Lecture #1 Maximum likelihood estimation of spatial regression models

James P. LeSage
Department of Economics
University of Toledo
Toledo, Ohio 43606
e-mail: jlesage@spatial-econometrics.com

March 2004
**Introduction**

- This material should provide a reasonably complete introduction to traditional spatial econometric regression models. These models represent relatively straightforward extensions of regression models.
- Spatial dependence in a collection of sample data implies that observations at location $i$ depend on other observations at locations $j \neq i$. Formally, we might state:

$$y_i = f(y_j), \ i = 1, \ldots, n \ j \neq i$$  \hspace{1cm} (1)

Note that we allow the dependence to be among several observations, as the index $i$ can take on any value from $i = 1, \ldots, n$.
- This is different from traditional regression where observations of the dependent variable vector $y$ are assumed independent.
- Recall in the traditional regression model:

$$y = X\beta + \varepsilon$$  \hspace{1cm} (2)

- Gauss-Markov assumptions are that: $E(\varepsilon_i, \varepsilon_j) = 0$
- Which translates through the FIXED $X$’s assumption to $E(y_i, y_j) = 0$. 

Dependence arising from theory and statistical considerations

- **A theoretical motivation for spatial dependence.**
  From a theoretical viewpoint, consumers in a neighborhood may emulate each other leading to spatial dependence. Local governments might engage in competition that leads to local uniformity in taxes and services. Pollution can create systematic patterns over space, and clusters of consumers who travel to a more distant store to avoid a high crime zone would also generate these patterns. A concrete example will be given later.

- **A statistical motivation for spatial dependence.**
  Spatial dependence can arise from unobservable latent variables that are spatially correlated. Consumer expenditures collected at spatial locations such as Census tracts exhibit spatial dependence, as do other variables such as housing prices. It seems plausible that difficult-to-quantify or unobservable characteristics such as the quality of life may also exhibit spatial dependence. A concrete example will be given later.
Estimation consequences of spatial dependence

- In some applications, the spatial structure of the dependence may be a subject of interest or provide a key insight.
- In other cases, it may be a nuisance similar to serial correlation.
- In either case, inappropriate treatment of sample data with spatial dependence can lead to inefficient and/or biased and inconsistent estimates.

- For models of the type: \( y_i = f(y_j) + X_i \beta + \varepsilon_i \)
  Least-squares estimates for \( \beta \) are biased and inconsistent, similar to the simultaneity problem.

- For model of the type:

\[
\begin{align*}
  y_i &= X_i \beta + u_i \\
  u_i &= f(u_j) + \varepsilon_i
\end{align*}
\]

Least-squares estimates for \( \beta \) are inefficient, but consistent, similar to the serial correlation problem.
Specifying dependence using weight matrices

There are several ways to quantify the structure of spatial dependence between observations, but a common specification relies on an $n \times n$ spatial weight matrix $D$ with elements $D_{ij} > 0$ for observations $j = 1 \ldots n$ that are neighbors to observation $i$.

- Example #1 Let three individuals be located such that individual 1 is a neighbor to 2, and 2 is a neighbor to both 1 and 3, while individual 3 is a neighbor to 2.

<table>
<thead>
<tr>
<th>Individual #1</th>
<th>Individual #2</th>
<th>Individual #3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$x_{31}$ $x_{30}$</td>
<td></td>
</tr>
<tr>
<td>$0$ $1$ $0$</td>
<td>$1$ $0$ $1$</td>
<td>$0$ $1$ $0$</td>
</tr>
</tbody>
</table>

The first row in $D$ represents observation #1, so we place a value of 1 in the second column to reflect that #2 is a neighbor to #1. Similarly, both #1 and #3 are neighbors to observation #2 resulting in 1’s in the first and third columns of the second row.

In (3) we set $D_{ii} = 0$ for reasons that will become apparent shortly. Another convention is to normalize the spatial weight matrix $D$ to have row-sums of unity, which we denote as $W$, known as a “row-stochastic matrix”.

\[ D = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \]
Example #2 Consider the 5 states shown, where States 2 and 3 are first-order contiguous to State 1, while State 4 is second-order contiguous to State 1, and State 5 is third-order contiguous to State 1.

A spatial weighting matrix based on the first-order contiguity relationships for the five areas is as follows:

\[
W = \begin{pmatrix}
0 & 1/2 & 1/2 & 0 & 0 \\
1/2 & 0 & 1/2 & 0 & 0 \\
1/3 & 1/3 & 0 & 1/3 & 0 \\
0 & 0 & 1/2 & 0 & 1/2 \\
0 & 0 & 0 & 1 & 0 \\
\end{pmatrix}
\]  (4)
Spatial regression relationships

Basic spatial regression models include spatial autoregression (SAR) and spatial error models (SEM).

\[ y = \rho W y + X \beta + \varepsilon \]  \hspace{1cm} (5)

\[ (I_n - \rho W) y = X \beta + \varepsilon \]

\[ y = (I_n - \rho W)^{-1} X \beta + (I_n - \rho W)^{-1} \varepsilon \]

