Spatial Dependence with Polygons

Ignacio Sarmiento-Barbieri

Universidad de los Andes

June 28, 2022

Recap

- Types of Spatial Data
- Reading and Mapping spatial data in R
- Projections
- Creating Spatial Objects
- Measuring Distances

Agenda

Motivation

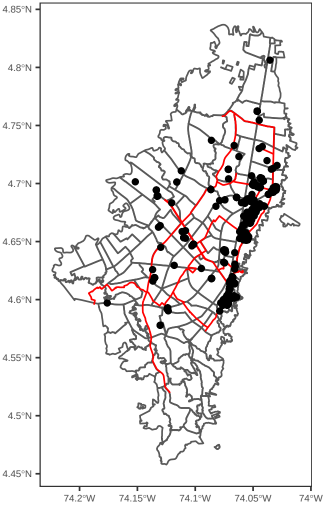
Cross-sectional iid non-spatial data

Standard cross-sectional models

$$y_i = X_i\beta + \epsilon_i \tag{1}$$
$$i = 1, \ldots, n \tag{2}$$

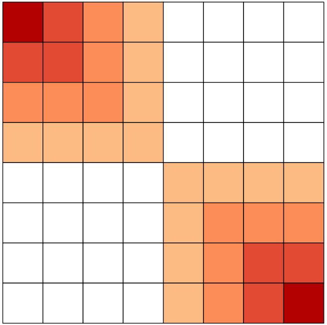
Motivation

- Independence assumption between observation is no longer valid
- Attributes of observation i may influence the attributes of observation j.
- We will consider various alternatives to model spatial dependence



Motivation

- Independence assumption between observation is no longer valid
- Attributes of observation i may influence the attributes of observation j.
- Positive spatial correlation arises when units that are *close* to one another are more similar than units that are far apart



Spatial Dependence

- We now take a closer look at spatial dependence, or to be more precise on it's weaker expression spatial autocorrelation.
- Spatial autocorrelation measures the degree to which a phenomenon of interest is correlated to itself in space (Cliff and Ord (1973)).
- Following Anselin and Bera (1998) we can express the existence of spatial autocorrelation with the following moment condition:

$$Cov(y_i, y_j) \neq 0 \;\; for \;\; i \neq j \tag{3}$$

  were y_i and y_j are observations on a random variable at locations i and j.
- The problem here is that we need to estimate N by N covariance terms directly for N observations.
- To overcome this problem we impose restrictions on the nature of the interactions.

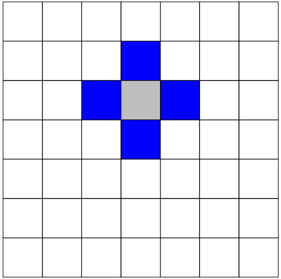
Closeness

"Everything is related to everything else, but close things are more related than things that are far apart" (Tobler, 1979).

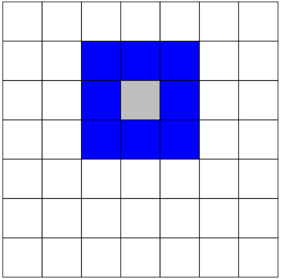
- One of the major differences between standard econometrics and standard spatial econometrics lies, in the fact that, in order to treat spatial data, we need to use two different sets of information
    1. Observed values of the variables
    2. Particular location where those variables are observed and to the various links of proximity between all spatial observations

Closeness

Rook criterion: two units are close to one another if they share a side



Queen criterion: two units are close if they share a side or an edge.



Weights Matrix

- At the heart of traditional spatial econometrics is the definition of the *weights matrix*:

$$W = \begin{pmatrix} w_{11} & \dots & \dots & w_{n1} \\ \vdots & w_{ij} & & \vdots \\ \vdots & & \ddots & \vdots \\ w_{n_1} & \dots & \dots & w_{nn} \end{pmatrix}_{n \times n} \tag{4}$$

with generic element:

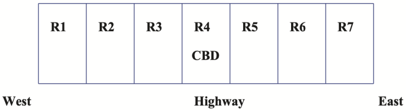
$$w_{ij} = \begin{cases} 1 & \textit{if } j \in N(i) \\ 0 & o.w \end{cases} \tag{5}$$

$N(i)$ being the set of neighbors of location j. By convention, the diagonal elements are set to zero, i.e. $w_{ii} = 0$.

