The Stata Journal (2022)
**22**, Number 2, pp. 293–318 DOI: 10.1177/1536867X221106373

# Fitting spatial autoregressive logit and probit models using Stata: The spatbinary command

Daniele Spinelli
Department of Statistics and Quantitative Methods
University of Milan–Bicocca
Milan, Italy
daniele.spinelli@unimib.it

**Abstract.** Starting from version 15, Stata allows users to manage data and fit regressions accounting for spatial relationships through the `sp` commands. Spatial regressions can be estimated using the `spregress`, `spxtregress`, and `spivregress` commands. These commands allow users to fit spatial autoregressive models in cross-sectional and panel data. However, they are designed to estimate regressions with continuous dependent variables. Although binary spatial regressions are important in applied econometrics, they cannot be estimated in Stata. Therefore, I introduce `spatbinary`, a Stata command that allows users to fit spatial logit and probit models.

**Keywords:** st0672, spatbinary, spatbinary_impact, postestimation, spatial logit, spatial probit, spatial autoregressive models, marginal effects

## 1 Introduction

In recent decades, spatial econometrics has been a growing field of study. In Stata software packages, spatial data were originally managed by a plethora of community-contributed commands. Pisati (2001) and Drukker et al. (2013) released `spatwmat` and `spmat`, respectively, for managing spatial matrices. Other relevant contributions include software to perform spatial correlation tests (the `spatcorr` command of Pisati [2001]), geocode data (Ozimek and Miles 2011), calculate travel time (Huber and Rust 2016; Weber and Péclat 2017), and visualize detailed maps (Pisati 2018). Other than these utilities, community-contributed packages address spatial regression models in terms of cross-sectional data (Pisati 2001), panel data (Belotti, Hughes, and Mortari 2017), and endogenous regressors (Drukker, Prucha, and Raciborski 2013).

In addition, Stata 15 introduced `sp`, a suite of commands that allows users to manage spatial data and fit regressions with underlying spatial relationships. Some of the above-mentioned community-contributed packages were embedded in `sp`.

As of Stata 17, official commands for spatial regression models (that is, `spregress`, `spxtregress`, and `spivregress`) are designed to fit models with continuous dependent variables. For models with binary responses, it is only possible to use `spregress` to fit spatial linear probability models. These models do not constrain the predicted values in $[0, 1]$ (Wooldridge 2010). Conversely, spatial probit or logit models can be fit only by using other software.[1]

In applied economics, spatial models for binary outcomes have many uses. For instance, Klier and McMillen (2008) analyzed clustering in the automotive sector; Flores-Lagunes and Schnier (2012) used them to solve sample selection issues; Brasington, Flores-Lagunes, and Guci (2016) analyzed the adoption of open enrollment in schools; and more recently Prato et al. (2018) investigated the role of urban environment in pedestrian accidents.

The aim of this article is therefore to address the lack of Stata packages that are specifically aimed at fitting spatial models with binary response variables. Thus, I introduce `spatbinary`, a command that allows the estimation of logit and probit spatial autoregressive models. The `spatbinary` command is equivalent to `spregress` (with the option `dvarlag`) for binary dependent variables and to `logit` and `probit` for spatial data. The command is based on the generalized method of moments (GMM) estimators outlined in Pinkse and Slade (1998) and their approximations (Klier and McMillen 2008). The remainder of this article is organized as follows: the next section outlines the econometric setting for `spatbinary`, sections 3 and 4 present the syntax, section 5 presents some examples of the use of `spatbinary`, and section 6 concludes.

## 2 The econometric setting

Consider a binary outcome model in which the observed binary response $y_i = 1$ only if the unobserved latent variable $u_i \in \boldsymbol{U}$ is greater than 0; otherwise, $y_i = 0$. For instance, in the random utility setting (McFadden 1974), the latent variable represents the utility of a consumer facing the decision of whether to purchase a product or a service. If the underlying latent variable depends on a set of independent variables $\boldsymbol{X}$ and is also spatially autocorrelated, the binary spatial autoregressive (BSAR) model assumes the form

$$\boldsymbol{U} = \rho \boldsymbol{W} \boldsymbol{U} + \boldsymbol{X} \boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{1}$$

where $\boldsymbol{W}$ is a row-standardized contiguity matrix, $\boldsymbol{\epsilon}$ represents the error term, and parameters $\boldsymbol{\beta}$ and $\rho$ are to be estimated. Spatial autocorrelation of the (latent) dependent variable $\boldsymbol{U}$ through parameter $\rho$ implies a form of endogeneity because $\boldsymbol{U}$ appears in (1) both as a dependent variable and as a covariate (Arbia 2014). Specifically, this is a form of simultaneity that "arises when at least one of the explanatory variables is determined simultaneously along with the dependent variable" (Wooldridge 2010). In (1), it is straightforward to note that $\boldsymbol{U}$ is determined simultaneously with itself because it is the same variable.

Because $\boldsymbol{U}$ is unobserved, the spatial autocorrelation parameter $\rho$ implies that the *propensity* of having a positive outcome is correlated with the *propensity* to have a positive outcome in nearby units of observation (Klier and McMillen 2008). A positive

1. For instance, R users can rely on packages such as `McSpatial` and `spatialprobit` (Wilhelm and de Matos 2013).

$\rho$ implies clustering: units with a high probability of a positive outcome are located close to other peers characterized by a high probability that $y_i = 1$. Conversely, $\rho < 0$ means that the propensity is dispersed in space.

Depending on the distribution of the error term, (1) can be estimated through a logit or a probit model characterized by "transformed" independent variables $\boldsymbol{X}^{**}$, predicted probabilities $P(y_i = 1)$, and generalized residuals $e_i$ (table 1). The error-term variance is proportional to $\boldsymbol{V} = E(\boldsymbol{\epsilon}'\boldsymbol{\epsilon}) = [(\boldsymbol{I} - \rho\boldsymbol{W})'(\boldsymbol{I} - \rho\boldsymbol{W})]^{-1}$ (Calabrese and Elkink 2014; McMillen 1992), and $\boldsymbol{D}$ contains the square root of the elements of $\boldsymbol{V}$ on its main diagonal.

Table 1. Relevant quantities for logit and probit BSAR models

| | Probit | Logit |
|---|---|---|
| $\boldsymbol{D}$ | $\sqrt{\text{diag}(\boldsymbol{V})}$ | |
| $\boldsymbol{X}^{**}$ | $\boldsymbol{D}^{-1}(\boldsymbol{I} - \rho\boldsymbol{W})^{-1}\boldsymbol{X}$ | |
| $P(y_i = 1)$ | $\Phi(\boldsymbol{X}_i^{**}\boldsymbol{\beta})$ | $\dfrac{\exp(\boldsymbol{X}_i^{**}\boldsymbol{\beta})}{1 + \exp(\boldsymbol{X}_i^{**}\boldsymbol{\beta})}$ |
| $e_i$ | $\dfrac{\{y_i - P(y_i)\}\phi(\boldsymbol{X}_i^{**}\boldsymbol{\beta})}{P(y_i) * \{1 - P(y_i)\}}$ | $y_i - P(y_i)$ |

To estimate BSAR parameters $\boldsymbol{\beta}$ and $\rho$, Pinkse and Slade (1998) proposed a GMM-type estimator (Hansen 1982) where coefficient estimates are chosen to minimize the quantity in (2).

$$Q = N^{-1}\{\boldsymbol{e}(\boldsymbol{\beta}, \rho)'\boldsymbol{Z}\boldsymbol{M}\boldsymbol{Z}'\boldsymbol{e}(\boldsymbol{\beta}, \rho)\} \tag{2}$$

Such a model requires a set of instruments $\boldsymbol{Z}$ that may include the independent variables $\boldsymbol{X}$ and their spatial lags $\boldsymbol{W}\boldsymbol{X}, \ldots, \boldsymbol{W}^n\boldsymbol{X}$ (Kelejian and Prucha 1998). Klier and McMillen (2008) noted that the estimator in (2) reduces to nonlinear two-stage least squares (N2SLS) if $\boldsymbol{M} = (\boldsymbol{Z}'\boldsymbol{Z})^{-1}$ and proposed a simplified version of the N2SLS model by introducing a linear approximation around a convenient starting point. The `spatbinary` command is designed to fit both models.

