

spxtregress — Spatial autoregressive models for panel data

Description	Quick start	Menu
Syntax	Options for spxtregress, fe	Options for spxtregress, re
Remarks and examples	Stored results	Methods and formulas
References	Also see	

Description

`spxtregress` fits spatial autoregressive (SAR) models, also known as simultaneous autoregressive models, for panel data. The commands `spxtregress, fe` and `spxtregress, re` are extensions of `xtreg, fe` and `xtreg, re` for spatial data; see [\[XT\] xtreg](#).

If you have not read [\[SP\] Intro 1](#)–[\[SP\] Intro 8](#), you should do so before using `spxtregress`.

To use `spxtregress`, your data must be Sp data and `xtset`. See [\[SP\] Intro 3](#) for instructions on how to prepare your data.

To specify spatial lags, you will need to have one or more spatial weighting matrices. See [\[SP\] Intro 2](#) and [\[SP\] spmatrix](#) for an explanation of the types of weighting matrices and how to create them.

Quick start

SAR fixed-effects model of y on x_1 and x_2 with a spatial lag of y specified by the spatial weighting matrix W

```
spxtregress y x1 x2, fe dvarlag(W)
```

Add a spatially lagged error term also specified by W

```
spxtregress y x1 x2, fe dvarlag(W) errorlag(W)
```

Add spatial lags of covariates x_1 and x_2

```
spxtregress y x1 x2, fe dvarlag(W) errorlag(W) ivarlag(W: x1 x2)
```

Add an additional spatial lag of the covariates specified by the matrix M

```
spxtregress y x1 x2, fe dvarlag(W) errorlag(W) ivarlag(W: x1 x2) ///
ivarlag(M: x1 x2)
```

SAR random-effects model

```
spxtregress y x1 x2, re dvarlag(W) errorlag(W) ivarlag(W: x1 x2) ///
ivarlag(M: x1 x2)
```

An `re` model with panel effects that follow the same spatial process as the errors using `sarpanel`

```
spxtregress y x1 x2, re sarpanel dvarlag(W) errorlag(W) ///
ivarlag(W: x1 x2) ivarlag(M: x1 x2)
```

Menu

Statistics > Spatial autoregressive models

Syntax

Fixed-effects maximum likelihood

```
spxtregress depvar [indepvars] [if] [in], fe [fe_options]
```

Random-effects maximum likelihood

```
spxtregress depvar [indepvars] [if] [in], re [re_options]
```

fe_options

Description

Model

* fe	use fixed-effects estimator
<u>dvarlag</u> (<i>spmatname</i>)	spatially lagged dependent variable
<u>errorlag</u> (<i>spmatname</i>)	spatially lagged errors
<u>ivarlag</u> (<i>spmatname</i> : <i>varlist</i>)	spatially lagged independent variables; repeatable
force	allow estimation when estimation sample is a subset of the sample used to create the spatial weighting matrix
<u>gridsearch</u> (#)	resolution of the initial-value search grid; seldom used

Reporting

<u>level</u> (#)	set confidence level; default is level(95)
<i>display_options</i>	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization

<i>maximize_options</i>	control the maximization process; seldom used
<u>coeflegend</u>	display legend instead of statistics

<i>re_options</i>	Description
Model	
* re	use random-effects estimator
dvarlag (<i>spmatname</i>)	spatially lagged dependent variable
errorlag (<i>spmatname</i>)	spatially lagged errors
ivarlag (<i>spmatname</i> : <i>varlist</i>)	spatially lagged independent variables; repeatable
sarpanel	alternative formulation of the estimator in which the panel effects follow the same spatial process as the errors
noconstant	suppress constant term
force	allow estimation when estimation sample is a subset of the sample used to create the spatial weighting matrix
Reporting	
level (#)	set confidence level; default is <code>level(95)</code>
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
coeflegend	display legend instead of statistics

* You must specify either `fe` or `re`.

indepvars and *varlist* specified in `ivarlag()` may contain factor variables; see [U] 11.4.3 Factor variables.

`collect` is allowed; see [U] 11.1.10 Prefix commands.

`coeflegend` does not appear in the dialog box.

See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

Options for `spxtregress`, `fe`

Model

`fe` requests the fixed-effects regression estimator.

`dvarlag(spmatname)` specifies a spatial weighting matrix that defines a spatial lag of the dependent variable. Only one `dvarlag()` option may be specified. By default, no spatial lags of the dependent variable are included.

`errorlag(spmatname)` specifies a spatial weighting matrix that defines a spatially lagged error. Only one `errorlag()` option may be specified. By default, no spatially lagged errors are included.

`ivarlag(spmatname : varlist)` specifies a spatial weighting matrix and a list of independent variables that define spatial lags of the variables. This option is repeatable to allow spatial lags created from different matrices. By default, no spatial lags of the independent variables are included.

`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 first to understand the

spatial space of the estimation sample and, if it is sensible, then create new weighting matrices for it. See [SP] **spmat** and *Missing values, dropped observations, and the W matrix* in [SP] **Intro 2**. `gridsearch(#)` specifies the resolution of the initial-value search grid. The default is `gridsearch(0.1)`. You may specify any number between 0.001 and 0.1 inclusive.

Reporting

`level(#)`; see [R] **Estimation options**.

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

Maximization

`maximize_options`: `difficult`, `technique(algorithm_spec)`, `iterate(#)`, `[no]log`, `trace`, `gradient`, `showstep`, `hessian`, `showtolerance`, `tolerance(#)`, `ltolerance(#)`, `nrtolerance(#)`, and `nonrtolerance`; see [R] **Maximize**.

The following option is available with `spxtregress`, `fe` but is not shown in the dialog box: `coeflegend`; see [R] **Estimation options**.

Options for `spxtregress`, `re`

Model

`re` requests the generalized least-squares random-effects estimator.

`dvarlag(spmatname)` specifies a spatial weighting matrix that defines a spatial lag of the dependent variable. Only one `dvarlag()` option may be specified. By default, no spatial lags of the dependent variable are included.

`errorlag(spmatname)` specifies a spatial weighting matrix that defines a spatially lagged error. Only one `errorlag()` option may be specified. By default, no spatially lagged errors are included.

`ivarlag(spmatname : varlist)` specifies a spatial weighting matrix and a list of independent variables that define spatial lags of the variables. This option is repeatable to allow spatial lags created from different matrices. By default, no spatial lags of the independent variables are included.

`sarpanel` requests an alternative formulation of the estimator in which the panel effects follow the same spatial process as the errors. By default, the panel effects are included in the estimation equation as an additive term, just as they are in the standard nonspatial random-effects model. When `sarpanel` and `errorlag(spmatname)` are specified, the panel effects also have a spatial autoregressive form based on `spmatname`. If `errorlag()` is not specified with `sarpanel`, the estimator is identical to the estimator when `sarpanel` is not specified. The `sarpanel` estimator was originally developed by Kapoor, Kelejian, and Prucha (2007); see *Methods and formulas*.

