



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Spatial panel data models using Stata

Citation for published version:

Belotti, F, Hughes, G & Piano Mortari, A 2017, 'Spatial panel data models using Stata' The Stata Journal, vol. 17, no. 1, pp. 139-180.

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Peer reviewed version

Published In:

The Stata Journal

Publisher Rights Statement:

This is the Accepted Manuscript of the article Spatial panel-data models using Stata published at <http://www.stata-journal.com/article.html?article=st0470>
© Copyright 2001–2017 StataCorp LLC

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



Spatial panel data models using Stata

Federico Belotti

Centre for Economic and International Studies
University of Rome Tor Vergata

Gordon Hughes

University of Edinburgh

Andrea Piano Mortari

Centre for Economic and International Studies
University of Rome Tor Vergata

Abstract. `xsmle` is a new command for spatial analysis using Stata. We consider the quasi-maximum likelihood estimation of a wide set of both fixed- and random-effects spatial models for balanced panel data. Of special note is that `xsmle` allows to handle unbalanced panels thanks to its full compatibility with the `mi` suite of commands, to use spatial weight matrices in the form of both Stata matrices and `spmat` objects, to compute direct, indirect and total marginal effects and related standard errors for linear (in variables) specifications, and to exploit a wide range of postestimation features, extending to the panel data case the predictors proposed by Kelejian and Prucha (2007). Moreover, it also allows the use of `margins` to compute total marginal effects in presence of nonlinear specifications obtained using `factor variables`. This paper describes the command and all its functionalities using both simulated and real data.

Keywords: `st000`, spatial analysis, panel data, maximum likelihood estimation.

1 Introduction

It is widely recognized that sample data collected from geographically close entities are not independent, but *spatially* correlated, which means that observations of closer units tend to be more similar than further ones (Tobler 1970).¹ Spatial clustering or geographic-based correlation is often observed for economic and socio-demographic variables such as unemployment, crime rates, house prices, per-capita health expenditures and the alike (Sollé Ollé 2003; Moscone and Knapp 2005; Revelli 2005; Sollé Ollé 2005; Kostov 2009; Elhorst and Freret 2009; Elhorst et al. 2010; Moscone et al. 2012). Theoretical models usually recognize the existence of spatial spillovers which decline as distance between units increases; empirically, spatial panel data models have become a popular tool for measuring such spillovers.

To the best of our knowledge, while both R and Matlab offer large suite of functions for the estimation of spatial panel data models (Millo and Piras 2012; LeSage and Pace 2009), with the notable exemption represented by the accompanying code of Kapoor et al. (2007), Stata's capabilities include a wide set of commands which are designed to deal only with cross-sectional data (Drukker et al. 2013a,c,b). We developed the `xsmle`

1. It is worth emphasizing that non-spatial structured dependence may also be observed. In these cases, measures of geographical proximity are replaced by measures of similarity allowing to investigate peer effects through social or industrial networks (LeSage and Pace 2009; Bramoullé et al. 2009).

command to fill this gap, providing the possibility to estimate a wide range of spatial panel data models using Stata. In particular, `xsmle` allows to estimate both fixed- and random-effects spatial autoregressive, spatial Durbin and spatial error models, fixed-effects spatial autocorrelation models and generalized spatial random-effects models. For spatial autoregressive and spatial Durbin models with fixed-effects `xsmle` also allows a dynamic specification by implementing the bias corrected maximum likelihood approach described in Yu et al. (2008). Among other interesting features described in the paper, of special note is that `xsmle` allows to *i*) use spatial weight matrices created through the Drukker et al. (2013a)'s `spmat` command; *ii*) compute direct, indirect and total marginal effects; *iii*) compute both clustered and Driscoll-Kraay standard errors; *iv*) test whether a fixed- or random-effects model is appropriate using a robust Hausman test; *v*) exploit a wide range of predictors, extending to the panel data case the estimators proposed by Kelejian and Prucha (2007).

The rest of this paper is organized as follows. In Section 2 we present a brief review of spatial panel data models estimable using the new command. Section 3 documents `xsmle` syntax and its main options while Section 4 illustrates its main features using simulated and real datasets. The last section offers some conclusions.

2 Spatial panel data models

In this section we give a brief overview of spatial panel data models focusing on those that can be estimated using `xsmle`. It is worth emphasizing that `xsmle` is primarily designed to deal with balanced panel data in which n units are observed for exactly T periods. We will turn to unbalanced panels in Section 4.1, where we show how to handle this case by exploiting the official `mi` suite of commands.

In what follows we denote, for each period $t = 1, \dots, T$, by \mathbf{y}_t the $n \times 1$ column vector of the dependent variable and by \mathbf{X}_t the $n \times k$ matrix of regressors. For each cross-section, \mathbf{W} is the $n \times n$ matrix describing the spatial arrangement of the n units and each entry $w_{ij} \in \mathbf{W}$ represents the spatial weight associated to units i and j .² In order to exclude self-neighbors, the diagonal elements w_{ii} are conventionally set equal to zero. It is worth emphasizing that `xsmle` allows the use of two different formats for the weight matrix, that is \mathbf{W} can be a Stata matrix or a `spmat` object. This feature allows the user to leverage the capabilities of other Stata commands which allow the creation and management of weight matrices like `spmat`, Pisati (2010)'s `spatwmat` or Jeanty (2010)'s `spwmatrix`. Furthermore, the command automatically takes care of the

2. Two sources of locational information are generally exploited. First, the location in Cartesian space (e.g latitude and longitude) is used to compute distances among units. Second, the knowledge of the size and shape of observational units allows the definition of contiguity measures, e.g., one can determine which units are neighbors in the sense that they share common borders. Thus, the former source points towards the construction of spatial distance matrices while the latter is used to build spatial contiguity matrices. It is worth noting that the aforementioned sources of locational information are not necessarily different. For instance, a spatial contiguity matrix can be constructed by defining units as contiguous when they lie within a certain distance; on the other hand by computing the coordinates of the centroid of each observational unit, approximated spatial distance matrices can be obtained using the distances between centroids. More details are available in LeSage and Pace (2009).

longitudinal nature of the data, hence users only need to provide the cross-sectional $n \times n$ weight matrix in order to estimate a specific model.

`xsmle` allows to fit the following models:

SAR) Spatial autoregressive model. The basic equation for the SAR model is

$$\mathbf{y}_t = \rho \mathbf{W} \mathbf{y}_t + \mathbf{X}_t \boldsymbol{\beta} + \boldsymbol{\mu} + \boldsymbol{\epsilon}_t \quad t = 1 \dots, T, \quad (1)$$

It is assumed that $\boldsymbol{\mu} \sim N(0, \sigma_\mu^2)$ in the random-effects case, while the $\boldsymbol{\mu}$ is a vector of parameters to be estimated in the fixed-effects variant. The standard assumptions that $\epsilon_{it} \sim N(0, \sigma_\epsilon^2)$ and $E(\epsilon_{it}\epsilon_{js}) = 0$ for $i \neq j$ and/or $t \neq s$ apply in this case;

SDM) Spatial Durbin model. This model is a generalization of the SAR model which also includes spatially weighted independent variables as explanatory variables

$$\mathbf{y}_t = \rho \mathbf{W} \mathbf{y}_t + \mathbf{X}_t \boldsymbol{\beta} + \mathbf{W} \mathbf{Z}_t \boldsymbol{\theta} + \boldsymbol{\mu} + \boldsymbol{\epsilon}_t. \quad (2)$$

The model can be generalized by using different spatial weights for the spatially lagged dependent variable ($\mathbf{W} \mathbf{y}$) and the spatially weighted regressors ($\mathbf{W} \mathbf{Z}$) or by using $\mathbf{Z}_t \neq \mathbf{X}_t$;

SAC) Spatial Autocorrelation Model. This model (alternatively referred to as the spatial autoregressive with spatially autocorrelated errors, SARAR) combines the SAR with a spatial autoregressive error

$$\mathbf{y}_t = \rho \mathbf{W} \mathbf{y}_t + \mathbf{X}_t \boldsymbol{\beta} + \boldsymbol{\mu} + \boldsymbol{\nu}_t, \quad (3)$$

$$\boldsymbol{\nu}_t = \lambda \mathbf{M} \boldsymbol{\nu}_t + \boldsymbol{\epsilon}_t, \quad (4)$$

where \mathbf{M} is a matrix of spatial weights which may or may not be equal to \mathbf{W} . The literature focuses on the fixed-effects variant of this specification as the random-effects variant can be written as a special case of the SAR specification;

SEM) Spatial error model. The spatial error model focuses on spatial auto-correlation in the error term as in

$$\mathbf{y}_t = \mathbf{X}_t \boldsymbol{\beta} + \boldsymbol{\mu} + \boldsymbol{\nu}_t, \quad (5)$$

$$\boldsymbol{\nu}_t = \lambda \mathbf{M} \boldsymbol{\nu}_t + \boldsymbol{\epsilon}_t. \quad (6)$$

It is clear that this is a special case of the SAC model but it can also be shown that it is a special case of the SDM.

GSPRE) Generalized spatial random-effects model. This model can be represented as

$$\mathbf{y}_t = \mathbf{X}_t \boldsymbol{\beta} + \boldsymbol{\mu} + \boldsymbol{\nu}_t, \quad (7)$$

$$\boldsymbol{\nu}_t = \lambda \mathbf{M} \boldsymbol{\nu}_t + \boldsymbol{\epsilon}_t, \quad (8)$$

$$\boldsymbol{\mu} = \phi \mathbf{W} \boldsymbol{\mu} + \boldsymbol{\eta}. \quad (9)$$

This is a generalization of the spatial error model in which the panel effects, represented by the vector $\boldsymbol{\mu}$, are spatially correlated. The vectors $\boldsymbol{\mu}$ and $\boldsymbol{\epsilon}_t$ are assumed to be independently normally-distributed errors, so the model is necessarily a random-effects specification with $\boldsymbol{\mu} = (\mathbf{I} - \phi\mathbf{W})^{-1}\boldsymbol{\eta}$ and $\boldsymbol{\nu}_t = (\mathbf{I} - \lambda\mathbf{W})^{-1}\boldsymbol{\epsilon}_t$. There are various special cases of the general specification with (a) $\lambda \neq \phi \neq 0$, (b) $\lambda = 0$, (c) $\phi = 0$, (d) $\lambda = \phi$.

In addition to the distinction between the fixed- and random-effects, there is a separate distinction between static and dynamic specifications. The aforementioned models are all static in that they involve contemporaneous values of the dependent and independent variables. `xsmle` also allows the estimation of SAR and SDM models like

$$\mathbf{y}_t = \tau\mathbf{y}_{t-1} + \psi\mathbf{W}\mathbf{y}_{t-1} + \rho\mathbf{W}\mathbf{y}_t + \mathbf{X}_t\boldsymbol{\beta} + \boldsymbol{\mu} + \boldsymbol{\epsilon}_t, \quad (10)$$

where the lagged (in time) dependent variable and/or the lagged (in both time and space) dependent variable can be included in the specification.

2.1 Estimation

Various methods of estimating spatial panel models have been proposed. Broadly, they fall into two categories: *i*) Generalized Method of Moments (GMM) and *ii*) Quasi-Maximum Likelihood (QML) estimators. All the models that can be estimated using `xsmle` fall into the second category. A synopsis guide with all estimable models and their features is reported in Table 1.³ The gain from programming gradients is large, so `v1` evaluators are used for all but one of the specifications. The exception is the random-effects SEM, whose likelihood function involves a transformation using the Cholesky factors of a rather complicated matrix containing the parameters to be estimated, so that the matrix differentiation is extremely messy.

