

**Outline of the Wild Bootstrap Procedure in Cameron et al.  
ECON 60303**

**Bill Evans  
Spring 2011**

**Problem set up**

The notation for the model in Cameron et al., is slightly different from the notation we use in class. The data set varies across two dimensions (i and g) but in this case, g measures groups and i measures an individual within a group

$$\begin{aligned}y_{ig} &= x_{ig}\beta + u_{ig} \\g &= 1, 2, \dots, G \\i &= 1, 2, \dots, N_g\end{aligned}$$

$x_{ig}$  is a (1 x k) row vector of data for observation (i,g) so  $\beta$  is (k x 1). There are  $N = \sum_{g=1}^G N_g$  observations and the model for the g'th panel can be written as

$$y_g = x_g\beta + u_g$$

Where  $y_g$  is ( $N_g \times 1$ ) and  $x_g$  is ( $N_g \times k$ ). The model for all N observations is then simply

$$y = x\beta + u$$

The OLS estimate can be written as

$$\hat{\beta} = (x'x)^{-1}x'y = \left[ \sum_{g=1}^G x_g'x_g \right]^{-1} \left[ \sum_{g=1}^G x_g'y_g \right]$$

Given this estimate, define  $\hat{u}_g$  to be the  $N_g \times 1$  estimated errors for panel g where

$$\hat{u}_g = y_g - x_g\hat{\beta}$$

The Cluster-robust variance estimate (VCRE) is then

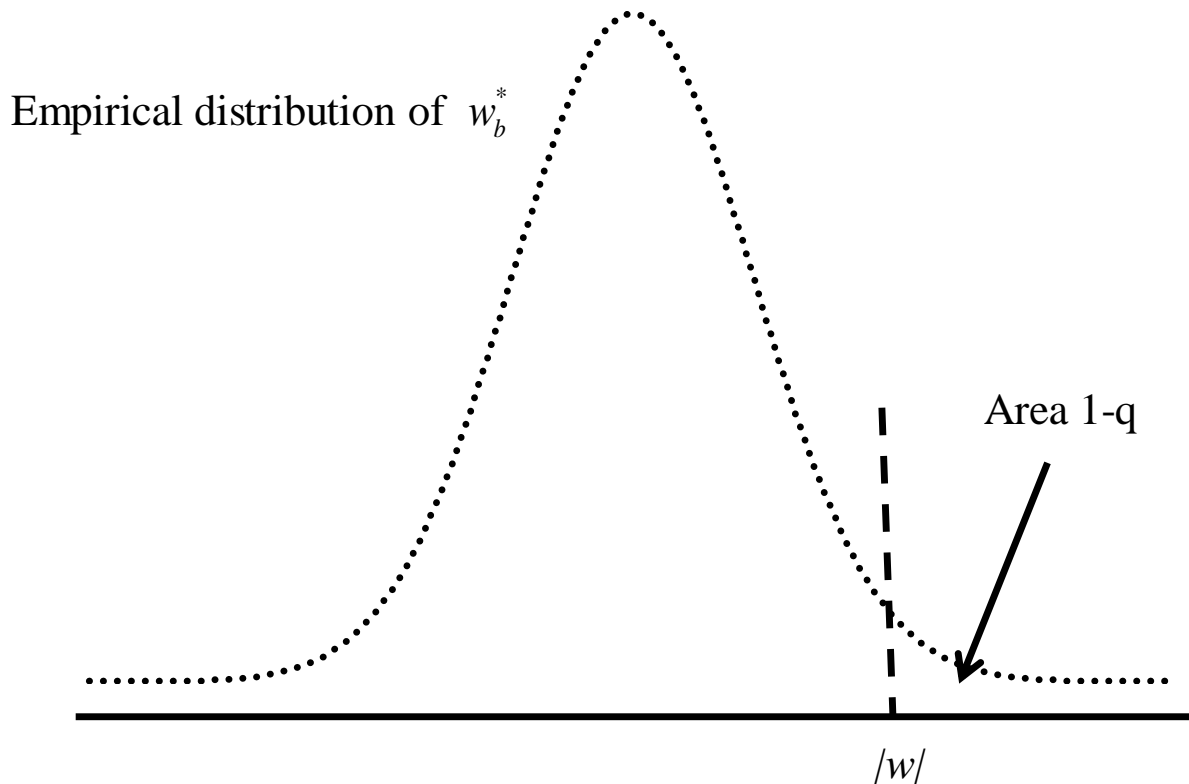
$$\hat{V}_{cr}[\hat{\beta}] = (x'x)^{-1} \left( \sum_{g=1}^G x_g'(\hat{u}_g\hat{u}_g')x_g \right) (x'x)^{-1}$$