`noconstant`; 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. This is the same `force` option described for use with `spxtregress`, `fe`.

Reporting

level(#); see [R] [Estimation options](#).

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](#).

Maximization

maximize_options: [difficult](#), [technique\(algorithm_spec\)](#), [iterate\(#\)](#), [\[no\]log](#), [trace](#), [gradient](#), [showstep](#), [hessian](#), [showtolerance](#), [tolerance\(#\)](#), [ltolerance\(#\)](#), [nrtolerance\(#\)](#), and [nonrtolerance](#); see [R] [Maximize](#).

The following option is available with `spxtregress`, `re` but is not shown in the dialog box: `coeflegend`; see [R] [Estimation options](#).

Remarks and examples

stata.com

See [SP] [Intro](#) for an overview of SAR models.

Datasets for Sp panel models contain observations on geographical areas or other units with multiple observations on each unit. See [SP] [Intro 3](#) for an explanation of how to work with Sp panel data. The data must be `xtset` and must be strongly balanced. There must be a within-panel identifier, a variable indicating time or the equivalent, and the values of this identifier must be the same for every panel. The command `spbalance` will strongly balance datasets that are not strongly balanced. See [SP] [Intro 3](#), [SP] [Intro 7](#), and [SP] [spbalance](#).

Remarks and examples are presented under the following headings:

- [Sp panel models](#)
- [The fixed-effects model](#)
- [The random-effects model](#)
- [The random-effects model with autoregressive panel effects](#)
- [Differences among models](#)
- [Examples](#)

Sp panel models

Both the fixed-effects and the random-effects models for spatial panel data can be written as

$$\begin{aligned} \mathbf{y}_{nt} &= \lambda \mathbf{W} \mathbf{y}_{nt} + \mathbf{X}_{nt} \boldsymbol{\beta} + \mathbf{c}_n + \mathbf{u}_{nt} \\ \mathbf{u}_{nt} &= \rho \mathbf{M} \mathbf{u}_{nt} + \mathbf{v}_{nt} \end{aligned} \quad t = 1, 2, \dots, T \tag{1}$$

where $\mathbf{y}_{nt} = (y_{1t}, y_{2t}, \dots, y_{nt})'$ is an $n \times 1$ vector of observations for the dependent variable for time period t with n number of panels; \mathbf{X}_{nt} is a matrix of time-varying regressors; \mathbf{c}_n is a vector of panel-level effects; \mathbf{u}_{nt} is the spatially lagged error; \mathbf{v}_{nt} is a vector of disturbances and is independent and identically distributed (i.i.d.) across panels and time with variance σ^2 ; and \mathbf{W} and \mathbf{M} are spatial weighting matrices.

The fixed-effects model

For fixed effects, `spxtregress`, `fe` implements the quasimaximum likelihood (QML) estimator in [Lee and Yu \(2010a\)](#) to fit the model. A transformation is used to eliminate the fixed effects from the equations, yielding

$$\begin{aligned}\tilde{\mathbf{y}}_{nt} &= \lambda \mathbf{W} \tilde{\mathbf{y}}_{nt} + \tilde{\mathbf{X}}_{nt} \beta + \tilde{\mathbf{u}}_{nt} \\ \tilde{\mathbf{u}}_{nt} &= \rho \mathbf{M} \tilde{\mathbf{u}}_{nt} + \tilde{\mathbf{v}}_{nt} \quad t = 1, 2, \dots, T - 1\end{aligned}$$

Panel effects, which are effects that are constant within panels, are conditioned out of the likelihood. Only covariates that vary within panels can be fit with this estimator.

The random-effects model

For random effects, `spxtregress`, `re` assumes that \mathbf{c}_n in (1) is normal i.i.d. across panels with mean 0 and variance σ_c^2 . The output of `spxtregress`, `re` displays estimates of σ_c , labeled as `/sigma_u`, and σ , labeled as `/sigma_e`, which is consistent with how `xtreg`, `re` labels the output.

The random-effects model with autoregressive panel effects

The `sarpanel` option for random-effects models fits a slightly different set of equations from (1):

$$\begin{aligned}\mathbf{y}_{nt} &= \lambda \mathbf{W} \mathbf{y}_{nt} + \mathbf{X}_{nt} \beta + \mathbf{u}_{nt} \\ \mathbf{u}_{nt} &= \rho \mathbf{M} \mathbf{u}_{nt} + \mathbf{c}_n + \mathbf{v}_{nt}, \quad t = 1, 2, \dots, T\end{aligned}$$

In this variant due to [Kapoor, Kelejian, and Prucha \(2007\)](#), the panel-level effects \mathbf{c}_n are considered a disturbance in the error equation. Because \mathbf{c}_n enters the equation as an additive term next to \mathbf{v}_{nt} , the panel-level effects \mathbf{c}_n have the same autoregressive form as the time-level errors \mathbf{v}_{nt} .

Differences among models

All three of the models—`fe`, `re`, and `re sarpanel`—are fit using maximum likelihood (ML) estimation. The differences are 1) `fe` removes the panel-level effects from the estimation and no distributional assumptions are made about them; 2) `re` models the panel-level effects as normal i.i.d.; and 3) `re sarpanel` assumes a normal distribution for panel-level effects but with the same autoregressive form as the time-level errors. The `fe` model allows the panel-level effects to be correlated with the observed covariates, whereas the `re` models require that the panel-level effects are independent of the observed covariates. See [Methods and formulas](#) for details. Also see [Choosing weighting matrices and their normalization](#) in [SP] `spregress`; the discussion there applies to these three estimation models.

Examples

▷ Example 1: spxtregress, re

We have data on the homicide rate in counties in southern states of the U.S. for the years 1960, 1970, 1980, and 1990. `homicide_1960_1990.dta` contains `hrate`, the county-level homicide rate per year per 100,000 persons for each of the four years. It also contains `ln_population`, the logarithm of the county population; `ln_pdensity`, the logarithm of the population density; and `gini`, the Gini coefficient for the county, a measure of income inequality where larger values represent more inequality (Gini 1909). The data are an extract of the data originally used by Messner et al. (2000); see Britt (1994) for a literature review of the topic. The 1990 data are used in the examples in [SP] [spregress](#).

We used `spshape2dta` to convert shapefiles into Stata `.dta` files, and then we merged the data file by county ID with our homicide-rate data. See [SP] [Intro 4](#), [SP] [Intro 7](#), [SP] [spshape2dta](#), and [SP] [spset](#).