- Where the implied data generating process for the traditional spatial autoregressive (SAR) model is shown in the last expression in (5).
- The first expression for the SAR model makes it clear why \( W_{ii} = 0 \), as this precludes an observation \( y_i \) from directly predicting itself.
- It also motivates the use of row-stochastic \( W \), which makes each observation \( y_i \) a function of the “spatial lag” \( Wy \), an explanatory variable representing an average of spatially neighboring values, e.g.,

\[ y_2 = \rho \left( \frac{1}{2} y_1 + \frac{1}{2} y_3 \right) \]  \hspace{1cm} (6)
Spatial lags $Wy$

\[
\begin{pmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_5 \\
\end{pmatrix} = \rho \begin{pmatrix}
0 & 0.5 & 0.5 & 0 & 0 \\
0.5 & 0 & 0.5 & 0 & 0 \\
0.33 & 0.33 & 0 & 0.33 & 0 \\
0 & 0 & 0.5 & 0 & 0.5 \\
0 & 0 & 0 & 1 & 0 \\
\end{pmatrix} \begin{pmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_5 \\
\end{pmatrix}
\]

\[
\begin{pmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_5 \\
\end{pmatrix} = \rho \begin{pmatrix}
1/2y_2 + 1/2y_3 \\
1/2y_1 + 1/2y_3 \\
1/3y_1 + 1/3y_2 + 1/3y_4 \\
1/2y_3 + 1/2y_5 \\
y_4 \\
\end{pmatrix}
\]
The Role of \((I_n - \rho W)^{-1}\)

Assigning a spatial correlation parameter value of \(\rho = 0.5\), \((I_3 - \rho W)^{-1}\) is as shown in (7).

\[
S^{-1} = (I - 0.5W)^{-1} = \begin{pmatrix}
1.1667 & 0.6667 & 0.1667 \\
0.3333 & 1.3333 & 0.3333 \\
0.1667 & 0.6667 & 1.1667
\end{pmatrix}
\]

(7)

- This model reflects a data generating process where \(S^{-1}X\beta\) indicates that individual (or observation) #1 derives utility that reflects a linear combination of the observed characteristics of their own house as well as characteristics of both other homes in the neighborhood.
- The weight placed on own-house characteristics is slightly less than twice that of the neighbor’s house (observation/individual #2) and around 7 times that of the non-neighbor (observation/individual #3). An identical linear combination exists for individual #3 who also has a single neighbor.
- For individual #2 with two neighbors we see a slightly different weighting pattern to the linear combination of own and neighboring house characteristics in generation of utility. Here, both neighbors characteristics are weighted equally, accounting for around one-fourth the weight associated with the own-house characteristics.
• Other points to note are that:
• Increasing the magnitude of the spatial dependence parameter $\rho$ would lead to an increase in the magnitude of the weights as well as a decrease in the decay as we move to neighbors and more distant non-neighbors:

For $\rho = 0.8$ is:

$$S^{-1} = \begin{pmatrix} 1.8889 & 2.2222 & 0.8889 \\ 1.1111 & 2.7778 & 1.1111 \\ 0.8889 & 2.2222 & 1.8889 \end{pmatrix}$$ (8)

For $\rho = 0.3$ is:

$$S^{-1} = \begin{pmatrix} 1.0495 & 0.3297 & 0.0495 \\ 0.1648 & 1.0989 & 0.1648 \\ 0.0495 & 0.3297 & 1.0495 \end{pmatrix}$$ (9)

• The addition of another observation/individual not in the neighborhood, represented by another row and column in the weight matrix with zeros in all positions will have no effect on the linear combinations.
• All connectivity relationships come into play through the matrix inversion process. By this we mean that house #3 which is a neighbor to #2 influences the utility of #1 because there is a connection between #3 and #2 which is a neighbor of #1.
Computing weight matrices

Figure 1: Delaunay triangularization

- We require knowledge of the location (latitude-longitude coordinates) associated with the observational units to determine the closeness of individual observations to other observations.

- Given these, we can rely on a Delaunay triangularization scheme to find neighboring observations.

- To illustrate this approach, Figure 1 shows a Delaunay triangularization centered on an observation located at point $A$. The space is partitioned into triangles such that there are no points in the interior of the circumscribed circle of any triangle.
• Neighbors could be specified using Delaunay contiguity defined as two points being a vertex of the same triangle. The neighboring observations to point $A$ that could be used to construct a spatial weight matrix are $B, C, E, F$.

• One way to specify a spatial weight matrix would be to set column elements associated with neighboring observations $B, C, E, F$ equal to 1 in row $A$. This would reflect that these observations are neighbors to observation $A$.

• Typically, the weight matrix is standardized so that row sums equal unity, producing a row-stochastic weight matrix.

• An alternative approach would be to rely on neighbors ranked by distance from observation $A$. We can simply compute the distance from $A$ to all other observations and rank these by size.

• In the case of figure 1 we would have:
  - a nearest neighbor $E$,
  - the nearest 2 neighbors $E, C$,
  - nearest 3 neighbors $E, C, D$, and so on.