## 2.1 N2SLS estimator

The N2SLS procedure for estimating model parameters $\boldsymbol{\Theta}$ is the following:

1. Assume initial values $\boldsymbol{\Theta}_0$.
2. Repeat until convergence[2] or until the maximum number of iterations is reached:
   a. Calculate the guess of the generalized residuals at iteration $t$ ($\boldsymbol{e}_t$).
   b. Obtain the fitted values $\widehat{\boldsymbol{G}}_t$ from a linear regression of the gradient terms $\boldsymbol{G}_t = (\partial \boldsymbol{e}_t)/(\partial \boldsymbol{\Theta}_t)$ on the instruments $\boldsymbol{Z}$.
   c. Update $\boldsymbol{\Theta}_{t+1} = \boldsymbol{\Theta}_t + [\widehat{\boldsymbol{G}}_t' \widehat{\boldsymbol{G}}_t]^{-1} \widehat{\boldsymbol{G}}_t' \boldsymbol{e}_t$.
3. The robust variance–covariance estimator is given by the following matrix: $[\widehat{\boldsymbol{G}}' \widehat{\boldsymbol{G}}]^{-1} \text{diag}(\widehat{\boldsymbol{G}}' \boldsymbol{e}\boldsymbol{e}' \widehat{\boldsymbol{G}}) [\widehat{\boldsymbol{G}}' \widehat{\boldsymbol{G}}]^{-1}$.

## 2.2 Linearized estimator

The linearized estimator procedure is the following:

1. Through a nonspatial logit or probit model, obtain initial estimates of $\boldsymbol{\beta}_0$ and $\boldsymbol{e}_0$ (that is, assume $\rho = 0$).
2. Calculate the gradient $\boldsymbol{G} = (\partial \boldsymbol{e})/(\partial \boldsymbol{\Theta})$. These gradients are much simpler than in the N2SLS case.
3. Regress $\boldsymbol{G}$ on $\boldsymbol{Z}$ and obtain the predicted gradient estimates $\widehat{\boldsymbol{G}}$.
4. Calculate $\boldsymbol{e}_0 + \boldsymbol{G}_\beta' \boldsymbol{\beta}_0$ and regress it on $\widehat{\boldsymbol{G}}$. The obtained coefficients are the desired estimates of $\boldsymbol{\beta}$ and $\rho$.

## 2.3 Using the estimators in practice

The main advantage of the linearized model is computational. Indeed, it avoids the inversion of the matrix $\boldsymbol{I} - \rho \boldsymbol{W}$ during each iteration. However, this advantage becomes less pronounced if the matrix $\boldsymbol{W}$ is small or sparse (that is, with many zero entries).

Furthermore, Klier and McMillen (2008) show that the linearized model provides a good approximation of the spatial autocorrelation parameter $\rho$ only if the absolute value of the data-generating autocorrelation parameter lies in the interval 0.1–0.5. Outside this interval, estimates from the linearized model are upwardly biased (Arbia 2014). Moreover, assuming that the model is correctly specified, the standard errors produced by the linearized model are larger than those obtained by N2SLS; thus, the linearized BSAR is less efficient (Klier and McMillen 2008; Billé 2013). To summarize, the practical situations for which the linearized model can be used are the following:

2. Convergence is achieved if $(|Q_t - Q_{t-1}|)/(Q_{t-1} + 1e - 3) < \text{tol}$. In `spatbinary`, the convergence tolerance (tol) is controlled by the option `tolerance()`.

- $\boldsymbol{W}$ is large.
- $\boldsymbol{W}$ is dense.
- The data have a weak spatial structure.

Otherwise, the N2SLS approach is preferable, and coefficients from the linearized model can be used as starting values for the N2SLS model. Neither model guarantees that the confidence intervals for $\rho$ are bounded in the $[-1, 1]$ interval (Billé 2013).

## 2.4 Measures of impact and marginal effects

As in standard probit and logit models, the coefficients $\boldsymbol{\beta}$ represent the impact of the independent variables $\boldsymbol{X}$ on the latent variable $U$. Such estimates are difficult to interpret because $U$ is unobserved; however, they provide useful information about the direction of the impact of $\boldsymbol{X}$ on the probability that the outcome $y$ is observed through their sign. To overcome this difficulty, a common strategy is to report measures of impact using `margins` (Williams 2012). In the general spatial autoregressive framework, a variation of an explanatory variable in a certain geographical unit affects the response variable both in the same location and in other locations because of spatial dependence (Anselin, Florax, and Rey 2004; Arbia 2014). Thus, measures of impact are split into direct and indirect components. The former measures a unit's predicted contribution to its own probability of a positive outcome, and the latter measures the predicted impact of the other units' contributions to a unit's probability. In Stata, four types of measures of marginal impact are generally reported:

- the marginal effect (ME), which represents the variation of the dependent variable in response to a unit variation in an explanatory variable (in `margins`, this is calculated using the `dydx` option);
- the elasticity, which represents the percent variation in the response variable in relation to a 1% variation of an explanatory variable (`eyex` option in `margins`);
- the semielasticity representing the percent variation in the response variable in relation to a unit variation of an explanatory variable (`eydx` option in `margins`); and
- a second type of semielasticity representing the variation in the dependent variable in natural units in relation to a 1% variation of an explanatory variable (`dyex` option in `margins`).

In the BSAR framework, the response variable is $P_i = P(y_i = 1)$. Table 2 summarizes the measures of impact related to the $i$th observation of covariate $\boldsymbol{X}^k$. In table 2, $g_i$ is the first derivative of $P_i$ with respect to the $i$th row of $X_i^k \beta_{X_i^k}$. Such a derivative could be direct (that is, measuring a unit $i$'s predicted contribution to $P_i$) or indirect (that is, measuring the other units' contributions to the variation of $P_i$).[3] In the `spatbinary`

3. For a thorough analytic formulation of $g_i$, see Beron and Vijverberg (2004) and Billé and Leorato (2020).

framework, the measures of impact in table 2 can be obtained using the postestimation command `spatbinary_impact`. Users can also use `margins` by exploiting the calculation of $g_i$ using `predict` along with options `directmargin`, `indirectmargin`, and `totalmargin`.

Table 2. Relevant measures of impact for BSAR models

| Measure | Expression |
|---|---|
| `dydx` | $g_i\beta_{X^k}$ |
| `eyex` | $g_iX_i^k\beta_{X^k}[P_i]^{-1}$ |
| `eydx` | $g_i\beta_{X^k}[P_i]^{-1}$ |
| `dyex` | $g_iX_i^k\beta_{X^k}$ |

# 3 The spatbinary command

## 3.1 Syntax

Data should be `spset` before using `spatbinary`. The syntax of `spatbinary` is

`spatbinary` *depvar indepvars* [*if*] [*in*] [*weight*], `wmat(`*matname*`)` [`linearized` `n2sls` `logit` `probit` `noconstant` `force` `instr(`*varlist*`)` `winstr(`*varlist*`)` `impower(`#`)` `noinstrconstant` `iterate(`#`)` `start(`*matname*`)` `tolerance(`#`)` *display_options* `level(`#`)` `coeflegend`]

`fweight`s are allowed; see [U] **11.1.6 weight**.

