Spatial standard errors for several commonly used M-estimators

Leander Heldring [*]   **Luis Calderon** [†]

January 2026

Abstract

We provide a unified implementation of existing asymptotic theory that computes Conley-style spatial HAC standard errors for a wide range of commonly used (non-linear) estimators. We cover OLS, logit, probit, Poisson, and negative binomial regressions, as well as the fixed-effects estimators *areg* and *reghdfe*—extending commonly used publicly available routines (currently limited to linear models) to nonlinear M-estimators and fixed-effects workflows. We provide Stata and Python software implementing the procedure.

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

JEL classification: C21

[*]Nortwestern University, Kellogg School of Management, 2211 Campus Drive, Evanston, IL 60208, USA. E-mail: `leander.heldring@kellogg.northwestern.edu`. Website: `www.leanderheldring.com`

[†]University of Bonn, Adenauerallee 24-42, 53113 Bonn, Germany. E-mail: `luis.calderon@uni-bonn.de`. Website: `https://luiscald.github.io`. You can find the code for our implementation here.

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. 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 commonly used 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 et al. (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.

---

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

2 M-estimators

Consider a population $\boldsymbol{q} = (y,\ \boldsymbol{x})$, and a sample $\boldsymbol{q}_i$, $i = 1, 2, ..., N$. $\boldsymbol{x}_i$ 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)[2], $\hat{\theta} \xrightarrow{p} \theta_0$ and

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

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 $\mathcal{N}$ is the normal distribution. Formally:

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

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

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

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]The standard regularity conditions for consistency and asymptotic normality of M estimators include: (i) the parameter space Θ is compact and the true value θ_0 lies in the interior; (ii) $Q(q, \theta)$ is measurable in q (no wild jumps) and (almost surely) continuous in θ; (iii) identification: the population objective $\mathbb{E}[Q(q, \theta)]$ is uniquely minimized at θ_0; (iv) uniform convergence: $\sup_{\theta \in \Theta} \left| N^{-1} \sum_{i=1}^{N} Q(q_i, \theta) - \mathbb{E}[Q(q, \theta)] \right| \xrightarrow{p} 0$ (a uniform law of large numbers); and for asymptotic normality, (v) $Q(q, \theta)$ is twice continuously differentiable in a neighborhood of θ_0, with sufficient integrability to interchange differentiation and expectation; (vi) the Hessian matrix $A_0 = \mathbb{E}\left[\nabla_\theta^2 Q(q, \theta_0)\right]$ exists and is nonsingular; and (vii) the score/gradient $v(q, \theta_0) = \nabla_\theta Q(q, \theta_0)$ has finite second moments and satisfies a central limit theorem (under i.i.d. sampling or appropriate weak dependence), so that $\sqrt{N}\, N^{-1} \sum_i v(q_i, \theta_0) \Rightarrow \mathcal{N}(0, B_0)$ with $B_0 = \mathbb{E}[v(q, \theta_0) v(q, \theta_0)']$.

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 et al. (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 $\sigma^2\mathbb{E}[\boldsymbol{x}'\boldsymbol{x}]^{-1}$.

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 $c_i = \Phi\left(\boldsymbol{x_i}\boldsymbol{\theta}\right)$:

---

[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).

$$Q(\boldsymbol{q_i}, \boldsymbol{\theta}) = - \left[y_i \ln(c_i) + (1 - y_i) \ln(1 - c_i) \right]$$

$$v(\boldsymbol{q_i}, \boldsymbol{\theta}) = \mathbb{E} \left[\frac{(y - c_i)\, \phi(\boldsymbol{x_i \theta})}{c_i(1 - c_i)} \cdot \boldsymbol{x_i} \right]$$

$$\boldsymbol{B}_0 = \mathbb{E} \left[(\lambda \cdot \boldsymbol{x})^{'} (\lambda \cdot \boldsymbol{x}) \right] \tag{7}$$

$$\boldsymbol{A}_0 = -\mathbb{E} \left[\left(\boldsymbol{x\theta} + \frac{\phi(\boldsymbol{x\theta})}{c} \right) \cdot \left(\frac{\phi(\boldsymbol{x\theta})}{c} \right) \cdot \boldsymbol{x'x} \right]$$