• Again, we could set elements $D_{Aj} = 1$ for observations $j$ in row $A$ to reflect any number of nearest neighbors to observation $A$, and transform to row-stochastic form.

• This approach might involve including a determination of the appropriate number of neighbors as part of the model estimation problem.
Sparsity of weight matrices

- Sparse matrices are those that contain a large proportion of zeros.

- An example is the spatial weight matrix $W$ for a sample of 3,107 U.S. counties. When you consider the first-order contiguity structure of this sample, individual counties exhibited at most 8 first-order (borders touching) contiguity relations. This means that the remaining 2,999 entries in this row of $W$ are zero. The average number of contiguity relationships between the sample of counties was 4, so a great many of the elements in the matrix $W$ are zero, which is the definition of a sparse matrix.

- To understand how sparse matrix algorithms conserve on storage space and computer memory, consider that we need only record the non-zero elements of a sparse matrix for storage. Since these represent a small fraction of the total $3107 \times 3107 = 9,653,449$ elements in our example weight matrix, we save a tremendous amount of computer memory.

- In fact for this case of 3,107 counties, only 12,429 non-zero elements were found in the first-order spatial contiguity matrix, representing a very small fraction (far less than 1 percent) of the total elements.
MATLAB and sparse matrices

MATLAB provides a function `sparse` that can be used to construct a large sparse matrix by simply indicating the row and column positions of non-zero elements and the value of the matrix element for these non-zero row and column elements. Continuing with our county data example, we could store the first-order contiguity matrix in a single data file containing 12,429 rows with 3 columns that take the form: row column value

This represents a considerable savings in computational space when compared to storing a matrix containing 9,653,449 elements.

A handy utility function in MATLAB is `spy` which allows one to produce a specially formatted graph showing the sparsity structure associated with sparse matrices.

We demonstrate by executing `spy(W)` on our weight matrix $W$ from the Pace and Barry data set, which produced the graph shown in Figure 2. As we can see from the figure, most of the non-zero elements reside near the diagonal.
Figure 2: Sparsity structure of $W$ from Pace and Barry
Figure 3: Third-order contiguity
Spatial Durbin and spatial error models

We can extend the model in (5) to a spatial Durbin model (SDM), that allows for explanatory variables from neighboring observations, created by $WX$ as shown in (10).

$$\begin{align*}
y &= \rho Wy + X\beta + WX\gamma + \varepsilon \\
(1_n - \rho W)y &= X\beta + WX\gamma + \varepsilon \\
y &= (1_n - \rho W)^{-1}X\beta + (1_n - \rho W)^{-1}WX\gamma \\
&+ (1_n - \rho W)^{-1}\varepsilon
\end{align*}$$

The $k \times 1$ parameter vector $\gamma$ measures the marginal impact of the explanatory variables from neighboring observations on the dependent variable $y$. Multiplying $X$ by $W$ produces “spatial lags” of the explanatory variables that reflect an average of neighboring observations $X$—values.

Another model that has been used is the spatial error model (SEM):

$$\begin{align*}
y &= X\beta + u \\
u &= \rho W + \varepsilon \\
y &= X\beta + (1_n - \rho W)^{-1}\varepsilon
\end{align*}$$
A statistical motivation for spatial dependence

• The source of spatial dependence may be unobserved variables whose spatial variation over the sample of observations is the source of spatial dependence in $y$.

• Here we have a potentially different situation than described previously. There may be no underlying theoretical motivation for spatial dependence in the data generating process.

• Spatial dependence may arise from census tract boundaries that do not accurately reflect neighborhoods which give rise to the variables being collected for analysis.

Intuitively, one might suppose solutions to this type of dependence would be:

1) to incorporate proxies for the unobserved explanatory variables that would eliminate the spatial dependence;
2) collect sample data based on alternative administrative jurisdictions that give rise to the information being collected;
3) rely on latitude and longitude coordinates of the observations as explanatory variables, and perhaps interact these location variables with other explanatory variables in the relationship. The goal here would be to eliminate the spatial dependence in $y$, allowing us to proceed with least-squares estimation.

Some general empirical truths regarding spatial data samples.

- A conventional regression augmented with geographic dichotomous variables (e.g., region or state dummy variables), or variables reflecting interaction with locational coordinates that allow variation in the parameters over space can rarely outperform a simpler spatial model.
- Spatial models provide a parsimonious approach to modeling spatial dependence, or filtering out these influences when they are perceived as a nuisance.
- Gathering sample data from the appropriate administrative units is seldom a realistic option.

Note this is similar to the time-series case of serial dependence, where the source may be something inherent in the data generating process, or excluded important explanatory variables. However, I argue that it is much more difficult to “filter out” spatial dependence than it is to deal with serial correlation in time series.
Maximum likelihood estimation of SAR, SEM, SDM models

Maximum likelihood estimation of the SAR, SDM and SEM models described here and in Anselin (1988) involves maximizing the log likelihood function (concentrated with respect to $\beta$ and $\sigma^2$, the noise variance associated with $\varepsilon$) with respect to the parameter $\rho$.

