Lecture 12:
Spatial Dependence

Big Data and Machine Learning for Applied Economics
Econ 4676

Ignacio Sarmiento-Barbieri

Universidad de los Andes

September 17, 2020

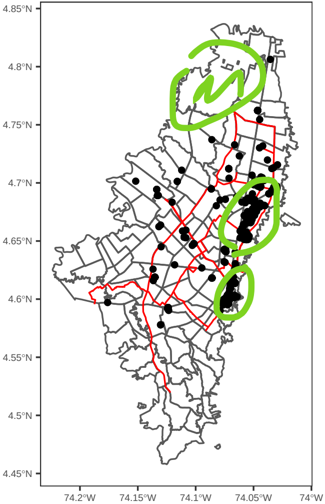
Recap

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

Agenda

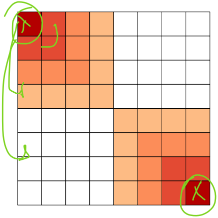
Motivation

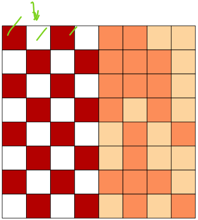
- Independence assumption between observation is no longer valid
- Attributes of observation i may influence the attributes of observation j.
- We will consider various alternatives to model spatial dependence
- Think as a way to model $f(X)$



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- Independence assumption between observation is no longer valid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
- Similarly spatial heterogeneity arises when some areas present more variability than others





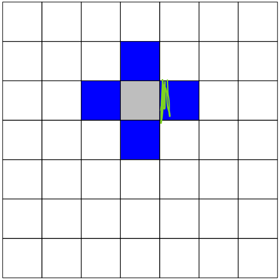
Closeness

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

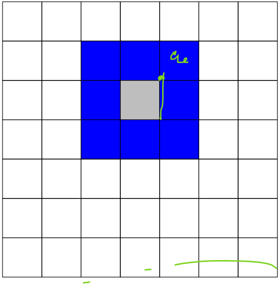
- One of the major differences between standard econometrics and standard spatial econometrics lies, in the fact that, in order to treat spatial data, we need to use two different sets of information
  1. Observed values of the economic variables
  2. Particular location where those variables are observed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

$$
\begin{array}{c} \\ 1 \\ 2 \\ \vdots \\ n \end{array}
\begin{array}{c}
\begin{array}{cccc} 1 & 2 & \cdots & n \end{array} \\
W = \begin{pmatrix} w_{11} & \dots & \dots & w_{n1} \\ \vdots & w_{ij} & & \vdots \\ \vdots & & \ddots & \vdots \\ w_{n_1} & \dots & \dots & w_{nn} \end{pmatrix}_{n \times n}
\end{array} \rightarrow 0 \tag{1}
$$

with generic element:

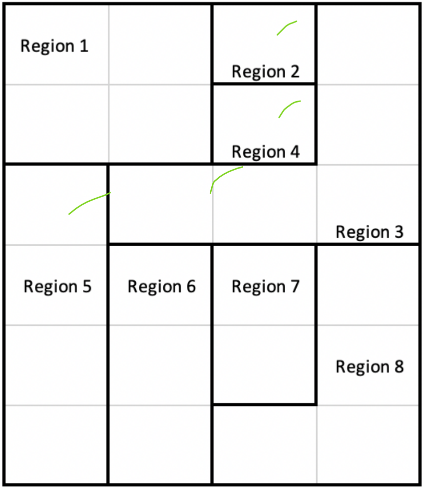
$$
w_{ij} = \begin{cases} 1 & \text{if } j \in N(i) \\ 0 & o.w \end{cases} \tag{2}
$$

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

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

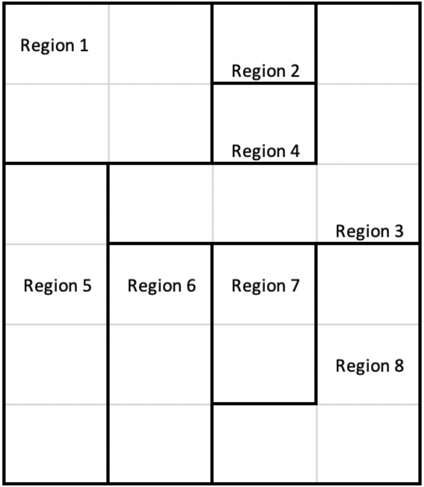
Some Examples of Weights Matrices



Adjacency Criterion

$$W = \begin{array}{c} \\ 1 \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}_{8\times 8} \end{array}$$