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

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

$$Q(\boldsymbol{q_i}, \boldsymbol{\theta}) = - \left[-c_i + y_i \cdot \boldsymbol{x_i \theta} - \ln(y_i!) \right]$$

$$v(\boldsymbol{q_i}, \boldsymbol{\theta}) = \mathbb{E} \left[(y_i - c_i) \cdot \boldsymbol{x_i} \right]$$

$$\boldsymbol{B}_0 = \mathbb{E} \left[(\lambda \cdot \boldsymbol{x})^{'} (\lambda \cdot \boldsymbol{x}) \right] \tag{8}$$

$$\boldsymbol{A}_0 = -\mathbb{E} \left[c \cdot \boldsymbol{x'x} \right]$$

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

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

$$Q(\boldsymbol{q_i}, \boldsymbol{\theta}) = - \left[y_i \ln \left(\frac{\alpha c_i}{1 + \alpha c_i} \right) - \frac{1}{\alpha} \ln(1 + \alpha c_i) + \ln \Gamma \left(y_i + \frac{1}{\alpha} \right) - \ln \Gamma(y_i + 1) - \ln \Gamma \left(\frac{1}{\alpha} \right) \right]$$

$$v(\boldsymbol{q_i}, \boldsymbol{\theta}) = \mathbb{E} \left[\frac{(y_i - c_i)}{1 + \alpha c_i} \cdot \boldsymbol{x_i} \right]$$

$$\boldsymbol{B}_0 = \mathbb{E} \left[(\lambda \cdot \boldsymbol{x})^{'} (\lambda \cdot \boldsymbol{x}) \right] \tag{9}$$

$$\boldsymbol{A}_0 = -\mathbb{E} \left[\frac{c(1 + \alpha y)}{(1 + \alpha c)^2} \cdot \boldsymbol{x'x} \right]$$

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.

More precisely, the sandwich "meat" can be written as a sum of outer products of per-observation score contributions. For the models above, each score contribution takes the form $\boldsymbol{s}_i = \lambda_i \boldsymbol{x}_i$ with a model-specific generalized residual λ_i (e.g., probit and negative binomial rescale $y_i - \hat{c}_i$). The spatial HAC correction is

---

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

unified across estimators because it reweights the cross-observation score covariances $\boldsymbol{s}_i \boldsymbol{s}_j'$ using a kernel and cutoff window, regardless of how λ_i is defined.

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 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:

$$var(\hat{\theta}) = [\boldsymbol{x}'\boldsymbol{x}]^{-1} \left[\sum_i^N \boldsymbol{x_i} \boldsymbol{x_i'} u_i^2 \right] [\boldsymbol{x}'\boldsymbol{x}]^{-1} \tag{10}$$

Where u_i is the OLS residual. We apply the method pioneered in Conley (1999) which reweighs the sandwich "meat" (the outer product of score contributions) using a two-dimensional Bartlett Kernel:

$$\sum_i^N \sum_j^N K(i,j) \left(\boldsymbol{x}_i \lambda_i \right) \left(\boldsymbol{x}_j' \lambda_j \right) \tag{11}$$

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:

$$K(i,j) = \begin{cases} \left(1 - \frac{|d_{ij}^H|}{L_H}\right) \left(1 - \frac{|d_{ij}^V|}{L_V}\right) & \text{for } |d_{ij}^H| < L_H,\ |d_{ij}^V| < L_V \\ 0 \text{ otherwise} \end{cases} \tag{12}$$

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.

In addition to the tapered Bartlett kernel, researchers sometimes use a uniform (rectangle) kernel that assigns equal weight to all pairs within the cutoff window (e.g., Bester et al., 2011). In two dimensions, the uniform kernel is:

$$K(i,j) = \begin{cases} 1 \text{ for } |d_{ij}^H| < L_H, \ |d_{ij}^V| < L_V \\ \\ 0 \text{ otherwise} \end{cases} \tag{13}$$

Both kernels set correlations to zero beyond the cutoff; they differ only in how covariances are weighted within the cutoff region.

As the sandwich "meat" can be written as a sum of outer products of score contributions for all M-estimators, this method extends directly to all models introduced above: we reweigh the cross-observation score covariances using $K(i,j)$ to obtain spatial standard errors for each estimator.

Note: The Conley HAC correction is a variance correction for a given M-estimator. It delivers valid inference around the estimator's probability limit. If the stipulated estimator is misspecified, the estimator may converge to a pseudo-true parameter i.e., a best approximation under the stipulated model. In that case, spatial (or non-spatial) sandwich standard errors quantify uncertainty around this approximation, as opposed to the true value. For example, under logit/probit, we estimate $P(y = 1 \mid x)$. If the true DGP implies $P(y = 1 \mid x) = F(x'\theta/\sigma(x))$ and we estimate $P(y = 1 \mid x) = F(x'\theta)$, for F some link function, then these generally converge to a pseudo-true value and valid inference is obtained around this approximation. In practice, when the link function F provides a good approximation to the conditional mean and misspecification is mild, the bias concern may be second-order relative to the benefits of allowing for spatial dependence in the score covariances.

