The Stata Journal (*yyyy*) *vv*, Number *ii*, pp. 1–19

netivreg: Estimation of Peer Effects in Endogenous Social Networks

Pablo Estrada
Emory University
Atlanta, USA
pestrad@emory.edu

Juan Estrada
Emory University
Atlanta, USA
jjestra@emory.edu

Kim P. Huynh
Bank of Canada
Ottawa, Canada
kim@huynh.tv

David Jacho-Chávez
Emory University
Atlanta, USA
djachocha@emory.edu

Leonardo Sánchez-Aragón
ESPOL University
Guayaquil, Ecuador
lfsanche@espol.edu.ec

Abstract. The command `netivreg` is presented here that implements the Generalized Three Stage Least Squares (G3SLS) estimator for the endogenous linear-in-means model developed in Estrada et al. (2020, "*On the Identification and Estimation of Endogenous Peer Effects in Multiplex Networks*"). The G3SLS procedure utilizes full observability of a two-layered multiplex network data structure using Stata's 16 new multiframes capabilities and Python integration. Implementations of the command utilizing simulated data as well as three years worth of data on peer-reviewed articles published in top general-interest journals in Economics in Estrada et al. (2020) are also included.

Keywords: st0001, Instrumental variables, `ivregress`, multiplex networks, Python

1 Introduction

In various settings, the decision of agents (people, firms, countries, etc.) to exert effort in some activity does not only depend on their own characteristics, but also on the efforts and characteristics of their peers. This idea is captured in Manski's (1993) so-called *linear-in-means* model where an outcome variable for agent $i \in \{1, \ldots, n\}$, y_i, is determined according to

$$y_i = \alpha + \beta \sum_{j \neq i} w_{i,j} y_j + \sum_{j \neq i} w_{i,j} \mathbf{x}_j^\top \boldsymbol{\delta} + \mathbf{x}_i^\top \boldsymbol{\gamma} + v_i, \tag{1}$$

where $j \in \{1, \ldots, n\}$, $\mathbf{x}_i$ is agent i's $k \times 1$ vector of attributes, $w_{i,j} = 1$ if agent j shares a social connection with i, and 0 otherwise, v_i represents agent i's unobservales, and n is the number of agents in the sample. The structure of the social network is fully characterized by the square $n \times n$ matrix, $\mathbf{W}$, with (i, j) entry given by $w_{i,j}$, i.e., the adjacency matrix. This general econometric network model can be written in matrix form as

$$\mathbf{y} = \boldsymbol{\iota}\alpha + \mathbf{W}\mathbf{y}\beta + \mathbf{W}\mathbf{X}\boldsymbol{\delta} + \mathbf{X}\boldsymbol{\gamma} + \mathbf{v}, \tag{2}$$

where the *peer effect*, captured by β, measures how an agent's outcome may depend on those of her peers, the *contextual effect*, captured by the coefficients $\boldsymbol{\delta}$, where an agent's outcome may depend on the exogenous characteristics of her peers, and the *direct effects*, captured by the coefficients $\boldsymbol{\gamma}$, where an agent's outcome may depent on her's own characteristics.

Under the assumption that $E[\mathbf{v}|\mathbf{X}, \mathbf{W}] = \mathbf{0}$, the `netivreg` command implements Bramoullé et al.'s (2009) Generalized Two Stage Least Squares (G2SLS) estimator of the structural parameters $\boldsymbol{\psi} \equiv [\alpha, \beta, \boldsymbol{\delta}^\top, \boldsymbol{\gamma}^\top]^\top$ in (2) as a special case of Estrada et al.'s (2020) Generalized Three Stage Least Squares (G3SLS) estimator that assumes $E[\mathbf{v}|\mathbf{X}, \mathbf{W}_0] = \mathbf{0}$ instead. The square $n \times n$ matrix, $\mathbf{W}_0$, with (i, j) entry given by $w_{0;i,j}$ represents *another* adjacency matrix of exogenous connections.

The `netivreg`'s internal numerical implementation of the G3SLS estimator is done entirely using the Python language. It makes full use of Stata 16's new integrability with Python, as well as Stata 16's new data frames capabilities to handle the data sets $[\mathbf{y}, \mathbf{X}]$, $\mathbf{W}$, and $\mathbf{W}_0$, see, e.g., Ho et al. (forthcoming). It exploits Python architecture to handle sparse matrices by asking the user to provide the $\mathbf{W}$, and $\mathbf{W}_0$ adjacency matrices as simple (i, j) lists for all pairs for which $w_{i,j} = 1$, and $w_{0;i,j} = 1$. Because of this, the following pre-requisites are needed.

1.1 Pre-Requisites

The `netivreg` requires a working Python 3.7 or higher (Van Rossum and Drake 2009) distribution already installed – the Anaconda's (Anaconda Software Distribution 2020) distribution is strongly recommended. It also needs the NetworkX (Hagberg et al. 2008), Numpy (Oliphant 2006), Pandas (McKinney et al. 2010), Scikit-learn (Pedregosa et al. 2011), and SciPy (Virtanen et al. 2020) Python packages and their dependencies installed. The command also makes use of the Python native `os` and `sys` modules

Any Stata's 16.0 or higher flavor works. However, the user must make sure that the Stata Function Interface (sfi) Python module shipped with the installed Stata version and flavor is working, as it provides a bidirectional connection between the local installation of Stata and Python. Table 1 list the required software and needed versions.

Table 1: Required Software

Language	Version	Python Packages	Version
Stata	16.0 or higher	NetworkX	2.4
Python	3.7 or higher	Numpy	1.19.1
		Pandas	1.1.0
		Scikit-learn	0.23.1
		SciPy	1.5.0

As to maintain backward compatibility as time passes, the user is strongly encouraged to create a virtual environment with the required Python packages versions listed in Table 1 using Anaconda for example.

The rest of the paper is organized as follows: Section 2 introduces the theoretical framework and the identification conditions of model (2) with endogenously formed social interactions. The estimation algorithm implemented by the `netivreg` command is provided in Section 3, while Section 4 provides the command syntax information. Section 5 illustrates how the command can be used with simulated data and an empirical application. Section 6 concludes.