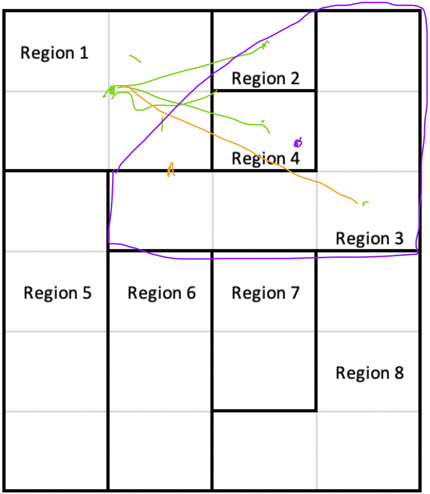
Some Examples of Weights Matrices



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

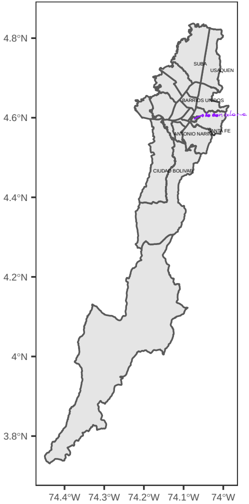


Distance < 2

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

convex hull

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

Some Examples of Weights Matrices

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

$$w_{ij}^* = \frac{w_{ij}}{\sum_{j=1}^n} w_{ij} \tag{3}$$

This can be quite useful since

$$L(y) = W^* y \tag{4}$$

in which each single element is equal to

$$L(y_i) = \sum_{j=1}^n w_{ij}^* y_j \tag{5}$$

$$= \sum_{j=1}^n \frac{w_{ij} y_j}{\sum_{j=1}^n w_{ij}}$$

$$= \frac{\sum_{j \in N(i)} y_j}{\# N(i)} \tag{6}$$

cardinalidad
↓
$\# N(i)$ elementos
en el conjunto N_i

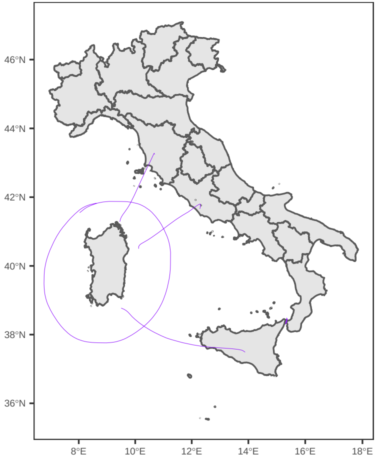
$\# N_i = 6$

Some Examples of Weights Matrices

$\frac{1}{6} = 0.16$

	ANTONIO NARIÑO	TUNJUELITO	RAFAEL URIBE URIBE	CANDELARIA	BARRIOS UNIDOS	TEUSAQUILLO	PUENTE ARANDA	LOS MARTIRES	SUMAPAZ	USAQUEN	CHAPINERO	SANTA FE	SAN CRISTOBAL	USME	CIUDAD BOLIVAR	BOSA	KENNED
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	0.0000000	0.0000000	0.00	0.000000
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	0.1666667	0.1666667	0.00	0.166666
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	0.2500000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.166666
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	0.0000000	0.0000000	0.00	0.000000
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	1.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000
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	0.2500000	0.0000000	0.00	0.000000
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	0.0000000	0.2000000	0.00	0.000000
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	0.2500000	0.0000000	0.25	0.250000
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	0.0000000	0.5000000	0.00	0.500000
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	0.0000000	0.2000000	0.20	0.000000
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	0.0000000	0.0000000	0.00	0.250000
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	0.0000000	0.0000000	0.00	0.000000
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	0.0000000	0.0000000	0.00	0.000000