For dynamic models, i.e. those including a time-lagged dependent variable, a time and space-lagged dependent variable or both of them, `xsmle` implements only the fixed-effects variant of the SAR and SDM models using the bias corrected QML approach described by Yu et al. (2008), which is consistent when both $n \rightarrow \infty$ and $T \rightarrow \infty$. The command starts by constructing ML estimates treating the lagged dependent variables as exogenous regressors. Then bias corrections are computed for each of the coefficients and used to adjust the initial ML estimates.

For each model, the *default* asymptotic variance-covariance matrix of the coefficients is obtained from the observed information matrix.⁴ Angrist and Pischke (2009) emphasize the potential dangers of this approach for datasets for which there may be unknown serial correlation in the errors within each panel unit. To our knowledge, there are no

3. Elhorst (2009) suggests that the computation time required to carry out full maximum likelihood estimation can be reduced by transforming variables in a way that permits the likelihood function to be concentrated, so that the estimation can be carried out in two steps. In translating his routines to Mata, we found that the use of a concentrated likelihood tended to increase both the number of iterations and the time required to estimate the models.

4. A variant obtained from the outer product of the gradients is also available specifying `vce(opg)`.

established methods of computing robust standard errors for spatial panel data models. Mimicking the derivation of robust standard errors for non-spatial panel models, `xsmle` implements two different approaches: i) one-way clustered standard errors, ii) Driscoll and Kraay (1998) standard errors. As in other panel data official Stata commands, specifying `vce(robust)` is equivalent here to specify `vce(cluster panelvar)`, where `panelvar` is the variable that identifies the panels. As for the Driscoll-Kraay standard errors, the `xsmle` implementation is based on Hoechle (2007)'s `xtscc` command. The Driscoll-Kraay approach provides a specific variant of the Newey-West robust covariance estimator computed using the Bartlett kernel and a time series of scores' cross-sectional averages.⁵

In our test runs the differences between the asymptotic and robust standard errors are usually small, but we have not focused on cases with small values of n and T . In principle, it would be useful to include also a bootstrap estimator for the variance-covariance matrix. Unfortunately, there is a major barrier to the application of standard bootstrap methods in this case. The crucial assumption for resampling is that the errors for the observations or units from which each sample is drawn should be independent. For panel or clustered data this means that the resampling is based on panel units or clusters. For spatial panels our base model assumes that the observations for different panel units are correlated over space for any given period t . It follows that resampling based on panel units cannot be reconciled with the hypothesis of spatial interactions in the relationship(s) of interest. As an alternative, we could use time periods as the resampling unit but this will only be valid if there is no serial correlation within panels. Further, for many applications of spatial panel estimation the value of T is considerably smaller than n , so that the large sample assumptions of bootstrap statistics do not apply. Statisticians have developed bootstrap methods for spatial data but at the cost of imposing substantial restrictions on the extent of spatial interactions that can be examined. The methods have tended to focus on regular lattices but they can be applied to spatial data for fairly small economic units such as counties and labour market areas.

Direct, indirect, total marginal effects

Since spatial regression models exploit the complicated dependence structure between units, the effect of an explanatory variable's change for a specific unit will affect the unit itself and, potentially, all other units indirectly. This implies the existence of direct, indirect and total marginal effects. With the exception of the SEM and the GSPRE models, and only if the `effects` option is specified, these effects are computed using the formulas reported in Table 2. The command automatically distinguish also between short- and long-run marginal effects when a dynamic spatial model is estimated.

It is important to highlight that the analytical results reported in Table 2 are only valid for linear (in variables) specifications. Thus, *by default*, a "factor variables" specification will block the computation of these effects.⁶ Nonetheless, in these cases `xsmle`

5. The bandwidth for the kernel is specified with a default value of `floor [4(T/100)2/5]` if no value is specified.

6. We thank an anonymous referee for bringing this point to our attention. It is worth noting that,

allows for the use of `margins` to at least compute total marginal effects. As described in section 4.1 `xsmle` also computes marginal effect's standard errors using Monte Carlo simulation (the default) or Delta Method.

Robust Hausman test

A classical question in panel data empirical analyses refers to the choice between fixed- and random-effects variants (when both can be estimated). An answer to this question can be given using the Hausman (1978) statistic

$$\hat{\xi} = \hat{\delta}' \widehat{\mathbf{V}}_0^{-1} \hat{\delta}, \quad (11)$$

where $\hat{\delta} = (\hat{\beta}_{FE} - \hat{\beta}_{RE})$ is the difference between the fixed- and random-effects estimates and $\widehat{\mathbf{V}}_0$ is an estimate of the variance-covariance matrix of $\hat{\delta}$. The asymptotic distribution of (11) under the null hypothesis of no systematic difference between the two set of estimates is χ^2 with c degrees of freedom, with c being usually the size of the estimated parameter vector. This test can be easily implemented in Stata using the official `hausman` command. However, one of the common issues with spatial panel data models is that the Hausman specification test often fails to meet its asymptotic assumptions, especially in small samples. This is due to the fact that, under the alternative hypothesis, $\widehat{\mathbf{V}}_0 = \widehat{\mathbf{V}}_{FE} - \widehat{\mathbf{V}}_{RE}$ is not ensured to be positive definite. `xsmle` allows to overcome this issue since it directly takes into account the $\widehat{\mathbf{Cov}}(\hat{\beta}_{FE}, \hat{\beta}_{RE})$, that is $\widehat{\mathbf{V}}_0 = \widehat{\mathbf{V}}_{FE} + \widehat{\mathbf{V}}_{RE} - 2\widehat{\mathbf{Cov}}(\hat{\beta}_{FE}, \hat{\beta}_{RE})$. In particular, `xsmle` estimates $\widehat{\mathbf{V}}_0$ through $D\widehat{\mathbf{W}}_0D'$, where $D = (\mathbf{I}_c, -\mathbf{I}_c)$ and \mathbf{I}_c denotes the identity matrix of size c . The joint variance-covariance matrix $\widehat{\mathbf{W}}_0$ is consistently estimated using the following sandwich formula

$$\widehat{\mathbf{W}}_0 = \begin{pmatrix} \mathbf{H}_{FE} & \mathbf{O} \\ \mathbf{O} & \mathbf{H}_{RE} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{S}_{FE,FE} & \mathbf{S}_{FE,RE} \\ \mathbf{S}_{RE,FE} & \mathbf{S}_{RE,RE} \end{pmatrix} \begin{pmatrix} \mathbf{H}_{FE} & \mathbf{O} \\ \mathbf{O} & \mathbf{H}_{RE} \end{pmatrix}^{-1}, \quad (12)$$

with

$$\begin{aligned} \widehat{\mathbf{H}}_p &= -\frac{1}{n} \sum_{i=1}^n \frac{\partial^2 L_{pi}(\hat{\beta}_p)}{\partial \beta \partial \beta'}, & p = FE, RE, \\ \widehat{\mathbf{S}}_{pq} &= \frac{1}{n} \sum_{i=1}^n \frac{\partial L_{pi}(\hat{\beta}_p)}{\partial \delta} \frac{\partial L_{qi}(\hat{\beta}_q)}{\partial \beta'}, & p, q = FE, RE. \end{aligned}$$

where $\widehat{\mathbf{H}}_{FE}^{-1} (\widehat{\mathbf{S}}_{FE,FE}) \widehat{\mathbf{H}}_{FE}^{-1}$ and $\widehat{\mathbf{H}}_{RE}^{-1} (\widehat{\mathbf{S}}_{RE,RE}) \widehat{\mathbf{H}}_{RE}^{-1}$ are the cluster-robust variance-covariance matrices of $\hat{\beta}_{FE}$ and $\hat{\beta}_{RE}$, where the cluster is represented by the panel unit. Note that the `hausman` option is only allowed for static models.

as is for other Stata estimation commands, `xsmle` is unable to recognize non-linear specifications not based on factor variables, e.g. user defined second order terms or interactions.

3 The `xsmle` command

The `xsmle` command is written using the `optimize()` suite of functions, the optimization engine used by `ml`, and shares the same features of all Stata estimation commands. Version 10.1 is the earliest version of Stata that can be used to run `xsmle`. Only analytic weights (`aweight`) are allowed but the declared `weights` variable must be constant within each unit of the panel. `xsmle` supports the `mi` prefix while it does not support the `svy` prefix. Factor variables are allowed if version 11 (or later) is used to run the command.

There is one major prerequisite for using the command, which concerns the construction of the $n \times n$ matrix of spatial weights. This matrix can be a Stata matrix or a `spmat` object and it may follow any spatial weighting scheme, though it is usual to normalize spatial weights so that either the row or column sums are equal to 1.⁷ `xsmle` does not allow the use of time-varying weight matrices. This means that the weights matrix is forced to be the same for each cross-section and `xsmle` will automatically replicate it for all time periods. This could be a limitation especially in long panels, so a possible extension to `xsmle` may provide the option to read multiple (time-varying) weight matrices. It is worth noting that the maximum dimension of a single Stata matrix depends on Stata's flavor: 40×40 (Small) 800×800 and $11,000 \times 11,000$ (SE or MP). In order to overcome this limitation, bigger matrices have to be specified as `spmat` objects. A second requirement for `xsmle` is that the data must be `tsset` or `xtset` by the `panel` and `time` variables before the command is executed.⁸

The basic `xsmle` syntax is the following

```
xsmle depvar [indepvars] [if] [in] [weight] [, options]
```

The default is the random-effects SAR model. A description of the main estimation and postestimation options is provided below. A full description of all available options is provided in the `xsmle` help file.

3.1 Main options for `xsmle`

Options common to all spatial models

`model(name)` specifies the spatial model to be estimated. May be `sar` for the Spatial-AutoRegressive model, `sdm` for the Spatial Durbin Model, `sem` for the Spatial-Error Model, `sac` for the spatial-autoregressive with spatially autocorrelated errors model,

7. It is not assumed that \mathbf{W} is symmetric but $(\mathbf{I} - \rho\mathbf{W})$ must be non-singular. This implies conditions on the eigenvalues of \mathbf{W} that are discussed extensively in the literature (e.g., see LeSage and Pace (2009), chapter 3).

8. This avoids the necessity of adding syntax to specify the panel and time variables. However, there is a corollary that should be noted. The natural way of organizing spatial panel data for estimation purposes is to stack each panel unit for period $t = 1$ followed by panel units for $t = 2$ and so on. For this reason, `xsmle` internally sorts the dataset by time and panel unit but restores the original sorting on exit.

gspre for the generalised spatial random effects model. Default is `model(sar)`.

re use the random effects estimator; the default. This option cannot be specified when `model(sac)`.

fe use the fixed effects estimator. This option cannot be specified when `model(gspre)`.

`type(type_options [, leeyu])` specifies fixed-effects type; only for **fe** estimators. May be **ind** for spatial fixed effects, **time** for time fixed effects or **both** for both spatial and time fixed effects. Suboption **leeyu** allows to transform the data according to Lee and Yu (2010) approach and can be used only when `type(ind)`.

noconstant suppresses the constant term in the model. Only for **re** estimators.

effects computes direct, indirect and total effects and adds them to `e(b)`.

