Durham University
Business School

Department of Economics & Finance

vce2way: A one-stop solution for two-way clustered standard errors

Hong Il Yoo

Working Paper No. 01, 2017

Department Economics and Finance
Durham University Business School
Mill Hill Lane
Durham DH1 3LB, UK
Tel: +44 (0)191 3345200
https://www.dur.ac.uk/business/research/economics/

The Stata Journal (*yyyy*) *vv*, Number *ii*, pp. 1–8

vce2way: A one-stop solution for two-way clustered standard errors

Hong Il Yoo
Durham University Business School
Durham University
Durham, UK
h.i.yoo@durham.ac.uk

Abstract. Using `ereturn repost`, it is possible to write a simple `eclass` program which adjusts an existing command's standard errors for two-way clustering. This approach works with any command which allows `vce(cluster` *varname*`)` as an option, and the results are compatible with the command's postestimation tools making use of `e(V)`. A new command `vce2way` automates this approach.

Note: This working paper is not a Stata Journal publication. The associated software component can be downloaded by typing `ssc install vce2way` at Stata's command prompt.

Keywords: st0001, vce2way, ivreg2, cmgreg, two-way clustering, robust inference

1 Introduction

The use of (one-way) clustered standard errors in empirical research is now commonplace. In comparison with usual sandwich-type, including heteroskedasticity-robust, standard errors which assume the independence of regression errors across all observations, clustered standard errors offer an extra layer of robustness by allowing for arbitrary correlations across observations which belong to the same cluster. In the analysis of a labor force panel survey, for example, repeated observations on an individual may form a cluster, and standard errors may be clustered at the individual level to attain robustness to within-individual autocorrelations.

In recent years, the use of two-way clustered standard errors has also received growing attention. This extension robustifies one-way clustered standard errors further, by allowing for second and non-nested clusters within which regression errors may be correlated. In the analysis of panel data on bilateral trade flows, for instance, two-way clustering allows the researcher to robustify standard errors to autocorrelations within the same country as well as the same pair of countries. Cameron et al. (2011) summarize this and other potential areas of applications.

Almost all Stata commands allow `vce(cluster` *varname*`)` as an option, facilitating and popularizing the use of one-way clustered standard errors. No similar one-stop solution is available for two-way clustering. Some user-written commands, such as `cmgreg` (Gelbach and Miller, 2009) and `ivreg2` (Baum et al., 2010), support two-way clustering but only in the context of specific procedures (e.g. `cmgreg` in the context of

`regress` and `ivreg2` in the context of `ivregress`). In general, for each procedure, the researcher interested in two-way clustering must conduct a new search for a relevant user-written command which may or may not exist.

This article introduces a simple approach to obtain almost any Stata command's output with two-way clustered standard errors, and a new command `vce2way` which automates the approach. In a nutshell, this approach uses `ereturn repost` to replace an active output's variance-covariance matrix, stored in `e(V)`, with its two-way clustered counterpart that has been computed by the researcher. The researcher can use this approach to adjust a Stata command's standard errors for two-way clustering, whenever they can use `vce(cluster` *varname*`)` to adjust for one-way clustering in each relevant dimension separately.

2 A one-stop Stata solution for two-way clustering

As a working example, consider the following situation. The researcher is interested in running a probit regression of binary outcome `y` on two regressors `X` and `Z`, by executing `probit y X Z`. She is also interested in clustering standard errors in two non-nested dimensions, identified by variables `id1` and `id2` respectively. For a reason to become obvious, she has generated a derived identifier `id12` which identifies unique pairs comprising one cluster in `id1` and one cluster in `id2`, by executing: `egen id12 = group(id1 id2)`. Example data file `vce2way.dta` illustrates this situation.

The task of computing a two-way clustered variance-covariance matrix in Stata is something that many users may find straightforward. As Cameron et al. (2011) show, two-way clustered matrix V_{twoway} can be derived as

$$V_{twoway} = V_1 + V_2 - V_{12} \tag{1}$$

where V_1, V_2 and V_{12} are variance-covariance matrices adjusted for one-way clustering in `id1`, `id2`, and `id12` respectively. Two-way clustered standard errors are the square roots of the diagonal elements of V_{twoway}. In Stata, the variance-covariance matrix of active estimation results is saved in `ereturn` matrix `e(V)`. The following series of self-explanatory commands uses `e(V)` to implement equation (1), and stores the resulting V_{twoway} in matrix `V_2way`.

```
. use vce2way.dta, clear
.
. probit y X Z, vce(cluster id1)
[output omitted]
. matrix V1 = e(V)
.
. probit y X Z, vce(cluster id2)
[output omitted]
. matrix V2 = e(V)
.
. probit y X Z, vce(cluster id12)
Iteration 0:   log pseudolikelihood = -676.12234
```

```
[output omitted]
Iteration 4:   log pseudolikelihood = -369.74412
Probit regression                                 Number of obs     =      1,000
                                                  Wald chi2(2)      =     277.29
                                                  Prob > chi2       =     0.0000
Log pseudolikelihood = -369.74412                 Pseudo R2         =     0.4531
                                 (Std. Err. adjusted for 962 clusters in id12)
------------------------------------------------------------------------------
             |               Robust
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           X |    1.113323   .0752924    14.79   0.000     .9657522    1.260893
           Z |    1.081183   .0758083    14.26   0.000     .9326011    1.229764
       _cons |   -.4574496   .0555331    -8.24   0.000    -.5662924   -.3486068
------------------------------------------------------------------------------

. matrix V12 = e(V)

.
. matrix V_2way = V1 + V2 - V12
```