Weights Matrix

- The specification of the neighboring set ($N(i)$) is quite arbitrary and there's a wide range of suggestions in the literature.
  - Rook criterion
  - Queen criterion
  - Two observations are neighbors if they are within a certain distance, i.e., $j \in N(j)$ if $d_{ij} < d_{max}$ where d is the distance between location i and j.
  - Closest neighbor, ties can be solved randomly
  - More general matrices can also be specified by considering entries of w_{ij} as functions of geographical, economic or social distances between areas rather than simply characterized by dichotomous entries

Some Examples of Weights Matrices

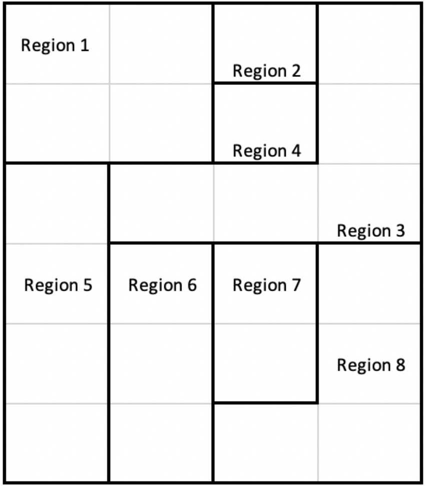


Fuente: LeSage & Pace (2009)

Adjacency Criterion

$$
W = \begin{array}{c} \\ R1 \\ R2 \\ R3 \\ R4 \\ R5 \\ R6 \\ R7 \end{array}
\begin{array}{c}
\begin{array}{ccccccc} R1 & R2 & R3 & R4 & R5 & R6 & R7 \end{array} \\
\begin{pmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0
\end{pmatrix}
\end{array}
\tag{6}
$$