2 Methodology

Firstly, let $\mathbf{S}$ be a $n \times (k+1)$ matrix given by $\mathbf{S} \equiv [\mathbf{y} \quad \mathbf{X}]$, and $\boldsymbol{\theta} \equiv (\beta, \boldsymbol{\delta}^\top)^\top$ be a $(k+1) \times 1$ vector of parameter such that $\beta\mathbf{W}\mathbf{y} + \mathbf{W}\mathbf{X}\boldsymbol{\delta} = \mathbf{W}\mathbf{S}\boldsymbol{\theta}$. Therefore, equation (2) can be written as

$$\mathbf{y} = \alpha\boldsymbol{\iota} + \mathbf{W}\mathbf{S}\boldsymbol{\theta} + \mathbf{X}\boldsymbol{\gamma} + \mathbf{v}. \tag{3}$$

Estrada et al. (2020) also introduces the following system regression condition

$$\mathbf{W}\mathbf{S} = \mathbf{W}_0\mathbf{S}\boldsymbol{\Pi} + \mathbf{U}, \tag{4}$$

where $\boldsymbol{\Pi} = [\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_{k+1}]^\top$ represents a full rank $(k+1) \times (k+1)$ matrix of system coefficients and the $n \times (k+1)$ matrix of system errors $\mathbf{U}$ is such that $E[\mathbf{U}|\mathbf{S}, \mathbf{W}_0] = \mathbf{O}$ (a matrix of zeros). Furthermore, the matrix $E[\mathbf{S}^\top\mathbf{W}_0^\top\mathbf{W}_0\mathbf{S}]$ is positive definite and the first row of $\boldsymbol{\Pi}$, $\boldsymbol{\pi}_1$, is such that $\boldsymbol{\pi}_1^\top\boldsymbol{\theta} < 1/\lambda_{\max}$, where $\lambda_{\max}$ is the largest eigenvalue of $\mathbf{W}_0$.

By substituting (4) in (3) one has

$$\mathbf{y} = \alpha\boldsymbol{\iota} + \mathbf{W}_0\mathbf{S}\boldsymbol{\Pi}\boldsymbol{\theta} + \mathbf{X}\boldsymbol{\gamma} + \mathbf{e}, \tag{5}$$

where $\mathbf{e} \equiv \mathbf{U}\boldsymbol{\theta} + \mathbf{v}$. However, (5) cannot be estimated by simple Ordinary Least Squares (OLS) because $E[\mathbf{S}^\top\mathbf{W}_0\mathbf{e}] \neq \mathbf{0}$, i.e., the simultaneity of $\mathbf{W}_0\mathbf{y}$ still persists, and an Instrumental Variable (IV) procedure is called for. Estrada et al. (2020) shows that a reduced form for $\mathbf{y}$ in (5) is given by

$$\begin{aligned}\mathbf{y} =& [\mathbf{I} - (\boldsymbol{\pi}_1^\top\boldsymbol{\theta})\mathbf{W}_0]^{-1}\alpha\boldsymbol{\iota} + \gamma_1\mathbf{x}_1 + [\gamma_1(\boldsymbol{\pi}_1^\top\boldsymbol{\theta}) + \boldsymbol{\pi}_2^\top\boldsymbol{\theta}]\sum_{r=0}^{\infty}(\boldsymbol{\pi}_1^\top\boldsymbol{\theta})^r\mathbf{W}_0^{r+1}\mathbf{x}_1 + \cdots + \gamma_k\mathbf{x}_k \\ &+ [\gamma_k(\boldsymbol{\pi}_1^\top\boldsymbol{\theta}) + \boldsymbol{\pi}_{k+1}^\top\boldsymbol{\theta}]\sum_{r=0}^{\infty}(\boldsymbol{\pi}_1^\top\boldsymbol{\theta})^r\mathbf{W}_0^{r+1}\mathbf{x}_k + \sum_{r=0}^{\infty}(\boldsymbol{\pi}_1^\top\boldsymbol{\theta})^r\mathbf{W}_0^r\mathbf{e},\end{aligned} \tag{6}$$

and therefore $\mathbf{W}_0^2\mathbf{X}$ is a valid instrument for $\mathbf{W}_0\mathbf{y}$ in (5), i.e., if agents (i,j) have a connection and (j,l) also have a connection, it does not necessarily imply that (i,l) also have a connection. Therefore, following Bramoullé et al. (2009), results in Estrada et al. (2020) have shown that if the matrices $\mathbf{I}$, $\mathbf{W}_0$, and $\mathbf{W}_0^2$ are linearly independent and $\beta(\gamma_k\pi_{1,1}+\pi_{k,1})+\sum_{i=1}^{k}\delta_i(\gamma_i\pi_{1,i+1}+\pi_{k,i+1})\neq 0$ for all k, the social effects $\boldsymbol{\psi}$ are identified.

Notice that the optimal instrument for the regressor $\mathbf{W}_0\mathbf{y}$ in (5) is given by

$$
\begin{aligned}
&E[\mathbf{W}_0\mathbf{y}|\mathbf{X},\mathbf{W}_0](\boldsymbol{\psi},\boldsymbol{\Pi})\\
&\quad=\mathbf{W}_0[\mathbf{I}-(\boldsymbol{\pi}_1^{\top}\boldsymbol{\theta})\mathbf{W}_0]^{-1}\left\{\alpha\boldsymbol{\iota}+[\gamma_1\mathbf{I}+(\boldsymbol{\pi}_2^{\top}\boldsymbol{\theta})\mathbf{W}_0]\mathbf{x}_1+\cdots+[\gamma_k\mathbf{I}+(\boldsymbol{\pi}_{k+1}^{\top}\boldsymbol{\theta})\mathbf{W}_0]\mathbf{x}_k\right\}.
\end{aligned}
\tag{7}
$$

2.1 Correlated Effects: Transformed Model

Now consider the following structure of the structural error in (2)

$$\mathbf{v}=\boldsymbol{\lambda}_0+\boldsymbol{\lambda}+\mathbf{v}^{+}, \tag{8}$$

where $\boldsymbol{\lambda}_0$ represents the correlated effects in the exogenous network $\mathbf{W}_0$, $\boldsymbol{\lambda}$ allows for correlated effects coming from the network $\mathbf{W}$ that are common for the agents who belong to that network, and it is further assumed that $E[\mathbf{v}^{+}|\mathbf{X},\mathbf{W}_0,\boldsymbol{\lambda}_0,\boldsymbol{\lambda}]=\mathbf{0}$. Pre-multiplying both sides of (2) by $(\mathbf{I}-\mathbf{W}_0)$ removes the network specific unobserved heterogeneity for both networks and Estrada et al. (2020) shows that the structural parameters in this transformed model are also identified under similar conditions as before provided matrices $\mathbf{I}$, $\mathbf{W}_0$, $\mathbf{W}_0^2$, and $\mathbf{W}_0^3$ are linearly independent.

If one sets $\mathbf{W}=\mathbf{W}_0$ in (4), then $\boldsymbol{\Pi}=\mathbf{I}$, $\mathbf{U}=\mathbf{O}$ (a matrix of zeroes), and one has Bramoullé et al.'s (2009) methodology.

