SAS/ETS® 15.1 User's Guide

The SPATIALREG Procedure

This document is an individual chapter from *SAS/ETS® 15.1 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2018. *SAS/ETS® 15.1 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/ETS® 15.1 User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

November 2018

Chapter 31
The SPATIALREG Procedure

Contents

Overview: SPATIALREG Procedure

The SPATIALREG (spatial regression) procedure analyzes spatial econometric models for cross-sectional data whose observations are spatially referenced or georeferenced. For example, housing price data that are collected from 48 continental states in the United States fall into the category of spatially referenced data. Compared to nonspatial regression models, spatial econometric models are capable of handling spatial interaction and spatial heterogeneity in a regression setting (Anselin 2001).

The SPATIALREG procedure supports the following models:

- linear model
- linear model with spatial lag of X (SLX) effects
- spatial autoregressive (SAR) model
- spatial Durbin model (SDM)
- spatial error model (SEM)
- spatial Durbin error model (SDEM)
- spatial moving average (SMA) model
- spatial Durbin moving average (SDMA) model
- spatial autoregressive moving average (SARMA) model
- spatial Durbin autoregressive moving average (SDARMA) model

- spatial autoregressive confused (SAC) model
- spatial Durbin autoregressive confused (SDAC) model

In general, SARMA, SDARMA, SAC, and SDAC models can require two spatial weights matrices. If you fit these four types of models by using the SPATIALREG procedure in SAS/ETS 14.2, the two spatial weights matrices are assumed to be identical.

Spatial econometric models have been widely used in economics, political science, sociology, and other fields. For example, LeSage and Pace (2009) provide a detailed introduction to commonly used spatial econometric models from both frequentist and Bayesian perspectives. A brief introduction to spatial econometric models is also provided by Elhorst (2013).

The SPATIALREG procedure in SAS/ETS 14.2 primarily uses the maximum likelihood estimation to achieve parameter estimation. Initial values for the nonlinear optimizations are usually calculated by ordinary least squares (OLS).

Getting Started: SPATIALREG Procedure

The SPATIALREG procedure is similar to other SAS regression model procedures, except that you usually need to provide a secondary data set (in the WMAT= option). The spatial weights matrix defines all pairwise spatial relationships and is the most vital component of a spatial regression model. For more information about how to create spatial weights matrix, see the section "Specifying the Spatial Weights Matrix" on page 2327.

The following statements fit a SAR model:

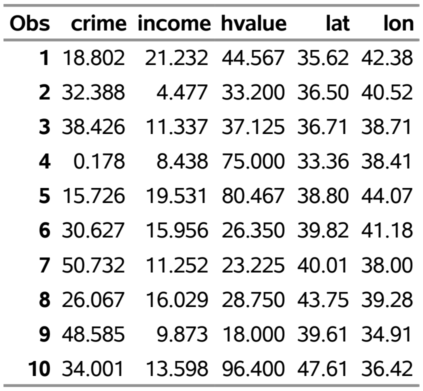
```
proc spatialreg data=one Wmat=W;
   model y = x1 x2 / type=SAR;
run;
```

The response variable y is continuous, and the data set W, which you specify in the WMAT= option, contains the spatial relationships among all spatial units in the data. In this case, W is either contiguity or weights. You specify the TYPE=SAR option to request a SAR model.

The following example illustrates PROC SPATIALREG by using a real-world data set. The data set CRIMEOH is taken from Anselin (1988) and can be found in the SAS/ETS Sample Library. This data set contains variables such as INCOME (household income, measured in $1000), HVALUE (housing value by $1000), and CRIME (number of crimes, including residential burglaries and vehicle thefts, measured per 1,000 households) in 49 neighborhoods in Columbus, Ohio. You want to examine how household income and housing value affect the number of crimes in the 49 neighborhoods of interest.

The first 10 observations in the CRIMEOH data set are shown in Figure 31.1.

Figure 31.1 Columbus Crime Data



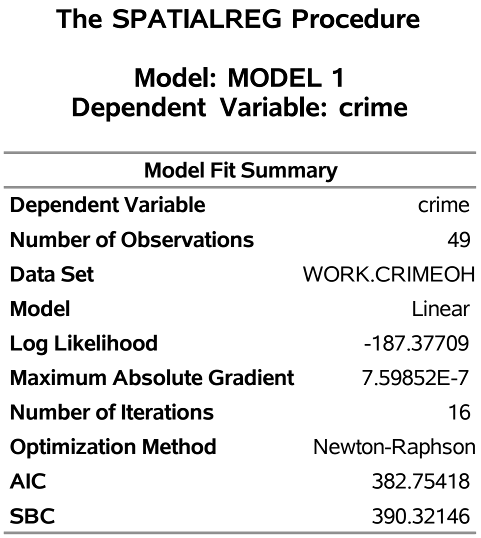
Obs	crime	income	hvalue	lat	lon
1	18.802	21.232	44.567	35.62	42.38
2	32.388	4.477	33.200	36.50	40.52
3	38.426	11.337	37.125	36.71	38.71
4	0.178	8.438	75.000	33.36	38.41
5	15.726	19.531	80.467	38.80	44.07
6	30.627	15.956	26.350	39.82	41.18
7	50.732	11.252	23.225	40.01	38.00
8	26.067	16.029	28.750	43.75	39.28
9	48.585	9.873	18.000	39.61	34.91
10	34.001	13.598	96.400	47.61	36.42

The following SAS statements fit a linear regression model to the CRIMEOH data set:

```
proc spatialreg data=crimeoh;
   model crime = income hvalue / type=LINEAR;
run;
```

The "Model Fit Summary" table, shown in Figure 31.2, lists several fit summary statistics about the model. By default, the SPATIALREG procedure uses the Newton-Raphson optimization technique. The maximum log-likelihood value is shown, in addition to two information measures, Akaike's information criterion (AIC) and Schwarz's Bayesian information criterion (SBC). AIC or SBC can be used for model selection. For a set of candidate models, the model with the smallest AIC or SBC is often preferred.

Figure 31.2 Fit Summary Statistics for a Linear Model



The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Model Fit Summary	
Dependent Variable	crime
Number of Observations	49
Data Set	WORK.CRIMEOH
Model	Linear
Log Likelihood	-187.37709
Maximum Absolute Gradient	7.59852E-7
Number of Iterations	16
Optimization Method	Newton-Raphson
AIC	382.75418
SBC	390.32146

The parameter estimates of the model and their standard errors are shown in Figure 31.3. Based on the *p*-values, both INCOME and HVALUE are significant at the 0.05 level.

Figure 31.3 Parameter Estimates of the Linear Model

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	68.618863	4.588210	14.96	<.0001
income	1	-1.597304	0.323739	-4.93	<.0001
hvalue	1	-0.273931	0.099989	-2.74	0.0062
_sigma2	1	122.751696	24.799493	4.95	<.0001

The following statements fit a SAR model to the CRIMEOH data set:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime = income hvalue / type=SAR;
run;
```

The NONORMALIZE option requests that the spatial weights matrix that is specified in the CRIMEWMAT data set be used "as is" rather than be row-standardized. The "Model Fit Summary" table, shown in Figure 31.4, lists several fit summary statistics about the SAR model. For this model, the value of AIC is about 374.78—smaller than 382.75, which is the AIC value for the preceding linear model. Based on AIC, the SAR model is preferred.

Figure 31.4 Fit Summary Statistics for a SAR Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Model Fit Summary

Dependent Variable	crime
Number of Observations	49
Data Set	WORK.CRIMEOH
Spatial Weights	WORK.CRIMEWMAT
Model	SAR
Log Likelihood	-182.38860
Maximum Absolute Gradient	2.7871E-7
Number of Iterations	16
Optimization Method	Newton-Raphson
AIC	374.77720
SBC	384.23630

The parameter estimates of the SAR model and their standard errors are shown in Figure 31.5. According to the p-values, both INCOME and HVALUE are significant at the 0.05 level. In addition, the spatial autoregressive coefficient ρ is estimated to be about 0.431, with a p-value of 0.0005.

Figure 31.5 Parameter Estimates of the SAR Model

Parameter Estimates					
Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	45.077070	7.870590	5.73	<.0001
income	1	-1.031531	0.328403	-3.14	0.0017
hvalue	1	-0.265924	0.088218	-3.01	0.0026
_rho	1	0.431020	0.123594	3.49	0.0005
_sigma2	1	95.487066	19.506312	4.90	<.0001

The following statements fit an SDM model. Unlike the previous SAR model, SDM accounts for exogenous interaction effects by introducing spatial lags of two explanatory variables—INCOME and HVALUE.

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime = income hvalue / type=SAR;
   spatialeffects income hvalue;
run;
```

The fit summary statistics for the SDM model are shown in Figure 31.6. Parameter estimates are provided in Figure 31.7.

Figure 31.6 Fit Summary Statistics for the SDM Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Model Fit Summary	
Dependent Variable	crime
Number of Observations	49
Data Set	WORK.CRIMEOH
Spatial Weights	WORK.CRIMEWMAT
Model	SDM
Log Likelihood	-181.39141
Maximum Absolute Gradient	5.44803E-8
Number of Iterations	16
Optimization Method	Newton-Raphson
AIC	376.78282
SBC	390.02556

Figure 31.7 Parameter Estimates for the SDM Model

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	42.803457	13.924487	3.07	0.0021
income	1	-0.914206	0.336439	-2.72	0.0066
hvalue	1	-0.293745	0.088857	-3.31	0.0009
W_income	1	-0.519640	0.594772	-0.87	0.3823
W_hvalue	1	0.245716	0.176854	1.39	0.1647
_rho	1	0.426492	0.167492	2.55	0.0109
_sigma2	1	91.779519	18.909222	4.85	<.0001

The spatial autoregressive coefficient ρ is estimated to be 0.426 with a p-value of 0.0109 based on an asymptotic t test. This result seems to suggest that there is a significantly positive spatial dependence in the number of crimes.

In the SPATIALREG procedure, the null hypothesis $H_0 : \rho = 0$ can also be tested against the alternative $H_a : \rho \neq 0$ by using the likelihood ratio (LR) test, Lagrange multiplier (LM) test, and Wald test. For the LR test, the test statistic is equal to $-2(\mathcal{L}_{\mathrm{linear}} - \mathcal{L}_{\mathrm{SAR}}) = -2(-187.38 + 182.39) = 9.98$, where $\mathcal{L}_{\mathrm{linear}}$ and $\mathcal{L}_{\mathrm{SAR}}$ are the log likelihoods for the linear regression model and SAR model, respectively. The likelihood ratio test is significant at the 0.05 level, providing strong evidence of spatial dependence in the data.

Syntax: SPATIALREG Procedure

The following statements are available in the SPATIALREG procedure:

```
PROC SPATIALREG < options > ;
    BOUNDS bound1 < , bound2 . . . > ;
    BY variables ;
    CLASS variables ;
    INIT initvalue1 < , initvalue2 . . . > ;
    MODEL dependent = < regressors >< / options > ;
    NLOPTIONS < options > ;
    OUTPUT < OUT=SAS-data-set >< output-options > ;
    PERFORMANCE options ;
    RESTRICT restriction1 < , restriction2 . . . > ;
    TEST equation1 < , equation2 . . . >< /test-options > ;
    SPATIALEFFECTS < model-spatial-effect-regressors > ;
    SPATIALID variable ;
```

You can specify more than one MODEL statement, as shown in the section "Example 31.3: Fitting Multiple Models" on page 2352. The CLASS statement must precede the MODEL statement. If you include the SPATIALEFFECTS statement, it must be paired with and appear after the MODEL statement.

Functional Summary

Table 31.1 summarizes the statements and options that you can use with the SPATIALREG procedure.

Table 31.1 Functional Summary

Description	Statement	Option
Data Set Options		
Specifies the input primary data set	PROC SPATIALREG	DATA=
Specifies the input spatial weights data set	PROC SPATIALREG	WMAT=
Suppresses normalization of the spatial weights	PROC SPATIALREG	NONORMALIZE
Writes parameter estimates to an output data set	PROC SPATIALREG	OUTEST=
Writes estimates of $\mathbf{x}_i'\boldsymbol{\beta}$, predicted values, and residuals to an output data set	OUTPUT	OUT=
Approximation Control Options		
Specifies the approximation-related options	PROC SPATIALREG	APPROXIMATION=
Declaring the Role of Variables		
Specifies BY-group processing	BY	
Specifies classification variables	CLASS	
Specifies a spatial ID variable	SPATIALID	
Printing Control Options		
Prints the correlation matrix of the estimates	MODEL	CORRB
Prints the covariance matrix of the estimates	MODEL	COVB
Prints a summary iteration listing	MODEL	ITPRINT
Suppresses the normal printed output	PROC SPATIALREG	NOPRINT
Prints all available output	MODEL	PRINTALL
Optimization Process Control Options		
Specifies maximum number of iterations allowed	MODEL	MAXITER=
Selects the iterative minimization method to use	PROC SPATIALREG	METHOD=
Sets boundary restrictions on parameters	BOUNDS	
Sets initial values for parameters	INIT	
Sets linear restrictions on parameters	RESTRICT	
Sets the number of threads to use	PERFORMANCE	NTHREADS=
Specifies the optimization options	NLOPTIONS	See Chapter 6, "Nonlinear Optimization Methods."
Model Estimation Options		
Specifies the spatial lag of covariate effect	SPATIALEFFECTS	
Specifies the type of model	MODEL	TYPE=
Specifies the type of covariance matrix	MODEL	COVEST=
Suppresses the intercept parameter	MODEL	NOINT

Table 31.1 *continued*

Description	Statement	Option
Output Control Options		
Includes covariances in the OUTEST= data set	PROC SPATIALREG	COVOUT
Outputs the residual	OUTPUT	RESID=
Outputs the expected value of the response variable	OUTPUT	PRED=
Outputs estimates of $\mathbf{x}_i'\boldsymbol{\beta}$	OUTPUT	XBETA=

PROC SPATIALREG Statement

PROC SPATIALREG < *options* > **;**

You can specify the following options in the PROC SPATIALREG statement.

Data Set Options

DATA=*SAS-data-set*

specifies the primary SAS data set that contains dependent variables, and explanatory variables, and so on.

WMAT=*SAS-data-set*

specifies the secondary spatial weights data set, which can be used to construct the spatial weights matrix $\mathbf{W}$. Loosely speaking, the entries of $\mathbf{W}$, $w(\mathrm{s}_i, \mathrm{s}_j)$, define the amount of influence that a unit s_j has over a unit s_i. The entries $w(\mathrm{s}_i, \mathrm{s}_j)$ must be nonnegative and have zeros on the diagonal; that is, $w(\mathrm{s}_i, \mathrm{s}_j) \geq 0$ and $w(\mathrm{s}_i, \mathrm{s}_i) = 0$, where $i, j = 1, 2, \ldots, n$, with n being the total number of spatial units in the data. Any nonzero diagonal elements $w(\mathrm{s}_i, \mathrm{s}_i)$ are replaced with 0. The spatial weights matrix can be asymmetric; that is, it is not necessary that $w(\mathrm{s}_i, \mathrm{s}_j) = w(\mathrm{s}_j, \mathrm{s}_i)$. For information about missing spatial weights in $\mathbf{W}$, see the section “NONORMALIZE” on page 2306.

The $\mathbf{W}$ matrix can take two different forms. First, you can provide a full spatial weights matrix. In this case, the data set that you specify in the WMAT= option has n rows. However, the number of columns can be either $n + 1$ or n, depending on whether you need a spatial ID variable to match observations in two data sets that are specified by the DATA= option and WMAT= option. If you need a SPATIALID statement to specify a spatial ID variable for the purpose of matching observations, the data set that you specify in the WMAT= option needs to have *n+1* columns. In this case, the spatial ID variable can appear in any column in the data set. Otherwise, the number of columns in the data set that you specify in the WMAT= option should be n.