## 3.2 Options

### Main options

`wmat(`*matname*`)` specifies that the spatial weight matrix $\boldsymbol{W}$ is stored in matrix *matname*. This option is required and is equivalent to `dvarlag` in `spregress`. The weight matrix must be created using `spmatrix`.

`linearized` requests that the linearized estimator be used. This is the default.

`n2sls` requests the N2SLS estimator be used. Specifying both `n2sls` and `linearized` will cause `spatbinary` to ignore `linearized`.

`logit` requests to fit the logit BSAR model. This is the default setting. If both `probit` and `logit` are specified, the program returns an error message and stops the execution of `spatbinary`.

`probit` requests to fit the probit BSAR model.

`noconstant` suppresses the constant term in the model; see [R] **Estimation options**.

`force` requests that estimation be done when the estimation sample is a proper subset of the sample used to create the spatial weighting matrices. The default is to refuse to fit the model. Weighting matrices potentially connect all the spatial units. When the estimation sample is a subset of this space, the spatial connections differ, and spillover effects can be altered. In addition, the normalization of the weighting matrix differs from what it would have been had the matrix been normalized over the estimation sample. The better alternative to `force` is to first understand the spatial space of the estimation sample and then, if it is sensible, create new weighting matrices for it. See [SP] **spmatrix** and *Missing values, dropped observations, and the W matrix* in [SP] **Intro 2**.

### Instruments options

The following options control the instruments specification. The number of instruments must be equal to or greater than the number of parameters to be estimated. Otherwise, `spatbinary` stops its execution.

`instr(`*varlist*`)` specifies a list of instruments. By default, the independent variables are included.

`winstr(`*varlist*`)` specifies a list of instruments to be spatially lagged using `wmat(`*matname*`)`. By default, the spatial lags of the independent variables are included. The option may be specified with `instr(`*varlist*`)`.

`impower(`#`)` specifies the order of an instrumental-variable approximation used in fitting the model. The derivation of the estimator involves a product of # matrices. Increasing # may improve the precision of the estimation and will not cause harm but will require more computer time. The default is `impower(1)`.

`noinstrconstant` omits the intercept from the instruments.

### Options for the N2SLS model

These options affect the estimation only if `n2sls` is specified.

`iterate(`#`)` specifies the number of iterations for N2SLS estimation.

`start(`*matname*`)` specifies that the starting values for the N2SLS estimation be stored in vector *matname*. The default starting values are estimated using the linearized approach.

`tolerance(`#`)` specifies the tolerance for the GMM criterion. The default is `tolerance(1e-5)`.

### Reporting

*display_options*: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(#), fvwrapon(*style*), cformat(%*fmt*), pformat(%*fmt*), sformat(%*fmt*), and nolstretch; see [R] **Estimation options**.

level(#) specifies the confidence level, as a percentage, for confidence intervals. The default is level(95) or as set by set level; see [U] **20.8 Specifying the width of confidence intervals**.

coeflegend displays the legend instead of statistics.

## 3.3 Stored results

spatbinary stores the following in e():

| | |
|---|---|
| Scalars | |
| e(N) | number of observations |
| e(level) | confidence level |
| e(impower) | order of the instrumental-variable approximation used to fit the model |
| e(iterations) | number of GMM iterations; applies only to n2sls |
| e(converged) | convergence indicator; applies only to n2sls |
| e(J) | Hansen's $J$ |
| e(J_df) | Hansen's $J$ degrees of freedom |
| e(Q) | GMM objective function |
| Macros | |
| e(cmd) | spatbinary |
| e(depvar) | dependent variable |
| e(wtype) | weight type |
| e(wexp) | weight expression |
| e(properties) | b V |
| e(predict) | program used to implement predict |
| e(w) | name of the spatial weight matrix |
| e(indepvars) | independent variables |
| e(logit) | logit if a logit model is fit |
| e(probit) | probit if a probit model is fit |
| e(estimator) | returns n2sls; applies only to n2sls |
| Matrices | |
| e(b) | coefficient vector |
| e(V) | variance–covariance matrix of the estimators |
| Functions | |
| e(sample) | marks estimation sample |

# 4 Postestimation

After spatbinary, predict and spatbinary_impact are available. The latter is a wrapper of margins and estimates measures of impact such as marginal effects, elasticities, and semielasticities; it corresponds with estat impact following spregress; see [SP] **spregress postestimation**.

## 4.1 Syntax for predict

The available predictions are the following:

`predict` *newvar*, `pr`, the default, calculates the probability of a positive outcome.

`predict` *newvar*, `xb` calculates $\boldsymbol{X}_i^{**}\boldsymbol{\beta}$.

`predict` *newvar*, `totalmargin` calculates the total marginal effect with respect to $\boldsymbol{X}_i^{**}\boldsymbol{\beta}$. It is used with `margins`.

`predict` *newvar*, `directmargin` calculates the direct marginal effect of $\boldsymbol{X}_i^{**}\boldsymbol{\beta}$ ($g_i$ in section 2.4). It is used with `margins`.

`predict` *newvar*, `indirectmargin` calculates the indirect marginal effect of $\boldsymbol{X}_i^{**}\boldsymbol{\beta}$. It is used with `margins`.

`predict` *newvar*, `residuals` calculates probit or logit generalized residuals.

## 4.2 Syntax for spatbinary_impact

`spatbinary_impact` *varlist* [ , [`dydx`|`eyex`|`dyex`|`eydx`] `total` `direct` `indirect` ]

Options `dydx`, `eyex`, `dyex`, and `eydx` are mutually exclusive. If `total`, `direct`, and `indirect` are all left unspecified, `spatbinary_impact` estimates them all.

### 4.2.1 Options

`dydx` calculates the marginal effect of *varlist* on the predicted probability (option `pr` in `predict`). `dydx` is the program default.

`eyex` calculates the elasticity of the predicted probability (option `pr` in `predict`) with respect to *varlist*.

`dyex` calculates the semielasticity ($dP/d\ln x$) of the predicted probability (option `pr` in `predict`) with respect to *varlist*.

`eydx` calculates the semielasticity ($d\ln P/dx$) of the predicted probability (option `pr` in `predict`) with respect to *varlist*.

`total` calculates the total measure of impact of *varlist* (defined by `dydx`, `eyex`, `dyex`, and `eydx`).

`direct` calculates the direct measure of impact. It captures own-unit contributions of *varlist* on a unit's prediction.

`indirect` calculates the indirect measure of impact. It captures the contributions of the other units' *varlist* on a unit's prediction.

# 5 Examples

## 5.1 Setup

I illustrate spatbinary with five examples using homicide1990.dta (Messner et al. 2000); this dataset also provides examples for Stata's official spatial regression command (spregress). Each observation in homicide1990 is a county in the southern United States.

