

# Package: spdepbi

*Student: Alexys H Rodríguez A - Subject SDCWR - WWU*

*August 6 - 2019. Professor Edzer Pebesma*

## Contents

<b>1. Introduction</b>	<b>1</b>
1.1. Motivation . . . . .	1
1.2. Package Warnings . . . . .	2
<b>2. Data Description &amp; Exploration</b>	<b>7</b>
<b>3. Data Analysis</b>	<b>11</b>
3.1. Bivariate Moran's I (moran.bi) . . . . .	11
3.2 Formal Test for Bivariate Moran's I (moranbi.test) . . . . .	12
3.3. Plot of Bivariate Spatial Autocorrelation (moranbi.plot) . . . . .	15
3.4. Local Bivariate Spatial Autocorrelation with Moran's I (localmoran.bi) . . . . .	16
3.5. LISA Spatial Autocorrelation and Significance Map (moran.cluster) . . . . .	17
3.6. Getis Gi* Spatial Autocorrelation and Significance Map (getis.cluster) . . . . .	20
<b>4. Discussion</b>	<b>22</b>
<b>References</b>	<b>22</b>

Acknowledgment: Work done in collaboration with Melo (2016). Contact information: Alexys H Rodríguez, Carlos E Melo

## 1. Introduction

Anselin (1995) and Ord and Getis (1995) introduced the classic exploratory data analysis techniques to test local indicators of spatial association - LISA, showing its relation with global indicators, such as Moran's I Moran (1950), in the univariate context. This package has been create to deal with global and local indicators of spatial association, but in the bivariate context.

### 1.1. Motivation

Despite it is possible to find software examples of bivariate spatial autocorrelation in areal data, as Dray et al. (2019), there is not common to find R package focused in bivariate context. The package 'spdepbi' is a contribution in this regard. It is important to analyze mutual influences of variables to analyze spatial dependences in lattice data.

## 1.2. Package Warnings

The only one warning that the package ‘spdepbi’ send at compiling time will fix itself automatically when packages ‘spdep’ and ‘spatialreg’ complete their migration process, see Bivan (2019).

```
library(spdepbi)
#> Registered S3 methods overwritten by 'spatialreg':
#>   method                      from
#> residuals.stsls              spdep
#> deviance.stsls              spdep
#> coef.stsls                  spdep
#> print.stsls                 spdep
#> summary.stsls              spdep
#> print.summary.stsls        spdep
#> residuals.gmsar            spdep
#> deviance.gmsar            spdep
#> coef.gmsar                 spdep
#> fitted.gmsar               spdep
#> print.gmsar                spdep
#> summary.gmsar             spdep
#> print.summary.gmsar       spdep
#> print.lagmess              spdep
#> summary.lagmess           spdep
#> print.summary.lagmess     spdep
#> residuals.lagmess         spdep
#> deviance.lagmess          spdep
#> coef.lagmess              spdep
#> fitted.lagmess            spdep
#> logLik.lagmess            spdep
#> fitted.SFResult           spdep
#> print.SFResult            spdep
#> fitted.ME_res             spdep
#> print.ME_res              spdep
#> print.lagImpact           spdep
#> plot.lagImpact            spdep
#> summary.lagImpact         spdep
#> HPDinterval.lagImpact    spdep
#> print.summary.lagImpact   spdep
#> print.sarlm               spdep
#> summary.sarlm             spdep
#> residuals.sarlm           spdep
#> deviance.sarlm            spdep
#> coef.sarlm                spdep
#> vcov.sarlm                spdep
#> fitted.sarlm              spdep
#> logLik.sarlm              spdep
#> anova.sarlm               spdep
#> predict.sarlm             spdep
#> print.summary.sarlm       spdep
#> print.sarlm.pred          spdep
#> as.data.frame.sarlm.pred spdep
#> residuals.spautolm        spdep
#> deviance.spautolm         spdep
#> coef.spautolm             spdep
```

```

#> fitted.spautolm      spdep
#> print.spautolm      spdep
#> summary.spautolm    spdep
#> logLik.spautolm     spdep
#> print.summary.spautolm spdep
#> print.WXImpact      spdep
#> summary.WXImpact    spdep
#> print.summary.WXImpact spdep
#> predict.SLX         spdep
#> Warning: replacing previous import 'spatialreg::coef.sarlm' by
#> 'spdep::coef.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::HPDinterval.lagImpact' by
#> 'spdep::HPDinterval.lagImpact' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::residuals.spautolm' by
#> 'spdep::residuals.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::cheb_setup' by
#> 'spdep::cheb_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::eigenw' by 'spdep::eigenw'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::griffith_sone' by
#> 'spdep::griffith_sone' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::coef.gmsar' by
#> 'spdep::coef.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.summary.stsls' by
#> 'spdep::print.summary.stsls' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::get.coresOption' by
#> 'spdep::get.coresOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.summary.spautolm' by
#> 'spdep::print.summary.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.summary.sarlm' by
#> 'spdep::print.summary.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::get.ZeroPolicyOption' by
#> 'spdep::get.ZeroPolicyOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.gmsar' by
#> 'spdep::print.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::fitted.SFResult' by
#> 'spdep::fitted.SFResult' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::as_dsCMatrx_I' by
#> 'spdep::as_dsCMatrx_I' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::LU_prepermute_setup' by
#> 'spdep::LU_prepermute_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.ME_res' by
#> 'spdep::print.ME_res' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::logLik.sarlm' by
#> 'spdep::logLik.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::subgraph_eigenw' by
#> 'spdep::subgraph_eigenw' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::fitted.gmsar' by
#> 'spdep::fitted.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::LR1.sarlm' by
#> 'spdep::LR1.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::fitted.ME_res' by
#> 'spdep::fitted.ME_res' when loading 'spdepbi'

```

```

#> Warning: replacing previous import 'spatialreg::deviance.sarlm' by
#> 'spdep::deviance.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::set.mcOption' by
#> 'spdep::set.mcOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::moments_setup' by
#> 'spdep::moments_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::Matrix_setup' by
#> 'spdep::Matrix_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::intImpacts' by
#> 'spdep::intImpacts' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::l_max' by 'spdep::l_max'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::errorsarlm' by
#> 'spdep::errorsarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::deviance.spautolm' by
#> 'spdep::deviance.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::sacsarlm' by
#> 'spdep::sacsarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::bptest.sarlm' by
#> 'spdep::bptest.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::eigen_pre_setup' by
#> 'spdep::eigen_pre_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::impacts' by
#> 'spdep::impacts' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::stsls' by 'spdep::stsls'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::GMargminImage' by
#> 'spdep::GMargminImage' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::SE_classic_setup' by
#> 'spdep::SE_classic_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::GMerrorsar' by
#> 'spdep::GMerrorsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.stsls' by
#> 'spdep::print.stsls' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::set.ClusterOption' by
#> 'spdep::set.ClusterOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::can.be.simmmed' by
#> 'spdep::can.be.simmmed' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::Jacobian_W' by
#> 'spdep::Jacobian_W' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::spBreg_sac' by
#> 'spdep::spBreg_sac' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::deviance.stsls' by
#> 'spdep::deviance.stsls' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::ME' by 'spdep::ME' when
#> loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::set.ZeroPolicyOption' by
#> 'spdep::set.ZeroPolicyOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::get.VerboseOption' by
#> 'spdep::get.VerboseOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::do_ldet' by
#> 'spdep::do_ldet' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::deviance.gmsar' by