Second, you can also specify the spatial weights matrix by using a compact form when appropriate. In this form, the number of observations in the data set that you specify in the WMAT= option should match the number of nonzero elements in the spatial weights matrix. Moreover, the number of columns in this data set should be three. The first two columns give the row and column indices for nonzero entries in the spatial weights matrix. The third column in the data set contains the nonzero entries in the spatial weights matrix. If you use the compact form for the spatial weights matrix, you must include a SPATIALID statement to match observations in the two data sets that are specified in the DATA=

option and WMAT= option. For more information about the SPATIALID statement, see the section “SPATIALID Statement” on page 2314. For more information about the compact representation of the spatial weights matrix, see the section “Compact Representation of Spatial Weights Matrix” on page 2329.

NONORMALIZE

suppresses the row standardization of the spatial weights matrix that is specified in the WMAT= option. By default, the spatial weights matrix is row-standardized; that is, the spatial weights matrix has unit row sum. If the NONORMALIZE option is specified, spatial weights are used “as is” except for $w(\mathrm{s}_i, \mathrm{s}_i)$, which is always treated as 0. This implies that an entry $w(\mathrm{s}_i, \mathrm{s}_j)$ in the $\mathbf{W}$ matrix cannot be missing for $i \neq j$ if the NONORMALIZE option is specified. If this option is not specified, missing spatial weights are replaced with zeros.

Approximation Control Options

For the SAR, SDM, SEM, and SDEM models, you can specify the following *options*:

APPROXIMATION=(< ***approx-options*** **>)**

specifies options that are related to approximating the Jacobian, as described in the section “Approximations to the Jacobian” on page 2332. You must specify one or more of the following *approx-options*:

CHEBYSHEV | TAYLOR

specifies the approximation method. By default, Chebyshev approximation is used. The Taylor approximation is used only if you specify the TAYLOR option.

NMC=*number*

specifies a positive integer as the number of standard random normal draws for the Monte Carlo simulation that approximates the traces of powers of the spatial weights matrix $\mathbf{W}$. If the SEED= option is specified, NMC=100 by default. If neither the NMC= option nor the SEED= option is specified, Monte Carlo simulation is not used and the traces of powers of the spatial weights matrix $\mathbf{W}$ are computed exactly. For more information, see the section “Approximations to the Jacobian” on page 2332.

ORDER=*number*

specifies a positive integer as the order of series in Taylor approximation or Chebyshev approximation. If Taylor approximation is used, ORDER=50 by default. If Chebyshev approximation is used, ORDER=5 by default.

SEED=*number*

specifies an integer seed in the range 1 to $2^{31} - 1$ for the random number generator that is used for the Monte Carlo simulation that approximates the traces of powers of the spatial weights matrix $\mathbf{W}$. If the NMC= option is specified, SEED=1 by default. If neither the NMC= option nor the SEED= option is specified, Monte Carlo simulation is not used and the traces of powers of the spatial weights matrix $\mathbf{W}$ are computed exactly. For more information, see the section “Approximations to the Jacobian” on page 2332. Specifying a seed enables you to reproduce your analysis.

Output Data Set Options

COVOUT

writes the covariance matrix for the parameter estimates to the OUTEST= data set. This option is valid when you specify the OUTEST= option.

OUTEST=*SAS-data-set*

writes the parameter estimates to the specified output data set.

Printing Options

CORRB

prints the correlation matrix of the parameter estimates. You can also specify this option in the MODEL statement.

COVB

prints the covariance matrix of the parameter estimates. You can also specify this option in the MODEL statement.

NOPRINT

suppresses all printed output.

Estimation Control Options

COVEST=HESSIAN | OP | QML

specifies the type of covariance matrix for the parameter estimates. You can specify the following types:

HESSIAN	specifies the covariance from the Hessian matrix.
OP	specifies the covariance from the outer product matrix.
QML	specifies the covariance from the outer product and Hessian matrices.

By default, COVEST=HESSIAN. The quasi-maximum-likelihood estimates are computed using COVEST=QML. For all models except the linear and SLX models, only COVEST=HESSIAN is supported.

Optimization Process Control Options

PROC SPATIALREG uses the nonlinear optimization (NLO) subsystem to perform nonlinear optimization tasks. All the NLO options are available in the NLOPTIONS statement. For more information, see the section "NLOPTIONS Statement" on page 2312. In addition, you can specify the following option in the PROC SPATIALREG statement:

METHOD=CONGRA | DBLDOG | NEWRAP | NMSIMP | NONE | NRRIDG | QUANEW | TRUREG

specifies the iterative minimization method to use. You can specify the following *values*:

CONGRA	specifies the conjugate-gradient method.
DBLDOG	specifies the double-dogleg method.
NEWRAP	specifies the Newton-Raphson method.

NMSIMP specifies the Nelder-Mead simplex method.

NONE specifies that optimization not be performed.

NRRIDG specifies the Newton-Raphson ridge method.

QUANEW specifies the quasi-Newton method.

TRUREG specifies the trust region method.

By default, METHOD=NEWRAP.

BOUNDS Statement

BOUNDS *bound1* < *, bound2 . . .* > **;**

The BOUNDS statement imposes simple boundary constraints on the parameter estimates. You can specify any number of BOUNDS statements.

Each *bound* is composed of parameter names, constants, and inequality operators as follows:

item operator item < *operator item operator item . . .* >

Each *item* can be a constant, a parameter name, or a list of parameter names. Each *operator* can be <, >, <=, or >=.

You can use both the BOUNDS statement and the RESTRICT statement to impose boundary constraints; however, the BOUNDS statement provides a simpler syntax for specifying these kinds of constraints. For more information about the RESTRICT statement, see the section "RESTRICT Statement" on page 2313.

The following BOUNDS statement constrains the estimates of the parameter for z to be negative, the parameters for x1 through x10 to be between 0 and 1, and the parameter for spatial lag of the x1 to be less than 1:

```
bounds z < 0,
       0 < x1-x10 < 1,
       W_x1 < 1;
```

BY Statement

BY *variables* **;**

A BY statement can be used in PROC SPATIALREG to obtain separate analyses of observations in groups that are defined by the BY variables. When you use a BY statement, the primary input data set (specified in the DATA= option) should be sorted by the BY variables.

CLASS Statement

CLASS *variable* < *(options)* > . . . < *variable* < *(options)* > > < */global-options* > **;**

The CLASS statement names the classification variables that are used to group (classify) data in the analysis. Classification variables can be either character or numeric.

Class levels are determined from the formatted values of the CLASS *variables*. Thus, you can use formats to group values into levels. For more information, see the discussion of the FORMAT procedure in *SAS Language Reference: Dictionary*. The CLASS statement must precede the MODEL statement.

Most options can be specified as either individual variable *options* or *global-options*. You can specify *options* for each variable by enclosing the options in parentheses after the *variable* name. You can also specify *global-options* for the CLASS statement by placing them after a slash (/). *Global-options* are applied to all the variables that are specified in the CLASS statement. If you specify more than one CLASS statement, the *global-options* that are specified in any one CLASS statement apply to all CLASS statements. However, individual CLASS variable *options* override the *global-options*.

You can specify the following values for either an *option* or a *global-option*:

MISSING

treats missing values (., ._, .A, . . . , .Z for numeric variables and blanks for character variables) as valid levels for the CLASS variable.

ORDER=DATA | FORMATTED | FREQ | INTERNAL

specifies the sort order for the levels of classification variables. This ordering determines which parameters in the model correspond to each level in the data, so the ORDER= option can be useful when you use the CONTRAST statement. By default, ORDER=FORMATTED. For ORDER=FORMATTED and ORDER=INTERNAL, the sort order is machine-dependent. When ORDER=FORMATTED is in effect for numeric variables for which you have supplied no explicit format, the levels are ordered by their internal values.

The following table shows how PROC SPATIALREG interprets values of the ORDER= option:

Value of ORDER=	Levels Sorted By
DATA	Order of appearance in the input data set
FORMATTED	External formatted values, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) values
INTERNAL	Unformatted value

For more information about sort order, see the chapter on the SORT procedure in *Base SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

PARAM=*keyword*

specifies the parameterization method for the classification variable or variables. You can specify any of the *keywords* shown in the following table; by default, PARAM=GLM.

Design matrix columns are created from CLASS variables according to the corresponding coding schemes:

Value of PARAM=	Coding
EFFECT	Effect coding
GLM	Less-than-full-rank reference cell coding (this *keyword* can be used only as a *global-option*)
REFERENCE **REF**	Reference cell coding

All parameterizations are full rank, except for the GLM parameterization. The REF= option in the CLASS statement determines the reference level for effect and reference coding and for their orthogonal parameterizations. It also indirectly determines the reference level for a singular GLM parameterization through the order of levels.

REF='*level*' | *keyword*

specifies the reference level for PARAM=EFFECT, PARAM=REFERENCE, and their orthogonalizations. When PARAM=GLM, the REF= option specifies a level of the classification variable to be put at the end of the list of levels. This level thus corresponds to the reference level in the usual interpretation of the linear estimates with a singular parameterization.

For an individual variable REF= option (but not for a global REF= option), you can specify the *level* of the variable to use as the reference level. Specify the formatted value of the variable if a format is assigned. For a global or individual variable REF= option, you can specify one of the following *keywords*:

FIRST designates the first ordered level as reference.

LAST designates the last ordered level as reference.

By default, REF=LAST.

INIT Statement

INIT *initvalue1* < , *initvalue2* . . . > **;**

The INIT statement sets initial values for parameters in the optimization.

Each *initvalue* is written as a parameter or parameter list, followed by an optional equal sign (=), followed by a number:

parameter < = > *number*

For continuous regressors, the names of the parameters are the same as the corresponding variables. For a regressor that is a CLASS variable, the parameter name combines the corresponding CLASS variable name with the variable level. For interaction and nested regressors, the parameter names combine the names of all the regressors. The names of the parameters can be seen in the OUTEST= data set. By default, initial values are determined by OLS regression. Initial values can be displayed by using the ITPRINT option in the PROC SPATIALREG statement.

MODEL Statement

MODEL *dependent-variable = <regressors> </ options>* **;**

The MODEL statement specifies the *dependent-variable* and independent covariates (*regressors*) for the regression model. If you specify no *regressors*, PROC SPATIALREG fits a model that contains only an intercept. The *dependent-variable* is treated as a continuous variable in the primary input data set (specified in the DATA= option). Models in PROC SPATIALREG do not allow missing values. If there are missing values, you get an error message.

You can specify more than one MODEL statement. You can specify the following *options* in the MODEL statement after a slash (/):

NOINT

suppresses the intercept parameter.

TYPE=LINEAR | SAC | SAR | SARMA | SEM | SMA

specifies the type of model to be fitted. If you specify this option in both the MODEL statement and the PROC SPATIALREG statement, the MODEL statement overrides the PROC SPATIALREG statement. You can specify the following model types:

LINEAR	specifies the linear model.
SAC	specifies the spatial autoregressive confused model.
SAR	specifies the spatial autoregressive model.
SARMA	specifies the spatial autoregressive moving average model.
SEM	specifies the spatial error model.
SMA	specifies the spatial moving average model.

By default, TYPE=SAR.

Printing Options

CORRB

prints the correlation matrix of the parameter estimates. You can also specify this option in the PROC SPATIALREG statement.

COVB

prints the covariance matrix of the parameter estimates. You can also specify this option in the PROC SPATIALREG statement.

ITPRINT

prints the objective function and parameter estimates at each iteration. The objective function is the negative log-likelihood function. You can also specify this option in the PROC SPATIALREG statement.

PRINTALL

requests all available output. You can also specify this option in the PROC SPATIALREG statement.

NLOPTIONS Statement

NLOPTIONS < *options* > **;**

The NLOPTIONS statement provides the options to control the nonlinear optimization (NLO) subsystem to perform nonlinear optimization tasks. For a list of all the options available in the NLOPTIONS statement, see Chapter 6, "Nonlinear Optimization Methods."

OUTPUT Statement

OUTPUT < **OUT=***SAS-data-set* > < *output-options* > **;**

The OUTPUT statement creates a new SAS data set that contains all the variables in the input data set and, optionally, the estimates of $\mathbf{x}_i'\boldsymbol{\beta}$, the expected value of the response variable, and the residual.

You can specify only one OUTPUT statement for each MODEL statement. You can specify the following *output-options*:

OUT=*SAS-data-set*

names the output data set.

XBETA=*name*

names the variable that contains estimates of $\mathbf{x}_i'\boldsymbol{\beta}$.

PRED=*name*

MEAN=*name*

assigns a name to the variable that contains the predicted value of the response variable.

RESID=*name*

RESIDUAL=*name*

assigns a name to the variable that contains the residuals (that is, the difference between the observed and predicted values of the response variable).

PERFORMANCE Statement

PERFORMANCE < *performance-options* > **;**

The PERFORMANCE statement controls the number of threads that are used in the optimization phase. You can also specify that multithreading not be used in the optimization phase by using the NOTHREADS option.

You can specify only one PERFORMANCE statement. You can specify the following *performance-options*:

DETAILS
specifies that a timing table be included in the output.

NOTHREADS
specifies that no threads be used during optimization.

NTHREADS=*number*
specifies the number of threads to be used during optimization.

If you use both the NTHREADS= and NOTHREADS options, then the NTHREADS= option is ignored. If you use a PERFORMANCE statement, then it overrides any global threading settings that might have been set using the CPUCOUNT=, THREADS, or NOTHREADS system option.

RESTRICT Statement

RESTRICT *restriction1 < , restriction2 . . . >* **;**

The RESTRICT statement imposes linear restrictions on the parameter estimates. You can specify any number of RESTRICT statements.

Each *restriction* is written as an expression, followed by an equality operator (=) or an inequality operator (<, >, <=, >=), followed by a second expression:

parameter < number

Restriction expressions can be composed of parameter names; constants; and the operators times ($*$), plus ($+$), and minus ($-$). The restriction expressions must be a linear function of the parameters. For continuous regressors, the names of the parameters are the same as the names of the corresponding variables. For a regressor that is a CLASS variable, the parameter name combines the corresponding CLASS variable name with the variable level. For interaction and nested regressors, the parameter names combine the names of all the regressors. The names of the parameters can be seen in the OUTEST= data set.

Lagrange multipliers are reported in the “Parameter Estimates” table for all the active linear constraints. They are identified by the names Restrict1, Restrict2, and so on. The p-values of these Lagrange multipliers are computed using a beta distribution (LaMotte 1994). Nonactive (nonbinding) restrictions have no effect on the estimation results and are not noted in the output.

For example, the following RESTRICT statement constrains the spatial autoregressive coefficient ρ to 0, which removes endogenous interaction effects:

```
restrict _rho = 0;
```

TEST Statement

<'label':> **TEST** *equation [,equation. . .] < / options>* **;**

The TEST statement performs Wald, Lagrange multiplier, and likelihood ratio tests of linear hypotheses about the parameters in your model. Each equation specifies a linear hypothesis to be tested. All hypotheses in one TEST statement are tested jointly. Variable names in the equations must correspond to regressors in the preceding MODEL statement, and each name represents the coefficient of the corresponding regressor.

The keyword INTERCEPT refers to the coefficient of the intercept. The keywords _rho and _lambda refer to the autoregressive coefficients ρ and λ, respectively. In addition, the keyword _sigma2 refers to the variance parameter σ^2.

You can specify the following options after the slash (/):

ALL

requests Wald, Lagrange multiplier, and likelihood ratio tests.

LM

requests the Lagrange multiplier test.

LR

requests the likelihood ratio test.

WALD

requests the Wald test.