`vceeffetcsv(vee.type [, nsim(#)])` sets how the standard errors for the direct, indirect and total effects are computed. May be **dm** for Delta Method , **none** for no standard errors or **sim**[,nsim(#)] for Monte Carlo standard errors (the default), using # replications.

`constraints(constraints_list)` applies specified linear constraints.

`from(init_specs)` specifies initial values for the coefficients.

`level(#)` sets confidence level for confidence intervals; default is `level(95)`.

postscore save observation-by-observation scores in the estimation results list.

posthessian save the Hessian corresponding to the full set of coefficients in the estimation results list.

Variance estimation

This section describes the arguments of the `vce(vcetype)` option. `vce(vcetype)` specifies how to estimate the variance-covariance matrix (VCE) corresponding to the parameter estimates. The standard errors reported in the estimation result's table are the square root of the variances (diagonal elements) of the VCE. See `help vcetype` for more detail.

oim observed information matrix.

opg outer product of the gradients.

robust synonym for clustered sandwich estimator where the *clustvar* is the *panelvar*.

cluster *clustvar* clustered sandwich estimator.

dkraay(#) Driscoll-Kraay robust estimator. # is the maximum lag used in the calculation.

SAR model

wmatrix(*name*) specifies the weight matrix for the spatial-autoregressive term. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

dlag(*#*) defines the structure of the spatio-temporal model. When *#* is equal to 1, only time lagged dependent variable is included; when *#* is equal to 2, only space-time lagged dependent variable is included; when *#* is equal to 3, both time lagged and space-time lagged dependent variable are included.

`hausman` performs a robust version of the Hausman test. Allowed only for static models.

SDM model

wmatrix(*name*) specifies the weight matrix for the spatial-autoregressive term. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

dmatrix(*name*) specifies the weight matrix for the spatially lagged regressors; default is to use the matrix specified in `wmat`(*name*). *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

`durbin`(*dvarlist*) specifies the regressors that have to be spatially lagged; default is to lag all independent variables specified in *varlist*.

dlag(*#*) defines the structure of the spatio-temporal model. When *#* is equal to 1, only time lagged dependent variable is included; when *#* is equal to 2, only space-time lagged dependent variable is included; when *#* is equal to 3, both time lagged and space-time lagged dependent variable are included.

`hausman` performs a robust version of the Hausman test. Allowed only for static models

SEM model

ematrix(*name*) specifies the weight matrix for the spatial-autocorrelated error term. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

`hausman` performs a robust version of the Hausman test.

SAC model

wmatrix(*name*) specifies the weight matrix for the spatial-autoregressive term. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

ematrix(*name*) specifies the weight matrix for the spatial-autocorrelated error term. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

GSPRE model

`wmatrix(name)` specifies the weight matrix for the spatial-autocorrelated random-effects. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

`ematrix(name)` specifies the weight matrix for the spatial-autocorrelated error term. *name* can be a Stata matrix or a `spmat` object. This matrix can be standardized or not.

`error(#)` defines the structure of the model. When `#` is equal to 1, $\lambda \neq \phi \neq 0$; when `#` is equal to 2, $\lambda = 0$; when `#` is equal to 3, $\phi = 0$; when `#` is equal to 4, $\lambda = \phi$.

3.2 Postestimation command after xsmle

After an `xsmle` estimation, the `predict` command can be used to compute predicted values. Moreover, the latter allows the postestimation of fixed or random-effects. The methods implemented in this command are the panel data extension of those available in Kelejian and Prucha (2007) and Drukker et al. (2013c). See Section 4.1 for details. The syntax of the command is the following

```
predict [type] newvar [if] [in] [, statistics noie]
```

where `statistics` includes `rform`, `full`, `limited`, `naive`, `xb`, `a`.

`rform` the default, calculates predicted values from the reduced-form equation, $y_{it} = (\mathbf{I}_n - \rho \mathbf{W})^{-1}(x_{it}\boldsymbol{\beta} + \alpha_i)$.

`full` predicted values based on the full information set. This option is available only when `model(sac)`.

`limited` predicted values based on the limited information set. This option is available only when `model(sac)`.

`naive` predicted values based on the observed values of $y_{it} = \rho \mathbf{W}y_{it} + x_{it}\boldsymbol{\beta} + \alpha_i$.

`xb` calculates the linear prediction including the fixed or random effect $x_{it}\boldsymbol{\beta} + \alpha_i$.

`a` estimates α_i , the fixed or random-effect. In the case of fixed-effects models, this statistic is allowed only when `type(ind)`.

`noie` the estimated α_i , the fixed or random-effect, are excluded from the prediction.

4 Examples

4.1 Simulated data

In this section, we use simulated data to illustrate `xsmle` estimation capabilities focusing on model selection, prediction and estimation in presence of missing data.⁹ In particular, we consider the following fixed-effects SDM model

$$\begin{aligned}
 y_{it} = & 0.3 \sum_{j=1}^n w_{ij} y_{jt} + 0.5x_{1it} - 0.3x_{2it} - 0.2x_{3it} + 0.3 \sum_{j=1}^n w_{ij} x_{1it} + \\
 & + 0.6 \sum_{j=1}^n w_{ij} x_{2it} + 0.9 \sum_{j=1}^n w_{ij} x_{3it} + \mu_i + \epsilon_{it},
 \end{aligned} \tag{13}$$

where the nuisance parameters (μ_i) are drawn from an i.i.d. standard Gaussian random variable. To allow for dependence between the unit-specific effects and the regressors, we generate the latter as follows

$$x_{kit} = 0.4\mu_i + (1 - 0.4^2)^{1/2} z_{kit}, \tag{14}$$

where z_{kit} is standard Gaussian with $k = 1, 2, 3$. The sample size is set to 940 ($n = 188$ and $T = 5$) observations.¹⁰

Let us begin by importing a first-order spatial contiguity matrix of the italian local health authorities using the `spmat` command

```
. use ASL_contiguity_mat_ns.dta, clear
. spmat dta W W*, replace
```

The `spmat dta` command allows to store in the Stata memory a `spmat` object called `W`. Notice that, in order to estimate a model using `xsmle`, the spatial weight matrix must be available as Stata matrix or `spmat` object. The following `spmat` entry allows to easily summarize the `W` object

```
. spmat summarize W, links
Summary of spatial-weighting object W
```

Matrix	Description
Dimensions	188 x 188
Stored as	188 x 188
Links	
total	906
min	1
mean	4.819149
max	13

9. We report the code used for each example in the `sj_examples_simdata.do` accompanying file.

10. The chosen cross-sectional dimension ($n = 188$) depends on the dimension of the used weight matrix, a contiguity matrix of the italian local health authorities.

As can be seen, the imported spatial matrix consists of 188 cross-sectional units with at least one neighbor, with about 4.8 contiguous units on average. Since `xsmle` does not make this transformation automatically, the next step consists in the row-normalization of the `W` object. This task can be easily performed using the following

```
. spmat dta W W*, replace normalize(row)
```

We have now all the ingredients to estimate a spatial panel data model using `xsmle`. In particular, the syntax for estimating a fixed-effects SDM is

```
. xtset id t
      panel variable: id (strongly balanced)
      time variable: t, 1 to 5
      delta: 1 unit

. xsmle y x1 x2 x3, wmat(W) model(sdm) fe type(ind) nolog
Warning: All regressors will be spatially lagged

SDM with spatial fixed-effects                Number of obs =      940
Group variable: id                            Number of groups =    188
Time variable: t                              Panel length =       5

R-sq:    within = 0.3852
         between = 0.3705
         overall = 0.3635

Mean of fixed-effects = 0.0314
Log-likelihood = -1204.1194
```

	y	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
Main						
	x1	.5456416	.034473	15.83	0.000	.4780758 .6132075
	x2	-.2798453	.0356246	-7.86	0.000	-.3496683 -.2100224
	x3	-.1896873	.0356751	-5.32	0.000	-.2596093 -.1197654
Wx						
	x1	.3093954	.0716979	4.32	0.000	.16887 .4499207
	x2	.5063665	.0759508	6.67	0.000	.3575057 .6552273
	x3	.9072591	.0748364	12.12	0.000	.7605825 1.053936
Spatial						
	rho	.2274947	.0425135	5.35	0.000	.1441699 .3108196
Variance						
	sigma2_e	.7500305	.0347637	21.58	0.000	.6818948 .8181661

```
. estimates store sdm_fe
```

When the option `fe` is specified, `xsmle` fits a model with unit-specific fixed-effects. This means that, in the example above, we might omit the option `type(ind)`.¹¹ The

11. The option `nolog` is seldom used and allows to omit the display of the log-likelihood function iteration log. `xsmle` allows to use all `maximize` options available for `m1` estimation commands (see `help maximize`) plus the additional options `postscore` and `posthessian` which report the score and the hessian as an `e()` matrix. Notice that the usual limit for matrix dimension do apply in this case.

latter allows to specify alternative forms for the fixed-effects: `type(time)` allows for time fixed-effects while `type(both)` specifies both time and unit fixed-effects. Of special note is that, in the case of SDMs, `xsmle` also allows to specify a different set of spatially lagged explanatory variables through the option `durbin(varlist)`. As the warning message reports, the default is to lag all independent variables in `varlist`.

In order to simplify the task of producing publication quality tables, `xsmle` reports labeled estimation results. The **Main** equation contains the β vector, the **Wx** equation reports (only for SDM) the θ vector, the **Spatial** equation reports the spatial coefficients (in this case ρ) and the **Variance** equation reports ancillary parameters as the variance of the error (σ_ϵ^2 in this case).¹²

Even if we already know that in this example the fixed-effects SDM is correctly specified, we might be interested in testing the appropriateness of a random-effects variant using the official `hausman` command

```
. xsmle y x1 x2 x3, wmat(W) model(sdm) re type(ind) nolog
Warning: Option type(ind) will be ignored
Warning: All regressors will be spatially lagged

SDM with random-effects                Number of obs =      940
Group variable: id                     Number of groups =    188
Time variable: t                       Panel length =       5

R-sq:  within = 0.3671
       between = 0.5567
       overall = 0.4429

Log-likelihood = -1461.5464
```

	y	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
Main							
	x1	.6278704	.0383441	16.37	0.000	.5527173	.7030236
	x2	-.1595226	.0402597	-3.96	0.000	-.2384301	-.0806151
	x3	-.0807422	.0400913	-2.01	0.044	-.1593197	-.0021648
	_cons	.0214849	.0669073	0.32	0.748	-.109651	.1526208
Wx							
	x1	.3042129	.0784076	3.88	0.000	.1505368	.4578889
	x2	.5215032	.0805461	6.47	0.000	.3636356	.6793707
	x3	.9631849	.0813256	11.84	0.000	.8037896	1.12258
Spatial							
	rho	.2558274	.040904	6.25	0.000	.175657	.3359977
Variance							
	lgt_theta	-.0751917	.1284863	-0.59	0.558	-.3270202	.1766369
	sigma2_e	.9648846	.0515123	18.73	0.000	.8639224	1.065847

```
. estimates store sdm_re
. hausman sdm_fe sdm_re, eq(1:1 2:2 3:3)

----- Coefficients -----
| (b) (B) (b-B) sqrt(diag(V_b-V_B))
```

12. Notice that for models other than SDM, the ancillary equations will be different following the specific parametrization used.

