Cluster–robust inference in Stata

Eduardo García Echeverri

Stata Webinar, June 2024

1. Clustered errors: Why are they important?
2. The cluster–robust variance estimator (CRVE)
3. Alternatives when the assumptions of the CRVE fail
   - Adjusted degrees of freedom for `vce(hc2)`
   - Wild cluster bootstrap
4. Conclusion

Outline

Clustered errors in your data

1. Data on firms operating in different sectors:
   - Errors are (probably) correlated within industries.
2. Data on high school students in the USA:
   - Errors are (probably) correlated within schools.
3. Panel data on individuals:
   - For each individual, errors are (probably) serially correlated.
4. Experiments with treatment at an aggregated level.
   - Errors are (probably) correlated within such levels.

Example: States changing minimum wage.

Linear model with clustered errors

Consider the model:

$$y_{ig} = X_{ig}\beta + \varepsilon_{ig}$$

where,

y_{ig}: **outcome** for observation i in cluster g;

X_{ig}: vector of **covariates** for observation i in cluster g;

ε_{ig}: **error term** for observation i in cluster g;

β: **coefficients** of interest;

$g = 1, 2, \ldots \mathbf{G}$.

Clustered errors

Errors **between clusters** are **uncorrelated**:

$$\mathrm{Cor}\left(\varepsilon_{ig}, \varepsilon_{j\tilde{g}}\right) = 0$$

Errors **within the same cluster** are (possibly) **correlated**:

$$\mathrm{Cor}\left(\varepsilon_{ig}, \varepsilon_{jg}\right) \neq 0$$

Thus, we are relaxing the assumption of **i.i.d. errors**.

Clustered data complicates inference

Problem: CI's assuming i.i.d don't have the right coverage

- Coverage is (typically) less than 95%
- SE are (typically) too small
- May lead to overreject null hypotheses (false positives).

Let's see this in some Monte Carlo simulations.

Linear experimental design

Data generating process:

$$y_{ig} = 1 + x_{ig} + z_{ig} + T_g + \mu_{ig} + \nu_g$$

where,

y_{ig}: **outcome** for observation i in cluster g;

x_{ig}, z_{ig}: **control** variables, $N(0, 3)$ and $\chi^2(7)$ respectively.

Obs $=$ 1000.

g: observations randomly assigned among 100 clusters;

T_g: 33 clusters randomly assigned to treatment ($T_g = 1$)

Clustered errors

Data generating process:

$$y_{ig} = 1 + x_{ig} + z_{ig} + T_g + \mu_{ig} + \nu_g$$

where,

ν_g: clustered component of error term, $N(0, 0.5)$;

μ_{ig}: individual component of error term, $N(0, 0.5)$.

$\Rightarrow$ Errors are **correlated** within clusters.

Monte Carlo simulations

Procedure:

1. Simulate the DGP.
2. `regress y x z treat`
3. Store coeffficient for `treat`: [`beta`]
4. Check if CI for `treat` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Monte Carlo simulations

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	1.008647	.2367399	.1830714	1.775514
contained	1,000	.564	.4961352	0	1

Remarks:

1. Coverage is just 56.4% (vs. 95% nominal size)
2. Estimator is still consistent.
3. CI's are too narrow.

Controlling for cluster dummies is not the solution

Procedure:

1. Simulate the DGP.
2. `regress y x z treat i.cvar`
3. Store coeffficient for `treat`: [`beta`]
4. Check if CI for `treat` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Monte Carlo controlling for cluster dummies

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	1.01037	1.482467	-3.426341	5.811284
contained	1,000	.471	.4994081	0	1

Remarks:

1. Coverage fell to 47.1% (vs. 95% nominal size)
2. Inefficient: estimating coefficients of irrelevant variables.

Nonlinear experimental design

Data generating process (Probit model):

$$y^*_{ig} = 1 + x_{ig} + z_{ig} + T_g - \mu_{ig} - \nu_g$$

$$y_{ig} = 1\{y^*_{ig} \geq 0\}$$

where,

y_{ig}: **outcome** for observation i in cluster g;

x_{ig}, z_{ig}: **control** variables, $N(0,3)$ and $\chi^2(7)$ respectively.