The following statements illustrate the use of the TEST statement:

```
proc spatialreg data=dat;
   model y = x1 x2 x3/type=LINEAR;
   test x1 = 0, x2 * .5 + 2 * x3 = 0/ALL;
   test_int: test intercept = 0, x3 = 0/LR;
run;
```

The first test investigates the joint hypothesis that $\beta_1 = 0$ and $0.5\beta_2 + 2\beta_3 = 0$.

Only linear equality tests are permitted in PROC SPATIALREG. Tests expressions can be composed only of algebraic operations that use the addition symbol (+), subtraction symbol (–), and multiplication symbol (*).

The TEST statement accepts labels that are reproduced in the printed output. TEST statements can be labeled in two ways: a TEST statement can be preceded by a label followed by a colon, or the keyword TEST can be followed by a quoted string. If both are present, PROC SPATIALREG uses the label preceding the colon. If no label is specified, PROC SPATIALREG automatically labels the tests.

SPATIALID Statement

SPATIALID *variable* **;**

For models that require a spatial weights matrix, the SPATIALID statement specifies a variable that identifies a spatial unit for each observation in the two data sets that are specified in the DATA= option and WMAT= option in the PROC SPATIALREG statement. The variable that is specified in the SPATIALID statement is also used to match the rows and columns within the WMAT= data set. You do not need a SPATIALID statement if no matching is needed for the two data sets specified in the DATA= option and WMAT= option. If you do need a SPATIALID statement, only one SPATIALID statement and one spatial ID variable are allowed. The values of the spatial ID variable in either the DATA= data set or the WMAT= data set cannot be missing.

The variable in the SPATIALID statement can be either numeric or character. However, the type of spatial ID variable in the two data sets specified in the DATA= option and WMAT= option must be the same. When the spatial ID variable is numeric, it needs to be integer-valued. If you specify a number that is not an integer, PROC SPATIALREG uses the integer part of that number for matching.

SPATIALEFFECTS Statement

SPATIALEFFECTS < *model-spatial-effect-regressors* > < */options* > **;**

The SPATIALEFFECTS statement enables you to specify covariates (such as X) whose spatial lag, WX, is to be added to the MODEL statement.

PROC SPATIALREG adds the spatially weighted *model-spatial-effect-regressors* to regressors that are specified in the MODEL statement. For example, if you specify q variables $z_1, \ldots, z_q$ in the SPATIALEFFECTS statement, then each of q spatially weighted variables, as represented by each column of $\mathbf{WZ}$, has a parameter to be included in the regression. Here, $\mathbf{WZ}$ denotes the matrix product of $\mathbf{W}$ and $\mathbf{Z}$. In addition, $\mathbf{Z}$ is the design matrix formed by the q variables $z_1, \ldots, z_q$. The spatial weights matrix $\mathbf{W}$ comes from the data set that is specified in the WMAT= option. The "Parameter Estimates" table in the displayed output shows the estimates for spatially weighted *model-spatial-effect-regressors*; they are labeled with the prefix "W_". For example, if you specify z (a variable in your primary data set) as a spatial effect explanatory variable, then the "Parameter Estimates" table labels the corresponding parameter estimate "W_z".

Details: SPATIALREG Procedure

Specification of Regressors

Each term in a model, called a *regressor*, is a variable or combination of variables. Regressors are specified in a special notation that uses variable names and operators. There are two kinds of variables: classification (CLASS) variables and continuous variables. There are two primary operators: crossing and nesting. A third operator, the bar operator, is used to simplify effect specification.

In the SAS System, classification variables are declared in the CLASS statement. (They can also be called categorical, qualitative, discrete, or nominal variables.) Classification variables can be either numeric or character. The values of a classification variable are called *levels*. For example, the classification variable Sex has the levels "male" and "female."

In a model, an independent variable that is not declared in the CLASS statement is assumed to be continuous. Continuous variables, which must be numeric, are used for covariates. For example, the heights and weights of subjects are continuous variables. A response variable is a continuous variable and must also be numeric.

Types of Regressors

Seven different types of regressors are used in the SPATIALREG procedure. In the following list, assume that A, B, C, D, and E are CLASS variables and that X1 and X2 are continuous variables:

- Regressors are specified by writing continuous variables by themselves: X1   X2.
- Polynomial regressors are specified by joining (crossing) two or more continuous variables with asterisks: X1*X1   X1*X2.
- Dummy regressors are specified by writing CLASS variables by themselves: A   B   C.

- Dummy interactions are specified by joining classification variables with asterisks: A*B B*C A*B*C.
- Nested regressors are specified by following a dummy variable or dummy interaction with a classification variable or list of classification variables enclosed in parentheses. The dummy variable or dummy interaction is nested within the regressor that is listed in parentheses: B(A) C(B*A) D*E(C*B*A). In this example, B(A) is read as "B nested within A."
- Continuous-by-class regressors are written by joining continuous variables and classification variables with asterisks: X1*A.
- Continuous-nesting-class regressors consist of continuous variables followed by a classification variable interaction enclosed in parentheses: X1(A) X1*X2(A*B).

An example of the general form of an effect that involves several variables is

X1*X2*A*B*C(D*E)

This example contains an interaction between continuous terms and classification terms that are nested within more than one classification variable. The continuous list comes first, followed by the dummy list, followed by the nesting list in parentheses. Note that asterisks can appear within the nested list but not immediately before the left parenthesis.

The MODEL statement uses these effects. Some examples of MODEL statements that use various kinds of effects are shown in the following table, where a, b, and c represent classification variables. The variables x and z are continuous.

Specification	**Type of Model**
`model y=x;`	Simple regression
`model y=x z;`	Multiple regression
`model y=x x*x;`	Polynomial (quadratic) regression
`model y=a;`	Regression with one classification variable
`model y=a b c;`	Regression with multiple classification variables
`model y=a b a*b;`	Regression with classification variables and their interactions
`model y=a b(a) c(b a);`	Regression with classification variables and their interactions
`model y=a x;`	Regression with both continuous and classification variables
`model y=a x(a);`	Separate-slopes regression
`model y=a x x*a;`	Homogeneity-of-slopes regression

Bar Operator

You can shorten the specification of a large factorial model by using the bar operator. For example, two ways of writing the model for a full three-way factorial model follow:

```
model Y = A B C   A*B A*C B*C   A*B*C;

model Y = A|B|C;
```

When the bar (|) is used, the right and left sides become effects, and the cross between them becomes an effect. Multiple bars are permitted. The expressions are expanded from left to right, using rules 2–4 from Searle (1971, p. 390).

- Multiple bars are evaluated from left to right. For example, A | B | C is evaluated as follows:

$$\begin{aligned} \text{A | B | C} &\rightarrow \{\text{ A | B }\}\text{ | C} \\ &\rightarrow \{\text{ A B A*B }\}\text{ | C} \\ &\rightarrow \text{A B A*B C A*C B*C A*B*C} \end{aligned}$$

- Crossed and nested groups of variables are combined. For example, A(B) | C(D) generates A*C(B D), among other terms.
- Duplicate variables are removed. For example, A(C) | B(C) generates A*B(C C), among other terms, and the extra C is removed.
- Effects are discarded if a variable occurs in both the crossed and nested parts of an effect. For example, A(B) | B(D E) generates A*B(B D E), but this effect is discarded immediately.

You can also specify the maximum number of variables involved in any effect that results from bar evaluation by specifying that maximum number, preceded by an @ sign, at the end of the bar effect. For example, the specification A | B | C@2 would result in only those effects that contain no more than two variables: in this case, A B A*B C A*C and B*C.

More examples of using the | and @ operators follow:

A | C(B)	is equivalent to	A C(B) A*C(B)
A(B) | C(B)	is equivalent to	A(B) C(B) A*C(B)
A(B) | B(D E)	is equivalent to	A(B) B(D E)
A | B(A) | C	is equivalent to	A B(A) C A*C B*C(A)
A | B(A) | C@2	is equivalent to	A B(A) C A*C
A | B | C | D@2	is equivalent to	A B A*B C A*C B*C D A*D B*D C*D
A*B(C*D)	is equivalent to	A*B(C D)

Missing Values

Missing values can occur in both the primary data set (specified in the DATA= option) and the secondary spatial weights data set (specified in the WMAT= option). PROC SPATIALREG does not allow missing values in the primary input data set. If any observation in the primary input data set has a missing value for one or more of the regressors or the dependent variable, PROC SPATIALREG will not fit the model. If any observation in the primary data set has a missing value for the spatial ID variable when you specify the SPATIALID statement, PROC SPATIALREG will not fit the model. In such cases, an error message is issued.

For the secondary spatial weights data set, a missing value for the spatial ID variable is not allowed when you specify the SPATIALID statement. Moreover, missing spatial weights are not allowed if you specify the NONORMALIZE option. In these cases, PROC SPATIALREG issues an error message and skips the model fitting.

Spatial Autoregressive Models

The spatial autoregressive (SAR) model is useful for incorporating the spatial dependence in the dependent variable—that is, the endogenous interaction effect. Let y_i denote the observation associated with a spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let an $n \times n$ matrix $\mathbf{W}$ with nonnegative elements be a spatial weights matrix. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded for the spatial unit s_i. The SAR model can be formulated as

$$y_i = \rho \sum_{j=1}^{n} W_{ij} y_j + \mathbf{x}_i' \boldsymbol{\beta} + \epsilon_i$$

where $i = 1, 2, \ldots, n$. Here ρ is the spatial autoregressive coefficient and $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector. Moreover, W_{ij} is the (i, j)th element of the matrix $\mathbf{W}$ subject to $W_{ii} = 0$. For the error term ϵ_i related to the spatial unit s_i, it is assumed that $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$.

The SAR model is often described in vector form as

$$\mathbf{y} = \rho \mathbf{W} \mathbf{y} + \mathbf{X} \boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$, $\mathbf{X}$ is an $n \times p$ matrix where each row consists of $\mathbf{x}_i'$, and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$.

The standard estimator for the SAR model is the maximum likelihood estimator (MLE). For the SAR model, the log-likelihood function is (Anselin 2001)

$$\mathcal{L} = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})}{2\sigma^2} + \ln|\mathbf{A}|$$

where $\mathbf{A} = \mathbf{I}_n - \rho \mathbf{W}$, with $\mathbf{I}_n$ being an $n \times n$ identity matrix. $|\mathbf{A}|$ denotes the determinant of $\mathbf{A}$.

The gradients can be derived as follows:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\beta}} = \frac{\mathbf{X}'(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \rho} = \frac{1}{\sigma^2} \mathbf{y}'\mathbf{W}'(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) - \mathrm{tr}(\mathbf{A}^{-1}\mathbf{W})$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})}{2\sigma^4}$$

For the $n \times n$ matrix $\mathbf{A}$, $\mathrm{tr}(\mathbf{A}) = \sum_{i=1}^{n} a_{ii}$, where a_{ii} is the ith diagonal element of $\mathbf{A}$.

A SAR model does not account for exogenous interaction effects. However, in practice, the value of the dependent variable y for a spatial unit might be affected by some independent exploratory variables of other spatial units as well. In such a case, you can use the SDM model instead.

Spatial Durbin Models

Unlike a SAR model, a spatial Durbin model (SDM) can account for exogenous interaction effects in addition to the endogenous interaction effects. Let y_i denote the observation associated with a spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}$ be an $n \times n$ spatial weights matrix of your choice. Further, assume that $\mathbf{x}_i$ is a $p \times 1$ vector that denotes values of p regressors recorded for the spatial unit s_i. Similarly, assume that $\mathbf{z}_i$ is a $q \times 1$ vector that denotes values of q regressors measured at unit s_i.

The SDM model can be described in vector form as (LeSage and Pace 2009)

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{Z}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$ with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix where each row consists of $\mathbf{x}_i'$, and $\mathbf{Z}$ is an $n \times q$ matrix where each row consists of $\mathbf{z}_i'$. In addition, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ are $p \times 1$ and $q \times 1$ parameter vectors, respectively.

By letting $\widetilde{\mathbf{X}} = [\mathbf{X} \ \ \mathbf{W}\mathbf{Z}]$ and $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}' \ \boldsymbol{\theta}')'$, you can rewrite the SDM model as

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}} + \boldsymbol{\epsilon}$$

The log-likelihood function for the SDM model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})'(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{2\sigma^2} + \ln|\mathbf{A}|$$

where $\mathbf{A} = \mathbf{I}_n - \rho\mathbf{W}$.

For the SDM model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}}} = \frac{\widetilde{\mathbf{X}}'(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \rho} = \frac{1}{\sigma^2}\mathbf{y}'\mathbf{W}'(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}}) - \mathrm{tr}(\mathbf{A}^{-1}\mathbf{W})$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})'(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{2\sigma^4}$$

Both the SAR model and the SDM model account for endogenous interaction effects. However, in some cases there might be an interaction among error terms. In such cases, you might consider a spatial error model, which addresses spatial interaction among error terms.

Spatial Error Models

The spatial error model (SEM) accounts for spatial dependence in the error terms rather than in the dependent variable. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}$ be an $n \times n$ spatial weights matrix. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit s_i.

The SEM model can be described in vector form by using the following two-stage formulation (LeSage and Pace 2009),

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix where each row consists of $\mathbf{x}_i'$. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector.

The log-likelihood function for the SEM model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{[\mathbf{B}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]'\,[\mathbf{B}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]}{2\sigma^2} + \ln|\mathbf{B}|$$

where $\mathbf{B} = \mathbf{I}_n - \lambda\mathbf{W}$.

For the SEM model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\beta}} = \frac{(\mathbf{B}\mathbf{X})'\,[\mathbf{B}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{1}{\sigma^2}[\mathbf{W}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]'\,[\mathbf{B}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})] - \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W})$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{[\mathbf{B}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]'\,[\mathbf{B}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]}{2\sigma^4}$$

In addition to the interaction effects among error terms, you might also want to include exogenous interaction effects in the model. In such cases, you need to consider a spatial Durbin error model (SDEM).

Spatial Durbin Error Models

The spatial Durbin error model (SDEM) accounts for spatial dependence among the error terms and the exogenous interaction effect. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}$ be an $n \times n$ spatial weights matrix. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit s_i. Similarly, let $\mathbf{z}_i$ be a $q \times 1$ vector that denotes values of q regressors measured at unit s_i.

The SDEM can be described in vector form by using the following two-stage formulation (LeSage and Pace 2009),

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{Z}\boldsymbol{\theta} + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ and $\mathbf{Z}$ are $n \times p$ and $n \times q$ matrices, where each row consists of $\mathbf{x}_i'$ and $\mathbf{z}_i'$, respectively. In addition, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ are $p \times 1$ and $q \times 1$ parameter vectors, respectively.

By letting $\widetilde{\mathbf{X}} = [\mathbf{X} \;\; \mathbf{W}\mathbf{Z}]$ and $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}' \; \boldsymbol{\theta}')'$, the SDEM model can be rewritten as

$$\mathbf{y} = \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}} + \mathbf{B}^{-1}\boldsymbol{\epsilon}$$

where $\mathbf{B} = \mathbf{I}_n - \lambda \mathbf{W}$.

The log-likelihood function for the SDEM model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\left[\mathbf{B}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^2} + \ln|\mathbf{B}|$$

For the SDEM model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}}} = \frac{(\mathbf{B}\widetilde{\mathbf{X}})'\left[\mathbf{B}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{1}{\sigma^2}\left[\mathbf{W}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}(\mathbf{y} - \widetilde{\mathbf{X}}\boldsymbol{\beta})\right] - \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W})$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{\left[\mathbf{B}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^4}$$

Spatial Moving Average Models

The spatial moving average (SMA) model accounts for spatial dependence among the error terms; thus it is similar to the SEM model but with a different autocorrelation structure. The SMA model is used for modeling local autocorrelation. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}$ be an $n \times n$ spatial weights matrix. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit s_i.