```

```

#> 'spdep::deviance.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::create_WX' by
#> 'spdep::create_WX' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::predict.SLX' by
#> 'spdep::predict.SLX' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::fitted.sarlm' by
#> 'spdep::fitted.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::coef.stsls' by
#> 'spdep::coef.stsls' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::mom_calc_int2' by
#> 'spdep::mom_calc_int2' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::mom_calc' by
#> 'spdep::mom_calc' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::mcdet_setup' by
#> 'spdep::mcdet_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::residuals.stsls' by
#> 'spdep::residuals.stsls' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.sarlm' by
#> 'spdep::print.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::SpatialFiltering' by
#> 'spdep::SpatialFiltering' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::anova.sarlm' by
#> 'spdep::anova.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::logLik.spautolm' by
#> 'spdep::logLik.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::get.mcOption' by
#> 'spdep::get.mcOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::set.coresOption' by
#> 'spdep::set.coresOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::SE_interp_setup' by
#> 'spdep::SE_interp_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::get.ClusterOption' by
#> 'spdep::get.ClusterOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::LR1.spautolm' by
#> 'spdep::LR1.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::jacobianSetup' by
#> 'spdep::jacobianSetup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::coef.spautolm' by
#> 'spdep::coef.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::lagsarlm' by
#> 'spdep::lagsarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::Matrix_J_setup' by
#> 'spdep::Matrix_J_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.SFResult' by
#> 'spdep::print.SFResult' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.spautolm' by
#> 'spdep::print.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::LR.sarlm' by
#> 'spdep::LR.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::lmSLX' by 'spdep::lmSLX'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::Wald1.sarlm' by
#> 'spdep::Wald1.sarlm' when loading 'spdepbi'

```

```

#> Warning: replacing previous import 'spatialreg::spam_update_setup' by
#> 'spdep::spam_update_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.summary.gmsar' by
#> 'spdep::print.summary.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::set.VerboseOption' by
#> 'spdep::set.VerboseOption' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::powerWeights' by
#> 'spdep::powerWeights' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::spBreg_err' by
#> 'spdep::spBreg_err' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::residuals.gmsar' by
#> 'spdep::residuals.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::spautolm' by
#> 'spdep::spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::spBreg_lag' by
#> 'spdep::spBreg_lag' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::residuals.sarlm' by
#> 'spdep::residuals.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::spam_setup' by
#> 'spdep::spam_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::Hausman.test' by
#> 'spdep::Hausman.test' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::trW' by 'spdep::trW' when
#> loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::print.sarlm.pred' by
#> 'spdep::print.sarlm.pred' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::fitted.spautolm' by
#> 'spdep::fitted.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::summary.stsls' by
#> 'spdep::summary.stsls' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::eigen_setup' by
#> 'spdep::eigen_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::lextrB' by 'spdep::lextrB'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::summary.gmsar' by
#> 'spdep::summary.gmsar' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::as_dgRMATRIX_listw' by
#> 'spdep::as_dgRMATRIX_listw' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::gstsls' by 'spdep::gstsls'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::as_dsCMATRIX_IrW' by
#> 'spdep::as_dsCMATRIX_IrW' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::SE_whichMin_setup' by
#> 'spdep::SE_whichMin_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::MCMCsamp' by
#> 'spdep::MCMCsamp' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::vcov.sarlm' by
#> 'spdep::vcov.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::lextrS' by 'spdep::lextrS'
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::similar.listw' by
#> 'spdep::similar.listw' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::lextrW' by 'spdep::lextrW'

```



```
#> when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::as_dsTMATRIX_listw' by
#> 'spdep::as_dsTMATRIX_listw' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::LU_setup' by
#> 'spdep::LU_setup' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::lagmess' by
#> 'spdep::lagmess' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::summary.spautolm' by
#> 'spdep::summary.spautolm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::predict.sarlm' by
#> 'spdep::predict.sarlm' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::as.spam.listw' by
#> 'spdep::as.spam.listw' when loading 'spdepbi'
#> Warning: replacing previous import 'spatialreg::summary.sarlm' by
#> 'spdep::summary.sarlm' when loading 'spdepbi'
```

## 2. Data Description & Exploration

The 49 neighbourhoods in Columbus, OH (1980) dataset with data related to crime, housing and income, will be used for the analysis, see Anselin (1988). The variables of interest are HOVAL, INCOME (INC) and CRIME, namely housing value (in \$1,000), household income (in \$1,000), and residential burglaries and vehicle thefts per 1000 households respectively. The spatial reference system is unknown.

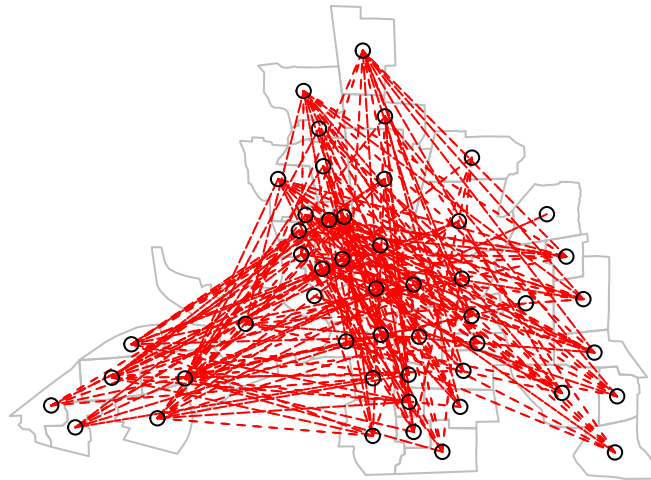
```
library(sf)
#> Linking to GEOS 3.6.1, GDAL 2.2.3, PROJ 4.9.3
library(spdep)
#> Loading required package: sp
#> Loading required package: spData
#> To access larger datasets in this package, install the spDataLarge
#> package with: `install.packages('spDataLarge',
#> repos='https://nowosad.github.io/drat/', type='source')`
columbus <- st_read(system.file("extdata/columbus.shp", package="spdepbi")[1], quiet=TRUE)
```

The exploratory data analysis (EDA) of areal data include the creation of spatial neighbours information and its related spatial weights Edzer J. Pebesma (2008). Spatial neighbours lists hold a relationship of the areas that are in the vicinity of other areas, and can be done taking into consideration the order or closeness. First order are the closer ones polygons, namely the first neighbourhood ring, and as the order increases, subsequent neighbourhood rings (increased radius - farther distances) are created. There are two typical styles to construct neighbour information for areal entities, rook contiguity (common edge), and queen contiguity (common edge or common vertex), see Anselin, Syabri, and Kho (2006). Weights are related with distance between neighbour links, if distances are short weights increase, showing a measure of spatial dependence reflecting Tobler's first law. There are many ways to assigns weights to neighbours lists, see Tiefelsdorf and Friedman (2000), and 'nb2listw' in 'spdep' package, Bivand and Wong (2018). Used format of spatial data in this case study is simple features, see Pebesma (2018).