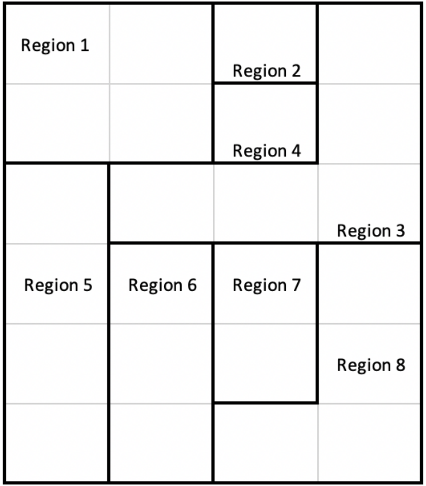
Some Examples of Weights Matrices



Adjacency Criterion

$$W = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}_{8\times 8}$$

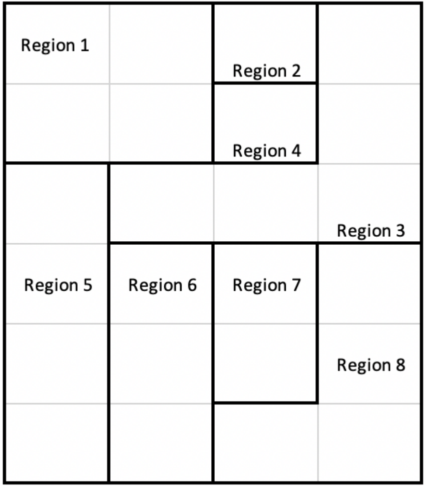
Some Examples of Weights Matrices



Nearest Neighbor

$$W = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}_{8\times 8}$$

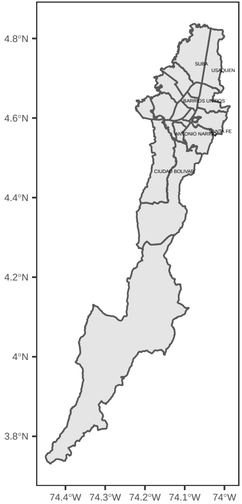
Some Examples of Weights Matrices



Distance < 2

$$W = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}_{8\times 8}$$

Some Examples of Weights Matrices



Some Examples of Weights Matrices

	ANTONIO NARIÑO	TUNJUELITO	RAFAEL URIBE URIBE	CANDELARIA	BARRIOS UNIDOS	TEUSAQUILLO	PUENTE ARANDA	LOS MARTIRES	SUMAPAZ	USAQUEN	CHAPINERO	SANTA FE	SAN CRISTOBAL
ANTONIO NARIÑO	0	1	1	0	0	0	1	1	0	0	0	1	1
TUNJUELITO	1	0	1	0	0	0	1	0	0	0	0	0	0
RAFAEL URIBE URIBE	1	1	0	0	0	0	0	0	0	0	0	0	1
CANDELARIA	0	0	0	0	0	0	0	0	0	0	0	1	0
BARRIOS UNIDOS	0	0	0	0	0	1	0	0	0	1	1	0	0
TEUSAQUILLO	0	0	0	0	1	0	1	1	0	0	1	1	0
PUENTE ARANDA	1	1	0	0	0	1	0	1	0	0	0	0	0
LOS MARTIRES	1	0	0	0	0	1	1	0	0	0	0	1	0
SUMAPAZ	0	0	0	0	0	0	0	0	0	0	0	0	0
USAQUEN	0	0	0	0	1	0	0	0	0	0	1	0	0
CHAPINERO	0	0	0	0	1	1	0	0	0	1	0	1	0
SANTA FE	1	0	0	1	0	1	0	1	0	0	1	0	1
SAN CRISTOBAL	1	0	1	0	0	0	0	0	0	0	0	1	0
USME	0	1	1	0	0	0	0	0	1	0	0	0	1
CIUDAD BOLIVAR	0	1	0	0	0	0	0	0	0	0	0	0	0
BOSA	0	0	0	0	0	0	0	0	0	0	0	0	0
KENNEDY	0	1	0	0	0	0	1	0	0	0	0	0	0
FONTIBON	0	0	0	0	0	1	1	0	0	0	0	0	0
ENGATIVA	0	0	0	0	1	1	0	0	0	0	0	0	0
SUBA	0	0	0	0	1	0	0	0	0	1	1	0	0

Some Examples of Weights Matrices

Quite often the W matrices are standardized to sum to one in each row

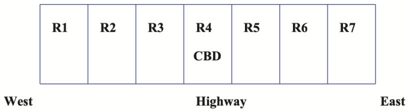
$$w_{ij}^{*} = \frac{w_{ij}}{\sum_{j=1}^{n} w_{ij}} \tag{7}$$

This can be quite useful since

$$L(y) = W^{*} y \tag{8}$$

Some Examples of Weights Matrices

Quite often the W matrices are standardized to sum to one in each row



Fuente: LeSage & Pace (2009)

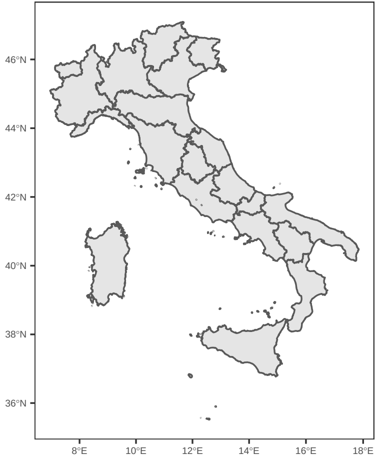
$$W = \begin{array}{c} \\ R1 \\ R2 \\ R3 \\ R4 \\ R5 \\ R6 \\ R7 \end{array} \begin{array}{ccccccc} R1 & R2 & R3 & R4 & R5 & R6 & R7 \\ \end{array}$$

$$W = \begin{array}{cccccccc} & R1 & R2 & R3 & R4 & R5 & R6 & R7 \\ R1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ R2 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ R3 & 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 \\ R4 & 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 \\ R5 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \\ R6 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ R7 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \tag{9}$$