3 Estimation

Equations (4) and (7) can then be used to estimate the social parameters in (2) via (5) as follows:

Step 1: Regress $\mathbf{W}\mathbf{y}$ on $\mathbf{W}_0\mathbf{y}$ and $\mathbf{W}_0\mathbf{X}$ by Ordinary Least Squares (OLS) and get $\widehat{\mathbf{W}\mathbf{y}}=\mathbf{W}_0\mathbf{y}\widehat{\pi}_{1,1}+\mathbf{W}_0\mathbf{X}\widehat{\boldsymbol{\pi}}_{12}$, and $\widehat{\mathbf{u}}_1=\mathbf{W}\mathbf{y}-\widehat{\mathbf{W}\mathbf{y}}$.
Regress $\mathbf{W}\mathbf{X}$ on $\mathbf{W}_0\mathbf{y}$ and $\mathbf{W}_0\mathbf{X}$ by OLS and get $\widehat{\mathbf{W}\mathbf{X}}=\mathbf{W}_0\mathbf{y}\widehat{\boldsymbol{\pi}}_{21}+\mathbf{W}_0\mathbf{X}\widehat{\boldsymbol{\Pi}}_{22}$, $\widehat{\mathbf{U}}_2=\mathbf{W}\mathbf{X}-\widehat{\mathbf{W}\mathbf{X}}$.

Step 2: Regress $\mathbf{y}$ on $\boldsymbol{\iota}$, $\mathbf{X}$, $\mathbf{W}_0\mathbf{y}$, and $\mathbf{W}_0\mathbf{X}$ by 2SLS using $\boldsymbol{\iota}$, $\mathbf{X}$, $\mathbf{W}_0^2\mathbf{X}$, and $\mathbf{W}_0\mathbf{X}$ as instruments. From $\boldsymbol{\iota}\widehat{\alpha}_{\text{2SLS}}+\mathbf{X}\widehat{\boldsymbol{\gamma}}_{\text{2SLS}}+\mathbf{W}_0\mathbf{y}\widehat{\theta}^{*}_{1;\text{2SLS}}+\mathbf{W}_0\mathbf{X}\widehat{\boldsymbol{\theta}}^{*}_{2;\text{2SLS}}$, get $\widehat{\boldsymbol{\psi}}_{\text{2SLS}}\equiv$

$[\widehat{\alpha}_{\text{2SLS}}, \widehat{\boldsymbol{\gamma}}_{\text{2SLS}}^{\top}, \widehat{\theta}_{1;\text{2SLS}}, \widehat{\boldsymbol{\theta}}_{2;\text{2SLS}}^{\top}]^{\top}$ where $\widehat{\boldsymbol{\theta}}_{\text{2SLS}} \equiv \begin{pmatrix} \widehat{\theta}_{1;\text{2SLS}} \\ \widehat{\boldsymbol{\theta}}_{2;\text{2SLS}} \end{pmatrix} = \begin{pmatrix} \widehat{\pi}_{1,1} & \widehat{\boldsymbol{\pi}}_{12}^{\top} \\ \widehat{\boldsymbol{\pi}}_{21} & \widehat{\boldsymbol{\Pi}}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \widehat{\theta}^{*}_{1;\text{2SLS}} \\ \widehat{\boldsymbol{\theta}}^{*}_{2;\text{2SLS}} \end{pmatrix} = \begin{pmatrix} \widehat{\boldsymbol{\pi}}_{1}^{\top} \\ \widehat{\boldsymbol{\Pi}}_{2}^{\top} \end{pmatrix}^{-1} \begin{pmatrix} \widehat{\theta}^{*}_{1;\text{2SLS}} \\ \widehat{\boldsymbol{\theta}}^{*}_{2;\text{2SLS}} \end{pmatrix}$.

Step 3: Regress $\mathbf{y}$ on $\boldsymbol{\iota}$, $\mathbf{X}$, $\widehat{\mathbf{Wy}}$, and $\widehat{\mathbf{WX}}$ by IV using $\boldsymbol{\iota}$, $\mathbf{x}$, and $[\widehat{\mathbf{Z}} \quad \mathbf{W}_0\mathbf{X}]\widehat{\boldsymbol{\Pi}}$ as instruments, where $\widehat{\mathbf{Z}} = \mathbf{W}_0[\mathbf{I} - (\widehat{\boldsymbol{\pi}}_1^{\top}\widehat{\boldsymbol{\theta}}_{\text{2SLS}})\mathbf{W}_0]^{-1}\{\boldsymbol{\iota}\widehat{\alpha}_{\text{2SLS}} + \mathbf{X}\widehat{\boldsymbol{\gamma}}_{\text{2SLS}} + \mathbf{W}_0\mathbf{X}(\widehat{\boldsymbol{\Pi}}_2^{\top}\widehat{\boldsymbol{\theta}}_{\text{2SLS}})\}$. Call these IV estimates the resulting G3SLS estimator. From $\boldsymbol{\iota}\widehat{\alpha}_{\text{G3SLS}} + \mathbf{X}\widehat{\boldsymbol{\gamma}}_{\text{G3SLS}} + \widehat{\mathbf{Wy}}\widehat{\beta}_{\text{G3SLS}} + \widehat{\mathbf{WX}}\widehat{\boldsymbol{\delta}}_{\text{G3SLS}}$, get

$$\widehat{\boldsymbol{\psi}}_{\text{G3SLS}} \equiv [\widehat{\alpha}_{\text{G3SLS}}, \widehat{\boldsymbol{\gamma}}_{\text{G3SLS}}^{\top}, \widehat{\beta}_{\text{G3SLS}}, \widehat{\boldsymbol{\delta}}_{\text{G3SLS}}^{\top}]^{\top},$$

and $\widehat{\mathbf{v}} \equiv \mathbf{y} - \boldsymbol{\iota}\widehat{\alpha}_{\text{G3SLS}} - \mathbf{X}\widehat{\boldsymbol{\gamma}}_{\text{G3SLS}} - \mathbf{Wy}\widehat{\beta}_{\text{G3SLS}} - \mathbf{WX}\widehat{\boldsymbol{\delta}}_{\text{G3SLS}}$.

