

# Robust Standard Errors in Small Samples

Michal Kolesár

August 10, 2021

## Contents

<b>Description</b>	<b>1</b>
<b>Methods</b>	<b>3</b>
Variance estimate . . . . .	4
Degrees of freedom correction . . . . .	5

## Description

This package implements small-sample degrees of freedom adjustments to robust and cluster-robust standard errors in linear regression, as discussed in Imbens and Kolesár [2016]. The implementation can handle models with fixed effects, and cases with a large number of observations or clusters<sup>1</sup>.

```
library(dfadjust)
```

To give some examples, let us construct an artificial dataset with 11 clusters

```
set.seed(7)
d1 <- data.frame(y = rnorm(1000), x1 = c(rep(1, 3), rep(0,
  997)), x2 = c(rep(1, 150), rep(0, 850)), x3 = rnorm(1000),
  cl = as.factor(c(rep(1:10, each = 50), rep(11, 500))))
```

Let us first run a regression of  $y$  on  $x_1$ . This is a case in which, in spite of moderate data size, the effective number of observations is small since there are only three treated units:

```
r1 <- lm(y ~ x1, data = d1)
## No clustering
dfadjustSE(r1)
#>
#> Coefficients:
#>           Estimate HC1 se HC2 se Adj. se      df p-value
#> (Intercept)  0.00266 0.0311  0.031 0.0311 996.00  0.932
#> x1          0.12940 0.8892  1.088 2.3743   2.01  0.916
```

We can see that the usual robust standard errors (HC1 se) are much smaller than the effective

---

<sup>1</sup>We thank Ulrich Müller for suggesting to us the lemma below.

standard errors (Adj. se), which are computed by taking the HC2 standard errors and applying a degrees of freedom adjustment.

Now consider a cluster-robust regression of  $y$  on  $x_2$ . There are only 3 treated clusters, so the effective number of observations is again small:

```
r1 <- lm(y ~ x2, data = d1)
# Default Imbens-Kolesár method
dfadjustSE(r1, clustervar = d1$c1)
#>
#> Coefficients:
#>           Estimate HC1 se HC2 se Adj. se    df p-value
#> (Intercept) -0.0236 0.0135 0.0169  0.0222 4.94  0.2215
#> x2           0.1778 0.0530 0.0621  0.1157 2.43  0.0826
# Bell-McCaffrey method
dfadjustSE(r1, clustervar = d1$c1, IK = FALSE)
#>
#> Coefficients:
#>           Estimate HC1 se HC2 se Adj. se    df p-value
#> (Intercept) -0.0236 0.0135 0.0169  0.0316 2.42  0.2766
#> x2           0.1778 0.0530 0.0621  0.1076 2.70  0.0731
```

Now, let us run a regression of  $y$  on  $x_3$ , with fixed effects. Since we're only interested in  $x_3$ , we specify that we only want inference on the second element:

```
r1 <- lm(y ~ x3 + c1, data = d1)
dfadjustSE(r1, clustervar = d1$c1, ell = c(0, 1, rep(0,
  r1$rank - 2)))
#>
#> Coefficients:
#>           Estimate HC1 se HC2 se Adj. se    df p-value
#> Estimate    0.0261 0.0463 0.0595  0.0928 3.23  0.688
dfadjustSE(r1, clustervar = d1$c1, ell = c(0, 1, rep(0,
  r1$rank - 2)), IK = FALSE)
#>
#> Coefficients:
#>           Estimate HC1 se HC2 se Adj. se    df p-value
#> Estimate    0.0261 0.0463 0.0595  0.0928 3.23  0.688
```

Finally, an example in which the clusters are large. We have 500,000 observations:

```
d2 <- do.call("rbind", replicate(500, d1, simplify = FALSE))
d2$y <- rnorm(length(d2$y))
r2 <- lm(y ~ x2, data = d2)
summary(r2)
#>
#> Call:
#> lm(formula = y ~ x2, data = d2)
#>
#> Residuals:
```

```

#>      Min      1Q  Median      3Q      Max
#> -5.073 -0.675  0.000  0.675  4.789
#>
#> Coefficients:
#>              Estimate Std. Error t value Pr(>|t|)
#> (Intercept) -0.000991   0.001535  -0.65    0.52
#> x2          -0.003590   0.003963  -0.91    0.37
#>
#> Residual standard error: 1 on 499998 degrees of freedom
#> Multiple R-squared:  1.64e-06,   Adjusted R-squared:  -3.59e-07
#> F-statistic: 0.821 on 1 and 5e+05 DF,  p-value: 0.365
# Default Imbens-Kolesár method
dfadjustSE(r2, clustervar = d2$c1)
#>
#> Coefficients:
#>              Estimate HC1 se  HC2 se Adj. se   df p-value
#> (Intercept) -0.000991 0.00133 0.00168 0.00294 2.66   0.603
#> x2          -0.003590 0.00483 0.00568 0.00997 2.65   0.578
# Bell-McCaffrey method
dfadjustSE(r2, clustervar = d2$c1, IK = FALSE)
#>
#> Coefficients:
#>              Estimate HC1 se  HC2 se Adj. se   df p-value
#> (Intercept) -0.000991 0.00133 0.00168 0.00315 2.42   0.607
#> x2          -0.003590 0.00483 0.00568 0.00984 2.70   0.577

```

## Methods

This section describes the implementation of the Imbens and Kolesár [2016] and Bell and McCaffrey [2002] degrees of freedom adjustments.