		sdm_fe	sdm_re	Difference	S.E.
comp1	x1	.5456416	.6278704	-.0822288	.
	x2	-.2798453	-.1595226	-.1203227	.
	x3	-.1896873	-.0807422	-.1089451	.
comp2	x1	.3093954	.3042129	.0051825	.
	x2	.5063665	.5215032	-.0151366	.
	x3	.9072591	.9631849	-.0559257	.
comp3	rho	.2274947	.2558274	-.0283326	.011587

b = consistent under Ho and Ha; obtained from xsmle
 B = inconsistent under Ha, efficient under Ho; obtained from xsmle
 Test: Ho: difference in coefficients not systematic

$$\chi^2(7) = (b-B)'[(V_b-V_B)^{-1}](b-B)$$

$$= -75.83 \quad \chi^2 < 0 \implies \text{model fitted on these data fails to meet the asymptotic assumptions of the Hausman test; see suest for a generalized test}$$

In this example the Hausman statistic fails to meet its asymptotic assumptions. This problem can be overcome by adding the option `hausman` to the estimation command, that is

```
. xsmle y x1 x2 x3, wmat(W) model(sdm) fe type(ind) hausman nolog
Warning: All regressors will be spatially lagged
... estimating random-effects model to perform Hausman test
SDM with spatial fixed-effects                               Number of obs =      940
Group variable: id                                         Number of groups =     188
Time variable: t                                           Panel length =       5
R-sq:  within = 0.3852
      between = 0.3705
      overall = 0.3635
Mean of fixed-effects = 0.0314
Log-likelihood = -1204.1194
```

	y	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
Main	x1	.5456416	.034473	15.83	0.000	.4780758 .6132075
	x2	-.2798453	.0356246	-7.86	0.000	-.3496683 -.2100224
	x3	-.1896873	.0356751	-5.32	0.000	-.2596093 -.1197654
Wx	x1	.3093954	.0716979	4.32	0.000	.16887 .4499207
	x2	.5063665	.0759508	6.67	0.000	.3575057 .6552273
	x3	.9072591	.0748364	12.12	0.000	.7605825 1.053936
Spatial	rho	.2274947	.0425135	5.35	0.000	.1441699 .3108196

Variance						
sigma2_e	.7500305	.0347637	21.58	0.000	.6818948	.8181661

Ho: difference in coeffs not systematic chi2(7) = 91.10 Prob>=chi2 = 0.0000

As expected, in this case we strongly reject the null hypothesis, with a χ^2 test statistic equal to 91.10 and a p -value lower than one percent. It is worth noting that, if specified, the `hausman` option automatically detects which is the alternative model, in our example the random-effects one.

Another common task routinely undertaken by spatial practitioners is model selection. Following the strategy described in LeSage and Pace (2009) and Elhorst (2010), investigators should start with the SDM as a general specification and test for the alternatives. That is, we estimate a SDM but we would like to know if it is the best model for the data at hand. This kind of procedure can be easily implemented using `xsmle`. For instance, one may be interested in testing for SAR or SEM specifications. Since the SDM may be easily derived starting from a SEM, it is easily shown that if $\theta = 0$ and $\rho \neq 0$ the model is a SAR, while if $\theta = -\beta\rho$ the model is a SEM. After the estimation of the SDM, these tests can be performed by exploiting the `xsmle` “equation-labeled” vector of estimated coefficients and the official `test` and `testnl` commands as follows

```
. test [Wx]x1 = [Wx]x2 = [Wx]x3 = 0
( 1) [Wx]x1 - [Wx]x2 = 0
( 2) [Wx]x1 - [Wx]x3 = 0
( 3) [Wx]x1 = 0
      chi2( 3) = 203.77
      Prob > chi2 = 0.0000
. testnl ([Wx]x1 = -[Spatial]rho*[Main]x1) ([Wx]x2 = -[Spatial]rho*[Main]x2) ///
>        ([Wx]x3 = -[Spatial]rho*[Main]x3)
(1) [Wx]x1 = -[Spatial]rho*[Main]x1
(2) [Wx]x2 = -[Spatial]rho*[Main]x2
(3) [Wx]x3 = -[Spatial]rho*[Main]x3
      chi2(3) = 193.70
      Prob > chi2 = 0.0000
```

Finally, since the SAC and SDM are non-nested, information criteria can be used to test if the most appropriate model is the SAC using

```
. est restore sdm_fe
(results sdm_fe are active now)
. estat ic
Akaike's information criterion and Bayesian information criterion
```

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
sdm_fe	940	.	-1204.119	8	2424.239	2463.006

Note: N=Obs used in calculating BIC; see [R] BIC note.

and

```
. xsmle y x1 x2 x3, wmat(W) emat(W) model(sac) fe type(ind) nolog
SAC with spatial fixed-effects          Number of obs =      940
Group variable: id                      Number of groups =    188
Time variable: t                        Panel length =       5
R-sq:  within = 0.2208
      between = 0.0007
      overall = 0.0667
Mean of fixed-effects = 0.0831
Log-likelihood = -1290.9574
```

	y	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
Main							
	x1	.4860935	.0415495	11.70	0.000	.4046579	.5675291
	x2	-.3332588	.0370124	-9.00	0.000	-.4058019	-.2607158
	x3	-.3039008	.0371472	-8.18	0.000	-.3767081	-.2310936
Spatial							
	rho	-.134535	.1106866	-1.22	0.224	-.3514768	.0824067
	lambda	.4760945	.0877639	5.42	0.000	.3040804	.6481085
Variance							
	sigma2_e	1.073918	.0469018	22.90	0.000	.9819918	1.165844

```
. estat ic
Akaike's information criterion and Bayesian information criterion
```

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
.	940	.	-1290.957	6	2593.915	2622.99

Note: N=Obs used in calculating BIC; see [R] BIC note.

As expected, in this case all tests point towards a fixed-effects SDM. Finally, one may be interested in the post-estimation of fixed-effects and/or predicted values of the outcome variable. Section 4.1 gives a brief overview of the spatial predictors implemented in `xsmle`. They are the panel data extension of those discussed in Kelejian and Prucha (2007), which range from the suboptimal *naive* predictor to the efficient minimum mean square error *full information* predictor. Here, we just give some example on the `xsmle` post-estimation syntax. For instance, to post-estimate the fixed-effects once a fixed-effects spatial model has been estimated, the syntax is

```
. est restore sdm_fe
(results sdm_fe are active now)
. predict alphahat, a
```

Now, in order to immediately visualize the deviation between the true (simulated) and the estimated μ_i 's, we may plot them using

```
. tw (kdensity alpha,lp(dot) lw(*2)) (kdensity alphahat, lp(dash)), ///
```

```
> legend(row(1) lab(1 "True") lab(2 "Estimated"))
```

The resulting plot is shown in Figure 1. Similarly, we can obtain a reduced form and *naive* prediction of the outcome variable using (the resulting plot is shown in Figure 2)

```
. predict yhat_rform
(option rform assumed)
. predict yhat_naive, naive
.
. tw (kdensity y,lp(dot) lw(*2)) (kdensity yhat_rform, lp(dash)) ///
> (kdensity yhat_naive), legend(row(1) lab(1 "True") ///
> lab(2 "Reduced form") lab(3 "Naive"))
```

Post-estimation

In this section, we briefly discuss the predictors available in `xsmle` and replicate the Kelejian and Prucha (2007) Monte Carlo study extending it to the case of panel data. Let us consider the following spatial autoregressive with spatially autocorrelated errors model

$$\mathbf{y}_t = \rho \mathbf{W} \mathbf{y}_t + \mathbf{X}_t \boldsymbol{\beta} + \boldsymbol{\mu} + \boldsymbol{\nu}_t, \quad (15)$$

$$\boldsymbol{\nu}_t = \lambda \mathbf{M} \boldsymbol{\nu}_t + \boldsymbol{\epsilon}_t, \quad (16)$$

for which we use the same notation discussed in Section 2. In this model y_{it} is determined as

$$y_{it} = \rho \mathbf{w}_i \cdot \mathbf{y}_t + \mathbf{x}_{it} \boldsymbol{\beta} + \mu_i + \nu_{it}, \quad (17)$$

$$\nu_{it} = \lambda \mathbf{m}_i \cdot \boldsymbol{\nu}_t + \epsilon_{it}, \quad (18)$$

where, for $t = 1, \dots, T$, \mathbf{w}_i and \mathbf{m}_i are the i -th rows of \mathbf{W} and \mathbf{M} , \mathbf{x}_{it} is the i -th row of \mathbf{X}_t , ν_{it} and ϵ_{it} are the i -th elements of $\boldsymbol{\nu}_t$ and $\boldsymbol{\epsilon}_t$, μ_i is the i -th element of $\boldsymbol{\mu}$, and $\mathbf{w}_i \cdot \mathbf{y}_t$ and $\mathbf{m}_i \cdot \boldsymbol{\nu}_t$ denote the i -th elements of the spatial lags $\mathbf{W} \mathbf{y}_t$ and $\mathbf{M} \boldsymbol{\nu}_t$ with $\mathbf{w}_i \cdot \mathbf{y}_t$ that does not include y_{it} . By making the same assumptions of Kelejian and Prucha (2007), we have that (see Kelejian and Prucha 2007, for more details on model assumptions)

$$\boldsymbol{\nu}_t \sim \mathcal{N}(0, \sigma_\epsilon^2 \boldsymbol{\Sigma}^{\nu_t}),$$

$$\mathbf{y}_t \sim \mathcal{N}(\boldsymbol{\xi}_t, \sigma_\epsilon^2 \boldsymbol{\Sigma}^{\mathbf{y}_t}),$$

with

$$\boldsymbol{\xi}_t = (\mathbf{I} - \rho \mathbf{W})^{-1} (\mathbf{X}_t \boldsymbol{\beta} + \boldsymbol{\mu}), \quad (19)$$

$$\boldsymbol{\Sigma}^{\nu_t} = (\mathbf{I} - \lambda \mathbf{M})^{-1} (\mathbf{I} - \lambda \mathbf{M}')^{-1}, \quad (20)$$

$$\boldsymbol{\Sigma}^{\mathbf{y}_t} = (\mathbf{I} - \rho \mathbf{W})^{-1} \boldsymbol{\Sigma}^{\nu_t} (\mathbf{I} - \rho \mathbf{W}')^{-1}. \quad (21)$$