Because the analysis dataset and the Stata-formatted shapefile must be in our working directory to `spset` the data, we first save both `homicide_1960_1990.dta` and `homicide_1960_1990_shp.dta` to our working directory by using the `copy` command. We then load the data and type `spset` to see the Sp settings.

```
. copy https://www.stata-press.com/data/r18/homicide_1960_1990.dta .
. copy https://www.stata-press.com/data/r18/homicide_1960_1990_shp.dta .
. use homicide_1960_1990
(S.Messner et al.(2000), U.S southern county homicide rate in 1960-1990)
. spset
      Sp dataset: homicide_1960_1990.dta
Linked shapefile: homicide_1960_1990_shp.dta
      Data: Cross sectional
      Spatial-unit ID: _ID
      Coordinates: _CX, _CY (planar)
variable _ID does not uniquely identify the observations in the master data
r(459);
```

We get an error! The data have not been `xtset`, and `spxtregress` requires it. Our data consist of 1,412 counties, and for each county we have data for four years. Our data look like this:

```
. list _ID year in 1/8, sepby(_ID)
```

	_ID	year
1.	876	1960
2.	876	1970
3.	876	1980
4.	876	1990
5.	921	1960
6.	921	1970
7.	921	1980
8.	921	1990

We type

```
. xtset _ID year
Panel variable: _ID (strongly balanced)
Time variable: year, 1960 to 1990, but with gaps
Delta: 1 unit
```

`xtset` reports that our data are strongly balanced. Each county has data for the same four years. `spxtregress` requires the data to be strongly balanced. Missing values in our variables could cause the estimation sample to be unbalanced. The Sp panel estimators will complain, and we will have to make the data strongly balanced for the nonmissing values of the variables in our model. If you get a message that your data are not strongly balanced, see [SP] [spbalance](#).

After having `xtset` our data, we type `spset` to check our Sp settings.

```
. spset
      Sp dataset: homicide_1960_1990.dta
Linked shapefile: homicide_1960_1990_shp.dta
      Data: Panel
Spatial-unit ID: _ID
      Time ID: year (see xtset)
Coordinates: _CX, _CY (planar)
```

We first run a nonspatial random-effects model by using `xtreg`, `re` and include dummies for the years by using the `i.year` [factor-variable](#) notation.

```
. xtreg hrate ln_population ln_pdensity gini i.year, re
Random-effects GLS regression           Number of obs   =       5,648
Group variable: _ID                    Number of groups =       1,412
R-squared:                               Obs per group:
    Within = 0.0478                      min =           4
    Between = 0.1666                      avg =          4.0
    Overall = 0.0905                      max =           4
Wald chi2(6) =          414.32
Prob > chi2 =          0.0000
```

```
corr(u_i, X) = 0 (assumed)
```

	hrate	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
ln_populat~n		.4394103	.1830599	2.40	0.016	.0806194	.7982012
ln_pdensity		.3220698	.1591778	2.02	0.043	.0100872	.6340525
gini		34.43792	2.905163	11.85	0.000	28.7439	40.13193
year							
1970		1.411074	.2579218	5.47	0.000	.9055562	1.916591
1980		1.347822	.2499977	5.39	0.000	.8578352	1.837808
1990		.3668468	.2648395	1.39	0.166	-.1522291	.8859228
_cons		-10.07267	1.800932	-5.59	0.000	-13.60243	-6.542908
sigma_u		3.5995346					
sigma_e		5.646151					
rho		.28898083	(fraction of variance due to u_i)				

We emphasize that you can ignore the spatial aspect of the data and use any of Stata's estimation commands even though the data are spatial. Doing that is often a good idea. It provides a baseline against which you can compare subsequent spatial results.

We are now going to fit a spatial random-effects model. To do that, we need a spatial weighting matrix. We will create one that puts the same positive weight on contiguous counties and a 0 weight

on all other counties—a matrix known as a contiguity matrix. We will use the default spectral normalization for this example. See [SP] [spmatrix create](#). When we create the matrix, we must restrict `spmatrix create` to one observation per panel. That is easy to do using an `if` statement:

```
. spmatrix create contiguity W if year == 1990
```

Do not misinterpret the purpose of `if year == 1990`. The matrix created will be appropriate for creating spatial lags for any year, because our map does not change. If two counties share a border in 1990, they share it in the other years too.

We can now fit our model. We include a spatial lag of the dependent variable and a spatially autoregressive error term.

```
. spxtregress hrate ln_population ln_pdensity gini i.year, re dvarlag(W)
> errorlag(W)
(5648 observations)
(5648 observations used)
(data contain 1412 panels (places) )
(weighting matrix defines 1412 places)
```

Fitting starting values:

```
Iteration 0: Log likelihood = -13299.332
Iteration 1: Log likelihood = -13298.431
Iteration 2: Log likelihood = -13298.43
Iteration 3: Log likelihood = -13298.43
```

Optimizing concentrated log likelihood:

```
Initial:      Log likelihood = -18826.009
Improvement: Log likelihood = -18826.009
Rescale:     Log likelihood = -18826.009
Rescale eq:  Log likelihood = -18500.374
Iteration 0: Log likelihood = -18500.374 (not concave)
Iteration 1: Log likelihood = -18473.617 (not concave)
Iteration 2: Log likelihood = -18465.327
Iteration 3: Log likelihood = -18433.377
Iteration 4: Log likelihood = -18356.27
Iteration 5: Log likelihood = -18354.861
Iteration 6: Log likelihood = -18354.84
Iteration 7: Log likelihood = -18354.84
```

Optimizing unconcentrated log likelihood:

```
Iteration 0: Log likelihood = -18354.84
Iteration 1: Log likelihood = -18354.84 (backed up)
```

```

Random-effects spatial regression      Number of obs   =    5,648
Group variable: _ID                   Number of groups =    1,412
                                       Obs per group   =         4
                                       Wald chi2(7)    =   1421.81
                                       Prob > chi2     =    0.0000
Log likelihood = -1.835e+04           Pseudo R2      =    0.0911

```

hrate	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
hrate						
ln_populat~n	-.2988717	.1622145	-1.84	0.065	-.6168063	.019063
ln_pdensity	.7893218	.138061	5.72	0.000	.5187272	1.059916
gini	22.77052	2.604623	8.74	0.000	17.66555	27.87549
year						
1970	.3977167	.1906034	2.09	0.037	.024141	.7712925
1980	.4033443	.1825721	2.21	0.027	.0455096	.7611789
1990	-.1284625	.1946898	-0.66	0.509	-.5100474	.2531225
_cons	-4.182031	1.607558	-2.60	0.009	-7.332787	-1.031275
W						
hrate	.5740163	.0249799	22.98	0.000	.5250565	.6229761
e.hrate	-.4626345	.0508732	-9.09	0.000	-.5623441	-.3629248
/sigma_u	3.087656	.1046893			2.889138	3.299815
/sigma_e	5.40831	.0661566			5.280188	5.539542

```

Wald test of spatial terms:          chi2(2) = 713.88      Prob > chi2 = 0.0000

```

`spxtregress, re` first fits an `spxtregress, fe` model to get starting values. Then, it optimizes the concentrated log likelihood and then optimizes the unconcentrated log likelihood. The final log likelihood of the concentrated will always be equal to the optimized log likelihood of the unconcentrated. The unconcentrated starts at the right point, takes a step to check that it is the right point, backs up to this point, and declares convergence as it should.