Some Examples of Weights Matrices

	ANTONIO NARIÑO	TUNJUELITO	RAFAEL URIBE URIBE	CANDELARIA	BARRIOS UNIDOS	TEUSAQUILLO	PUENTE ARANDA	LOS MARTIRES	SUMAPAZ	USAQUEN	CHAPINERO	SANTA FE	SAN CRISTOBAL	
ANTONIO NARIÑO	0.0000000	0.1666667	0.1666667	0.0000000	0.0000000	0.0000000	0.1666667	0.1666667	0.0	0.00	0.0000000	0.1666667	0.1666667	0
TUNJUELITO	0.1666667	0.0000000	0.1666667	0.0000000	0.0000000	0.0000000	0.1666667	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	0
RAFAEL URIBE URIBE	0.2500000	0.2500000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.2500000	0
CANDELARIA	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0	0.00	0.0000000	1.0000000	0.0000000	0
BARRIOS UNIDOS	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.2000000	0.0000000	0.0000000	0.0	0.20	0.2000000	0.0000000	0.0000000	0
TEUSAQUILLO	0.0000000	0.0000000	0.0000000	0.0000000	0.1428571	0.0000000	0.1428571	0.1428571	0.0	0.00	0.1428571	0.1428571	0.0000000	0
PUENTE ARANDA	0.1666667	0.1666667	0.0000000	0.0000000	0.0000000	0.1666667	0.0000000	0.1666667	0.0	0.00	0.0000000	0.0000000	0.0000000	0
LOS MARTIRES	0.2500000	0.0000000	0.0000000	0.0000000	0.0000000	0.2500000	0.2500000	0.0000000	0.0	0.00	0.0000000	0.2500000	0.0000000	0
SUMAPAZ	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	1
USAQUEN	0.0000000	0.0000000	0.0000000	0.0000000	0.3333333	0.0000000	0.0000000	0.0000000	0.0	0.00	0.3333333	0.0000000	0.0000000	0
CHAPINERO	0.0000000	0.0000000	0.0000000	0.0000000	0.2000000	0.2000000	0.0000000	0.0000000	0.0	0.20	0.0000000	0.2000000	0.0000000	0
SANTA FE	0.1666667	0.0000000	0.0000000	0.1666667	0.0000000	0.1666667	0.0000000	0.1666667	0.0	0.00	0.1666667	0.0000000	0.1666667	0
SAN CRISTOBAL	0.2500000	0.0000000	0.2500000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0	0.00	0.0000000	0.2500000	0.0000000	0
USME	0.0000000	0.2000000	0.2000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.2	0.00	0.0000000	0.0000000	0.2000000	0
CIUDAD BOLIVAR	0.0000000	0.2500000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	0
BOSA	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	0
KENNEDY	0.0000000	0.2000000	0.0000000	0.0000000	0.0000000	0.0000000	0.2000000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	0
FONTIBON	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.2500000	0.2500000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	0
ENGATIVA	0.0000000	0.0000000	0.0000000	0.0000000	0.2500000	0.2500000	0.0000000	0.0000000	0.0	0.00	0.0000000	0.0000000	0.0000000	0
SUBA	0.0000000	0.0000000	0.0000000	0.0000000	0.2500000	0.0000000	0.0000000	0.0000000	0.0	0.25	0.2500000	0.0000000	0.0000000	0

Some Examples of Weights Matrices



Some Examples of Weights Matrices

	Piemonte	Valle D'Aosta	Lombardia	Trentino-Alto Adige	Veneto	Friuli Venezia Giulia	Liguria	Emilia-Romagna	Toscana	Umbria	Marche
Piemonte	0.0000000	0.25	0.2500000	0.00	0.0000000	0.00	0.2500000	0.2500000	0.0000000	0.0000000	0.0000000
Valle D'Aosta	1.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Lombardia	0.2500000	0.00	0.0000000	0.25	0.2500000	0.00	0.0000000	0.2500000	0.0000000	0.0000000	0.0000000
Trentino-Alto Adige	0.0000000	0.00	0.5000000	0.00	0.5000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Veneto	0.0000000	0.00	0.2500000	0.25	0.0000000	0.25	0.0000000	0.2500000	0.0000000	0.0000000	0.0000000
Friuli Venezia Giulia	0.0000000	0.00	0.0000000	0.00	1.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Liguria	0.3333333	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.3333333	0.3333333	0.0000000	0.0000000
Emilia-Romagna	0.1666667	0.00	0.1666667	0.00	0.1666667	0.00	0.1666667	0.0000000	0.1666667	0.0000000	0.1666667
Toscana	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.2000000	0.2000000	0.0000000	0.2000000	0.2000000
Umbria	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.3333333	0.0000000	0.3333333
Marche	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.2000000	0.2000000	0.2000000	0.0000000
Lazio	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.1666667	0.1666667	0.1666667
Abruzzo	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.3333333
Molise	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Campania	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Puglia	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Basilicata	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Calabria	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Sicilia	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Sardegna	0.0000000	0.00	0.0000000	0.00	0.0000000	0.00	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

	Lazio	Abruzzo	Molise	Campania	Puglia	Basilicata	Calabria	Sicilia	Sardegna
Piemonte	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Valle D'Aosta	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Lombardia	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Trentino-Alto Adige	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Veneto	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Friuli Venezia Giulia	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Liguria	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Emilia-Romagna	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Toscana	0.2000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Umbria	0.3333333	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Marche	0.2000000	0.2000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Lazio	0.0000000	0.1666667	0.1666667	0.1666667	0.0000000	0.0000000	0.0000000	0	0
Abruzzo	0.3333333	0.0000000	0.3333333	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Molise	0.2500000	0.2500000	0.0000000	0.2500000	0.2500000	0.0000000	0.0000000	0	0
Campania	0.2500000	0.0000000	0.2500000	0.0000000	0.2500000	0.2500000	0.0000000	0	0
Puglia	0.0000000	0.0000000	0.3333333	0.3333333	0.0000000	0.3333333	0.0000000	0	0
Basilicata	0.0000000	0.0000000	0.0000000	0.3333333	0.3333333	0.0000000	0.3333333	0	0
Calabria	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	1.0000000	0.0000000	0	0
Sicilia	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0
Sardegna	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0	0