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` and Hsiang (2010). **?** 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.

5.1 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:

```
spacereg depvar regressorlist, coords(coordlist) cutoffs(numlist) model(str) [kernel(str)]
```

where `spacereg` is the command, `depvar` is the dependent variable, `regressorlist` is the list of regressors, `coords(coordlist)` specifies the coordinate variables, `cutoffs(numlist)` specifies the kernel cutoffs as numeric values (either one value replicated across dimensions or one value per coordinate), `model(str)` selects the parametric model, and `kernel(str)` selects the spatial kernel/weighting scheme (`bartlett` by default; `uniform` for the rectangle kernel). In this interface, the number of coordinates and regressors is inferred directly from the variable lists, removing the need to manually supply `coord(#)` and `xreg(#)`.[5] The user can select from: (1) `ols` (2) `logit` (3) `probit` (4) `poisson` (5) `nb` (6) *areg* and (7) *reghdfe*. For fixed-effects models, fixed effects are specified inside `model()`, e.g., `model(areg, fe1)` or `model(reghdfe, fe1 fe2)`. We illustrate this below:

For `logit`:
`spacereg dep indep1 const, coords(C1 C2) cutoffs(4 4) model(logit)`

Uniform (rectangle) kernel: `spacereg dep indep1 const, coords(C1 C2) cutoffs(4 4) model(logit) kernel(uniform)`

For *reghdfe*:

---

[5]There must be as many cutoffs as there are coordinates (or a single cutoff value that is replicated). The `regressorlist` must include a constant term except for when *areg* or *reghdfe* are used. For details, consult the `ado` file.

```
spacereg dep indep1, ///

coords(C1 C2) cutoffs(4 4) 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:

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, $\text{fe1}_i = \{1, 2\}$ and $\text{fe2}_i = \{1, 2, 3, 4\}$.

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 now follows standard Stata estimation conventions: coefficients are reported together with Conley (spatial) standard errors, test statistics, and p-values in the usual `ereturn` table format. This makes it easier to compare results across estimators and to use post-estimation tools that expect `e(b)` and `e(V)`. The command also stores the uncorrected variance matrix for reference.

```
spacereg dep indep1 const_v, coords(C1 C2) cutoffs($cutoff1_val $cutoff2_val) model(ols)


      Source |       SS           df       MS      Number of obs   =       100
-------------+----------------------------------   F(2, 98)        =     47.09
       Model |  5433.15577         2  2716.57788   Prob > F        =    0.0000
    Residual |  5653.83472        98   57.692191   R-squared       =    0.4900
-------------+----------------------------------   Adj R-squared   =    0.4796
       Total |  11086.9905        100  110.869905   Root MSE        =    7.5955



------------------------------------------------------------------------------
             |               Conley
         dep | Coefficient  std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
      indep1 |   .5682841    .214463     2.65   0.009     .1426892     .993879
     const_v |   6.414527   1.331088     4.82   0.000     3.773027    9.056028
------------------------------------------------------------------------------
```

Interfacing with *areg* and *reghdfe* (partialling out fixed effects). For fixed-effects models, the goal is to construct the spatial variance–covariance matrix for the coefficients on the included regressors after removing the variation explained by the fixed effects. Our implementation therefore applies the Frisch–Waugh–Lovell (FWL) logic: it (i) estimates the fixed-effects model using *areg* or *reghdfe* and restricts to the estimation sample, (ii) residualizes the dependent variable with respect to the absorbed fixed effects, (iii) residualizes each regressor of interest with respect to the same fixed effects, and (iv) computes the Conley

"meat" and "bread" using these residualized objects. Intuitively, this ensures that spatial correlation is accounted for in the part of the error term that remains after fixed effects are absorbed. The kernel weights (based on geographic coordinates and cutoffs) are unchanged; only the score contributions $\lambda_i \boldsymbol{x}_i$ are formed using the residualized regressors and residuals.

5.2 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_`

```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:

- `show_sandwich_bread()` returns the inverse hessian

- `show_sandwich_filling()` returns the variance of the total

- `show_data()` returns the data

- `show_coordinates()` returns the coordinates

- `show_cutoffs()` returns the cutoffs

- `show_rhs_outcome_rhs_vars()` returns the dependent variable

- `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, C. A., T. G. Conley, and C. B. Hansen (2011). Inference with dependent data using cluster covariance estimators. *Journal of Econometrics 165*(2), 137–151.