The SMA model can be described in vector form by using the following two-stage formulation,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

$$\mathbf{u} = (\mathbf{I}_n - \lambda\mathbf{W})\boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that has $\mathbf{x}_i'$ in each row, and $\mathbf{Z}$ is an $n \times q$ matrix that has of $\mathbf{z}_i'$in each row. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector.

The log-likelihood function for the SMA model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]'\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]}{2\sigma^2} - \ln|\mathbf{B}|$$

where $\mathbf{B} = \mathbf{I}_n - \lambda \mathbf{W}$.

For the SMA model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\beta}} = \frac{(\mathbf{B}^{-1}\mathbf{X})'\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = -\frac{1}{\sigma^2}\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]'\left[\mathbf{B}^{-1}\mathbf{W}\right]\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right] + \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W})$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]'\left[\mathbf{B}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]}{2\sigma^4}$$

Spatial Durbin Moving Average Models

The term spatial Durbin moving average (SDMA) model is used to refer to the SMA model with exogenous interaction effects. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}$ be an $n \times n$ spatial weights matrix. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit s_i. Similarly, let $\mathbf{z}_i$ be a $q \times 1$ vector that denotes values of q covariates measured at unit s_i.

The SDMA model can be described in vector form as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + (\mathbf{I}_n - \lambda\mathbf{W})\boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ and $\mathbf{Z}$ are $n \times p$ and $n \times q$ matrices, where each row consists of $\mathbf{x}_i'$ and $\mathbf{z}_i'$, respectively. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector and $\boldsymbol{\theta}$ is a $q \times 1$ parameter vector, respectively.

By letting $\widetilde{X} = [\mathbf{X}\ \mathbf{WZ}]$ and $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}'\ \boldsymbol{\theta}')'$, the SDMA model can be written as

$$\mathbf{y} = \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}} + \mathbf{B}\boldsymbol{\epsilon}$$

The log-likelihood function for the SDMA model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^2} - \ln|\mathbf{B}|$$

where $\mathbf{B} = \mathbf{I}_n - \lambda\mathbf{W}$ and $|\mathbf{B}|$ denotes the determinant of matrix $\mathbf{B}$.

For the SDMA model, the gradients are

$$\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{\beta}}} = \frac{(\mathbf{B}^{-1}\widetilde{\mathbf{X}})'\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{\sigma^2}$$

$$\frac{\partial\mathcal{L}}{\partial\lambda} = -\frac{1}{\sigma^2}\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}^{-1}\mathbf{W}\right]\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right] + \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W})$$

$$\frac{\partial\mathcal{L}}{\partial\sigma^2} = -\frac{n}{2\sigma^2} + \frac{\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}^{-1}(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^4}$$

Spatial Autoregressive Moving Average Models

The spatial autoregressive moving average (SARMA) model, like the SMA model, can account for spatial dependence among the error terms. In addition, the SARMA model enables you to account for spatial dependence in the dependent variable, as the SAR model does. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let the $n \times n$ matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ with nonnegative elements be two spatial weights matrices. In practice, $\mathbf{W}_1$ and $\mathbf{W}_2$ can be identical. Further, it is assumed that $\mathbf{x}_i$ is a $p \times 1$ vector that denotes values of p covariates recorded at unit s_i.

The SARMA model can be described in the vector form by using the following two-stage formulation (LeSage and Pace 2009),

$$\mathbf{y} = \rho\mathbf{W}_1\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

$$\mathbf{u} = (\mathbf{I}_n - \lambda \mathbf{W}_2)\boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \text{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that consists of $\mathbf{x}_i'$ in each row. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector, and $\mathbf{I}_n$ is an $n \times n$ identity matrix.

The log-likelihood function for the SARMA model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]'\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]}{2\sigma^2} + \ln|\mathbf{A}| - \ln|\mathbf{B}|$$

where $\mathbf{A} = \mathbf{I}_n - \rho\mathbf{W}_1$, $\mathbf{B} = \mathbf{I}_n - \lambda\mathbf{W}_2$ and $|\cdot|$ denotes the matrix determinant operator.

For the SARMA model, the gradients are

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\beta}} = \frac{(\mathbf{B}^{-1}\mathbf{X})'\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]}{\sigma^2}$$

$$\frac{\partial\mathcal{L}}{\partial\rho} = \frac{\left(\mathbf{B}^{-1}\mathbf{W}_1\mathbf{y}\right)'\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})}{\sigma^2} - \text{tr}(\mathbf{A}^{-1}\mathbf{W}_1)$$

$$\frac{\partial\mathcal{L}}{\partial\lambda} = -\frac{1}{\sigma^2}\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]'\left[\mathbf{B}^{-1}\mathbf{W}_2\right]\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right] + \text{tr}(\mathbf{B}^{-1}\mathbf{W}_2)$$

$$\frac{\partial\mathcal{L}}{\partial\sigma^2} = -\frac{n}{2\sigma^2} + \frac{\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]'\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\right]}{2\sigma^4}$$

Spatial Durbin Autoregressive Moving Average Models

You can also accommodate exogenous interaction effects in the SARMA model. The term spatial Durbin autoregressive moving average (SDARMA) model is used to refer to such an extension of the SARMA model. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}_1$ and $\mathbf{W}_2$ be two spatial weights matrices. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit s_i. Similarly, let $\mathbf{z}_i$ be a $q \times 1$ vector that denotes values of q regressors measured at unit s_i.

The SDARMA model can be described in vector form by using the following two-stage formulation,

$$\mathbf{y} = \rho\mathbf{W}_1\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{W}_1\mathbf{Z}\boldsymbol{\theta} + \mathbf{u}$$

$$\mathbf{u} = (\mathbf{I}_n - \lambda\mathbf{W}_2)\boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \text{N}(0, \sigma^2)'$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that has $\mathbf{x}_i'$ in each row, $\mathbf{Z}$ is an $n \times q$ matrix that has $\mathbf{z}_i'$ in each row. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector.

By letting $\widetilde{X} = [\mathbf{X} \ \ \mathbf{W}_1\mathbf{Z}]$ and $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}' \ \boldsymbol{\theta}')'$, the SDARMA model can be written as

$$\mathbf{y} = \rho\mathbf{W}_1\mathbf{y} + \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}} + (\mathbf{I}_n - \lambda\mathbf{W}_2)\boldsymbol{\epsilon}$$

The log-likelihood function for the SDARMA model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]'\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^2} + \ln|\mathbf{A}| - \ln|\mathbf{B}|$$

where $\mathbf{A} = \mathbf{I}_n - \rho \mathbf{W}_1$ and $\mathbf{B} = \mathbf{I}_n - \lambda \mathbf{W}_2$.

For the SDARMA model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}}} = \frac{(\mathbf{B}^{-1}\widetilde{\mathbf{X}})' \left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \rho} = \frac{\left(\mathbf{B}^{-1}\mathbf{W}_1\mathbf{y}\right)' \mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{\sigma^2} - \mathrm{tr}(\mathbf{A}^{-1}\mathbf{W}_1)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = -\frac{1}{\sigma^2} \left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]' \left[\mathbf{B}^{-1}\mathbf{W}_2\right] \left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right] + \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W}_2)$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{\left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]' \left[\mathbf{B}^{-1}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^4}$$

Spatial Autoregressive Confused Models

The spatial autoregressive confused (SAC) model, like the SARMA model, can accommodate spatial dependence in both the dependent variable and error terms. However, the covariance structure for the error terms in a SAC model is different from that of the SARMA model. Let y_i denote the observation associated with the spatial unit $\mathbf{s}_i$ for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}_1$ and $\mathbf{W}_2$ be two spatial weights matrices. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit $\mathbf{s}_i$.

The SAC model can be described in vector form by using the following two-stage formulation (LeSage and Pace 2009),

$$\mathbf{y} = \rho \mathbf{W}_1 \mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}_2 \boldsymbol{u} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)'$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that has $\mathbf{x}_i'$ in each row. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector.

The log-likelihood function for the SAC model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{[\mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]' [\mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]}{2\sigma^2} + \ln|\mathbf{A}| + \ln|\mathbf{B}|$$

where $\mathbf{A} = \mathbf{I}_n - \rho \mathbf{W}_1$ and $\mathbf{B} = \mathbf{I}_n - \lambda \mathbf{W}_2$.

For the SAC model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\beta}} = \frac{(\mathbf{B}\mathbf{X})' [\mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \rho} = \frac{(\mathbf{B}\mathbf{W}_1\mathbf{y})' \mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})}{\sigma^2} - \mathrm{tr}(\mathbf{A}^{-1}\mathbf{W}_1)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{1}{\sigma^2} [\mathbf{W}_2(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]' [\mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})] - \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W}_2)$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{[\mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]' [\mathbf{B}(\mathbf{A}\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]}{2\sigma^4}$$

Spatial Durbin Autoregressive Confused Models

The SAC model can be extended to account for exogenous interaction effects. The term spatial Durbin autoregressive confused (SDAC) model is used to refer to such an extension of the SAC model. Let y_i denote the observation associated with the spatial unit $\mathbf{s}_i$ for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}_1$ and $\mathbf{W}_2$ be two spatial weights matrices. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit $\mathbf{s}_i$. Similarly, assume that $\mathbf{z}_i$ is a $q \times 1$ vector that denotes values of q regressors measured at unit $\mathbf{s}_i$.

The SDAC model can be described in vector form by using the following two-stage formulation,

$$\mathbf{y} = \rho \mathbf{W}_1 \mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{W}_1 \mathbf{Z}\boldsymbol{\theta} + \mathbf{u}$$

$$\mathbf{u} = (\mathbf{I}_n - \lambda \mathbf{W}_2)^{-1}\boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that has $\mathbf{x}_i'$ in each row, $\mathbf{Z}$ is an $n \times q$ matrix that has $\mathbf{z}_i'$ in each row. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector.

By letting $\widetilde{X} = [\mathbf{X} \;\; \mathbf{W}_1\mathbf{Z}]$ and $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}' \; \boldsymbol{\theta}')'$, the SDAC model can be rewritten as

$$\mathbf{y} = \rho \mathbf{W}_1 \mathbf{y} + \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}} + (\mathbf{I}_n - \lambda \mathbf{W}_2)^{-1}\boldsymbol{\epsilon}$$

The log-likelihood function for the SDAC model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{\left[\mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]' \left[\mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^2} + \ln|\mathbf{A}| + \ln|\mathbf{B}|$$

For the SDAC model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}}} = \frac{(\mathbf{B}\widetilde{\mathbf{X}})' \left[\mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \rho} = \frac{(\mathbf{B}\mathbf{W}_1\mathbf{y})' \mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{\sigma^2} - \mathrm{tr}(\mathbf{A}^{-1}\mathbf{W}_1)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{1}{\sigma^2}\left[\mathbf{W}_2(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]' \left[\mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right] - \mathrm{tr}(\mathbf{B}^{-1}\mathbf{W}_2)$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{\left[\mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]' \left[\mathbf{B}(\mathbf{A}\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})\right]}{2\sigma^4}$$

Linear Regression Models

You can also fit a linear regression model in PROC SPATIALREG. In this case, let y_i denote the observation associated with the spatial unit $\mathbf{s}_i$ for $i = 1, 2, \ldots, n$. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit $\mathbf{s}_i$.

The linear regression model can be described in vector form as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that has $\mathbf{x}_i'$ in each row.

The log-likelihood function for the linear regression model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{2\sigma^2}$$

For the linear regression model, the gradients are

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\beta}} = \frac{\mathbf{X}'(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{\sigma^2}$$

$$\frac{\partial\mathcal{L}}{\partial\sigma^2} = -\frac{n}{2\sigma^2} + \frac{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{2\sigma^4}$$

The Hessians take the following forms:

$$\frac{\partial^2\mathcal{L}}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}'} = -\frac{\mathbf{X}'\mathbf{X}}{\sigma^2}$$

$$\frac{\partial^2\mathcal{L}}{\partial\boldsymbol{\beta}\partial\sigma^2} = -\frac{\mathbf{X}'(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{\sigma^4}$$

$$\frac{\partial^2\mathcal{L}}{\partial\sigma^4} = \frac{n}{2\sigma^4} - \frac{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{\sigma^6}$$

Spatial Lag of X Models

The spatial lag of X (SLX) model assumes no endogenous interaction effects or spatial dependence in the error terms. Instead, it incorporates only exogenous interaction effects into the linear regression model. Let y_i denote the observation associated with the spatial unit s_i for $i = 1, 2, \ldots, n$. For these spatial units, let $\mathbf{W}$ be a spatial weights matrix. Further, let $\mathbf{x}_i$ be a $p \times 1$ vector that denotes values of p regressors recorded at unit s_i. Similarly, let $\mathbf{z}_i$ be a $q \times 1$ vector that denotes values of q regressors measured at unit s_i.

The SLX model can be described in vector form as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{Z}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)'$ and $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)'$, with $\epsilon_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$. Moreover, $\mathbf{X}$ is an $n \times p$ matrix that has $\mathbf{x}_i'$ in each row, $\mathbf{Z}$ is an $n \times q$ matrix that has $\mathbf{z}_i'$ in each row. In addition, $\boldsymbol{\beta}$ is a $p \times 1$ parameter vector.

By letting $\widetilde{X} = [\mathbf{X} \;\; \mathbf{W}\mathbf{Z}]$ and $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}' \; \boldsymbol{\theta}')'$, the SLX model can be rewritten as

$$\mathbf{y} = \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}} + \boldsymbol{\epsilon}$$

The log-likelihood function for the SLX model is

$$\mathcal{L} = -\frac{n}{2}\ln(2\pi\sigma^2) - \frac{(\mathbf{y}-\widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})'(\mathbf{y}-\widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{2\sigma^2}$$

For the SLX model, the gradients are

$$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}}} = \frac{(\widetilde{\mathbf{X}})'(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{\sigma^2}$$

$$\frac{\partial \mathcal{L}}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})'(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{2\sigma^4}$$

The Hessians take the following forms:

$$\frac{\partial^2 \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}} \partial \widetilde{\boldsymbol{\beta}}'} = -\frac{\widetilde{\mathbf{X}}'\widetilde{\mathbf{X}}}{\sigma^2}$$

$$\frac{\partial^2 \mathcal{L}}{\partial \widetilde{\boldsymbol{\beta}} \partial \sigma^2} = -\frac{\widetilde{\mathbf{X}}'(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{\sigma^4}$$

$$\frac{\partial^2 \mathcal{L}}{\partial \sigma^4} = \frac{n}{2\sigma^4} - \frac{(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})'(\mathbf{y} - \widetilde{\mathbf{X}}\widetilde{\boldsymbol{\beta}})}{\sigma^6}$$

Specifying the Spatial Weights Matrix

The spatial weights matrix **W** plays a vital role in spatial econometric modeling. If you fit a purely linear model without SLX effects, you do not need a **W** matrix. For other types of models in PROC SPATIALREG, you need to provide a spatial weights matrix to fit the model. Although the creation of the **W** matrix is often problem-specific, there are some general guidelines to consider. Two common ways to create the **W** matrix are *k*-order binary contiguity matrices and *k*-nearest neighbor matrices (Elhorst 2013).

k-Order Binary Contiguity Matrices

You start with the spatial contiguity matrix **C**. In the case of the first-order neighbors ($k = 1$), a value of 1 for the (i, j)th entry in **C** indicates that the two units *i* and *j* are neighbors to each other, and 0 indicates otherwise. The neighbor relationship is often defined based on sharing of a common boundary. To generalize this, a *k*-order neighbor ($k \geq 2$) of a unit *i* can be any units whose neighbors are $(k - 1)$-order neighbors of unit *i*. In this sense, the two units *i* and *j* that are not first-order neighbors can still be second-order neighbors if unit *j* is the neighbor to a first-order neighbor of unit *i*.