Estrada et al. (2020) shows that $\widehat{\boldsymbol{\psi}}_{\text{G3SLS}}$ is a consistent estimator of the structural parameters $\boldsymbol{\psi}$, and that $\sqrt{n}(\widehat{\boldsymbol{\psi}}_{\text{G3SLS}} - \boldsymbol{\psi})$ has an asymptotic multivariate normal distribution with a variance-covariance matrix that can be consistently estimated by

$$\widehat{\mathbf{V}}_{\boldsymbol{\psi}} = (n^{-1}\widehat{\widetilde{\mathbf{Z}}}^{*\top}\widehat{\mathbf{D}})^{-1}(n^{-1}\widehat{\widetilde{\mathbf{Z}}}^{*\top}\widehat{\mathbf{e}}^{*}\widehat{\mathbf{e}}^{*\top}\widehat{\widetilde{\mathbf{Z}}}^{*})(n^{-1}\widehat{\mathbf{D}}^{\top}\widehat{\widetilde{\mathbf{Z}}}^{*})^{-1},$$

where $\widehat{\mathbf{e}}^{*} = \mathbf{M}_{\mathbf{W}_0}\widehat{\mathbf{U}}\widehat{\boldsymbol{\theta}}_{\text{G3SLS}} + \widehat{\mathbf{v}}$ with $\mathbf{M}_{\mathbf{W}_0} \equiv \mathbf{I} - \mathbf{W_0}\mathbf{S}(\mathbf{S}^{\top}\mathbf{W}_0^2\mathbf{S})^{-1}\mathbf{S}^{\top}\mathbf{W_0}$. The residuals $\widehat{\mathbf{U}} \equiv [\widehat{\mathbf{u}}_1 \quad \widehat{\mathbf{U}}_2]$ are obtained from step 1, $\widehat{\boldsymbol{\theta}}_{\text{G3SLS}} \equiv [\widehat{\beta}_{G3SLS}, \widehat{\boldsymbol{\delta}}_{\text{G3SLS}}^{\top}]^{\top}$, and the residuals $\widehat{\mathbf{v}}$ are taken from step 3. Similarly $\widehat{\mathbf{D}} = [\iota, \mathbf{X}, \mathbf{W}_0\mathbf{y}, \mathbf{W}_0\mathbf{X}]\widehat{\boldsymbol{\Gamma}}$, where

$$\widehat{\boldsymbol{\Gamma}} = \begin{bmatrix} \mathbf{I}_{k+1} & \mathbf{O}_{k+1} \\ \mathbf{O}_{k+1} & \widehat{\boldsymbol{\Pi}} \end{bmatrix}, \tag{9}$$

is a $(2k+2) \times (2k+2)$ matrix, $\mathbf{O}_{k+1}$ is a $(k+1) \times (k+1)$ matrix of zeros, and $\mathbf{I}_{k+1}$ represents the identity matrix of order $k+1$. The matrix $\widehat{\widetilde{\mathbf{Z}}}^{*} = \widehat{\mathbf{Z}}^{*}\widehat{\boldsymbol{\Gamma}}$, where $\widehat{\mathbf{Z}}^{*} = [\boldsymbol{\iota}, \mathbf{X}, E_{\mathbf{X},\mathbf{W}_0}[\mathbf{W}_0\mathbf{y}](\widehat{\boldsymbol{\psi}}_{\text{2SLS}}, \widehat{\boldsymbol{\Pi}}), \mathbf{W}_0\mathbf{X}]$.

3.1 Correlated Effects: Transformed Model

In this case, one simply needs to pre-multiply both $\mathbf{y}$ and $\mathbf{X}$ by $(\mathbf{I} - \mathbf{W}_0)$ before implementing the three steps above. A consistent estimator of the asymptotic variance covariance matrix of the estimated structural parameters is then given by $\widehat{\mathbf{V}}_{\boldsymbol{\psi}} = (n^{-1}\widehat{\widetilde{\mathbf{Z}}}^{*\top}(\mathbf{I} - \mathbf{W}_0)^2\widehat{\mathbf{D}})^{-1}(n^{-1}\widehat{\widetilde{\mathbf{Z}}}^{*\top}(\mathbf{I} - \mathbf{W}_0)\widehat{\mathbf{e}}^{*}\widehat{\mathbf{e}}^{*\top}(\mathbf{I} - \mathbf{W}_0)\widehat{\widetilde{\mathbf{Z}}}^{*})(n^{-1}\widehat{\mathbf{D}}^{\top}(\mathbf{I} - \mathbf{W}_0)^2\widehat{\widetilde{\mathbf{Z}}}^{*})^{-1}$.

4 The netivreg Command

This section describes the full syntax of the new `netivreg` command. Stata 16.0 is the earliest version that can run `netivreg`, and a working Python 3.0 or higher installation

with the required packages listed in 1.1 are also needed. The `netivreg` does not use a Stata matrix or `spmat` object to store the adjacency matrices $\mathbf{W}$ and $\mathbf{W}_0$, but uses the Python package Numpy's sparse matrices architecture inside the NetworkX package to handle them in the numerical implementations. Therefore, once the main node-specific data set is loaded into memory, both adjacency matrices must be provided as adjacency *lists* instead stored as Stata frames, see, i.e., Section 5.1.

By default the `netivreg` command expects these adjacency matrices to describe directed graphs. Therefore the user must remember to list *both* entries (i, j) and (j, i) when working with undirected graphs instead.

4.1 Syntax

The syntax of `netivreg` is as follows:

`netivreg` *depvar* [*varlist*] `(W = W0)` [`, tr`ansformed `wx(`*varname*`) id(`*varname*`)` `cl`uster`(`*varname*`)` *display_options* `first second`]

`netivreg` estimates a linear-in-means regression of *depvar* on *varlist* and social interaction network `W`, using the exogenous network `W0` as instrument of the endogenous network `W`. The social networks `W` and `W0` are defined by two adjacency lists stored as Stata frames.

4.2 Options

`tr`ansformed estimates the linear-in-means model with the transformed variables multiplied by $(\mathbf{I} - \mathbf{W}_0)$.

`wx(`*varname*`)` indicates the variables from *varlist* to be included as contextual effects. By default, it includes all the variables.

`id(`*varname*`)` identifies the variable to match covariates with network data. Default varname is id.

`cl`uster`(`*varname*`)` produces standard errors and statistics that are robust to both arbitrary heteroskedasticity and intragroup correlation, where varname identifies the group. Default is non-clustered standard errors.

`first` reports first-stage results of the linear projection of `W` on `W0`.

`second` reports second-stage results of the 2SLS estimation of the linear-in-means model.

4.3 Stored results

`netivreg` stores the following in `e()`:

Scalars			
`e(N)`	number of observations	`e(mss)`	model sum of squares
`e(df_m)`	model degrees of freedom	`e(r2)`	R-squared
`e(df_r)`	residual degrees of freedom	`e(r2_a)`	adjusted R-squared
`e(rank)`	rank of `e(V)`	`e(rmse)`	root mean squared error
`e(chi2)`	chi-squared	`e(N_clust)`	number of clusters
`e(rss)`	residual sum of squares		
Macros			
`e(cmd)`	`netivreg`	`e(exogr)`	exogenous regressor
`e(wx)`	contextual effects	`e(depvar)`	name of dependent variable
`e(clustvar)`	name of cluster variable	`e(properties)`	`b V`
Matrices			
`e(b)`	coefficient vector	`e(V)`	variance–covariance matrix of the estimators
`e(first)`	first-stage regression results	`e(second)`	second-stage regression results

5 Examples

In this section, simulated data and three years worth of data on peer-reviewed articles in Estrada et al. (2020) are used to illustrate the `netivreg` command's estimation capabilities. The command requires two types of data files, one containing the outcome variable and covariates in the traditional format, i.e., a unit record per row (nodes data file), and another where all the pair-wise associations per network among units are recorded per row (edges data files). Note that apart from the nodes data file, at least one edge data file is needed.