For the case study data, next code will construct the neighbours list with queen style (poly2nb), then will calculate from 1 to 6 orders of neighbours (nblag), followed by the plot of the areal units borders, and finally will join with a red line all the polygon's centroids on the fifth neighbourhood ring (order).

```
coords = st_geometry(st_centroid(columbus$geometry))
col_nbq <- poly2nb(columbus) # Queen-Style
par.lags1 <- nblag(col_nbq, 6) # Order 6
```

```
plot(st_geometry(columbus), border="grey")
plot(par.lags1[[5]], coords, add=TRUE, col="red", lty=2)
```



Next code, use row standardized (W) method to assign weights to the previous queen-style (col\_nbq) neighbours list and shows a summary of the neighbours-weights.

```
a.lw <- nb2listw(col_nbq, style="W")
summary(a.lw)
```

*#> Characteristics of weights list object:*  
*#> Neighbour list object:*  
*#> Number of regions: 49*  
*#> Number of nonzero links: 236*  
*#> Percentage nonzero weights: 9.829238*  
*#> Average number of links: 4.816327*  
*#> Link number distribution:*  
*#>*  

2	3	4	5	6	7	8	9	10
5	9	12	5	9	3	4	1	1

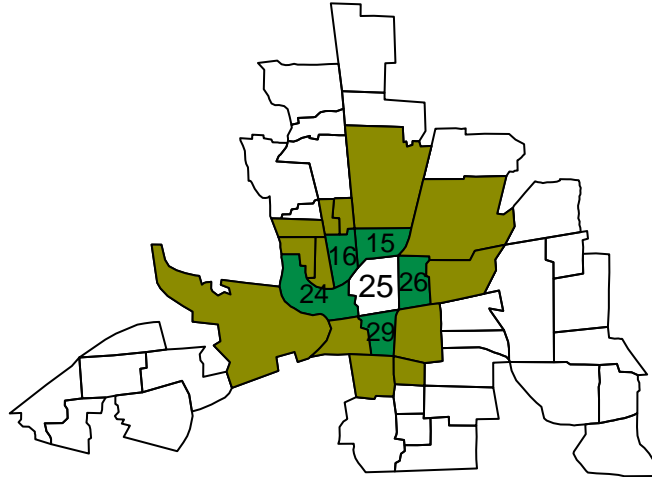
*#> 5 least connected regions:*  
*#> 1 6 42 46 47 with 2 links*  
*#> 1 most connected region:*  
*#> 20 with 10 links*  
*#>*  
*#> Weights style: W*  
*#> Weights constants summary:*



```
#>      n   nn S0      S1      S2
#> W 49 2401 49 22.75119 203.7091
```

Following EDA, next chunk of code, will display the first (with labels) and second order neighbours for the polygon with ID equal to 25, using the rook-style.

```
##Rook-Style
columbus <- st_read(system.file("extdata/columbus.shp", package="spdepbi")[1], quiet=TRUE)
plot(st_geometry(columbus))
coords = st_geometry(st_centroid(columbus$geometry))
p1 = st_point(c(8.940741,12.35734))
(p1 = st_sfc(p1))
#> Geometry set for 1 feature
#> geometry type: POINT
#> dimension: XY
#> bbox: xmin: 8.940741 ymin: 12.35734 xmax: 8.940741 ymax: 12.35734
#> epsg (SRID): NA
#> proj4string: NA
#> POINT (8.940741 12.35734)
(polcontain = st_within(p1,columbus))
#> Sparse geometry binary predicate list of length 1, where the predicate was `within'
#> 1: 25
(myPol = which(as.matrix(polcontain)))
#> [1] 25
result = columbus[myPol, ]
#Get centroid of selected polygon and plot its ID
centroidsp <- st_geometry(st_centroid(result$geometry))
text(st_coordinates(centroidsp)[,1],st_coordinates(centroidsp)[,2],columbus[myPol,]$POLYID)
col_nbr <- poly2nb(columbus,queen=F) # rook-style
col.lags10 <- nblag(col_nbr, 10) # Order 10
#Plot first and second order polygons
plot(st_geometry(columbus[col.lags10[[1]][[myPol]],]), col = "springgreen4", add = TRUE)
plot(st_geometry(columbus[col.lags10[[2]][[myPol]],]), col = "yellow4", add = TRUE)
#Get centroid of first order neighbours
centroids <- st_geometry(st_centroid(columbus[col.lags10[[1]][[myPol]],)$geometry))
#plot text of first order neighbours
text(st_coordinates(centroids)[,1],st_coordinates(centroids)[,2],columbus[col.lags10[[1]][[myPol]],)$POLYID)
```



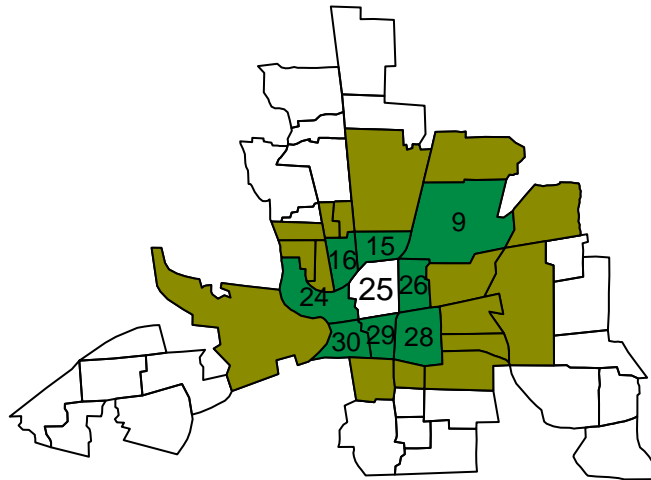
And using queen-style:

```
##Queen-Style
plot(st_geometry(columbus))
coords = st_geometry(st_centroid(columbus$geometry))
p1 = st_point(c(8.940741,12.35734))
(p1 = st_sfc(p1))
#> Geometry set for 1 feature
#> geometry type: POINT
#> dimension: XY
#> bbox: xmin: 8.940741 ymin: 12.35734 xmax: 8.940741 ymax: 12.35734
#> epsg (SRID): NA
#> proj4string: NA
#> POINT (8.940741 12.35734)
(polcontain = st_within(p1,columbus))
#> Sparse geometry binary predicate list of length 1, where the predicate was `within'
#> 1: 25
(myPol = which(as.matrix(polcontain)))
#> [1] 25
result = columbus[myPol, ]
#Get centroid of selected polygon and plot its ID
centroidsp <- st_geometry(st_centroid(result$geometry))
text(st_coordinates(centroidsp)[,1],st_coordinates(centroidsp)[,2],columbus[myPol,]$POLYID)
col_nbq1 <- poly2nb(columbus) #Queen
col.lags10 <- nblag(col_nbq1, 10) #Order 10
#Plot first and second order polygons
```

```

plot(st_geometry(columbus[col.lags10[[1]][[myPol]],), col = "springgreen4", add = TRUE)
plot(st_geometry(columbus[col.lags10[[2]][[myPol]],), col = "yellow4", add = TRUE)
#Get centroid of first order neighbours
centroids <- st_geometry(st_centroid(columbus[col.lags10[[1]][[myPol]],)$geometry))
#plot text of first order neighbours
text(st_coordinates(centroids)[,1],st_coordinates(centroids)[,2],columbus[col.lags10[[1]][[myPol]],)$POLYID)

