Exploratory Spatial Data Analysis (ESDA) with R

Juan Tomás Sayago Gómez

Análisis espacial'
Universidad Icesi - Cali
April 23, 2018

1. Introduction to ESDA
2. Spatial Autocorrelation
   - Issues to consider
   - Hypothesis
   - Tests for the presence of Spatial autocorrelation
   - Spatial weights matrices
   - Global Spatial Autocorrelation
     - Moran's I
     - Geary's C
     - Moran's Scatterplot
   - Local Spatial Autocorrelation
3. ESDA using R

What is Exploratory Spatial Data Analysis?

"ESDA is a collection of techniques to describe and visualize spatial distributions, identify atypical locations or spatial outliers, discover patterns of spatial association, clusters or hot-spots, and suggest spatial regimes or other forms of spatial heterogeneity. Central to this conceptualization is the notion of spatial autocorrelation or spatial association, i.e., the phenomenon where locational similarity (observations in spatial proximity) is matched by value similarity (attribute correlation)" Anselin(1998,79-80)

Introduction to Exploratory Data Analysis (EDA)

Graphical and visual methods or tools that are used to identify data properties for purposes of

- Detect patterns in the data
- Formulate hypothesis from the data
- Assess various aspects of models (e.g., goodness-of-fit)

Introduction to Exploratory Spatial Data Analysis (ESDA)

In many instances it is important to be able to link numerical and graphical procedures with a map to answer questions such as ”Where are those cases?”

Introduction to Exploratory Data Analysis (EDA)

EDA emphasizes on the **interaction** between **human cognition** and **computation** in the form of dynamic statistical graphics that allow the user to manipulate "views" of the data (via box-plots, scatterplots, etc).

EDA also uses statistics to identify outliers and detect patterns and other characteristics in the data.

Introduction to Exploratory Spatial Data Analysis (ESDA)

It is an extension of EDA to detect spatial properties of data.

There is a need for additional techniques to

- detect spatial patterns in data
- formulate hypotheses based on the geography of the data
- assess spatial models

Introduction to Exploratory Spatial Data Analysis (ESDA)

First example of ESDA by Mark Monmonier (1989) Interactive Spatial Data Analysis

With modern graphical interfaces this is often done by "brushing" or "highlighting" observations

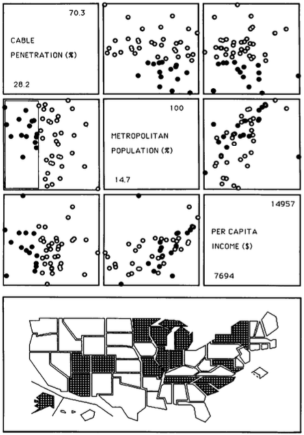
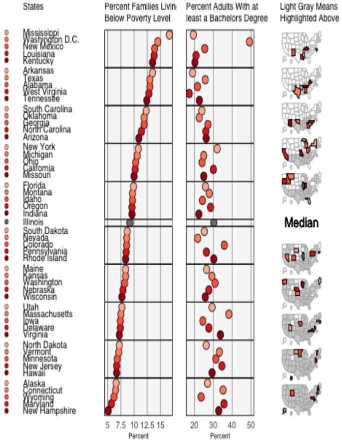


FIG. 2. Scatterplot array with map added to highlight states selected by the brush, which now identifies states with the 15 lowest rates of cable penetration.

Introduction to Exploratory Spatial Data Analysis (ESDA)

These geographic brushing tools are now found in some GIS packages. In addition to choropleth maps, other displays/views include histograms, box-plots, scatterplots (see GeoDa or the Micromaps package).



Introduction to Exploratory Spatial Data Analysis (ESDA)

"True ESDA pays attention to both spatial and attribute association" (Anselin, 1998, p.79).

Spatial Autocorrelation

Tobler's First Law of Geography

"All places are related but nearby places are more related than distant place"
Spatial autocorrelation is the formal property that measures the degree to which near and distant things are related

Issues to consider

We remember the types of data

- Point data:
  Accuracy of location is very important
- Area/lattice dat:
  Data reported for some regular or irregular areal unit

Issues to consider

There are 2 key components of spatial data:

- Attribute data
- Spatial data

And How you choose to sample, or aggregate, your data is very important

Affect spatial relationships in your data

Issues to consider

There are problems/challenges with areal data

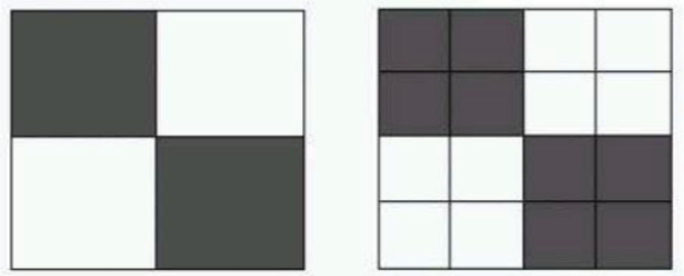
- Modifiable areal unit problem (MAUP)
- Scale effect spatial data analysis at different scales may produce different results
- Zoning effect regrouping zones at a given scale may produce different results

What are the ?

- Optimal neighborhood/area size.
- Alternative zoning schemes.