Obs $=$ 100000

g: observations randomly assigned among 100 clusters;

T_g: 33 clusters randomly assigned to treatment ($T_g = 1$)

Clustered errors

Data generating process:

$$y_{ig}^{*} = 1 + x_{ig} + z_{ig} + T_g - \mu_{ig} - \nu_g$$

$$y_{ig} = 1\{y_{ig}^{*} \geq 0\}$$

where,

ν_g: clustered component of error term, $N(0, 0.5)$;

μ_{ig}: individual component of error term, $N(0, 0.5)$.

Monte Carlo simulations

Procedure:

1. Simulate the DGP.
2. `probit y x z treat`
3. Store coefficient for `treat`: [`beta`]
4. Check if CI for `treat` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Monte Carlo nonlinear experiment

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	1.022974	.1905835	.3237958	1.550756
contained	1,000	.558	.4968731	0	1

Remarks:

1. Coverage is just 55.8% (vs. 95% nominal size)
2. Estimator is still consistent.
3. CI's are too narrow.

Solutions

1. Cluster–robust variance estimator:
   - Option `vce(cluster` *cvarlist*`)`
     Liang and Zeger (1986) ....
   - Adjust degrees of freedom: `vce(hc2` *cvar*`, dfadjust)` **[Stata 18]**
     Bell and McCaffrey (2002)

2. Wild cluster bootstrap **[Stata 18]**
   Cameron, Gelbach, and Miller (2008)

Outline

Cluster–robust variance estimator (CRVE)

CI's can be **corrected** using the **CRVE**:

$$\hat{V} = \frac{G(N-1)}{(G-1)(N-k)} (XX')^{-1} \left(\sum_{g=1}^{G} X_g' \hat{\epsilon}_g \hat{\epsilon}_g' X_g \right) (XX')^{-1}$$

The 95% **corrected CI** for β_k: $\left[\hat{\beta}_k - 1.96\sqrt{\hat{V}_{k,k}}, \hat{\beta}_k + 1.96\sqrt{\hat{V}_{k,k}} \right]$

Implementation – option `vce(cluster)`

Example:

estimation_command ..., `vce(cluster` *cvarlist*`)`

Check availability:

`help` *estimation_command*

Monte Carlo simulations – Linear experimental design

$$y_{ig} = 1 + x_{ig} + z_{ig} + T_g + \mu_{ig} + \nu_g$$

Procedure:

1. Simulate the DGP.
2. `regress y x z treat, vce(cluster cvar)`
3. Store coeffficient for `treatment`: [beta]
4. Check if CI for `treatment` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Monte Carlo CRVE

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	.9974977	.2236554	.1211772	1.817533
contained	1,000	.959	.1983894	0	1

Remarks:

1. Coverage is 95.9% (vs. 95% nominal size)

Monte Carlo simulations – Probit experimental design

$$y_{ig}^{*} = 1 + x_{ig} + z_{ig} + T_g - \mu_{ig} - \nu_g$$

$$y_{ig} = 1\{y_{ig}^{*} \geq 0\}$$

Procedure:

1. Simulate the DGP.
2. `probit y x z treat, vce(cluster cvar)`
3. Store coeffficient for `treat`: [`beta`]
4. Check if CI for `treat` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Monte Carlo Probit CRVE

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	1.022447	.1861081	.3533935	1.591309
contained	1,000	.929	.2569534	0	1

Remarks:

1. Coverage is 92.9% (vs. 95% nominal size)

CRVE typically increases SE, improving CI coverage

Example: Linear regression with and without CRVE (`wage_work.dta`)

	wage_CRVE	wage_iid
Job tenure	**0.609**	**0.609**
	(0.020)	**(0.016)**
Labor-market condition	**-0.049**	**-0.049**
	(0.029)	**(0.030)**
Age in years	**0.198**	**0.198**
	(0.007)	**(0.005)**
Intercept	**13.506**	**13.506**
	(0.284)	**(0.212)**
Number of observations	**1928**	**1928**

Limitations of CRVE

The **CRVE can work well**, but the asymptotics depend on G.