As an example, a first-order binary contiguity matrix might look like the following:

$$\mathbf{C} = \left(\begin{array}{ccccc} \text{SID} & \text{L1} & \text{L2} & \text{L3} & \text{L4} \\ \text{L1} & 0 & 1 & 0 & 1 \\ \text{L2} & 1 & 0 & 0 & 0 \\ \text{L3} & 0 & 0 & 0 & 1 \\ \text{L4} & 1 & 0 & 1 & 0 \end{array}\right)$$

The diagonal elements of **C** are zeros because, in general, a unit is not considered to be a neighbor of itself. Moreover, the two units L2 and L4 are neighbors of L1; L2 has L1 as its only neighbor; L3 has L4 as its only neighbor; and L4 has L1 and L3 as its neighbors. You can create the spatial weights matrix **W** by row-standardizing the contiguity matrix **C**. To do so, you divide entries in each row of **C** by the sum of that row. The spatial weights matrix **W**, which is the row-standardized version of **C**, is as follows:

$$\mathbf{W} = \begin{pmatrix} \text{SID} & \text{L1} & \text{L2} & \text{L3} & \text{L4} \\ \text{L1} & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \text{L2} & 1 & 0 & 0 & 0 \\ \text{L3} & 0 & 0 & 0 & 1 \\ \text{L4} & \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{pmatrix}$$

k-Nearest Neighbor Matrices

You can create a spatial contiguity matrix based on a distance metric. Let d_{ij} denote the distance between the two units *i* and *j*, which might be the Euclidean distance between centroids of the two spatial units. Let $(\text{lon}_i, \text{lat}_i)$ and $(\text{lon}_j, \text{lat}_j)$ be the centroids of units *i* and *j*, where $1 \leq i, j \leq n$, and lon and lat denote the longitude and latitude, respectively. Under the Euclidean distance metric, the distance d_{ij} between units *i* and *j* is

$$d_{ij} = \sqrt{(\text{lat}_i - \text{lat}_j)^2 + (\text{lon}_i - \text{lon}_j)^2}$$

After computing the distance between the unit *i* and other units under a certain metric, you sort d_{ij} in ascending order; for example, $d_{ij_1} \leq d_{ij_2} \leq \cdots \leq d_{ij_k} \leq \cdots \leq d_{ij_{n-1}}$. For a given *k*, let $N_k(i) = \{j_1, j_2, \ldots, j_k\}$ be the set that contains the indices of *k*-nearest neighbors of unit *i*; then the (i, j)th entries of the contiguity matrix **C** are defined as

$$C_{ij} = \begin{cases} 1 & \text{if } j \in N_k(i) \\ 0 & \text{otherwise} \end{cases}$$

The (i, j)th entry of the corresponding row-standardized matrix **W** is $W_{ij} = C_{ij} \left\{ \sum_{j \in N_k(i)} C_{ij} \right\}^{-1}$.

Unlike the *k*-order binary contiguity matrix, which is often symmetric by construction, *k*-nearest neighbor matrices can be asymmetric. To obtain a symmetric *k*-nearest neighbor matrices, you can define the (i, j)th entries of the contiguity matrix **C** as follows:

$$C_{ij} = \begin{cases} 1 & \text{if } j \in N_k(i) \text{ or } i \in N_k(j) \\ 0 & \text{otherwise} \end{cases}$$

In addition to the Euclidean distance measure, you can use other distance metrics as appropriate. A variant of *k*-nearest neighbor matrices $\mathbf{C}^*$ that is used in some empirical studies defines its (i, j)th entries as

$$C_{ij}^* = \begin{cases} 1 & \text{if } d_{ij} \leq d_{\text{cutoff}} \\ 0 & \text{otherwise} \end{cases}$$

where d_{cutoff} is a prespecified threshold distance.

In addition to the two constructions of spatial weights matrices that are presented earlier, see Elhorst (2013) and the references therein for more information about other ways to create a spatial weights matrix. In practice, you can define the neighbor relation that is problem-specific. For example, you can define two spatial units that are far apart to be neighbors because they share some attributes (such as population sizes larger than 500,000).

The data set that you specify in the WMAT= option is row-standardized by default to create a spatial weights matrix. This means that if you specify WMAT=C, PROC SPATIALREG row-standardizes the spatial contiguity matrix to create a spatial weights matrix. If you want to suppress row standardization, you must specify the NONORMALIZE option.

Compact Representation of Spatial Weights Matrix

When the number of spatial units n increases, the amount of memory that it takes to store n^2 entries of the spatial contiguity matrix $\mathbf{C}$ or the spatial weights matrix $\mathbf{W}$ increases dramatically. To circumvent the storage issue, PROC SPATIALREG enables you to provide a compact representation of $\mathbf{W}$ (or $\mathbf{C}$) when appropriate. With the compact matrix representation, you provide a data set that contains three variables by using the WMAT= option. The first two variables identify the row r and column c of $\mathbf{W}$ (or $\mathbf{C}$), and (r, c) can be expressed either as numerical indices or as values of the variable specified in the SPATIALID statement. The third variable contains the nonzero value of $\mathbf{W}$ (or $\mathbf{C}$) for row r and column c. With this compact representation, the number of observations in the data set specified in the WMAT= option equals the total number of nonzero entries in $\mathbf{W}$ (or $\mathbf{C}$).

You must use a SPATIALID statement if you want to use the compact representation of the spatial contiguity or spatial weights matrix. With the compact representation, the first two variables of the data set that you specify in the WMAT= option must be of the same type. First, the first two columns in that data set can be row and column index for each nonzero entry in $\mathbf{W}$ (or $\mathbf{C}$). In this case, the SPATIALID variable is numeric type. Alternatively, the first two columns in the WMAT= data set can be characters that are the names of two neighboring spatial units in $\mathbf{W}$ (or $\mathbf{C}$). In this second case, the SPATIALID variable is character type.

For example, the compact representation of the spatial weights matrix $\mathbf{W}$,

$$\mathbf{W} = \begin{pmatrix} 0 & 0.5 & 0 & 0.5 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0.5 & 0 & 0.5 & 0 \end{pmatrix}$$

would look like the following:

```
data Ws;
  input SID cSID Weight;
  datalines;
  1 2 0.5
  1 4 0.5
  2 1 1.0
  3 4 1.0
  4 1 0.5
  4 3 0.5
  ;
run;
```

For the spatial contiguity matrix $\mathbf{C}$,

$$\mathbf{C} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

the compact representation would look like the following:

```
data Cs;
  input SID cSID Weight;
  datalines;
  1 2 1.0
  1 4 1.0
  2 1 1.0
  3 4 1.0
  4 1 1.0
  4 3 1.0
  ;
run;
```

If the spatial weights matrix is the same as matrix **W** in the section "*k*-Order Binary Contiguity Matrices" on page 2327, its compact representation would be as follows:

```
data Ws2;
  input SID $ cSID $ Weight;
  datalines;
  L1 L2 0.5
  L1 L4 0.5
  L2 L1 1.0
  L3 L4 1.0
  L4 L1 0.5
  L4 L3 0.5
  ;
run;
```

If the spatial contiguity matrix is the same as matrix **C** in the section "*k*-Order Binary Contiguity Matrices" on page 2327, its compact representation can be given in the data set Cs2 as follows:

```
data Cs2;
  input SID $ cSID $ Weight;
  datalines;
  L1 L2 1.0
  L1 L4 1.0
  L2 L1 1.0
  L3 L4 1.0
  L4 L1 1.0
  L4 L3 1.0
  ;
run;
```

Spatial ID Matching

Depending on the type of model that you use in PROC SPATIALREG, you might need to specify two data sets: one in the DATA= option and the other in the WMAT= option. However, in some cases, these two data sets might not come in the same order in terms of spatial units. In such cases, you must use a SPATIALID statement to specify a spatial ID variable in order to match observations in these two data sets.

As an example, assume that the data set you specify in the DATA= option looks like the following:

```
data example;
  input SID $ x1 x2 y;
  datalines;
  L1  0.3  0.5  0.9
  L3 -0.7  0.8 -0.4
  L2  0.4 -1.2  0.6
  L8 -1.7  1.2 -0.5
  L4  1.4  0.9  0.3
  L5  2.3  1.5  1.9
  L7 -0.9 -0.8 -1.3
  L6  1.4 -1.6 -2.0
  ;
run;
```

Suppose the spatial contiguity matrix that you specify in the WMAT= option looks like the following:

```
data cmat;
  input SID $ L1 L8 L3 L4 L7 L6 L5 L2;
  datalines;
  L1  0 1 0 1 0 1 0 1
  L2  1 0 0 0 1 0 0 0
  L6  1 0 1 0 0 0 0 0
  L4  1 0 0 0 1 0 0 0
  L3  0 0 0 0 1 1 0 0
  L7  0 0 1 1 0 0 0 1
  L5  0 1 0 0 0 0 0 0
  L8  1 0 0 0 0 0 1 0
  ;
run;
```

As you can see, rows in the two data sets Example and Cmat do not share identically sorted SID values. The second row in the Example data set contains the observation for a spatial unit L3, and its neighbor information is given in the fifth row of the Cmat data set. Moreover, the rows and columns of the spatial weights data set Cmat are not in the same order. The following SAS statements fit a SAR model to these data:

```
proc spatialreg data=example Wmat=cmat;
  model y=x1 x2/type=SAR;
  spatialid SID;
run;
```

The SPATIALID statement enables you to match rows and columns of Cmat in addition to rows of Example and Cmat. Without the SPATIALID statement, you need to sort Cmat so that the order of its rows and columns matches that of Example. The sorted data set, Cmat2, would look like the following:

```
data cmat2;
  input SID $ L1 L3 L2 L8 L4 L5 L7 L6;
  datalines;
  L1  0 0 1 1 1 0 0 1
  L3  0 0 0 0 0 0 1 1
  L2  1 0 0 0 0 0 1 0
  L8  1 0 0 0 0 1 0 0
  L4  1 0 0 0 0 0 1 0
  L5  0 0 0 1 0 0 0 0
  L7  0 1 1 0 1 0 0 0
```

```
   L6  1 1 0 0 0 0 0 0
   ;
run;
```

Parameter Space of Autoregressive Parameters

For all models except linear regression models in PROC SPATIALREG, the autoregressive parameters ρ and λ are often assumed to satisfy some assumptions to ensure consistency of the maximum likelihood estimator (Elhorst 2013). For SAR and SDM models, the Jacobian term involves the log-determinant of $\mathbf{I} - \rho\mathbf{W}_1$, and the parameter space of ρ is often specified such that $\mathbf{I} - \rho\mathbf{W}_1$ is nonsingular. For SEM, SMA, SDM, and SDMA models, the Jacobian term involves the log-determinant of $\mathbf{I} - \lambda\mathbf{W}_1$, and the parameter space of λ is often specified such that $\mathbf{I} - \lambda\mathbf{W}_1$ is nonsingular. For SAC, SDAC, SARMA, and SDARMA models, the Jacobian term involves the log-determinants of both $\mathbf{I} - \rho\mathbf{W}_1$ and $\mathbf{I} - \lambda\mathbf{W}_1$. As a result, the parameter space of ρ and λ is often specified such that both $\mathbf{I} - \rho\mathbf{W}_1$ and $\mathbf{I} - \lambda\mathbf{W}_1$ are nonsingular.

In the SPATIALREG procedure, the parameter space of autoregressive parameters ρ and λ depends on the spatial weights matrix $\mathbf{W}$ that you choose. For $\mathbf{W}$, the parameter space of the autoregressive parameters ρ and λ in PROC SPATIALREG is determined as follows:

1. For a symmetric $\mathbf{W}$, the nonsingularity condition requires $\rho \in \left(\omega_{\text{min}}^{-1}, \omega_{\text{max}}^{-1}\right)$ and $\lambda \in \left(\omega_{\text{min}}^{-1}, \omega_{\text{max}}^{-1}\right)$. Here ω_{min} and ω_{max} denote the smallest (that is, most negative) and largest real eigenvalues of $\mathbf{W}$, respectively.

2. If $\mathbf{W}$ is symmetric and subsequently row-standardized, the nonsingularity condition requires $\rho \in \left({\omega^*_{\text{min}}}^{-1}, 1\right)$ and $\lambda \in \left({\omega^*_{\text{min}}}^{-1}, 1\right)$. Here ω^*_{min} denotes the smallest purely real eigenvalue of the row-standardized $\mathbf{W}$.

3. If $\mathbf{W}$ is asymmetric and subsequently row-standardized, the nonsingularity condition requires $\rho \in \left({r^*_{\text{min}}}^{-1}, 1\right)$ and $\lambda \in \left({r^*_{\text{min}}}^{-1}, 1\right)$. Here r^*_{min} denotes the smallest purely real eigenvalue of the row-standardized $\mathbf{W}$.

4. When Taylor approximation or Chebyshev approximation is used for SAR and SDM models, $\mathbf{W}$ is required to be row-standardized. In these cases, the restriction on the autoregressive coefficient ρ is $\rho \in (-1, 1)$.

Approximations to the Jacobian

In order for PROC SPATIALREG to obtain maximum likelihood estimates for all models except linear regression models, it needs to compute the Jacobian term because that term appears in the log-likelihood function. The Jacobian term is $\ln|\mathbf{I}_n - \rho\mathbf{W}|$ for SDMs and SAR models and $\ln|\mathbf{I}_n - \lambda\mathbf{W}|$ for SEMs and SDEMs, where n is the number of observations and $\mathbf{W}$ is the spatial weights matrix. When n is not large, the Jacobian is computed as

$$\ln|\mathbf{I}_n - \rho\mathbf{W}| = \sum_{i=1}^{n} \ln|1 - \rho\omega_i|$$

where ω_i are the eigenvalues of $\mathbf{W}$. This computation requires that all eigenvalues of $\mathbf{W}$ be precomputed, which works fine for small data sets. However, when n is large, computing the Jacobian term by using the eigenvalue method can be computationally infeasible. Instead, you can use approximations to the Jacobian.

The SPATIALREG procedure supports two different approximations to the Jacobian for SDMs, SEMs, SDEMs, and SAR models: Taylor approximation or Chebyshev approximation. Using SDMs and SAR models as an example, the two approximations for the Jacobian term can be described as follows (for more information, see LeSage and Pace 2009, and the references therein):

- Taylor approximation uses finite, lower-order series to approximate the log-determinant as

$$\ln|\mathbf{I}_n - \rho\mathbf{W}| \approx -\sum_{k=1}^{q} \frac{\rho^k \mathrm{tr}\left(\mathbf{W}^k\right)}{k}$$

- Chebyshev approximation uses finite, lower-order Chebyshev polynomials of the first kind to approximate the log-determinant as

$$\ln|\mathbf{I}_n - \rho\mathbf{W}| \approx \sum_{k=0}^{q} c_k(\rho)\mathrm{tr}\left(T_k(\mathbf{W})\right)$$

  where $T_0(\mathbf{W}) = \mathbf{I}_n$, $T_1(\mathbf{W}) = \mathbf{W}$, and $T_{k+1}(\mathbf{W}) = 2\mathbf{W}T_k(\mathbf{W}) - T_{k-1}(\mathbf{W})$ for $k = 1, 2, \ldots, q$. The coefficients $c_k(\rho)$ are defined as

$$c_k(\rho) = \begin{cases} \frac{1}{q+1}\sum_{j=0}^{q} \ln(1-\rho\cos\theta_j)\cos(k\theta_j) & \text{if } k = 0 \\ \frac{2}{q+1}\sum_{j=0}^{q} \ln(1-\rho\cos\theta_j)\cos(k\theta_j) & \text{if } k > 0 \end{cases}$$

  with $\theta_j = (j+1)\pi/(q+1)$ for $j = 0, 1, \ldots, q$.