Weights Matrix in R

```r
require("sf")
require("spdep")
require("dplyr")
```

```r
chi.poly<-read_sf("foreclosures/foreclosures.shp")
st_crs(chi.poly) #doesn't have a projection
```

```
## Coordinate Reference System: NA
```

```r
st_crs(chi.poly)<-4326 #WGS84 set it in the map
```

Weights Matrix in R

```r
chi.poly<-st_transform(chi.poly,26916) #reproject planarly
#NAD83 UTM Zone 16N
st_crs(chi.poly)
```

```
## Coordinate Reference System:
##   User input: EPSG:26916
##   wkt:
## PROJCS["NAD83 / UTM zone 16N",
##     GEOGCS["NAD83",
##         DATUM["North_American_Datum_1983",
##             SPHEROID["GRS 1980",6378137,298.257222101,
##                 AUTHORITY["EPSG","7019"]],
##             TOWGS84[0,0,0,0,0,0,0],
##             AUTHORITY["EPSG","6269"]],
##         PRIMEM["Greenwich",0,
##             AUTHORITY["EPSG","8901"]],
##         UNIT["degree",0.0174532925199433,
##             AUTHORITY["EPSG","9122"]],
##         AUTHORITY["EPSG","4269"]],
##     PROJECTION["Transverse_Mercator"],
##     PARAMETER["latitude_of_origin",0],
##     PARAMETER["central_meridian",-87],
##     PARAMETER["scale_factor",0.9996],
##     PARAMETER["false_easting",500000],
##     PARAMETER["false_northing",0],
##     UNIT["metre",1,
##         AUTHORITY["EPSG","9001"]],
##     AXIS["Easting",EAST],
##     AXIS["Northing",NORTH],
##     AUTHORITY["EPSG","26916"]]
```

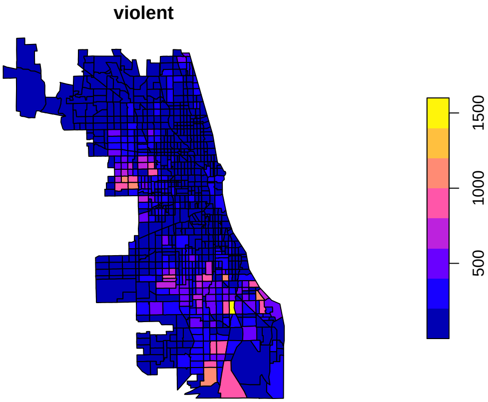
Weights Matrix in R

```r
str(chi.poly)
```

```
## tibble [897 x 17] (S3: sf/tbl_df/tbl/data.frame)
##  $ SP_ID     : chr [1:897] "1" "2" "3" "4" ...
##  $ fips      : chr [1:897] "17031010100" "17031010200" "17031010300" "17031010400" ...
##  $ est_fcs   : int [1:897] 43 129 55 21 64 56 107 43 7 51 ...
##  $ est_mtgs  : int [1:897] 904 2122 1151 574 1427 1241 1959 830 208 928 ...
##  $ est_fcs_rt: num [1:897] 4.76 6.08 4.78 3.66 4.48 4.51 5.46 5.18 3.37 5.5 ...
##  $ res_addr  : int [1:897] 2530 3947 3204 2306 5485 2994 3701 1694 443 1552 ...
##  $ est_90d_va: num [1:897] 12.61 12.36 10.46 5.03 8.44 ...
##  $ bls_unemp : num [1:897] 8.16 8.16 8.16 8.16 8.16 8.16 8.16 8.16 8.16 8.16 ...
##  $ county    : chr [1:897] "Cook County" "Cook County" "Cook County" "Cook County" ...
##  $ fips_num  : num [1:897] 1.7e+10 1.7e+10 1.7e+10 1.7e+10 1.7e+10 ...
##  $ totpop    : int [1:897] 5391 10706 6649 5325 10944 7178 10799 5403 1089 3634 ...
##  $ tothu     : int [1:897] 2557 3981 3281 2464 5843 3136 3875 1768 453 1555 ...
##  $ huage     : int [1:897] 61 53 56 60 54 58 48 57 61 48 ...
##  $ oomedval  : int [1:897] 169900 147000 119800 151500 143600 145900 153400 170500 215900 114700 ...
##  $ property  : num [1:897] 646 914 478 509 641 612 678 332 147 351 ...
##  $ violent   : num [1:897] 433 421 235 159 240 266 272 146 78 84 ...
##  $ geometry  :sfc_POLYGON of length 897; first list element: List of 1
##   ..$ : num [1:15, 1:2] 443923 444329 444814 444839 444935 ...
##   ..- attr(*, "class")= chr [1:3] "XY" "POLYGON" "sfg"
##  - attr(*, "sf_column")= chr "geometry"
##  - attr(*, "agr")= Factor w/ 3 levels "constant","aggregate",..: NA NA NA NA NA NA NA NA NA NA ...
##   ..- attr(*, "names")= chr [1:16] "SP_ID" "fips" "est_fcs" "est_mtgs" ...
```

