Journal of Statistical Software

November 2020, Volume tba, Issue tba. *doi: 10.18637/jss.v000.i00*

spacereg: A Spatial Standard Errors Implementation for Several Commonly Used M-estimators

Luis Calderon
Universität Bonn

Leander Heldring
Institute on Behavior & Inequality (briq)

Abstract

We present the asymptotic variance-covariance matrix for M-estimators, and show how it can be used to compute spatial standard errors for a large number of commonly used (non-linear) estimators. We consider OLS, Logit, Probit estimators, Poisson and Negative Binomial regressions, and the special Stata estimators `areg` and `regdhfe`. We provide Stata and Python software to implement our findings.

Keywords: Spatial standard errors, causal inference, Stata, Python.

1. Introduction

Real-world data generation processes may exhibit spatial correlation. The seminal contribution of Conley (1999) has enabled applied researchers to compute 'Conley standard errors', which allow for arbitrary spatial correlation in residuals. In this note, we show how his methods can be extended to several commonly used non-linear estimators, and provide a software implementation. At its core, Conley (1999)'s method provides a HAC estimator, allowing for heteroskedasticity and n-dimensional auto-correlation in space for some distance metric. In practice, researchers have focused on the two-dimensional case relying on geographical coordinates to determine distance. Beyond a cutoff, spatial dependence is assumed to be zero.

Conley (1999) introduced his estimator for the Generalized Method of Moments (GMM), and provided Stata code for GMM with linear moment conditions, and Ordinary Least Squares (OLS). In this note, we present a Stata and Python implementation for the construction of asymptotic, spatially dependent variance-covariance matrices for general M-estimators, following Jenish and Prucha (2009). Since the asymptotic variance of M-estimators takes a simple sandwich form, we can reweigh the 'filling' part of a consistent estimator for the asymptotic variance following Conley (1999), and compute spatial standard errors for any M-estimator.[1] We then apply this estimator to several commonly used models: a linear model of a continuous dependent variable, the logit and probit models for binary dependent variables, Poisson and negative-binomial generalized linear models for count dependent variables, and the special `Stata` estimators `areg` and `reghdfe`.

The theory for the asymptotic distribution of M-estimators is developed in Amemiya (1985), Newey and McFadden (1994) and was extended to the spatial case by Jenish and Prucha (2009). Our contribution is in the concise presentation of these results, the application of the theoretical result in Jenish and Prucha (2009) to commonly used estimators, and their software implementation. Our presentation relies primarily on Newey and McFadden (1994) and Bester, Conley, and Hansen (2011).

In the next section, we review the asymptotic variance for M-estimators. In section 3, we present the M-estimators for the commonly used models we study in this paper. In section 4, we rely on the fact that Jenish and Prucha (2009) showed that the derived asymptotic variance-covariance matrix for M-estimators is consistent under spatial dependence to apply the method of Conley (1999) to M-estimators. We arrive at a simple general formulation for spatial standard errors for M-estimators. In section 5 we discuss implementation. Section 6 concludes.

2. M-estimators

Consider a population $\boldsymbol{q} = (y,\ \boldsymbol{x})$, and a sample $\boldsymbol{q}_i$, $i = 1, 2, ..., N$. $\boldsymbol{x}$ is a K vector of explanatory variables and y is an outcome of interest. $\boldsymbol{x}$ also includes two coordinates, which determine the location of i in space, h_i and v_i. Most commonly, such coordinates are latitude and longitude. Consider a $K \times 1$ parameter vector $\theta \in \Theta \subseteq \mathbb{R}^K$, where Θ is the set of possible values for θ. Let $Q(\boldsymbol{q},\ \boldsymbol{\theta_0})$ be some function of the (random) matrix $\boldsymbol{q}$ and the true parameter vector θ_0.