```



### 3. Data Analysis

With the weights inside the neighbour list (a.lw), bivariate spatial autocorrelation using variables CRIME, INCOME and HOVAL, will be analyzed.

```

CRIME <- columbus$CRIME
INCOME <- columbus$INC
HOVAL <- columbus$HOVAL

```

#### 3.1. Bivariate Moran's I (moran.bi)

Analysis of the global bivariate spatial autocorrelation of Columbus dataset taking into consideration variables CRIME and INCOME.

```
moran.bi(CRIME,INCOME,a.lw,zero.policy =T)
#> [1] -0.4286322
```

The expected value for no autocorrelation is -0.02, from  $-1/(N-1)$ , where N equals 49 (number of polygons). The Bivariate Moran's I equal to -0.4286 seems significant below to -0.02, which may indicate that the global bivariate spatial autocorrelation is dispersed (spatial outliers) in the dataset.

### 3.2 Formal Test for Bivariate Moran's I (moranbi.test)

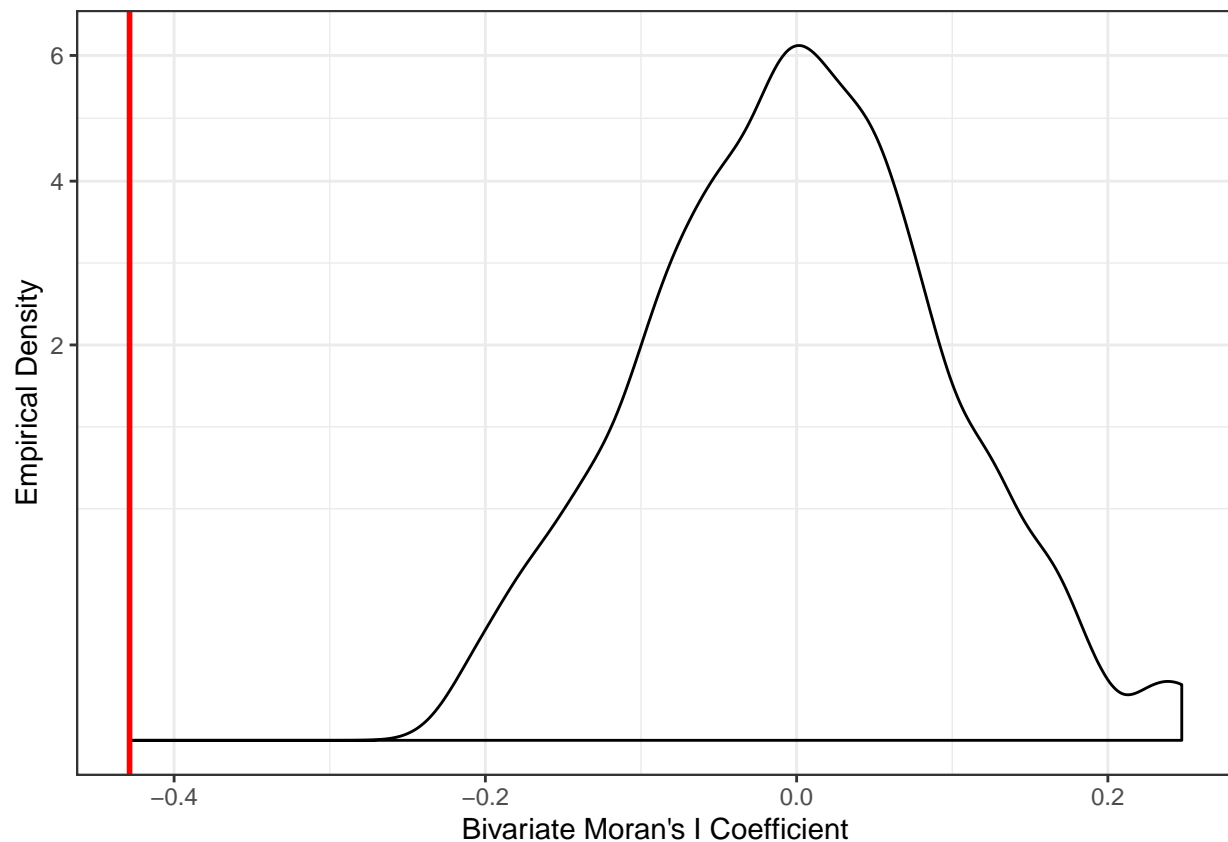
Analysis of the significance of bivariate Moran's I (observed value = -0.4286) to declare global dispersed behavior for CRIME and INCOME in the dataset, but first, in the following graphics is described the hypothesis test method using the p-value or z-score statistics when dealing with Moran's I.

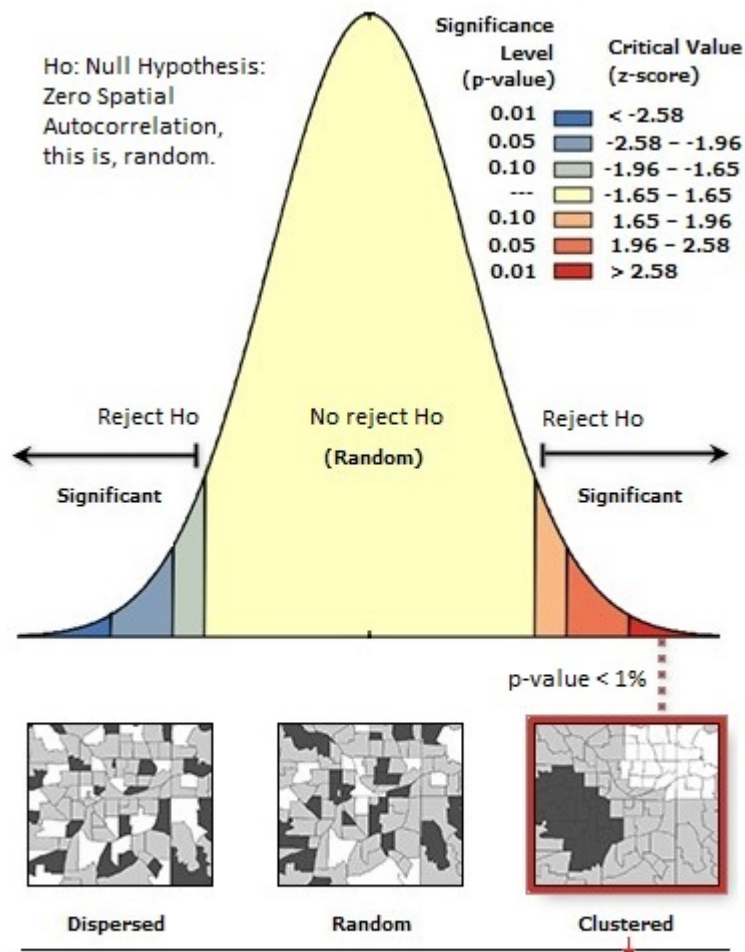
```
knitr::include_graphics("test1ho.jpg")
```

See the interpretation at the bottom of previous graphic.