The traces of powers of $\mathbf{W}$ can be computed exactly or approximated using Monte Carlo simulation. The Monte Carlo simulation is done as follows,

$$\mathrm{tr}\left(\mathbf{W}^k\right) \approx \frac{1}{M}\sum_{l=1}^{M} \frac{n}{\mathbf{u}_l'\mathbf{u}_l}\mathbf{u}_l'\mathbf{W}^k\mathbf{u}_l$$

where $\mathbf{u}_l \overset{\text{iid}}{\sim} \mathrm{N}(\mathbf{0}, \mathbf{I}_n)$ and M is the total number of Monte Carlo samples.

When you apply these two approximations, it is often assumed that the maximum eigenvalue of $\mathbf{W}$ equals 1 and the minimum eigenvalue of $\mathbf{W}$ is greater than or equal to –1 (see LeSage and Pace 2009, and the references therein). One way to satisfy this assumption is to use a row-standardized spatial weights matrix that is similar to a symmetric matrix. If the spatial weights matrix is not symmetric or similar to a symmetric matrix, it becomes more difficult to apply Chebyshev approximation and thus requires extra care (LeSage and Pace 2009).

When you request an approximation to the Jacobian, the choices that you need to make might include the approximation method to use (that is, Taylor approximation or Chebyshev approximation); the order of series q; and the number of Monte Carlo samples (that is, M). Your choice can be accommodated through the APPROXIMATION= option in the PROC SPATIALREG statement. For the approximation method, you can use the keyword TAYLOR in the APPROXIMATION= option to request Taylor approximation.

Otherwise, the approximation method defaults to Chebyshev approximation. You specify ORDER=q in the APPROXIMATION= option to request a series of order q when approximating the log-determinant. In addition, you specify NMC=M in the APPROXIMATION= option to request M Monte Carlo samples to be drawn when approximating the traces of powers of $\mathbf{W}$. In addition, you can use the SEED= suboption of the APPROXIMATION= option to specify an integer seed for a random number generator to replicate your analysis.

Parameter Naming Conventions for RESTRICT, TEST, BOUNDS, and INIT Statements

This section describes how you refer to the parameters when using either the RESTRICT, TEST, BOUNDS, or INIT statement. The examples are presented using the RESTRICT statement. However, the same remarks apply to referencing parameters when you use the TEST, BOUNDS, or INIT statement.

To impose a restriction on a parameter related to a regressor in the MODEL statement, you simply use the name of the regressor itself. Suppose your model is

```
model y = x1-x3 / type=SAR;
```

where x1-x3 are continuous variables. Suppose you want to restrict the parameter associated with the regressor x3 to be greater than 1.7. You should provide the following statement:

```
RESTRICT x3 > 1.7;
```

To impose a restriction on a parameter associated with a regressor in the SPATIALEFFECTS statement, you can form the name of the parameter by prefixing W_ to the name of the regressor. Suppose your MODEL and SPATIALEFFECTS statements are as follows:

```
model y = x1-x3 / type=SAR;
spatialeffects x1 x2 x3;
```

Suppose you want to restrict the parameter related to the x3 regressor in the SPATIALEFFECTS statement to be less than 1.0. You should refer to the parameter as W_x3 and provide the following statement:

```
RESTRICT W_x3 < 1.0;
```

Even though the regressor x3 appears in both the MODEL and SPATIALEFFECTS statements, the parameter associated with x3 in the MODEL statement is, of course, different from the parameter associated with x3 in the SPATIALEFFECTS statement. Thus, when the name of a regressor is used in a RESTRICT statement without any prefix, it refers to the parameter associated with that regressor in the MODEL statement. Meanwhile, when the name of a regressor is used in a RESTRICT statement with the prefix W_, it refers to the parameter associated with that regressor in the SPATIALEFFECTS statement. Note that the intercept is not included in the SPATIALEFFECTS statement.

Referring to Class Level Parameters

When your MODEL includes a CLASS variable, you can impose restrictions on the parameters associated with each of the levels related to that variable as described in this section.

Suppose your CLASS variable is named C and has three levels: 0, 1, 2. Suppose your model is the following:

```
class C;
model y = x1 x2 C;
```

Adding a CLASS variable as a regressor to your model introduces additional parameters to your model, each of which is associated with one of the levels of that variable. You can form the name of the parameter associated with a particular level of your CLASS variable by inserting the underscore character between the name of the variable and the value of the level. Thus, to restrict the parameter associated with level 0 of the CLASS variable C to always be greater than 0.7, you should refer to the parameter as C_0 and provide the following statement:

```
RESTRICT C_0 > 0.7;
```

When the value of a level is a negative number, you must replace the minus sign with an underscore when you form the name of the parameter associated with that particular level of the CLASS variable. For example, suppose your CLASS variable is named D and has four levels: –1, 0, 1, 2. Suppose your model is the following:

```
class D;
model y = x1 x2 D;
```

To restrict the parameter associated with level –1 of the CLASS variable D to always be less than 0.4, you should refer to the parameter as D__1 (note that there are two underscores in this parameter name: one to connect the name of the variable to its value and the other to replace the minus sign in the value itself). The following statement imposes the restriction on the parameter in question:

```
RESTRICT D__1 < 0.4;
```

Depending on the parameterization that you impose on your CLASS variable, one of the parameters associated with its levels can be dropped from your model before optimization in order to avoid collinearity. For example, when the default parameterization GLM is imposed, the parameter associated with the last level of your CLASS variable is dropped before optimization. If you attempt to impose a restriction on a dropped parameter by using the RESTRICT statement, you receive an error message in the log.

For example, suppose once again that your CLASS variable is named C and that it has three levels: 0, 1, 2. Suppose your model is the following:

```
class C;
model y = x1 x2 C;
```

Because no additional options were specified in the CLASS statement, the GLM parameterization is assumed. This entails that the parameter named C_2 (which is the parameter associated with the last level of your CLASS variable) will be dropped from your model before the optimizer is invoked. Therefore, you generate an error if you attempt to restrict the C_2 parameter in any way by referring to it in a RESTRICT statement. For example, the following RESTRICT statement generates an error:

```
RESTRICT C_2 < 0.3;
```

Referring to Parameters Associated with Interactions between Regressors

When a regressor in your model involves an interaction between other regressors, you can impose restrictions on the parameters associated with the interaction as described in this section.

Suppose you have the following model:

```
model y = x1 x2 x3*x4;
```

You can form the name of the parameter associated with the interaction regressor x3*x4 by replacing the multiplication sign with an underscore. Thus, x3_x4 refers to the parameter associated with the interaction regressor x3*x4.

Referring to interactions between regressors and CLASS variables is handled in exactly the same way. Suppose you have a CLASS variable named C that has three levels (0, 1, 2), and that your model is the following:

```
class C;
model y = x1 x2 C*x3;
```

The interaction between the continuous variable x3 and the CLASS variable C introduces three additional parameters, which are named x3_C_0, x3_C_1, and x3_C_2. Note that, although the order of the terms in the interaction is C followed by x3, the name of the parameter associated with the interaction is formed by placing the name of the continuous variable x3 first, followed by an underscore, followed by the name of the CLASS variable C, followed by another underscore, and then followed by the level value. Once again, depending on the parameterization that you specify in your CLASS statement, for each interaction in your model that involves a CLASS variable, one of the parameters associated with that interaction can be dropped from your model before optimization.

The name of a parameter associated with a nested interaction is formed in a slightly different way. Suppose you have a CLASS variable named C that has three levels (0, 1, 2) and your model is the following:

```
class C;
model y = x1 x2 x3(C);
```

The nested interaction between the continuous variable x3 and the CLASS variable C introduces three additional parameters, which are named x3_C__0, x3_C__1, and x3_C__2. Note how the name in each case was formed from the name of the regressor by replacing the left and right parentheses with underscores and then appending another underscore followed by the level value.

Referring to Implicit Parameters

For all models in PROC SPATIALREG, one or more implicit parameters are added to your model before optimization. You can impose restrictions on these implicit parameters as follows.

If you have a linear model or SLX model, the _sigma2 parameter is added to your model. For the SAR or SDM model, the _rho and _sigma2 parameters are added to your model.

If you specify TYPE=SEM or TYPE=SMA, the _lambda and _sigma2 parameters are added to your model. If you specify the TYPE=SAC or TYPE=SARMA option, then three implicit parameters are added to your model: _rho, _lambda, and _sigma2.

Whenever your model type dictates the addition of one or more of these implicit parameters, you can impose restrictions on the implicit parameters by referring to them by name. For example, assuming that your model type implies the existence of the _rho parameter, you can restrict _rho to be greater than 0 as follows:

```
RESTRICT _rho > 0.0;
```

Computational Resources

The time and memory that PROC SPATIALREG requires are proportional to the number of parameters in the model and the number of observations in the data set being analyzed. Also affecting resources are the method that is chosen to calculate the variance-covariance matrix and the optimization method. All optimization methods available through the METHOD= option have similar memory use requirements.

The processing time might differ for each method, depending on the number of iterations and functional calls needed. The data set is read into memory to save processing time. If not enough memory is available to hold the data, the SPATIALREG procedure stores the data in a utility file on disk and rereads the data as needed from this file. When this occurs, the execution time of the procedure increases substantially. The gradient and the variance-covariance matrix must be held in memory. If the model has p parameters, including the intercept, then at least $8(p + p(p + 1)/2)$ bytes are needed. If the quasi–maximum likelihood method is used to estimate the variance-covariance matrix (COVEST=QML), an additional $8p(p + 1)/2$ bytes of memory are needed.

Processing time is also a function of the number of iterations needed to converge to a solution for the model parameters. The number of iterations cannot be known in advance. The MAXITER= option can be used to limit the number of iterations that PROC SPATIALREG performs. The convergence criteria can be altered by nonlinear optimization options available in the PROC SPATIALREG statement. For a list of all the nonlinear optimization options, see Chapter 6, “Nonlinear Optimization Methods.”

Nonlinear Optimization Options

PROC SPATIALREG uses the nonlinear optimization (NLO) subsystem to perform nonlinear optimization tasks. In the PROC SPATIALREG statement, you can specify nonlinear optimization options that are then passed to the NLO subsystem. For a list of all the nonlinear optimization options, see Chapter 6, “Nonlinear Optimization Methods.”

Covariance Matrix Types

The SPATIALREG procedure enables you to specify the estimation method for the covariance matrix. The COVEST=HESSIAN option estimates the covariance matrix based on the inverse of the Hessian matrix, COVEST=OP uses the outer product of gradients, and COVEST=QML produces the covariance matrix based on both the Hessian and outer product matrices. By default, COVEST=HESSIAN.

Although all three methods produce asymptotically equivalent results, they differ in computational intensity and produce results that might differ in finite samples. The COVEST=OP option provides the covariance matrix that is usually the easiest to compute. In some cases, the OP approximation is considered more efficient than the Hessian or QML approximation because it contains fewer random elements. The QML approximation is computationally the most complex, because both the outer product of gradients and the Hessian matrix are required. In most cases, the OP or Hessian approximation is preferred to QML. The need to use QML approximation arises in some cases when the model is misspecified and the information matrix equality does not hold.

When Taylor approximation or Chebyshev approximation is used for the SAR and SDM models, only COVEST=HESSIAN is supported.

Displayed Output

PROC SPATIALREG produces the following displayed output.

Class Level Information

If you specify the CLASS statement, the SPATIALREG procedure displays a table that contains the following information:

- CLASS variable name
- number of levels of the CLASS variable
- list of values of the CLASS variable

Iteration History for Parameter Estimates

If you specify the ITPRINT or PRINTALL option in the PROC SPATIALREG statement, PROC SPATIALREG displays a table that contains the following information for each iteration. Some information is specific to the model-fitting procedure that you choose (for example, Newton-Raphson, trust region, quasi-Newton).

- iteration number
- number of restarts since the fitting began
- number of function calls
- number of active constraints at the current solution
- value of the objective function (the negative log-likelihood value) at the current solution
- change in the objective function from previous iteration
- value of the maximum absolute gradient element
- step size (for Newton-Raphson and quasi-Newton methods)
- slope of the current search direction (for Newton-Raphson and quasi-Newton methods)
- lambda (for trust region method)
- radius value at current iteration (for trust region method)

Model Fit Summary

The “Model Fit Summary” table contains the following information:

- dependent variable name
- number of observations used
- data set name
- name of the spatial weights data set (specified by the WMAT= option)
- type of model that was fit
- log-likelihood value at solution
- maximum absolute gradient at solution
- number of iterations
- AIC value at the solution (a smaller value indicates a better fit)
- SBC value at the solution (a smaller value indicates a better fit)

Below the “Model Fit Summary” table is a statement about whether the algorithm successfully converged.

Parameter Estimates

The “Parameter Estimates” table displays the estimates of the model parameters. In the SAR model, estimates are also displayed for the autoregressive coefficient ρ and the variance of the error terms σ^2. For the SEM, SDEM, SMA, and SDMA models, estimates are given for the autoregressive coefficient λ and the variance of the error terms σ^2. In addition, for SAC, SDAC, SARMA, and SDARMA models, estimates are given for the autoregressive coefficients ρ and λ, and the variance of the error terms σ^2. In the linear and SLX models, estimates are given for the variance of the error terms σ^2.

“_rho” is the internal name of the autoregressive coefficient ρ in the SAR, SDM, SARMA, SDARMA, SAC, and SDAC models. The *t* statistic given for “_rho” is a test of autoregressive coefficient. In addition, “_lambda” is the internal name of the autoregressive coefficient λ in the SEM, SDEM, SMA, SARMA, SAC, and SDAC models. Moreover, “_sigma2” is the internal name of the variance parameter σ^2.

Last Evaluation of the Gradient

If you specify the ITPRINT option in the MODEL statement, the SPATIALREG procedure displays the last evaluation of the gradient vector.

Covariance of Parameter Estimates

If you specify the COVB option in the MODEL statement or in the PROC SPATIALREG statement, the SPATIALREG procedure displays the estimated covariance matrix, defined as the inverse of the information matrix, evaluated at the final iteration.

Correlation of Parameter Estimates

If you specify the CORRB option in the MODEL statement or in the PROC SPATIALREG statement, PROC SPATIALREG displays the estimated correlation matrix. It is based on the Hessian matrix that is used in the final iteration.

OUTPUT OUT= Data Set

The OUTPUT statement creates a new SAS data set that contains all the variables in the input data set and, optionally, the estimates of $\mathbf{x}_i'\boldsymbol{\beta}$, the expected value of the response variable, and the residual.

OUTEST= Data Set

The OUTEST= data set has two rows: the first row (with _TYPE_='PARM') contains each of the parameter estimates in the model, and the second row (with _TYPE_='STD') contains the standard errors for the parameter estimates in the model.

If you specify the COVOUT option in the PROC SPATIALREG statement, the OUTEST= data set also contains the covariance matrix for the parameter estimates. The covariance matrix appears in the observations for which _TYPE_='COV', and the _NAME_ variable labels the rows with the parameter names.

The names of the parameters are used as variable names. These are the same names that are used in the INIT, BOUNDS, and RESTRICT statements.

ODS Table Names

PROC SPATIALREG assigns a name to each table that it creates. You can use these names to denote the table when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 31.2.