5.1 Simulated Data

The following version of the linear-in-means model in (1) is used,

$$\begin{aligned} y_i =& 1 + 0.7 \sum_{j=1}^{n} \overline{w}_{ij} y_j + 0.33 \sum_{j=1}^{n} \overline{w}_{ij} x_{1i} + 0.33 \sum_{j=1}^{n} \overline{w}_{ij} x_{2i} + 0.33 \sum_{j=1}^{n} \overline{w}_{ij} x_{3i} \\ &+ 0.33 x_{1i} + 0.33 x_{2i} + 0.33 x_{3i} + v_i, \end{aligned} \tag{10}$$

where x_{ki} are drawn from an independent and identically (i.i.d.) normal random variable with mean zero and variance 3 for $k = 1, 2, 3$ and independently of each other. The weights $\overline{w}_{ij}$ are row-normalized versions of the adjacency matrix $\mathbf{W} = [w_{ij}]$, i.e., $\overline{w}_{ij} = w_{ij} / \sum_{j=1}^{n} w_{ij}$. The $\mathbf{W}$ adjacency matrix is generated from $\mathbf{W}_0 = [w_{0;ij}]$ which in turn is generated from a Erdös and Rényi's (1959) random graph with density 0.01. Two sets of i.i.d. variables, ε_{1i}^* and ε_{2i}, are drawn from standard normal distributions and

$$w_{ij} = \begin{cases} \mathbb{I}[|\varepsilon_{1i}^* - \varepsilon_{1j}^*| < \widehat{F}_{\varepsilon_1^*}^{-1}(0.95)] \times (1 - w_{0;ij}) + w_{0;ij} & ; \text{ if } \varepsilon_{1i}^* > \Phi^{-1}(0.95), \\ \mathbb{I}[|\varepsilon_{1i}^* - \varepsilon_{1j}^*| < \widehat{F}_{\varepsilon_1^*}^{-1}(0.95)] \times w_{0;ij} & ; \text{ if } \varepsilon_{1i}^* < \Phi^{-1}(0.05), \\ w_{0;ij} & ; \text{ otherwise,} \end{cases}$$

where $\widehat{F}_{\varepsilon_1^*}^{-1}(0.95)$ represents the 95% empirical quantile of the ε_{1i}^* sample, $\Phi^{-1}(\cdot)$ repre-

sents the inverse of the cumulative distribution function of a standard normal random variable, and $I(\cdot)$ is the indicator function that equals one if its argument is true and zero otherwise.

The structural error in (10) is then defined as $v_i = m \times \varepsilon_{1i} + \varepsilon_{2i}$ where

$$\varepsilon_{1i} = \begin{cases} \varepsilon^*_{1i} & ; \text{if } \varepsilon^*_{1i} < \Phi^{-1}(0.05) \text{ or } \varepsilon^*_{1i} > \Phi^{-1}(0.95), \\ 0 & ; \text{otherwise.} \end{cases}$$

The design parameter $m \in \{0, 1\}$ acts as switch to generate either an exogenous $\mathbf{W}$ adjacency matrix ($m = 0$) or an endogenous one ($m = 1$). The sample size is set to $n = 400$.

One first need to import the nodes data file that contains nodes' outcome variable and connections.

```
. use data_sim.dta
. format y_endo y_exo x1 x2 x3 x4 %9.3f
. list in 1/5, table
```

	id	y_exo	y_endo	x1	x2	x3	x4
1.	1	4.072	4.555	-0.523	0.926	2.136	-0.546
2.	2	4.584	4.665	2.611	1.455	-0.926	0.759
3.	3	3.887	3.671	3.125	0.513	-2.718	-2.132
4.	4	3.736	3.962	-2.674	1.504	1.769	0.091
5.	5	6.360	7.002	-0.993	0.345	1.126	1.120

The `data_sim.dta` identifies each node by the `id` variable and two outcomes, i.e., `y_exo` when there is no endogeneity ($m = 0$), and `y_endo` when there is ($m = 1$). The nodes data set also includes the covariate `x4` which was generated from a standard normal distribution indepedent of the outcome variables and covariates `x1`, `x2`, and `x3`.

The edges data sets have the following structure,

```
. use W_sim.dta
. list in 113/117, table
```

	source	target
113.	28	259
114.	28	361
115.	29	67
116.	29	79
117.	29	196

and

```
. use W0_sim.dta
```

```
. list in 113/117, table
```

	source	target
113.	30	167
114.	30	325
115.	31	38
116.	31	83
117.	31	132

where each row records the connection between the node listed as `id` in the `data_sim.dta` as either `source` and `target`. This structure allows for directed or undirected network data, and when invoke the `netivreg` command will check that all the unique identifiers under `source` and `target` are a subset of those listed as `id` in the `data_sim.dta`. An error will be generated otherwise. Nodes that are not listed in either column of these edges data sets are assumed isolated and their corresponding row/column in the adjacency matrices will be made of zeroes.

Exogenous Network

As pointed out by Estrada et al. (2020), if the adjacency matrix $\mathbf{W}$ is exogenous, it can then be used as an instrument for itself and Estrada et al.'s (2020) G3SLS collapses to Bramoullé et al.'s (2009) G2SLS and the `netivreg` command implements it as follows,

```
. use data_sim.dta
. frame create edges
. frame edges: use W_sim.dta
. netivreg y_exo x1 x2 x3 x4 (edges = edges)
Network IV Regression                                 Number of obs =      400
                                                      Wald chi2(10)  =  2021.41
                                                      Prob > chi2    =   0.0000
                                                      R-squared      =   0.8571
                                                      Root MSE       =     .966
```

y_exo	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
W_y						
y_exo	.7187389	.0459317	15.65	0.000	.6284341	.8090436
W_x						
x1	.3628661	.0614882	5.90	0.000	.2419763	.4837559
x2	.3197861	.051635	6.19	0.000	.2182683	.421304
x3	.3968142	.0558112	7.11	0.000	.2870856	.5065427
x4	.0671387	.0886706	0.76	0.449	-.1071935	.2414709
X						
x1	.3749567	.0303577	12.35	0.000	.3152716	.4346419
x2	.3971781	.0302632	13.12	0.000	.3376787	.4566774
x3	.3118756	.0282055	11.06	0.000	.2564218	.3673295
x4	.0614943	.0490757	1.25	0.211	-.0349917	.1579803
_cons	.9875521	.1453766	6.79	0.000	.7017322	1.273372

As expected the G2SLS estimates are numerically close to the true parameters in (10) and the irrelevance of `x4` is picked up by the default heteroskedastic-robust t statistics.