Observed value for CRIME and INCOME, will be compared with the expected value empirical density constructed with MonteCarlo simulations.

```
set.seed(123)
MBCrime <- moranbi.test(CRIME,INCOME,a.lw,999,graph=T,zero.policy =T,N=1000)
```





Given p-value less than 1%, there is less than 1% likelihood that this clustered pattern could be the result of random chance.

Cluster: Moran's I significant above of  $-1/(N-1)$ , where N is the sample size. In summary positive values of index.

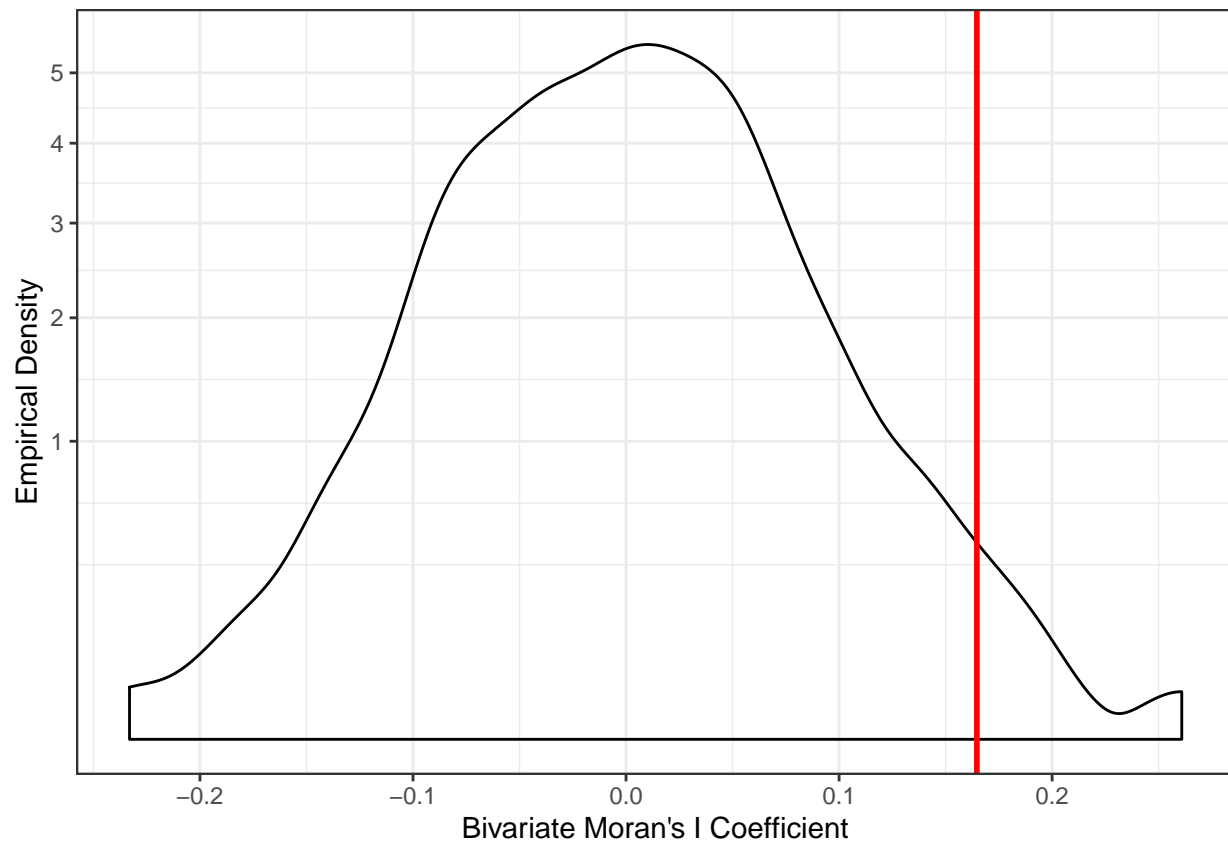
Figure 1: Hypothesis Testing - 1

```
#> $Observed
#> [1] -0.4286322
#>
#> $Expected
#> [1] -0.02083333
#>
#> $p.value
#> [1] 0.000999001
```

According to previous graphic and the resulting p-value (0.000999001), there is less than 0.1% likelihood that this dispersed spatial pattern between CRIME and INCOME could be the result of random chance.

Next the analysis between INCOME and HOVAL.

```
moranbi.test(INCOME,HOVAL,a.lw,999,graph=T,zero.policy =T,N=1000)
```



```
#> $Observed
#> [1] 0.164677
#>
#> $Expected
#> [1] -0.02083333
#>
#> $p.value
#> [1] 0.01098901
```

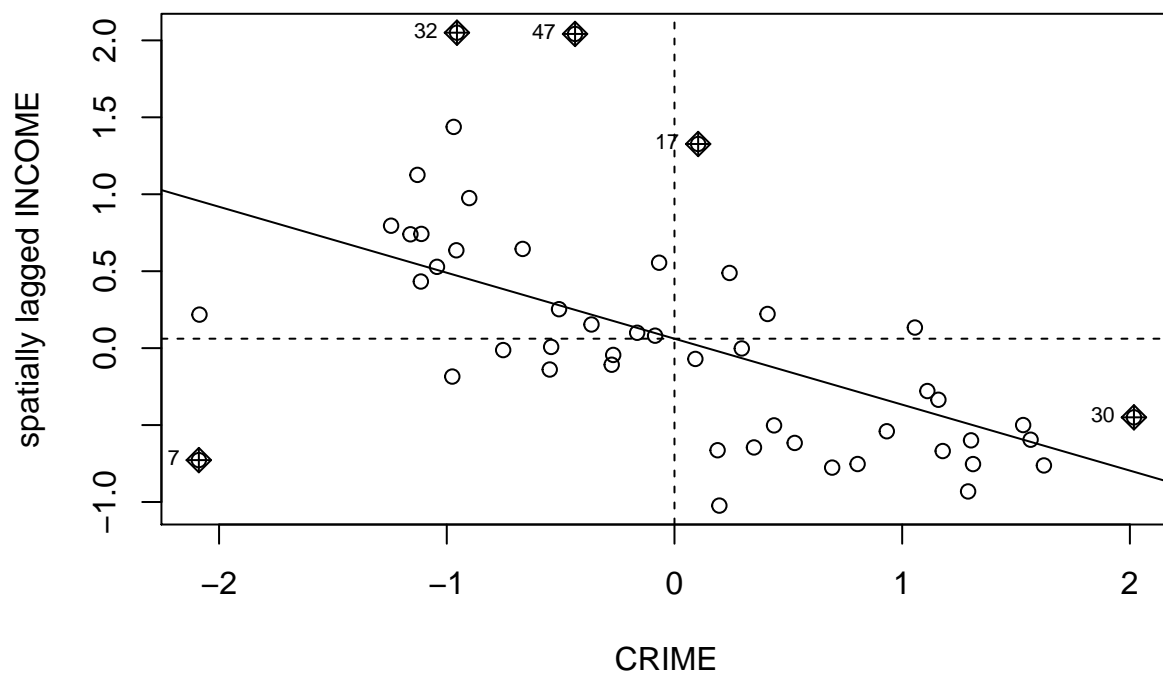


The observed bivariate Moran's I between INCOME and HOVAL of 0.1646, suggest clustered spatial autocorrelation. According to previous graphic and the resulting p-value (0.01098), there is less than 5% likelihood that this clustered pattern could be the result of random chance.