Issues to consider

Spatial autocorrelation is Scale-Dependent



Spatial Autocorrelation

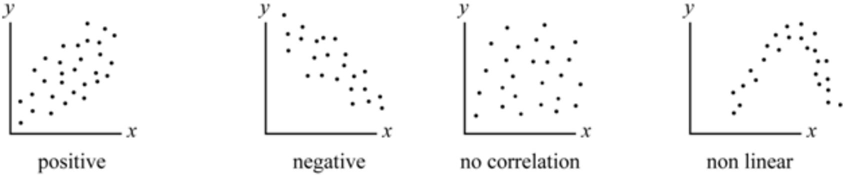
Statistical test of match between locational similarity and attribute similarity

Correlation between

- Positive
- Negative
- Random (Absense)

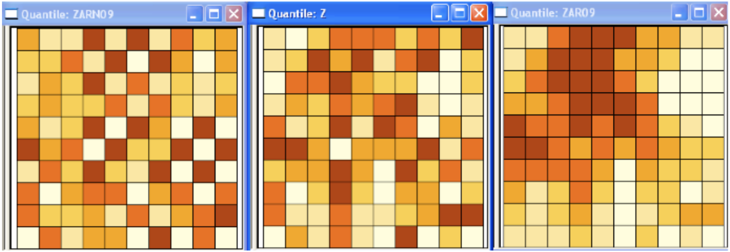
Spatial Autocorrelation

Let's remember from statistics class.



Spatial Autocorrelation

We need to consider how spatially correlated variables can be:



Which is negative? Which is positive? Which is random?

Spatial Autocorrelation

Let's remember statistics again:

Null hypothesis:

- Spatial randomness
- Values observed at one location do not depend on values observed at neighboring locations
- Observed spatial pattern of values is equally likely as any other spatial pattern
- The location of values may be altered without affecting the information content of the data

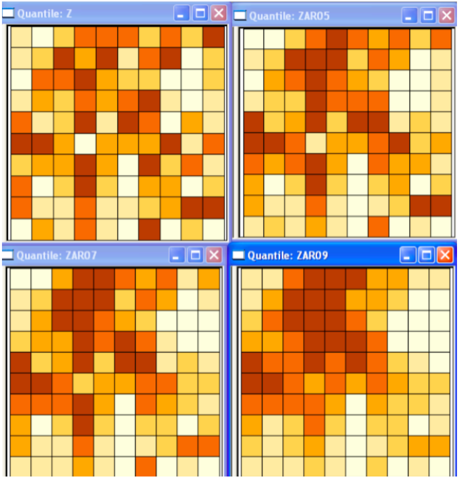
Spatial Autocorrelation

Positive spatial Autocorrelation

- Clustering:
  like values tend to be in similar locations
- Neighbors are Similar
  more alike than they would be under spatial randomness
- Compatible with Diffusion:
  but not necessarily caused by diffusion

Spatial Autocorrelation

Positive spatial Autocorrelation



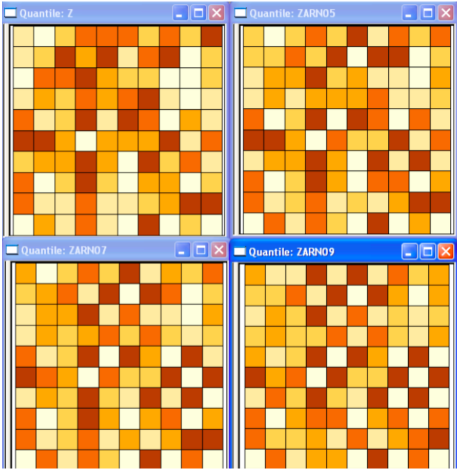
Spatial Autocorrelation

Negative spatial Autocorrelation

- Checkerboard Pattern
- “opposite” of clustering
- Neighbors are Dissimilar, but more dissimilar than they would be under spatial randomness
- Compatible with Competition, but not necessarily competition

Spatial Autocorrelation

Negative spatial Autocorrelation



Spatial Autocorrelation

Positive spatial Autocorrelation

- Positive Spatial Autocorrelation Does Not Imply Diffusion
- diffusion tends to yield positive spatial autocorrelation, but the reverse is not necessary
- spatial correlation may be due to structural factors, without contagion or diffusion

Spatial Autocorrelation

Positive spatial Autocorrelation

- What is the Cause Behind Clustering
- True Contagion is the result of a contagious process, social interaction, a dynamic process.
- Apparent Contagion could be due to spatial heterogeneity, stratification
- It Cannot Be Distinguished In a Pure Cross Section

Global Spatial Autocorrelation Tests

Measure Degree of Clustering
Global Characteristic of Spatial Pattern - NOT Local

- are like values more grouped in space than random
- property of overall pattern = all the observations
- test by means of a global spatial autocorrelation statistic
- One statistic to summarize pattern in whole study area

Local Spatial Autocorrelation

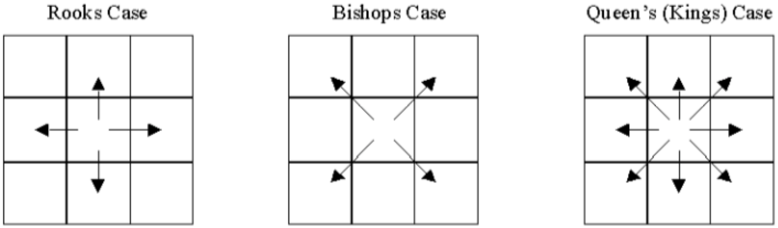
Measure presence of Clusters or Hot-spots
Local Characteristic of Spatial Pattern

- where are like values more grouped in space than random
- property of local pattern = location-specific
- test by means of a local spatial autocorrelation statistic
- local clusters may be compatible with global spatial randomness
- Location-specific statistics

Spatial weights matrices

- Neighborhoods can be defined in a number of ways Contiguity (common boundary)
- What is a "shared" boundary?
- Distance (distance band, K-nearest neighbors)
- How many "neighbors" to include, what distance do we use? General weights (social distance, distance decay)
- Other measures: Economic, social, political, trade.