Weights Matrix in R

```
plot(chi.poly['violent'])
```



Weights Matrix in R

```r
list.queen<-poly2nb(chi.poly, queen=TRUE)
W<-nb2listw(list.queen, style="W", zero.policy=TRUE)
W
```

```
## Characteristics of weights list object:
## Neighbour list object:
## Number of regions: 897
## Number of nonzero links: 6140
## Percentage nonzero weights: 0.7631036
## Average number of links: 6.845039
##
## Weights style: W
## Weights constants summary:
##     n     nn  S0       S1       S2
## W 897 804609 897 274.4893 3640.864
```

Weights Matrix in R

```
plot(W,st_geometry(st_centroid(chi.poly)))
```

Weights Matrix in R

```
coords <- st_centroid(st_geometry(chi.poly), of_largest_polygon=TRUE)
W_dist<-dnearneigh(coords,0,1000)
W_dist
```

```
## Neighbour list object:
## Number of regions: 897
## Number of nonzero links: 5448
## Percentage nonzero weights: 0.6770991
## Average number of links: 6.073579
## 55 regions with no links:
## 141 142 143 145 153 154 155 158 462 631 637 638 642 643 644 645 655 656 657 658 659 758 759 769 820 821 822 823 824 855 856 857 861 862 8
```

```
plot(W_dist, coords)
```

Weights Matrix in R

```r
W_dist<-dnearneigh(coords,0,4300)
W_dist
```

```
## Neighbour list object:
## Number of regions: 897
## Number of nonzero links: 87988
## Percentage nonzero weights: 10.9355
## Average number of links: 98.09142
```

```r
plot(W_dist, coords)
```

Testing for Spatial Dependence

$$y = X\beta + \epsilon$$

```
chi.ols<-lm(violent~est_fcs_rt+bls_unemp, data=chi.poly@data)
```

```
summary(chi.ols)
```

```
##
## Call:
## lm(formula = violent ~ est_fcs_rt + bls_unemp, data = chi.poly@data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -892.02  -77.02  -23.73   41.90 1238.22
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -18.627     45.366  -0.411    0.681
## est_fcs_rt    28.298      1.435  19.720   <2e-16 ***
## bls_unemp     -0.308      5.770  -0.053    0.957
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

Testing for Spatial Dependence

- We can use the OLS reiduals to test for spatial correlation.
- The most basic one is Moran's I test (1950), a test statistics for the null of uncorrelation among regression residuals.

$$I = \left(\frac{e'We}{e'e} \right) \tag{10}$$

- where $e = y - X\beta$ is a vector of OLS residuals $\beta = (X'X)^{-1}X'y$, W is the row standardized spatial weights matrix
- Moran's I test was originally developed as a two-dimensional analog of Durbin-Watson's test

Testing for Spatial Dependence

```r
moran.lm<-lm.morantest(chi.ols, W, alternative="two.sided")
print(moran.lm)
```

```
##
##  Global Moran I for regression residuals
##
## data:
## model: lm(formula = violent ~ est_fcs_rt + bls_unemp, data =
## chi.poly@data)
## weights: W
##
## Moran I statistic standard deviate = 11.785, p-value < 2.2e-16
## alternative hypothesis: two.sided
## sample estimates:
## Observed Moran I      Expectation         Variance
##     0.2142252370    -0.0020099108     0.0003366648
```

Spatial Autoregressive (SAR) Model

- We can think of situations where values observed at one location or region, say observation i, depend on the values of neighboring observations at nearby locations.

$$y_i = \rho_i y_j + X_i\beta + \epsilon_i \tag{11}$$
$$y_j = \rho_j y_i + X_j\beta + \epsilon_j \tag{12}$$

- This situation suggests a simultaneous data generating process, where the value taken by y_i depends on that of y_j and vice versa.

Spatial Autoregressive (SAR) Model

- Spatial lag dependence in a regression setting can be modeled similar to an autoregressive process in time series. Formally,

$$y = \rho W y + X\beta + u$$

- Wy induces a nonzero correlation with the error term, similar to the presence of an endogenous variable.
- Unlike to time series, Wy_i is always correlated with u
- OLS estimates in the non spatial model will be biased and inconsistent. (Anselin and Bera, 1998)
- In `R` the function `lagsarlm` uses MLE

Spatial Autoregressive (SAR) Model