We can compare estimates of `/sigma_u`, the standard deviation of the panel effects, and `/sigma_e`, the standard deviation of the errors, with those fit by `xtreg, re`. They are similar. We cannot, however, directly compare the coefficient estimates with those of `xtreg, re`. When a spatial lag of the dependent variable is included in the model, covariates have both direct and indirect effects, as explained in [example 1](#) of [\[SP\] spxtregress](#). To obtain the direct, indirect, and total effects of the covariates, we must use `estat impact`.

Here are the averages of the effects of gini:

```
. estat impact gini
progress   :100%
Average impacts                Number of obs   =       5,648
```

	Delta-Method dy/dx	std. err.	z	P> z	[95% conf. interval]	
direct gini	24.11439	2.7159	8.88	0.000	18.79133	29.43746
indirect gini	22.73745	2.787573	8.16	0.000	17.27391	28.201
total gini	46.85185	5.126095	9.14	0.000	36.80489	56.89881

The percentages at the top of the output indicate progress in the estimation process. For large datasets, calculating standard errors of the effects can be time consuming, so `estat impact` reports its progress as it does the computations.

`gini` has significant average direct and average indirect effects on `hrate`, with both being positive. An increase in inequality is associated with an increase in the homicide rate.

We used a contiguity weighting matrix `W` for the spatial lags. Alternatively, we can use a weighting matrix based on the inverse distance between counties. We create this matrix, using again the default spectral normalization:

```
. spmatrix create idistance M if year == 1990
. spmatrix dir
```

Weighting matrix name	N x N	Type	Normalization
M	1412 x 1412	idistance	spectral
W	1412 x 1412	contiguity	spectral

We would like to know if the effects of `gini` differ over time, so we include an interaction of `gini` and `year` in our model, and we use the weighting matrix `M` that we just created.

```
. spxtregress hrate ln_population ln_pdensity c.gini##i.year, re
> dvarlag(M) errorlag(M)
(5648 observations)
(5648 observations used)
(data contain 1412 panels (places) )
(weighting matrix defines 1412 places)
(output omitted)

Random-effects spatial regression      Number of obs   =      5,648
Group variable: _ID                   Number of groups =      1,412
                                       Obs per group   =           4
                                       Wald chi2(10)   =      710.10
                                       Prob > chi2     =      0.0000
Log likelihood = -1.827e+04            Pseudo R2      =      0.1150
```

hrate	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
hrate						
ln_populat~n	.7908003	.1764819	4.48	0.000	.444902	1.136698
ln_pdensity	-.1223671	.166526	-0.73	0.462	-.4487521	.2040179
gini	17.82039	4.278782	4.16	0.000	9.43413	26.20665
year						
1970	-2.456656	2.303073	-1.07	0.286	-6.970596	2.057284
1980	-9.470622	2.501528	-3.79	0.000	-14.37353	-4.567717
1990	-22.81817	2.528691	-9.02	0.000	-27.77432	-17.86203
year#c.gini						
1970	6.664314	6.130454	1.09	0.277	-5.351156	18.67978
1980	24.86122	6.715029	3.70	0.000	11.70001	38.02243
1990	57.40946	6.691102	8.58	0.000	44.29514	70.52378
_cons	-11.17804	2.061047	-5.42	0.000	-15.21762	-7.138466
M						
hrate	.694492	.0496075	14.00	0.000	.5972631	.7917209
e.hrate	1.950078	.0513563	37.97	0.000	1.849422	2.050735
/sigma_u	2.696022	.1147302			2.480277	2.930533
/sigma_e	5.645628	.0618616			5.525674	5.768186

Wald test of spatial terms: chi2(2) = 1711.11 Prob > chi2 = 0.0000

Using the `contrast` command, we test the significance of the `gini` and `year` interaction:

```
. contrasts c.gini#year
Contrasts of marginal linear predictions
Margins: asbalanced
```

	df	chi2	P>chi2
hrate			
year#c.gini	3	81.59	0.0000

The interaction is significant. We can explore the effect of gini by year using estat impact with an if statement.

```
. estat impact gini if year == 1960
progress :100%
Average impacts                               Number of obs    =      1,412
```

		Delta-Method dy/dx	std. err.	z	P> z	[95% conf. interval]	
direct	gini	17.85376	4.285828	4.17	0.000	9.453696	26.25383
indirect	gini	37.06435	11.60647	3.19	0.001	14.3161	59.81261
total	gini	54.91812	14.85784	3.70	0.000	25.79729	84.03894

```
. estat impact gini if year == 1970
progress :100%
Average impacts                               Number of obs    =      1,412
```

		Delta-Method dy/dx	std. err.	z	P> z	[95% conf. interval]	
direct	gini	24.53056	5.033544	4.87	0.000	14.66499	34.39613
indirect	gini	50.92535	15.21236	3.35	0.001	21.10968	80.74103
total	gini	75.45591	18.81752	4.01	0.000	38.57425	112.3376

```
. estat impact gini if year == 1980
progress :100%
Average impacts                               Number of obs    =      1,412
```

		Delta-Method dy/dx	std. err.	z	P> z	[95% conf. interval]	
direct	gini	42.76155	5.683654	7.52	0.000	31.62179	53.9013
indirect	gini	88.77282	23.09515	3.84	0.000	43.50716	134.0385
total	gini	131.5344	26.20928	5.02	0.000	80.16512	182.9036

```
. estat impact gini if year == 1990
progress   :100%
Average impacts                                Number of obs   =       1,412
```

		Delta-Method		z	P> z	[95% conf. interval]	
		dy/dx	std. err.				
direct							
	gini	75.37074	5.628584	13.39	0.000	64.33892	86.40256
indirect							
	gini	156.4694	37.24056	4.20	0.000	83.47923	229.4596
total							
	gini	231.8401	39.01862	5.94	0.000	155.365	308.3152

The `if year == ...` statement used with `estat impact` allows us to estimate the average effects for each year. The direct, indirect, and total effects of `gini` trend upward.