The **CRVE can perform poorly** when:

1. The number of clusters **G is small**.
2. Cluster have very **different sizes**.

Again, let see it in a Monte Carlo simulation.

Linear experimental design with few clusters

Data generating process:

$$y_{ig} = 1 + x_{ig} + z_{ig} + T_g + \mu_{ig} + \nu_g$$

where,

y_{ig}: **outcome** for observation i in cluster g;

x_{ig}, z_{ig}: **control** variables, $N(0, 3)$ and $\chi^2(7)$ respectively.

Obs $=$ 1000.

g: observations randomly assigned among 21 clusters;

T_g: 7 clusters randomly assigned to treatment ($T_g = 1$)

Results – Monte Carlo few clusters

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	1.021723	.8693036	-1.44612	3.697521
contained	1,000	.879	.3262905	0	1

Remarks:

1. Coverage of 87.9% (vs. 95% nominal size)

Outline

Solution 1: Adjusted degrees of freedom (DoF)

Bell and McCaffrey (2002): adjust DoF based on `cvar`

- Improves CI coverage when clusters are very few.

`hc2`: requests more conservative estimator for residuals

Implementation:

estimation_command ..., `vce(hc2` *cvar*`, dfadjust)`

Check availability:

`help` *estimation_command*

Let's see it in action in a Monte Carlo simulation.

Back to the linear experimental design (7 clusters)

$$y_{ig} = 1 + x_{ig} + z_{ig} + T_g + \mu_{ig} + \nu_g$$

Procedure:

1. Simulate the DGP.
2. `regress y x z treat, vce(hc2 cvar, dfadjust)`
3. Store coeffficient for `treat`: [beta]
4. Check if CI for `treat` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Monte Carlo with few clusters, adjusted DoF

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	1.040855	.835802	-1.290347	4.12301
contained	1,000	.978	.1467567	0	1

Remarks:

1. Coverage of 97.8% (vs. 95% nominal size)

Solution 2: The `wildbootstrap` command

Syntax:
`wildbootstrap` *estimator depvar [indepvars] [if] [in] [weight] [, options]*

estimator:

- `regress`
- `areg`
- `xtreg` (**fixed-effects** model only; no need to specify `fe` option)

Quick Start

Estimate the WCB p-value and CI for the coefficient on $\texttt{x}_1$ in a linear regression of `y` on $\texttt{x}_1$ with clusters identified in `cvar`

```
wildbootstrap regress y x1, cluster(cvar)
```

Wild cluster bootstrap algorithm

Consider the model:

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_G \end{bmatrix} = X\beta + \epsilon = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_G \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_G \end{bmatrix}$$

where,

y_g: vector of **outcomes** for cluster g,

X_g: vector of **covariates** for cluster g,

ε_g: vector of **errors** for cluster g,

β: **coefficients** of interest.

The wild cluster restricted bootstrap (WCRB)

The **WCRB algorithm** consists of 4 steps:

Suppose, for example, we want to test $\mathbb{H}_0 : \beta_k = 0$.