In this setup, an M-estimator of θ_0 is:

$$\hat{\boldsymbol{\theta}} = \underset{\theta \in \Theta}{\arg\min}\, N^{-1} \sum_i Q(\boldsymbol{q_i}, \boldsymbol{\theta}) \tag{1}$$

Under the regularity conditions set out by Amemiya (1985), $\hat{\theta} \xrightarrow{p} \theta_0$ and

$$\sqrt{N}(\hat{\theta} - \theta_0) \xrightarrow{d} \mathbb{N}\left(0, \boldsymbol{A}_0^{-1}\boldsymbol{B}_0\boldsymbol{A}_0^{-1}\right) \tag{2}$$

Where, intuitively, $\boldsymbol{A_0}^{-1}$ is the inverse of $Q(\boldsymbol{q},\ \boldsymbol{\theta_0})$'s Hessian matrix, and $\boldsymbol{B}_0$ is the outer product of the gradient vectors, and $\mathbb{N}$ is the normal distribution. Formally:

$$\boldsymbol{B}_0 = \mathbb{E}[v(\boldsymbol{q}, \boldsymbol{\theta_0})' v(\boldsymbol{q}, \boldsymbol{\theta_0})] \tag{3}$$

[1]For our choice of M-estimators, the 'filling' is the variance of the total. See Stata *et al.* (2011), page 12.

Where $v(\boldsymbol{q_i}, \boldsymbol{\theta_0})' = \nabla_{\theta_0} Q(\boldsymbol{q_i}, \boldsymbol{\theta_0})$ is the gradient vector, or the vector of first derivatives of $Q(\boldsymbol{q_i}, \boldsymbol{\theta_0})$ with respect to $\boldsymbol{\theta_0}$. And

$$\boldsymbol{A}_0 = \mathbb{E}\left[\nabla^2_{\theta_0} Q(\boldsymbol{q}, \boldsymbol{\theta_0})\right] = \mathbb{E}\left(\frac{\partial^2 Q(\boldsymbol{q}, \boldsymbol{\theta_0})}{\partial \boldsymbol{\theta_0} \partial \boldsymbol{\theta_0}'}\right) \tag{4}$$

is the Hessian matrix of second derivatives. Because $\hat{\theta} \xrightarrow{p} \theta_0$, $\boldsymbol{A}_0$ and $\boldsymbol{B}_0$ can be consistently estimated by their sample equivalents.[2]

3. Selected M-estimators

We study several commonly used models for which M-estimators are available: the OLS estimator for a linear model, the logit and probit estimators for models with a binary dependent variable and the Poisson and negative binomial (NB2) models for count data.[3] We consider $\boldsymbol{q_i}$ with heteroskedastic disturbances and treat $\boldsymbol{x_i}$ as a $1 \times K$ row vector.

Linear model:

$$\begin{aligned} Q(\boldsymbol{q_i}, \boldsymbol{\theta}) &= [y_i - \boldsymbol{x_i}\boldsymbol{\theta}]^2 \\ v(\boldsymbol{q_i}, \boldsymbol{\theta}) &= -2\mathbb{E}\left[(y_i - \boldsymbol{x_i}\boldsymbol{\theta}) \cdot \boldsymbol{x_i}\right] \\ \boldsymbol{B}_0 &= 4\mathbb{E}\left[(\lambda \cdot \boldsymbol{x})'(\lambda \cdot \boldsymbol{x})\right] \\ \boldsymbol{A}_0 &= 2\mathbb{E}[\boldsymbol{x}'\boldsymbol{x}] \end{aligned} \tag{5}$$

where $\lambda_i = [y_i - \boldsymbol{x_i}\theta]$. This expression, which we will see return in for every model, is of course equivalent to the OLS residual. For other models, this expression has been termed the 'generalized residual', see Chesher, Lancaster, and Irish (1985) and Greene (2003). For the homoskedastic case, let σ^2 be the variance of $\boldsymbol{x_i}$. The (asymptotic) variance of $\hat{\theta}$ in this case is the familiar $N^{-1}\sigma^2\mathbb{E}[\boldsymbol{x}'\boldsymbol{x}]$.