For the case of the SAR model we have:

\[
\ln L = C + \ln \left| I_n - \rho W \right| - (n/2) \ln (e'e) \\
e = e_o - \rho e_d \\
e_o = y - X\beta_o \\
e_d = Wy - X\beta_d \\
\beta_o = (X'X)^{-1}X'y \\
\beta_d = (X'X)^{-1}X'Wy \\
\]

(12)

Where $C$ represents a constant not involving the parameters. The computationally troublesome aspect of this is the need to compute the log-determinant of the $n \times n$ matrix $(I_n - \rho W)$. Operation counts for computing this determinant grow with the cube of $n$ for dense matrices. This same approach can be applied to the SDM model by simply defining $X = [X \ W X]$ in (12).
Approach #1 to efficient computation

- Pace and Barry (1997), use direct sparse matrix algorithms such as the Cholesky or LU decompositions to compute the log-determinant over a grid of values for the parameter $\rho$ restricted to the interval $[0, 1)$ or $(-1, 1)$.
- Vector evaluation of the SAR or SDM log-likelihood functions over a grid of $q$ values of $\rho \in (-1, 1)$ can be used to find maximum likelihood estimates.

\[
\begin{pmatrix}
L(\beta, \rho_1) \\
L(\beta, \rho_2) \\
\vdots \\
L(\beta, \rho_q)
\end{pmatrix} \propto \begin{pmatrix}
\ln|S(\rho_1)| \\
\ln|S(\rho_2)| \\
\vdots \\
\ln|S(\rho_q)|
\end{pmatrix} - \frac{n}{2} \begin{pmatrix}
\ln(\phi(\rho_1)) \\
\ln(\phi(\rho_2)) \\
\vdots \\
\ln(\phi(\rho_q))
\end{pmatrix}
\]

(13)

- $|S(\rho_1)| = |(I_n - \rho_1 W)|$
- $\phi(\rho_i) = e_o'e_o - 2\rho_i e_d'e_o + \rho_i^2 e_d'e_d$.
- For the SDM model, we replace $X$ with $[X \ W X]$ in (12).
- Calculating the log-determinant, takes 201 seconds for a sample of 57,647 observations representing all Census tracts in the continental US. This is based on a grid of 100 values from $\rho = 0$ to 1 using sparse matrix algorithms in MATLAB version 6.0 on a 600 Mhz Pentium III computer.
Approach #2 to efficient computation

- A Monte Carlo estimator for the log determinant suggested by Barry and Pace (1999) allows larger problems to be tackled without the memory requirements or sensitivity to orderings associated with the direct sparse matrix approach.
- It also provides an asymptotic 95% confidence interval for the approximation.
- As an illustration of these computational advantages, the time required to compute a grid of log-determinant values for $\rho = 0, \ldots, 1$ based on 0.001 increments for the sample of 57,647 observations was 3.6 seconds, which compares quite favorably to 201 seconds for the direct sparse matrix computations cited earlier. This approach yielded nearly the same estimate of $\rho$ as the direct method (0.91 versus 0.92), despite the use of a spatial weight matrix based on pure nearest neighbors rather than the symmetricized nearest neighbor matrix used in the direct approach.
Estimates of dispersion

- So far, the estimation procedure set forth produces an estimate for the spatial dependence parameter $\rho$ through maximization of the log-likelihood function concentrated with respect to $\beta, \sigma$.
- Estimates for these parameters can be recovered given $\hat{\rho}$, the likelihood maximizing value of $\rho$ using:

$$
\hat{e} = e_o - \hat{\rho} e_d
$$
$$
e_o = y - X \beta_o
$$
$$
e_d = W y - X \beta_d
$$
$$\hat{\sigma}^2 = (\hat{e}' \hat{e}) / (n - k)
$$
$$
\beta_o = (X' X)^{-1} X' y
$$
$$\beta_d = (X' X)^{-1} X' W y
$$
$$\hat{\beta} = \beta_o - \hat{\rho} \beta_d
$$

- An implementation issue is constructing estimates of dispersion for these parameter estimates that can be used for inference. For problems involving a small number of observations, we can use our knowledge of the theoretical information matrix to produce estimates of dispersion.
- An asymptotic variance matrix based on the Fisher information matrix shown below for the parameters
\( \theta = (\rho, \beta, \sigma^2) \) can be used to provide measures of dispersion for the estimates of \( \rho, \beta \) and \( \sigma^2 \). Anselin (1988) provides the analytical expressions needed to construct this information matrix.

\[
[I(\theta)]^{-1} = -E\left[ \frac{\partial^2 L}{\partial \theta \partial \theta'} \right]^{-1} \tag{15}
\]

- This approach is computationally impossible when dealing with large scale problems involving thousands of observations. The expressions used to calculate terms in the information matrix involve operations on very large matrices that would take a great deal of computer memory and computing time. In these cases we can evaluate the numerical hessian matrix using the maximum likelihood estimates of \( \rho, \beta \) and \( \sigma^2 \) and our sparse matrix representation of the likelihood. Given the ability to evaluate the likelihood function rapidly, numerical methods can be used to compute approximations to the gradients shown in (15).