Chesher, A., T. Lancaster, and M. Irish (1985). On detecting the failure of distributional assumptions. *Annales de linsee, No. 59/60, Econometrie non lineaire asymptotique*, 7–45.

Conley, T. G. (1999). Gmm estimation with cross sectional dependence. *Journal of econometrics 92*(1), 1–45.

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

Greene, W. H. (2003). *Econometric analysis.* Pearson Education.

Hilbe, J. M. (2011). *Negative binomial regression.* Cambridge: Cambridge University Press.

Hsiang, S. M. (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. and I. R. Prucha (2009). Central limit theorems and uniform laws of large numbers for arrays of random fields. *Journal of Econometrics 150*(1), 86–98.

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

StataCorp. (2011). Stata survey data reference manual release 13.

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

A   Finite-sample simulation evidence for non-linear models

This appendix provides a finite-sample Monte Carlo exercise for the non-linear models covered by our software (logit, probit, Poisson, negative binomial). The goal is to illustrate how the spatial HAC correction behaves in finite samples when observations are spatially dependent—complementary evidence to Jenish and Prucha (2009), where the focus was on establishing the asymptotic theory behind these estimators.

Design

We simulate $N = 100$ observations on a 10×10 grid with two integer coordinates. For each replication, we draw a regressor $x_i \sim \mathcal{N}(0, 1)$ and set the index to $\eta_i = \beta x_i$ with $\beta = 0.5$. Spatial dependence is introduced using a Gaussian copula: we draw a spatially correlated Gaussian vector z with correlation $\mathrm{corr}(z_i, z_j) = \exp(-d_{ij}/\phi)$ (where d_{ij} is Euclidean distance on the grid) and transform to uniforms $u_i = \Phi(z_i)$. Conditional on x_i, outcomes are then generated using the inverse CDF (or thresholding for binary outcomes) so that the marginal model is correctly specified while dependence arises purely through cross-observation dependence in u_i. We report results for $\phi = 1$ and a Conley cutoff of 3 grid units, using 200 replications.

Results

Table 2 reports the empirical standard deviation of $\hat{\beta}$ across replications, the mean reported standard errors, and the empirical 95% coverage of nominal 95% Wald confidence intervals.

Table 2: Finite-sample summary (200 replications; $\beta = 0.5$, $\phi = 1$, cutoff=3)

Model	Emp. SD ($\hat{\beta}$)	Mean SE (HC1)	Mean SE (Conley–Bartlett)	Mean SE (Conley–Uniform)	Cov.95 (HC1)	Cov.95 (Conley–Bartlett)	Cov.95 (Conley–Uniform)
Logit	0.227	0.228	0.220	0.203	0.950	0.940	0.865
Probit	0.145	0.147	0.140	0.130	0.970	0.960	0.875
Poisson	0.105	0.094	0.092	0.088	0.900	0.900	0.830
NegBin	0.148	0.135	0.128	0.114	0.935	0.880	0.815

Notes: **Emp. SD ($\hat{\beta}$)** is the Monte Carlo standard deviation of the coefficient estimate across replications. **Mean SE (HC1)** is the average heteroskedasticity-robust (non-spatial) HC1 standard error reported by the estimator. **Mean SE (Conley–Bartlett)** and **Mean SE (Conley–Uniform)** are averages of Conley/HAC standard errors computed using Bartlett (tapered) and uniform (rectangle) kernels within the cutoff window, respectively. **Cov.95 (HC1)**, **Cov.95 (Conley–Bartlett)**, and **Cov.95 (Conley–Uniform)** are empirical coverage rates of nominal 95% Wald confidence intervals, i.e., the fraction of replications in which $\beta = 0.5$ lies in $[\hat{\beta} \pm 1.96\, \widehat{se}]$.

Overall, in this design, HC1 and Conley–Bartlett coverages are closer to the nominal 0.95 than Conley–Uniform. This is consistent with the fact that a uniform kernel places relatively more weight on distant within-cutoff pairs, which can increase finite-sample sensitivity to cutoff choice. These results illustrate the diagnostics (empirical SD, mean SE, and coverage) that applied users can use to assess sensitivity to kernel and cutoff choices.