Binary dependent variable: logit, let $c_i = \frac{e^{\boldsymbol{x_i}\boldsymbol{\theta}}}{1+e^{\boldsymbol{x_i}\boldsymbol{\theta}}}$:

$$\begin{aligned} Q(\boldsymbol{q_i}, \boldsymbol{\theta}) &= -\left[y_i(\boldsymbol{x_i}\boldsymbol{\theta}) - \ln(1 + e^{\boldsymbol{x_i}\boldsymbol{\theta}})\right] \\ v(\boldsymbol{q_i}, \boldsymbol{\theta}) &= \mathbb{E}\left[(y_i - c_i) \cdot \boldsymbol{x_i}\right] \\ \boldsymbol{B}_0 &= \mathbb{E}\left[(\lambda \cdot \boldsymbol{x})'(\lambda \cdot \boldsymbol{x})\right] \\ \boldsymbol{A}_0 &= -\mathbb{E}\left[c \cdot (1 - c) \cdot \boldsymbol{x}'\boldsymbol{x}\right] \end{aligned} \tag{6}$$

where $\lambda_i = [y_i - c_i]$.

Binary dependent variable: probit, let $c_i = \Phi\left(\boldsymbol{x_i}\boldsymbol{\theta}\right)$:

[2] Additional standard regularity conditions have to hold (Amemiya 1985). If so, then $\frac{1}{n}\sum_{i=1}^{n} v(\boldsymbol{q_i}, \hat{\boldsymbol{\theta}}) \xrightarrow{p} \mathbb{E}[v(\boldsymbol{q}, \boldsymbol{\theta_0})]$. The Hessian similarly converges to A_0.

[3] For OLS, see Greene (2003), page 583 and Wooldridge (2002). For logit and probit, see also Greene (2003), pages 731 and 732. For Poisson and negative binomial, see Hilbe (2011), page 84 and 192 respectively. The negative of the likelihood is considered an M-estimator. For more details, see Croux and Haesbroeck (2003).

$$
\begin{aligned}
Q(\boldsymbol{q_i}, \boldsymbol{\theta}) &= -\left[y_i \ln(c_i) + (1-y_i)\ln(1-c_i)\right] \\
v(\boldsymbol{q_i}, \boldsymbol{\theta}) &= \mathbb{E}\left[\frac{(y-c_i)\,\phi(\boldsymbol{x_i}\boldsymbol{\theta})}{c_i(1-c_i)} \cdot \boldsymbol{x_i}\right] \\
\boldsymbol{B}_0 &= \mathbb{E}\left[(\lambda \cdot \boldsymbol{x})'\,(\lambda \cdot \boldsymbol{x})\right] \\
\boldsymbol{A}_0 &= -\mathbb{E}\left[\left(\boldsymbol{x}\boldsymbol{\theta} + \frac{\phi(\boldsymbol{x}\boldsymbol{\theta})}{c}\right) \cdot \left(\frac{\phi(\boldsymbol{x}\boldsymbol{\theta})}{c}\right) \cdot \boldsymbol{x}'\boldsymbol{x}\right]
\end{aligned}
\tag{7}
$$

where $\phi(\cdot)$ is the probability density function of the normal distribution and $\lambda_i = \frac{(y-c_i)\phi(\boldsymbol{x_i}\boldsymbol{\theta})}{c_i(1-c_i)}$.

Count dependent variable: Poisson, let $c_i = e^{\boldsymbol{x_i}\boldsymbol{\theta}}$:

$$
\begin{aligned}
Q(\boldsymbol{q_i}, \boldsymbol{\theta}) &= -\left[-c_i + y_i \cdot \boldsymbol{x_i}\boldsymbol{\theta} - \ln(y_i!)\right] \\
v(\boldsymbol{q_i}, \boldsymbol{\theta}) &= \mathbb{E}\left[(y_i - c_i) \cdot \boldsymbol{x_i}\right] \\
\boldsymbol{B}_0 &= \mathbb{E}\left[(\lambda \cdot \boldsymbol{x})'(\lambda \cdot \boldsymbol{x})\right] \\
\boldsymbol{A}_0 &= -\mathbb{E}\left[c \cdot \boldsymbol{x}'\boldsymbol{x}\right]
\end{aligned}
\tag{8}
$$

where $\lambda_i = (y_i - c_i)$.