Until now, we used the default form of the random-effects estimator. Let's run the command again, specifying the `sarpanel` option to use the alternative form of the estimator, where the panel-level effects have the same autoregressive form as the time-level errors.

```

. spxtregress hrate ln_population ln_pdensity c.gini##i.year, re sarpanel
> dvarlag(M) errorlag(M)
(5648 observations)
(5648 observations used)
(data contain 1412 panels (places) )
(weighting matrix defines 1412 places)
(output omitted)

Random-effects spatial regression          Number of obs   =      5,648
Group variable:  _ID                      Number of groups =      1,412
                                           Obs per group   =           4
                                           Wald chi2(10)  =     1136.49
                                           Prob > chi2    =      0.0000
                                           Pseudo R2     =      0.1177

Log likelihood = -1.824e+04

```

	hrate	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
hrate							
ln_population		.4366742	.1752499	2.49	0.013	.0931906	.7801578
ln_pdensity		.1896	.1641331	1.16	0.248	-.1320949	.511295
gini		18.92328	4.426236	4.28	0.000	10.24802	27.59854
year							
1970		-.9590229	2.362018	-0.41	0.685	-5.588493	3.670447
1980		-8.19778	2.554509	-3.21	0.001	-13.20453	-3.191035
1990		-22.4189	2.610158	-8.59	0.000	-27.53472	-17.30309
year#c.gini							
1970		5.865776	6.255307	0.94	0.348	-6.3944	18.12595
1980		24.20335	6.83421	3.54	0.000	10.80855	37.59816
1990		58.38273	6.881913	8.48	0.000	44.89443	71.87103
_cons		-6.535916	2.257848	-2.89	0.004	-10.96122	-2.110615
M							
hrate		.3317434	.0967132	3.43	0.001	.142189	.5212978
e.hrate		2.860571	.0558304	51.24	0.000	2.751145	2.969996
/sigma_u		2.686156	.1123355			2.474764	2.915605
/sigma_e		5.609948	.0612095			5.491253	5.731208

Wald test of spatial terms: chi2(2) = 2685.83 Prob > chi2 = 0.0000

The re and re sarpanel estimators give appreciably different estimates for the coefficient of the spatial lag of hrate and for the autoregressive error term. Estimates of other terms are similar. It appears that some of the spatial-lag effect of hrate is being accounted for by the autoregressive form of the panel effects in the sarpanel model.

► Example 2: `spxtregress`, `fe`

The random-effects estimator assumes that the panel-level effects are uncorrelated with the covariates in the model. We can relax that assumption using the fixed-effects estimator.

We will fit fixed-effects models for the same data we used in [example 1](#). Here's a nonspatial model fit with `xtreg`, `fe`.

```
. xtreg hrate ln_population ln_pdensity gini, fe
Fixed-effects (within) regression      Number of obs   =      5,648
Group variable:  _ID                  Number of groups =      1,412
R-squared:                             Obs per group:
    Within = 0.0356                    min =          4
    Between = 0.0084                   avg  =          4.0
    Overall = 0.0131                    max  =          4
                                         F(3, 4233)     =      52.04
                                         Prob > F       =      0.0000
```

	hrate	Coefficient	Std. err.	t	P> t	[95% conf. interval]
ln_populat	ln	-2.16467	1.702073	-1.27	0.204	-5.501627 1.172286
ln_pdensity		1.007573	1.659751	0.61	0.544	-2.246409 4.261555
gini		35.12694	2.816652	12.47	0.000	29.60483 40.64906
_cons		13.90421	10.91007	1.27	0.203	-7.485242 35.29366
sigma_u		5.2469262				
sigma_e		5.7428609				
rho		.45496484	(fraction of variance due to u_i)			

F test that all u_i=0: F(1411, 4233) = 2.61 Prob > F = 0.0000

We now use `spxtregress`, `fe` and include a spatial lag of the dependent variable `hrate`.

```
. spxtregress hrate ln_population ln_pdensity gini, fe dvarlag(M)
(5648 observations)
(5648 observations used)
(data contain 1412 panels (places) )
(weighting matrix defines 1412 places)
Performing grid search ... finished
Optimizing concentrated log likelihood:
Iteration 0: Log likelihood = -13321.27
Iteration 1: Log likelihood = -13321.27 (backed up)
Iteration 2: Log likelihood = -13321.269
Optimizing unconcentrated log likelihood:
Iteration 0: Log likelihood = -13321.269
Iteration 1: Log likelihood = -13321.269 (backed up)
```



```

Fixed-effects spatial regression      Number of obs   =    5,648
Group variable: _ID                 Number of groups =    1,412
                                      Obs per group   =         4
                                      Wald chi2(4)    =    548.39
                                      Prob > chi2     =    0.0000
Log likelihood = -1.332e+04          Pseudo R2      =    0.0146
    
```

	hrate	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
hrate							
ln_populat~n		-1.852636	1.662249	-1.11	0.265	-5.110586	1.405313
ln_pdensity		-.0352675	1.621715	-0.02	0.983	-3.21377	3.143235
gini		11.58058	3.001197	3.86	0.000	5.698348	17.46282
M							
hrate		.8982519	.0457977	19.61	0.000	.80849	.9880138
/sigma_e		5.608237	.0609629			5.490016	5.729004

Wald test of spatial terms: chi2(1) = 384.69 Prob > chi2 = 0.0000

spxtregress, fe does not give an estimate of /sigma_u because the spatial fixed-effects estimator does not give consistent estimates for the levels of the panel fixed effects nor for their standard deviation. See [Methods and formulas](#).

We cannot fit a fixed-effects model with all the terms we included in [example 1](#). The i.year dummies are not allowed because spxtregress, fe assumes individual fixed effects only, as specified in section 2 of [Lee and Yu \(2010a\)](#).

In [example 1](#), we found that `gini` was an important regressor and that the effect of `gini` differed across time. We will use Stata's [factor-variable](#) notation and add to the model `c.gini#i.year`, which is `gini` interacted by `year` without main effects.

```
. spxtregress hrate ln_population ln_pdensity c.gini#i.year, fe
> dvarlag(M) errorlag(M)
(5648 observations)
(5648 observations used)
(data contain 1412 panels (places) )
(weighting matrix defines 1412 places)
(output omitted)
Fixed-effects spatial regression          Number of obs   =    5,648
Group variable:  _ID                     Number of groups =    1,412
                                           Obs per group   =         4
                                           Wald chi2(7)    =   128.16
                                           Prob > chi2     =    0.0000
Log likelihood = -1.330e+04              Pseudo R2       =    0.0001
```

hrate	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
hrate						
ln_populat~n	-2.169113	1.70931	-1.27	0.204	-5.519298	1.181073
ln_pdensity	-.7395584	1.638919	-0.45	0.652	-3.95178	2.472663
year#c.gini						
1960	4.637191	4.648659	1.00	0.319	-4.474013	13.74839
1970	11.15786	4.234694	2.63	0.008	2.858016	19.45771
1980	11.92355	4.158854	2.87	0.004	3.77235	20.07476
1990	11.13694	3.975613	2.80	0.005	3.344885	18.929
M						
hrate	.1251126	.2552472	0.49	0.624	-.3751629	.625388
e.hrate	1.604259	.1898228	8.45	0.000	1.232213	1.976305
/sigma_e	5.582721	.0606909			5.465027	5.702949