Spatial weights matrices



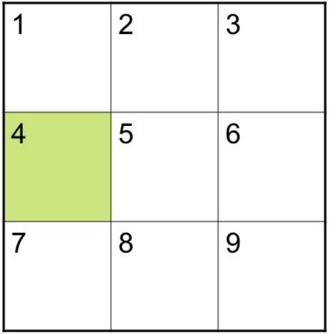
Spatial weights matrices

- Most common is using binary connectivity based on contiguity
  $w_{ij} = 1$ if regions i and j are contiguous,
  $w_{ij} = 0$ otherwise
- May also be defined as a function of the distance between i and j
- Distance of the line connecting the centroids of two areas
  $w_{ij} = d_{ij}^{-\beta}$
  $w_{ij} = exp^{(-\beta d_{ij})}$

Spatial weights matrices

Weights Matrix Example

Sample Region and Units

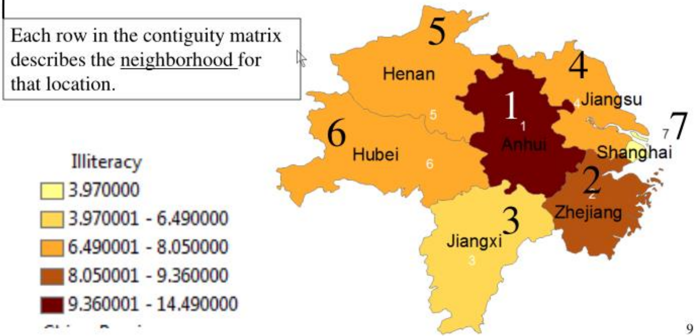


Simple Contiguity (rook) Matrix

	1	2	3	4	5	6	7	8	9
1	0	1	0	1	0	0	0	0	0
2	1	0	1	0	1	0	0	0	0
3	0	1	0	0	0	1	0	0	0
4	1	0	0	0	1	0	1	0	0
5	0	1	0	1	0	1	0	1	0
6	0	0	1	0	1	0	0	0	1
7	0	0	0	1	0	0	0	1	0
8	0	0	0	0	1	0	1	0	1
9	0	0	0	0	0	1	0	0	0

Spatial weights matrices

Contiguity Matrix	Code	1 Anhui	2 Zhejiang	3 Jiangxi	4 Jiangsu	5 Henan	6 Hubei	7 Shanghai	Sum	Neighbors	Illiteracy
Anhui	1	0	1	1	1	1	1	0	5	6 5 4 3 2	**14.49**
Zhejiang	2	1	0	1	1	0	0	1	4	7 4 3 1	**9.36**
Jiangxi	3	1	1	0	0	0	1	0	3	6 2 1	**6.49**
Jiangsu	4	1	1	0	0	0	0	1	3	7 2 1	**8.05**
Henan	5	1	0	0	0	0	1	0	2	6 1	**7.36**
Hubei	6	1	0	1	0	1	0	0	3	1 3 5	**7.69**
Shanghai	7	0	1	0	1	0	0	0	2	2 4	**3.97**



Spatial weights matrices

- A binary weights matrix looks like:

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

- Observation 1 has neighbor 2
- Observation 2 has neighbors 3 and 4
- Observation 3 has neighbors 1 and 2
- Observation 4 has neighbor 2, 3 and 4
- A row-standardized matrix it looks like:

$$W = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0 & 0 \\ 0 & .33 & .33 & .33 \end{pmatrix} \tag{2}$$

Spatial Autocorrelation Statistics

- Formal Test of Match Between Value Similarity and Locational Similarity
- Statistic Summarizes Both Aspects
- Significance
  how likely is it (p-value) that the computed statistic would take this (extreme) value in a spatially random pattern

Spatial Autocorrelation Statistics

Tests for the presence of spatial autocorrelation

- Global
  - Moran's I
  - Geary's C
- Local (LISA – Local Indicators of Spatial Autocorrelation)
  - Local Moran's
  - Getis G_i^*
- Other tests that are more simple:
  - The Chi-square Test for Spatial Independence
  - The Join Count Statistic

Moran's Index

- Classic/best measure of spatial autocorrelation
- Depends upon definition of neighboring unit via the spatial weights matrix
- Typically ranges from -1 to 1
- Like regression, it has a few assumptions
  - Regional x/y values all come from normal distributions w/same mean and variance for each region
  - Randomly rearrange the data on map and compute I many times, would have a normal distribution
  - Why? Because we use the normal distribution to calculate the p-value

Moran's Index

$$I = \frac{n \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{(\sum_{i=1}^{n}(y_i - \bar{y})^2)(\sum \sum_{i \neq j} w_{ij})} \tag{3}$$

$n =$ number of regions.
$W_{ij} =$ Measure of Spatial proximity between region i and j

Moran's Index

Product of the deviation from the mean for all pairs of adjacent regions (w_{ij}=1)

$$I = \frac{n\sum_{i=1}^{n}\sum_{i=j}^{n} w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\left(\sum_{i=1}^{n}(y_i - \bar{y})^2\right)\left(\sum\sum_{i\neq j} w_{ij}\right)}$$

Essentially a measure of variance across the regions

Sum of the weights (count of all adjacent pairs)

Moran's Index

Inference

- Normal: assume uncorrelated normal distribution
- Z-Value: compute $E(I)$ and $Var(I)$
  $z = (I - E(I))/SD(I)$ Regular Standarization.