Before we use spatbinary, the dataset must be declared to hold spatial data using spset, and a spatial weight matrix should be specified. To this end, homicide1990.dta comes with an ancillary shapefile storing the spatial information (coordinates and proximity) and is already spset. In the remainder of this section, the spatial weight matrix is created using spmatrix create and stored in W2, a row-standardized contiguity matrix. The examples include both probit and logit BSAR models for illustrative purposes. Like their nonspatial counterparts, they generally give similar predictions (Greene 2018); however, the logit estimator is less likely to estimate probability estimates near 0 or 1. The choice between them depends on the specific application and field of research. Because coefficients are not readily interpretable, measures of impact are calculated. Thus, measures of impact throughout this section are related to gini.

```
. webuse homicide1990
(S.Messner et al.(2000), U.S southern county homicide rates in 1990)
. copy https://www.stata-press.com/data/r17/homicide1990_shp.dta .
. spmatrix clear
. spmatrix create contiguity W2, normalize(row)
. spset
      Sp dataset: homicide1990.dta
Linked shapefile: homicide1990_shp.dta
            Data: Cross sectional
 Spatial-unit ID: _ID
     Coordinates: _CX, _CY (planar)
```

The first three examples are devoted to logit and probit BSARs with a dichotomized version of hrate, the county-level homicide rate per year per 100,000 persons, as the response variable. The independent variables are the Gini index, a measure of income inequality (gini), and the logarithm of the population (ln_population). Let us assume that the impact of income inequality on the probability of having a high homicide rate (also referred to as "the predicted probability" or "the probability that hrate_gt_p95=1") is the main research interest of the analysis.

```
. quietly summarize hrate, detail
. generate hrate_gt_p95=hrate>r(p95)
```

Examples 4 and 5 are based on simulated dependent variables Y1 and Y2. In both examples, the underlying latent variable is xb=-10+22*gini. Examples 4 and 5 are respectively based on $\rho = 0.7$ and $\rho = 0.2$.

```
. set seed 123456789
. spmatrix matafromsp W id=W2
. mata:
──────────────────────────────────────────────── mata (type end to exit) ──────
: x=st_data(.,"gini")
: xb=-10:+22*x
: xb1=qrsolve(I(1412)-.7*W,xb)
: xb2=qrsolve(I(1412)-.2*W,xb)
: end
───────────────────────────────────────────────────────────────────────────────
. capture drop xb* Y*
. getmata xb1=xb1 xb2=xb2
. generate Y1=rbinomial(1,invlogit(xb1))
. generate Y2=rbinomial(1,invlogit(xb2))
```

## 5.2 Example 1—Linearized models with program defaults

This example illustrates the estimation of the linearized BSAR using the default settings of `spatbinary`. In the first model, neither `probit` nor `logit` is specified; hence, a logit model is fit. The second model is a spatial probit. Further, because option `n2sls` is not specified, linearized probit and logit models are fit. Program defaults imply that the instruments are an intercept, the independent variables, and their spatial lag. This is equivalent to specifying options `instr(ln_population gini)`, `winstr(ln_population gini)`, and `impower(1)`.

```
. spatbinary hrate_gt_p95 ln_population gini, wmat(W2)
instruments set as (X,WX...W^n X) where X= ln_population gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
LINEARIZED LOGIT
```

| hrate_gt_p95 | Coefficient | Robust std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| hrate_gt_p95 | | | | | | |
| ln_population | .2403071 | .1620309 | 1.48 | 0.138 | -.0772677 | .5578819 |
| gini | 40.14493 | 6.682836 | 6.01 | 0.000 | 27.04681 | 53.24305 |
| _cons | -23.19683 | 4.486104 | -5.17 | 0.000 | -31.98943 | -14.40423 |
| rho | | | | | | |
| _cons | -.4276641 | .2635575 | -1.62 | 0.105 | -.9442273 | .0888992 |

```
Test of overidentifying restriction:
Hansen's J chi2(1) = .0783474, p = .7795495
```

```
. spatbinary hrate_gt_p95 ln_population gini, wmat(W2) probit
instruments set as (X,WX...W^n X) where X= ln_population gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
LINEARIZED PROBIT
```

| hrate_gt_p95 | Coefficient | Robust std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| hrate_gt_p95 | | | | | | |
| ln_population | .0845121 | .037548 | 2.25 | 0.024 | .0109194 | .1581047 |
| gini | 19.34454 | 2.247513 | 8.61 | 0.000 | 14.9395 | 23.74959 |
| _cons | -11.0464 | 1.28814 | -8.58 | 0.000 | -13.57111 | -8.521691 |
| rho | | | | | | |
| _cons | -.3749664 | .155578 | -2.41 | 0.016 | -.6798937 | -.0700391 |

```
Test of overidentifying restriction:
Hansen's J chi2(1) = .3138245, p = .5753428
```

The coefficient `rho` is negative in both models and is significant only in the probit specification. Therefore, the propensity of having high homicide rates is characterized by weak dispersion: the probability of having `hrate` $= 1$ in a county decreases if the same propensity is high in the neighboring counties. The coefficient attached to `gini` is positive and highly significant in both models; this means that the higher the income inequality, the higher the probability of having high homicide rates. The coefficient of `ln_population` is positive and significant only in the probit model, suggesting a weak correlation between population and the probability of having high homicide rates. The package also provides Hansen's test for overidentification, which applies when the number of instruments (moment conditions) is larger than the number of parameters to be estimated (see [R] **gmm postestimation**). In this example, there are four parameters to be estimated and five instruments. The test is not significant and safely suggests that the instruments are valid.

Because probit and logit are nonlinear models, their coefficients do not convey information about the magnitude of the relationship between the attached explanatory variable (`gini` and `ln_population`) and the probability that `hrate_gt_p95=1`. Moreover, the structure of the model [see (1) and section 2.4] implies spillovers through the existence of the spatial autocorrelation coefficient $\rho$. In this example, the implication is that the propensity of having high homicide rates is affected by own-county inequality (`gini`), own-county population (`ln_population`), and the propensity of having high homicide rates of neighboring counties, which in turn are affected by inequality and population. Thus, a county's propensity of having high homicide rates is indirectly influenced by inequality and population in the neighboring counties. This fact adds complexity to the interpretation of the coefficients attached to the explanatory variables. MEs are reported for the probit model because it exhibits a significant pattern of spatial autocorrelation ($\rho$ is significant). MEs for the logit model are also reported for appreciating the similarities between the two sets of ME estimates.

```
. quietly spatbinary hrate_gt_p95 ln_population gini, wmat(W2) probit
. spatbinary_impact gini, dydx
Impact measures for gini
```

| | dydx | Delta-M˽d std. err. | z | p>\|z\| | [95 conf. | interval] |
|---|---|---|---|---|---|---|
| gini | | | | | | |
| total | 1.310977 | .1250546 | 10.48324 | 1.03e-25 | 1.065875 | 1.556079 |
| direct | 1.843956 | .3229716 | 5.709343 | 1.13e-08 | 1.210943 | 2.476968 |
| indirect | -.5329786 | .2562527 | -2.079894 | .0375352 | -1.035225 | -.0307325 |

In the output above, all the MEs are significant. Furthermore, the total effect of `gini` as estimated by the probit model is equal to 1.31, meaning that, on average, an increase of 0.01 units of `gini` is related to an additional 1.31% probability of high homicide rates as a result of the summation of the direct and indirect effects. The direct ME is equal to 1.84.[4] This finding suggests that an own-county 0.01-unit increase in the Gini index (`gini`) is to increase the probability of having high homicide rates by 1.84 percentage points. Furthermore, the indirect ME is equal to $-0.53$. Hence, the across-county spillover effect of a 0.01-unit `gini` increase is associated with a reduction of probability of having high homicide rates by 0.53%. The negative sign of this ME is coherent with the negative sign of $\rho$: the total effect of `gini` is positive, but the cross-county component opposes to the own-county effect because of the negative spatial correlation ($\rho = -0.375$). Overall, the MEs suggest that own-county income inequality increases the probability of having a high homicide rate, while other counties' income inequality reduces the probability of having a high homicide rate.