We consider three information sets

$$\begin{aligned} \boldsymbol{\Lambda}_1 &= \{\mathbf{X}_t, \mathbf{W}\}, \\ \boldsymbol{\Lambda}_2 &= \{\mathbf{X}_t, \mathbf{W}, \mathbf{w}_i \cdot \mathbf{y}_t\}, \\ \boldsymbol{\Lambda}_3 &= \{\mathbf{X}_t, \mathbf{W}, \mathbf{y}_{t,-1}\}, \quad t = 1, \dots, T, \end{aligned} \quad (22)$$

where Λ_3 is the full information set containing all $n - 1$ observations on \mathbf{y}_t ; Λ_1 and Λ_2 are both subsets of Λ_3 . We consider the following four predictors of y_{it} (denoted as $y_{it}^{(p)}$ with $p = 1 \dots, 4$)¹³

$$\begin{aligned} y_{it}^{(1)} &= E(y_{it}|\Lambda_1) \\ &= (\mathbf{I} - \rho\mathbf{W})_{i.}^{-1}(\mathbf{X}_t\boldsymbol{\beta} + \boldsymbol{\mu}), \end{aligned} \quad (23)$$

$$\begin{aligned} y_{it}^{(2)} &= E(y_{it}|\Lambda_2) \\ &= \rho\mathbf{w}_i.\mathbf{y}_t + \mathbf{x}_{it}\boldsymbol{\beta} + \mu_i + \frac{\text{cov}(\nu_{it}, \mathbf{w}_i.\mathbf{y}_t)}{\text{var}(\mathbf{w}_i.\mathbf{y}_t)}[\mathbf{w}_i.\mathbf{y}_t - E(\mathbf{w}_i.\mathbf{y}_t)], \end{aligned} \quad (24)$$

$$\begin{aligned} y_{it}^{(3)} &= E(y_{it}|\Lambda_3) \\ &= \rho\mathbf{w}_i.\mathbf{y}_t + \mathbf{x}_{it}\boldsymbol{\beta} + \mu_i + \text{cov}(\nu_{it}, \mathbf{y}_{t,-i})[\text{VC}(\mathbf{y}_{t,-i})]^{-1}[\mathbf{y}_{t,-i} - E(\mathbf{y}_{t,-i})], \end{aligned} \quad (25)$$

$$y_{it}^{(4)} = \rho\mathbf{w}_i.\mathbf{y}_t + \mathbf{x}_{it}\boldsymbol{\beta} + \mu_i, \quad (26)$$

where

$$\begin{aligned} E(\mathbf{w}_i.\mathbf{y}_t) &= \mathbf{w}_i.(\mathbf{I} - \rho\mathbf{W})^{-1}(\mathbf{X}_t\boldsymbol{\beta} + \boldsymbol{\mu}), \\ \text{var}(\mathbf{w}_i.\mathbf{y}_t) &= \sigma_\epsilon^2\mathbf{w}_i.\boldsymbol{\Sigma}^{\mathbf{y}_t}\mathbf{w}_i', \\ \text{cov}(\mu_i, \mathbf{w}_i.\mathbf{y}_t) &= \sigma_\epsilon^2\boldsymbol{\Sigma}_{i.}^{\nu_t}(\mathbf{I} - \rho\mathbf{W}')^{-1}\mathbf{w}_i', \\ E(\mathbf{y}_{t,-i}) &= \mathbf{S}_{t,-i}(\mathbf{I} - \rho\mathbf{W})^{-1}(\mathbf{X}_t\boldsymbol{\beta} + \boldsymbol{\mu}), \\ \text{VC}(\mathbf{y}_{t,-i}) &= \sigma_\epsilon^2\mathbf{S}_{t,-i}\boldsymbol{\Sigma}^{\mathbf{y}_t}\mathbf{S}_{t,-i}', \\ \text{cov}(\nu_{it}, \mathbf{y}_{t,-i}) &= \sigma_\epsilon^2\boldsymbol{\Sigma}_{i.}^{\nu_t}(\mathbf{I} - \rho\mathbf{W}')^{-1}\mathbf{S}_{t,-i}'. \end{aligned}$$

In the above expressions, $(\mathbf{I} - \rho\mathbf{W})_{i.}^{-1}$ and $\boldsymbol{\Sigma}_{i.}^{\nu_t}$ denote the i -th rows of $(\mathbf{I} - \rho\mathbf{W})^{-1}$ and $\boldsymbol{\Sigma}^{\nu_t}$ respectively, while $\mathbf{S}_{t,-i}$ is the $n - 1 \times n$ selector matrix which is identical to the $n \times n$ identity matrix \mathbf{I} except that the i -th row of \mathbf{I} is deleted.

We now turn to the comparison of the above predictors in terms of predictive efficiencies, extending the Kelejian and Prucha (2007) Monte Carlo design to fixed-effects models like the one reported in equations (15)-(16). In particular, we consider the following fixed-effects SAC model

$$\begin{aligned} y_{it} &= \rho \sum_{j=1}^n w_{ij}y_{jt} + 0.5x_{1it} + \mu_i + \nu_{it} \\ \nu_{it} &= \lambda \sum_{j=1}^n w_{ij}\nu_{jt} + \epsilon_{it} \end{aligned}$$

where the nuisance parameters μ_i are drawn from an i.i.d. standard Gaussian random variable while the x_{1it} regressor is generated according to (14). The simulation is based on what Kelejian and Prucha (2007) describe as ‘‘two ahead and two behind’’ weight

13. $p = 1$ indicates the *reduced form* predictor, $p = 2$ indicates the *limited information* predictor, $p = 3$ indicates the *full information* predictor and $p = 4$ indicates the *naive* predictor.

matrix in which each unit is directly related to the two units which are immediately after it and immediately before it in the ordering. The matrix is row normalized and all of its nonzero elements are equal to $1/4$.¹⁴ As in Kelejian and Prucha (2007), we report results for 25 combinations of $\rho, \lambda = -0.9, -0.4, 0, 0.4, 0.9$ and set $\sigma_\epsilon^2 = 1$. The sample size is set to 500 ($n = 100$ and $T = 5$) observations. Notice that when $\rho = 0$ results refer to the SEM.

Simulation results in terms of sample averages over $i = 1, \dots, 100$ and $t = 1, \dots, 5$ for $MSE(y_{it}^{(p)})$ for $p = 2, \dots, 4$ are given in Table 3.¹⁵ As expected, even in the panel data case, numerical results are fully consistent with the theoretical notions reported in Kelejian and Prucha (2007): the biased **naive** predictor is the worst especially when $\rho = \lambda = 0.9$ while the **full information** predictor is always the best.

On marginal effects

As already mentioned in Section 2.1, a peculiar feature of spatial regression models is the feedback process among spatially correlated units which leads to the distinction between direct, indirect and total marginal effects. In order to show how to compute these effects using `xsmle`, let us consider the data generating process of the following dynamic fixed-effects SDM model

$$\begin{aligned}
 y_{it} = & \tau y_{it-1} + \psi \sum_{j=1}^n w_{ij} y_{jt-1} + 0.2 \sum_{j=1}^n w_{ij} y_{jt} + 0.5x_{1it} - 0.3x_{2it} - 0.2x_{3it} + \\
 & + 0.3 \sum_{j=1}^n w_{ij} x_{1it} + 0.6 \sum_{j=1}^n w_{ij} x_{2it} + 0.9 \sum_{j=1}^n w_{ij} x_{3it} + \mu_i + \epsilon_{it}, \quad (27)
 \end{aligned}$$

where, as for the d.g.p. reported in equation (13), the nuisance parameters are drawn from an i.i.d. standard Gaussian random variable and the correlation between unit-specific effects and regressors is obtained according to equation (14). The sample size is set to 1960 observations ($n = 196$ and $T = 10$) and $\tau = \psi = 0.3$.¹⁶

As documented in Section 3.1, `xsmle` allows the estimation of model (27) by specifying the option `dlag(3)`.¹⁷ By adding the `effects` option `xsmle` will also compute direct, indirect and total effects

```
. xsmle y x1 x2 x3, wmat(Wspmat) model(sdm) fe dlag(3) effects nolog
Warning: All regressors will be spatially lagged
```

14. See Kelejian and Prucha (2007) for more details on the structure of this weight matrix. Clearly, the results reported here depend on the structure of this matrix.

15. Since it has by far the worst performance, we do not report results for the *reduced form* predictor.

16. We thank Jihai Yu for sharing his Matlab code for creating the rook spatial weights matrix used in this example. The original code has been translated into mata for our purposes (see the `sj_examples_simdata.do` accompanying file for details).

17. `dlag(1)` allows the estimation of model (27) in which $\psi = 0$, while `dlag(2)` the case in which $\tau = 0$.

LR_Total							
x1	3.646213	.4830534	7.55	0.000	2.699446	4.59298	
x2	1.229374	.3028385	4.06	0.000	.6358214	1.822927	
x3	3.391038	.5056929	6.71	0.000	2.399898	4.382178	

Notice that, when the `effects` option is specified, the marginal effects will be both displayed and added to the estimated vector $e(b)$. Given its dynamic nature, model (27) implies both short- and long-run effects (see Table 2). In these cases, short-run effects are reported under the three equations labeled `SR.Direct`, `SR.Indirect` and `SR.Total` while long-run effects are reported under `LR.Direct`, `LR.Indirect` and `LR.Total`.¹⁸ Equivalently, short-run total effects can be obtained through `margins` using the following syntax

```
. margins, dydx(*) predict(rform noie)
Average marginal effects           Number of obs   =       1,764
Model VCE      : OIM
Expression    : Reduced form prediction, predict(rform noie)
dy/dx w.r.t.  : x1 x2 x3
```

	Delta-method		z	P> z	[95% Conf. Interval]	
	dy/dx	Std. Err.				
x1	.9699528	.0681065	14.24	0.000	.8364665	1.103439
x2	.3283914	.0669413	4.91	0.000	.1971888	.459594
x3	.9068859	.0712235	12.73	0.000	.7672903	1.046481

In order to ensure that `margins` works, the `noie`'s `xsmle` postestimation option has been added through the `predict()`'s option of `margins`. As can be noted the two procedures produce slightly different results. This is due to the fact that `xsmle`, by default, uses the Monte Carlo procedure outlined in LeSage and Pace (2009), hence the point estimates (standard errors) are averages (standard deviations) over the (default) 500 Monte Carlo replications. The same point estimates can be obtained using `xsmle` with the `vceeffects(none)` option¹⁹

```
. qui xsmle y x1 x2 x3, wmat(Wspmat) model(sdm) fe dlag(3) effects vceeffects(none)
. est sto dsdm_fe
. estout dsdm_fe, keep(SR_Total:) c(b)
```

	dsdm_fe
	b
SR_Total	
x1	.9699528
x2	.3283914
x3	.9068859

18. Equation names follow Elhorst (2014) terminology on short- and long-run marginal effects.

19. The `vceeffects(none)` option suppresses the computation of standard errors.

Since the analytical formulas for direct, indirect and total effects reported in Table 2 imply a linear (in variables) specification, `xsmle` suppresses the computation of these effects when factor variables are specified, as shown in the example below

```
. xsmle y c.x1#c.x1 c.x1#c.x2 c.x2 c.x3, wmat(Wspmat) model(sdm) fe dlag(3) ///
> effects nolog
Warning: All regressors will be spatially lagged
Warning: direct and indirect effects cannot be computed if factor variables are specified
        option -effects- ignored. Notice that total effects can be obtained using -margins-
Dynamic SDM with spatial fixed-effects          Number of obs =      1764
Group variable: id                             Number of groups =      196
Time variable: t                               Panel length =        9
R-sq:      within = 0.3929
           between = 0.9122
           overall = 0.8378
Mean of fixed-effects = 0.0395
Log-likelihood = -2389.1429
```

	y	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
Main						
	y					
	L1.	.282856	.0187596	15.08	0.000	.2460878 .3196242
	Wy					
	L1.	.3418741	.0311604	10.97	0.000	.2808008 .4029474
	x1	.4766421	.0261214	18.25	0.000	.4254451 .5278391
	c.x1#c.x1	.0446299	.0171106	2.61	0.009	.0111027 .0781571
	c.x1#c.x2	-.0847021	.0243227	-3.48	0.000	-.1323737 -.0370306
	x2	-.2720875	.0263194	-10.34	0.000	-.3236725 -.2205025
	x3	-.1824236	.0267751	-6.81	0.000	-.2349018 -.1299453
Wx						
	x1	.3633111	.0517657	7.02	0.000	.2618521 .4647701
	c.x1#c.x1	.0073658	.0327561	0.22	0.822	-.056835 .0715667
	c.x1#c.x2	-.0608174	.0472681	-1.29	0.198	-.1534612 .0318264
	x2	.5585988	.0496918	11.24	0.000	.4612047 .6559929
	x3	.9484272	.0501452	18.91	0.000	.8501445 1.04671
Spatial						
	rho	.1509144	.0287019	5.26	0.000	.0946597 .2071691
Variance						
	sigma2_e	.9534039	.0289549	32.93	0.000	.8966533 1.010154