- A technical point is that the bounds on \( \rho \) for a row-standardized weight matrix are defined by the interval \([\mu_{\min}^{-1}, \mu_{\max}^{-1}]\) where \( \mu \) denote eigenvalues of the spatial weight matrix \( W \) (Lemma 2 in Sun et al., 1999). Further, \( \mu_{\min} < 0, \mu_{\max} > 0 \), so \( \mu_{\min}^{-1} < \rho < \mu_{\max}^{-1} \).
Applied examples

The first example represents a hedonic pricing model often used to model house prices, using the selling price as the dependent variable and house characteristics as explanatory variables. Housing values exhibit a high degree of spatial dependence.

Here is a MATLAB program that uses functions from the Spatial econometrics Toolbox available at www.spatial-econometrics.com to carry out estimation of a least-squares model, spatial Durbin model and spatial autoregressive model.

```matlab
% example1.m file
% An example of spatial model estimation compared to least-squares
load house.dat;
% an ascii datafile with 8 columns containing 30,987 observations on:
% column 1 selling price
% column 2 YrBlt, year built
% column 3 tla, total living area
% column 4 bedrooms
% column 5 rooms
% column 6 lotsize
% column 7 latitude
% column 8 longitude
y = log(house(:,1)); % selling price as the dependent variable
n = length(y);
xc = house(:,7); % latitude coordinates of the homes
yc = house(:,8); % longitude coordinates of the homes
% xy2cont() is a spatial econometrics toolbox function
[j W j] = xy2cont(xc,yc); % constructs a 1st-order contiguity
% spatial weight matrix W, using Delauney triangles
age = 1999 - house(:,2); % age of the house in years
age = age/100;
```
x = zeros(n,8); % an explanatory variables matrix
x(:,1) = ones(n,1); % an intercept term
x(:,2) = age; % house age
x(:,3) = age.*age; % house age-squared
x(:,4) = age.*age.*age; % house age-cubed
x(:,5) = log(house(:,6)); % log of the house lotsize
x(:,6) = house(:,5); % the # of rooms
x(:,7) = log(house(:,3)); % log of the total living area in the house
x(:,8) = house(:,4); % the # of bedrooms

vnames = strvcat('log(price)','constant','age','age2','age3','lotsize', ... 
                    'rooms','tla','beds');</ref>
result0 = ols(y,x); % ols() is a toolbox function for least-squares
prt(result0,vnames); % print results using prt() toolbox function

% compute ols mean absolute predicton error
mae0 = mean(abs(y - result0.yhat));
fprintf(1,'ols mean absolute prediction error = %8.4f 
',mae0);

% compute spatial Durbin model estimates
info.rmin = 0; % restrict rho to 0,1 interval
info.rmax = 1;
result1 = sdm(y,x,W,info); % sdm() is a toolbox function
prt(result1,vnames); % print results using prt() toolbox function

% compute mean absolute prediction error
mae1 = mean(abs(y - result1.yhat));
fprintf(1,'sdm mean absolute prediction error = %8.4f 
',mae1);

% compute spatial autoregressive model estimates
result2 = sar(y,x,W,info); % sar() is a toolbox function
prt(result2,vnames); % print results using prt() toolbox function

% compute mean absolute prediction error
mae2 = mean(abs(y - result2.yhat));
fprintf(1,'sar mean absolute prediction error = %8.4f 
',mae2);
### Table 1: A comparison of least-squares and SAR model estimates

<table>
<thead>
<tr>
<th>Variable</th>
<th>OLS $\beta$</th>
<th>OLS t-statistic</th>
<th>SAR $\beta$</th>
<th>SAR t-statistic</th>
</tr>
</thead>
<tbody>
<tr>
<td>constant</td>
<td>2.7599</td>
<td>35.6434</td>
<td>-0.3588</td>
<td>-10.9928</td>
</tr>
<tr>
<td>age</td>
<td>1.9432</td>
<td>28.5152</td>
<td>1.0798</td>
<td>22.4277</td>
</tr>
<tr>
<td>age2</td>
<td>-4.0476</td>
<td>-32.8713</td>
<td>-1.8522</td>
<td>-21.2607</td>
</tr>
<tr>
<td>age3</td>
<td>1.2549</td>
<td>18.9105</td>
<td>0.5042</td>
<td>10.7939</td>
</tr>
<tr>
<td>lotsize</td>
<td>0.1857</td>
<td>48.1380</td>
<td>0.0517</td>
<td>19.7882</td>
</tr>
<tr>
<td>rooms</td>
<td>0.0112</td>
<td>2.9778</td>
<td>-0.0033</td>
<td>-1.3925</td>
</tr>
<tr>
<td>tla</td>
<td>0.9233</td>
<td>74.8484</td>
<td>0.5317</td>
<td>70.6514</td>
</tr>
<tr>
<td>beds</td>
<td>-0.0150</td>
<td>-2.6530</td>
<td>0.0197</td>
<td>4.9746</td>
</tr>
</tbody>
</table>

It is instructive to compare the biased least-squares estimates for $\beta$ to those from the SAR model shown in Table 1. We see upward bias in the least-squares estimates indicating over-estimation of the sensitivity of selling price to the house characteristics when spatial dependence is ignored. This is a typical result for hedonic pricing models. The SAR model estimates reflect the marginal impact of home characteristics after taking the spatial location into account.