Moran's Index

- For Significant Statistics Only
- Use z-value
- The result Index depends on W
- Positive S.A.:
  $zI > 0$ for $p < 0.05$
- no distinction between clustering of high or low values
- Negative S.A.:
  $zI < 0$ for $p < 0.05$

Moran's Index

- Variance Instability of Rates
  - non-constant variance violates assumption of stationarity
  - may lead to spurious indication of spatial autocorrelation
- Empirical Bayes Adjustment (Assuncao-Reis)
  - standardize each rate
  - use standardized rate

Moran's Index

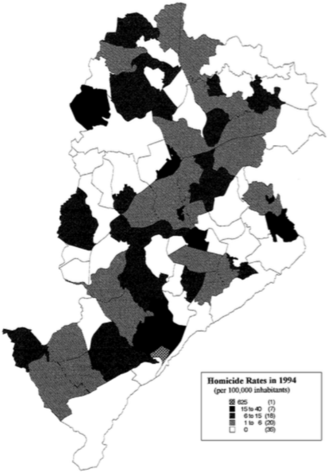


Figure 1. Map of homicide rates (per 100,000). Geographical units are the 81 planning units (UP) of Belo Horizonte, MG, Brazil, in 1994. The data were obtained from the Polícia Militar de Minas Gerais

Geary's Coefficient

- Geary's C typically ranges from 0 to 3
- It Cannot be **Negative**
- A Spatially uncorrelated process has an expected $C = 1$
- Values less than 1 indicate positive spatial autocorrelation
- Values greater than 1 indicate negative autocorrelation

Geary's Coefficient

$$I = \frac{(n-1)\sum_{i=1}^{n}\sum_{i=j}^{n} w_{ij}(y_i - y_j)^2}{2\left(\sum_{i=1}^{n}(y_i - \bar{y})^2\right)\left(\sum\sum_{i \neq j} w_{ij}\right)}$$

Geary's Coefficient

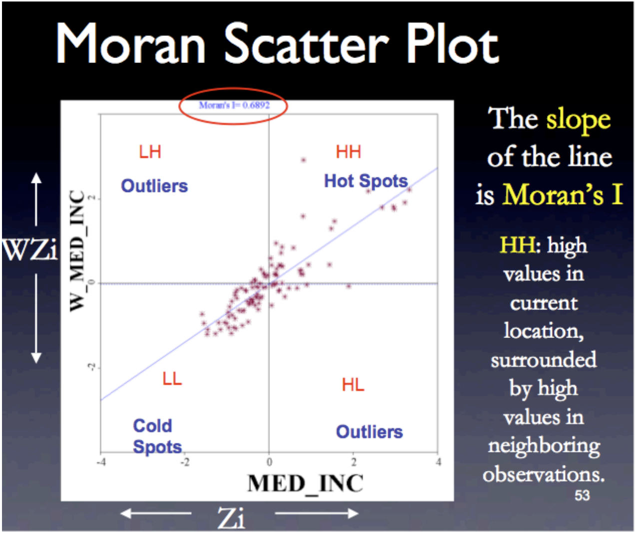
Relationship of Moran's I and Geary's C

- C approaches 0 and I approaches 1 when similar values are clustered
- C approaches 3 and I approaches -1 when dissimilar values tend to cluster
- High values of C measures correspond to low values of I
- So the two measures are inversely related

Moran's Scatterplot

- Moran's I as a Regression Slope
  - in matrix notation:
    $I = \dfrac{z'Wz}{z'z}$
  - The Moran's I is the **slope** in regression
    $Wz = a + I \times z$
- Moran Scatter Plot
  - linear association between Wz on the y-axis and z on the x-axis
  - each point is pair (z_i, Wz_i), slope is Moran's I

Moran's Scatterplot



Moran's Scatterplot

Link with Local Spatial Autocorrelation

- Four Categories of SA
- Positive Spatial Autocorrelation
  high-high and low-low: spatial clusters
- Negative Spatial Autocorrelation
  high-low and low-high: spatial outliers
- Only Suggestive
  no suggestion of significance
  relative to mean

Local Indicators of Spatial Autocorrelation – LISA

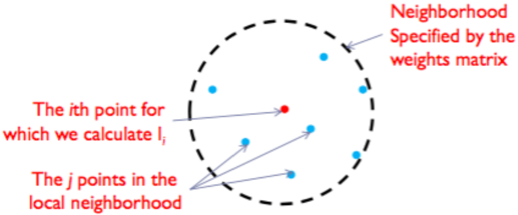
- Anselin (1995)
- Local Spatial Statistic
  indicate significant spatial autocorrelation for each location
- Local-Global Relation
  sum of LISA proportional to a corresponding global indicator of spatial autocorrelation

LISA- Local Moran's I

- Used to determine if local autocorrelation exists around each region Index
- Returns an Index value for each region
- Often used after global Moran's Index to see if:
- The study area is homogeneous (local statistics similar across regions)
- There are local outliers that contribute to a significant global statistic

LISA- Local Moran's I

$$I_i = \frac{n(y_i - \bar{y})}{\sum_j (y_i - \bar{y})^2} \sum_j w_{ij}(y_i - \bar{y}) \tag{4}$$



Load required library and Data

First We need to install the packages **spdep** and maptools

```
>   install.packages(c('sp','spdep','maptools'))
```

```
The downloaded binary packages are in
        /var/folders/qq/d9kxh5bj3xs7hs77g3fhrzp80000gn/T//Rtmp
```

After installing the libraries we need to load them up

```
>   library(spdep)
> library(maptools)
```