Wald test of spatial terms: chi2(2) = 116.83 Prob > chi2 = 0.0000

We look at the effects:

```
. estat impact
progress   : 33% 67% 100%
Average impacts                               Number of obs   =       5,648
```

	Delta-Method		z	P> z	[95% conf. interval]	
	dy/dx	std. err.				
direct						
ln_populat~n	-2.169186	1.709375	-1.27	0.204	-5.5195	1.181127
ln_pdensity	-.7395835	1.638973	-0.45	0.652	-3.951911	2.472744
gini	9.714218	4.112071	2.36	0.018	1.654706	17.77373
indirect						
ln_populat~n	-.2894662	.7155598	-0.40	0.686	-1.691938	1.113005
ln_pdensity	-.0986934	.3143279	-0.31	0.754	-.7147649	.517378
gini	1.29631	3.022576	0.43	0.668	-4.62783	7.22045
total						
ln_populat~n	-2.458653	2.065714	-1.19	0.234	-6.507378	1.590073
ln_pdensity	-.838277	1.867989	-0.45	0.654	-4.499469	2.822915
gini	11.01053	5.357526	2.06	0.040	.5099696	21.51109

The output shows the effects of gini across all the years. `estat impact` is smart enough to know that there are not year effects in the fixed-effects model. When it looks at the term `c.gini#i.year`, it only gives the effects for gini. If year were replaced by a variable that varied within time, `estat impact` would show the effects for that variable, too.

If we want to see how the effects of gini change across the years, we can use `if` with `estat impact` as we did in [example 1](#).

```
. estat impact gini if year == 1960
progress   :100%
Average impacts                               Number of obs   =       1,412
```

	Delta-Method		z	P> z	[95% conf. interval]	
	dy/dx	std. err.				
direct						
gini	4.637349	4.648981	1.00	0.319	-4.474487	13.74918
indirect						
gini	.6188292	1.70156	0.36	0.716	-2.716167	3.953826
total						
gini	5.256178	5.794721	0.91	0.364	-6.101266	16.61362

```
. estat impact gini if year == 1970
progress :100%
Average impacts                               Number of obs   =       1,412
```

		Delta-Method		z	P> z	[95% conf. interval]	
		dy/dx	std. err.				
direct							
	gini	11.15824	4.234355	2.64	0.008	2.859058	19.45743
indirect							
	gini	1.489007	3.335444	0.45	0.655	-5.048344	8.026358
total							
	gini	12.64725	5.00173	2.53	0.011	2.844038	22.45046

```
. estat impact gini if year == 1980
progress :100%
Average impacts                               Number of obs   =       1,412
```

		Delta-Method		z	P> z	[95% conf. interval]	
		dy/dx	std. err.				
direct							
	gini	11.92396	4.158654	2.87	0.004	3.773147	20.07477
indirect							
	gini	1.591188	3.62961	0.44	0.661	-5.522717	8.705093
total							
	gini	13.51515	5.380726	2.51	0.012	2.969119	24.06118

```
. estat impact gini if year == 1990
progress :100%
Average impacts                               Number of obs   =       1,412
```

		Delta-Method		z	P> z	[95% conf. interval]	
		dy/dx	std. err.				
direct							
	gini	11.13732	3.975637	2.80	0.005	3.345217	18.92943
indirect							
	gini	1.486215	3.459169	0.43	0.667	-5.293632	8.266062
total							
	gini	12.62354	5.485123	2.30	0.021	1.872894	23.37418

There is no evidence of a trend in the average total effect of `gini` from the `fe` model.

Stored results

`spxtregress, fe` and `spxtregress, re` store the following in `e()`:

Scalars

<code>e(N)</code>	number of observations
<code>e(N_g)</code>	number of groups (panels)
<code>e(g)</code>	group size
<code>e(k)</code>	number of parameters
<code>e(df_m)</code>	model degrees of freedom
<code>e(df_c)</code>	degrees of freedom for test of spatial terms
<code>e(ll)</code>	log likelihood
<code>e(iterations)</code>	number of maximum log-likelihood estimation iterations
<code>e(rank)</code>	rank of $e(V)$
<code>e(r2_p)</code>	pseudo- R^2
<code>e(chi2)</code>	χ^2
<code>e(chi2_c)</code>	χ^2 for test of spatial terms
<code>e(p)</code>	p -value for model test
<code>e(p_c)</code>	p -value for test of spatial terms
<code>e(converged)</code>	1 if converged, 0 otherwise

Macros

<code>e(cmd)</code>	<code>spxtregress</code>
<code>e(cmdline)</code>	command as typed
<code>e(depvar)</code>	name of dependent variable
<code>e(indeps)</code>	names of independent variables
<code>e(idvar)</code>	name of ID variable
<code>e(model)</code>	<code>fe</code> , <code>re</code> , or <code>re sarpanel</code>
<code>e(title)</code>	title in estimation output
<code>e(constant)</code>	<code>hasconstant</code> or <code>noconstant</code> (<code>re</code> only)
<code>e(dlmat)</code>	name of spatial weighting matrix applied to <i>depvar</i>
<code>e(elmat)</code>	name of spatial weighting matrix applied to errors
<code>e(chi2type)</code>	Wald; type of model χ^2 test
<code>e(vce)</code>	<code>oim</code>
<code>e(ml_method)</code>	type of ml method
<code>e(technique)</code>	maximization technique
<code>e(properties)</code>	<code>b V</code>
<code>e(estat_cmd)</code>	program used to implement <code>estat</code>
<code>e(predict)</code>	program used to implement <code>predict</code>
<code>e(marginsok)</code>	predictions allowed by <code>margins</code>
<code>e(asbalanced)</code>	factor variables <code>fvset</code> as <code>asbalanced</code>
<code>e(asobserved)</code>	factor variables <code>fvset</code> as <code>asobserved</code>

Matrices

<code>e(b)</code>	coefficient vector
<code>e(ilog)</code>	iteration log (up to 20 iterations)
<code>e(gradient)</code>	gradient vector
<code>e(Hessian)</code>	Hessian matrix
<code>e(V)</code>	variance–covariance matrix of the estimators

Functions

<code>e(sample)</code>	marks estimation sample
------------------------	-------------------------

In addition to the above, the following is stored in `r()`:

Matrices

<code>r(table)</code>	matrix containing the coefficients with their standard errors, test statistics, p -values, and confidence intervals
-----------------------	---

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any `r`-class command is run after the estimation command.