A more tricky task is to have Stata's existing postestimation tools make use of `V_2way`. And completing this task is crucial. Otherwise, to carry out statistical inferences with two-way clustering, the researcher will have to reinvent the wheel by coding a host of self-written programs to compute z statistics, 95% confidence intervals, the standard errors of average partial effects, and so on.

Stata's programming command `ereturn repost` will do the required trick here. This command is designed to aid programmers to update the contents of `ereturn` results `e(b)` and `e(V)` when coding their own estimation routines. Below, this command is used for a minimal purpose of replacing `e(V)` of active estimation results with matrix `V_2way`. The active results in the present example is `probit y X Z, cluster(id12)`.

```
. capture program drop mytwoway

. program define mytwoway, eclass
  1.         ereturn repost V = V_2way
  2.         probit
  3. end

. mytwoway
Probit regression                                 Number of obs     =      1,000
                                                  Wald chi2(2)      =     277.29
                                                  Prob > chi2       =     0.0000
Log pseudolikelihood = -369.74412                 Pseudo R2         =     0.4531
                                 (Std. Err. adjusted for 962 clusters in id12)
------------------------------------------------------------------------------
             |               Robust
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           X |    1.113323   .1072957    10.38   0.000      .903027    1.323618
           Z |    1.081183    .091039    11.88   0.000     .9027495    1.259616
       _cons |   -.4574496   .0986657    -4.64   0.000    -.6508308   -.2640684
------------------------------------------------------------------------------

. test X = Z = 0
 ( 1)  [y]X - [y]Z = 0
 ( 2)  [y]X = 0
```

```
              chi2(  2) =  151.39
            Prob > chi2 =    0.0000
```

The new `eclass` program `mytwoway` comprises two simple steps. The first step uses `ereturn repost` replaces `e(V)` with `V_2way` as described. The second step executes `probit` without any further specification, to display the updated estimation results that use `V_2way` to compute two-way clustered standard errors and associated 95% confidence intervals. The preceding example is intended to be minimal and does not include several extra command lines for updating other summary information. In consequence, the information on the number of clusters, `Wald chi2(2)` and `Prob > chi2` still refers to the case of one-way clustering in `id12`. However, as the comparisons of standard errors and 95% confidence intervals suggest, the underlying variance-covariance matrix `e(V)` is now `V_2way`, which has been adjusted for two-way clustering in `id1` and `id2`. Henceforth, all postestimation tools will produce standard errors and/or test statistics that have been adjusted for two-way clustering. The preceding example illustrates the use of `test` command to obtain the updated `Wald chi2(2)` statistic for the joint significance of `X` and `Y`. The resulting statistic is 151.39, quite a departure from 277.29 based on one-way clustering in `id12`.

Note that this approach is generally applicable to every command which allows `vce(cluster` *varname*`)` as an option. The preceding example does not make use of any feature specific to `probit` command, and the researcher can replace `probit` command lines with other command lines of interest including the like of `ml lf myprogram, maximize` for a self-written evaluator. The updated results are compatible with the underlying command's postestimation tools which make use of `e(V)`, such as `test`, `margins`, `nlcom` to name a few.

One drawback of this approach, as implemented above, is that it involves executing the same optimization task three times. For commands which can be executed within a few seconds, this drawback is unlikely to be an issue. For commands which require potentially substantial computer time, such as `asroprobit`, this drawback can be a source of major inconvenience if not practical infeasibility. Fortunately, however, most of such commands allow `from()` or `init()` as an option. The researcher can minimize inconvenience by running a full optimization run only once, and then using the resulting coefficient estimates as starting values for all subsequent optimization runs. Though `probit` can be executed in the blink of an eye, the preceding example may be modified as follows for illustration.

```
. probit y X Z
[output omitted]
. matrix b = e(b)
.
. probit y X Z, vce(cluster id1) from(b)
[output omitted]
. matrix V1 = e(V)
.
. probit y X Z, vce(cluster id2) from(b)
[output omitted]
```

```
. matrix V2 = e(V)
.
. probit y X Z, vce(cluster id12) from(b)
[output omitted]
. matrix V12 = e(V)
.
```

