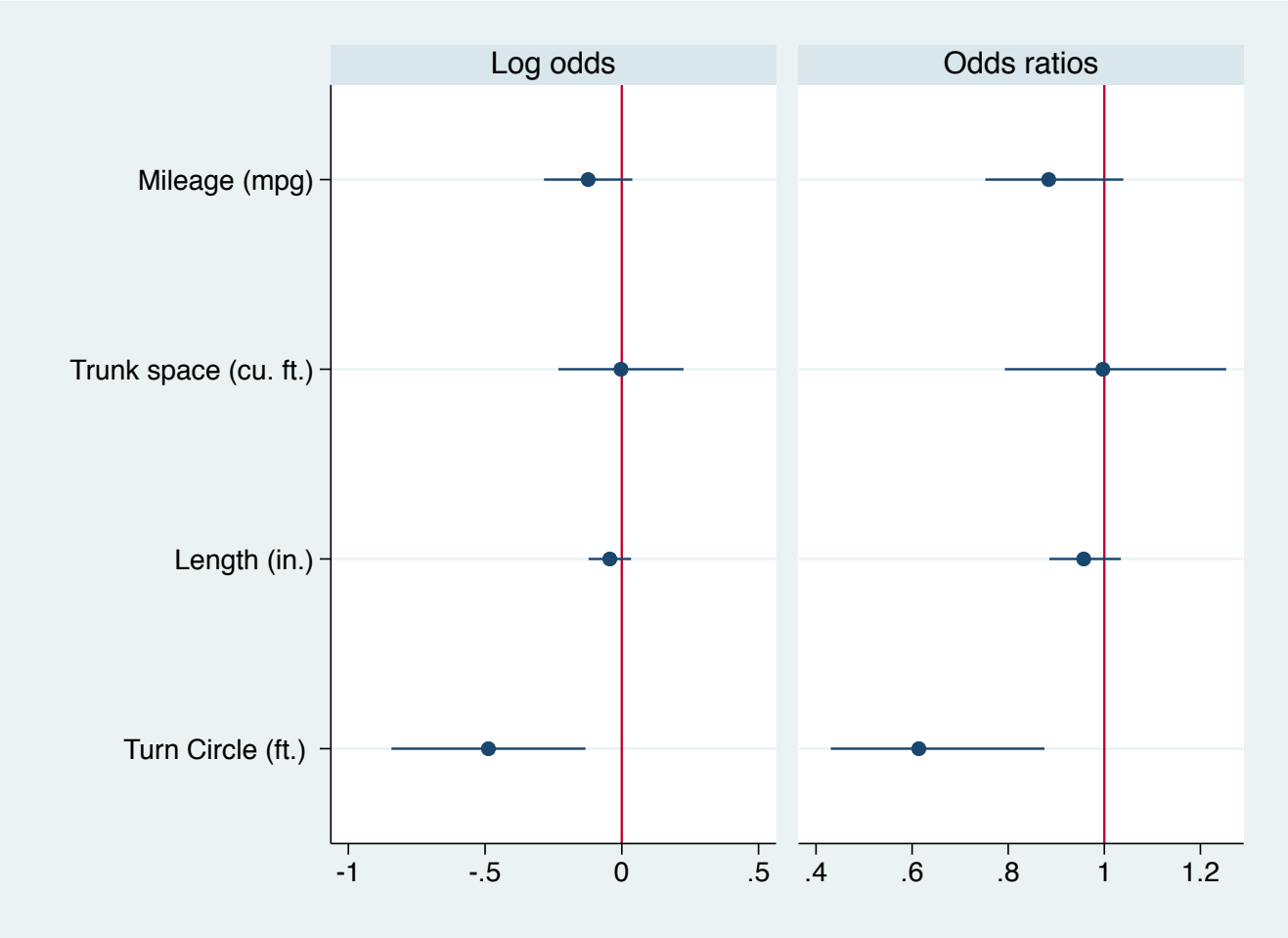


The addplot command

- 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

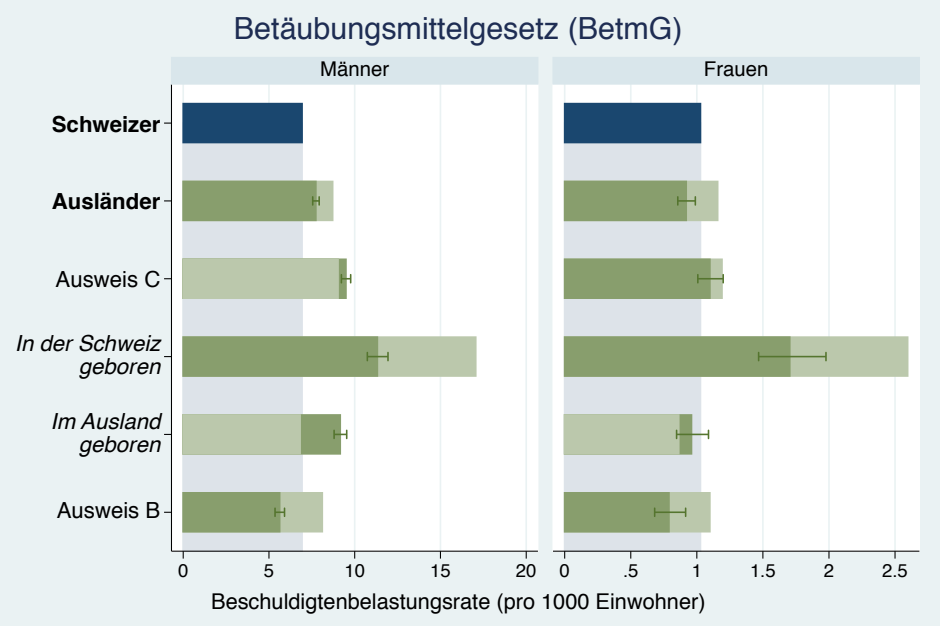
```
. quietly logit foreign mpg trunk length turn
. coefplot ., bylabel(Log odds) || ., bylabel(Odds ratios) eform ///