Methods and formulas

`spxtregress, fe` estimates the parameters of the SAR model with spatially autoregressive errors and fixed effects using the QML estimator derived by [Lee and Yu \(2010a\)](#).

`spxtregress, re` estimates the parameters of two different SAR models with spatially autoregressive errors and random effects. In the default model, the random effects enter the equation for the dependent variable linearly. This model and the ML estimator for its parameters were derived by [Lee and Yu \(2010b\)](#). When the `sarpanel` option is specified, the random effects are subject to the same spatial autoregressive process as the idiosyncratic errors. This model and the ML estimator of its parameters were derived by [Lee and Yu \(2010b\)](#), which builds on the original formulation by [Kapoor, Kelejian, and Prucha \(2007\)](#). All of these papers build on theoretical work in [Kelejian and Prucha \(2001\)](#) and [Lee \(2004\)](#). We use the estimator derived by [Baltagi and Liu \(2011\)](#) to get initial values.

Methods and formulas are presented under the following headings:

Fixed-effects estimators

Random-effects estimators

Fixed-effects estimators

The [Lee and Yu \(2010a\)](#) SAR model for panel data with fixed effects is

$$\begin{aligned} \mathbf{y}_{nt} &= \lambda \mathbf{W} \mathbf{y}_{nt} + \mathbf{X}_{nt} \beta + \mathbf{c}_n + \mathbf{u}_{nt} \\ \mathbf{u}_{nt} &= \rho \mathbf{M} \mathbf{u}_{nt} + \mathbf{v}_{nt} \quad t = 1, 2, \dots, T \end{aligned} \quad (2)$$

where

$\mathbf{y}_{nt} = (y_{1t}, y_{2t}, \dots, y_{nt})'$ is an $n \times 1$ vector of observations on the dependent variable for time period t ;

\mathbf{X}_{nt} is an $n \times k$ matrix of nonstochastic time-varying regressors for time period t . \mathbf{X}_{nt} may also contain spatial lag of exogenous covariates;

\mathbf{c}_n is an $n \times 1$ vector of individual effects;

\mathbf{u}_{nt} is an $n \times 1$ vector of spatially lagged error;

$\mathbf{v}_{nt} = (v_{1t}, v_{2t}, \dots, v_{nt})'$ is an $n \times 1$ vector of innovations, and v_{it} is i.i.d. across i and t with variance σ^2 ; and

\mathbf{W} and \mathbf{M} are $n \times n$ spatial weighting matrices.

`spxtregress, fe` estimates the parameters in this model by using the QML estimator derived by [Lee and Yu \(2010a\)](#). [Lee and Yu \(2010a\)](#) uses an orthogonal transformation to remove the fixed effects \mathbf{c}_n without inducing dependence in the transformed errors. The transform $\mathbf{F}_{T,T-1}$ is part of $[\mathbf{F}_{T,T-1}, 1/\sqrt{T} \mathbf{1}_T]$, which is the orthonormal eigenvector matrix of $(\mathbf{I}_T - 1/T \mathbf{1}_T \mathbf{1}'_T)$, where \mathbf{I}_T is the $T \times T$ identity matrix and $\mathbf{1}_T$ is a $T \times 1$ vector of 1s. [Kuersteiner and Prucha \(2015\)](#) discuss this class of transforms.

For any $n \times T$ matrix $[\mathbf{z}_{n1}, \mathbf{z}_{n2}, \dots, \mathbf{z}_{nT}]$, the transformed $n \times (T-1)$ matrix is defined as

$$[\tilde{\mathbf{z}}_{n1}, \tilde{\mathbf{z}}_{n2}, \dots, \tilde{\mathbf{z}}_{n,T-1}] = [\mathbf{z}_{n1}, \mathbf{z}_{n2}, \dots, \mathbf{z}_{nT}] \mathbf{F}_{T,T-1}$$

Thus, the transformed model for (2) is

$$\begin{aligned} \tilde{\mathbf{y}}_{nt} &= \lambda \mathbf{W} \tilde{\mathbf{y}}_{nt} + \tilde{\mathbf{X}}_{nt} \beta + \tilde{\mathbf{u}}_{nt} \\ \tilde{\mathbf{u}}_{nt} &= \rho \mathbf{M} \tilde{\mathbf{u}}_{nt} + \tilde{\mathbf{v}}_{nt} \quad t = 1, 2, \dots, T-1 \end{aligned}$$

The transformed innovations $\tilde{\mathbf{v}}_{nt}$ are uncorrelated for all i and t .

The log-likelihood function for the transformed model is

$$\ln L_{n,T}(\theta) = -\frac{n(T-1)}{2} \ln(2\pi\sigma^2) + (T-1)[\ln|\mathbf{S}_n(\lambda)| + \ln|\mathbf{R}_n(\rho)|] - \frac{1}{2\sigma^2} \sum_{t=1}^{T-1} \tilde{\mathbf{v}}'_{nt}(\theta) \tilde{\mathbf{v}}_{nt}(\theta)$$

where $\mathbf{S}_n(\lambda) = \mathbf{I}_n - \lambda \mathbf{W}$, $\mathbf{R}_n(\rho) = \mathbf{I}_n - \rho \mathbf{M}$, and $\theta = (\beta', \lambda, \rho, \sigma^2)'$.

Random-effects estimators

`spxtregress`, `re` fits two different random-effects SAR models for panel data. In the default model, the random effects enter the equation for \mathbf{y}_{nt} linearly.

$$\begin{aligned} \mathbf{y}_{nt} &= \lambda \mathbf{W} \mathbf{y}_{nt} + \mathbf{Z}_{nt} \beta + \mathbf{c}_n + \mathbf{u}_{nt} \\ \mathbf{u}_{nt} &= \rho \mathbf{M} \mathbf{u}_{nt} + \mathbf{v}_{nt} \quad t = 1, 2, \dots, T \end{aligned} \quad (3)$$

where

\mathbf{Z}_{nt} may contain time-variant and -invariant regressors;

\mathbf{c}_n is random effects with mean 0 and variance σ_c^2 ; and

all the other terms are defined as in (2).

When the `sarpanel` option is specified, `spxtregress`, `re` fits a model in which the random effects \mathbf{c}_n are subject to the same spatial autoregressive process as the errors.

$$\begin{aligned} \mathbf{y}_{nt} &= \lambda \mathbf{W} \mathbf{y}_{nt} + \mathbf{Z}_{nt} \beta + \mathbf{u}_{nt} \\ \mathbf{u}_{nt} &= \rho \mathbf{M} \mathbf{u}_{nt} + \mathbf{c}_n + \mathbf{v}_{nt} \quad t = 1, 2, \dots, T \end{aligned} \quad (4)$$

When the \mathbf{c}_n are treated as fixed effects and transformed out of the model, the default model in (3) is equivalent to the `sarpanel` model in (4). When treating the \mathbf{c}_n as random effects, these two models are different.