Cameron et al. (2011) point out that in some applications, V_{twoway} computed using equation (1) may not be positive semi-definite. As a solution, they suggest that the researcher may replace negative eigenvalues of V_{twoway} with 0s, and reconstruct the variance-covariance matrix using the updated eigenvalues and the original eigenvectors. The following three command lines execute this solution in Mata, and can be inserted after computing `V_2way` and before issuing `program define mytwoway, eclass`. Note that in the preceding example, the said problem did not arise. This means that the reconstructed `V_2way` and the original `V_2way` are the same, as the researcher can verify by issuing `matrix list V_2way` before and after executing the Mata commands.

```
. mata: symeigensystem(st_matrix("V_2way"), EVEC = ., eval = .)
. mata: eval = eval :* (eval :> 0)
. mata: st_matrix("V_2way", EVEC*diag(eval)*EVEC´)
```

3 Command vce2way

`vce2way` is a new user-written command which automates the two-way clustering approach described in Section 2. The command includes an extra step for checking whether one-way clustering in different identifiers result in the same coefficient vector: if not, for example due to missing values in some identifier, it will abort with a telltale error message. It also incorporates extra command lines to display notes which clarify that the standard errors have been adjusted for two-way clustering and, where applicable, negative eigenvalues have been replaced with zeroes.

The syntax for `vce2way` is

`vce2way` *cmdline_main*, `cluster(`*varname1 varname2*`)` [*cmdline_options*]

In the required option `cluster()`, *varname1* and *varname2* are the names of variables identifying two clustering dimensions. In the remaining syntax diagram, *cmdline_main* (*cmdline_options*) is the non-optional (optional) component of the command line to execute a procedure for which two-way clustering is requested. In the context of Section 2, *varname1* and *varname2* are `id1` and `id2`; *cmdline_main* is `probit y X Z`; and *cmdline_options* is either blank or `from(b)`.

Perhaps an easier way to understand `vce2way`'s syntax diagram is by posing the following question. If Stata had a built-in `cluster(`*varname1 varname2*`)` option to request two-way clustering, what command line would the researcher specify? Prefixing the answer to this question by `vce2way` satisfies the syntax requirements.

Small sample bias corrections that `vce2way` makes to variance-covariance matrices are implicit in the Stata implementation of equation (1), which has been documented in Section 2. Like `cgmreg` of Gelbach and Miller (2009), `vce2way` applies the first correction method of Cameron et al. (2011, p.241) that adjusts each of three one-way clustered matrices in equation (1) separately according to the number of clusters affecting that matrix. As Baum et al. (2010) point out, their `ivreg2` applies the second method of Cameron et al. (2011, p.241) that adjusts all three matrices by a common factor based on the number of clusters in either *varname1* or *varname2*, depending on which is smaller. This potential variation in bias correction methods should be kept in mind when comparing the output of `vce2way` with that of other user-written commands which implement two-way clustering.

3.1 Examples

The `probit` example in Section 2 can be replicated using `vce2way` as follows. The standard errors and confidence intervals remain unchanged, but the output table includes extra information to facilitate interpretation. By default, `vce2way` suppresses iteration logs. The researcher can request iteration logs by executing `vce2way noisily probit y X Z, cluster(id1 id2)` instead.

```
. use vce2way.dta, clear
. vce2way probit y X Z, cluster(id1 id2)
Probit regression                                 Number of obs     =      1,000
                                                  Wald chi2(2)      =          .
                                                  Prob > chi2       =          .
Log pseudolikelihood = -369.74412                 Pseudo R2         =     0.4531
                          (Std. Err. adjusted for clustering on id1 and id2)
------------------------------------------------------------------------------
             |               Robust
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           X |   1.113323   .1072957    10.38   0.000      .903027    1.323618
           Z |   1.081183    .091039    11.88   0.000     .9027495    1.259616
       _cons |  -.4574496   .0986657    -4.64   0.000    -.6508308   -.2640684
------------------------------------------------------------------------------
Notes:
    Std. Err. adjusted for 143 clusters in id1, AND 100 clusters in id2.
    Ignore default Wald chi2(.) and Prob > chi2, or F(.,.) and Prob > F, results above.
      If needed, use command -test- to compute the test statistic and p-value of interest.
```

As discussed earlier, the underlying approach executes the same optimization task three times to compute three one-way clustered variance-covariance matrices in equation (1). If needed, the researcher may save computer run time by using `from()` or `init()` option as follows, to start each optimization run from an optimal solution computed prior to executing `vce2way`.

```
. probit y X Z
[output omitted]
. matrix b = e(b)
. vce2way probit y X Z, cluster(id1 id2) from(b)
```

```
[output omitted]
```