There are  $S$  clusters, and we observe  $n_s$  observations in cluster  $s$ , for a total of  $n = \sum_{s=1}^S n_s$  observations. We handle the case with independent observations by letting each observation be in its own cluster, with  $S = n$ . Consider the linear regression of a scalar outcome  $Y_i$  onto a  $p$ -vector of regressors  $X_i$ ,

$$Y_i = X_i' \beta + u_i, \quad E[u_i | X_i] = 0.$$

We're interested in inference on  $\ell' \beta$  for some fixed vector  $\ell \in \mathbb{R}^p$ . Let  $X$ ,  $u$ , and  $Y$  denote the design matrix, and error and outcome vectors, respectively. For any  $n \times k$  matrix  $M$ , let  $M_s$  denote the  $n_s \times k$  block corresponding to cluster  $s$ , so that, for instance,  $Y_s$  corresponds to the outcome vector in cluster  $s$ . For a positive semi-definite matrix  $M$ , let  $M^{1/2}$  be a matrix satisfying  $M^{1/2'} M^{1/2} = M$ , such as its symmetric square root or its Cholesky decomposition.

Assume that

$$E[u_s u_s' | X] = \Omega_s, \quad \text{and} \quad E[u_s u_t' | X] = 0 \quad \text{if } s \neq t.$$

Denote the conditional variance matrix of  $u$  by  $\Omega$ , so that  $\Omega_s$  is the block of  $\Omega$  corresponding to cluster  $s$ . We estimate  $\ell' \beta$  using OLS. In R, the OLS estimator is computed via a QR decomposition,

$X = QR$ , where  $Q'Q = I$  and  $R$  is upper-triangular, so we can write the estimator as

$$\ell' \hat{\beta} = \ell' \left( \sum_s X_s' X_s \right)^{-1} \sum_s X_s' Y_s = \tilde{\ell}' \sum_s Q_s' Y_s, \quad \tilde{\ell} = R^{-1} \ell.$$

It has variance

$$V := \text{var}(\ell' \hat{\beta} \mid X) = \ell' (X'X)^{-1} \sum_s X_s' \Omega_s X_s (X'X)^{-1} \ell = \tilde{\ell}' \sum_s Q_s' \Omega_s Q_s \tilde{\ell}.$$

## Variance estimate

We estimate  $V$  using a variance estimator that generalizes the HC2 variance estimator to clustering. Relative to the LZ2 estimator described in Imbens and Kolesár [2016], we use a slight modification that allows for fixed effects:

$$\hat{V} = \ell' (X'X)^{-1} \sum_s X_s' A_s \hat{u}_s \hat{u}_s' A_s' X_s (X'X)^{-1} \ell = \ell' R^{-1} \sum_s Q_s' A_s \hat{u}_s \hat{u}_s' A_s' Q_s R^{-1} \ell = \sum_{s=1}^S (\hat{u}_s' a_s)^2,$$

where

$$\hat{u}_s := Y_s - X_s \hat{\beta} = u_s - Q_s Q' u, \quad a_s = A_s' Q_s \tilde{\ell},$$

and the matrix  $A_s$  is given by the symmetric square root of the inverse of  $I - Q_s Q_s'$ , or else its pseudo-inverse if it is singular, as is the case, for example, if  $X$  contains fixed effects. We do not need to insist on  $I - Q_s Q_s'$  to be invertible, since, using the identity

$$\hat{V} = u \sum_s (I - Q Q')_s' a_s a_s' (I - Q Q')_s u,$$

one can verify by simple algebra that a sufficient condition for  $\hat{V}$  to be unbiased under homoskedasticity is that  $Q_s' A_s (I - Q_s Q_s') A_s Q_s = Q_s' Q_s$  (see, for example, Pustejovsky and Tipton [2018], for details).

If the observations are independent, the vector of leverages  $(Q_1' Q_1, \dots, Q_n' Q_n)$  can be computed directly using the `stats::hatvalues` function. In this case, use this function to compute  $A_i = 1/\sqrt{1 - Q_i' Q_i}$  directly, and we then compute  $a_i = A_i Q_i' \tilde{\ell}$  using vector operations. For the case with clustering, computing the spectral decomposition of  $I - Q_s Q_s'$  can be expensive or even infeasible if the cluster size  $n_s$  is large. We therefore use the following result, suggested to us by Ulrich Müller, which allows us to compute  $a_s$  by computing a spectral decomposition of a  $p \times p$  matrix.