Count dependent variable: negative binomial, let $c_i = e^{\boldsymbol{x_i}\boldsymbol{\theta}}$:

$$
\begin{aligned}
Q(\boldsymbol{q_i}, \boldsymbol{\theta}) &= -\left[y_i \ln\left(\frac{\alpha c_i}{1+\alpha c_i}\right) - \frac{1}{\alpha}\ln(1+\alpha c_i) + \ln\Gamma\left(y_i + \frac{1}{\alpha}\right) - \ln\Gamma(y_i+1) - \ln\Gamma\left(\frac{1}{\alpha}\right)\right] \\
v(\boldsymbol{q_i}, \boldsymbol{\theta}) &= \mathbb{E}\left[\frac{(y_i - c_i)}{1+\alpha c_i} \cdot \boldsymbol{x_i}\right] \\
\boldsymbol{B}_0 &= \mathbb{E}\left[(\lambda \cdot \boldsymbol{x})'(\lambda \cdot \boldsymbol{x})\right] \\
\boldsymbol{A}_0 &= -\mathbb{E}\left[\frac{c(1+\alpha y)}{(1+\alpha c)^2} \cdot \boldsymbol{x}'\boldsymbol{x}\right]
\end{aligned}
\tag{9}
$$

where $\lambda_i = \frac{(y_i-c_i)}{1+\alpha c_i}$, and α reflects the over-dispersion in the Poisson distribution[4] and the $\ln\Gamma$ is the natural logarithm of the gamma function.

Note that the 'filling' of the sandwich estimator for each of these models is the same. This fact allows us to state one method for adjusting standard errors for each of these models in the presence of spatial dependence, and allows for straightforward software implementation.

4. The spatial case

In an important contribution, Jenish and Prucha (2009) showed that the asymptotic distribution of the M-estimator of the conditional mean extends to spatially correlated data. In the remainder of this paper, we therefore apply the method from Conley (1999) to the M-estimators introduced above. We first provide intuition for the M-estimator for the linear model. For this model, the expression for the asymptotic variance-covariance matrix coincides

[4] For more information, see Hilbe (2011), page 2.

with the Ordinary Least Squares expression, and therefore provides a familiar starting point. The standard heteroskedasticity robust White estimator of the variance of $\hat{\theta}$ is:

$$var(\hat{\theta}) = N^{-1}\mathbb{E}[\boldsymbol{x}'\boldsymbol{x}]^{-1}\mathbb{E}[\boldsymbol{x}'uu'\boldsymbol{x}]\mathbb{E}[\boldsymbol{x}'\boldsymbol{x}]^{-1}$$

Where $u_i = \lambda_i = (y_i - c_i)$ is the OLS residual. We apply the method pioneered in Conley (1999) which reweighs the 'filling' of the sandwich estimator using a two-dimensional Bartlett Kernel:

$$\mathbb{E}\left[\sum_{j}^{N} K(i,j)\left(\lambda_j \boldsymbol{x}_j'\right)\left(\lambda'\boldsymbol{x}\right)\right] \tag{10}$$

Where $K(i,j)$ is a two-dimensional kernel function. Let d_{ij} be the distance between observation i and observation j along one of the axes. Recall h_i is i's value for the first coordinate, H. Then, $d_{ij}^H = h_i - h_j$. With V denoting the second coordinate, for two cutoffs L_H and L_V, Conley (1999) proposed using the following kernel:

$$K(i,j) = \begin{cases} \left(1 - \frac{|d_{ij}^H|}{L_H}\right)\left(1 - \frac{|d_{ij}^V|}{L_V}\right) \text{ for } |d_{ij}^H| < L_H,\ |d_{ij}^V| < L_V \\ 0 \text{ otherwise} \end{cases} \tag{11}$$

Intuitively, this kernel sets the covariance between observation i and j to zero if they are beyond a cutoff distance from one another. Within the cutoff the covariance declines linearly with distance.