In the analysis of panel data, the researcher may be interested in clustering standard errors in panel units as well as time periods. As an example, consider a random effects linear regression of an individual's log-wage (`ln_wage`) on their age (`age`) and years of education (`grade`). The following command lines execute such a regression, while clustering standard errors in individuals (identified by `idcode`) and time periods (identified by `year`). Note the use of `xtreg`'s undocumented (as of Stata 14.2) option `nonest`. Without this option, `xtreg` will abort with an error when one-way clustering in `year` is requested since the group variable `idcode` is not nested within `year`. Note also that two-way clustered standard errors are consistent when the sizes of both clustering dimensions become arbitrarily large (Cameron et al., 2011). As Baum et al. (2010) comment in the context of an `ivreg2` application using the same data set, the results below may need to be interpreted with some caution because the number of clusters in `year` (i.e. time periods) is rather small, specifically 15.

```
. webuse nlswork, clear
(National Longitudinal Survey.  Young Women 14-26 years of age in 1968)

. vce2way xtreg ln_w grade age, cluster(idcode year) re nonest
Random-effects GLS regression                   Number of obs     =     28,508
Group variable: idcode                          Number of groups  =      4,708

R-sq:                                           Obs per group:
     within  = 0.1026                                         min =          1
     between = 0.3211                                         avg =        6.1
     overall = 0.2318                                         max =         15

                                                Wald chi2(2)      =          .
corr(u_i, X)   = 0 (assumed)                    Prob > chi2       =          .

                        (Std. Err. adjusted for clustering on idcode and year)
------------------------------------------------------------------------------
             |               Robust
     ln_wage |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       grade |   .0809639   .0057431    14.10   0.000     .0697075    .0922202
         age |   .0172326   .0009896    17.41   0.000     .0152931    .0191722
       _cons |    .131286   .0682674     1.92   0.054    -.0025157    .2650876
-------------+----------------------------------------------------------------
     sigma_u |  .30563456
     sigma_e |  .30349389
         rho |  .50351427   (fraction of variance due to u_i)
------------------------------------------------------------------------------

Notes:
    Std. Err. adjusted for 4708 clusters in idcode, AND 15 clusters in year.

    Ignore default Wald chi2(.) and Prob > chi2, or F(.,.) and Prob > F, results above.
      If needed, use command -test- to compute the test statistic and p-value of interest.
```

In the preceding examples, equation (1) has led to positive semi-definite variance-covariance matrices V_{twoway}. Following Cameron et al. (2011), in case V_{twoway} is not positive semi-definite, `vce2way` reconstructs it as a positive semi-definite matrix by replacing negative eigenvalues with 0s. If such an operation has taken place, `vce2way` will issue an appropriate notice. The following OLS regression of `y` on `x` illustrates this feature, using a simulated data set accompanying user-written command `cgmreg`

(Gelbach and Miller, 2009).

```
. use cgmreg.dta, clear
. vce2way reg y x, cluster(clu_id_1 clu_id_2)
Linear regression                               Number of obs     =        100
                                                F(1, 99)          =     438.32
                                                Prob > F          =     0.0000
                                                R-squared         =     0.8376
                                                Root MSE          =      1.025
                (Std. Err. adjusted for clustering on clu_id_1 and clu_id_2)
------------------------------------------------------------------------------
             |               Robust
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |   2.056308    .039556    51.98   0.000      1.97782    2.134795
       _cons |   .1058816    .0847436    1.25   0.214    -.0622682    .2740313
------------------------------------------------------------------------------
Notes:
    Std. Err. adjusted for 10 clusters in clu_id_1, AND 10 clusters in clu_id_2.
    Ignore default Wald chi2(.) and Prob > chi2, or F(.,.) and Prob > F, results above.
      If needed, use command -test- to compute the test statistic and p-value of interest.
    The initial variance-covariance matrix, e(V_raw), was not positive semi-definite.
      The final matrix, e(V), was computed by replacing negative eigenvalues with 0s.
```

4 Acknowledgments

I would like to thank Yi Gu and Kenju Kamei for alerting me to the literature on two-way clustering and helpful discussion.

5 References

Baum, C.F., M.E. Schaffer, and S. Stillman. 2010. ivreg2: Stata module for extended instrumental variables/2SLS, GMM and AC/HAC, LIML and k-class regression. http://ideas.repec.org/c/boc/bocode/s425401.html.

Cameron, A. C., J.B. Gelbach, and D.L. Miller. 2011. Robust Inference With Multiway Clustering. *Jorunal of Business and Economic Statistics* 29(2): 238-249.

Gelbach, J.B., and D.L. Miller. 2009. Multi-way clustering with OLS. http://faculty.econ.ucdavis.edu/faculty/dlmiller/statafiles/.

About the authors

Hong Il Yoo is a Lecturer in Economics at Durham University Business School, Durham University, UK.