- Let  $Q_s' Q_s = \sum_{i=1}^p \lambda_{is} r_{is} r_{is}'$  be the spectral decomposition of  $Q_s' Q_s$ . Then  $A_s = \sum_{i: \lambda_i \neq 1} (1 - \lambda_i)^{-1/2} Q_s r_{is} r_{is}' Q_s'$ , satisfies  $A_s (I - Q_s Q_s') A_s = I$ .

This follows from the fact that  $I - Q_s Q_s'$  has eigenvalues  $1 - \lambda_{is}$  and eigenvectors  $Q_s r_{is}$ , and hence its pseudoinverse is  $\sum_{i: \lambda_i \neq 1} (1 - \lambda_i)^{-1} Q_s r_{is} r_{is}' Q_s'$ .

Using the lemma, we can compute  $a_s$  efficiently as:

$$a_s = \sum_{i: \lambda_i \neq 1} (1 - \lambda_i)^{-1/2} Q_s r_{is} r_{is}' Q_s' \tilde{\ell} = Q_s D_s \tilde{\ell}, \quad D_s = \sum_{i: \lambda_i \neq 1} \lambda_i (1 - \lambda_i)^{-1/2} r_{is} r_{is}'.$$

## Degrees of freedom correction

Let  $G$  be an  $n \times S$  matrix with columns  $(I - QQ')'_s a_s$ . Then the Bell and McCaffrey [2002] adjustment sets the degrees of freedom to

$$f_{\text{BM}} = \frac{\text{tr}(G'G)^2}{\text{tr}((G'G)^2)}.$$

Since  $(G'G)_{st} = a'_s(I - QQ')_s(I - QQ')'_t a_t = a_s(\mathbb{1}\{s = t\} - Q_s Q'_t) a_t$ , the matrix  $G'G$  can be efficiently computed as

$$G'G = \text{diag}(a'_s a_s) - BB' \quad B_{sk} = a'_s Q_{sk}.$$

Note that  $B$  is an  $S \times p$  matrix, so that computing the degrees of freedom adjustment only involves  $p \times p$  matrices:

$$f_{\text{BM}} = \frac{(\sum_s a'_s a_s - \sum_{s,k} B_{sk}^2)^2}{\sum_s (a'_s a_s)^2 - 2 \sum_{s,k} (a'_s a_s) B_{sk}^2 + \sum_{s,t} (B'_s B_t)^2}.$$

If the observations are independent, we compute  $B$  directly as  $B \leftarrow a * Q$ , and since  $a_i$  is a scalar, we have

$$f_{\text{BM}} = \frac{(\sum_i a_i^2 - \sum_{sk} B_{sk}^2)^2}{\sum_i a_i^4 - 2 \sum_i a_i^2 B'_i B_i + \sum_{i,j} (B'_i B_j)^2}.$$

The Imbens and Kolesár [2016] degrees of freedom adjustment instead sets

$$f_{\text{IK}} = \frac{\text{tr}(G' \hat{\Omega} G)^2}{\text{tr}((G' \hat{\Omega} G)^2)},$$

where  $\hat{\Omega}$  is an estimate of the Moulton [1986] model of the covariance matrix, under which  $\Omega_s = \sigma_\epsilon^2 I_{n_s} + \rho \iota_{n_s} \iota'_{n_s}$ . Using simple algebra, one can show that in this case,

$$G' \Omega G = \sigma_\epsilon^2 \text{diag}(a'_s a_s) - \sigma_\epsilon^2 BB' + \rho (D - BF')(D - BF')',$$

where

$$F_{sk} = \iota'_{n_s} Q_{sk}, \quad D = \text{diag}(a'_s \iota_{n_s})$$

which can again be computed even if the clusters are large. The estimate  $\hat{\Omega}$  replaces  $\sigma_\epsilon^2$  and  $\rho$  with analog estimates.

## References

- Robert M. Bell and Daniel F. McCaffrey. Bias reduction in standard errors for linear regression with multi-stage samples. *Survey Methodology*, 28(2):169–181, December 2002.
- Guido W. Imbens and Michal Kolesár. Robust standard errors in small samples: Some practical advice. *Review of Economics and Statistics*, 98(4):701–712, October 2016. doi: 10.1162/REST\_a\_00552.
- Brent R. Moulton. Random group effects and the precision of regression estimates. *Journal of Econometrics*, 32(3):385–397, August 1986. doi: 10.1016/0304-4076(86)90021-7.
- James E. Pustejovsky and Elizabeth Tipton. Small-sample methods for cluster-robust variance estimation and hypothesis testing in fixed effects models. *Journal of Business & Economic Statistics*, 36(4):672–683, October 2018. doi: 10.1080/07350015.2016.1247004.