As the 'filling' of the sandwich estimator of the asymptotic variance of $\hat{\theta}$ is the same for all M-estimators and takes the form above, this method extends trivially to all models introduced above. We reweigh B accordingly for each estimator to arrive at the spatial standard errors for each model.

Note: In principle logit and probit estimators with heteroskedastic standard errors are biased, see (Greene 2003, p. 582), and the methods pioneered in Conley (1999) may therefore not straightforwardly extend to these models (because $\hat{\theta} \xrightarrow{p} \theta_0$ no longer holds). We are not aware of any study regarding the quantitative significance of this bias. (Greene 2003, p. 588) notes the issue but does not provide recommendations. Note that Stata computes heteroskedasticity robust standard errors for both logit and probit, using in essence the same sandwich estimator we derived here.

4.1. Other models

For GMM with linear moment conditions, Conley (1999) provides Stata implementations. For time-series and panel-data models, which may require standard errors corrected for both spatial and time-series dependence, there are packages available from Thiemo Fetzer (`http://www.trfetzer.com/using-r-to-estimate-spatial-hac-errors-per-conley/`) and Hsiang (2010). Colella, Lalive, Sakalli, and Thoenig (2019) provide a Stata implementation, `acreg`, for inference on arbitrary correlation for OLS and 2SLS.

5. Implementation

We provide implementation of the models introduced above in Stata and Python. Standard errors from existing implementations for OLS made available by Conley (1999) coincide exactly with ours. On our websites, we provide the full code as well as do-file and Jupyter notebook that provides examples. In Python, we have three dependencies: **NumPy**, **Pandas** and **StatsModels**. In the following, we demonstrate the implementation, explaining its syntax and use.

Stata *Demonstration*

For the demonstration, we first employ simulated spatial data provided by Conley (1999) as a means to compare our implementation with theirs. The data includes a continuous, binary and count dependent variable needed for the various estimators we offer as well as a matrix of regressors. Coordinates on each observation are also provided to map the data onto a 2-dimensional plane. The implementation can then weigh the standard errors based on these coordinates and exogenously defined cutoffs points. Beyond the cutoff, spatial dependence is assumed to be zero. The syntax is the following:

```
space_reg coordlist cutofflist depvar regressorlist, xreg(#) coord(#) model(str)
```

where `space_reg` is the command, `coordlist` is the variable list of coordinates, `cutofflist` is the variable list of cutoffs, `depvar` is the dependent variable, `regressorlist` is the list of regressors, `xreg(#)` is the number of regressors specified in `regressorlist`, `coord(#)` is the number of coordinates specified in `coordlist` and `model(str)` is the parametric model specification.[5] The user can select from the following models: (1) `ols` (2) `logit` (3) `probit` (4) `poisson` (5) `nb` (6) `areg` and (7) `reghdfe`. The novelty of our implementation is precisely the `model(str)` option. The user can specify the canonical OLS estimator as well as other limited dependent variable models. For fixed-effects models, `areg` can also be specified as well as `reghdfe` for high-dimensional fixed-effects models. To distinguish the matrix of regressors from the fixed-effects, an additional variable list must be included if `areg` or `reghdfe` is specified. I illustrate this in the examples below:

For `logit`:

```
space_reg C1 C2 cutoff1 cutoff2 dep indep1 const, xreg(2) coord(2) model(logit)
```

For `reghdfe`:

```
space_reg C1 C2 cutoff1 cutoff2 dep indep1, ///
xreg(1) coord(2) model(reghdfe, fe1 fe2)
```

Shown below are the data used to test the models. For comparison, the table is followed up by Conley (1999)'s implementation results and our implementation results for the same specification:

[5] There must be as many cutoffs as there are coordinates. The `regressorlist` must include a constant term except for when `areg` or `reghdfe` are specified for model. For more details on the arguments, please consult our `ado` file.