```r
sar.chi<-lagsarlm(violent~est_fcs_rt+bls_unemp, data=chi.poly@data, W)
summary(sar.chi)
```

```
##
## Call:
## lagsarlm(formula = violent ~ est_fcs_rt + bls_unemp, data = chi.poly@data,
##     listw = W)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -519.127  -65.003  -15.226   36.423 1184.193
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -93.7885    41.3162  -2.270  0.02321
## est_fcs_rt   15.6822     1.5600  10.053  < 2e-16
## bls_unemp     8.8949     5.2447   1.696  0.08989
##
## Rho: 0.49037, LR test value: 141.33, p-value: < 2.22e-16
## Asymptotic standard error: 0.039524
##     z-value: 12.407, p-value: < 2.22e-16
## Wald statistic: 153.93, p-value: < 2.22e-16
##
## Log likelihood: -5738.047 for lag model
## ML residual variance (sigma squared): 20200, (sigma: 142.13)
## Number of observations: 897
## Number of parameters estimated: 5
## AIC: 11486, (AIC for lm: 11625)
## LM test for residual autocorrelation
```

Spatial Autoregressive (SAR) Model

Let's consider the following model:

$$y = \rho W y + X\beta + u$$

with $|\rho| < 1$, we also assume that W is exogenous

If W is row standardized:

- Guarantees $|\rho| < 1$ (Anselin, 1982)
- $[0, 1]$ Weights
- Wy Average of neighboring values
- W is no longer symmetric $\sum_j w_{ij} \neq \sum_i w_{ji}$ (complicates computation)

Spatial Autoregressive (SAR) Model
Maximum Likelihood Estimator

Note that we can write

$$(I - \lambda W)y = X\beta + u$$

- We can think this model as a way to correct for loss of information coming from spatial dependence.
- $(1 - \lambda W)y$ is a spatially filtered dependent variable, i.e., the effect of spatial autocorrelation taken out

Spatial Autoregressive (SAR) Model

- In this case, endogeneity emerges because the spatially lagged value of y is correlated with the stochastic disturbance.
- One solution that emerged in the literature is MLE
- We need an extra assumption, i.e., $u \sim_{iid} N(0, \sigma^2 I)$.

Spatial Autoregressive (SAR) Model

Maximum Likelihood Estimator

The associated likelihood function is then

$$\mathcal{L}\left(\sigma^2,\lambda,y\right)=\left(\frac{1}{\sqrt{2\pi}}\right)^n|\sigma^2\Omega|^{-\frac{1}{2}}exp\left\{-\frac{1}{2\sigma^2}(y-(I-\lambda W)^{-1}X\beta)'\Omega^{-1}(y-(I-\lambda W)^{-1}X\beta)\right\}$$

the log likelihood

$$l\left(\sigma^2,\lambda,y\right)=constant-\frac{1}{2}ln|\sigma^2\Omega|-\frac{1}{2\sigma^2}(y-(I-\lambda W)^{-1}X\beta)'\Omega^{-1}(y-(I-\lambda W)^{-1}X\beta)$$

so returning to the log likelihood we have that the log likelihood is

$$
\begin{aligned}
l\left(\sigma^2, \lambda, y\right) = {} & constant - \frac{n}{2} ln\left(\sigma^2\right) + ln\left(\left|(I - \lambda W)\right|\right) \\
& - \frac{1}{2\sigma^2}(y - (I - \lambda W)^{-1} X\beta)'(I - \lambda W)'(I - \lambda W)(y - (I - \lambda W)^{-1} X\beta)
\end{aligned} \tag{13}
$$

then

$$
\begin{aligned}
l\left(\sigma^2, \lambda, y\right) = {} & constant - \frac{n}{2} ln\left(\sigma^2\right) \\
& - \frac{1}{2\sigma^2}((I - \lambda W)y - X\beta)'((I - \lambda W) - X\beta) \\
& + ln\left(\left|(I - \lambda W)\right|\right)
\end{aligned} \tag{14}
$$

Spatial Autoregressive (SAR) Model

Maximum Likelihood Estimator

- The determinant $|(I - \lambda W)|$ is quite complicated because in contrast to the time series, where it is a triangular matrix, here it is a full matrix.