Nonetheless, when the specification includes factor variables `xsmle` allows for the use of `margins` to compute total marginal effects

```
. margins, dydx(x1 x2 x3) predict(rform noie)
Warning: cannot perform check for estimable functions.
Average marginal effects          Number of obs    =    1,764
Model VCE      : OIM
Expression    : Reduced form prediction, predict(rform noie)
dy/dx w.r.t. : x1 x2 x3
```

	Delta-method				
	dy/dx	Std. Err.	z	P> z	[95% Conf. Interval]
x1	.9952611	.0683488	14.56	0.000	.8613 1.129222
x2	.3278821	.0665554	4.93	0.000	.1974359 .4583284
x3	.9021512	.0707735	12.75	0.000	.7634377 1.040865

`xsmle` also offers the opportunity to compute standard errors using the Delta Method through the `vceffctcs(dm)` option²⁰.

```
. xsmle y x1 x2 x3, wmat(Wspmat) model(sdm) fe dlag(3) effects vceffects(dm) nolog
Warning: All regressors will be spatially lagged
Computing marginal effects standard errors using delta-method...
Dynamic SDM with spatial fixed-effects          Number of obs =    1764
Group variable: id                             Number of groups =    196
Time variable: t                               Panel length =     9
R-sq:      within = 0.3876
           between = 0.9108
           overall = 0.8354
Mean of fixed-effects = 0.0708
Log-likelihood = -2396.3051
```

	y	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
Main						
	y					
	L1.	.278483	.0187886	14.82	0.000	.2416579 .315308
	Wy					
	L1.	.3371464	.0312009	10.81	0.000	.2759938 .3982989
	x1	.471855	.0261821	18.02	0.000	.420539 .523171
	x2	-.2774485	.0263341	-10.54	0.000	-.3290623 -.2258347
	x3	-.1814445	.0268751	-6.75	0.000	-.2341187 -.1287704
Wx						
	x1	.3501276	.0516946	6.77	0.000	.2488081 .4514471
	x2	.5557425	.0498404	11.15	0.000	.4580572 .6534278
	x3	.9499813	.0503458	18.87	0.000	.8513054 1.048657
Spatial						

20. While using the Delta Method ensures that the results do not depend on stochastic variability, this is obtained at the price of a more computationally intensive procedure

rho	.152554	.0287441	5.31	0.000	.0962165	.2088915
Variance						
sigma2_e	.9612217	.0291937	32.93	0.000	.9040031	1.01844
SR_Direct						
x1	.4889378	.0262506	18.63	0.000	.4374875	.5403881
x2	-.2566706	.0262086	-9.79	0.000	-.3080385	-.2053026
x3	-.144119	.02613	-5.52	0.000	-.1953328	-.0929052
SR_Indirect						
x1	.481015	.058585	8.21	0.000	.3661905	.5958395
x2	.585062	.0580324	10.08	0.000	.4713206	.6988034
x3	1.051005	.0630431	16.67	0.000	.9274428	1.174567
SR_Total						
x1	.9699528	.0681065	14.24	0.000	.8364665	1.103439
x2	.3283914	.0669413	4.91	0.000	.1971888	.459594
x3	.9068859	.0712235	12.73	0.000	.7672903	1.046481
LR_Direct						
x1	.8860048	.0680006	13.03	0.000	.7527261	1.019283
x2	-.25736	.0472579	-5.45	0.000	-.3499839	-.1647362
x3	.0347029	.0566012	0.61	0.540	-.0762334	.1456392
LR_Indirect						
x1	2.659826	.5536009	4.80	0.000	1.574789	3.744864
x2	1.457852	.2844823	5.12	0.000	.9002767	2.015427
x3	3.280576	.5347181	6.14	0.000	2.232548	4.328604
LR_Total						
x1	3.545831	.6090295	5.82	0.000	2.352155	4.739507
x2	1.200492	.3154343	3.81	0.000	.5822519	1.818732
x3	3.315279	.5776118	5.74	0.000	2.183181	4.447377

Unbalanced panels

Missing data can pose major problems when estimating econometric models since it is generally unlikely that missing values are missing completely at random. Most important here is that, in general, `xsmle` cannot handle unbalanced panels. A strategy to address this issue without relying on more complex econometric approaches is represented by multiple imputation, that is the process of replacing missing values by multiple sets of plausible values. This section provides a simple example in which `xsmle` is used together with `mi`, the Stata's suite of commands dealing with multiple data imputation, to overcome the hurdle. Let us consider the same data generating process reported in equation (13). The following syntax allows to randomly assign 5 percent missing values to the x_{1it} covariate²¹

```
. set seed 12345
```

21. As usual, a good practice to obtain reproducible results is to set the seed of the Stata's pseudo random number generator using the command `set seed #`, where `#` is any number between 0 and $2^{31}-1$.

```
. replace x1 = . if uniform()<0.05
(49 real changes made, 49 to missing)
```

The first step is to declare the dataset as a `mi` dataset. The command `mi set` is the appropriate one. Indeed, data must be `mi set` before other `mi` commands can be used. In this example, we choose the `wide` style. The second step is to register (declare) the variables with missing values using the `mi register` command

```
. mi set wide
. mi register imputed x1
. mi impute regress x1 = z, add(50) rseed(12345)
Univariate imputation          Imputations =      50
Linear regression              added =      50
Imputed: m=1 through m=50      updated =      0
```

Variable	Observations per <i>m</i>			Total
	Complete	Incomplete	Imputed	
x1	891	49	49	940

(complete + incomplete = total; imputed is the minimum across *m* of the number of filled-in observations.)

Then, the command `mi impute regress x1 = z, add(50) rseed(12345)` is used here to fill in `x1`'s missing values using the linear regression method and the `z` covariate as predictor.²² The option `add(50)` specifies the number of imputations to be added (currently, the total number of imputations cannot exceed 1,000). After the command `mi impute` has been executed, fifty new variables `_#_x1` (with `# = 1, ..., 50`) will be added to the dataset, each representing an imputed version of `x1`. Finally, the command

```
. mi estimate, dots post: xsmle y x1 x2 x3, wmat(W) model(sdm) fe type(ind) nolog
Imputations (50):
.....10.....20.....30.....40.....50 done
Multiple-imputation estimates          Imputations =      50
SDM with spatial fixed-effects        Number of obs =     940
                                      Average RVI   =     0.0452
                                      Largest FMI   =     0.1304
DF adjustment:  Large sample          DF:    min    =  2,908.95
                                      avg      = 126,717.41
                                      max      = 516,684.95
Model F test:      Equal FMI          F( 8,205401.9) =    130.14
Within VCE type:  OIM                 Prob > F      =     0.0000
```

	y	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
Main	x1	.509667	.0367065	13.88	0.000	.4377079 .581626

22. See `help mi impute` for details on the available imputation methods. The `z` covariate is a standard Gaussian random variable specifically designed to be correlated with `x1`. See the code reported in the `sj_examples_simdata.do` file for details.

	x2	-.2737751	.0363977	-7.52	0.000	-.3451134	-.2024368
	x3	-.1947675	.036523	-5.33	0.000	-.2663518	-.1231832
Wx							
	x1	.2788079	.0769524	3.62	0.000	.1279211	.4296947
	x2	.5316003	.0779399	6.82	0.000	.3788391	.6843615
	x3	.8991836	.0768688	11.70	0.000	.748522	1.049845
Spatial							
	rho	.2471005	.042971	5.75	0.000	.1628754	.3313257
Variance							
	sigma2_e	.7751222	.0364928	21.24	0.000	.7035961	.8466484

allows to exploit `xsmle` to estimate the fixed-effects SDM using the fifty imputed versions of the `x1` variable. In this way, both the coefficients and the standard errors will be adjusted for the between-imputations variability according to the combination rules given in Rubin (1987). We replicated the same exercise by assigning (at random) a higher percentage (10 and 20 percent) of missing values to the `x1` covariate. In order to offer an example in which the multiple imputation strategy directly affects the ρ parameter value, we used the same strategy assigning 5, 10 and 20 percent missing values to the dependent variable.²³

The upper panel of Table 4 reports the results for the case in which `x1` is the missing variable. As expected, the bias affecting the β_1 parameter increases when the number of imputed values grows. The same is true for the ρ parameter when the missing values are in the dependent variable (lower panel of Table 4). It is worth noting that, even if these are not the results of a Monte Carlo simulation, it looks like that the effect of missing values is stronger on ρ than on β_1 .

4.2 Real data

As an example of the use of spatial panel models with real data, we make use of a dataset on electricity usage at state level in the United States. The data covers the 48 states in the continental United States plus the district of Columbia for the period 1990-2010. It is drawn from the Electric Power Annual compiled by the Department of Energy's Energy Information Agency together with general economic, demographic and weather information from other US statistical agencies including the Bureau of Labor Statistics and the Census Bureau.²⁴

The analysis focuses on the response of residential electricity demand to prices and weather/climate conditions. The spatial dimension arises in at least two ways:

- Relative prices in neighboring states may influence decisions about the location

23. The interested reader can find the related Stata code in the accompanying `sj_examples_simdata.do` file.

24. The interested reader can find the Stata code and data used for this application in the accompanying `sj_empirical_application.do`, `wstate_rook.spmat` and `state_spatial.dbf.dta` files.

of economic activities and subsequently of the residence. Electricity prices in California are high by comparison with prices in the North West and also by comparison with prices in parts of the Mid West, but one would expect that location decisions and, thus, electricity demand will be more strongly influenced by prices in the North West than in the Mid West. In modeling terms, this behavior may be manifested as a significant coefficient on spatially weighted prices or on the spatially lagged dependent variable.

- Both weather and climate variables may serve as proxies for short and long term regional influences on the location of economic activity, the energy efficiency of buildings, and other determinants of electricity use. Given the physical capital stock, annual variations in weather will affect electricity demand for air conditioning or heating. Hence, it is interesting to determine whether local or regional weather variables have a statistically distinct influence on electricity demand.

Note that the logic which suggests a role for spatial influences on electricity demand in each state, do not imply direct spatial interactions for the dependent variable as in cases where it is argued that policy decisions in one state, for example, taxes on property, are influenced by decisions made by neighboring states. Instead, the arguments reflect a combination of omitted variables that may be spatially correlated plus the spatially distributed influence of variables that would be included in any model of electricity demand.