>     || , drop(_cons) byopts(xrescale)
. addplot 1: , xline(0) norescaling
. addplot 2: , xline(1) norescaling
```

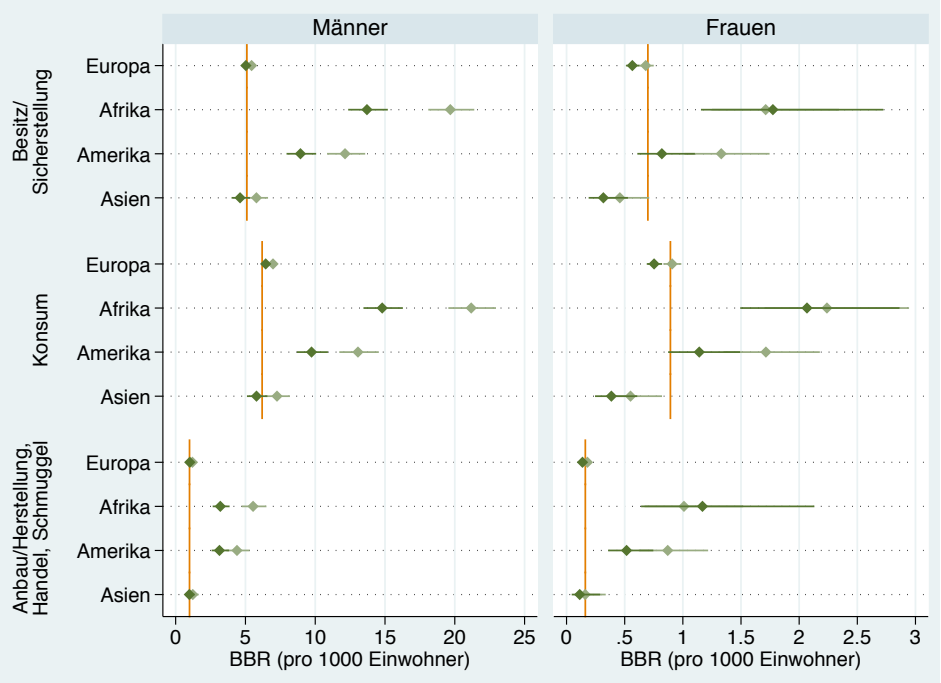


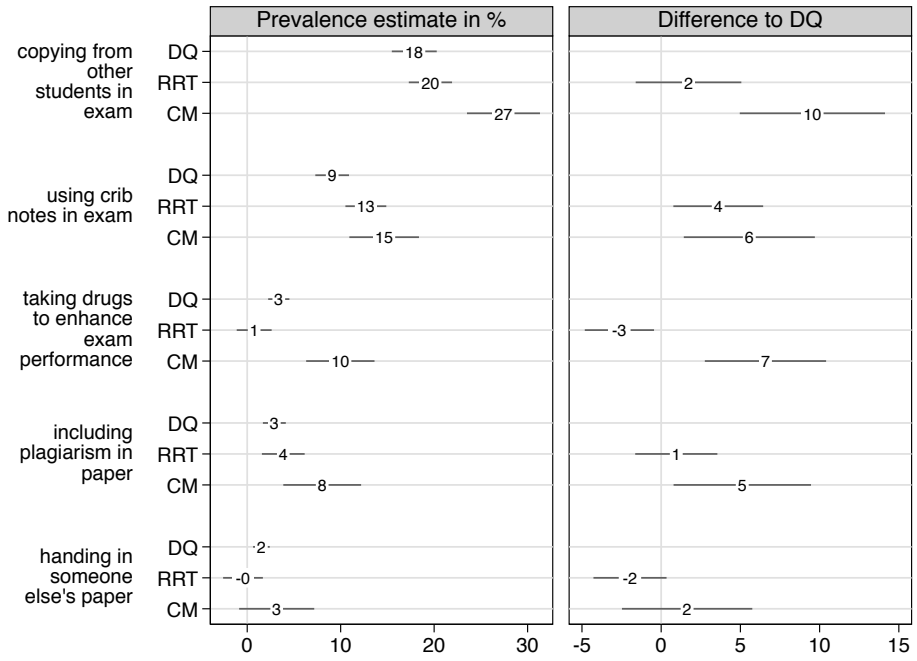
For more example see Jann (2014) or:

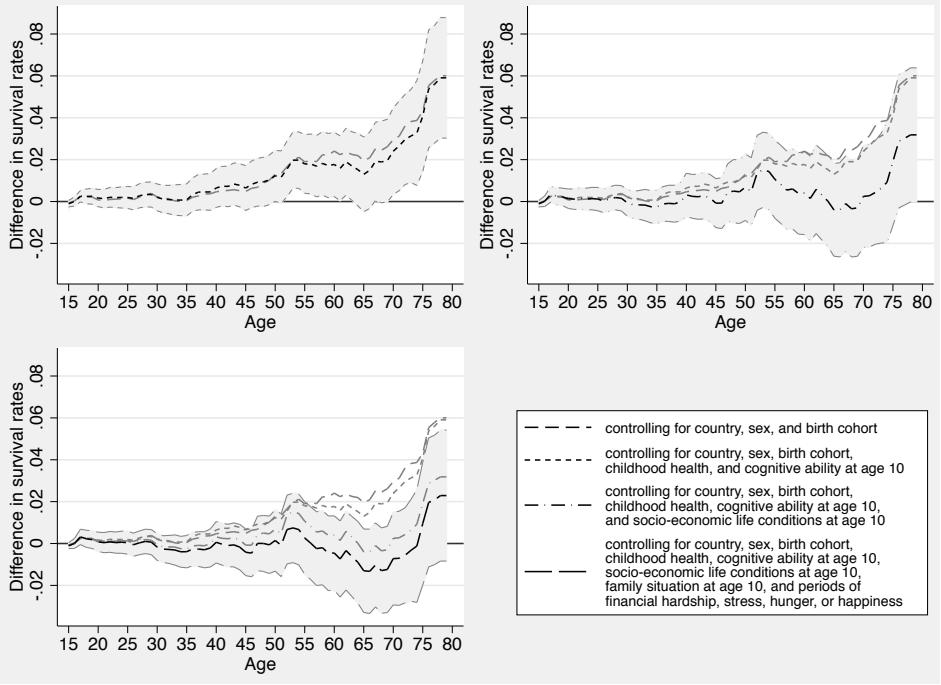
http://repec.sowi.unibe.ch/stata/coefplot

Some examples from my applied work









References I

- Chapin, F. Stuart (1924). The Statistical Definition of a Societal Variable. American Journal of Sociology 30(2): 154-171.
- Cleveland, William S. (1994). The Elements of Graphing Data (Revised Edition). Murray Hill, NJ: AT&T Bell Laboratories.
- Cleveland, William S., Robert McGill (1985). Graphical Perception and Graphical Methods for Analyzing Scientific Data. Science 299(4716): 828-833.
- Dice, Lee R., Harold J. Leraas (1936). A graphic method for comparing several sets of measurements. Contributions from the Laboratory of Vertebrate Genetics(3): 1-3.
- Gallup, John Luke (2012). A new system for formatting estimation tables. The Stata Journal 12(1): 3-28.
- Harrell, Jr., Frank E. (2001). Regression Modeling Strategies. With Applications to Linear Models, Logistic Regression, and Survival Analysis. New York: Springer.
- Jacoby, William G. (1997). Statistical Graphics for Univariate and Bivariate Data. Thousand Oaks, CA: Sage.

References II

- Jann, Ben (2007). Making regression tables simplified. The Stata Journal 7(2): 227-244.
- Jann, Ben (2014). Plotting regression coefficients and other estimates. The Stata Journal 14(4): 708-737.
- Jann, Ben (2015). A note on adding objects to an existing twoway graph. The Stata Journal 15(3): 751-755.
- Kastellec, Jonathan P., Eduardo L. Leoni (2007). Using Graphs Instead of Tables in Political Science. Perspectives on Politics 5(4): 755–771.
- Lewandowsky, Stephan, Ian Spence (1989). The Perception of Statistical Graphs. Sociological Methods & Research 18(2 & 3): 200-242.
- Newson, Roger (2003). Confidence intervals and p-values for delivery to the end user. The Stata Journal 3(3): 245-269.
- Student (1927). Errors of Routine Analysis. Biometrika 19(1/2): 151-164.