Load the Data

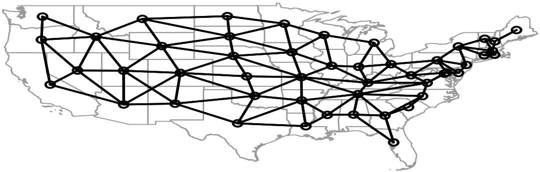
```
> shape <- readShapePoly("Statesmod.shp",IDvar="NAME" )
> summary(shape)
```

```
Object of class SpatialPolygonsDataFrame
Coordinates:
        min       max
x -124.7631 -66.94989
y   24.5231  49.38436
Is projected: NA
proj4string : [NA]
Data attributes:
         SP_ID        STATEFP        STATENS           AFFGEOID       GEOID
 Alabama     : 1   01       : 1   00068085: 1   0400000US01: 1   01      : 1
 Arizona     : 1   04       : 1   00294478: 1   0400000US04: 1   04      : 1
 Arkansas    : 1   05       : 1   00448508: 1   0400000US05: 1   05      : 1
 California  : 1   06       : 1   00481813: 1   0400000US06: 1   06      : 1
 Colorado    : 1   08       : 1   00606926: 1   0400000US08: 1   08      : 1
 Connecticut: 1   09       : 1   00662849: 1   0400000US09: 1   09      : 1
 (Other)     :43   (Other):43   (Other) :43   (Other)     :43   (Other):43
     STUSPS              NAME      LSAD        ALAND                AWATER
 AL      : 1   Alabama     : 1   00:49   Min.   :1.584e+08   Min.   :1.863e+07
 AR      : 1   Arizona     : 1           1st Qu.:9.279e+10   1st Qu.:1.536e+09
 AZ      : 1   Arkansas    : 1           Median :1.389e+11   Median :3.543e+09
 CA      : 1   California : 1           Mean   :1.562e+11   Mean   :8.744e+09
 CO      : 1   Colorado    : 1           3rd Qu.:2.062e+11   3rd Qu.:8.504e+09
 CT      : 1   Connecticut: 1           Max.   :6.766e+11   Max.   :1.040e+11
 (Other):43   (Other)     :43
         State        Violent_Cr        Murder          Forcible_r
 Alabama     : 1   Min.   : 119.4   Min.   : 0.900   Min.   :12.90
 Arizona     : 1   1st Qu.: 299.3   1st Qu.: 3.000   1st Qu.:26.90
 Arkansas    : 1   Median : 385.1   Median : 4.900   Median :32.70
 California : 1   Mean   : 432.1   Mean   : 5.486   Mean   :33.37
```

Create the Weight matrix- Queen

```
> nb2=poly2nb(shape, queen=TRUE)
> nb2

Neighbour list object:
Number of regions: 49
Number of nonzero links: 218
Percentage nonzero weights: 9.07955
Average number of links: 4.44898

> plot(shape, border="grey60")
> plot(nb2, coordinates(shape), add=TRUE, pcv=".", lwd=2)
```

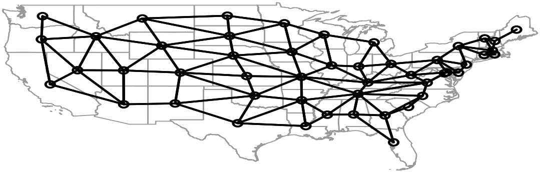
Create the Weight matrix- Queen



Create the Weight matrix- Rook

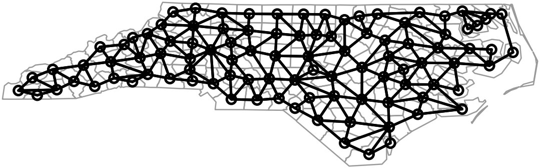
```
> nb2=poly2nb(shape, queen=FALSE)
> nb2

Neighbour list object:
Number of regions: 49
Number of nonzero links: 214
Percentage nonzero weights: 8.912953
Average number of links: 4.367347

> plot(shape, border="grey60")
> plot(nb2, coordinates(shape), add=TRUE, pcv=".", lwd=2)
```

Create the Weight matrix- Rook

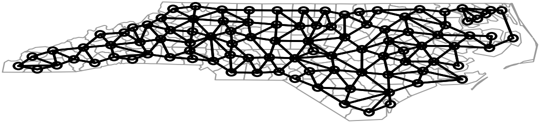


Create the Weight matrix- Rook

```
> sids <- readShapePoly("sids.shp", ID="FIPSNO")
> sids_nbr<-poly2nb(sids,queen=F)
> plot(sids, border="grey60")
> plot(sids_nbr, coordinates(sids), add=TRUE, pcv=".", lwd=2)
```

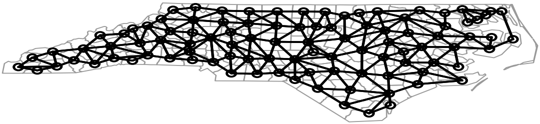


Create the Weight matrix- Rook



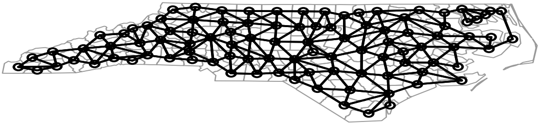
Create the Weight matrix - Queen

```
> sids_nbq<-poly2nb(sids,queen=T)
> plot(sids, border="grey60")
> plot(sids_nbq, coordinates(sids), add=TRUE, pcv=".", lwd=2)
```



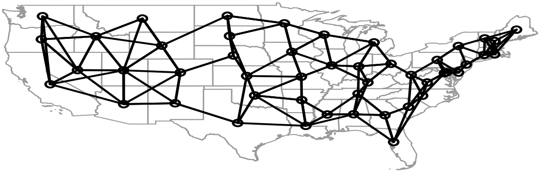
Create the Weight matrix- Comparison

```
> sids_nbq<-poly2nb(sids,queen=T)
>  sids_nbr<-poly2nb(sids,queen=F)
> plot(sids, border="grey60")
> plot(sids_nbq, coordinates(sids), add=TRUE, pcv=".",col="red", lwd=2)
> plot(sids_nbr, coordinates(sids), add=TRUE, pcv=".",col="blue", lwd=2)
```