In addition to the upward bias, there are some sign differences as well as different inferences that would arise from least-squares versus SAR model estimates. For instance, least-squares indicates that more bedrooms has a significant negative impact on selling price, an unlikely event. The SAR model suggest that bedrooms have a positive impact.
on selling prices.

Here are the results printed by MATLAB from estimation of the three models using the spatial econometrics toolbox functions.

Ordinary Least-squares Estimates
Dependent Variable = log(price)
R-squared = 0.7040
Rbar-squared = 0.7039
sigma^2 = 0.1792
Durbin-Watson = 1.0093
Nobs, Nvars = 30987, 8

***************************************************************
Variable          Coefficient  t-statistic  t-probability
constant          2.759874  35.643410  0.000000
age               1.943234  28.515217  0.000000
age2              -4.047618 -32.871346  0.000000
age3              1.254940  18.910472  0.000000
lotsize           0.185666  48.138001  0.000000
rooms             0.011204  2.977828  0.002905
sla               0.923255  74.848378  0.000000
beds              -0.014952  2.653031  0.007981

least-squares mean absolute prediction error = 0.3066

The fit of the two spatial models is superior to that from least-squares as indicated by both the $R^2$—squared statistics as well as the mean absolute prediction errors reported with the printed output.

Turning attention to the SDM model estimates, here we see that the lotsize, number of rooms, total living area and number of bedrooms for neighboring (first-order contiguous) properties that sold have a negative impact on selling price.
(These variables are reported using \( W \)–variable name, to reflect the spatially lagged explanatory variables \( W^X \) in this model.) This is as we would expect, the presence of neighboring homes with larger lots, more rooms and bedrooms as well as more living space would tend to depress the selling price. Of course, the converse also is true, neighboring homes with smaller lots, less rooms and bedrooms as well as less living space would tend to increase the selling price.

Spatial autoregressive Model Estimates
Dependent Variable = log(price)
R-squared = 0.8537
Rbar-squared = 0.8537
\( \sigma^2 \) = 0.0885
Nobs, Nvars = 30987, 8
log-likelihood = -141785.83
# of iterations = 11
min and max rho = 0.0000, 1.0000
total time in secs = 32.0460
time for \( \text{lndet} \) = 19.6480
time for t-stats = 11.8670
Pace and Barry, 1999 MC lndet approximation used
order for MC appr = 50
iter for MC appr = 30

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Asymptot t-stat</th>
<th>z-probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>constant</td>
<td>-0.358770</td>
<td>-10.992793</td>
<td>0.000000</td>
</tr>
<tr>
<td>age</td>
<td>1.079761</td>
<td>22.427691</td>
<td>0.000000</td>
</tr>
<tr>
<td>age2</td>
<td>-1.852236</td>
<td>-21.260658</td>
<td>0.000000</td>
</tr>
<tr>
<td>age3</td>
<td>0.504158</td>
<td>10.793884</td>
<td>0.000000</td>
</tr>
<tr>
<td>lotsize</td>
<td>0.051726</td>
<td>19.788158</td>
<td>0.000000</td>
</tr>
<tr>
<td>rooms</td>
<td>-0.003318</td>
<td>-1.392471</td>
<td>0.163780</td>
</tr>
<tr>
<td>tla</td>
<td>0.531746</td>
<td>70.651381</td>
<td>0.000000</td>
</tr>
<tr>
<td>beds</td>
<td>0.019709</td>
<td>4.974640</td>
<td>0.000001</td>
</tr>
<tr>
<td>rho</td>
<td>0.634595</td>
<td>242.925754</td>
<td>0.000000</td>
</tr>
</tbody>
</table>
Spatial Durbin model
Dependent Variable = \text{log(price)}
R-squared = 0.8622
Rbar-squared = 0.8621
\text{sigma}^2 = 0.0834
log-likelihood = -141178.27
Nobs, Nvars = 30987, 8
# iterations = 15
min and max rho = 0.0000, 1.0000
total time in secs = 54.3180
time for lndet = 18.7970
time for t-stats = 29.6230
Pace and Barry, 1999 MC lndet approximation used
order for MC appr = 50
iter for MC appr = 30