However, the G3SLS reamins a valid consistent estimator as well and it can be computed as follows,

```
. frame create edges0
. frame edges0: use W0_sim.dta
. netivreg y_exo x1 x2 x3 x4 (edges = edges0)
Network IV Regression                                  Number of obs =      400
                                                       Wald chi2(10)  =  1495.20
                                                       Prob > chi2    =   0.0000
                                                       R-squared      =   0.8562
                                                       Root MSE       =    .9712
------------------------------------------------------------------------------
       y_exo |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
W_y          |
       y_exo |   .7066647   .0675054    10.47   0.000     .5739446    .8393847
-------------+----------------------------------------------------------------
W_x          |
          x1 |   .3804765   .0824964     4.61   0.000     .2182832    .5426697
          x2 |    .328785   .0662719     4.96   0.000     .1984902    .4590798
          x3 |   .4072332   .0686797     5.93   0.000     .2722045    .5422619
          x4 |   .0585146   .1069111     0.55   0.584    -.1516796    .2687087
-------------+----------------------------------------------------------------
X            |
          x1 |    .371649   .0390688     9.51   0.000     .2948372    .4484607
          x2 |   .3661416   .0339244    10.79   0.000      .299444    .4328392
          x3 |   .3176699   .0329671     9.64   0.000     .2528543    .3824855
          x4 |   .0702635   .0545833     1.29   0.199    -.0370508    .1775777
       _cons |   1.014104   .2062919     4.92   0.000     .6085203    1.419687
------------------------------------------------------------------------------
```

One observes the estimates are again numerically close to the true values of the parameters and the regressor `x4` is found to be insignificant as well. Since the G2SLS is efficient in this case, the G3SLS estimates are less precise.

Endogenous Network

In the endogenous network formation case, the G2SLS becomes inconsistent but the G3SLS remains consistent. The basic implementation using `netivreg` is

```
. netivreg y_endo x1 x2 x3 x4 (edges = edges0)
Network IV Regression                                  Number of obs =      400
                                                       Wald chi2(10)  =   822.26
                                                       Prob > chi2    =   0.0000
                                                       R-squared      =   0.8176
                                                       Root MSE       =    1.194
------------------------------------------------------------------------------
      y_endo |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
```

W_y						
y_endo	.7059194	.0934719	7.55	0.000	.5221476	.8896911
W_x						
x1	.3464024	.1277675	2.71	0.007	.0952031	.5976017
x2	.3280795	.0870187	3.77	0.000	.1569951	.4991639
x3	.3615469	.0926147	3.90	0.000	.1794604	.5436334
x4	.0500988	.1476019	0.34	0.734	-.2400962	.3402939
X						
x1	.3782985	.0560235	6.75	0.000	.2681526	.4884443
x2	.3287283	.0426851	7.70	0.000	.2448066	.4126499
x3	.3442047	.0483468	7.12	0.000	.2491518	.4392576
x4	.0895948	.0745045	1.20	0.230	-.0568859	.2360756
_cons	1.035534	.3189017	3.25	0.001	.4085523	1.662515

The option `first` prints the point estimates, $\widehat{\mathbf{\Pi}}$, and heteroskedastic-robust standard errors in step 1 in Section 3.

```
. netivreg y_endo x1 x2 x3 x4 (edges = edges0), first
Projection of W on W0
```

	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
W_y_endo						
W0_y_endo	.9927161	.0110994	89.44	0.000	.9708939	1.014538
W0_x1	.0038403	.0419735	0.09	0.927	-.0786823	.086363
W0_x2	-.0030277	.0372046	-0.08	0.935	-.0761744	.070119
W0_x3	.0010825	.0378073	0.03	0.977	-.0732492	.0754142
W0_x4	.002885	.0664465	0.04	0.965	-.1277531	.133523
W_x1						
W0_y_endo	-.0924276	.0041002	-22.54	0.000	-.1004889	-.0843663
W0_x1	.8743972	.0155054	56.39	0.000	.8439126	.9048817
W0_x2	.0074888	.0137437	0.54	0.586	-.0195322	.0345098
W0_x3	.0466023	.0139664	3.34	0.001	.0191436	.0740611
W0_x4	.0143088	.0245459	0.58	0.560	-.03395	.0625676
W_x2						
W0_y_endo	-.0142514	.0034378	-4.15	0.000	-.0210103	-.0074925
W0_x1	-.010898	.0130003	-0.84	0.402	-.0364575	.0146614
W0_x2	.9506398	.0115233	82.50	0.000	.9279843	.9732953
W0_x3	.0197406	.0117099	1.69	0.093	-.0032819	.0427631
W0_x4	.0103946	.0205802	0.51	0.614	-.0300675	.0508567
W_x3						
W0_y_endo	-.0311842	.0037837	-8.24	0.000	-.0386233	-.0237451
W0_x1	.0378469	.0143085	2.65	0.008	.0097153	.0659784
W0_x2	.0052678	.0126829	0.42	0.678	-.0196675	.0302031
W0_x3	.9225525	.0128883	71.58	0.000	.8972132	.9478918
W0_x4	-.0186094	.0226513	-0.82	0.412	-.0631432	.0259244
W_x4						
W0_y_endo	.0140097	.0025567	5.48	0.000	.0089831	.0190363
W0_x1	.0419752	.0096683	4.34	0.000	.0229667	.0609838
W0_x2	.021705	.0085698	2.53	0.012	.0048561	.0385539

```
------------------------------------------------------------------------------
        W0_x3 |  -.0496623   .0087087    -5.70   0.000    -.0667841   -.0325404
        W0_x4 |   .8994643   .0153055    58.77   0.000     .8693726    .9295559
------------------------------------------------------------------------------
```

(*output omitted*)

The option `second` prints the point estimates, $\widehat{\psi}_{\text{2SLS}}$, and heteroskedastic-robust standard errors in step 2 in Section 3.

```
. netivreg y_endo x1 x2 x3 x4 (edges = edges0), second
2SLS Regression                                 Number of obs =      400
                                                Wald chi2(10)  =   839.20
                                                Prob > chi2    =   0.0000
                                                R-squared      =   0.8095
                                                Root MSE       =    1.224
------------------------------------------------------------------------------
      y_endo |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
W_y          |
      y_endo |   .6568384   .1030623     6.37   0.000     .4542113    .8594655
-------------+----------------------------------------------------------------
W_x          |
          x1 |   .3798965   .1172551     3.24   0.001     .1493653    .6104278
          x2 |   .3556006   .0846906     4.20   0.000     .1890934    .5221079
          x3 |   .3883685   .0915501     4.24   0.000      .208375     .568362
          x4 |   .0558494   .1324023     0.42   0.673    -.2044623     .316161
-------------+----------------------------------------------------------------
X            |
          x1 |    .386902   .0569697     6.79   0.000     .2748958    .4989081
          x2 |   .3387842   .0433598     7.81   0.000     .2535361    .4240324
          x3 |     .35358   .0498306     7.10   0.000     .2556099    .4515501
          x4 |   .0933147   .0753217     1.24   0.216    -.0547726    .2414021
       _cons |   1.194332   .3562142     3.35   0.001     .4939918    1.894673
------------------------------------------------------------------------------
```