Tables from 5 to 7 summarize the results obtained when fixed-effects models are used to examine residential demand for electricity using the log of residential consumption per person as the dependent variable. Elhorst and others argue that fixed-effects models are more appropriate for such data because the sample represents the complete population of US continental states rather than a random sample drawn from that population. This claim is supported by the evidence given in the last two lines of Table 5 where all the static random-effects specifications are strongly rejected by the Hausman test. The models do not provide a comprehensive analysis of factors which may influence demand but they have been refined to focus on key variables which explain changes in electricity demand over the last two decades. For residential consumption, the large differences between the within and between R^2 statistics, other than for the models which include the lagged (in time) dependent variable, confirm the importance of state fixed-effects associated with variables that are not included in the analysis or which cannot be identified in this specification. Nonetheless, the within R^2 statistics, at least equal to 0.82, show that the models can account for a large proportion of variation over time in electricity consumption for residential usage by state. Weather variables, both heating and cooling degree days, have an important influence on residential consumption and so does the size of the housing stock.²⁵

Table 5 shows that there are strong spatial interactions in residential consumption. The coefficients of the spatially lagged dependent variable (ρ) and of the spatially

25. We test for alternative measures of income and the best indicator seems to be personal disposable income adjusted for differences in the cost of living across states (using the ACCRA cost of living index) and for changes in the CPI over time.

weighted price are highly significant and appear to have quite separate influences on consumption. Spatial Durbin variables with coefficients that are not significantly different from zero have been dropped from the model. The results in Table 6 reinforce recommendations from LeSage & Pace and Elhorst that investigators should start with the SDM as a general specification and test for the exclusion of variables for nested models using LR tests while, for the SAC model, we adopted the modified AIC criterion as in Burnham (2004). The positive ρ coefficient is consistent with omitted regional factors that vary over time and affect residential consumption. The positive coefficient on the spatially weighted average price in neighboring states indicates a clear displacement effect by which an increase in electricity prices in one state encourages a shift in demand from that state to neighboring states. That is an important constraint on the impact of state programs to promote renewable energy or reduce *CO2* emissions. The coefficients on the lagged dependent variable are both highly significant in columns (3) and (5) of Table 5. The within panel time series are too short to carry out reliable tests, but the coefficient on the lagged dependent variable is so far from 1 that it is unlikely that the equations have a unit root. The λ coefficient in the SEM, column (6), is highly significant but this specification is dominated by the SAC model in column (7). The inclusion of the spatially lagged dependent variables reduces the estimate of λ from 0.39 to about 0.02 so that it is no longer significantly different from zero. Overall, the results in Table 5 together with the test reported in Table 6 suggest that the dynamic SDM (column 5) provides the best specification.

One of the reasons for studying such models is to estimate the price elasticities of demand. In the non-spatial specification the elasticity is simply the coefficient of the log price. As discussed in Section 2, the marginal effect of price on electricity demand may differ across states as a result of spatial interactions. The key difference between the direct and total impacts is that the direct impact measures the impact of a unit change in variable x_k in state i on demand in state i averaged over all states. In contrast, the total impact measures the impact of the same unit change in variable x_k in all states on demand in state i , again averaged over all states. `xsmle` displays values for the direct, indirect and total impact of changes in each of the independent variables. Differently from the values reported in Table 5, Table 7 reports elasticities taking into account spatial feedbacks. Moreover, for the SAR and SDM dynamic specifications, Table 7 also distinguishes between short- and long-run marginal effects. It is worth noting that marginal effects in static models have been labeled as long-run but they should be compared with short-run effects from dynamic models (see Table 2). These additional results are consistent across all spatial specifications, with the controls being significant and with the expected signs. It should be noted that the inclusion of the time-lagged dependent variable makes the coefficient for the real personal income not significant anymore and greatly reduces the elasticity of residential consumption with respect to the other controls.

5 Conclusions

In this paper we describe the new Stata command `xsmle`, which can be used to estimate an extensive array of spatial models for panel data. Of special note is that the command supports weight matrices in the form of both Stata matrices and `spmat` objects, allows the computation of direct, indirect and total effects and related standard errors, and provides various post-estimation features for obtaining predictions, including the use of `margins`. Furthermore, `xsmle` is fully compatible with the `mi` Stata suite of commands. We use simulated data to illustrate `xsmle` estimation capabilities focusing on model selection, prediction and estimation in presence of missing data, and provide an empirical application based on electricity usage data at state level in the United States.

Acknowledgement

We would like to thank, in particular, Paul Elhorst and Michael Pfaffermayr for giving permission to make use of their Matlab code. For most routines, we have modified or extended the way in which the routines operate, so they should not be held responsible for any errors that may be in our code.

6 References

- Angrist, J., and J.-S. Pischke. 2009. *Mostly Harmless Econometrics*. Princeton, NJ: Princeton University Press.
- Bramoullé, Y., H. Djebbari, and B. Fortin. 2009. Identification of peer effects through social networks. *Journal of Econometrics* 150(1): 41–55.
- Burnham, K. P. 2004. Multimodel Inference: Understanding AIC and BIC in Model Selection. *Sociological Methods & Research* 33(2): 261–304.
- Driscoll, J., and A. Kraay. 1998. Consistent Covariance Matrix Estimation with Spatially Dependent Panel Data. *Review of Economics and Statistics* 80: 549–560.
- Drukker, D. M., H. Peng, I. Prucha, and R. Raciborski. 2013a. Creating and managing spatial-weighting matrices with the `spmat` command. *Stata Journal* 13(2): 242–286.
- Drukker, D. M., I. Prucha, and R. Raciborski. 2013b. A command for estimating spatial-autoregressive models with spatial-autoregressive disturbances and additional endogenous variables. *Stata Journal* 13(2): 287–301.
- . 2013c. Maximum likelihood and generalized spatial two-stage least-squares estimators for a spatial-autoregressive model with spatial-autoregressive disturbances. *Stata Journal* 13(2): 221–241.
- Elhorst, J. P. 2009. *Handbook of Applied Spatial Analysis*, chap. Spatial panel data models. Berlin: Springer.

- . 2010. Applied spatial econometrics: raising the bar. *Spatial Economic Analysis* 5(1): 9–28.
- . 2014. *Spatial Econometrics: From Cross-Sectional Data to Spatial Panels*. SpringerBriefs, Heidelberg New York Dordrecht London: Springer.
- Elhorst, J. P., and S. Freret. 2009. Evidence of political yardstick competition in France using a two-regime spatial Durbin model. *Journal of Regional Science* 49(5): 931–951.
- Elhorst, P., G. Piras, and G. Arbia. 2010. Growth and Convergence in a Multiregional Model with Space-Time Dynamics. *Geographical Analysis* 42(3): 338–355.
- Hausman, J. A. 1978. Specification Tests in Econometrics. *Econometrica* 46(6): 1251–1271.
- Hoechle, D. 2007. Robust standard errors for panel regression with cross-section dependence. *The Stata Journal* 7: 281–312.
- Jeanty, P. W. 2010. SPWMATRIX: Stata module to generate, import, and export spatial weights. Statistical Software Components, Boston College Department of Economics. <https://ideas.repec.org/c/boc/bocode/s457111.html>.
- Kapoor, M., H. Kelejian, and I. Prucha. 2007. Panel data models with spatially correlated error components. *Journal of Econometrics* 140: 97–130.
- Kelejian, H. H., and I. R. Prucha. 2007. The relative efficiencies of various predictors in spatial econometric models containing spatial lags. *Regional Science and Urban Economics* 37(3): 363–374.
- Kostov, P. 2009. A Spatial Quantile Regression Hedonic Model of Agricultural Land Prices. *Spatial Economic Analysis* 4(1): 53–72.
- Lee, L. F., and J. Yu. 2010. Estimation of spatial autoregressive panel data models with fixed effects. *Journal of Econometrics* 154(2): 165–185.
- LeSage, J. P., and R. K. Pace. 2009. *Introduction to Spatial Econometrics*. Taylor & Francis.
- Millo, G., and G. Piras. 2012. splm: Spatial Panel Data Models in R. *Journal of Statistical Software* 47(1).
- Moscone, F., and M. Knapp. 2005. Exploring the spatial pattern of mental health expenditure. *Journal of mental health policy and economics* 8: 205–217.
- Moscone, F., E. Tosetti, and G. Vittadini. 2012. Social interaction in patients' hospital choice: evidence from Italy. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 175: 453–472.
- Parent, O., and J. P. LeSage. 2012. Spatial dynamic panel data models with random effects. *Regional Science and Urban Economics* 42(4): 727 – 738. <http://www.sciencedirect.com/science/article/pii/S0166046212000361>.

- Pisati, M. 2010. Tools for spatial data analysis. *Stata Technical Bulletin* 60.
- Revelli, F. 2005. On spatial public finance empirics. *International Tax and Public Finance* 12(4): 475–492.
- Rubin, D. 1987. *Multiple Imputation for Nonresponse in Surveys*. New York, NY: Wiley.
- Sollé Ollé, A. 2003. Electoral accountability and tax mimicking: the effects of electoral margins, coalition government, and ideology. *European Journal of Political Economy* 19(4): 685–713.
- . 2005. Expenditure spillovers and fiscal interactions: Empirical evidence from local governments in Spain. *Working Papers - Institut d’Economia de Barcelona* 3.
- Tobler, W. 1970. A Computer Movie Simulating Urban Growth in the Detroit Region. *Economic Geography* 46(2): 234–240.
- Yu, J., R. de Jong, and L. F. Lee. 2008. Quasi-maximum likelihood estimators for spatial dynamic panel data with fixed effects when both N and T are large. *Journal of Econometrics* 146: 118–134.

About the author

Federico Belotti is Assistant Professor of Econometrics at the Department of the Economics and Finance of the University of Rome Tor Vergata and a fellow of the Centre for Economics and International Studies (CEIS, University of Rome Tor Vergata).

Gordon Hughes is Professor of Economics at the School of Economics of the University of Edinburgh.

Andrea Piano Mortari is a researcher at the National Research Council, Institute for Research on Population and Social Policies (IRPPS), and a fellow of the Centre for Economics and International Studies (CEIS, University of Rome Tor Vergata).

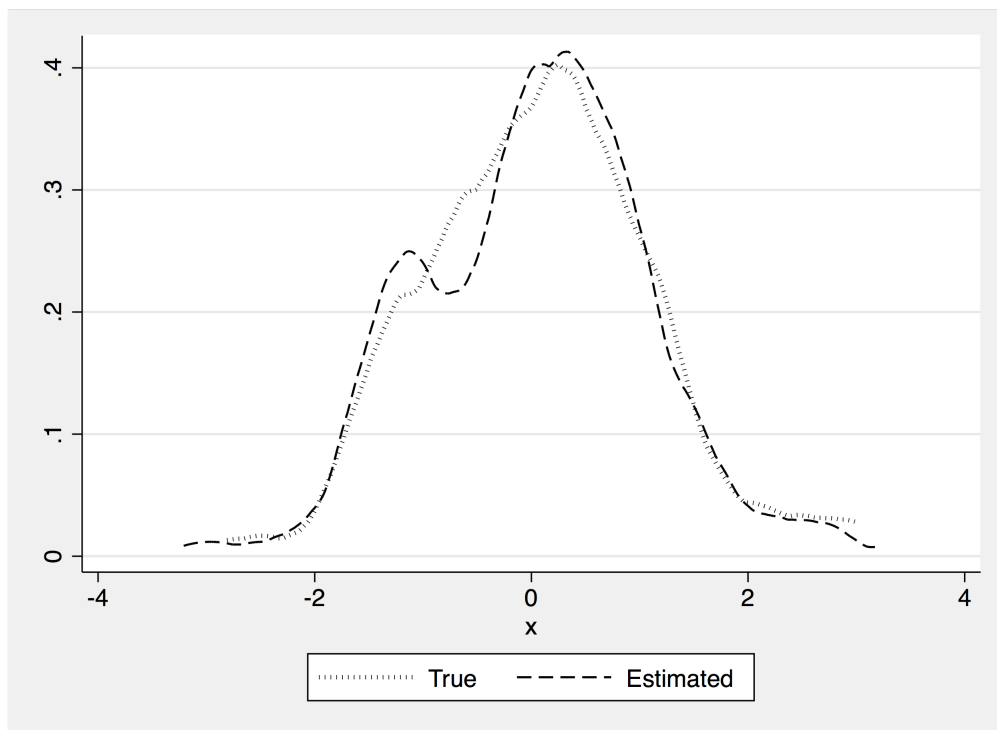
Figure 1: `xsmle` post-estimation: predicted fixed-effects