Table 1: Test Dataset

Parameter (1)	Column name (2)	Description (3)
		Coordinates
Coordinate 1 Coordinate 2	`C1 (int)` `C2 (int)`	Each coordinate describes one component of the coordinate vector. In this example, Coordinate 1 and Coordinate 2 define a position on the $\mathbb{R}^2$ plane i.e., $\mathbf{c^i} = (c_1^i, c_2^i)$ for $c_1, c_2 \in \mathbb{R}^2_+$. Each coordinate in this case is an integer from 1 to 10.
		Dependent Variables

Continuous
Binary
Poisson
	`dep (float32)`
	`binarydep (int)`
	`poissondep (int)`
		These are the three dependent variables to be able to test the simulated data on all of our models. `dep` and `binarydep` come from (Conley 1999) while `poissondep` is a generated count variable that takes values $\{1, 2, 3, 4\}$.

		Regressors
Independent Var. Constant	`indep1 (float32)` `const (int)`	The independent variable, `indep1`, is a float and for our example, is an i.i.d. random variable. `const` is a vector of 1s and does not need to be included for `areg` or `reghdfe`.
		Cutoffs
Cutoff 1 Cutoff 2	`cutoff1 (int)` `cutoff2 (int)`	These are the corresponding cutoffs for each coordinate. It is up to the researcher to define the cutoff. For our example, `cutoff1` and `cutoff2` are vectors of a single integer, `4`. Beyond the cutoff, spatial dependence is assumed to be zero.
		Fixed Effects
Fixed Effect 1 Fixed Effect 2	`fe1 (int)` `fe2 (int)`	To use `areg` and `reghdfe`, fixed effects must be specified inside the `model()` option e.g, `model(areg, fe1)`. For our example, $\texttt{fe1}_i = \{1, 2\}$ and $\texttt{fe2}_i = \{1, 2, 3, 4\}$.

Data description: The data contains 100 observations consisting of the 100 possible combinations of integer-valued coordinates that define the space. Data comes from Conley (1999). `poissondep`, `fe1` and `fe2` are the only generated variables.

This is the Stata output produced by Conley (1999)'s spatial standard errors package for an OLS specification:

```
Results for Cross Sectional OLS corrected for Spatial Dependence

number of observations= 100
Dependent Variable= dep

variable          ols estimates          White s.e.         Conley s.e.
---------       ----------------       --------------      --------------
indep1             .56828408             .1976207            .21446303
const              6.4145274             .79007819           1.3310881
```

The Stata output produced by our implementation:

```
Results for Spatial Errors for M-estimators

Number of observations = 100
Dependent variable = dep

Variable     Coef Est.     Standard SE     Standard t-stat       Spatial SE
--------    ----------    ------------    -----------------     -----------
indep1       .56828409       .19762069        2.8756306          .21446303
const       6.4145274       .79007816        8.1188517          1.3310881
```

Our implementation reports the list of regressors with their corresponding coefficient estimate, robust standard error (i.e., uncorrected), t-test statistic and spatial standard error. For the intersection of statistics, our results are identical. This also holds for Conley (1999)'s `xgmlt` implementation and our `model(logit)` implementation. Once these results had been verified, we expanded to include the Probit model, Poisson and Negative Binomial models as well as the fixed-effects models.