- However, Ord (1975) showed that it can be expressed as a function of the eigenvalues ω_i

$$|(I - \lambda W)| = \Pi_{i=1}^{n}(1 - \lambda\omega_i)$$

So the log likelihood is simplified to

$$\begin{aligned} l\left(\sigma^2, \lambda, y\right) &= constant - \frac{n}{2} ln\left(\sigma^2\right) \\ &- \frac{1}{2\sigma^2}((I - \lambda W)y - X\beta)'((I - \lambda W) - X\beta) \\ &+ \sum ln(1 - \lambda\omega_i) \end{aligned} \tag{15}$$

Spatial Autoregressive (SAR) Model

Maximum Likelihood Estimator

Applying FOC, the ML estimates for β and σ^2 are:

$$\beta_{MLE} = (X'X)^{-1}X'(I - \lambda W)y$$

$$\sigma^2_{MLE} = \frac{1}{n}(y - \lambda Xy - X\beta_{MLE})'(y - \lambda Xy - X\beta_{MLE})$$

- Conditional on λ these estimates are simply OLS applied to the spatially filtered dependent variable and explanatory variables X.

Spatial Autoregressive (SAR) Model

Maximum Likelihood Estimator

- Substituting these in the log likelihood we have a concentrated log-likelihood as a nonlinear function of a single parameter λ

$$l\left(\lambda\right) = -\frac{n}{2} ln\left(\frac{1}{n}(e_0 - \lambda e_L)'(e_0 - \lambda e_L)\right) + \sum ln(1 - \lambda\omega_i) \tag{16}$$

- where e_0 are the residuals in a regression of y on X and
- e_L of a regression of Wy on X.

- This expression can be maximized numerically to obtain the estimators for the unknown parameters λ.

Spatial Autoregressive (SAR) Model

Two-Stage Least Squares estimators

- An alternative to MLE we can us 2SLS to eliminate endogeneity.
- Key is to identify proper instruments
  - Need to be uncorrelated with the error term
  - Correlated with WY

Spatial Autoregressive (SAR) Model

Two-Stage Least Squares estimators

Consider the following

$$E(y) = (I - \lambda W)^{-1} X\beta$$

now, since $|\lambda| < 1$ we can use Neumann series property to expand the inverse matrix as

$$(I - \lambda W)^{-1} = I + \lambda W + \lambda^2 W^2 + \lambda^3 W^3 + ...$$

hence

$$E(y) = (I + \lambda W + \lambda^2 W^2 + \lambda^3 W^3 + ...) X\beta$$

$$= X\beta + \lambda W X\beta + \lambda^2 W^2 X\beta + \lambda^3 W^3 X\beta + ...$$

so we can express $E(y)$ as a function of X, WX, W^2X,...

Spatial Autoregressive (SAR) Model

Two-Stage Least Squares estimators

We can use the first three elements of the expansion as instruments. Let's define H as the matrix with our instruments

$$H = [X, WX, W^2X]$$

Spatial Error Model (SEM)

An OVB motivation

- True GDP

$$y = x\beta + z\theta$$

- but z is not observed and $z \perp x$
- we estimate

$$y = x\beta + \epsilon$$

- if z has a spatial autoregressive process

$$z = \rho W z + r$$

Spatial Error Model (SEM)

An OVB motivation

- Then

$$y = x\beta + (I - \rho W)^{-1}(\theta r)$$

- calling $(\theta r) = u$

$$y = x\beta + (I - \rho W)^{-1}u$$

- β will be unbiased but inefficient

Further Readings

- Arbia, G. (2014). A primer for spatial econometrics with applications in R. Palgrave Macmillan. (Chapter 2 and 3)
- Anselin, Luc, & Anil K Bera. 1998. "Spatial Dependence in Linear Regression Models with an Introduction to Spatial Econometrics." Statistics Textbooks and Monographs 155. MARCEL DEKKER AG: 237–90.
- Sarmiento-Barbieri, I. (2016). An Introduction to Spatial Econometrics in R. http://www.econ.uiuc.edu/~lab/workshop/Spatial_in_R.html
- Tobler, WR. 1979. "Cellular Geography." In Philosophy in Geography, 379–86. Springer.