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Asymptot t-stat</th>
<th>z-probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>constant</td>
<td>-0.093823</td>
<td>-1.100533</td>
<td>0.271100</td>
</tr>
<tr>
<td>age</td>
<td>0.553216</td>
<td>7.664817</td>
<td>0.000000</td>
</tr>
<tr>
<td>age2</td>
<td>-1.324653</td>
<td>-11.221285</td>
<td>0.000000</td>
</tr>
<tr>
<td>age3</td>
<td>0.404283</td>
<td>6.988042</td>
<td>0.000000</td>
</tr>
<tr>
<td>lotsize</td>
<td>0.121762</td>
<td>26.722881</td>
<td>0.000000</td>
</tr>
<tr>
<td>rooms</td>
<td>0.004331</td>
<td>1.730088</td>
<td>0.083615</td>
</tr>
<tr>
<td>tla</td>
<td>0.583322</td>
<td>74.771529</td>
<td>0.000000</td>
</tr>
<tr>
<td>beds</td>
<td>0.017392</td>
<td>4.465188</td>
<td>0.000008</td>
</tr>
<tr>
<td>W-age</td>
<td>0.129150</td>
<td>1.370133</td>
<td>0.170645</td>
</tr>
<tr>
<td>W-age2</td>
<td>0.247994</td>
<td>1.507512</td>
<td>0.131679</td>
</tr>
<tr>
<td>W-age3</td>
<td>-0.311604</td>
<td>-3.609161</td>
<td>0.000307</td>
</tr>
<tr>
<td>W-lotsize</td>
<td>-0.095774</td>
<td>-17.313644</td>
<td>0.000000</td>
</tr>
<tr>
<td>W-rooms</td>
<td>-0.015411</td>
<td>-2.814934</td>
<td>0.004879</td>
</tr>
<tr>
<td>W-tla</td>
<td>-0.109262</td>
<td>-6.466610</td>
<td>0.000000</td>
</tr>
<tr>
<td>W-beds</td>
<td>-0.043989</td>
<td>-5.312003</td>
<td>0.000000</td>
</tr>
<tr>
<td>rho</td>
<td>0.690597</td>
<td>245.929569</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

sdm mean absolute prediction error = 0.2031
Data generated examples

% PURPOSE: An example of using sar()
% spatial autoregressive model
% (on a small data set)
%---------------------------------------------------
% USAGE: sar_example (see also sar_example2 for a large data set)
%---------------------------------------------------

clear all;

% W-matrix from Anselin's neighborhood crime data set
load anselin.dat; % standardized 1st-order spatial weight matrix
latt = anselin(:,4);
long = anselin(:,5);
[junk W junk] = xy2cont(latt,long);
[n junk] = size(W);
In = speye(n);
rho = 0.7; % true value of rho
sige = 0.5;
k = 3;
x = randn(n,k);
beta(1,1) = -1.0;
beta(2,1) = 0.0;
beta(3,1) = 1.0;

y = (In-rho*W)\(x*beta) + (In-rho*W)\(randn(n,1)*sqrt(sige));

info.lflag = 0; % use full lndet no approximation
result0 = sar(y,x,W,info);
prt(result0);

result1 = sar(y,x,W); % default to Barry-Pace lndet approximation
prt(result1);

% demonstrate nature of approximation
result2 = sar(y,x,W); % default to Barry-Pace lndet approximation
prt(result2);
Estimation results

Spatial autoregressive Model Estimates

R-squared = 0.7472
Rbar-squared = 0.7362
sigma^-2 = 0.3705
Nobs, Nvars = 49, 3
log-likelihood = -31.293806
# of iterations = 17
min and max rho = -1.0000, 1.0000
total time in secs = 0.2310
time for ln det = 0.0910
time for t-stats = 0.0100
No ln det approximation used

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Asymptot t-stat</th>
<th>z-probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>variable 1</td>
<td>-1.133185</td>
<td>-12.353660</td>
<td>0.000000</td>
</tr>
<tr>
<td>variable 2</td>
<td>0.157118</td>
<td>1.852774</td>
<td>0.063915</td>
</tr>
<tr>
<td>variable 3</td>
<td>0.922562</td>
<td>12.042031</td>
<td>0.000000</td>
</tr>
<tr>
<td>rho</td>
<td>0.712988</td>
<td>9.833270</td>
<td>0.000000</td>
</tr>
</tbody>
</table>
The Pace-Barry Approximation