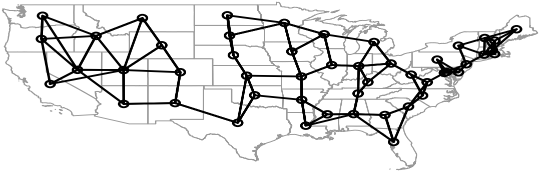
Create the Weight matrix - 4 Knearest Neighbors

```
> coords <- coordinates(shape)
> states.knn <- knearneigh(coords, k=4)
> plot(shape, border="grey60")
> plot(knn2nb(states.knn), coordinates(shape), add=TRUE, pcv=".", lwd=2)
```



Create the Weight matrix - 3 Knearest Neighbors

```
> coords <- coordinates(shape)
> states.knn <- knearneigh(coords, k=3)
> plot(shape, border="grey60")
> plot(knn2nb(states.knn), coordinates(shape), add=TRUE, pcv=".", lwd=2)
```



Binary W matrix or Weighted W matrix

We can standardize the weight matrix by row.

```
> nb2_B=nb2listw(nb2, style="B", zero.policy=TRUE)
> nb2_W=nb2listw(nb2, style="W", zero.policy=TRUE)
```

You can save the weights.

```
> write.nb.gal(nb2,"pesoscol.GAL")
```

Moran's I and Moran's Scatterplot

```
> moran.test(shape$Murder, listw=nb2_W,alternative="two.sided")

        Moran I test under randomisation

data:  shape$Murder
weights: nb2_W

Moran I statistic standard deviate = 3.2206, p-value = 0.001279
alternative hypothesis: two.sided
sample estimates:
Moran I statistic       Expectation          Variance
      0.207123212      -0.020833333       0.005009799
```

Moran's I and Moran's Scatterplot

```
> moran=moran.test(shape$Murder, listw=nb2_B)
> moran

        Moran I test under randomisation

data:  shape$Murder
weights: nb2_B

Moran I statistic standard deviate = 2.7617, p-value = 0.002875
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
      0.163703437      -0.020833333       0.004464824
```

Moran's I and Moran's Scatterplot

Montecarlo Moran's I

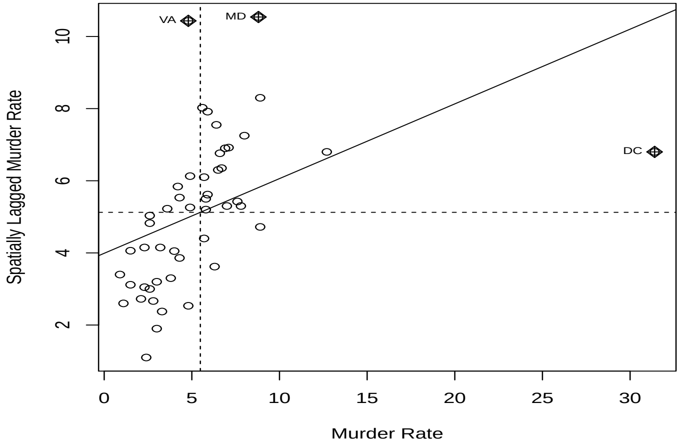
```
> set.seed(1234)
> bperm=moran.mc(shape$Murder, listw=nb2_W, nsim=999)
> bperm

        Monte-Carlo simulation of Moran I

data:  shape$Murder
weights: nb2_W
number of simulations + 1: 1000

statistic = 0.20712, observed rank = 998, p-value = 0.002
alternative hypothesis: greater
```

Geary's Coefficient

```
> gearyR=geary.test(shape$Murder, listw=nb2_W)
> gearyR

        Geary C test under randomisation

data:  shape$Murder
weights: nb2_W

Geary C statistic standard deviate = 3.1328, p-value = 0.0008658
alternative hypothesis: Expectation greater than statistic
sample estimates:
Geary C statistic       Expectation          Variance
       0.55757044        1.00000000        0.01994487
```

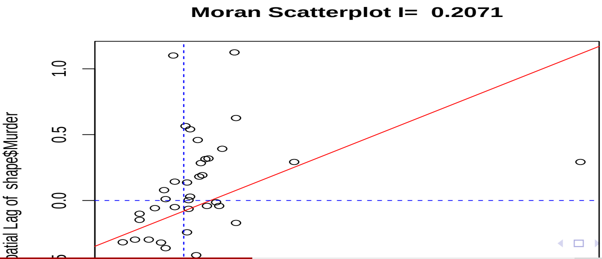
Moran's I and Moran's Scatterplot

```
> moran.plot(shape$Murder, listw=nb2_W,label=as.character(shape$STUSPS),xlab="Murder Rate", ylab="Spatially Lagge
>
```



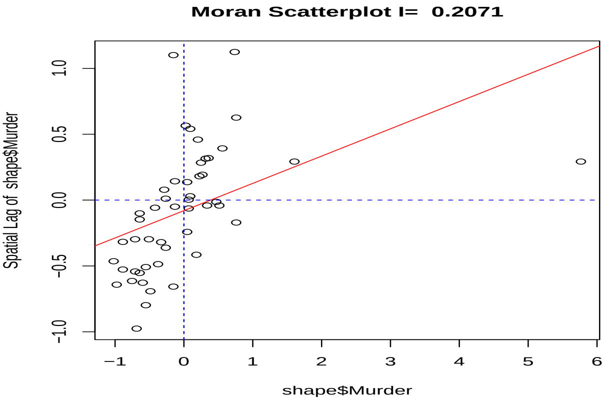
Moran's I and Moran's Scatterplot

```
> moran.plot2 <- function(x,wfile)
+ {
+ xname <- deparse(substitute(x)) # get name of variable
+ zx <- (x - mean(x))/sd(x)
+ wzx <- lag.listw(wfile,zx, zero.policy=TRUE)
+ morlm <- lm(wzx ~ zx)
+ aa <- morlm$coefficients[1]
+ mori <- morlm$coefficients[2]
+ par(pty="s")
+ plot(zx,wzx,xlab=xname,ylab=paste("Spatial Lag of ",xname))
+ abline(aa,mori,col=2)
+ abline(h=0,lty=2,col=4)
+ abline(v=0,lty=2,col=4)
+ title(paste("Moran Scatterplot I= ",format(round(mori,4))))
+ }
> moran.plot2(shape$Murder, nb2_W)
```