## Bootstrap t procedures

Let  $\beta_1$  be the 1<sup>st</sup> element of  $\beta$ . We typically use the estimates to test the null  $H_0 : \beta_1 = \beta_1^0$  against the alternative  $H_a : \beta_1 \neq \beta_1^0$  using the t-statistic  $w = (\hat{\beta}_1 - \beta_1^0) / s_{\hat{\beta}_1}$  where  $s_{\hat{\beta}_1}$  is the estimated standard error for  $\hat{\beta}_1$  from the VCRE. Bootstrap t-procedure produce inference about the underlying shape of the estimate t-statistic  $w$ . Specifically, let  $\hat{\beta}_{1,b}^*$  be the b'th bootstrap estimate of  $\hat{\beta}$  where  $b=1,2,\dots,B$ . For each replication (b), we form an estimate of the t-statistic

$$w_b^* = (\beta_{1,b}^* - \beta_1^0) / s_{\hat{\beta}_{1,b}^*}$$

The distribution of the  $w_b^*$  then tells us about the underlying distribution of  $w$ . Specifically, let  $w_q^*$  be the q'th quantile of the  $w_b^*$ 's and define the area to the right of  $|w_q^*| > w$  as  $1-q$ . The p-value that one could reject the null hypothesis is then  $p\text{-value} = 1 * (1-q)$ . [A nice check is that your empirical distribution of  $w_b^*$  should be centered on zero]. One thing to note: in the bootstrap procedures, the t-statistic is generated with the null hypothesis imposed.



## Wild bootstrap t-procedure

Cameron et al. suggest a wild bootstrap procedure that appears to not generate high Type I error rates. Instead of drawing  $x$ 's and replacing them, this procedure draws errors, uses the estimated beta to generate estimated  $y$ 's, then regresses these predicted  $y$ 's on actual  $x$ 's. The key to the wild bootstrap is that one wants to replicate the within-group correlation in errors when generating new estimates. This can only be done if one uses the original errors as the basis of the bootstrap exercise. This is accomplished through the following procedure. Let  $z_{gb}$  (iteration  $b$  for group  $g$ ) be a random variable that equals 1 with a 50% probability and equals zero otherwise. Define a new error  $\hat{u}_{gb}^*$  that is equal to

$$\hat{u}_{gb}^* = (2z_{gb} - 1)\hat{u}_g$$

Note that  $\hat{u}_{gb}^*$  equals  $\hat{u}_g$  with probability 0.5 and it equals  $-\hat{u}_g$  with probability 0.5. Given a draw to  $z$  that produces  $\hat{u}_{gb}^*$ , now construct a predicted  $y$ ,

$$\hat{y}_{gb} = x\hat{\beta} + \hat{u}_{gb}^*$$

And given  $\hat{y}_{gb}$  we can generate an estimate of  $\hat{\beta}_b^*$  the entire vector for  $\beta$  on the  $b$ 'th iteration

$$\hat{\beta}_b^* = (x'x)^{-1}x'\hat{y}_b = \left[ \sum_{g=1}^G x_g'x_g \right]^{-1} \left[ \sum_{g=1}^G x_g'\hat{y}_{gb} \right]$$

By re-generating positive and negative blocks of errors  $\hat{u}_g$  we preserve the observed correlation in errors. Note that this procedure must be done imposing the null hypothesis. So if the null is that  $H_0 : \beta_1 = 0$  then the original model that produces  $\hat{u}_g$  must exclude  $\beta_1$  from the model.

**Sample Program**  
**wild\_bs\_example\_1.do**

```
#delimit ;

* open log file;
log using wild_bs_example_1.log , replace ;

* set stata parameters;
set mem 5m ;
set more off ;

* fix seed for replication purposes and;
* set the number of bootstrap replications;
set seed 365476247 ;
global bootreps = 999;

tempfile main bootsave ;

use carton_sales_taxes;
drop if year<2004;
/*   the data contains monthly market share of
      cigarette sales by carton (compared to pack)
      for 29 states over the 2001-2006 period so there
      are 29*12*6 = 2088 observations.  I regress the market
      share on real taxes (state+federal in dollars/pack)
      and add state, year and month dummies.  Because
      taxes are at the state level, you cluster at the
      state level.  The parameter we will generate bootstrap
      p-values for is on real_tax and the null hypothesis we
      will impose is  $h_0: \beta(\text{real\_tax})=0$ 
*/

* means of key covariates;
sum carton_market_share real_tax;

* construct the dummies used in analysis;
xi i.state i.month i.year;

di ;
* run ols without clustered std errors, just for comparison;
reg carton_market_share _I* real_tax;

* now run ols and cluster at the state level;
reg carton_market_share _I* real_tax, cluster(state);
* save t-test as a global variable;
global maint = _b[real_tax] / _se[real_tax] ;
```

```

* now run OLS and impose null that real_tax=0;
reg carton_market_share _I*;

* output residuals;
predict epshtat , resid;
predict yhat , xb ;

* sort by state and temp save data;
sort state;
qui save `main' , replace ;

* get the number of states;
qui by state: keep if _n == 1 ;
qui summ ;
global numstates = r(N) ;

* output the t-statistics for real_tax to a file;
postfile bskeep t_wild using bs_results, replace;

* iterate over the bootstrap replications;
forvalues b = 1/$bootreps { ;

/* wild bootstrap */
use `main', replace ;

* with 50% probability constuct dummy;
* that adds or substracts Radamaker error;
qui by state: gen temp = uniform() ;
qui by state: gen pos = (temp[1] < .5) ;
gen wildresid = epshtat * (2*pos - 1) ;

* now construct y;
gen wildy = yhat + wildresid ;

* now regress y on all x variables;
qui reg wildy _I* real_tax, cluster(state);
* generate the t-stat;
local bst_wild = _b[real_tax] / _se[real_tax] ;

* add to the bottom of the post file;
post bskeep (`bst_wild') ;
} ;

/* end of bootstrap reps */

* save the post file;
postclose bskeep ;

* clear the current data set;
clear;