1. **Re-estimate** the linear model **with the restriction** $\beta_{\mathbf{k}} = \mathbf{0}$.
   - $\tilde{\beta}$: restricted **estimated coefficients**
   - $\tilde{\epsilon}$: restricted **estimated residuals**

WCRB algorithm

2. Create a **bootstrap replication b** (repeat B times):

   2.1 Generate **random variable** $\nu_{\mathbf{g}}^{\mathbf{b}}$ for each cluster g:
   - Distribution specified in `errorweight()`

   E.g. `rademacher`: -1 or $+1$ with equal probability

   2.2 Generate a **new dependent variable** $\mathbf{y}_{\mathbf{ig}}^{\mathbf{b}}$:

$$y_{ig}^{b} = X_{ig}\tilde{\beta} + \tilde{\epsilon}_{ig}\nu_{g}^{b}.$$

   2.3 **Re-estimate the model** using variable y_{ig}^{b} on the LHS.

   2.4 Calculate the **t-statistic** $t_k^b = \dfrac{\hat{\beta}_k^b}{\sqrt{\hat{V}_{k,k}^b}}$

   $\hat{\beta}_k^b$: **estimated coefficient** in bootstrap replication.
   $\hat{V}_{k,k}^b$: **CRVE** for k-th coefficient in the bootstrap replication.

WCRB algorithm

Suppose $\mathbb{H}_A : \beta_k \neq 0$.

3. **Calculate p-values** depending on the the t-statistics' distribution:
   - **Symmetric** around 0: $p_S = \frac{1}{B}\sum_{b=1}^{B} I\left(\left|t_k^b\right| > \left|t_k\right|\right)$

   t_k : original sample t-statistic
   - **Otherwise**: $p_e = 2\min\left(p_1, p_2\right)$

   p_1, p_2: Bootstrap p-values for **one-sided alternative hypotheses**.

4. Obtain CI's by **inverting the test**. Find t-statistics such that p-value is 0.05.

Back to the linear experimental design (7 clusters)

$$y_{ig} = 1 + x_{ig} + z_{ig} + T_g + \mu_{ig} + \nu_g$$

Procedure:

1. Simulate the DGP.
2. `wildbootstrap reg y x z treat, cluster(cvar)`
3. Store coeffficient for `treatment`: [beta]
4. Check if CI for `treatment` contains 1
5. Repeat 1000 times steps 1-4.
6. Count number of times 1 was contained in CI.

Results – Wild cluster bootstrap

Variable	Obs	Mean	Std. dev.	Min	Max
beta	1,000	.9961465	.4699228	-1.036362	2.653632
contained	1,000	.957	.2029586	0	1

Remarks:

1. Coverage of 95.7% (vs. 95% nominal size)

Example 1: Simple linear regression

```
. use https://www.stata-press.com/data/r18/wagework, replace
(Wages for 20 to 77 year olds, 2013-2016)

. wildbootstrap regress wage tenure, cluster(personid) rseed(12345)

Performing 1,000 replications for p-value for tenure = 0 ...
Computing confidence interval for tenure
  Lower bound: .........10.........20. done (21)
  Upper bound: .........10..... done (17)

Wild cluster bootstrap                          Number of obs      = 1,928
Linear regression                               Number of clusters =   589
                                                Cluster size:
Cluster variable: personid                                  min =     1
Error weight: Rademacher                                    avg =   3.3
                                                            max =     4
```

wage	Estimate	t	p-value	[95% conf.	interval]
constraint					
tenure = 0	.7807403	27.19	0.000	.7209754	.8368386

Example 1: Using CRVE instead

```
. regress wage tenure, vce(cluster personid)

Linear regression                               Number of obs     =      1,928
                                                F(1, 588)         =     739.36
                                                Prob > F          =     0.0000
                                                R-squared         =     0.4212
                                                Root MSE          =     3.5097

                                   (Std. err. adjusted for 589 clusters in personid)
------------------------------------------------------------------------------
             |               Robust
        wage |      Coef.   Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
      tenure |   .7807403   .028713    27.19   0.000     .7243477    .8371328
       _cons |   20.89884   .2135686    97.86   0.000     20.47939    21.31829
------------------------------------------------------------------------------
```