### 3.3. Plot of Bivariate Spatial Autocorrelation (morambi.plot)

Bivariate spatial autocorrelation can be seen graphically (morambi.plot) using spatial lags plots of scaled variables. Method used to create spatial lag values in morambi.plot is 'lag.listw', and is explained in pages 257 and 258 of Edzer J. Pebesma (2008). With scale function of R base, it is possible to transform CRIME and INCOME, to have the same relative scale of values for comparative analysis.

```
CRIME <- as.vector(scale(columbus$CRIME))
INCOME <- as.vector(scale(columbus$INC))
morambi.plot(CRIME, INCOME, quiet = F, zero.policy = F, listw = a.lw)
```



```
#> Potentially influential observations of
#> lm(formula = wy ~ x) :
#>
#>   dfb.1_ dfb.x   dffit   cov.r   cook.d hat
#> 7  -0.48  1.02_* -1.13_*  0.79_*  0.53  0.11
#> 17  0.33  0.03   0.33   0.86_*  0.05  0.02
#> 30  0.09  0.19   0.21   1.15_*  0.02  0.11
#> 32  0.41 -0.40   0.57   0.79_*  0.14  0.04
#> 47  0.47 -0.21   0.51   0.71_*  0.11  0.02
```

Previous graphic shows the negative correlation trend between scaled-lag variables INCOME and CRIME, pointing five potentially influential observations (IDs: 32, 47, 17, 30, and 7), this is, atypical observations that not follow the displayed pattern.

### 3.4. Local Bivariate Spatial Autocorrelation with Moran's I (localmoran.bi)

As proposed by Anselin (1995) and Ord and Getis (1995) with univariate Moran's I global indicator to be scaled in the context of local indicators (LISA), it is possible to calculate for each areal unit the local bivariate spatial autocorrelation index and its corresponding transformation to z-scores, which could be used to evaluate the significance of each local index.

```
CRIME <- columbus$CRIME
INCOME <- columbus$INC
HOVAL <- columbus$HOVAL
localmoran.bi(CRIME, INCOME, a.lw, zero.policy=FALSE, alternative="two.sided")
#>           Ii           E.Ii          Var.Ii          Z.Ii      Pr(z != 0)
#> 1  -0.8756965797 -0.02083333  0.47692245 -1.237863510  2.157667e-01
#> 2   0.1840173121 -0.02083333  0.31122146  0.367199913  7.134699e-01
#> 3   0.0121332471 -0.02083333  0.22837096  0.068984808  9.450017e-01
#> 4  -0.0166932676 -0.02083333  0.22837096  0.008663369  9.930877e-01
#> 5  -0.5142687410 -0.02083333  0.10409522 -1.529378303  1.261707e-01
#> 6  -0.0042713111 -0.02083333  0.47692245  0.023982225  9.808668e-01
#> 7   1.5518390559 -0.02083333  0.22837096  3.290923763  9.985897e-04
#> 8  -0.2058466511 -0.02083333  0.14552046 -0.484999125  6.276770e-01
#> 9   0.0303629410 -0.02083333  0.10409522  0.158680285  8.739208e-01
#> 10 -0.0381999045 -0.02083333  0.22837096 -0.036340729  9.710107e-01
#> 11 -1.2626171423 -0.02083333  0.17866066 -2.937862878  3.304832e-03
#> 12 -1.2260263675 -0.02083333  0.14552046 -3.159326981  1.581340e-03
#> 13 -0.5489155315 -0.02083333  0.22837096 -1.105047858  2.691389e-01
#> 14 -1.0085539855 -0.02083333  0.14552046 -2.589238751  9.618838e-03
#> 15 -0.6183497586 -0.02083333  0.14552046 -1.566346395  1.172675e-01
#> 16 -0.8042165103 -0.02083333  0.10409522 -2.428056875  1.517996e-02
#> 17  0.1408247858 -0.02083333  0.31122146  0.289776228  7.719874e-01
#> 18 -0.3321002453 -0.02083333  0.22837096 -0.651347149  5.148224e-01
#> 19 -0.3969839825 -0.02083333  0.31122146 -0.674259460  5.001464e-01
#> 20 -0.4639575165 -0.02083333  0.07924007 -1.574175088  1.154470e-01
#> 21 -0.0006139339 -0.02083333  0.31122146  0.036243780  9.710880e-01
#> 22 -0.0070247302 -0.02083333  0.14552046  0.036198261  9.711243e-01
#> 23 -0.8972070986 -0.02083333  0.31122146 -1.570921925  1.162008e-01
#> 24 -0.1283050034 -0.02083333  0.12184889 -0.307881222  7.581727e-01
#> 25 -0.9501364611 -0.02083333  0.10409522 -2.880328446  3.972611e-03
#> 26 -0.2299708617 -0.02083333  0.14552046 -0.548239010  5.835278e-01
#> 27  0.1442456207 -0.02083333  0.22837096  0.345438921  7.297644e-01
#> 28 -0.7967076323 -0.02083333  0.09028680 -2.582136734  9.819067e-03
#> 29 -0.7813883704 -0.02083333  0.12184889 -2.178812465  2.934560e-02
#> 30 -0.9263887953 -0.02083333  0.17866066 -2.142400116  3.216130e-02
#> 31 -0.5624913418 -0.02083333  0.31122146 -0.970935547  3.315804e-01
#> 32 -1.9989423799 -0.02083333  0.22837096 -4.139327498  3.483254e-05
#> 33  0.0925912054 -0.02083333  0.22837096  0.237348549  8.123864e-01
#> 34 -0.4387381479 -0.02083333  0.22837096 -0.874494201  3.818492e-01
#> 35  0.1205782629 -0.02083333  0.12184889  0.405111180  6.853958e-01
#> 36 -1.0107277996 -0.02083333  0.17866066 -2.341932779  1.918417e-02
#> 37 -0.2241558592 -0.02083333  0.14552046 -0.532995399  5.940368e-01
```

```
#> 38 -0.3160774223 -0.02083333 0.14552046 -0.773961175 4.389538e-01
#> 39 -0.6217071234 -0.02083333 0.31122146 -1.077081319 2.814439e-01
#> 40 -1.2974627238 -0.02083333 0.17866066 -3.020301979 2.525228e-03
#> 41 -1.4231112290 -0.02083333 0.31122146 -2.513618251 1.194997e-02
#> 42 -0.4921424256 -0.02083333 0.47692245 -0.682467435 4.949434e-01
#> 43 -0.0065744127 -0.02083333 0.14552046 0.037378736 9.701830e-01
#> 44 0.0777532421 -0.02083333 0.17866066 0.233240149 8.155749e-01
#> 45 -0.0572016358 -0.02083333 0.22837096 -0.076103142 9.393370e-01
#> 46 -0.8421703960 -0.02083333 0.47692245 -1.189316752 2.343150e-01
#> 47 -0.9101054018 -0.02083333 0.47692245 -1.287688351 1.978545e-01
#> 48 -0.1309589624 -0.02083333 0.22837096 -0.230445357 8.177457e-01
#> 49 0.0096865488 -0.02083333 0.31122146 0.054707653 9.563714e-01
#> attr("call")
#> localmoran.bi(x = CRIME, y = INCOME, listw = a.lw, zero.policy = FALSE,
#> alternative = "two.sided")
#> attr("class")
#> [1] "localmoran.bi" "matrix"
```

Interpretation of local index is similar to 3.1. moran.bi. Interpretation of its significance is similar to 3.2. moranbi.test, but usign z-score instead of p-value. Legend in next graphic shows typical significant values of p-value (alpha) and its equivalent z-score values.