For (3) or (4), we can stack all the time periods and write the equations as an $nT \times 1$ vector form

$$\mathbf{y}_{nT} = \lambda(\mathbf{I}_T \otimes \mathbf{W}) \mathbf{y}_{nT} + \mathbf{Z}_{nT} \beta + \xi_{nT} \quad (5)$$

where

$\mathbf{y}_{nT} = (\mathbf{y}'_{n1}, \mathbf{y}'_{n2}, \dots, \mathbf{y}'_{nT})'$ is an $nT \times 1$ vector of observations of the dependent variable for $i = 1, \dots, n$ and $t = 1, \dots, T$;

$\mathbf{v}_{nT} = (\mathbf{v}'_{n1}, \mathbf{v}'_{n2}, \dots, \mathbf{v}'_{nT})'$ is an $nT \times 1$ vector of innovations;

$\mathbf{Z}_{nT} = \{\mathbf{Z}'_{n1}, \mathbf{Z}'_{n2}, \dots, \mathbf{Z}'_{nT}(\rho)\}'$ is an $nT \times k$ matrix of k regressors for $i = 1, \dots, n$ and $t = 1, \dots, T$; and

ξ_{nT} is the overall disturbance $nT \times 1$ vector.

For (3), the overall disturbance vector ξ_{nT} is

$$\xi_{nT} = \mathbf{I}_T \otimes \mathbf{c}_n + \{\mathbf{I}_T \otimes \mathbf{R}_n(\rho)^{-1}\} \mathbf{v}_{nT}$$

where $\mathbf{R}_n(\rho) = \mathbf{I}_n - \rho \mathbf{M}$. Its variance matrix is

$$\Omega_{nT}(\theta) = \sigma_c^2 (\mathbf{1}_T \mathbf{1}'_T \otimes \mathbf{I}_T) + \sigma^2 \{\mathbf{I}_T \otimes \mathbf{R}_n(\rho)^{-1} \mathbf{R}'_n(\rho)^{-1}\}$$

For (4), the overall disturbance vector ξ_{nT} is

$$\xi_{nT} = \mathbf{I}_T \otimes \mathbf{R}_n(\rho)^{-1} \mathbf{c}_n + \{\mathbf{I}_T \otimes \mathbf{R}_n(\rho)^{-1}\} \mathbf{v}_{nT}$$

Its variance matrix is

$$\Omega_{nT}(\theta) = \sigma_c^2 \{\mathbf{1}_T \mathbf{1}_T' \otimes \mathbf{R}_n(\rho)^{-1} \mathbf{R}_n'(\rho)^{-1}\} + \sigma^2 \{\mathbf{I}_T \otimes \mathbf{R}_n(\rho)^{-1} \mathbf{R}_n'(\rho)^{-1}\}$$

The log-likelihood function for (5) is

$$\ln L_{nT}(\theta) = -\frac{nT}{2} \ln(2\pi) - \frac{1}{2} \ln |\Omega_{nT}(\theta)| + T \ln |\mathbf{S}_n(\lambda)| - \frac{1}{2} \xi_{nT}'(\theta) \Omega_{nT}(\theta)^{-1} \xi_{nT}(\theta)$$

where $\mathbf{S}_n(\lambda) = \mathbf{I}_n - \lambda \mathbf{W}$, and $\theta = (\beta', \lambda, \rho, \sigma_c^2, \sigma^2)'$.

References

- Baltagi, B. H., and L. Liu. 2011. Instrumental variable estimation of a spatial autoregressive panel model with random effects. *Economics Letters* 111: 135–137. <https://doi.org/10.1016/j.econlet.2011.01.016>.
- Britt, C. L. 1994. Crime and unemployment among youths in the United States, 1958–1990: A time series analysis. *American Journal of Economics and Sociology* 53: 99–109. <https://doi.org/10.1111/j.1536-7150.1994.tb02680.x>.
- Gini, C. 1909. Concentration and dependency ratios (in Italian). English translation in *Rivista di Politica Economica* 1997 87: 769–789.
- Kapoor, M., H. H. Kelejjan, and I. R. Prucha. 2007. Panel data models with spatially correlated error components. *Journal of Econometrics* 140: 97–130. <https://doi.org/10.1016/j.jeconom.2006.09.004>.
- Kelejjan, H. H., and I. R. Prucha. 2001. On the asymptotic distribution of the Moran I test statistic with applications. *Journal of Econometrics* 104: 219–257. [https://doi.org/10.1016/S0304-4076\(01\)00064-1](https://doi.org/10.1016/S0304-4076(01)00064-1).
- Kuersteiner, G. M., and I. R. Prucha. 2015. Dynamic spatial panel models: Networks, common shocks, and sequential exogeneity. Working paper, Department of Economics, University of Maryland. http://econweb.umd.edu/~prucha/Papers/WP_GMK_IRP_2015.pdf.
- Lee, L.-F. 2004. Asymptotic distributions of quasi-maximum likelihood estimators for spatial autoregressive models. *Econometrica* 72: 1899–1925. <https://doi.org/10.1111/j.1468-0262.2004.00558.x>.
- Lee, L.-F., and J. Yu. 2010a. Estimation of spatial autoregressive panel data models with fixed effects. *Journal of Econometrics* 154: 165–185. <https://doi.org/10.1016/j.jeconom.2009.08.001>.
- . 2010b. Some recent developments in spatial panel data models. *Regional Science and Urban Economics* 40: 255–271. <https://doi.org/10.1016/j.regsciurbeco.2009.09.002>.
- Li, J., Z. Liao, and W. Zhou. 2023. Uniform nonparametric inference for spatially dependent panel data: The `xtnpsreg` command. *Stata Journal* 23: 243–264.
- 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.

Also see

- [SP] **spxtregress postestimation** — Postestimation tools for `spxtregress`
- [SP] **estat moran** — Moran’s test of residual correlation with nearby residuals
- [SP] **Intro** — Introduction to spatial data and SAR models
- [SP] **spbalance** — Make panel data strongly balanced
- [SP] **spivregress** — Spatial autoregressive models with endogenous covariates

[SP] [spmatrix](#) — Categorical guide to the spmatrix command

[SP] [spregress](#) — Spatial autoregressive models

[XT] [xtreg](#) — Fixed-, between-, and random-effects and population-averaged linear models⁺

[U] [20 Estimation and postestimation commands](#)

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on [citing Stata documentation](#).