```

```

* load up the wild t-stats;
use bs_results;

* figure out where the main-t is in the;
* synthetic distribution;
gen positive=$maint>0;
gen pos=t_wild>$maint;
gen neg=t_wild<$maint;
gen reject=positive*pos + (1-positive)*neg;
sum reject;
local sumreject=r(sum);
local p_value_wild=2*`sumreject'/$bootreps;
local p_value_main=2*(ttail(($numstates-1),abs($maint)));

di "Number BS reps" = $bootreps";
di "P-value from clustered standard errors" = `p_value_main'";
di "P-value from wild bootstrap" = `p_value_wild'";
log close ;

```

```

. * run ols without clustered std errors, just for comparison;
. reg carton_market_share_I* real_tax;

```

Source	SS	df	MS	Number of obs =	1044
Model	30.3895294	42	.723560223	F( 42, 1001) =	1222.46
Residual	.592482903	1001	.000591891	Prob > F =	0.0000
Total	30.9820123	1043	.02970471	R-squared =	0.9809
				Adj R-squared =	0.9801
				Root MSE =	.02433

carton_mar~e	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
_Istate_2	-.1450251	.0063325	-22.90	0.000	-.1574516	-.1325987
_Istate_3	-.2283005	.0059946	-38.08	0.000	-.2400639	-.216537

DELETE SOME RESULTS

_Imonth_11	-.0053518	.0036984	-1.45	0.148	-.0126094	.0019058
_Imonth_12	.0040418	.0036942	1.09	0.274	-.0032075	.0112911
_Iyear_2005	-.0046846	.0018602	-2.52	0.012	-.0083349	-.0010343
_Iyear_2006	-.013917	.0018705	-7.44	0.000	-.0175875	-.0102464
real_tax	-.0201751	.003371	-5.98	0.000	-.0267903	-.01356
_cons	.5595832	.0054096	103.44	0.000	.5489677	.5701988

```
. * now run ols and cluster at the state level;
. reg carton_market_share _I* real_tax, cluster(state);
```

```
Linear regression                               Number of obs =    1044
                                                F( 13,    28) =      .
                                                Prob > F      =      .
                                                R-squared     =  0.9809
                                                Root MSE     =  .02433
```

(Std. Err. adjusted for 29 clusters in state)

	Coef.	Robust Std. Err.	t	P> t	[95% Conf. Interval]	
carton_mar~e						
_Istate_2	-.1450251	.0066001	-21.97	0.000	-.1585449	-.1315054
_Istate_3	-.2283005	.0042925	-53.19	0.000	-.2370932	-.2195078

DELETE SOME RESULTS

_Imonth_11	-.0053518	.0035491	-1.51	0.143	-.0126217	.0019182
_Imonth_12	.0040418	.0048803	0.83	0.415	-.005955	.0140387
_Iyear_2005	-.0046846	.0040704	-1.15	0.260	-.0130224	.0036533
_Iyear_2006	-.013917	.0070822	-1.97	0.059	-.0284241	.0005901
real_tax	-.0201751	.0082818	-2.44	0.021	-.0371397	-.0032106
_cons	.5595832	.0074706	74.90	0.000	.5442803	.5748862

```
. di "Number BS reps" = $bootreps";
Number BS reps = 999
```

```
. di "P-value from clustered standard errors = `p_value_main'";
P-value from clustered standard errors = .0214648522876161
```

```
. di "P-value from wild bootstrap" = `p_value_wild'";
P-value from wild bootstrap = .0640640640640641
```