Table 31.2 ODS Tables Produced in PROC SPATIALREG

ODS Table Name	Description	Option
ODS Tables Created by the MODEL Statement		
ClassLevels	Class levels	Default
FitSummary	Summary of nonlinear estimation	Default
ConvergenceStatus	Convergence status	Default
ParameterEstimates	Parameter estimates	Default
CovB	Covariance of parameter estimates	COVB
CorrB	Correlation of parameter estimates	CORRB
InputOptions	Input options	ITPRINT
IterStart	Optimization start	ITPRINT
IterHist	Iteration history	ITPRINT
IterStop	Optimization results	ITPRINT
ParameterEstimatesResults	Parameter estimates	ITPRINT

Table 31.2 *continued*

ODS Table Name	Description	Option
ParameterEstimatesStart	Parameter estimates	ITPRINT
ProblemDescription	Problem description	ITPRINT
ODS Tables Created by the TEST Statement		
TestResults	Test results	Default

Examples: SPATIALREG Procedure

Example 31.1: Columbus Crime Data

Data Description and Objective

The data set CRIMEOH contains data from Columbus, Ohio, about the number of crimes (including residential burglaries and vehicle thefts) and possible determinants of crime. This data set is taken from Anselin (1988) and can be found in the SAS/ETS Sample Library.

The variable CRIME represents the number of crimes in 49 neighborhoods of Columbus, Ohio. Additional variables in the data set that you want to evaluate as determinants of crimes include INCOME (household income by $1000) and HVALUE (housing value by $1000). Summary statistics for these variables are computed by the following statements and presented in Output 31.1.1:

```
proc means data=crimeoh;
   var crime income hvalue;
run;
```

Output 31.1.1 Summary Statistics

The MEANS Procedure

```
Variable  N        Mean     Std Dev     Minimum     Maximum
crime    49  35.1288367  16.7320385   0.1780000  68.8920000
income   49  14.3749388   5.7033781   4.4770000  31.0700000
hvalue   49  38.4362245  18.4660693  17.9000000  96.4000000
```

The spatial relationships among the 49 neighborhoods are summarized using the first-order neighbor contiguity matrix, contained in the CRIMEWMAT data set. This data set is also taken from Anselin (1988) and can be found in the SAS/ETS Sample Library.

Spatial Autoregressive (SAR) Model

The following statements fit a SAR model to the data by using the regressors INCOME and HVALUE:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SAR;
run;
```

In this example, the TYPE=SAR option in the MODEL statement specifies a SAR model. The NONORMALIZE option indicates that the spatial weights data set CRIMEWMAT should be used "as is" rather than be row-standardized. The parameter estimates for this model are shown in Output 31.1.2. According to the results, the spatial autoregressive coefficient ρ is positive and significant at the 0.05 level. This indicates that there is a positive spatial dependence in the data.

Output 31.1.2 Parameter Estimates of SAR Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	45.077070	7.870590	5.73	<.0001
income	1	-1.031531	0.328403	-3.14	0.0017
hvalue	1	-0.265924	0.088218	-3.01	0.0026
_rho	1	0.431020	0.123594	3.49	0.0005
_sigma2	1	95.487066	19.506312	4.90	<.0001

Spatial Durbin Model (SDM)

To fit an SDM model, you specify the SPATIALEFFECTS statement together with the TYPE=SAR option. In this example, the spatial lags of the regressors INCOME and HVALUE are considered in the SDM model.

The following statements fit an SDM model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SAR;
   spatialeffects income hvalue;
run;
```

The parameter estimates are given in Output 31.1.3. As in the SAR model, the spatial autoregressive coefficient ρ in the SDM model is positive and significant at the 0.05 level, indicating a positive spatial dependence in the data.

Output 31.1.3 Parameter Estimates of SDM Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	42.803457	13.924487	3.07	0.0021
income	1	-0.914206	0.336439	-2.72	0.0066
hvalue	1	-0.293745	0.088857	-3.31	0.0009
W_income	1	-0.519640	0.594772	-0.87	0.3823
W_hvalue	1	0.245716	0.176854	1.39	0.1647
_rho	1	0.426492	0.167492	2.55	0.0109
_sigma2	1	91.779519	18.909222	4.85	<.0001

In order to avoid potential collinearity with the intercept term in the MODEL statement, the SPATIALEFFECTS statement always excludes the intercept term. This means that only the explicitly specified variables in the SPATIALEFFECTS statement are used to construct spatial lag of covariate effects.

Spatial Error Model (SEM)

To fit an SEM model, use the TYPE=SEM option.

The following statements fit an SEM model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SEM;
run;
```

The parameter estimates are shown in Output 31.1.4. According to this output, the *p*-value for the spatial autoregressive parameter λ is 0.0002. The results indicate that there is a significant positive dependence in the error term.

Output 31.1.4 Parameter Estimates of SEM Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	59.891907	5.884103	10.18	<.0001
income	1	-0.941301	0.370267	-2.54	0.0110
hvalue	1	-0.302253	0.090552	-3.34	0.0008
_lambda	1	0.561781	0.152413	3.69	0.0002
_sigma2	1	95.572081	20.037403	4.77	<.0001

Spatial Durbin Error Model (SDEM)

To fit an SDEM model, use the SPATIALEFFECTS statement together with the TYPE=SEM option. In this example, the spatial lags of the regressors INCOME and HVALUE are considered in the SDEM model.

The following statements fit an SDEM model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SEM;
   spatialeffects income hvalue;
run;
```

The parameter estimates are shown in Output 31.1.5.

Output 31.1.5 Parameter Estimates of SDEM Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	73.540584	8.860968	8.30	<.0001
income	1	-1.051699	0.322436	-3.26	0.0011
hvalue	1	-0.275607	0.091154	-3.02	0.0025
W_income	1	-1.156553	0.592915	-1.95	0.0511
W_hvalue	1	0.111754	0.202366	0.55	0.5808
_lambda	1	0.425397	0.173831	2.45	0.0144
_sigma2	1	92.533614	19.090022	4.85	<.0001

Spatial Moving Average (SMA) Model

To fit an SMA model, use the TYPE=SMA option.

The following statements fit an SMA model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SMA;
run;
```

The parameter estimates are shown in Output 31.1.6.

Output 31.1.6 Parameter Estimates of SMA Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	59.252971	5.934861	9.98	<.0001
income	1	-0.921806	0.363482	-2.54	0.0112
hvalue	1	-0.287393	0.086880	-3.31	0.0009
_lambda	1	-0.799089	0.277861	-2.88	0.0040
_sigma2	1	117.731990	26.373322	4.46	<.0001

Spatial Durbin Moving Average (SDMA) Model

To fit an SDMA model, use the SPATIALEFFECTS statement together with the TYPE=SMA option. In this example, the spatial lags of the regressors INCOME and HVALUE are considered in the SDMA model.

The following statements fit an SDMA model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SMA;
   spatialeffects income hvalue;
run;
```

Partial output is shown in Output 31.1.7.

Output 31.1.7 Parameter Estimates of SDMA Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	73.944211	9.083977	8.14	<.0001
income	1	-1.065635	0.312045	-3.42	0.0006
hvalue	1	-0.266840	0.092400	-2.89	0.0039
W_income	1	-1.074757	0.584955	-1.84	0.0662
W_hvalue	1	0.067568	0.209867	0.32	0.7475
_lambda	1	-0.642124	0.296638	-2.16	0.0304
_sigma2	1	103.502516	22.487027	4.60	<.0001

Spatial Autoregressive Confused (SAC) Model

To fit an SAC model, use the TYPE=SAC option.

The following statements fit the SAC model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SAC;
run;
```

The parameter estimates are shown in Output 31.1.8.

Output 31.1.8 Parameter Estimates of SAC Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	47.778937	9.278438	5.15	<.0001
income	1	-1.025840	0.334006	-3.07	0.0021
hvalue	1	-0.281636	0.093366	-3.02	0.0026
_rho	1	0.368143	0.181118	2.03	0.0421
_lambda	1	0.166526	0.298115	0.56	0.5764
_sigma2	1	95.597117	19.474269	4.91	<.0001

Spatial Durbin Autoregressive Confused (SDAC) Model

To fit an SDAC model, use the SPATIALEFFECTS statement together with the TYPE=SAC option. In this example, the spatial lags of the regressors INCOME and HVALUE are considered in the SDAC model.

The following statements fit an SDAC model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SAC;
   spatialeffects income hvalue;
run;
```

The parameter estimates are shown in Output 31.1.9.

Output 31.1.9 Parameter Estimates of SDAC Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	50.827256	31.089621	1.63	0.1021
income	1	-0.950352	0.353961	-2.68	0.0073
hvalue	1	-0.286559	0.091261	-3.14	0.0017
W_income	1	-0.690471	0.839980	-0.82	0.4111
W_hvalue	1	0.208936	0.222585	0.94	0.3479
_rho	1	0.316760	0.414771	0.76	0.4450
_lambda	1	0.152884	0.475512	0.32	0.7478
_sigma2	1	93.133958	19.187743	4.85	<.0001

Spatial Autoregressive Moving Average (SARMA) Model

To fit a SARMA model, use the TYPE=SARMA option.

The following statements fit a SARMA model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SARMA;
run;
```

The parameter estimates are shown in Output 31.1.10.

Output 31.1.10 Parameter Estimates of SARMA Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	48.973247	9.602039	5.10	<.0001
income	1	-1.016359	0.337215	-3.01	0.0026
hvalue	1	-0.287458	0.093079	-3.09	0.0020
_rho	1	0.336281	0.204317	1.65	0.0998
_lambda	1	-0.271945	0.426840	-0.64	0.5241
_sigma2	1	97.992936	21.253768	4.61	<.0001

Spatial Durbin Autoregressive Moving Average (SDARMA) Model

To fit an SDARMA model, use the SPATIALEFFECTS statement together with the TYPE=SARMA option. In this example, the spatial lags of the regressors INCOME and HVALUE are considered in the SDARMA model.

The following statements fit an SDARMA model without an intercept term to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=SARMA noint;
   spatialeffects income hvalue;
run;
```

The parameter estimates are shown in Output 31.1.11.

Output 31.1.11 Parameter Estimates of SDARMA Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
income	1	-0.792292	0.379696	-2.09	0.0369
hvalue	1	-0.328521	0.095588	-3.44	0.0006
W_income	1	0.587122	0.457090	1.28	0.1990
W_hvalue	1	0.438500	0.136144	3.22	0.0013
_rho	1	0.957745	0.041913	22.85	<.0001
_lambda	1	0.691307	0.260974	2.65	0.0081
_sigma2	1	86.990404	19.034142	4.57	<.0001

Linear Regression Model

To fit a linear model, use the TYPE=LINEAR option.

The following statements fit a linear model to the CRIMEOH data:

```
proc spatialreg data=crimeoh;
   model crime=income hvalue / type=LINEAR;
run;
```

Partial output is shown in Output 31.1.12.

Output 31.1.12 Parameter Estimates of Linear Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

			Standard		Approx
Parameter	**DF**	**Estimate**	**Error**	**t Value**	**Pr > |t|**
Intercept	1	68.618863	4.588210	14.96	<.0001
income	1	-1.597304	0.323739	-4.93	<.0001
hvalue	1	-0.273931	0.099989	-2.74	0.0062
_sigma2	1	122.751696	24.799493	4.95	<.0001

Spatial Lag of X Model

To fit an SLX model, use the SPATIALEFFECTS statement together with the TYPE=LINEAR option. In this example, the spatial lags of the regressors INCOME and HVALUE are considered in the linear model.

The following statements fit an SLX model to the CRIMEOH data:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
   model crime=income hvalue / type=LINEAR;
   spatialeffects income hvalue;
run;
```

The parameter estimates are shown in Output 31.1.13.

Output 31.1.13 Parameter Estimates of SLX Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: crime

Parameter Estimates

			Standard		Approx
Parameter	**DF**	**Estimate**	**Error**	**t Value**	**Pr > |t|**
Intercept	1	75.028184	6.279950	11.95	<.0001
income	1	-1.109020	0.354232	-3.13	0.0017
hvalue	1	-0.289734	0.096058	-3.02	0.0026
W_income	1	-1.370866	0.531889	-2.58	0.0100
W_hvalue	1	0.191785	0.189841	1.01	0.3124
_sigma2	1	107.292373	21.676329	4.95	<.0001

Example 31.2: Models with Spatial ID Matching

Data Description and Objective

Two simulated data sets, SIMDATA and SIMW, are used to illustrate models with spatial ID matching in PROC SPATIALREG.

The SIMDATA data set contains 50 observations and five variables. The variable SID identifies each spatial unit in the data. Three explanatory variables are x1, x2, and x3. The dependent variable is y. The SIMW data set defines the spatial contiguity for all 50 spatial units. The first column, SID, in the SIMW data set identifies each spatial unit. The remaining entries in the SIMW data set are binary and define whether two spatial units are neighbors. A value of 1 indicates that two spatial units are neighbors, and 0 indicates otherwise.

Summary statistics for all variables except SID in the SIMDATA data set are computed by the following statements and presented in Output 31.2.1:

```
proc means data=simdata;
   var x1 x2 x3 y;
run;
```

Output 31.2.1 Summary Statistics

The MEANS Procedure

```
Variable  N       Mean    Std Dev     Minimum    Maximum
x1       50 -0.0076329  1.0989504  -2.4523193  1.6539456
x2       50 -0.0829941  0.9671181  -2.5725767  2.4034547
x3       50 -0.0894387  0.9975304  -2.4470049  2.6720533
y        50  1.1569199  1.5687060  -1.9399423  4.7136835
```

Because the SIMDATA and SIMW data sets are ordered differently in terms of the values of SID, the SPATIALID statement is needed to match observations in SIMDATA and SIMW. The following statements fit a SAR model to the data by using three regressors, x1, x2, and x3:

```
proc spatialreg data=simdata Wmat=simw;
  model y=x1-x3 / type=SAR;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.2.2.

Output 31.2.2 Parameter Estimates of SAR Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.780650	0.098703	18.04	<.0001
x1	1	0.573329	0.047395	12.10	<.0001
x2	1	0.707048	0.057181	12.37	<.0001
x3	1	-0.902843	0.053314	-16.93	<.0001
_rho	1	-0.473713	0.063008	-7.52	<.0001
_sigma2	1	0.131509	0.026350	4.99	<.0001

To fit an SDM model that includes exogenous interaction effects of x1, x2, and x3, submit the following statements:

```
proc spatialreg data=simdata Wmat=simw;
  model y=x1-x3/ type=SAR;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.2.3.

Output 31.2.3 Parameter Estimates of SDM Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.932575	0.198882	9.72	<.0001
x1	1	0.548505	0.049806	11.01	<.0001
x2	1	0.686012	0.056266	12.19	<.0001
x3	1	-0.890162	0.053516	-16.63	<.0001
W_x1	1	0.172300	0.154018	1.12	0.2633
W_x2	1	0.023744	0.198557	0.12	0.9048
W_x3	1	-0.324806	0.228032	-1.42	0.1543
_rho	1	-0.639755	0.164652	-3.89	0.0001
_sigma2	1	0.120527	0.024729	4.87	<.0001

If you want to fit another type of model, you need to change the TYPE= option. As an example, if you want to fit an SEM model instead of a SAR model to the data, you can use the following statements:

```
proc spatialreg data=simdata Wmat=simw;
  model y=x1-x3 / type=SEM;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.2.4.

Output 31.2.4 Parameter Estimates of SEM Model

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

			Standard		Approx
Parameter	**DF**	**Estimate**	**Error**	**t Value**	**Pr > |t|**
Intercept	1	1.166289	0.029514	39.52	<.0001
x1	1	0.487975	0.049086	9.94	<.0001
x2	1	0.634442	0.061776	10.27	<.0001
x3	1	-0.831250	0.054780	-15.17	<.0001
_lambda	1	-0.964826	0.132514	-7.28	<.0001
_sigma2	1	0.147434	0.031318	4.71	<.0001

Example 31.3: Fitting Multiple Models

You can fit more than one model by making only one call to PROC SPATIALREG. For example, if you want to fit both SAR and SEM models to the CRIMEOH data set, you can use the following statements:

```
proc spatialreg data=crimeoh Wmat=crimeWmat NONORMALIZE;
  model crime=income hvalue / type=SAR;
  model crime=income hvalue / type=SEM;
run;
```

The parameter estimates for the SAR and SEM models are shown in Output 31.3.1 and Output 31.3.2, respectively.