As expected, the logit MEs are remarkably similar to the corresponding probit ME estimates. The total-effect point estimate is equal to 1.28 (1.31 in the probit model), and the direct effect is equal to 1.88 (1.84 in the probit model); lastly, the indirect ME is equal to $-0.60$. However, the last is not significant; this reflects the fact that the nonsignificance of $\rho$ in the logit model, and thus the increase of predicted probability, is likely associated only with own-county income inequality.

```
. quietly spatbinary hrate_gt_p95 ln_population gini, wmat(W2)
. spatbinary_impact gini, dydx
Impact measures for gini
```

| | dydx | Delta-M˽d std. err. | z | p>\|z\| | [95 conf. | interval] |
|---|---|---|---|---|---|---|
| gini | | | | | | |
| total | 1.280354 | .1441128 | 8.884388 | 6.43e-19 | .997898 | 1.56281 |
| direct | 1.882478 | .5316348 | 3.540922 | .0003987 | .8404925 | 2.924463 |
| indirect | -.6021238 | .4444265 | -1.354833 | .1754707 | -1.473184 | .2689362 |

The same logit MEs can be obtained by coding the expression shown in the first row of table 2 directly into the `expression` option of `margins`. This is recommended for the more advanced Stata users who want to exploit the flexibility of `margins`. The

4. Usually, MEs refer to a unit increase in an explanatory variable. However, `gini` is defined in $[0, 1]$. Thus, a unit increase is very large. Because MEs refer to infinitesimal changes by definition, a 0.01-unit increase is used for reference.

following Stata output uses `margins` to replicate the logit MEs obtained above with `spatbinary_impact`. The reader is reminded that `totalmargin`, `directmargin`, and `indirectmargin` calculate the term $g_i$ (see section 2.4).

```
. margins, expression(predict(totalmargin)*_b[hrate_gt_p95:gini])
Predictive margins                                      Number of obs = 1,412
Model VCE: Robust

Expression: predict(totalmargin)*_b[hrate_gt_p95:gini]

------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       _cons |   1.280354   .1441128     8.88   0.000      .997898     1.56281
------------------------------------------------------------------------------

. margins, expression(predict(directmargin)*_b[hrate_gt_p95:gini])
Predictive margins                                      Number of obs = 1,412
Model VCE: Robust

Expression: predict(directmargin)*_b[hrate_gt_p95:gini]

------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       _cons |   1.882478   .5316348     3.54   0.000     .8404925    2.924463
------------------------------------------------------------------------------

. margins, expression(predict(indirectmargin)*_b[hrate_gt_p95:gini])
Predictive margins                                      Number of obs = 1,412
Model VCE: Robust

Expression: predict(indirectmargin)*_b[hrate_gt_p95:gini]

------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       _cons |  -.6021238   .4444265    -1.35   0.175    -1.473184    .2689362
------------------------------------------------------------------------------
```

## 5.3 Example 2—Beyond default instruments, the impower() option

In this example, the linearized models of example 1 are refit by replacing the default set of instruments with a more complex set. The task is achieved using the option `impower()`. A higher `impower()` value captures more complex spatial autocorrelation patterns and may result in more accurate estimates. For instance, Arbia (2014) suggests that `impower(2)` would eliminate the endogeneity implied in (1). Other contributions use `impower(1)` (Klier and McMillen 2008) or `impower(3)` (Calabrese and Elkink 2014). As in the previous example, leaving `instr()` and `winstr()` unspecified is equivalent to specifying `instr(ln_population gini)` and `winstr(ln_population gini)`. In this example, `impower()` is equal to 3; the instruments are based on the independent variables that are multiplied by the first, second, and third powers of the spatial weight matrix `W2`.[5]

5. In keeping with the matrix notation of section 2, the resulting instrument matrix is equal to the linearly independent columns of $\boldsymbol{Z} = [\boldsymbol{X}, \boldsymbol{W}\boldsymbol{X}, \boldsymbol{W}^2\boldsymbol{X}, \boldsymbol{W}^3\boldsymbol{X}]$.

```
. spatbinary hrate_gt_p95 ln_population gini, wmat(W2) impower(3)
instruments set as (X,WX...W^n X) where X= ln_population gini and W=W2 where n=3
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
LINEARIZED LOGIT
------------------------------------------------------------------------------
             |               Robust
hrate_gt_p95 | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
hrate_gt_p95 |
ln_population|   .2662776   .1624493     1.64   0.101    -.0521171    .5846723
        gini |   39.88435   6.826781     5.84   0.000      26.5041    53.26459
       _cons |  -23.34124   4.557517    -5.12   0.000    -32.27381   -14.40867
-------------+----------------------------------------------------------------
rho          |
       _cons |  -.4260932   .2674599    -1.59   0.111    -.9503049    .0981186
------------------------------------------------------------------------------
Test of overidentifying restriction:
Hansen's J chi2(5) = .2871224, p = .9978783
. spatbinary hrate_gt_p95 ln_population gini, wmat(W2) probit impower(3)
instruments set as (X,WX...W^n X) where X= ln_population gini and W=W2 where n=3
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
LINEARIZED PROBIT
------------------------------------------------------------------------------
             |               Robust
hrate_gt_p95 | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
hrate_gt_p95 |
ln_population|   .0898347   .0383285     2.34   0.019     .0147123    .1649571
        gini |   18.67784   2.281629     8.19   0.000     14.20593    23.14975
       _cons |  -10.77687   1.309242    -8.23   0.000    -13.34294     -8.2108
-------------+----------------------------------------------------------------
rho          |
       _cons |  -.3540264   .1596558    -2.22   0.027     -.666946   -.0411068
------------------------------------------------------------------------------
Test of overidentifying restriction:
Hansen's J chi2(5) = 1.352185, p = .9294714
```

Compared with example 1, the coefficients of gini and ln_population as well as the spatial correlation parameters are rather stable. Again, $\rho$ is significant only in the probit model, and the test of overidentification restriction is insignificant. The total effect of gini on the predicted probability that hrate_gt_p95 $= 1$ is positive, and MEs are reported to quantify such effects.

The gini ME estimates for the probit model are very close to those of example 1. A 0.01-unit increase in gini is related to a 1.297% additional probability of high homicide rate. Again, the direct and indirect MEs have an opposing effect. The former is equal to 1.792, and the latter is equal to $-0.495$. The low significance of the cross-county (indirect) ME suggests that the dispersion in the propensity is weak; the effect of the explanatory variables is predominantly direct.

```
. quietly spatbinary hrate_gt_p95 ln_population gini, wmat(W2) probit impower(3)
. spatbinary_impact gini, dydx
Impact measures for gini
```

| | dydx | Delta-M_d std. err. | z | p>\|z\| | [95 conf. | interval] |
|---|---|---|---|---|---|---|
| gini | | | | | | |
| total | 1.296701 | .1279006 | 10.13835 | 3.73e-24 | 1.046021 | 1.547382 |
| direct | 1.79178 | .316274 | 5.665277 | 1.47e-08 | 1.171894 | 2.411665 |
| indirect | -.4950784 | .2535009 | -1.952965 | .0508238 | -.9919311 | .0017743 |

## 5.4 Example 3—N2SLS estimation

In this example, the N2SLS model with default instruments is fit (that is, they are the same as example 1). Because the `start()` option is not specified, `spatbinary` internally fits the linearized model of example 1 to provide initial values. For both models, Hansen's test for overidentification restrictions is safely insignificant, suggesting that the instruments are relevant.