Spatial autoregressive Model Estimates
R-squared = 0.7497
Rbar-squared = 0.7388
sigma^2 = 0.3713
Nobs, Nvars = 49, 3
log-likelihood = -31.383845
# of iterations = 18
min and max rho = -1.0000, 1.0000
total time in secs = 0.1000
time for lndet = 0.0500
Pace and Barry, 1999 MC lndet approximation used
order for MC appr = 50
iter for MC appr = 30
***************************************************************
Variable Coefficient Asymptot t-stat z-probability
variable 1 -1.132033 -12.321720 0.000000
variable 2 0.156591 1.844467 0.065115
variable 3 0.922463 12.027199 0.000000
rho 0.707962 9.635943 0.000000
***************************************************************
Spatial autoregressive Model Estimates
R-squared = 0.7553
Rbar-squared = 0.7446
sigma^2 = 0.3737
Nobs, Nvars = 49, 3
log-likelihood = -31.659105
# of iterations = 16
min and max rho = -1.0000, 1.0000
total time in secs = 0.0910
time for lndet = 0.0400
Pace and Barry, 1999 MC lndet approximation used
order for MC appr = 50
iter for MC appr = 30
***************************************************************
Variable Coefficient Asymptot t-stat z-probability
variable 1 -1.129049 -12.236321 0.000000
variable 2 0.155226 1.822591 0.068365
variable 3 0.922206 11.985734 0.000000
rho 0.707962 9.635943 0.000000
Large data set example

```
% PURPOSE: An example of using sar() on a large data set
%---------------------------------------------------
% USAGE: sar_example2 (see sar_example for a small data set)
%---------------------------------------------------

clear all;
% NOTE a large data set with 3107 observations
% from Pace and Barry, takes around 150-250 seconds
load elect.dat; % load data on votes
latt = elect(:,5);
long = elect(:,6);
n = length(latt);
k = 4;
x = randn(n,k);
clear elect; % conserve on RAM memory
n = 3107;
[junk W junk] = xy2cont(latt,long);
vnames = strvcat('voters','const','educ','homeowners','income');

b = ones(k,1);
rho = 0.62;
sige = 0.5;
y = (speye(n) - rho*W)(x*b) + (speye(n) - rho*W)(randn(n,1)*sqrt(sige));

% use defaults including lndet approximation
result = sar(y,x,W); % maximum likelihood estimates
prt(result,vnames);

% use defaults including lndet approximation
result2 = sar(y,x,W); % maximum likelihood estimates
prt(result2,vnames);
```

Results

sar: hessian not positive definite augmenting small eigenvalues

Spatial autoregressive Model Estimates
Dependent Variable = voters
R-squared = 0.8841
Rbar-squared = 0.8840
sigma^2 = 0.4979
Nobs, Nvars = 3107, 4
log-likelihood = -2372.5798
# of iterations = 13
min and max rho = -1.0000, 1.0000
Pace and Barry, 1999 MC lndet approximation used

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Asymptotic t-stat</th>
<th>z-probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>const</td>
<td>0.997746</td>
<td>77.636101</td>
<td>0.000000</td>
</tr>
<tr>
<td>educ</td>
<td>0.993397</td>
<td>75.999664</td>
<td>0.000000</td>
</tr>
<tr>
<td>homeowners</td>
<td>1.014352</td>
<td>81.542086</td>
<td>0.000000</td>
</tr>
<tr>
<td>income</td>
<td>1.001651</td>
<td>78.517245</td>
<td>0.000000</td>
</tr>
<tr>
<td>rho</td>
<td>0.610956</td>
<td>62.458793</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

sar: hessian not positive definite augmenting small eigenvalues

Spatial autoregressive Model Estimates
Dependent Variable = voters
R-squared = 0.8841
Rbar-squared = 0.8840
sigma^2 = 0.4979
Nobs, Nvars = 3107, 4
log-likelihood = -2372.5764
# of iterations = 12
min and max rho = -1.0000, 1.0000
Pace and Barry, 1999 MC lndet approximation used

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Asymptotic t-stat</th>
<th>z-probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>const</td>
<td>0.997748</td>
<td>77.636040</td>
<td>0.000000</td>
</tr>
<tr>
<td>educ</td>
<td>0.993399</td>
<td>75.999618</td>
<td>0.000000</td>
</tr>
<tr>
<td>homeowners</td>
<td>1.014354</td>
<td>81.542007</td>
<td>0.000000</td>
</tr>
<tr>
<td>income</td>
<td>1.001654</td>
<td>78.517201</td>
<td>0.000000</td>
</tr>
<tr>
<td>rho</td>
<td>0.610944</td>
<td>62.455480</td>
<td>0.000000</td>
</tr>
</tbody>
</table>
Comparison with OLS

Spatial autoregressive Model Estimates
Dependent Variable = voters
R-squared = 0.8954
Rbar-squared = 0.8953
sigma^2 = 0.5152
Nobs, Nvars = 3107, 4
log-likelihood = -2430.2483
# of iterations = 18
min and max rho = -1.0000, 1.0000
total time in secs = 0.9710
time for lndet = 0.7710
time for t-stats = 0.1000
Pace and Barry, 1999 MC lndet approximation used
order for MC appr = 50
iter for MC appr = 30

***************************************************************
Variable Coefficient Asymptot t-stat z-probability
const 1.000692 77.304634 0.000000
educ 0.997515 76.395790 0.000000
homeowners 1.014264 77.045426 0.000000
income 0.992305 74.273462 0.000000
rho 0.618971 65.420021 0.000000

Ordinary Least-squares Estimates
Dependent Variable = voters
R-squared = 0.7855
Rbar-squared = 0.7853
sigma^2 = 1.4194
Durbin-Watson = 1.8322
Nobs, Nvars = 3107, 4

***************************************************************
Variable Coefficient t-statistic t-probability
const 1.122588 52.826400 0.000000
educ 1.100627 51.186126 0.000000
homeowners 1.132172 52.334730 0.000000
income 1.123492 51.271543 0.000000