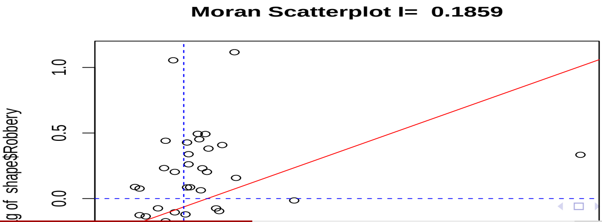
Moran's I and Moran's Scatterplot

```
> moran.plot2(shape$Murder, nb2_W)
```



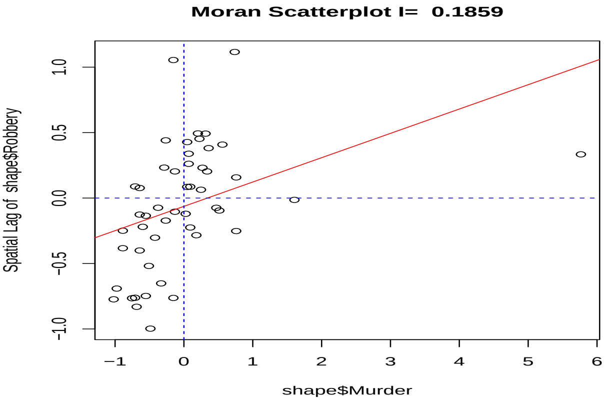
Moran's I and Moran's Scatterplot

```
> moran.plot3 <- function(x,y,wfile)
+ {
+ xname <- deparse(substitute(x)) # get name of variable
+ yname =deparse(substitute(y))
+ zx <- (x - mean(x))/sd(x)
+ zy =(y - mean(y))/sd(y)
+ wzy <- lag.listw(wfile,zy, zero.policy=TRUE)
+ morlm <- lm(wzy ~ zx)
+ aa <- morlm$coefficients[1]
+ mori <- morlm$coefficients[2]
+ par(pty="s")
+ plot(zx,wzy,xlab=xname,ylab=paste("Spatial Lag of ",yname))
+ abline(aa,mori,col=2)
+ abline(h=0,lty=2,col=4)
+ abline(v=0,lty=2,col=4)
+ title(paste("Moran Scatterplot I= ",format(round(mori,4))))
+ }
> moran.plot3(shape$Murder,shape$Robbery, nb2_W)
```



Moran's I and Moran's Scatterplot

```
> moran.plot3(shape$Murder,shape$Robbery, nb2_W)
```



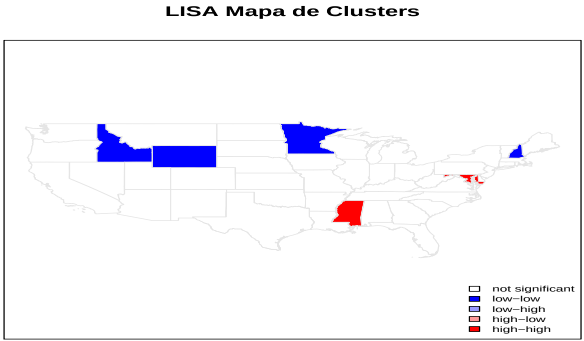
Local Indicators of Spatial Association

```
> LISA.plot <- function(var,listw,signif,mapa) {
+   mI.loc <- localmoran(var,listw, zero.policy=T)
+   c.var <- var - mean(var)
+   c.mI <- mI.loc[,1] - mean(mI.loc[,1])
+   quadrant <- vector(mode="numeric",length=nrow(mI.loc))
+ wzx <- lag.listw(listw,var, zero.policy=TRUE)
+   c.mI1 <- wzx - mean(wzx)
+   quadrant[c.var>0 & c.mI1>0] <- 4
+   quadrant[c.var<0 & c.mI1<0] <- 1
+   quadrant[c.var<0 & c.mI1>0] <- 2
+   quadrant[c.var>0 & c.mI1<0] <- 3
+   quadrant[mI.loc[,5]>signif] <- 0
+   brks <- c(0,1,2,3,4)
+   colors <- c("white","blue",rgb(0,0,1,alpha=0.4),rgb(1,0,0,alpha=0.4),"red")
+   plot(mapa,border="gray90",col=colors[findInterval(quadrant,brks,all.inside=FALSE)])
+   box()
+    legend("bottomright",legend=c("not significant","low-low","low-high","high-low","high-high"), fill=colors,bt
+    title("LISA Mapa de Clusters")
+   }
```

Local Indicators of Spatial Association

Using 0.1 significance threshold.