The coefficient estimates are remarkably similar to those of the previous examples. However, `rho` is significant only at the 10% level of confidence, suggesting that the underlying spatial structure is weak. Moreover, the probit model standard errors of `rho` are larger than those reported in the corresponding linearized model (example 1): this may reflect model misspecification. Indeed, as reported in section 2.3, a correctly specified N2SLS model should produce smaller standard errors than the corresponding linearized model.

```
. spatbinary hrate_gt_p95 ln_population gini, wmat(W2) n2sls
instruments set as (X,WX...W^n X) where X= ln_population gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
Iteration        1:   GMM criterion Q(b) =        0.000146697964
Iteration        2:   GMM criterion Q(b) =        0.000042062807
Iteration        3:   GMM criterion Q(b) =        0.000041673907
Iteration        4:   GMM criterion Q(b) =        0.000041671036
N2SLS LOGIT
```

| hrate_gt_p95 | Coefficient | Robust std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| hrate_gt_p95 | | | | | | |
| ln_population | .2088806 | .176295 | 1.18 | 0.236 | -.1366513 | .5544124 |
| gini | 41.17571 | 6.693724 | 6.15 | 0.000 | 28.05625 | 54.29517 |
| _cons | -23.58003 | 4.516926 | -5.22 | 0.000 | -32.43304 | -14.72702 |
| rho | | | | | | |
| _cons | -.4242538 | .2173661 | -1.95 | 0.051 | -.8502837 | .001776 |

```
Test of overidentifying restriction:
Hansen's J chi2(1) = .0588395, p = .8083396
```

```
. spatbinary hrate_gt_p95 ln_population gini, wmat(W2) probit n2sls
instruments set as (X,WX...W^n X) where X= ln_population gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
Iteration        1:   GMM criterion Q(b) =        0.000424593826
Iteration        2:   GMM criterion Q(b) =        0.000178838091
Iteration        3:   GMM criterion Q(b) =        0.000178187561
Iteration        4:   GMM criterion Q(b) =        0.000178181342
N2SLS PROBIT
```

| hrate_gt_p95 | Coefficient | Robust std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| hrate_gt_p95 | | | | | | |
| ln_population | .0706771 | .0753894 | 0.94 | 0.349 | -.0770834 | .2184376 |
| gini | 19.87325 | 3.100784 | 6.41 | 0.000 | 13.79583 | 25.95068 |
| _cons | -11.25317 | 2.000446 | -5.63 | 0.000 | -15.17397 | -7.332368 |
| rho | | | | | | |
| _cons | -.3841139 | .2059677 | -1.86 | 0.062 | -.7878033 | .0195754 |

```
Test of overidentifying restriction:
Hansen's J chi2(1) = .2515921, p = .6159563
```

Again, MEs for the probit model are reported. Consistent with examples 1 and 2, the direct and total point estimates are positive, while indirect effects are negative. The total effect suggests that a 0.01-unit increase of income inequality as measured by the Gini index (`gini`) is associated with a 1.19% additional probability of having a high homicide rate. This is the combination of a positive direct effect and a negative cross-county effect ($-0.50$, $p = 0.05$). In absolute value, MEs are lower than those estimated in examples 1 and 2 by the linearized models.

```
. quietly spatbinary hrate_gt_p95 ln_population gini, wmat(W2) probit n2sls
. spatbinary_impact gini, dydx
Impact measures for gini
```

| | dydx | Delta-M˽d std. err. | z | p>\|z\| | [95 conf. | interval] |
|---|---|---|---|---|---|---|
| gini | | | | | | |
| total | 1.198613 | .1923828 | 6.230355 | 4.65e-10 | .8215498 | 1.575676 |
| direct | 1.698983 | .2554343 | 6.65135 | 2.90e-11 | 1.198341 | 2.199625 |
| indirect | -.5003701 | .2555695 | -1.957863 | .0502461 | -1.001277 | .0005369 |

## 5.5 Example 4—Simulated data with $\rho = 0.7$

In this example, the models are correctly specified because they are based on the data-generating function. The first BSAR logit model is based on the linearized estimator, and the second and third are based on N2SLS. The estimated value of $\rho$ is maximum in the first regression. This is expected because the data-generating value is higher than 0.5 (the linearized model is upwardly biased). However, the real value (0.7) is included in the confidence interval. In general, the true coefficients ($-10$ for the intercept and 22 for `gini`) are safely included in their related confidence intervals.

The N2SLS models have lower standard errors; this was expected because this estimator is more efficient than the linearized version. In the second model, the number of instruments (3) is equal to the number of parameters to be estimated; hence, instrument validity cannot be tested. To perform this test, `impower(2)` is specified in the third regression. Specifically, Hansen's test in the third model implies that the second-order instruments (a constant, `gini`, `W2*gini`, and `(W2^2)*gini`) are valid. Overall, the positive sign of $\rho$ suggests that the propensities of `Y1=1` are clustered in space: counties with high probability of `Y1=1` are located close to each other. This pattern also suggests that the cross-county (indirect) effect of the explanatory variables have the same sign of the direct effect.

```
. spatbinary Y1 gini, logit wmat(W2)
instruments set as (X,WX...W^n X) where X= gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
LINEARIZED LOGIT
```

| Y1 | Coefficient | Robust std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| Y1 | | | | | | |
| gini | 26.07231 | 6.827789 | 3.82 | 0.000 | 12.69009 | 39.45453 |
| _cons | -11.59576 | 3.3201 | -3.49 | 0.000 | -18.10303 | -5.08848 |
| rho | | | | | | |
| _cons | .6625091 | .1468327 | 4.51 | 0.000 | .3747222 | .9502959 |

```
Test of overidentifying restriction:
n2sls is just-identified, not possible to estimate Hansen's J
```

```
. spatbinary Y1 gini, logit wmat(W2) n2sls
instruments set as (X,WX...W^n X) where X= gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
Iteration        1:   GMM criterion Q(b) =        0.003925561540
Iteration        2:   GMM criterion Q(b) =        0.000063763964
Iteration        3:   GMM criterion Q(b) =        0.000000809832
Iteration        4:   GMM criterion Q(b) =        0.000000000450
Iteration        5:   GMM criterion Q(b) =        0.000000000000
N2SLS LOGIT

                           Robust
          Y1 | Coefficient std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
Y1           |
        gini |   27.87301  4.555002     6.12   0.000     18.94537    36.80065
       _cons |  -12.77346  2.157557    -5.92   0.000    -17.00219   -8.544722
-------------+----------------------------------------------------------------
rho          |
       _cons |   .6496192  .1083006     6.00   0.000     .4373539    .8618845
------------------------------------------------------------------------------
Test of overidentifying restriction:
n2sls is just-identified, not possible to estimate Hansen's J
. spatbinary Y1 gini, logit wmat(W2) impower(2) n2sls
instruments set as (X,WX...W^n X) where X= gini and W=W2 where n=2
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
Iteration        1:   GMM criterion Q(b) =        0.004657264068
Iteration        2:   GMM criterion Q(b) =        0.000089837854
Iteration        3:   GMM criterion Q(b) =        0.000002027419
Iteration        4:   GMM criterion Q(b) =        0.000000448757
Iteration        5:   GMM criterion Q(b) =        0.000000447104
N2SLS LOGIT

                           Robust
          Y1 | Coefficient std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
Y1           |
        gini |   27.51214  4.478326     6.14   0.000     18.73478     36.2895
       _cons |  -12.59915   2.11081    -5.97   0.000    -16.73626    -8.46204
-------------+----------------------------------------------------------------
rho          |
       _cons |   .6584248  .1024808     6.42   0.000      .457566    .8592835
------------------------------------------------------------------------------
Test of overidentifying restriction:
Hansen's J chi2(1) = .0006313, p = .9799545
```