Figure 2: `xsmle` post-estimation: reduced form and *naive* predictors

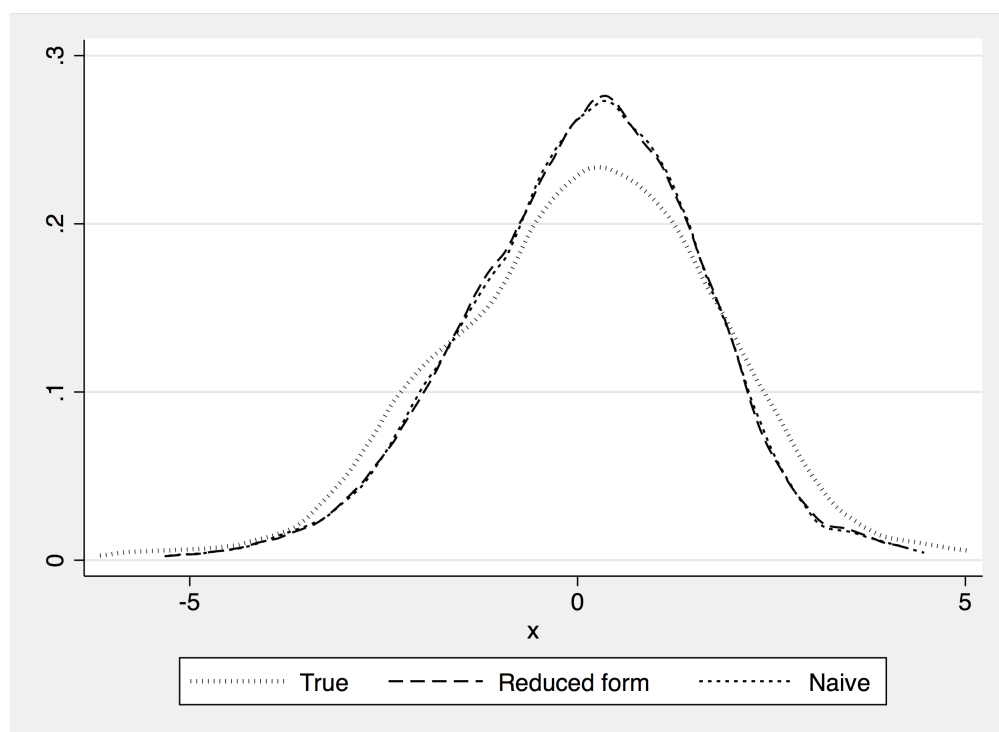


Table 1: A summary of `xsmle` estimation capabilities.

Model	Est _i method	Time Fixed Effects	Individual Fixed Effects	Random Effects	Wmatrix	Ematrix	Dmatrix	Dynamic
SAR	QML	x	x	x	x			x
SEM	QML	x	x	x	x	x		
SAC	QML	x	x	x	x	x		
SDM	QML	x		x	x	x	x	
GSPRE	QML			x	x	x		

Table 2: Direct, Indirect and Total Effects

Type of model	Short-term direct effect	Short-term indirect effect	Static	Long-term direct effect	Long-term indirect effect
SDM	none	none	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{rsum}}$
SEM	none	none	β_k	none	none
SAR	none	none	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{rsum}}$
SAC	none	none	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{rsum}}$
Dynamic					
SDM	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{rsum}}$	$[(1 - \tau) \mathbf{I} - (\rho + \psi) \mathbf{W}]^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{d}}$	$[(1 - \tau) \mathbf{I} - (\rho + \psi) \mathbf{W}]^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{d}}$	$[(1 - \tau) \mathbf{I} - (\rho + \psi) \mathbf{W}]^{-1} \times (\beta_k \mathbf{I} + \theta_k \mathbf{W})]^{\bar{rsum}}$
SAR	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{d}}$	$[(\mathbf{I} - \rho \mathbf{W})^{-1} \times (\beta_k \mathbf{I})]^{\bar{rsum}}$	$[(1 - \tau) \mathbf{I} - (\rho + \psi) \mathbf{W}]^{-1} \times (\beta_k \mathbf{I})^{\bar{d}}$	$[(1 - \tau) \mathbf{I} - (\rho + \psi) \mathbf{W}]^{-1} \times (\beta_k \mathbf{I})^{\bar{d}}$	$[(1 - \tau) \mathbf{I} - (\rho + \psi) \mathbf{W}]^{-1} \times (\beta_k \mathbf{I})^{\bar{rsum}}$

Source: Adapted from Elhorst (2014). Note: The superscript \bar{d} denotes the operator that calculates the mean diagonal element of a matrix and the superscript \bar{rsum} denotes the operator that calculates the mean row sum of the non-diagonal elements.

Table 3: Simulation results (MSEs)

ρ	λ	naive	limited	full	$\hat{\alpha}$
-0.9	-0.9	1.441	0.488	0.308	0.398
-0.9	-0.4	0.931	0.547	0.464	0.263
-0.9	0	0.811	0.678	0.659	0.221
-0.9	0.4	0.940	0.932	0.913	0.244
-0.9	0.9	6.384	1.331	1.155	1.594
-0.4	-0.9	1.348	0.547	0.465	0.359
-0.4	-0.4	0.902	0.649	0.625	0.246
-0.4	0	0.809	0.764	0.761	0.224
-0.4	0.4	0.937	0.863	0.857	0.252
-0.4	0.9	6.362	0.856	0.849	1.586
0	-0.9	1.340	0.678	0.659	0.346
0	-0.4	0.887	0.764	0.761	0.240
0	0	0.803	0.793	0.792	0.220
0	0.4	0.937	0.765	0.762	0.257
0	0.9	6.019	0.702	0.660	1.548
0.4	-0.9	1.337	0.931	0.913	0.341
0.4	-0.4	0.885	0.863	0.857	0.233
0.4	0	0.811	0.765	0.762	0.229
0.4	0.4	0.966	0.673	0.651	0.274
0.4	0.9	5.541	0.659	0.518	1.445
0.9	-0.9	1.340	1.334	1.156	0.339
0.9	-0.4	0.887	0.857	0.849	0.230
0.9	0	0.805	0.702	0.660	0.215
0.9	0.4	1.445	0.659	0.518	0.758
0.9	0.9	8.150	1.127	0.391	2.250

Table 4: Summary of estimation results by % of missing values[†]

Missing x1				
	No missing	5% missing	10% missing	20% missing
β_1	0.546	0.510	0.471	0.425
	(0.034)	(0.037)	(0.040)	(0.043)
Missing y				
	No missing	5% missing	10% missing	20% missing
ρ	0.227	0.192	0.171	0.103
	(0.043)	(0.047)	(0.053)	(0.060)

[†] Standard Errors in parentheses. True values: $\beta_1 = 0.5$, $\rho = 0.3$.

Table 5: Fixed-effects models for residential electricity demand

	Fixed-effects		SAR	dynamic SAR	SDM	dynamic SDM	SEM	SAC
Real personal income	0.391***	0.203***	0.033	0.212***	0.039	0.375***	0.207***	
Real average residential price	-0.235***	-0.239***	-0.144***	-0.293***	-0.165***	-0.271***	-0.241***	
Housing units	1.019***	0.747***	0.150*	0.629***	0.128*	0.818***	0.748***	
Cooling degree days	0.075***	0.057***	0.073***	0.057***	0.072***	0.071***	0.057***	
Heating degree days	0.188***	0.140***	0.146***	0.131***	0.140***	0.156***	0.141***	
L.Residential electricity consumption			0.554***		0.534***			
ρ			0.367***	0.261***	0.416***	0.284***	0.390***	0.359***
λ								0.018
Real average total price					0.165***	0.057**		
Log-lik	2029.91	2108.54	2315.67	2144.51	2333.58	2071.08	2108.58	
Obs	1078	1078	1029	1078	1029	1078	1078	
R^2	0.82	0.82	0.89	0.85	0.89	0.82	0.82	
R_{adj}^2	0.12	0.14	0.91	0.16	0.91	0.21	0.14	
R_b^2	0.18	0.20	0.89	0.22	0.90	0.26	0.20	
Hausman χ^2		28.24		27.33		33.19		
Hausman p -value		0.00		0.00		0.00		

Table 6: Residential electricity demand - Test for model selection

	χ^2	P-value	AIC
SAR vs dynamic SAR	414.26	0.000	.
SDM vs dynamic SDM	358.14	0.000	.
dynamic SAR vs dynamic SDM	15.82	0.000	.
SEM vs dynamic SDM	505.00	0.000	.
SAC	.	.	-4201.0
dynamic SDM	.	.	-4629.0

Table 7: Direct, indirect and total effects - residential electricity demand

	SAR	dynamic SAR	SDM	dynamic SDM	SAC
Long-run direct Effects					
Real personal income	0.210***	0.082	0.223***	0.096	0.214***
Real average residential price	-0.247***	-0.359***	-0.307***	-0.401***	-0.249***
Housing units	0.775***	0.375**	0.660***	0.312*	0.775***
Cooling degree days	0.059***	0.182***	0.059***	0.174***	0.059***
Heating degree days	0.145***	0.366***	0.138***	0.339***	0.146***
Real average total price			0.020***	0.026*	
Long-run indirect Effects					
Real personal income	0.110***	0.095	0.141***	0.121*	0.109***
Real average residential price	-0.130***	-0.417**	-0.194***	-0.508**	-0.126**
Housing units	0.406***	0.435*	0.417***	0.394	0.393*
Cooling degree days	0.031***	0.211***	0.038***	0.221***	0.030*
Heating degree days	0.076***	0.424**	0.087***	0.429**	0.074*
Real average total price			0.262***	0.285*	
Long-run total Effects					
Real personal income	0.321***	0.177	0.364***	0.217	0.323***
Real average residential price	-0.377***	-0.777***	-0.502***	-0.909***	-0.375***
Housing units	1.180***	0.809**	1.077***	0.706*	1.168***
Cooling degree days	0.090***	0.393***	0.097***	0.395***	0.089***
Heating degree days	0.222***	0.790***	0.225***	0.768***	0.220***
Real average total price			0.282***	0.311*	
Short-run direct Effects					
Real personal income		0.033		0.040	
Real average residential price		-0.146***		-0.169***	
Housing units		0.152*		0.131*	
Cooling degree days		0.074***		0.073***	
Heating degree days		0.149***		0.142***	
Real average total price				0.004*	
Short-run indirect Effects					
Real personal income		0.011		0.015	
Real average residential price		-0.048***		-0.062***	
Housing units		0.050*		0.048	
Cooling degree days		0.024***		0.027***	
Heating degree days		0.049***		0.052***	
Real average total price				0.075*	
Short-run total Effects					
Real personal income		0.044		0.055	
Real average residential price		-0.194***		-0.231***	
Housing units		0.203*		0.179	
Cooling degree days		0.098***		0.100***	
Heating degree days		0.198***		0.195***	
Real average total price				0.079*	