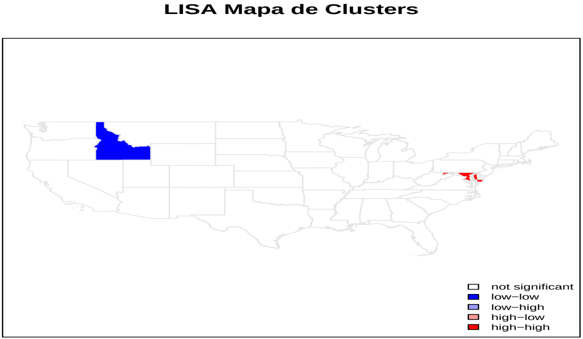
```
> LISA.plot(shape$Murder, nb2_W,0.1, shape)
```



Local Indicators of Spatial Association

Using 0.05 significance threshold.

```
> LISA.plot(shape$Murder, nb2_W,0.05, shape)
```

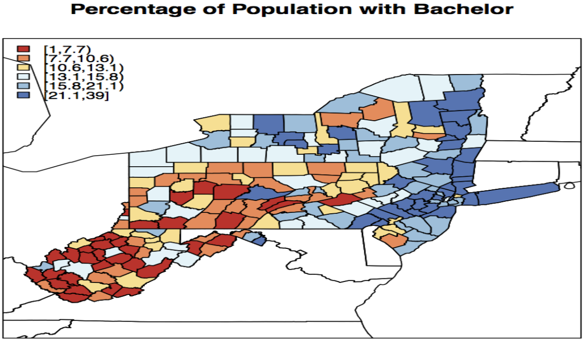


Load the Data - Example with counties

```
> shapec <- readShapePoly("countiesmod.shp",IDvar="GEOID" )
> summary(shapec)
```

```
Object of class SpatialPolygonsDataFrame
Coordinates:
        min       max
x -82.64459 -71.77749
y  37.20154  45.01586
Is projected: NA
proj4string : [NA]
Data attributes:
     SP_ID       STATEFP     COUNTYFP       COUNTYNS        GEOID
 34001   :  1   34:21    001     :  4   00882228:  1   34001   :  1
 34003   :  1   36:62    003     :  4   00882229:  1   34003   :  1
 34005   :  1   42:67    005     :  4   00882230:  1   34005   :  1
 34007   :  1   54:55    007     :  4   00882231:  1   34007   :  1
 34009   :  1            009     :  4   00882232:  1   34009   :  1
 34011   :  1            011     :  4   00882233:  1   34011   :  1
 (Other):199            (Other):181   (Other)  :199   (Other):199
        NAME                   NAMELSAD    LSAD      CLASSFP    MTFCC
 Jefferson:  3   Jefferson County:  3   06:205    H1:199   G4020:205
 Mercer   :  3   Mercer County   :  3             H6:  6
 Monroe   :  3   Monroe County   :  3
 Warren   :  3   Warren County   :  3
 Wayne    :  3   Wayne County    :  3
 Wyoming  :  3   Wyoming County  :  3
 (Other)  :187   (Other)         :187
     CSAFP       CBSAFP       METDIVFP    FUNCSTAT      ALAND
 408     :32   35620  : 25   35614  : 14   A:199    Min.   :5.868e+07
 428     :13   37980  :  9   35084  :  7   C:  6    1st Qu.:9.234e+08
 430     :11   38300  :  7   15804  :  3            Median :1.343e+09
 104     :10   40380  :  6   33874  :  3            Mean   :1.557e+09
```

Load the Data - Example with counties

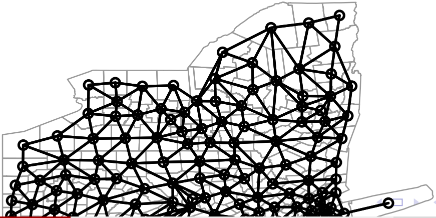


Create the Weight matrix- Queen

```
> nb2c=poly2nb(shapec, queen=TRUE)
> nb2c
```

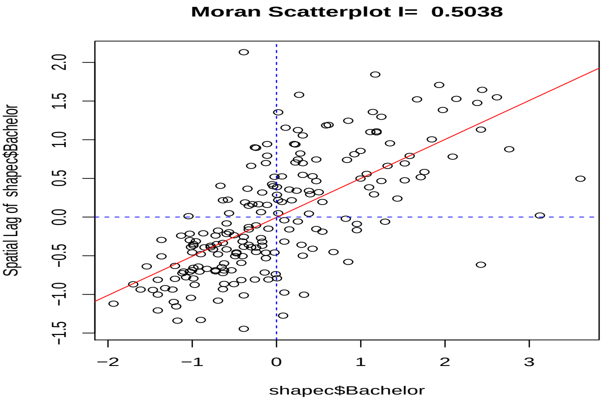
```
Neighbour list object:
Number of regions: 205
Number of nonzero links: 1080
Percentage nonzero weights: 2.569899
Average number of links: 5.268293
```

```
> plot(shapec, border="grey60")
> plot(nb2c, coordinates(shapec), add=TRUE, pcv=".", lwd=2)
>  nb2_Wc=nb2listw(nb2c, style="W", zero.policy=TRUE)
```



Moran's I and Moran's Scatterplot - Counties

```
> moran.plot2(shapec$Bachelor, nb2_Wc)
```

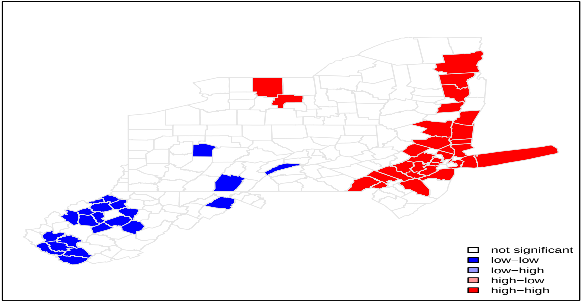


LISA map counties

Using 0.05 significance threshold.

```
> LISA.plot(shapec$Bachelor, nb2_Wc,0.05, shapec)
```

LISA Mapa de Clusters



LISA map counties

Taller hacer Moran y LISA de NBI y Cobertura de energia electrica en Colombia

1. Hacer Indice de Moran, diagrama de dispersión de Moran y LISA para el NBI de Colombia
2. Usando los datos de desplazamiento en el shapefile "$shape_{c}oltesis0708.shp$" hacer un mapa de expulsion, estandarizar la variable, y hacer análisis de autocorrelación espacial.
3. Elegir una variable y hacer repetir el análisis.