To give more precise quantitative indications about the effect of `gini` on the predicted probability that `Y1=1`, I report the four types of measures of impact (see section 2.4). As the choice between the logit and probit models, the type of measure of impact to be reported depends on the context of research. Measures of impact are estimated using **`margins`** for illustrative purposes. In the following Stata outputs, the order of reporting is total impact, direct impact, and indirect impact. The following command calculates the MEs of `gini`. It is equivalent to `spatbinary_impact gini,` dydx. The MEs depict that a 0.01-unit increase in the Gini index yields to a 2.31% additional predicted probability that Y1=1. Both the direct and indirect effects are positive; the larger component is related to the cross-county effect of gini because the indirect effect is equal to 1.43 and the direct effect is equal to 0.88.

```
. margins, expression(predict(totalmargin)*_b[Y1:gini])
Predictive margins                                        Number of obs = 1,412
Model VCE: Robust

Expression: predict(totalmargin)*_b[Y1:gini]

──────────────┬────────────────────────────────────────────────────────────────
              │            Delta-method
              │     Margin   std. err.      z    P>|z|     [95% conf. interval]
──────────────┼────────────────────────────────────────────────────────────────
        _cons │   2.311126   .2699797     8.56   0.000     1.781976    2.840277
──────────────┴────────────────────────────────────────────────────────────────

. margins, expression(predict(directmargin)*_b[Y1:gini])
Predictive margins                                        Number of obs = 1,412
Model VCE: Robust

Expression: predict(directmargin)*_b[Y1:gini]

──────────────┬────────────────────────────────────────────────────────────────
              │            Delta-method
              │     Margin   std. err.      z    P>|z|     [95% conf. interval]
──────────────┼────────────────────────────────────────────────────────────────
        _cons │   .8850184   .1768917     5.00   0.000      .538317     1.23172
──────────────┴────────────────────────────────────────────────────────────────

. margins, expression(predict(indirectmargin)*_b[Y1:gini])
Predictive margins                                        Number of obs = 1,412
Model VCE: Robust

Expression: predict(indirectmargin)*_b[Y1:gini]

──────────────┬────────────────────────────────────────────────────────────────
              │            Delta-method
              │     Margin   std. err.      z    P>|z|     [95% conf. interval]
──────────────┼────────────────────────────────────────────────────────────────
        _cons │   1.426108   .3510701     4.06   0.000     .7380231    2.114193
──────────────┴────────────────────────────────────────────────────────────────
```

Elasticity estimates are reported below. Specifying spatbinary_impact gini, eyex gives the same results. The elasticity represents the proportional increase of the predicted probability associated with a 1% proportional (that is, multiplicative) increase of gini. The total elasticity is equal to 24. Again, the indirect effect (15.11) is stronger than the direct effect (9.38).

```
. margins, expression(predict(totalmargin)*_b[Y1:gini]*gini/predict(pr))
Predictive margins                                        Number of obs = 1,412
Model VCE: Robust

Expression: predict(totalmargin)*_b[Y1:gini]*gini/predict(pr)

──────────────┬────────────────────────────────────────────────────────────────
              │            Delta-method
              │     Margin   std. err.      z    P>|z|     [95% conf. interval]
──────────────┼────────────────────────────────────────────────────────────────
        _cons │   24.49555   2.748893     8.91   0.000     19.10782    29.88328
──────────────┴────────────────────────────────────────────────────────────────
```

```
. margins, expression(predict(directmargin)*_b[Y1:gini]*gini/predict(pr))
Predictive margins                                 Number of obs = 1,412
Model VCE: Robust
Expression: predict(directmargin)*_b[Y1:gini]*gini/predict(pr)
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| _cons | 9.38348 | 1.874251 | 5.01 | 0.000 | 5.710015 | 13.05695 |

```
. margins, expression(predict(indirectmargin)*_b[Y1:gini]*gini/predict(pr))
Predictive margins                                 Number of obs = 1,412
Model VCE: Robust
Expression: predict(indirectmargin)*_b[Y1:gini]*gini/predict(pr)
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| _cons | 15.11207 | 3.671687 | 4.12 | 0.000 | 7.915697 | 22.30845 |

The semielasticities giving the additive increase in probability that `Y1=1` in response to a proportional increase of `gini` are reported in the Stata output below. They can be obtained by specifying `spatbinary_impact gini, dyex`. This means that a 1% (multiplicative) increase of `gini` leads to approximately a percentage point of the predicted probability that `Y1=1`. Specifically, this increase is mostly attributable to the cross-county (indirect) effect: a 1% proportional increase of `gini` is related to a 0.62% positive variation in terms of predicted probability.

```
. margins, expression(predict(totalmargin)*_b[Y1:gini]*gini)
Predictive margins                                 Number of obs = 1,412
Model VCE: Robust
Expression: predict(totalmargin)*_b[Y1:gini]*gini
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| _cons | 1.006526 | .1153539 | 8.73 | 0.000 | .7804369 | 1.232616 |

```
. margins, expression(predict(directmargin)*_b[Y1:gini]*gini)
Predictive margins                                 Number of obs = 1,412
Model VCE: Robust
Expression: predict(directmargin)*_b[Y1:gini]*gini
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| _cons | .3854259 | .0773847 | 4.98 | 0.000 | .2337548 | .5370971 |

```
. margins, expression(predict(indirectmargin)*_b[Y1:gini]*gini)
Predictive margins                                   Number of obs = 1,412
Model VCE: Robust
Expression: predict(indirectmargin)*_b[Y1:gini]*gini
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| _cons | .6211004 | .1515894 | 4.10 | 0.000 | .3239906 | .9182101 |

Lastly, to calculate the semielasticities giving the proportional increase in probability of `Y1=1` in response to a unit increase of `gini`, Stata users can plug into `margins` the expression `predict(totalmargin)*_b[Y1:gini]/predict(pr)`. The equivalent postestimation command is `spatbinary_impact gini, eydx`. As expected, the indirect effect is stronger than the direct effect. In quantitative terms, such an effect implies that a 0.01-unit increase in `gini` in other geographical units is related to a 38.7% proportional increase in the predicted probability.

```
. margins, expression(predict(totalmargin)*_b[Y1:gini]/predict(pr))
Predictive margins                                   Number of obs = 1,412
Model VCE: Robust
Expression: predict(totalmargin)*_b[Y1:gini]/predict(pr)
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| _cons | 62.74073 | 7.048419 | 8.90 | 0.000 | 48.92608 | 76.55538 |

```
. margins, expression(predict(directmargin)*_b[Y1:gini]/predict(pr))
Predictive margins                                   Number of obs = 1,412
Model VCE: Robust
Expression: predict(directmargin)*_b[Y1:gini]/predict(pr)
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| _cons | 24.03833 | 4.80989 | 5.00 | 0.000 | 14.61112 | 33.46555 |

```
. margins, expression(predict(indirectmargin)*_b[Y1:gini]/predict(pr))
Predictive margins                                   Number of obs = 1,412
Model VCE: Robust
Expression: predict(indirectmargin)*_b[Y1:gini]/predict(pr)
```

| | Margin | Delta-method std. err. | z | P>\|z\| | [95% conf. | interval] |
|---|---|---|---|---|---|---|
| _cons | 38.7024 | 9.398716 | 4.12 | 0.000 | 20.28125 | 57.12354 |

## 5.6 Example 5—Simulated data with $\rho = 0.2$

As in the previous example, the models are correctly specified. Because $\rho$ is small, the linearized model estimates and the first set of N2SLS coefficients are essentially the same, with a gain in efficiency related to the standard errors of the independent variables. Again, the data-generating coefficients are safely included in their related confidence intervals. The models highlight that the total effect of `gini` on the propensity of `Y1=1` is positive. Clustering of such propensity ($\rho > 0$) implies that the direct and indirect effects of `gini` act in the same direction.