(*output omitted*)

5.2 Real Data

As an example of using the linear-in-means model with real data, all 729 peer-reviewed research articles published in the *American Economics Review* (`aer`), *Econometrica* (`eca`), the *Journal of Political Economy* (`jpe`), and the *Quarterly Journal of Economics* (`qje`) in 2000, 2001, and 2002 taken from Estrada et al. (2020) is used here. The article specific information is as follows:

```
. use articles.dta
(Data on articles published in the aer, eca, jpe, & qje between 2000-2002)
. describe
Contains data from articles.dta
  obs:           729                          Data on articles published in the aer,
                                                eca, jpe, & qje between 2000-2002
 vars:            12                          12 Sep 2020 14:09
------------------------------------------------------------------------------
```

variable name	storage type	display format	value label	variable label
id	int	%9.0g		Article unique identifier
lcitations	float	%9.0g		Log of total citations 8 years post publication
editor	int	%8.0g		1 if at least one of the article´s authors was an editor of a T4 journal
diff_gender	int	%8.0g		1 if article´s co-authors are of different gender
isolated	int	%8.0g		1 if an article does not share a co-authorship relationship with others
n_pages	byte	%8.0g		Article´s number of pages
n_authors	int	%8.0g		Article´s number of authors
n_references	int	%8.0g		Article´s number of bibliographic references
journal	int	%9.0g	journallab	Journal=aer,eca,jpe,qje
year	int	%9.0g	yearlab	Year=2000,2001,2002
c_alumni	int	%9.0g		Alumni network components unique identifiers
c_coauthor	int	%8.0g		Co-author network components unique identifiers

Sorted by: year journal id

The c_alumni and c_coauthor variables provide information as to which of the 48 *components* in the *Alumni* (herafter alumni) and 575 *components* in the *Co-Author* (hereafter co-authors) networks (see below) an article belong to respectively, i.e., the number of components in a network refers to the number of disconnected parts in it.

```
. gen citations = exp(lcitations)
. tabulate journal year, summarize(citations)
```

Means, Standard Deviations and Frequencies of citations

Journal=aer,eca,jpe,qje	Year=2000,2001,2002 2000	2001	2002	Total
aer	52.417721	54.931821	48.652174	51.934364
	73.653308	90.712233	49.893607	72.800192
	79	88	92	259
eca	49.627451	43.328125	37.177779	42.195122
	52.045351	51.688336	43.565161	48.397466
	51	64	90	205
jpe	34.530612	32.863637	46.666667	38.141844
	26.257619	30.229811	35.717172	31.362075
	49	44	48	141
qje	59.380952	72.285714	102.475	77.653226
	74.70297	47.771446	100.33303	78.338854
	42	42	40	124
Total	49.131221	50.794119	52.448149	50.902607
	61.651186	66.741361	60.112435	62.735929
	221	238	270	729

The `aer` publishes the largest number of research articles, but on average papers published in the `qje` received the most citations 8-year post publication. The total number of articles published in these journals increased from 2000 to 2002.

```
. summarize editor diff_gender isolated n_pages n_authors n_references
```

Variable	Obs	Mean	Std. Dev.	Min	Max
editor	729	.0452675	.208033	0	1
diff_gender	729	.1303155	.3368814	0	1
isolated	729	.5281207	.4995513	0	1
n_pages	729	25.15775	11.53631	3	76
n_authors	729	1.888889	.7486251	1	5
n_references	729	31.40329	17.84755	0	177

Papers in these four journals in this time frame are on average 25 pages long, written by 2 co-authors, and have roughly 31 bibliograhic references. About 13% of them were written by co-authors of different genders, and only 4.5% of them list as a co-author an editor in charge of any of these journals. Finally, arond 53% of the articles do not share a co-authorship relationship with other articles (see below).

Networks

As explained in Estrada et al. (2020) two types of connections among these 729 research articles can be constructed. Since the names of each article's authors are known, a co-authors relationship can be formed among them, i.e.,

```
. frame create edges
. frame edges: use edges.dta
(Co-authorship network among articles published in the aer, eca, jpe, & qje betwe)
. frame edges: list in 1/5, table
```

	source	target
1.	4	472
2.	5	221
3.	5	463
4.	5	478
5.	5	665

```
. frame create edges0
. frame edges0: use edges0.dta
(Alumni network among articles published in the aer, eca, jpe, & qje between 2000)
. frame edges0: list in 1/5, table
```

	source	target
1.	2	482
2.	2	534
3.	4	129
4.	4	136

```
5. |      4      407 |
```

In this case, article with id number 4 is connected to article with id number 472 in the `edges.dta` frame because at least one of these articles' authors is the same. Similarly article with id number 5 is connected to articles 221, 463, 478, and 665 because at least one of the authors in these 5 articles is the same person. We refer to these connections as the co-authors network among articles.

Similarly, since authors' information was web-scrapped or text mined from online profiles, Estrada et al. (2020) also provide alumni connections among articles, i.e., article with id number 4 is connected with articles 129, 136, and 407 in the `edges0.dta` frame because at least one of the these articles' authors overlapped at least 3 years in graduate school at the same institution.

Since network data requires its own type of network descriptive statistics, Table 2 displays them.

Table 2: Network descriptive statistics

Statistics	Co-authors ($\mathbf{W}$)	Alumni ($\mathbf{W_0}$)
Number of nodes	729	729
Number of edges	674	8,838
Average degree	1.85	24.25
Density	0.00	0.03
Average clustering	0.71	0.55
Isolated nodes	385	41

Note: The degree of a node in a network is the number of connections (edges) it has to other nodes. The density of a network is the portion of the potential connections in a network that are actual connections. The average clustering of a network is the average of the local clustering coefficients of all the nodes, where the local clustering coefficient of a node is the proportion of edges between the nodes within its neighborhood divided by the number of edges that could possibly exist between them.

The typical article has two connections (edges) in the co-authors network but about 24 in the alumni network, i.e., there are considerably more connections in the latter (number of edges). Both networks have very low density and there are about 10 times more articles that do not have a co-authors connection than those that do not share an alumni connection.