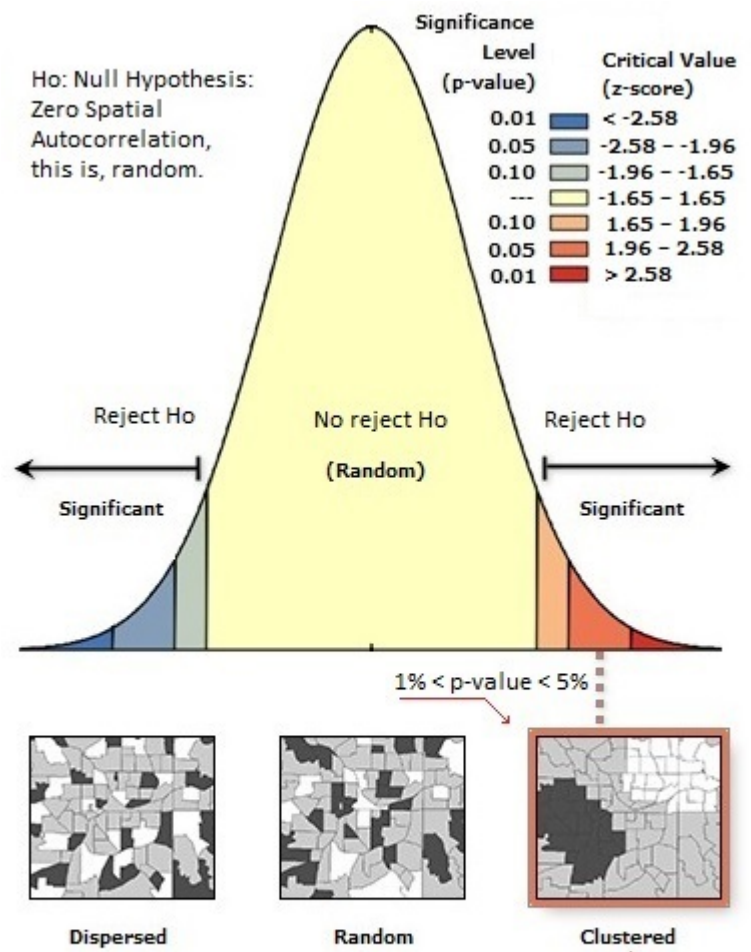
```
knitr::include_graphics("test5ho.jpg")
```

### 3.5. LISA Spatial Autocorrelation and Significance Map (moran.cluster)

Local spatial autocorrelation behavior can be seen in a map for the univariate approach. With a recursive use of the function 'localmoran' of 'spdep' package, it is possible to calculate local Moran's I for CRIME variable for each spatial unit, and then compare resulting p-values to establish significance of the index, represented as a choropleth map. Method and concept to evaluate significance is similar to 3.2. moranbi.test.

Next 'moran.cluster' function will create two maps: Lisa Cluster Map and Lisa Significance Map. Cluster Map can represent similar or dissimilar value concentrations, namely spatial clusters or spatial outliers (dispersed) respectively, and this behavior can be detected analyzing the resulting global univariate Moran's I, generally positive values means cluster, and negative values means dispersed (expected value is different for each case and depends of number of observations:  $-1/(N-1)$ )

```
moran.cluster(CRIME, a.lw, zero.policy = FALSE, columbus, significant=T)
```



Given p-value between 1% and 5%, there is less than 5% likelihood that this clustered pattern could be the result of random chance.

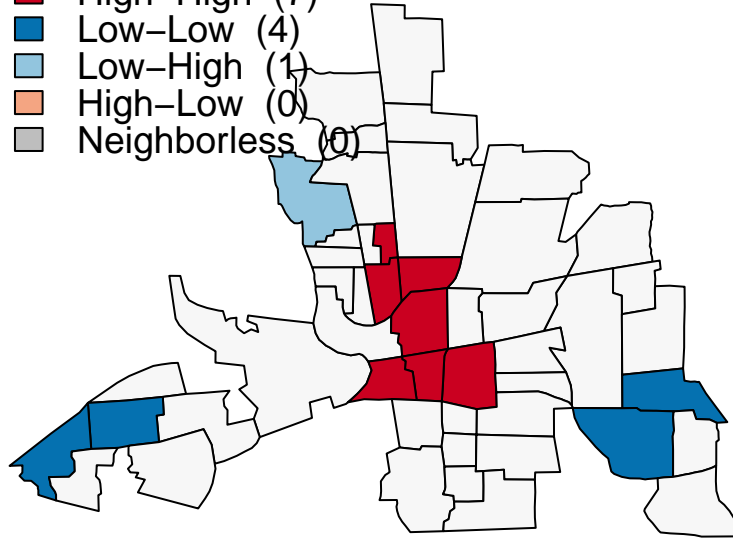
Cluster: Moran's I significant above of  $-1/(N-1)$ , where N is the sample size. In summary positive values of index.

Figure 2: Hypothesis Testing - 2

## CRIME

### LISA Cluster Map

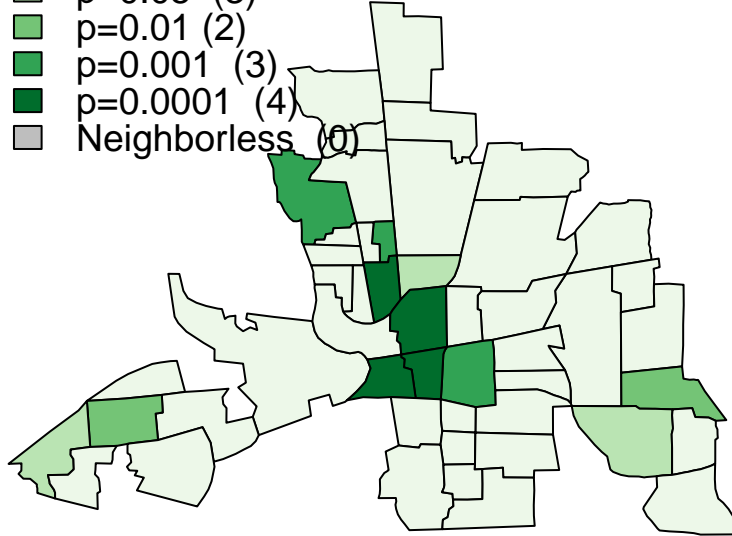
- Not Significant (37)
- High-High (7)
- Low-Low (4)
- Low-High (1)
- High-Low (0)
- Neighborless (0)



## CRIME

### LISA Significance Map

- Not Significant (37)
- $p=0.05$  (3)
- $p=0.01$  (2)
- $p=0.001$  (3)
- $p=0.0001$  (4)
- Neighborless (0)



### 3.6. Getis Gi\* Spatial Autocorrelation and Significance Map (getis.cluster)

Similar to 3.5. moran.cluster, but using local Gi\* (function 'localG' of 'spdep').