Remark: Large G and clusters have similar size, $CI_{CRVE} \approx CI_{WCRB}$

Example 2: Few clusters of heterogeneous size

```
. use https://www.stata-press.com/data/r18/nlsw88, replace
(NLSW, 1988 extract)

. wildbootstrap regress wage tenure, cluster(industry) rseed(12345)

Performing 1,000 replications for p-value for tenure = 0 ...
Computing confidence interval for tenure
  Lower bound: .........10.........20...... done (26)
  Upper bound: .........10.........20.... done (24)

Wild cluster bootstrap                          Number of obs      = 2,217
Linear regression                               Number of clusters =    12
                                                Cluster size:
Cluster variable: industry                                  min =     4
Error weight: Rademacher                                    avg = 184.8
                                                            max =   817
```

wage	Estimate	t	p-value	[95% conf.	interval]
constraint					
tenure = 0	.1830716	6.95	0.000	.1274023	.3258156

Example 2: Using CRVE instead

```
. regress wage tenure, vce(cluster industry)

Linear regression                               Number of obs     =      2,217
                                                F(1, 11)          =      48.30
                                                Prob > F          =     0.0000
                                                R-squared         =     0.0305
                                                Root MSE          =     5.6853

                                   (Std. err. adjusted for 12 clusters in industry)
------------------------------------------------------------------------------
             |               Robust
        wage |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      tenure |   .1830716   .026341     6.95   0.000     .1250953    .2410478
       _cons |   6.710915   .4147967   16.18   0.000     5.797954    7.623877
------------------------------------------------------------------------------
```

Remark: Small G and clusters of dissimilar size, $CI_{CRVE} \not\approx CI_{WCRB}$

Example 3: Two regressors

```
. wildbootstrap regress wage tenure age, cluster(industry) rseed(12345)

Performing 1,000 replications for p-value for tenure = 0 ...
Computing confidence interval for tenure
  Lower bound: .........10.........20.. done (22)
  Upper bound: .........10.........20.... done (24)

Performing 1,000 replications for p-value for age = 0 ...
Computing confidence interval for age
  Lower bound: .........10.........20.... done (24)
  Upper bound: .........10.........20....... done (27)

Wild cluster bootstrap                          Number of obs      = 2,217
Linear regression                               Number of clusters =    12
                                                Cluster size:
Cluster variable: industry                                 min =     4
Error weight: Rademacher                                   avg = 184.8
                                                           max =   817
```

wage	Estimate	t	p-value	[95% conf.	interval]
constraints					
tenure = 0	.1869715	7.18	0.000	.13478	.3280472
age = 0	-.0946592	-2.55	0.006	-.2232396	-.0279091

Outline

Conclusion

1. It's crucial to adjust standard errors when dealing with clustered data.
   - CI's can be very misleading otherwise
   - Specially in experimental designs with treatment by clusters.
2. When clusters are many and homogeneous:
   - CRVE: `vce(cluster` *cvar*`)`
3. When clusters are few and heterogeneous:
   - Adjust degrees of freedom: `vce(hc2` *cvar*`, dfadjust)`
   - Wild cluster bootstrap: `wildbootstrap`

Learning more...

1. `help` command
   - Access to all our documentation.
2. www.stata.com
   - Access to all our documentation;
   - Frequently asked questions.
3. www.youtube.com/@statacorp/featured
4. tech-support@stata.com
   - Specific questions about our software.

References

1. When should you adjust standard errors for clustering?
   - Abadie, Athey, Imbens, and Wooldridge (NBER, Working paper)
2. How much should we trust differences-in-differences estimates?
   - Esther Duflo (QJE)
3. Bootstrap-based improvements for inference with clustered errors
   - Cameron, Gelbach, and Miller (ReStat)
4. Bias reduction in standard errors for linear regression with multi-stage samples
   - Bell and McCaffrey (Survey Methodology 2002)

Thank you!