Some Examples of Weights Matrices



Some Examples of Weights Matrices

W y

	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
```

→ ver link de Tarea

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

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

Weights Matrix in 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

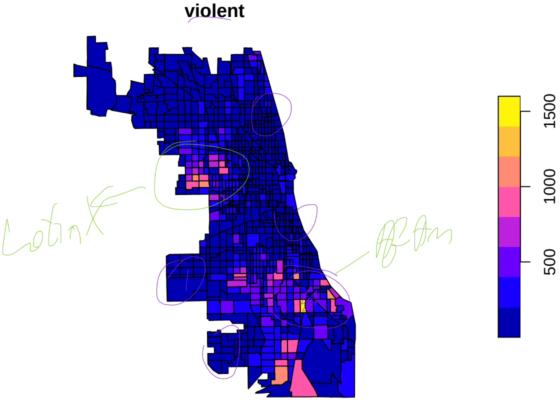
2014 - 2015

```
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

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



Weights Matrix in R

False (Rook)

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

nb2mat → matriz

row stand

incluye a los q' no tienen vecinos

```
## 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
```

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

Weights Matrix in 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
```

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

Traditional Spatial Econometrics

Spatial Autoregressive (SAR) Models

TS $y_t = y_{t-1} + \epsilon_t$

$y_t = L y_t + \epsilon_t$

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

$$y = \rho \underbrace{Wy}_{Ly} + X\beta + \epsilon$$

$y = X\beta + \epsilon$

$= f(X)$

- 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 ϵ_i
- OLS estimates in the non spatial model will be biased and inconsistent. (Anselin and Bera, 1998)
- The estimation of the SAR model can be approached in two ways.
  1. Assume normality of the error term and use maximum likelihood.
  2. Use 2SLS
- In R the function `lagsarlm` uses MLE

Prediction with SAR Models

- The usual *prolegomena*

```
set.seed(101010) #sets a seed
#70% train
indic<-sample(1:nrow(chi.poly),floor(.7*nrow(chi.poly)))

#Partition the sample
train<-chi.poly[indic,]
test<-chi.poly[-indic,]


ols<-lm(violent~est_fcs_rt+bls_unemp, data=train)
test$yhat<-predict(ols,newdata=test)
mean((test$violent-test$yhat)^2)
```

```
## [1] 29773.64
```

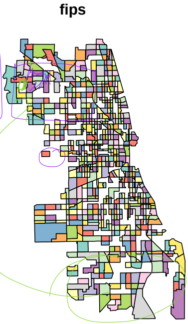
Prediction with SAR Models

- Modeling the spatial structure with a SAR Model

```
list.queen_train<-poly2nb(train, queen=TRUE)
W_train<-nb2listw(list.queen_train, style="W", zero.policy=TRUE)
W_train

Error in print.listw(x) : regions with no neighbours found, use zero.policy=TRUE

plot(train["fips"])
```



30%

70%

Prediction with SAR Models

- Use distance instead

```
coords <- st_centroid(st_geometry(train), of_largest_polygon=TRUE)
W_train<-dnearneigh(coords,0,4300)
W_train<-nb2listw(W_train, style="W", zero.policy=TRUE)



coords <- st_centroid(st_geometry(test), of_largest_polygon=TRUE)
W_test<-dnearneigh(coords,0,4300)
W_test<-nb2listw(W_test, style="W", zero.policy=TRUE)



require("spatialreg")

sar.chi<-lagsarlm(violent~est_fcs_rt+bls_unemp, data=train, W_train)

test$yhat_sar<-predict(sar.chi,newdata=test,listw=W_test)
```

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

ρ β

W weights matrix

Prediction with SAR Models

$y = f(x) + \varepsilon$

- Comparing to OLS

```
mean((test$violent-test$yhat)^2)
```

```
## [1] 29773.64
```

```
mean((test$violent-test$yhat_sar)^2)
```

```
## [1] 28662.23
```

Review & Next Steps

- Today:
  - Closeness
  - Weights Matrix
  - Examples of Weight Matrices Weights Matrix in R
  - Traditional Spatial Regressions
  - Prediction with SAR Models
- Next class: More on Spatial Regressions
- Questions? Questions about software?

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.