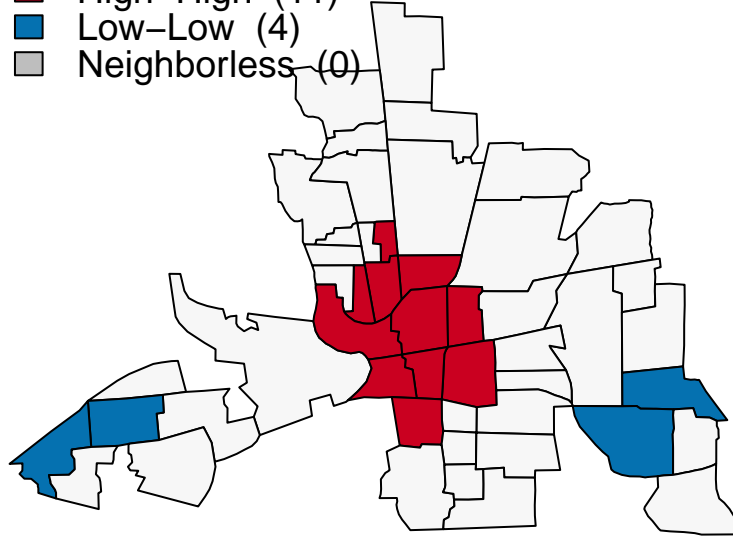
```
getis.cluster(CRIME, a.lw, zero.policy = FALSE, columbus, significant=T)
```



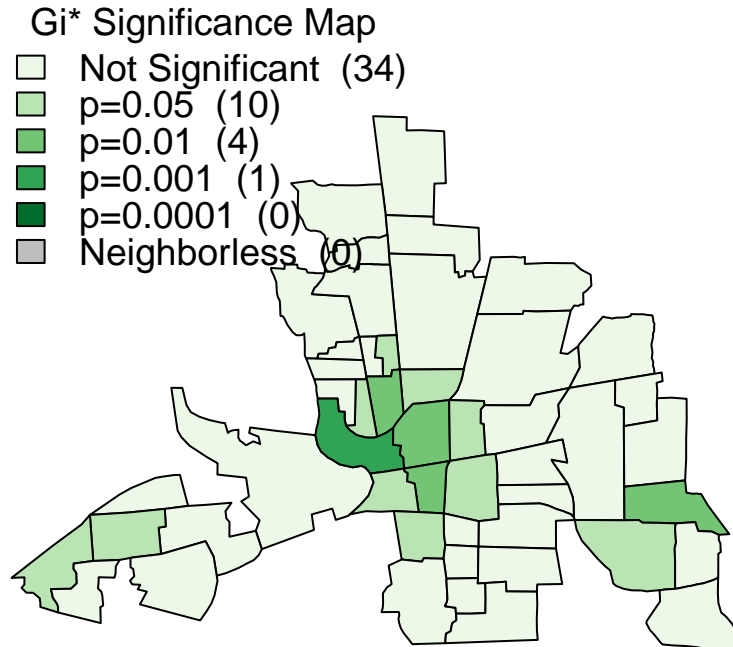
## CRIME

Gi\* Cluster Map

- Not Significant (34)
- High-High (11)
- Low-Low (4)
- Neighborless (0)



## CRIME



## 4. Discussion

With ‘spdepbi’ is possible to calculate, test and show graphically global and local bivariate Moran’s I for lattice data, and create clusters and significance maps for local univariate Getis\* and Moran’s I.

It is possible to combine results of function ‘localmorani.bi’ (see 3.4. localmorani.bi) with function ‘morani.cluster’ (see 3.5. morani.cluster) to evaluate if the if resulting local index correspond to a cluster, random, or dispersed behavior for each spatial unit. Same conclusion applies using function ‘localG’ of ‘spdep’ and 3.6. getis.cluster.

Conclusions about data analysis of Columbus dataset was done in each subtitle of 3. Data Analysis.

## References

- Anselin, Luc. 1988. “Spatial Econometrics. Boston, Kluwer Academic, Table 12.1, P. 189.” <https://geodacenter.github.io/data-and-lab/columbus/>.
- . 1995. “Local Indicators of Spatial Association—Lisa.” *Geographical Analysis* 27 (2): 93–115. <https://doi.org/10.1111/j.1538-4632.1995.tb00338.x>.
- Anselin, Luc, Ibnu Syabri, and Youngihn Kho. 2006. “GeoDa: An Introduction to Spatial Data Analysis.” *Geographical Analysis* 38 (January): 5–22. <https://doi.org/10.1111/j.0016-7363.2005.00671.x>.
- Bivan, Roger. 2019. “Spdep: Model Fitting Functions Moving to New Package Spatialreg.” *R*. <http://r-sig-geo.2731867.n2.nabble.com/spdep-model-fitting-functions-moving-to-new-package-spatialreg-td7592700.html>.

- Bivand, Roger, and David W. S. Wong. 2018. “Comparing Implementations of Global and Local Indicators of Spatial Association.” *TEST* 27 (3): 716–48. <https://doi.org/10.1007/s11749-018-0599-x>.
- Dray, Stéphane, David Bauman, Guillaume Blanchet, Daniel Borcard, Sylvie Clappe, Guillaume Guenard, Thibaut Jombart, et al. 2019. *Adespatial: Multivariate Multiscale Spatial Analysis*. <https://CRAN.R-project.org/package=adespatial>.
- Edzer J. Pebesma, Roger S. Bivand und Virgilio Gómez-Rubio. 2008. “Areal Data and Spatial Autocorrelation.” In *Applied Spatial Data Analysis with R*, Online-Ausg., 237–72. Springer New York. [/brokenurl#http://dx.doi.org/10.1007/978-0-387-78171-6](http://dx.doi.org/10.1007/978-0-387-78171-6).
- Melo, C.E. 2016. *Bivariate Spatial Autocorrelation with Moran’s Indexes*. NA. [https://www.researchgate.net/profile/Carlos\\_Melo\\_Martinez](https://www.researchgate.net/profile/Carlos_Melo_Martinez).
- Moran, P. A. P. 1950. “Notes on Continuous Stochastic Phenomena.” *Biometrika* 37 (1/2): 17–23. <http://www.jstor.org/stable/2332142>.
- Ord, J. K., and Arthur Getis. 1995. “Local Spatial Autocorrelation Statistics: Distributional Issues and an Application.” *Geographical Analysis* 27 (4): 286–306. <https://doi.org/10.1111/j.1538-4632.1995.tb00912.x>.
- Pebesma, Edzer. 2018. “Simple Features for R: Standardized Support for Spatial Vector Data.” *The R Journal* 10 (1): 439–46. <https://doi.org/10.32614/RJ-2018-009>.
- Tiefelsdorf, Michael, and Gerald M. Friedman. 2000. *Modelling Spatial Processes the Identification and Analysis of Spatial Relationships in Regression Residuals by Means of Morans I*. Springer.