Estimation

The empirical model of interest in Estrada et al. (2020) is

$$y_{i,r,t} = \alpha + \beta \sum_{j \neq i} w_{i,j} y_{j,r,t} + \sum_{j \neq i} w_{i,j} \mathbf{x}_{j,r,t}^{\top} \boldsymbol{\delta} + \mathbf{x}_{i,r,t}^{\top} \boldsymbol{\gamma} + \lambda_r + \lambda_t + \lambda_0 + v_{i,r,t}, \tag{11}$$

where $y_{i,r,t}$ represents the natural logarithm of article i's citations 8-year post publication (`lcitations`) in journal r in year t, $\mathbf{x}_{j,r,t}$ includes `diff_gender` and `editor` of article j in journal r in year t, and $\mathbf{x}_{i,r,t}$ include the same characteristics for article i plus its number of pages (`n_pages`), authors (`n_authors`), bibliographic references (`n_references`), and whether or not it shares a co-authors' relationship with other articles `isolated`. Fixed effects in terms of journal (λ_r), year (λ_t), and alumni component (λ_0) are also included. It is assumed that the co-authors network ($\mathbf{W}$) is endogenous, and that the alumni network ($\mathbf{W_0}$) is pre-determined and therefore assumed exogenous. The estimation of model (11) with clustered standard errors at the co-authors components yields

```
. tabulate journal, g(journal)
```

(*output omitted*)

```
. tabulate c_alumni, g(alumni)
```

(*output omitted*)

```
. tabulate year, g(year)
```

(*output omitted*)

```
. netivreg lcitations diff_gender editor n_pages n_authors n_references isolated
> journal1-journal3 year2-year3 alumni2-alumni48 (edges = edges0),
> wx(diff_gender editor) cluster(c_coauthor)
Network IV Regression                                 Number of obs =      729
Number of clusters (c_coauthor) =    575              Wald chi2(62)  =  1.2e+17
                                                      Prob > chi2    =   0.0000
                                                      R-squared      =   0.1723
                                                      Root MSE       =    1.339
                          (Std. Err. adjusted for 575 clusters in c_coauthor)
------------------------------------------------------------------------------
  lcitations |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
W_y          |
  lcitations |   .5200772   .3616317     1.44   0.151    -.1899963    1.230151
-------------+----------------------------------------------------------------
W_x          |
 diff_gender |  -2.095893   2.829443    -0.74   0.459     -7.65158    3.459794
      editor |   1.789686   5.362754     0.33   0.739    -8.740227     12.3196
-------------+----------------------------------------------------------------
X            |
 diff_gender |    .218709   .1305651     1.68   0.094    -.0376592    .4750771
------------------------------------------------------------------------------
```

```
------------------------------------------------------------------------------
      editor |   .1733642   .1157379     1.50   0.135    -.0538902    .4006187
     n_pages |   .0288947   .0044187     6.54   0.000     .0202184    .0375709
   n_authors |   .0719403   .0597035     1.20   0.229    -.0452891    .1891696
n_references |   .0119892   .0025599     4.68   0.000     .0069628    .0170156
    isolated |  -.2230689   .0897056    -2.49   0.013    -.3992083   -.0469295

(output omitted)

       _cons |   2.771797   .2151335    12.88   0.000     2.349377    3.194218
------------------------------------------------------------------------------
```

These results match those in Estrada et al. (2020). The hypothesis of a null peer effect is rejected at the 10% level of significance against a positive peer-effect hypothesis, i.e., a 10% increase in the number of citations of connected articles would increase a paper's citations by 5.2%. Holding everything else constant, articles with larger number pages and of bibliographic references get cited more and so do articles written by authors of different genders at the 10% level of significance. Finally, articles that do not share a co-authorship connections with others get cited roughly 22% less than those connected.

6 Conclusion

In this article, the new `netivreg` command is presented which can be used to fit a linear-in-means model with network data. Both exogenous or endogenous network formation are supported. `netivreg` main estimation routine is fully written in Python using Stata 16 or higher integration with Python. The command utilizes Stata 16 or higher new multi-frame capabilities to handle the required network data structure in the form of adjacency lists. These in turn are converted to sparse matrices within Python for the numerical implementation. Simulated data is then used to illustrate `netivreg` basic capabilities, while an empirical application based on peer-reviewed articles published in four top general interest journals in economics uncovers positive peer effects in terms of citations 8 years post publication.

7 Acknwoledgements

We thank David Drukker for his constant encouragement to make our estimation routines available in Stata, as well as for answering various clarifying questions on Stata 16's new Python integration. We thank Venkataraman Balasubramanian for providing invaluable expertise optimizing the Python code that the `netivreg` command uses and for helping with the extensive testing performed using the Bank of Canada's EDITH2 High-Performance Cluster. Pablo Estrada and Sánchez-Aragón acknowledge financial support from ESPOL University. The views expressed in this article are those of the authors. No responsibility for them should be attributed to the Bank of Canada. All remaining errors are the responsibility of the authors.

8 References

Anaconda Software Distribution. 2020. Anaconda Documentation. https://docs.anaconda.com/.

Bramoullé, Y., H. Djebbari, and B. Fortin. 2009. Identification of Peer Effects through Social Networks. *Journal of Econometrics* 150(1): 41–55.

Erdös, P., and A. Rényi. 1959. On Random Graphs. *Publicationes Mathematicae (Debrecen)* 6: 290–297.

Estrada, J., K. P. Huynh, D. T. Jacho-Chávez, and L. Sánchez-Aragón. 2020. On the Identification and Estimation of Endogenous Peer Effects in Multiplex Networks. Unpublished manuscript. http://kphuynh.pages.iu.edu/rsch/multiplex_ecmt.html.

Hagberg, A., P. Swart, and D. S Chult. 2008. Exploring Network Structure, Dynamics, and Function using NetworkX. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Ho, A. T. Y., K. P. Huynh, D. T. Jacho-Chavez, and D. Rojas. forthcoming. Data Science in Stata 16: Frames, Lasso, and Python Integration. *Journal of Statistical Software* .

Manski, C. F. 1993. Identification of Endogenous Social Effects: The Reflection Problem. *Review of Economic Studies* 60(3): 531–542.

McKinney, W., et al.. 2010. Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*. Vol. 445, 51–56. Austin, TX.

Oliphant, T. E. 2006. *A Guide to NumPy*. Vol. 1. Trelgol Publishing USA.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al.. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12(Oct): 2825–2830.

Van Rossum, G., and F. L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* .

About the authors

Pablo Estrada is a Ph.D. candidate in Economics at Emory University, Atlanta, United States of America.

Juan Estrada is a Ph.D. candidate in Economics at Emory University, Atlanta, United States of America.

Kim P. Huynh is a Director in the Currency Department at the Bank of Canada, Ottawa, Canada.

David Jacho-Chávez is a Professor of Economics at Emory University, Atlanta, United States of America.

Leonardo Sánchez-Aragón is a Professor of Economics at ESPOL University, Guayaquil, Ecuador.