Output 31.3.1 Parameter Estimates of SAR Model

Parameter Estimates

			Standard		Approx
Parameter	**DF**	**Estimate**	**Error**	**t Value**	**Pr > |t|**
Intercept	1	45.077070	7.870590	5.73	<.0001
income	1	-1.031531	0.328403	-3.14	0.0017
hvalue	1	-0.265924	0.088218	-3.01	0.0026
_rho	1	0.431020	0.123594	3.49	0.0005
_sigma2	1	95.487066	19.506312	4.90	<.0001

Output 31.3.2 Parameter Estimates of SEM Model

Parameter Estimates

			Standard		Approx
Parameter	**DF**	**Estimate**	**Error**	**t Value**	**Pr > |t|**
Intercept	1	59.891907	5.884103	10.18	<.0001
income	1	-0.941301	0.370267	-2.54	0.0110
hvalue	1	-0.302253	0.090552	-3.34	0.0008
_lambda	1	0.561781	0.152413	3.69	0.0002
_sigma2	1	95.572081	20.037403	4.77	<.0001

Example 31.4: Compact Representation of a Spatial Weights Matrix

When a spatial weights matrix is sparse, you might want to provide its compact representation rather than the full matrix to PROC SPATIALREG. In this case, you must use a SPATIALID statement. This example shows you how to use the compact representation of a spatial weights matrix in PROC SPATIALREG. For illustration, the simulated data sets SIMDATA and SIMW in "Example 31.2: Models with Spatial ID Matching" on page 2350 are used here. The compact representation of the spatial weights matrix in the SIMW data set is created and saved in the SIMW_COMPACT data set.

The first 10 observations in the SIMW_COMPACT data set are shown in Figure 31.4.1.

Output 31.4.1 SIMW_COMPACT Data Set

Obs	SID	cSID	Value
1	L50	L45	1
2	L30	L22	1
3	L42	L46	1
4	L32	L35	1
5	L7	L25	1
6	L33	L25	1
7	L50	L25	1
8	L23	L50	1
9	L9	L7	1
10	L45	L36	1

The following statements fit a SAR model:

```
proc spatialreg data=simdata Wmat=simw_compact;
  model y=x1-x3 / type=SAR;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.4.2.

Output 31.4.2 Parameter Estimates of SAR Model with Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.780650	0.098703	18.04	<.0001
x1	1	0.573329	0.047395	12.10	<.0001
x2	1	0.707048	0.057181	12.37	<.0001
x3	1	-0.902843	0.053314	-16.93	<.0001
_rho	1	-0.473713	0.063008	-7.52	<.0001
_sigma2	1	0.131509	0.026350	4.99	<.0001

To fit an SEM model instead of a SAR model to the data, you can use the following statements:

```
proc spatialreg data=simdata Wmat=simw_compact;
  model y=x1-x3 / type=SEM;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.4.3.

Output 31.4.3 Parameter Estimates of SEM Model with Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.166289	0.029514	39.52	<.0001
x1	1	0.487975	0.049086	9.94	<.0001
x2	1	0.634442	0.061776	10.27	<.0001
x3	1	-0.831250	0.054780	-15.17	<.0001
_lambda	1	-0.964826	0.132514	-7.28	<.0001
_sigma2	1	0.147434	0.031318	4.71	<.0001

Example 31.5: Taylor and Chebyshev Approximations

When you have a large data set (that is, the number of spatial units in your data is large), it becomes burdensome to fit some models. This is partially because all models except linear regression models involve the calculation of the determinant of the matrix of a large size (such as $|\mathbf{I} - \rho\mathbf{W}|$ in a SAR model). In these cases, Taylor and Chebyshev approximations in PROC SPATIALREG can be helpful. The SPATIALREG procedure enables you to estimate both SAR and SDM models with a relatively large spatial weights matrix by using these two approximations. Using the two small data sets SIMDATA and SIMW in "Example 31.2: Models with Spatial ID Matching" on page 2350, you will see how you can invoke the two approximations in PROC SPATIALREG.

The following statements fit a SAR model by using Chebyshev approximation:

```
proc spatialreg data=simdata Wmat=simw approximation=(ORDER=10);
  model y=x1-x3 / type=SAR;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.1. Note that the spatial weights matrix in the SIMW data set is a full matrix. Compared with Output 31.2.2, Chebyshev approximation yields very similar parameter estimates.

Output 31.5.1 Parameter Estimates of SAR Model with Chebyshev Approximation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.780638	0.098699	18.04	<.0001
x1	1	0.573329	0.047395	12.10	<.0001
x2	1	0.707050	0.057181	12.37	<.0001
x3	1	-0.902843	0.053314	-16.93	<.0001
_rho	1	-0.473704	0.063004	-7.52	<.0001
_sigma2	1	0.131509	0.026350	4.99	<.0001

Using the compact representation of the spatial weights matrix, you can submit the following statements to fit a SAR model by using Chebyshev approximation:

```
proc spatialreg data=simdata Wmat=simw_compact approximation=(ORDER=10);
  model y=x1-x3 / type=SAR;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.2, which is identical to Output 31.5.1.

Output 31.5.2 Parameter Estimates of SAR Model with Chebyshev Approximation and Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.780638	0.098699	18.04	<.0001
x1	1	0.573329	0.047395	12.10	<.0001
x2	1	0.707050	0.057181	12.37	<.0001
x3	1	-0.902843	0.053314	-16.93	<.0001
_rho	1	-0.473704	0.063004	-7.52	<.0001
_sigma2	1	0.131509	0.026350	4.99	<.0001

The following statements fit an SDM model by using Taylor approximation:

```
proc spatialreg data=simdata Wmat=simw approximation=(Taylor ORDER=50);
  model y=x1-x3/ type=SAR;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.3. Compared with Output 31.2.3, the SDM model that is fit using Taylor approximation yields almost identical parameter estimates.

Output 31.5.3 Parameter Estimates of SDM Model with Taylor Approximation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.932575	0.198882	9.72	<.0001
x1	1	0.548505	0.049806	11.01	<.0001
x2	1	0.686012	0.056266	12.19	<.0001
x3	1	-0.890162	0.053516	-16.63	<.0001
W_x1	1	0.172300	0.154018	1.12	0.2633
W_x2	1	0.023744	0.198557	0.12	0.9048
W_x3	1	-0.324806	0.228032	-1.42	0.1543
_rho	1	-0.639755	0.164652	-3.89	0.0001
_sigma2	1	0.120527	0.024729	4.87	<.0001

With the compact representation, the following statements fit the SDM model by using Taylor approximation:

```
proc spatialreg data=simdata Wmat=simw_compact
     approximation=(Taylor ORDER=50);
  model y=x1-x3/ type=SAR;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.4, which is identical to Output 31.5.3.

Output 31.5.4 Parameter Estimates of SDM Model with Taylor Approximation and Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.932575	0.198882	9.72	<.0001
x1	1	0.548505	0.049806	11.01	<.0001
x2	1	0.686012	0.056266	12.19	<.0001
x3	1	-0.890162	0.053516	-16.63	<.0001
W_x1	1	0.172300	0.154018	1.12	0.2633
W_x2	1	0.023744	0.198557	0.12	0.9048
W_x3	1	-0.324806	0.228032	-1.42	0.1543
_rho	1	-0.639755	0.164652	-3.89	0.0001
_sigma2	1	0.120527	0.024729	4.87	<.0001

To use Chebyshev approximation for the preceding SDM model, submit the following statements:

```
proc spatialreg data=simdata Wmat=simw approximation=(ORDER=10);
  model y=x1-x3/ type=SAR;
  spatialeffects x1-x3;
  spatialid SID;
  run;
```

The parameter estimates for this model are shown in Output 31.5.5, which is similar to Output 31.5.3.

Output 31.5.5 Parameter Estimates of SDM Model with Chebyshev Approximation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.932081	0.198494	9.73	<.0001
x1	1	0.548538	0.049802	11.01	<.0001
x2	1	0.686049	0.056262	12.19	<.0001
x3	1	-0.890191	0.053515	-16.63	<.0001
W_x1	1	0.172017	0.153857	1.12	0.2636
W_x2	1	0.023362	0.198331	0.12	0.9062
W_x3	1	-0.324342	0.227739	-1.42	0.1544
_rho	1	-0.639325	0.164295	-3.89	<.0001
_sigma2	1	0.120542	0.024731	4.87	<.0001

The following statements use Chebyshev approximation for this model with compact representation:

```
proc spatialreg data=simdata Wmat=simw_compact approximation=(ORDER=10);
  model y=x1-x3/ type=SAR;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.6.

Output 31.5.6 Parameter Estimates of SDM Model with Chebyshev Approximation and Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.932081	0.198494	9.73	<.0001
x1	1	0.548538	0.049802	11.01	<.0001
x2	1	0.686049	0.056262	12.19	<.0001
x3	1	-0.890191	0.053515	-16.63	<.0001
W_x1	1	0.172017	0.153857	1.12	0.2636
W_x2	1	0.023362	0.198331	0.12	0.9062
W_x3	1	-0.324342	0.227739	-1.42	0.1544
_rho	1	-0.639325	0.164295	-3.89	<.0001
_sigma2	1	0.120542	0.024731	4.87	<.0001

The following statements fit an SEM model by using Chebyshev approximation:

```
proc spatialreg data=simdata Wmat=simw approximation=(ORDER=10);
  model y=x1-x3 / type=SEM;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.7. Note that the spatial weights matrix in the SIMW data set is a full matrix. Chebyshev approximation yields parameter estimates that are very similar to those shown in Output 31.2.2.

Output 31.5.7 Parameter Estimates of SEM Model with Chebyshev Approximation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.166102	0.029941	38.95	<.0001
x1	1	0.488665	0.049412	9.89	<.0001
x2	1	0.637142	0.061543	10.35	<.0001
x3	1	-0.831674	0.055165	-15.08	<.0001
_lambda	1	-0.944395	0.110388	-8.56	<.0001
_sigma2	1	0.149103	0.031221	4.78	<.0001

The following statements fit an SEM model by using Chebyshev approximation and compact representation of the spatial weights matrix:

```
proc spatialreg data=simdata Wmat=simw_compact approximation=(ORDER=10);
  model y=x1-x3 / type=SEM;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.8, which is identical to Output 31.5.7.

Output 31.5.8 Parameter Estimates of SEM Model with Chebyshev Approximation and Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.166102	0.029941	38.95	<.0001
x1	1	0.488665	0.049412	9.89	<.0001
x2	1	0.637142	0.061543	10.35	<.0001
x3	1	-0.831674	0.055165	-15.08	<.0001
_lambda	1	-0.944395	0.110388	-8.56	<.0001
_sigma2	1	0.149103	0.031221	4.78	<.0001

The following statements fit an SDEM model by using Taylor approximation:

```
proc spatialreg data=simdata Wmat=simw approximation=(Taylor ORDER=50);
  model y=x1-x3/ type=SEM;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.9.

Output 31.5.9 Parameter Estimates of SDEM Model with Taylor Approximation

```
                    The SPATIALREG Procedure

                        Model: MODEL 1
                    Dependent  Variable: y

                      Parameter Estimates
                                 Standard             Approx
Parameter  DF   Estimate            Error   t Value  Pr > |t|
Intercept   1   1.193178         0.047129     25.32    <.0001
x1          1   0.566497         0.056782      9.98    <.0001
x2          1   0.722060         0.059717     12.09    <.0001
x3          1  -0.909212         0.061286    -14.84    <.0001
W_x1        1  -0.135284         0.128339     -1.05    0.2918
W_x2        1  -0.418116         0.152899     -2.73    0.0062
W_x3        1   0.290041         0.192397      1.51    0.1317
_lambda     1  -0.723853         0.209844     -3.45    0.0006
_sigma2     1   0.126723         0.026820      4.73    <.0001
```

The following statements fit an SDEM model by using Taylor approximation and compact representation of the spatial weights matrix:

```
proc spatialreg data=simdata Wmat=simw_compact
     approximation=(Taylor ORDER=50);
  model y=x1-x3/ type=SEM;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.10, which is identical to Output 31.5.9.

Output 31.5.10 Parameter Estimates of SDEM Model with Taylor Approximation and Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.193178	0.047129	25.32	<.0001
x1	1	0.566497	0.056782	9.98	<.0001
x2	1	0.722060	0.059717	12.09	<.0001
x3	1	-0.909212	0.061286	-14.84	<.0001
W_x1	1	-0.135284	0.128339	-1.05	0.2918
W_x2	1	-0.418116	0.152899	-2.73	0.0062
W_x3	1	0.290041	0.192397	1.51	0.1317
_lambda	1	-0.723853	0.209844	-3.45	0.0006
_sigma2	1	0.126723	0.026820	4.73	<.0001

The following statements use Chebyshev approximation for the preceding SDEM model:

```
proc spatialreg data=simdata Wmat=simw approximation=(ORDER=10);
  model y=x1-x3/ type=SEM;
  spatialeffects x1-x3;
  spatialid SID;
  run;
```

The parameter estimates for this model are shown in Output 31.5.11, which is similar to Output 31.5.10.

Output 31.5.11 Parameter Estimates of SDEM Model with Chebyshev Approximation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.193199	0.047156	25.30	<.0001
x1	1	0.566631	0.056730	9.99	<.0001
x2	1	0.722163	0.059688	12.10	<.0001
x3	1	-0.909288	0.061252	-14.84	<.0001
W_x1	1	-0.135591	0.128237	-1.06	0.2904
W_x2	1	-0.418437	0.152818	-2.74	0.0062
W_x3	1	0.290294	0.192253	1.51	0.1311
_lambda	1	-0.722074	0.208152	-3.47	0.0005
_sigma2	1	0.126797	0.026816	4.73	<.0001

The following statements use Chebyshev approximation for this model with compact representation of the spatial weights matrix:

```
proc spatialreg data=simdata Wmat=simw_compact approximation=(ORDER=10);
  model y=x1-x3/ type=SEM;
  spatialeffects x1-x3;
  spatialid SID;
run;
```

The parameter estimates for this model are shown in Output 31.5.12, which is similar to Output 31.5.11.

Output 31.5.12 Parameter Estimates of SDEM Model with Chebyshev Approximation and Compact Representation

The SPATIALREG Procedure

Model: MODEL 1
Dependent Variable: y

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|
Intercept	1	1.193199	0.047156	25.30	<.0001
x1	1	0.566631	0.056730	9.99	<.0001
x2	1	0.722163	0.059688	12.10	<.0001
x3	1	-0.909288	0.061252	-14.84	<.0001
W_x1	1	-0.135591	0.128237	-1.06	0.2904
W_x2	1	-0.418437	0.152818	-2.74	0.0062
W_x3	1	0.290294	0.192253	1.51	0.1311
_lambda	1	-0.722074	0.208152	-3.47	0.0005
_sigma2	1	0.126797	0.026816	4.73	<.0001

References

Anselin, L. (1988). *Spatial Econometrics: Methods and Models*. Amsterdam: Springer.

Anselin, L. (2001). "Spatial Econometrics." In *A Companion to Theoretical Econometrics*, edited by B. H. Baltagi, 310–330. Oxford: Wiley-Blackwell.

Elhorst, J. P. (2013). *Spatial Econometrics: From Cross-Sectional Data to Spatial Panels*. Berlin: Springer.

LaMotte, L. R. (1994). "A Note on the Role of Independence in *t* Statistics Constructed from Linear Statistics in Regression Models." *American Statistician* 48:238–240.

LeSage, J., and Pace, R. K. (2009). *Introduction to Spatial Econometrics*. Boca Raton, FL: CRC Press.

Searle, S. R. (1971). *Linear Models*. New York: John Wiley & Sons.

Subject Index

Syntax Index