```
. spatbinary Y2 gini, logit wmat(W2)
instruments set as (X,WX...W^n X) where X= gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
LINEARIZED LOGIT

------------------------------------------------------------------------------
             |               Robust
          Y2 | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
Y2           |
        gini |   19.56906   3.333957     5.87   0.000     13.03463     26.1035
       _cons |  -8.935875    1.52985    -5.84   0.000    -11.93433   -5.937424
-------------+----------------------------------------------------------------
rho          |
       _cons |   .1918303   .1650448     1.16   0.245    -.1316516    .5153121
------------------------------------------------------------------------------
Test of overidentifying restriction:
n2sls is just-identified, not possible to estimate Hansen's J
. spatbinary Y2 gini, logit wmat(W2) n2sls
instruments set as (X,WX...W^n X) where X= gini and W=W2 where n=1
  (1412 observations)
  (1412 observations (places) used)
  (weighting matrix defines 1412 places)
Iteration        1:   GMM criterion Q(b) =        0.000008245188
Iteration        2:   GMM criterion Q(b) =        0.000000000197
Iteration        3:   GMM criterion Q(b) =        0.000000000000
N2SLS LOGIT

------------------------------------------------------------------------------
             |               Robust
          Y2 | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
Y2           |
        gini |   19.61657   3.114418     6.30   0.000     13.51242    25.72072
       _cons |  -8.970221   1.432197    -6.26   0.000    -11.77728   -6.163167
-------------+----------------------------------------------------------------
rho          |
       _cons |   .1921856   .1654734     1.16   0.245    -.1321363    .5165074
------------------------------------------------------------------------------
Test of overidentifying restriction:
n2sls is just-identified, not possible to estimate Hansen's J
```

# 6 Conclusions

This article discussed the implementation of `spatbinary`, a Stata package that allows users to fit BSAR logit and probit models. The `spatbinary` command extends the suite of community-contributed and official commands for spatial regression. A possible development of `spatbinary` would be its extension to multinomial outcomes.

# 7 Acknowledgments

This article is adapted from the third chapter of my PhD thesis, titled "Patient choice: Two essays and a statistical software package." I am grateful to Gianmaria Martini for his supervision, to Paolo Berta, and to an anonymous referee for comments that greatly improved the manuscript.

# 8 Programs and supplemental materials

To install a snapshot of the corresponding software files as they existed at the time of publication of this article, type

```
. net sj 22-2
. net install st0672      (to install program files, if available)
. net get st0672          (to install ancillary files, if available)
```

# 9 References

Anselin, L., R. J. G. M. Florax, and S. J. Rey, eds. 2004. *Advances in Spatial Econometrics: Methodology, Tools and Applications*. Berlin: Springer.

Arbia, G. 2014. *A Primer for Spatial Econometrics: With Applications in R*. London: Palgrave Macmillan.

Belotti, F., G. Hughes, and A. P. Mortari. 2017. Spatial panel-data models using Stata. *Stata Journal* 17: 139–180. https://doi.org/10.1177/1536867X1701700109.

Beron, K. J., and W. P. M. Vijverberg. 2004. Probit in a spatial context: A Monte Carlo analysis. In *Advances in Spatial Econometrics: Methodology, Tools and Applications*, ed. L. Anselin, R. J. G. M. Florax, and S. J. Rey, 169–195. Berlin: Springer. https://doi.org/10.1007/978-3-662-05617-2_8.

Billé, A. G. 2013. Computational issues in the estimation of the spatial probit model: A comparison of various estimators. *Review of Regional Studies* 43: 131–154. https://doi.org/10.52324/001c.8088.

Billé, A. G., and S. Leorato. 2020. Partial ML estimation for spatial autoregressive nonlinear probit models with autoregressive disturbances. *Econometric Reviews* 39: 437–475. https://doi.org/10.1080/07474938.2019.1682314.

Brasington, D., A. Flores-Lagunes, and L. Guci. 2016. A spatial model of school district open enrollment choice. *Regional Science and Urban Economics* 56: 1–18. https://doi.org/10.1016/j.regsciurbeco.2015.10.005.

Calabrese, R., and J. A. Elkink. 2014. Estimators of binary spatial autoregressive models: A Monte Carlo study. *Journal of Regional Science* 54: 664–687. https://doi.org/10.1111/jors.12116.

Drukker, D. M., H. Peng, I. R. Prucha, and R. Raciborski. 2013. Creating and managing spatial-weighting matrices with the spmat command. *Stata Journal* 13: 242–286. https://doi.org/10.1177/1536867X1301300202.

Drukker, D. M., I. R. Prucha, and R. Raciborski. 2013. Maximum likelihood and generalized spatial two-stage least-squares estimators for a spatial-autoregressive model with spatial-autoregressive disturbances. *Stata Journal* 13: 221–241. https://doi.org/10.1177/1536867X1301300201.

Flores-Lagunes, A., and K. E. Schnier. 2012. Estimation of sample selection models with spatial dependence. *Journal of Applied Econometrics* 27: 173–204. https://doi.org/10.1002/jae.1189.

Greene, W. H. 2018. *Econometric Analysis*. 8th ed. New York: Pearson.

Hansen, L. P. 1982. Large sample properties of generalized method of moments estimators. *Econometrica* 50: 1029–1054. https://doi.org/10.2307/1912775.

Huber, S., and C. Rust. 2016. Calculate travel time and distance with OpenStreetMap data using the Open Source Routing Machine (OSRM). *Stata Journal* 16: 416–423. https://doi.org/10.1177/1536867X1601600209.

Kelejian, H. H., and I. R. Prucha. 1998. A generalized spatial two-stage least squares procedure for estimating a spatial autoregressive model with autoregressive disturbances. *Journal of Real Estate Finance and Economics* 17: 99–121. https://doi.org/10.1023/A:1007707430416.

Klier, T., and D. P. McMillen. 2008. Clustering of auto supplier plants in the United States: Generalized method of moments spatial logit for large samples. *Journal of Business and Economic Statistics* 26: 460–471. https://doi.org/10.1198/073500107000000188.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. In *Frontiers in Econometrics*, ed. P. Zerembka, 105–142. New York: Academic Press.

McMillen, D. P. 1992. Probit with spatial autocorrelation. *Journal of Regional Science* 32: 335–348. https://doi.org/10.1111/j.1467-9787.1992.tb00190.x.

Messner, S. F., L. Anselin, D. F. Hawkins, G. Deane, S. E. Tolnay, and R. D. Baller. 2000. An Atlas of the Spatial Patterning of County-Level Homicide, 1960–1990. Pittsburgh: National Consortium on Violence Research.

Ozimek, A., and D. Miles. 2011. Stata utilities for geocoding and generating travel time and travel distance information. *Stata Journal* 11: 106–119. https://doi.org/10.1177/1536867X1101100107.

Pinkse, J., and M. E. Slade. 1998. Contracting in space: An application of spatial statistics to discrete-choice models. *Journal of Econometrics* 85: 125–154. https://doi.org/10.1016/S0304-4076(97)00097-3.

Pisati, M. 2001. sg162: Tools for spatial data analysis. *Stata Technical Bulletin* 60: 21–37. Reprinted in *Stata Technical Bulletin Reprints*. Vol. 10, pp. 277–298. College Station, TX: Stata Press.

———. 2018. spmap: Stata module to visualize spatial data. Statistical Software Components S456812, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s456812.html.

Prato, C. G., S. Kaplan, A. Patrier, and T. K. Rasmussen. 2018. Considering built environment and spatial correlation in modeling pedestrian injury severity. *Traffic Injury Prevention* 19: 88–93. https://doi.org/10.1080/15389588.2017.1329535.

Weber, S., and M. Péclat. 2017. A simple command to calculate travel distance and travel time. *Stata Journal* 17: 962–971. https://doi.org/10.1177/1536867X1801700411.

Wilhelm, S., and M. G. de Matos. 2013. Estimating spatial probit models in R. *R Journal* 5: 130–143. https://doi.org/10.32614/RJ-2013-013.

Williams, R. 2012. Using the margins command to estimate and interpret adjusted predictions and marginal effects. *Stata Journal* 12: 308–331. https://doi.org/10.1177/1536867X1201200209.

Wooldridge, J. M. 2010. *Econometric Analysis of Cross Section and Panel Data*. 2nd ed. Cambridge, MA: MIT Press.

### About the author

Daniele Spinelli is a postdoc researcher at the University of Milan–Bicocca. He holds a PhD in economics and management of technology (DREAMT), and his main research interests are health econometrics, cluster-weighted models, and Stata.