Python *Demonstration*

Similar to the **StatsModels** implementations for common estimators, we allow the user to initialize a class and access its properties. The class is initialized with three arguments: (1) the model (2) the coordinates and (3) the cutoffs. For example:

```python
# Preliminaries
import pandas as pd

# Import python code for Conley computations
from space_reg import SpatialStandardErrorsComputer

# Load data, for all models
data = pd.read_stata("spatial_data.dta")

# Initialize class
model = "OLS"
coordinates = ["C1", "C2"]
cutoffs = ["cutoff1", "cutoff2"]
base = SpatialStandardErrorsComputer(data, coordinates, cutoffs)
```

where `model` is a string while `coordinates` and `cutoffs` may be a string or a list of strings. Together, they initialize the `base` class, an object where we can access the properties defined in the `SpatialStandardErrorsComputer` class. To run, we define our dependent and independent variables and call the function `compute_conley_standard_errors_all_models`:

```python
# Call function from our *base* class
y = "dep"
x =[ "indep1", "const"]
ols_se = base.compute_conley_standard_errors_all_models(model, y, x)
ols_se

# Standard Errors
```

```
indep1     0.214463
const      1.331088
Name: Conley s.e., dtype: float64
```

These standard errors are identical to those returned above by Stata. Apart from the standard errors, the user may also access properties such as:

1. `show_sandwich_bread()` returns the inverse hessian
2. `show_sandwich_filling()` returns the variance of the total
3. `show_data()` returns the data
4. `show_coordinates()` returns the coordinates
5. `show_cutoffs()` returns the cutoffs
6. `show_rhs_outcome_rhs_vars()` returns the dependent variable
7. `show_model_estimated()` returns the model in string

6. Conclusion

In this note, we introduced an extension of the computation of spatial standard errors for M-estimators. For a selection of estimators, we provide the relevant expressions for the asymptotic variance covariance matrix. We implement our correction for spatial dependence in Stata and Python.

In terms of future research, it is straightforward to extend the methods in this paper to clustered standard errors, see Bester *et al.* (2011).

References

Amemiya T (1985). *Advanced Econometrics*. Cambridge: Harvard university press.

Bester CA, Conley TG, Hansen CB (2011). "Inference with dependent data using cluster covariance estimators." *Journal of Econometrics*, **165**(2), 137–151.

Chesher A, Lancaster T, Irish M (1985). "On detecting the failure of distributional assumptions." *Annales de l'inséé, No. 59/60, Économétrie non linéaire asymptotique*, pp. 7–45.

Colella F, Lalive R, Sakalli SO, Thoenig M (2019). "Inference with arbitrary clustering." *Working paper*.

Conley TG (1999). "GMM estimation with cross sectional dependence." *Journal of econometrics*, **92**(1), 1–45.

Croux C, Haesbroeck G (2003). "Implementing the Bianco and Yohai estimator for logistic regression." *Computational statistics & data analysis*, **44**(1-2), 273–295.

Greene WH (2003). *Econometric analysis*. Pearson Education.

Hilbe JM (2011). *Negative binomial regression*. Cambridge: Cambridge University Press.

Hsiang SM (2010). "Temperatures and cyclones strongly associated with economic production in the Caribbean and Central America." *Proceedings of the National Academy of sciences*, **107**(35), 15367–15372.

Jenish N, Prucha IR (2009). "Central limit theorems and uniform laws of large numbers for arrays of random fields." *Journal of Econometrics*, **150**(1), 86–98.

Newey K, McFadden D (1994). "Large sample estimation and hypothesis." *HandBook of Econometrics, IV, Edited by R.F. Engle and D.L. McFadden*, pp. 2112–2245.

Stata A, *et al.* (2011). "Stata survey data reference manual release 13."

Wooldridge JM (2002). "Inverse probability weighted M-estimators for sample selection, attrition, and stratification." *Portuguese Economic Journal*, **1**(2), 117–139.

Affiliation:

Luis Calderon
Department of Economics
Universität Bonn
Adenauerallee 24-42, 53113 Bonn, Germany
s6lucald@uni-bonn.de

Leander Heldring
Institute on Behavior and Inequality (briq)
Schaumburg-Lippe-Straße 5-9, 53113 Bonn
leander.heldring@briq-institute.org
leanderheldring.com

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
November 2020, Volume tba, Issue tba *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd