Areal Data Group Project

Kayla, Ingmar, Hanmo, Will

2021-12-02

Contents

Set up

Resources we are referencing:

- http://www.css.cornell.edu/faculty/dgr2/_static/files/ov/ov_ADSA_Handout.pdf
- Banerjee, Carlin and Gelfand, Hierarchical Modeling and Analysis for Spatial Data, 1st Edition, Ch 3
- Bivand et al 2013 Ch 9
- rspatial.org

The packages we used are:

```r
library(tidyverse)
library(sp)
library(terra)
library(spdep)
library(sf)
library(GGally)
library(ape)
library(lmtest)
```

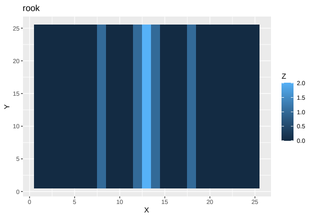
Introduction to Areal Data

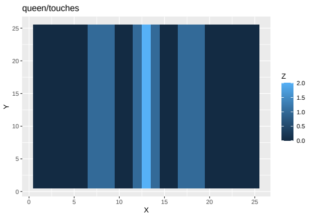
Areal data is spatial data which is observed or reported for spatial units which are polygons with defined boarders. Some examples are country level data, state level data, county level data, or zip code level data where the data is aggreated for the defined area. This is often the case for private information (i.e. mean income in a zip code instead of income of each individual with their address), or data the is collected at that scale (i.e. proportion of democratic voters in an electoral unit; Bivand et al., 2013).

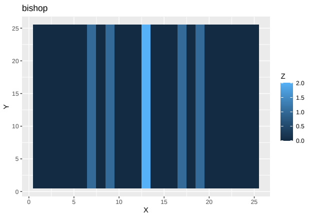
Some of the common problems we have with areal data have to do with the selected boundary. For example, gerrymandering is a common issue with votes because changing the boundaries yields different election results. Another issue is the choice of boundaries is often not made for the specific question being asked in the model. Bivand et al., 2013 gives the example of zipcodes which are a common unit for demograpsic data to be reported in, but were designed for postal delivery not demograpsic data collection. The boundaries not fitting the underlying patterns in the data can cause model misspecification, spatial autocorrelation, and make it difficult to assertain the number of independant observations in a datasete. Therefore, it is important to check for spatial pattterning due to this partitioning to see if the model indicated is appropriate (Bivand et al., 2013).

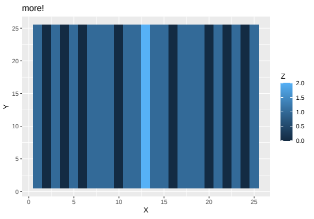
Neighbors and weights

Because we do not have a continuous surface on which to calculate distance between points to build a variogram we must consider neighbors instead. There are many ways to describe neighbors and build a **weight matrix**. Here are a few *(I didn't run this yet so might not do what I think it will)*:









So by looking at our dataset and understanding what the expected relationships are between neighbors we can build a weight matrix. After we do our analysis we also need to check the residuals to ensure we removed the spatial patterns.

more background on models??

Our dataset

The dataset we are using is published in `spData` and comes from: *Anselin, Luc. 1988. Spatial econometrics: methods and models. Dordrecht: Kluwer Academic, Table 12.1 p. 189.*

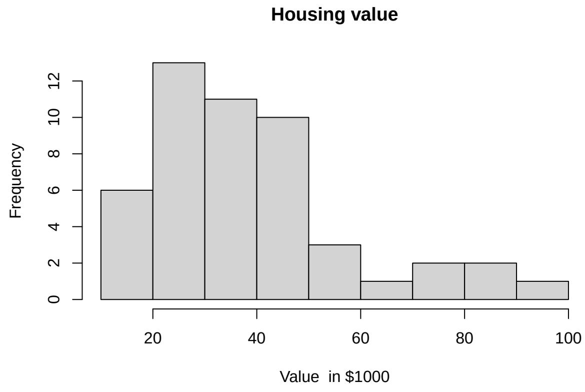
```
## load dataset
columbus <- vect(system.file("shapes/columbus.shp", package="spData")[1])
df.columbus <- as.data.frame(columbus)
glimpse(df.columbus)
```

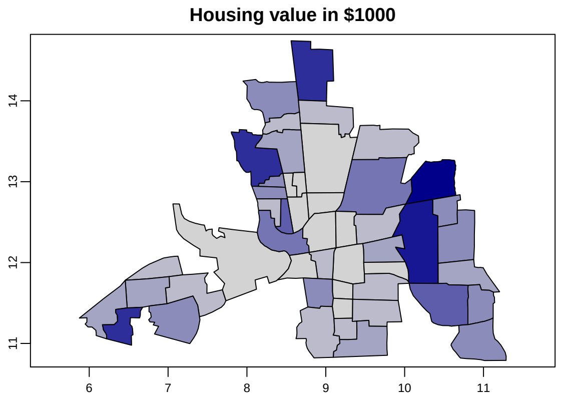
```
## Rows: 49
## Columns: 20
## $ AREA       <dbl> 0.309441, 0.259329, 0.192468, 0.083841, 0.488888, 0.2830...
## $ PERIMETER  <dbl> 2.440629, 2.236939, 2.187547, 1.427635, 2.997133, 2.3356...
## $ COLUMBUS_  <dbl> 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ...
## $ COLUMBUS_I <dbl> 5, 1, 6, 2, 7, 8, 4, 3, 18, 10, 38, 37, 39, 40, 9, 36, 1...
## $ POLYID     <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 1...
## $ NEIG       <dbl> 5, 1, 6, 2, 7, 8, 4, 3, 18, 10, 38, 37, 39, 40, 9, 36, 1...
## $ HOVAL      <dbl> 80.467, 44.567, 26.350, 33.200, 23.225, 28.750, 75.000, ...
## $ INC        <dbl> 19.531, 21.232, 15.956, 4.477, 11.252, 16.029, 8.438, 11...
## $ CRIME      <dbl> 15.725980, 18.801754, 30.626781, 32.387760, 50.731510, 2...
## $ OPEN       <dbl> 2.850747, 5.296720, 4.534649, 0.394427, 0.405664, 0.5630...
## $ PLUMB      <dbl> 0.217155, 0.320581, 0.374404, 1.186944, 0.624596, 0.2541...
## $ DISCBD     <dbl> 5.03, 4.27, 3.89, 3.70, 2.83, 3.78, 2.74, 2.89, 3.17, 4....
## $ X          <dbl> 38.80, 35.62, 39.82, 36.50, 40.01, 43.75, 33.36, 36.71, ...
## $ Y          <dbl> 44.07, 42.38, 41.18, 40.52, 38.00, 39.28, 38.41, 38.71, ...
## $ NSA        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ NSB        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ EW         <dbl> 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0,...
## $ CP         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1,...
## $ THOUS      <dbl> 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 10...
## $ NEIGNO     <dbl> 1005, 1001, 1006, 1002, 1007, 1008, 1004, 1003, 1018, 10...
```

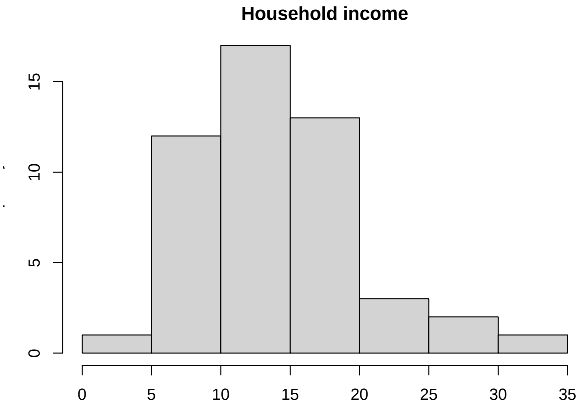
The county data is:

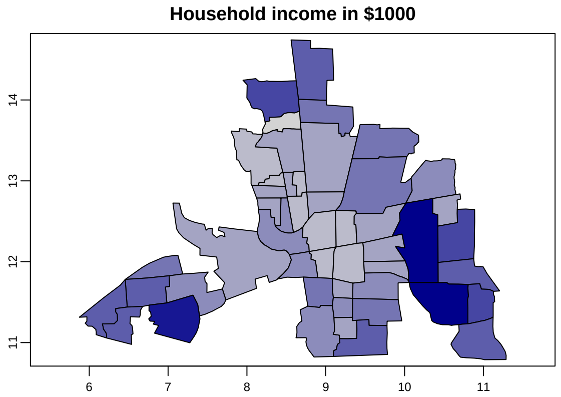
- HOVAL housing value (in $1,000)
- INC household income (in $1,000)
- CRIME residential burglaries and vehicle thefts per thousand households in the neighborhood
- OPEN open space in neighborhood
- PLUMB percentage housing units without plumbing
- CP core-periphery (an urban rural proxy)

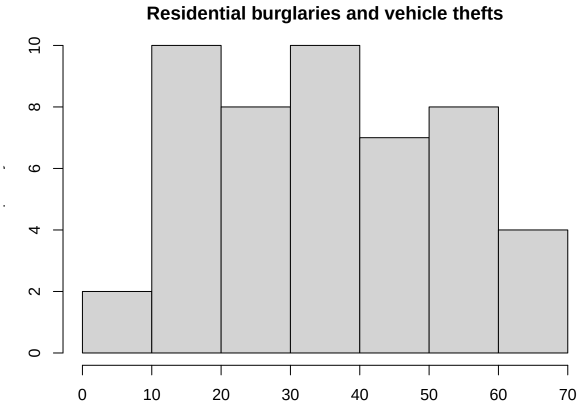
Look at some summaries of those metrics: *Kayla*

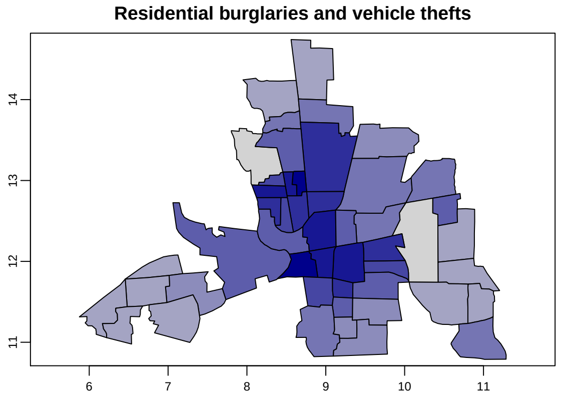

Housing value
Frequency
0
2
4
6
8
10
12
20
40
60
80
100
Value in $1000

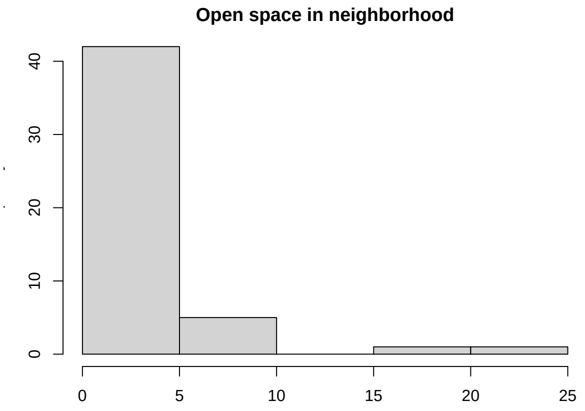

Housing value in $1000
11
12
13
14
6
7
8
9
10
11

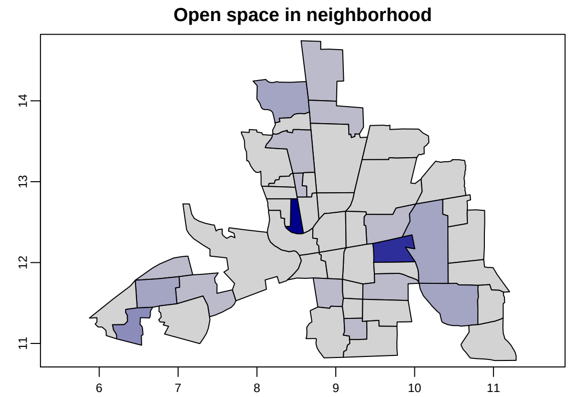

Household income
0
5
10
15
0
5
10
15
20
25
30
35

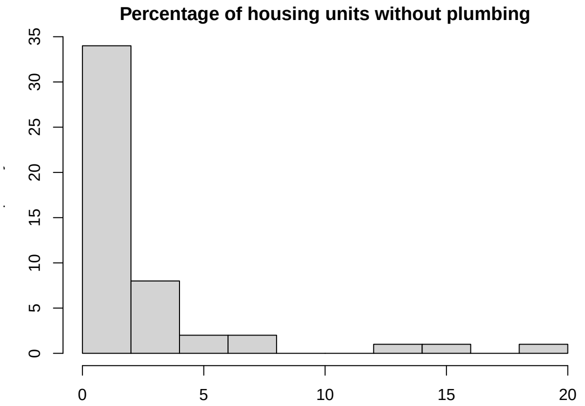

Household income in $1000
11
12
13
14
6
7
8
9
10
11

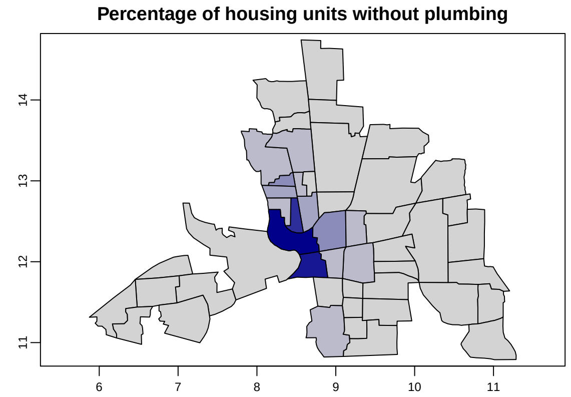

Residential burglaries and vehicle thefts
0
2
4
6
8
10
0
10
20
30
40
50
60
70

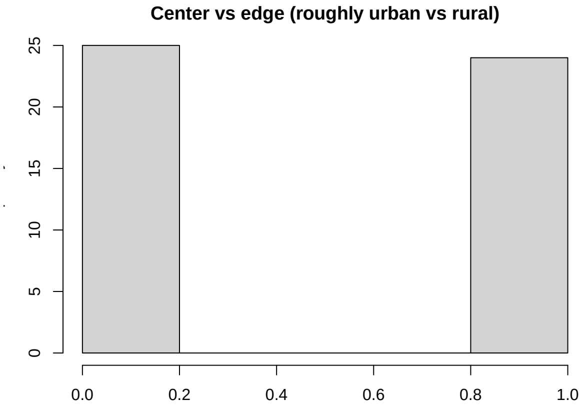

Residential burglaries and vehicle thefts
11
12
13
14
6
7
8
9
10
11

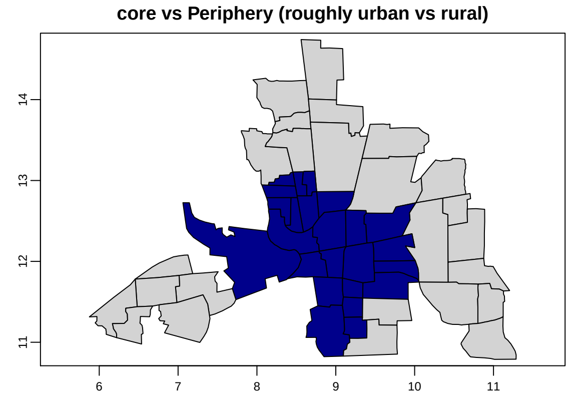






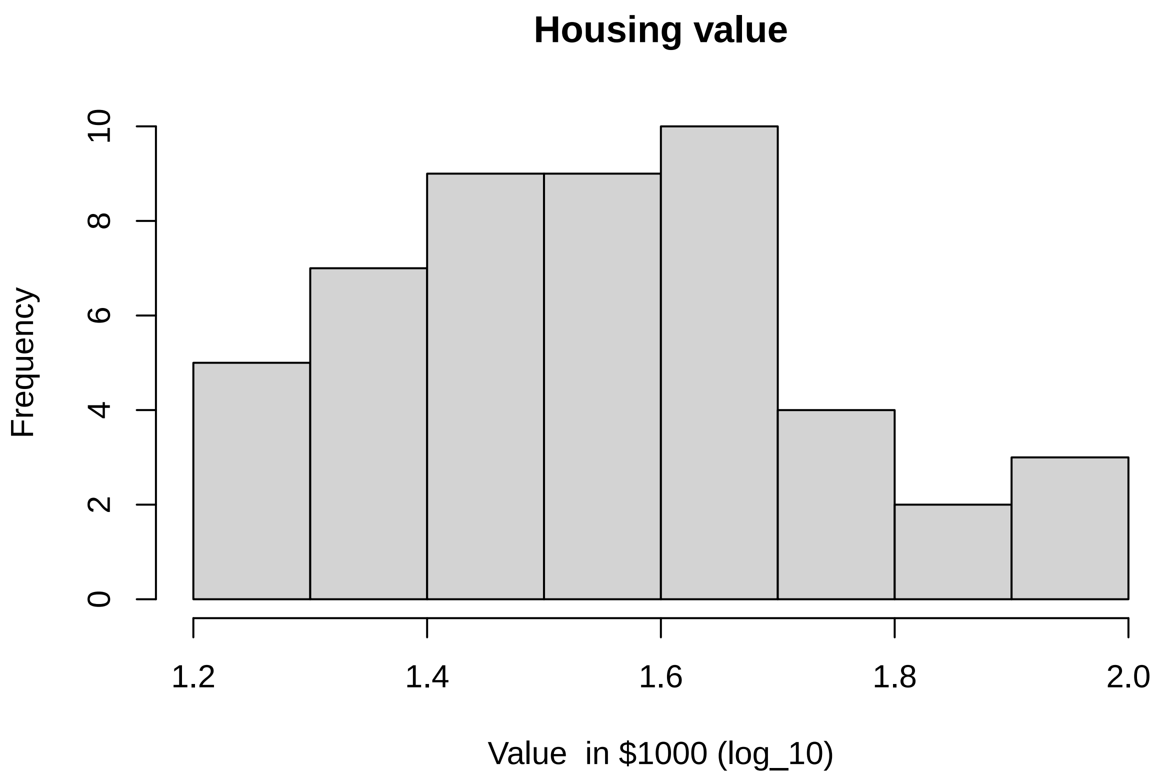






Because we plan to use housing value as our response variable, a logarithm transformation is applied to fix the skewness of the distribution for the variable housing value.

```
hist(log10(df.columbus$HOVAL),
     main = "Housing value",
     xlab = "Value  in $1000 (log_10)")
```



Exploratory Data Analysis

Measures of spatial association

Neighbors

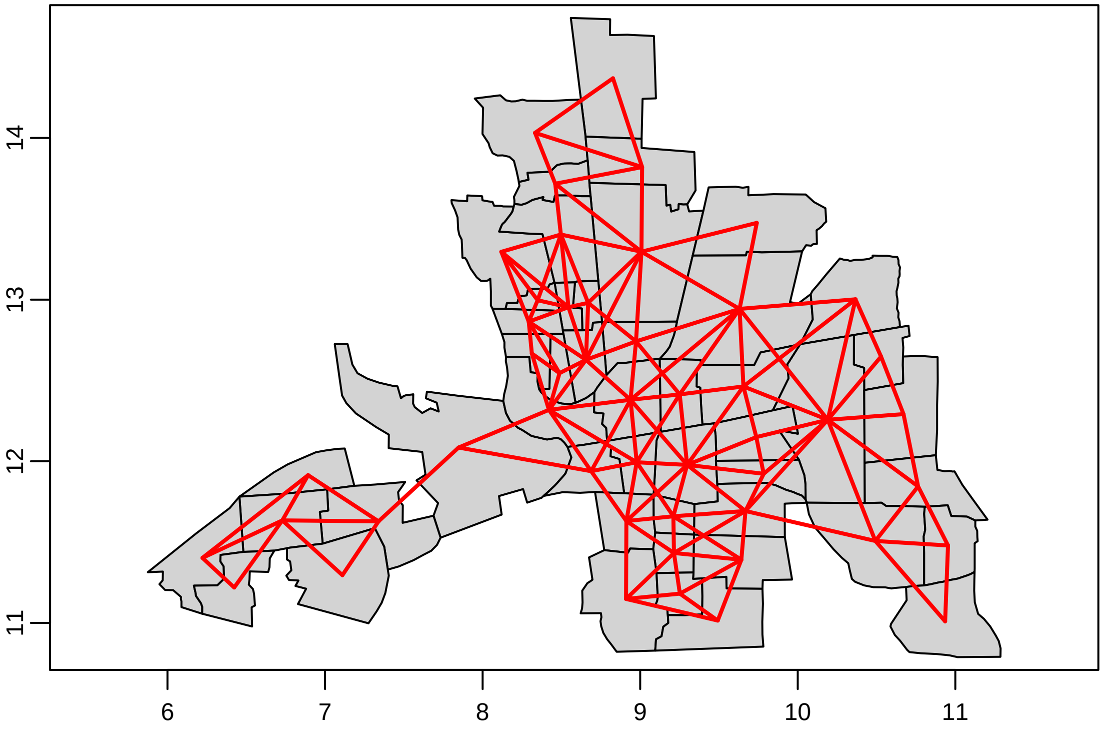
Because measures of spatial association are dependent on which neighborhoods are next to each other (or close to each other) we need to create an adjacency matrix to use in the models and test for spatial autocorrelation.

Figure out what the neighbors are:

```
xy <- centroids(columbus)
head(neighbors <- adjacent(columbus, "touches", symmetrical=TRUE))
```

```
##      from to
## [1,]    1  2
## [2,]    1  3
## [3,]    2  3
## [4,]    2  4
## [5,]    3  4
## [6,]    3  5
```

```
plot(columbus, col='lightgray', border='black', lwd=1)
p1 <- xy[neighbors[,1], ]
p2 <- xy[neighbors[,2], ]
lines(p1, p2, col='red', lwd=2)
```



There are a few options when making the adjecency matrix, but for our purposes we are saying that any counties touching each other are neighbors (1s) and any that aren't are not (0s). This is the neighbor schema shown in the queen/touches plot above.

Using that neighbor list we can construct a weight matrix ($w_{ij} looks like where if *i* is a neighbor of *j* the corresponding value in the matrix is 1 and if *i* is not a neighbor of *j* the corresponding value in the matrix is 0.

```
head(adjacent(columbus, "touches", pairs = FALSE))
```

```
##   1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
## 1 0 1 1 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 2 1 0 1 1 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 3 1 1 0 1 1 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 4 0 1 1 0 1 0 0 1 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 5 0 0 1 1 0 1 0 1 1  0  1  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0
## 6 0 0 0 0 1 0 0 0 1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
## 1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 3  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 4  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 5  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 6  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```

Using this weight matrix we can then test for spatial autocorrelation.

Moran's I

The assumptions for this test are (Bivand et al., 2013):

- " the mean model of the data removes systematic spatial patterning from the data"

- the observed spatial autocorrelation is not due to an underlying process in our model

Some limitations for Moran's I are
*

Using `terra` to test for spatial autocorrelation of the variable, by county in Columbus, OH.

$$I = \frac{n}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(y_i - \overline{y})(y_j - \overline{y})}{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}}$$

Moran's I is a global measure of spatial autocorrelation with values ranging -1 to 1. Here we are using a neighbor's matrix for any counties that are touching each other (w_{ij}). n is the number of neighborhoods in Columbus, OH which are indexed by i and j (is polygon i a neighbor with j. $\overline{y}$ is the mean value of the variable of interest and $y_{1 \ldots n}$ is that value in each polygon.

The adjacency matrix for all counties that touch each other (w_{ij} above, calling ww here):

```r
ww <-  adjacent(columbus, "touches", pairs=FALSE)
```

Roughly the expected value for Moran's I is $E(I) = \frac{-1}{n-1}$

```r
-1/(nrow(columbus)-1)
```

```
## [1] -0.02083333
```

Values significantly ($\alpha = 0.05$) below that indicate negative spatial autocorrelation (a phenomena that generally occurs in random datasets) and above that indicates positive spatial autocorrelation (neighbors are more similar to each other than non-neighbors).

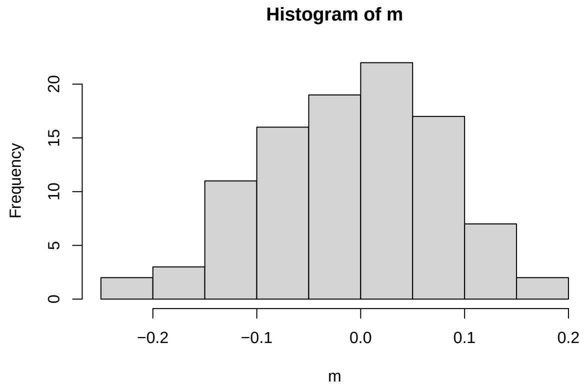
*** I'm seeing the formula written in 2 ways above, and with denominator's flipped

```r

## Moran's I
(ac <- autocor(columbus$HOVAL, ww, "moran"))
```

House value

```
## [1] 0.2213441
```

```r
## Monte Carlo sim to test for significance (I'm following https://rspatial.org/terra/analysis/3-spauto
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$HOVAL), ww, "moran")
})
hist(m) #distribution of values of I using subsets of dataset
```



```
## p-value
sum(m >= ac) / 100 # number of times I of subset is >= to I of entire dataset / number of trials
```

```
## [1] 0
```

So there is significant (Moran's I = 0.2213441, $p < 0.05$) spatial autocorrelation in house value, meaning the average value of houses in neighboring neighborhoods are different from the average value of all neighborhoods.

```

## Moran's I
(ac <- autocor(columbus$INC, ww, "moran"))
```

Household income

```
## [1] 0.412344
```

```
## sim
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$INC), ww, "moran")
})


## p-value
sum(m >= ac) / 100
```

```
## [1] 0
```

There is significant positive (Moran's I = 0.412344, $p < 0.05$) Spatial autocorrelation in household income.

```

## Moran's I
(ac <- autocor(columbus$CRIME, ww, "moran"))
```

Crime

```
## [1] 0.5154614
```

```
##sim
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$CRIME), ww, "moran")
})

## p-value
sum(m >= ac) / 100
```

```
## [1] 0
```

So again we see significant spatial autocorrelation (Moran's I = 0.5154614, $p < 0.05$).

```r
## Moran's I
(ac <- autocor(columbus$OPEN, ww, "moran"))
```

Open space

```
## [1] -0.03669849
```

```r
## sim
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$HOVAL), ww, "moran")
})

## p-value
sum(m >= ac) / 100
```

```
## [1] 0.55
```

There is not spatial autocorrelation with open space (Moran's I = -0.0366985; p-value > 0.05)

```r
## Moran's I
(ac <- autocor(columbus$PLUMB, ww, "moran"))
```

Plumbing

```
## [1] 0.4550575
```

```r
## sim
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$HOVAL), ww, "moran")
})

## p-value
sum(m >= ac) / 100
```

```
## [1] 0
```

There is significant spatial autocorrelation with plumbing (Moran's I = 0.4550575, $p < 0.05$).

Geary's C

N is the number of spatial units indexed by i and j; x is the variable of interest; $\bar{x}$ is the mean of x; w_{ij} is a matrix of spatial weights with zeroes on the diagonal (i.e., $w_{ii} = 0$); and W is the sum of all w_{ij}.

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

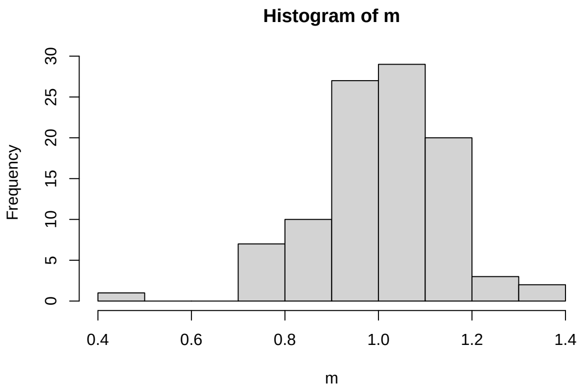
Geary's C is a measure of local spatial autocorrelation that is roughly inversely related to Moran's I.
The values ranging 0 to > 1, with values 0-1 representing positive spatial autocorrelation and values > 1 representing negative spatial autocorrelation. The neighbor's matrix (w_{ij}) is the same what was used for Moran's I. $\sum_{i \neq j} w_{ij}$ is the sum of that weight matrix with the diagonal equal to 0.

```
(gearyc <- autocor(columbus$HOVAL, ww, "geary"))
```

House Value

```
## [1] 0.7889937
```

```
## Monte Carlo sim to test for significance
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$HOVAL), ww, "geary")
})
hist(m)
```



```
## p-value
sum(m >= gearyc) / 100
```

```
## [1] 0.92
```

No significant spatial autocorrelation (geary's c = 0.7889937, p > 0.05).

```
(gearyc <- autocor(columbus$INC, ww, "geary"))
```

Household Income

```
## [1] 0.7137603
```

```
## Monte Carlo sim to test for significance
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$INC), ww, "geary")
})

## p-value
sum(m >= gearyc) / 100
```

```
## [1] 0.97
```

No significant spatial autocorrelation (geary's c = 0.7137603, p > 0.05).

```
(gearyc <- autocor(columbus$CRIME, ww, "geary"))
```

Crime

```
## [1] 0.5916113
```

```r
## Monte Carlo sim to test for significance
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$CRIME), ww, "geary")
})

## p-value
sum(m >= gearyc) / 100
```

```
## [1] 0.99
```

No significant spatial autocorrelation (geary's c = 0.5916113, p > 0.05).

```r
(gearyc <- autocor(columbus$OPEN, ww, "geary"))
```

Open space

```
## [1] 0.878182
```

```r
## Monte Carlo sim to test for significance
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$OPEN), ww, "geary")
})

## p-value
sum(m >= gearyc) / 100
```

```
## [1] 0.7
```

No significant spatial autocorrelation (geary's c = 0.878182, p > 0.05).

```r
(gearyc <- autocor(columbus$PLUMB, ww, "geary"))
```

Plumbing

```
## [1] 0.6806864
```

```r
## Monte Carlo sim to test for significance
m <- sapply(1:99, function(i) {
    autocor(sample(columbus$PLUMB), ww, "geary")
})

## p-value
sum(m >= gearyc) / 100
```

```
## [1] 0.95
```

No significant spatial autocorrelation (geary's c = 0.6806864, p > 0.05).

Compare Moran's I and Geary's C

Reinhard Furrer (Furrer and Applied Statistics Group 2021) suggests to take 1-C to compare it to Moran's I more easily.

	Morans_I	signif	Gearys_C	One_minus_Gearys_C	signif.1
House value	0.221	*	0.789	0.211	
Income	0.412	*	0.714	0.286	
Crime	0.515	*	0.592	0.408	
Open space	-0.037		0.878	0.122	
Plumbing	0.455	*	0.681	0.319	

There are differences observed in spatial autocorrelation in the data calculated with Moran's I and Geary's C. For both Moran's I and Geary's C there was not significant spatial autocorrelation in Open space. There was significant positive spatial autocorrelation in house value, household income, crime, and plumbing using Moran's I, but no significant spatial autocorrelation using Geary's C. However the trends are the same (both find weak to moderate positive spatial correlation)

Spatial regression models

Constant means

We decide to model the `HOVAL` (home value) variable for all of our models. The simplest, yet naive, model is the constant means model, which is essentially an intercept-only model, i.e. the average of the `HOVAL` variable.

$$Y_i = \mu + \varepsilon_i$$

Where Y_i is the value of a home in 1000s of dollars, μ is the mean home value and ϵ_i are the individual deviations from the mean, which we assume to be be i.i.d. distributed.

$$\hat{Y_i} = \bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$$

We estimate the model using `lm` function:

```r
# make spatial vector to simple feature
columbus.sf <- sf::st_as_sf(columbus)
zero.means <- lm(HOVAL ~ 1, data=columbus.sf)
summary(zero.means)
```

```
##
## Call:
## lm(formula = HOVAL ~ 1, data = columbus.sf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.536 -12.736  -4.936   4.864  57.964
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   38.436      2.638   14.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.47 on 48 degrees of freedom
```

We find the average home value is $38436. Since we will use the log-transformed home value for subsequent models, we repeat the zero means model for the transformed variable for purposes of comparison.

```r
# estimate log-transformed model
zero.means.log <- lm(log(HOVAL) ~ 1, data=columbus.sf)
summary(zero.means.log)
```

```
##
## Call:
## lm(formula = log(HOVAL) ~ 1, data = columbus.sf)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66751 -0.30582 -0.04076  0.21585  1.01620
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.55231    0.06178    57.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4324 on 48 degrees of freedom
```

We find that the model estimates `log(HOVAL)` to be 3.55231, which is \$ 3.4894×10^4 after exponentiating.

Linear Model with Independent Residuals

Improving over the zero means model, we model the home value `HOVAL` as a linear function of its (non-spatial) covariates with i.i.d. errors.

$$\mathbf{Y} = \mathbf{X}\beta + \varepsilon$$

where X is a matrix of the predictors `INC` (income), `CRIME`, `OPEN` (open space in neighborhood), and `CP` (whether the neighborhood is in the center or periphery). β is a vector of coefficients and ε is a vector of random, normally distributed errors.

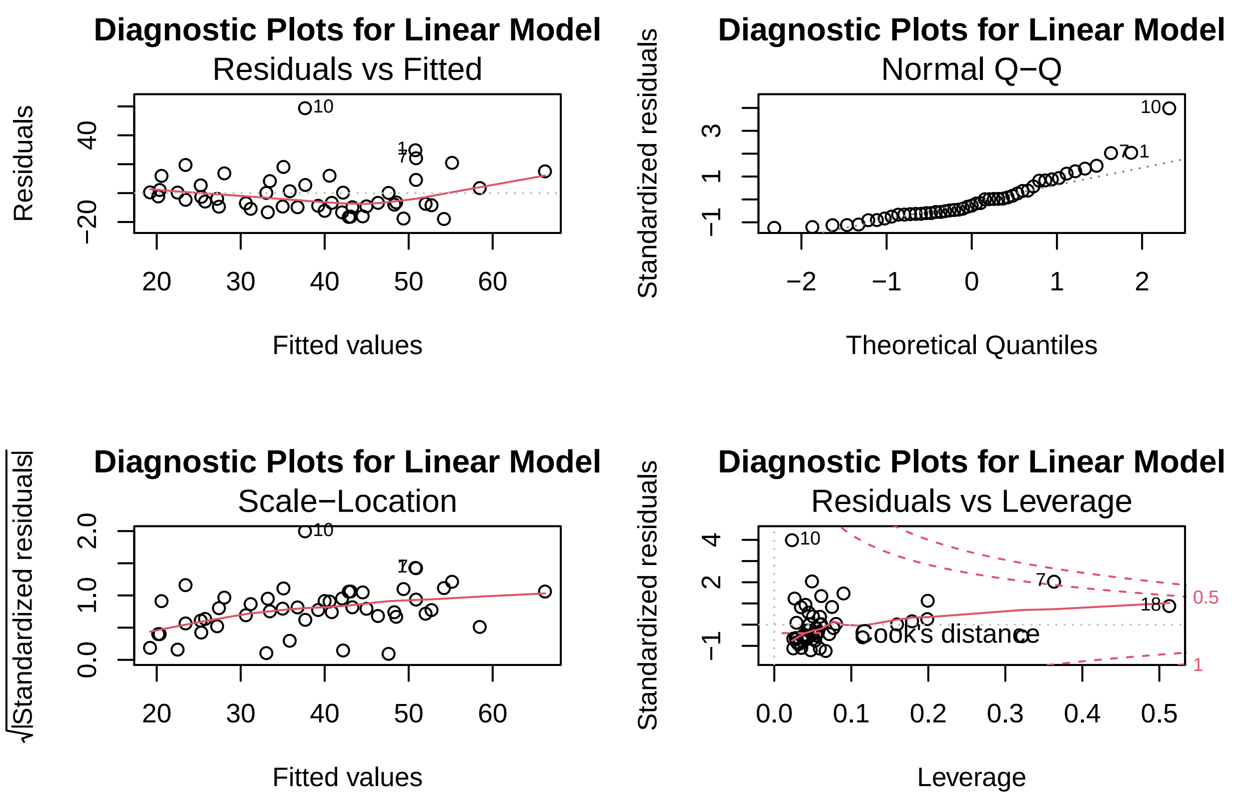
We estimate the model with

```r
col.lm <- lm(HOVAL~INC+CRIME+OPEN, data=columbus.sf)
summary(col.lm)
```

```
##
## Call:
## lm(formula = HOVAL ~ INC + CRIME + OPEN, data = columbus.sf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.902  -9.296  -3.969   5.608  58.742
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  46.7984    12.9397   3.617 0.000751 ***
## INC           0.4946     0.5316   0.930 0.357130
## CRIME        -0.5024     0.1795  -2.800 0.007509 **
## OPEN          0.7858     0.4677   1.680 0.099839 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 14.92 on 45 degrees of freedom
## Multiple R-squared:  0.3879, Adjusted R-squared:  0.3471
## F-statistic: 9.506 on 3 and 45 DF,  p-value: 5.588e-05
```

```
par(mfrow=c(2,2))
plot(col.lm, main = "Diagnostic Plots for Linear Model")
```



The diagnostic plots suggests that the assumptions of regression are not satisfied, which is to be expected since we know that most areal data is spatially dependent.

Simultaneous Autoregressive Models

SAR error model

To address the shortcomings of the linear model under the assumption of i.i.d. errors, we introduce **Simultaneous Autoregressive Models**. The models solve simultaneously for the regression coefficients and for the autoregressive error structure. In the **Spatial Error Model**, spatial autocorrelation enters in the specification only through the error terms.

To derive the model, we formulate the error as a first-order spatial autoregressive process

$$\varepsilon = \lambda W \varepsilon + u \tag{1}$$

where λ is the autoregressive parameter, W is the row-standardised spatial weights matrix W (that is, the weights are standardised such that $\Sigma_j W_{ij} = 1 \quad \text{for all} \quad i$), and u_i a random error term, assumed to be i.i.d. If $|\lambda| < 1$ and solving for ε yields

$$\varepsilon = (I - \lambda W)^{-1} u \tag{2}$$

We obtain the spatial error model by inserting ε into the standard regression model

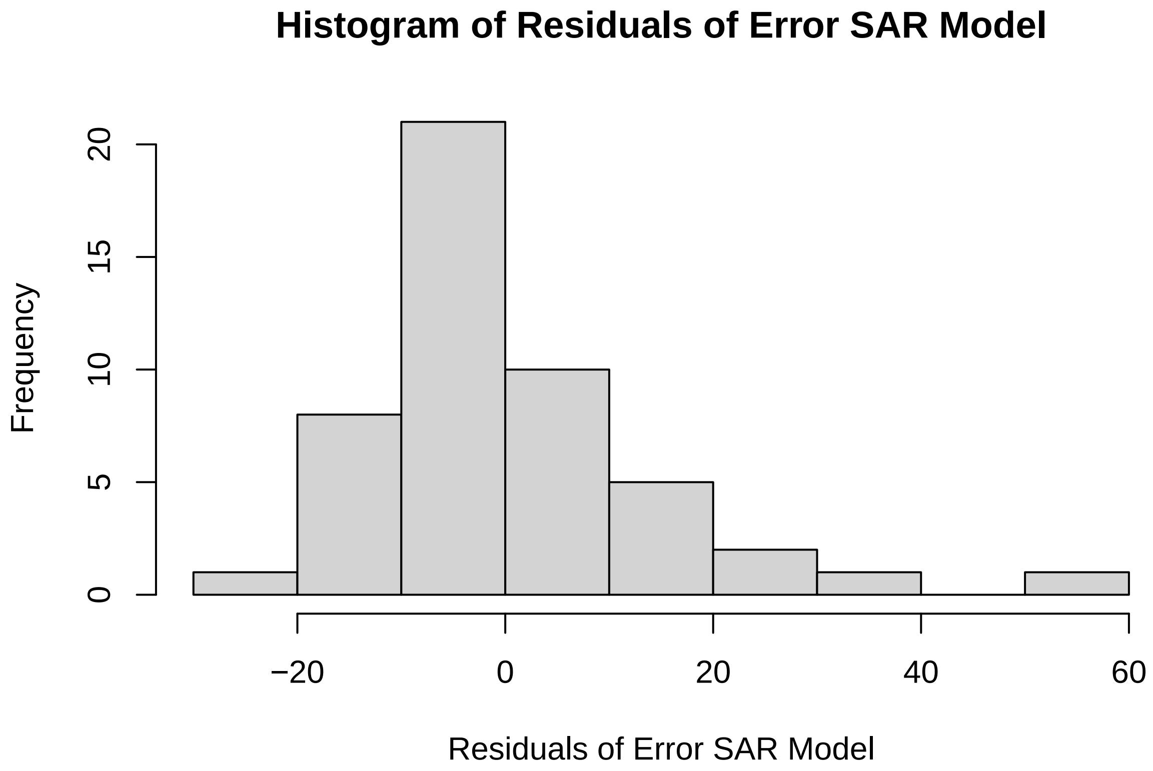
$$y = X\beta + (I - \lambda W)^{-1} u \tag{3}$$

where X is a matrix of the covariates `INC`, `CRIME`, `OPEN`, and `CP`. with $E\left[uu'\right] = \sigma^2 I$, which results in the following error variance-covariance matrix

$$E\left[\varepsilon\varepsilon'\right] = \sigma^2 (I - \lambda W)^{-1} \left(I - \lambda W'\right)^{-1} \tag{4}$$

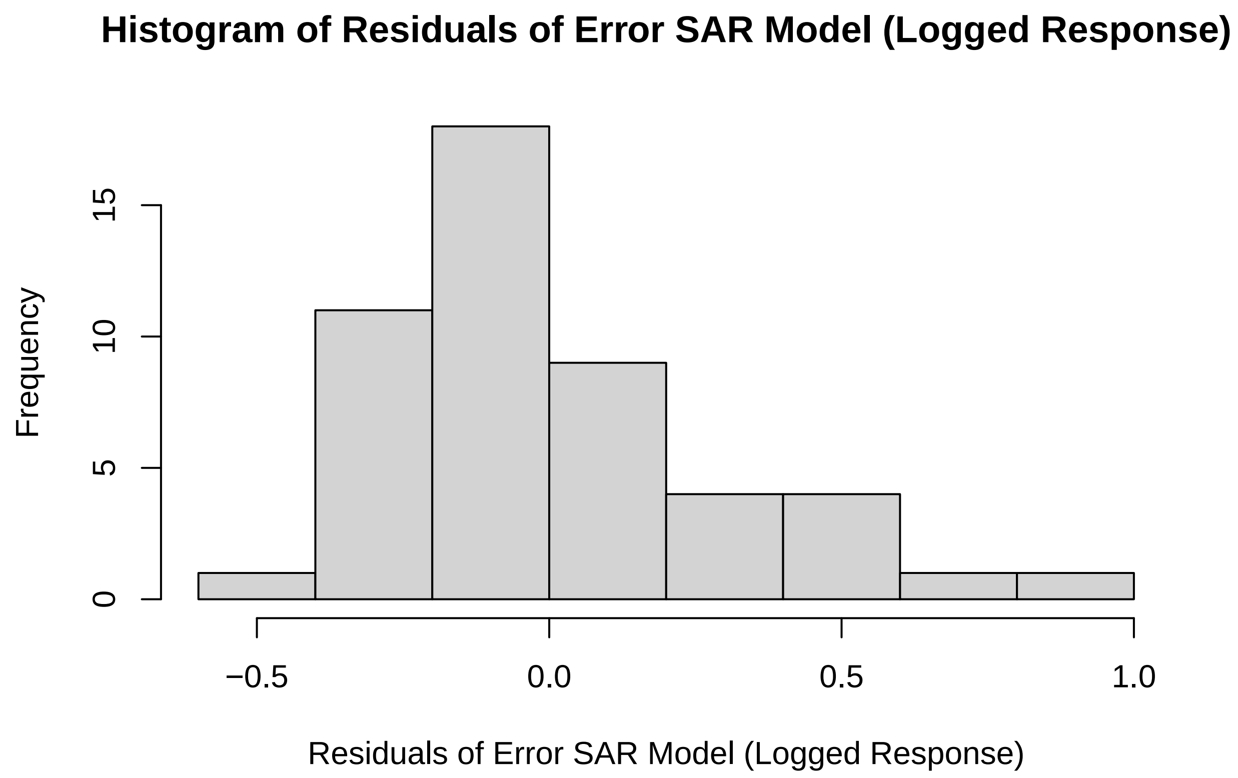
In order to estimate this model, we first create a list of spatial weights for neighbors. Then we estimate the model using the `errorsarlm` function from the `spatialreg` package.

```r
# make simple feature to neighborhood
columbus.nb <- poly2nb(columbus.sf)

# make neighborhood to list of weights
lw <- nb2listw(columbus.nb, style="W")

# estimate error SAR model without transformation
col.errW.eig <- errorsarlm(HOVAL~INC+CRIME+OPEN+CP, data=columbus.sf,
 lw, method="eigen", quiet=T)

# look at the residuals
hist(residuals(col.errW.eig), main = "Histogram of Residuals of Error SAR Model", xlab = "Residuals of
```



The residuals are not normally distributed. We therefore, log-transform the response to normalize them.

```
# estimate error SAR model with log transformation
col.errW.eig.log <- errorsarlm(log(HOVAL)~INC+CRIME+OPEN+CP, data=columbus.sf,
 lw, method="eigen", quiet=T)

hist(residuals(col.errW.eig.log), main = "Histogram of Residuals of Error SAR Model (Logged Response)",
```



```r
# print model summary
summary(col.errW.eig.log, correlation=TRUE)
```

```
##
## Call:
## errorsarlm(formula = log(HOVAL) ~ INC + CRIME + OPEN + CP, data = columbus.sf,
##     listw = lw, method = "eigen", quiet = T)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.503081 -0.190516 -0.051521  0.055445  0.855101
##
## Type: error
## Coefficients: (asymptotic standard errors)
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.7832959  0.2525868 14.9782  < 2e-16
## INC          0.0134509  0.0104770  1.2839  0.19919
## CRIME       -0.0099247  0.0042443 -2.3383  0.01937
## OPEN         0.0194551  0.0086313  2.2540  0.02420
## CP          -0.2520210  0.1435112 -1.7561  0.07907
##
## Lambda: 0.45625, LR test value: 5.3082, p-value: 0.021226
## Asymptotic standard error: 0.15476
##     z-value: 2.9482, p-value: 0.0031967
## Wald statistic: 8.6916, p-value: 0.0031967
##
```

```
## Log likelihood: -9.062429 for error model
## ML residual variance (sigma squared): 0.08037, (sigma: 0.2835)
## Number of observations: 49
## Number of parameters estimated: 7
## AIC: 32.125, (AIC for lm: 35.433)
##
##  Correlation of coefficients
##             sigma lambda (Intercept) INC   CRIME OPEN
## lambda      -0.20
## (Intercept)  0.00  0.00
## INC          0.00  0.00  -0.85
## CRIME        0.00  0.00  -0.67        0.38
## OPEN         0.00  0.00   0.04       -0.18  0.03
## CP           0.00  0.00  -0.09        0.20 -0.50 -0.17
```

We find that only CRIME and OPEN are significant at the $\alpha = 0.05$ level. For every additional major theft (residential burglaries and vehicle theft) per 1000 households, the predicted home value decreases by approximately \$1010 ($e^{0.0099247} = 1.009974$), holding other variables constant. For every additional unit (which is not provided in the description) of open space, the home value is expected to increase by \$1020 ($e^{0.0194551} = 1.019646$), holding other variables constant.

We also check whether there is some spatial dependence within the residuals of the model by performing Moran's I test. The test tests H_0: There is no spatial dependence against H_1: There is spatial dependence. We reject H_0 if the test statistic $< \alpha = 0.05$.

```
# Run Moran's I test for col.errW.eig.log
moran.test(residuals(col.errW.eig.log), lw) # not significant
```

```
##
##  Moran I test under randomisation
##
## data:  residuals(col.errW.eig.log)
## weights: lw
##
## Moran I statistic standard deviate = 0.36942, p-value = 0.3559
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##       0.013061791      -0.020833333       0.008418433
```

With a p-value of 0.3559, we fail to reject H_0. There does not seem to be any spatial dependence present in the residuals of our error model.

SAR lag and Durbin models

Next, we compute the SAR lag and SAR Durbin models on the log-transformed housing value with the same covariates used for the SAR error model.

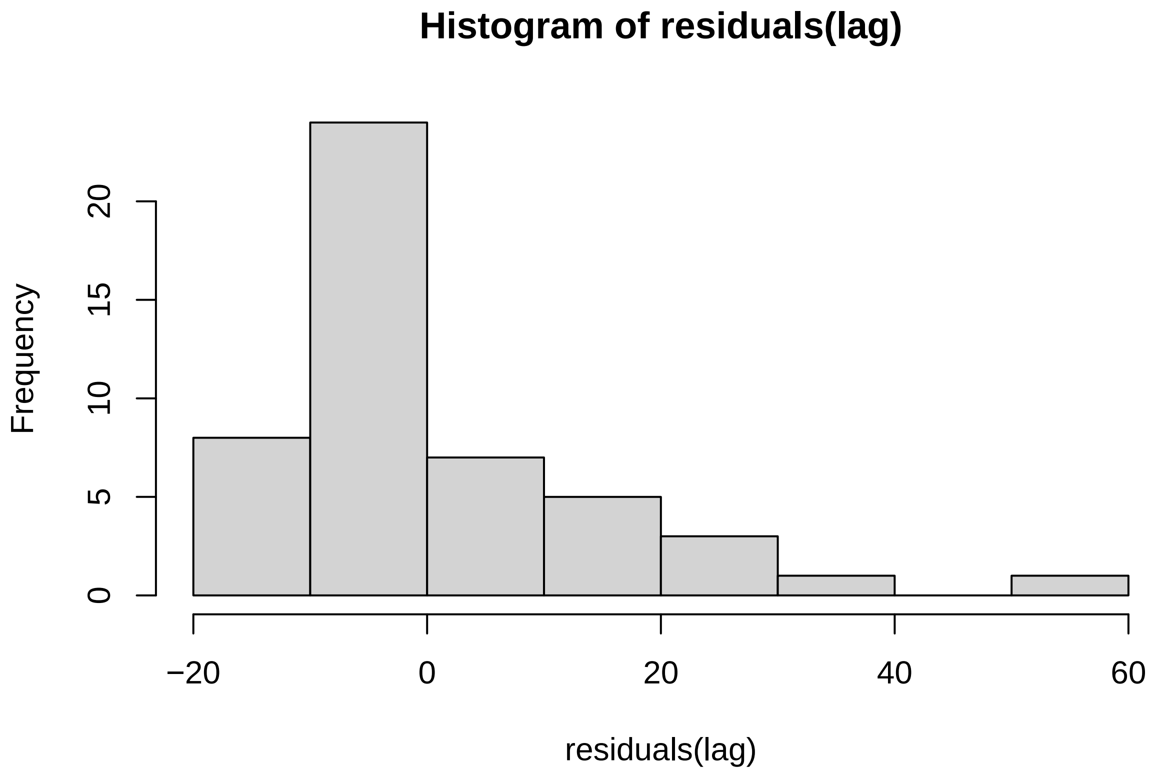
The SAR lag model assumes the form

$$y = \rho W y + X\beta + \varepsilon,$$

where W is the spatial neighbor matrix and ρ is a real parameter that controls the interaction strength between the data. Like the SAR error model, the SAR lag model is a global spatial model since it is autoregressive. In the lag model, the autoregressive structure comes from the data y rather than from the residuals. We histogram the SAR lag residuals below:

```
# estimate SAR lag model without transformation
lag <- lagsarlm(HOVAL~INC+CRIME+OPEN+CP, data=columbus.sf,
```

```r
 lw, Durbin=FALSE)

# look at the residuals
hist(residuals(lag))
```

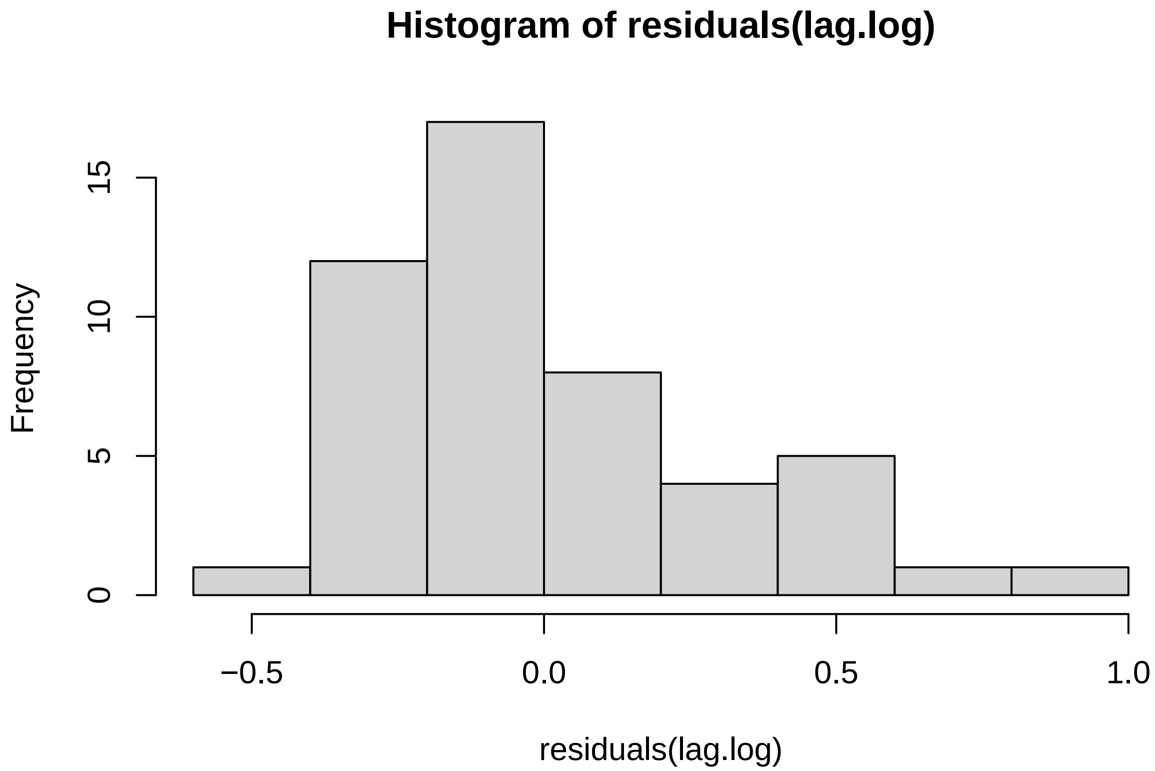


```r
# print model summary
summary(lag, correlation=TRUE)
```

```
##
## Call:lagsarlm(formula = HOVAL ~ INC + CRIME + OPEN + CP, data = columbus.sf,
##     listw = lw, Durbin = FALSE)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.6173  -8.1469  -3.9561   3.5458  54.2483
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 40.70225   14.07220  2.8924 0.003823
## INC          0.37919    0.50948  0.7443 0.456708
## CRIME       -0.37579    0.21022 -1.7876 0.073840
## OPEN         0.86299    0.44792  1.9267 0.054021
## CP          -5.47227    6.38795 -0.8567 0.391636
##
## Rho: 0.15013, LR test value: 0.72551, p-value: 0.39434
## Asymptotic standard error: 0.16869
##     z-value: 0.88999, p-value: 0.37347
## Wald statistic: 0.79209, p-value: 0.37347
##
## Log likelihood: -198.9841 for lag model
## ML residual variance (sigma squared): 196.14, (sigma: 14.005)
## Number of observations: 49
## Number of parameters estimated: 7
## AIC: 411.97, (AIC for lm: 410.69)
## LM test for residual autocorrelation
## test value: 4.1781, p-value: 0.04095
##
##  Correlation of coefficients
##             sigma rho   (Intercept) INC   CRIME OPEN
## rho         -0.06
## (Intercept)  0.03 -0.50
## INC          0.01 -0.12 -0.71
## CRIME       -0.01  0.11 -0.69        0.44
## OPEN         0.00  0.01  0.01       -0.18  0.07
## CP          -0.01  0.21 -0.08        0.14 -0.55 -0.19
```

The residuals do not appear normally distributed, so we log-transform the housing variable as before.

```
# estimate SAR lag model with log transformation
lag.log <- lagsarlm(log(HOVAL)~INC+CRIME+OPEN+CP, data=columbus.sf,
 lw, Durbin = FALSE)

hist(residuals(lag.log))
```



```r
# print model summary
summary(lag.log, correlation=TRUE)
```

```
##
## Call:
## lagsarlm(formula = log(HOVAL) ~ INC + CRIME + OPEN + CP, data = columbus.sf,
##     listw = lw, Durbin = FALSE)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.48184 -0.20602 -0.10922  0.11278  0.89848
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##               Estimate Std. Error z value  Pr(>|z|)
## (Intercept)  2.9330477  0.6201326  4.7297 2.248e-06
## INC          0.0109985  0.0109584  1.0037   0.31555
## CRIME       -0.0085071  0.0045184 -1.8828   0.05973
## OPEN         0.0218856  0.0096116  2.2770   0.02279
## CP          -0.1482052  0.1379725 -1.0742   0.28275
##
## Rho: 0.21738, LR test value: 1.6599, p-value: 0.19762
## Asymptotic standard error: 0.15432
##     z-value: 1.4086, p-value: 0.15895
## Wald statistic: 1.9842, p-value: 0.15895
##
```

```
## Log likelihood: -10.88657 for lag model
## ML residual variance (sigma squared): 0.090319, (sigma: 0.30053)
## Number of observations: 49
## Number of parameters estimated: 7
## AIC: 35.773, (AIC for lm: 35.433)
## LM test for residual autocorrelation
## test value: 4.6528, p-value: 0.031003
##
##  Correlation of coefficients
##             sigma rho   (Intercept) INC   CRIME OPEN
## rho         -0.08
## (Intercept)  0.07 -0.91
## INC          0.01 -0.14 -0.25
## CRIME       -0.01  0.12 -0.42         0.43
## OPEN         0.00  0.00  0.00       -0.18  0.07
## CP          -0.02  0.24 -0.20        0.13 -0.54 -0.19
```

As with the error model, the CRIME and OPEN covariates have the lowest p-values. However for this model, only OPEN falls below the threshold p-value of 0.05 instead of both OPEN and CRIME in the error model. The lag p-values are higher than those in the error model for all covariates except OPEN, suggesting less spatial correlation among the data overall in the lag model.

Next we test for spatial dependence among the residuals.

```
# Run Moran's I test
moran.test(residuals(lag.log), lw)
```

```
##
##  Moran I test under randomisation
##
## data:  residuals(lag.log)
## weights: lw
##
## Moran I statistic standard deviate = 1.5597, p-value = 0.05942
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##       0.122686854      -0.020833333       0.008467729
```

The p-value is 0.05942, so we cannot conclude that there is residual spatial dependence in the lag model. However we note that this p-value is markedly smaller than the p-value of 0.3559 in the error model.

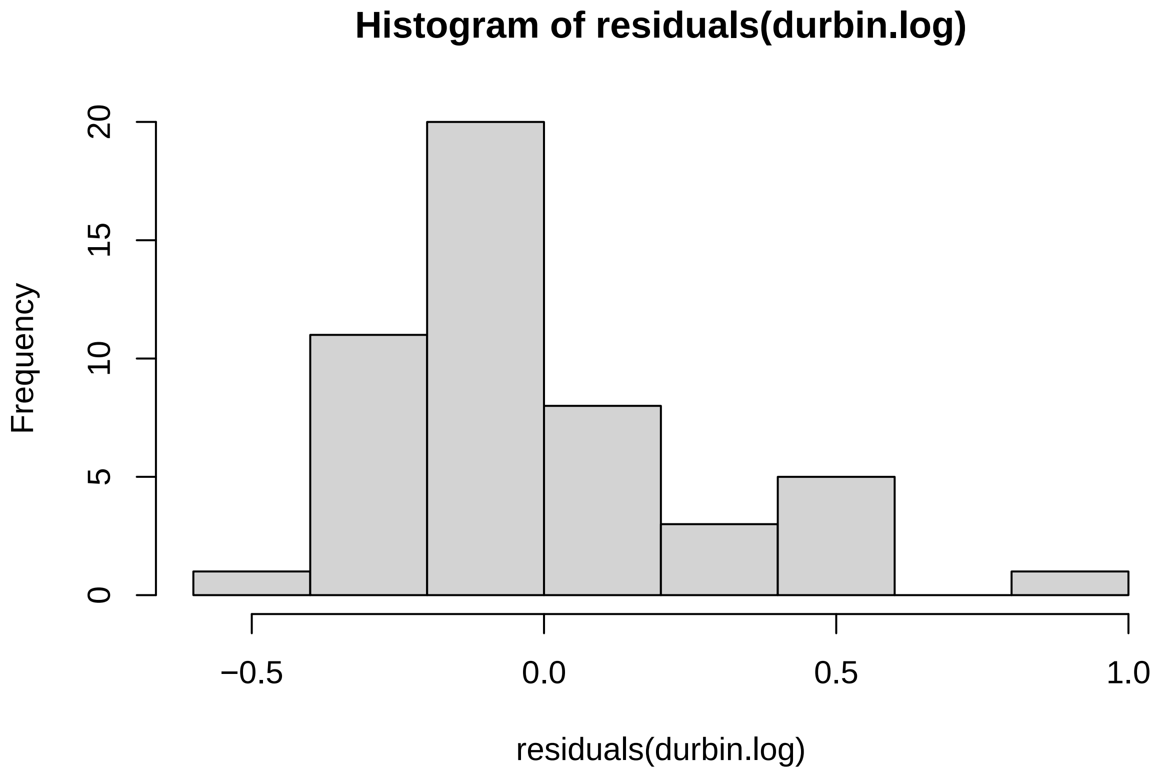
The spatial Durbin model (SDM) is a generalization of the SAR lag model and takes the form

$$y = \rho W y + W X \theta + X\beta + \varepsilon.$$

The additional term $WX\theta$ term introduces local spatial dependence to the model, where θ is a new model parameter. Hence, SDM in principle can capture both local and global dependence among the data.

We use this model on the log-transformed housing data as before:

```
# estimate Durbin model with log transformation
durbin.log <- lagsarlm(log(HOVAL)~INC+CRIME+OPEN+CP, data=columbus.sf,
 lw, Durbin = TRUE)

hist(residuals(durbin.log))
```



```
# print model summary
summary(durbin.log, correlation=TRUE)
```

```
##
## Call:
## lagsarlm(formula = log(HOVAL) ~ INC + CRIME + OPEN + CP, data = columbus.sf,
##     listw = lw, Durbin = TRUE)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.420011 -0.195515 -0.060449  0.087907  0.888759
##
## Type: mixed
## Coefficients: (asymptotic standard errors)
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.9491177  0.8220514  2.3710 0.017738
## INC          0.0124874  0.0104106  1.1995 0.230339
## CRIME       -0.0098983  0.0041739 -2.3715 0.017715
## OPEN         0.0250355  0.0091306  2.7419 0.006108
## CP          -0.3979069  0.1578926 -2.5201 0.011732
## lag.INC      0.0032787  0.0192872  0.1700 0.865014
## lag.CRIME    0.0094498  0.0092929  1.0169 0.309209
## lag.OPEN     0.0298926  0.0249782  1.1967 0.231405
## lag.CP       0.3121857  0.2454742  1.2718 0.203456
##
## Rho: 0.36181, LR test value: 3.6519, p-value: 0.056004
## Asymptotic standard error: 0.16861
##     z-value: 2.1458, p-value: 0.031888
## Wald statistic: 4.6045, p-value: 0.031888
##
## Log likelihood: -6.784626 for mixed model
## ML residual variance (sigma squared): 0.074806, (sigma: 0.27351)
## Number of observations: 49
## Number of parameters estimated: 11
## AIC: 35.569, (AIC for lm: 37.221)
## LM test for residual autocorrelation
## test value: 0.023684, p-value: 0.87769
##
##  Correlation of coefficients
##             sigma rho   (Intercept) INC   CRIME OPEN  CP    lag.INC lag.CRIME
## rho         -0.16
## (Intercept)  0.11 -0.72
## INC          0.00 -0.01 -0.24
## CRIME        0.01 -0.04 -0.16        0.38
## OPEN         0.02 -0.10 -0.03       -0.16  0.08
## CP           0.00 -0.03  0.13        0.17 -0.40 -0.19
## lag.INC      0.02 -0.15 -0.44       -0.12  0.01  0.11 -0.14
## lag.CRIME   -0.02  0.13 -0.65        0.13 -0.09  0.15 -0.05  0.60
## lag.OPEN     0.03 -0.20 -0.01       -0.02  0.08  0.26 -0.12  0.01    0.20
## lag.CP      -0.02  0.15  0.03       -0.19 -0.01 -0.09 -0.49  0.00   -0.48
##             lag.OPEN
## rho
## (Intercept)
## INC
## CRIME
## OPEN
## CP
## lag.INC
## lag.CRIME
## lag.OPEN
## lag.CP      -0.16
```

In this model, the covariates CP,CRIME and now also OPEN have significant p-values, and overall the p-values are lower in this model than the other SAR models considered. Since the Durbin model is the only SAR model we've considered that includes local spatial dependence of the data, it is possible that the local dependence is more significant, or at least that both local and global effects are necessary to describe the spatial dependence of the data.

A final remark on SAR models: we can further generalize the Durbin model to include autoregressive residuals as seen in the error model. The resulting model generalizes each of the 3 SAR models we have analyzed and assumes the form

$$y = \rho Wy + WX\theta + X\beta + (I - \lambda W)^{-1}u.$$

Likelihood Ratio

Since the SAR error and SAR lag models are special cases of the SAR Durbin model, we compare the performance of these models via likelihood ratio tests. These tests take the form

$$\lambda_{\text{LR}} = -2\ln\left[\frac{\sup_{\theta\in\Theta_0}\mathcal{L}(\theta)}{\sup_{\theta\in\Theta}\mathcal{L}(\theta)}\right],$$

where Θ_0 denotes the restricted parameter space of the restricted model (in our case, error or lag models) and Θ denotes the full parameter space of the general model (in our case, the Durbin model).

We compare the error SAR, SAR lag and SAR Durbin models on the log(HOVAL) data with identical covariates INC, CRIME, OPEN and CP using likelihood ratios below:

```
library(lmtest)

lrtest(col.errW.eig.log,durbin.log) # error vs. Durbin
```

```
## Likelihood ratio test
##
## Model 1: log(HOVAL) ~ INC + CRIME + OPEN + CP
## Model 2: log(HOVAL) ~ INC + CRIME + OPEN + CP
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -9.0624
## 2  11 -6.7846  4 4.5556      0.336
```

```
lrtest(lag.log, durbin.log)         # lag vs. Durbin
```

```
## Likelihood ratio test
##
## Model 1: log(HOVAL) ~ INC + CRIME + OPEN + CP
## Model 2: log(HOVAL) ~ INC + CRIME + OPEN + CP
##   #Df   LogLik Df  Chisq Pr(>Chisq)
## 1   7 -10.8866
## 2  11  -6.7846  4 8.2039    0.08439 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The first value of 0.336 is very large, so we cannot reject the null-hypothesis that the error model fits the data. The second value of 0.08439 is lower but still above 0.05, so again we cannot reject the hypothesis that the lag model fits the data. This seems to imply that the class of SAR error model outperforms the SAR lag model on our data.

Conditional Autoregressive Models

In the next step, we implement the conditional autoregressive (CAR) model on the Columbus data to study the impact of covariates on House values in Columbus, OH, 1980 with spatial information.

The CAR model essentially assumes the spatial estimation is conditional on the value of neighbors. As a typical Bayes model, CAR model assumes prior distribution on the model parameters and applies computationally intensive sampling techniques like Markov Chain Monte Carlo (MCMC) or MCMC with Gibbs sampling to find the fitted parameters.

Since we have one response variable *House value*, the CAR model has only one random effect ψ_k at each spatial location k. Because *House value* is continuous, Gaussian distribution is preferred for the CAR model and we take logarithm of the *House value* as well to make it normally distributed.

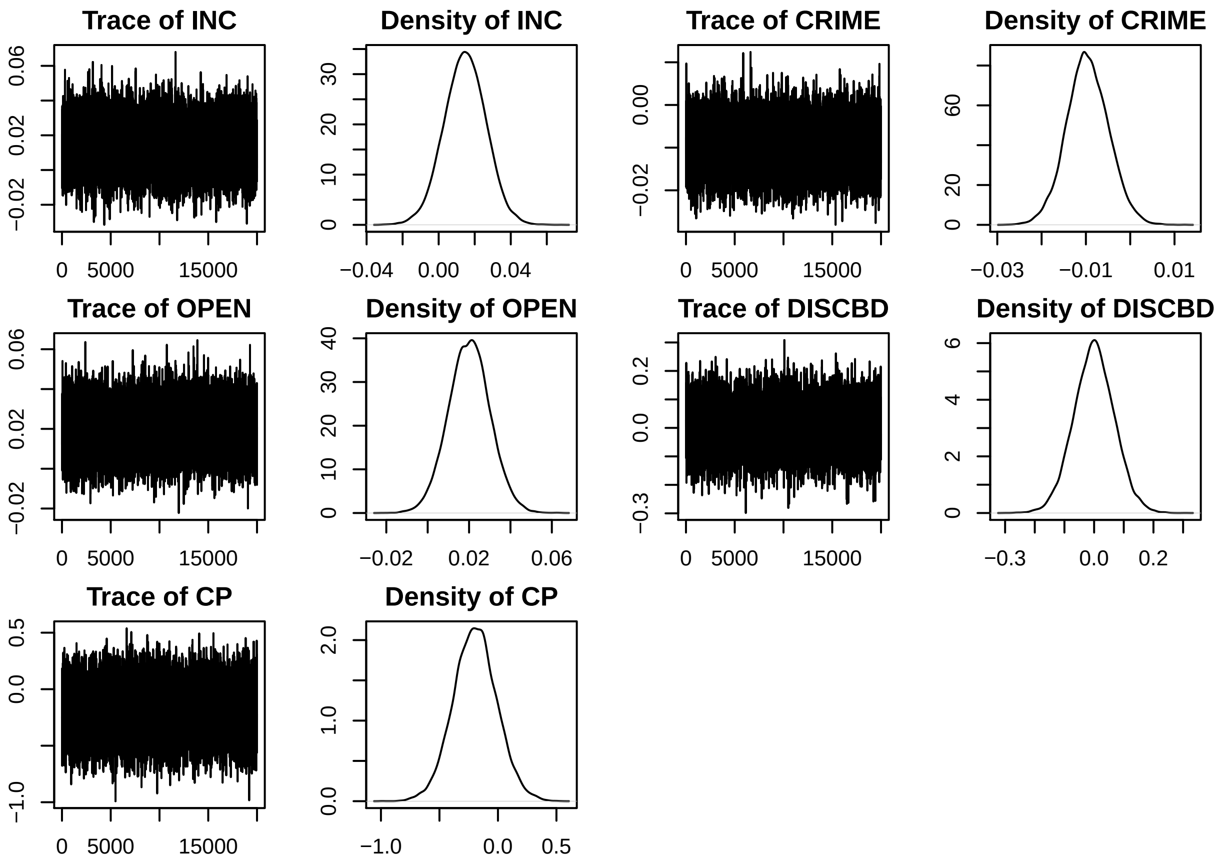
The package *CARBayes* by Duncan Lee is used to apply the CAR model in R. The specific function we use is *S.CARleroux*, where it specifies a CAR model proposed by Brian G. Leroux in 2000. The model expression is as follows.

$$\begin{aligned} log(y_k) \mid \mu_k &\sim N\left(\mu_k, \nu^2\right) \qquad \text{for } k = 1, \ldots, K \\ g\left(\mu_k\right) &= \beta_I Inc_k + \beta_C Crime_k + \beta_O Open_k + \beta_D DisCBD_k + \beta_{CP1} \mathbf{1}\{CP_k = 1\} + \psi_k \end{aligned}$$

$$
\begin{aligned}
\psi_k \mid \psi_{-k}, \mathbf{W}, \tau^2, \rho &\sim \mathrm{N}\left(\frac{\rho \sum_{i=1}^{K} w_{ki}\psi_i}{\rho \sum_{i=1}^{K} w_{ki} + 1 - \rho}, \frac{\tau^2}{\rho \sum_{i=1}^{K} w_{ki} + 1 - \rho}\right) \\
[\beta_I, \beta_C, \beta_O, \beta_D, \beta_{CP1}]^T &\sim \mathrm{N}\left(\mathbf{0}, 10^5 \boldsymbol{I}\right) \\
\nu^2 &\sim \text{Inverse-Gamma}(1, 0.01) \\
\tau^2 &\sim \mathrm{Inverse-Gamma}(1, 0.01) \\
\rho &\sim \text{ Uniform } (0, 1)
\end{aligned}
$$

where

- $[\beta_I, \beta_C, \beta_O, \beta_D, \beta_{CP1}]^T$ are the linear coefficients for predictors.
- ν^2 measures the variance for the logarithm of the housing values $(log(y_k))$.
- ψ_k is the spatial structure component that measures the spatial autocorrelation.
- $\mathbf{W}$ is a neighborhood or adjacency matrix.
- ρ is a spatial correlation parameter, where $\rho = 0$ means spatial independence and $\rho = 1$ indicates strong spatial correlation.



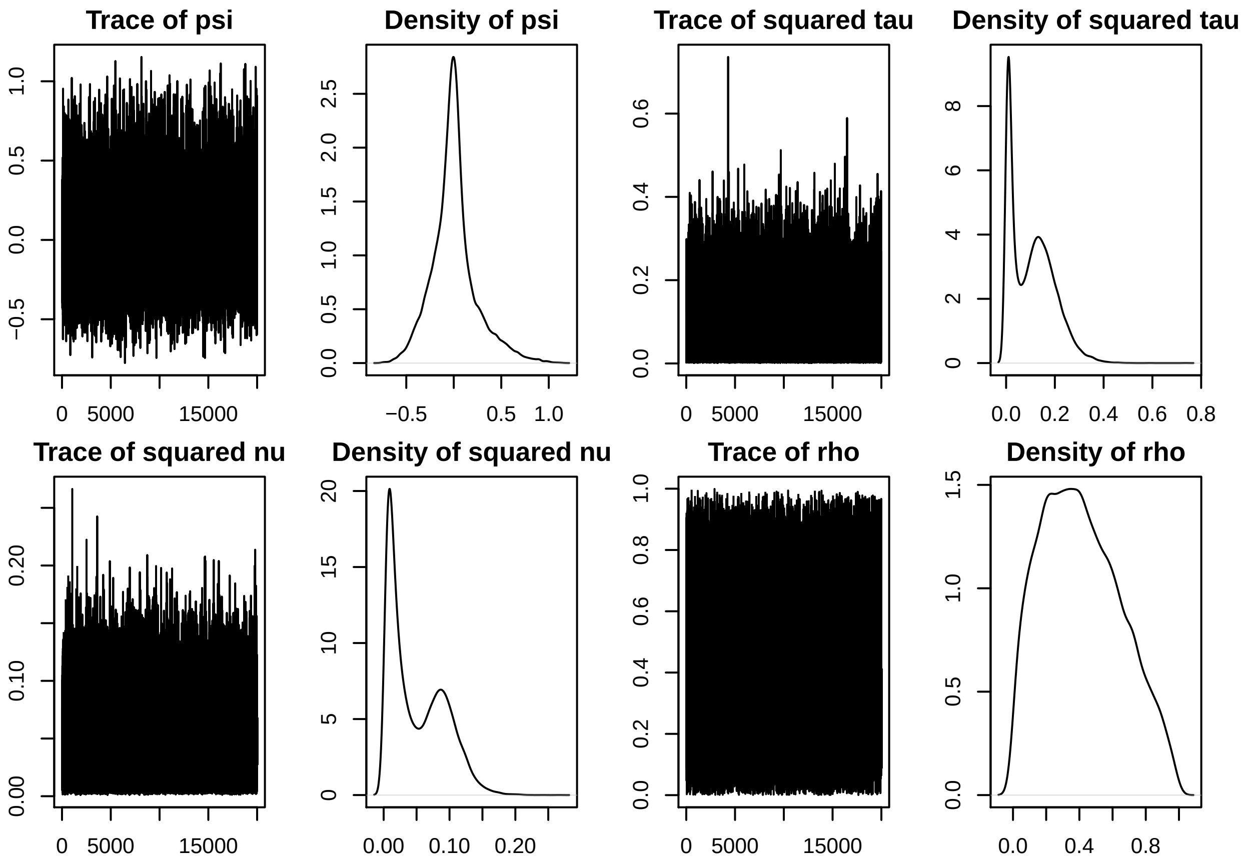
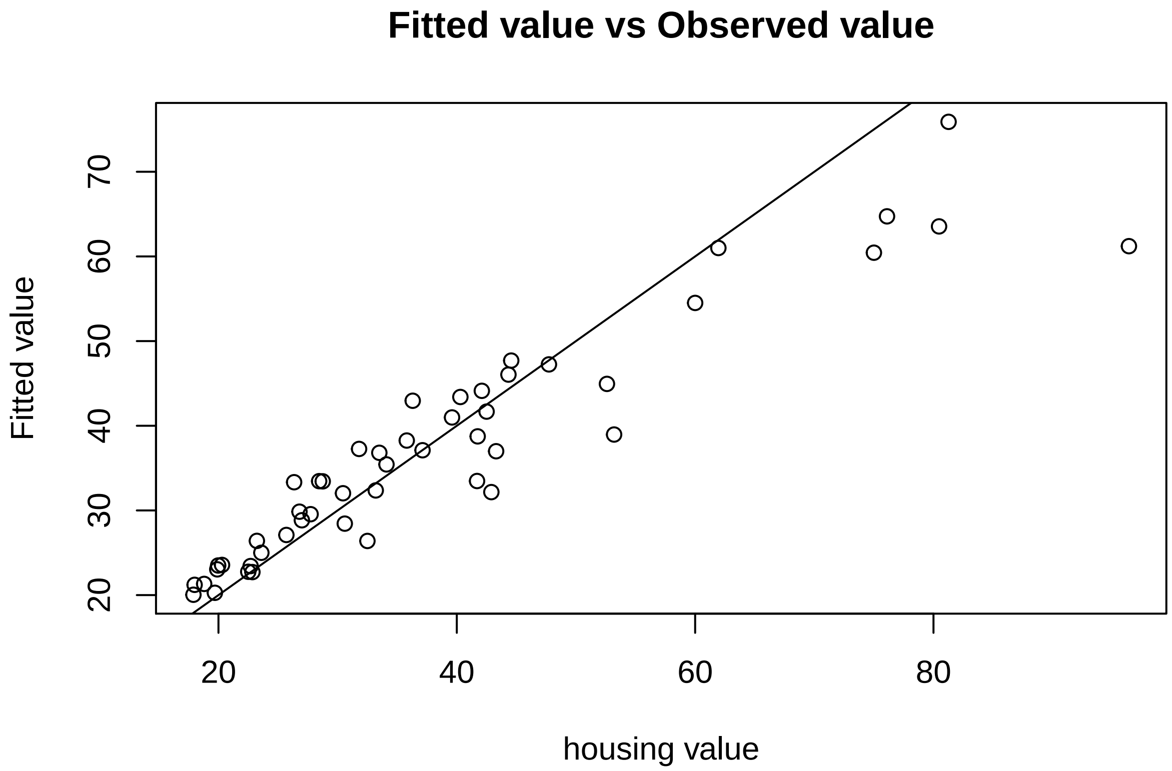


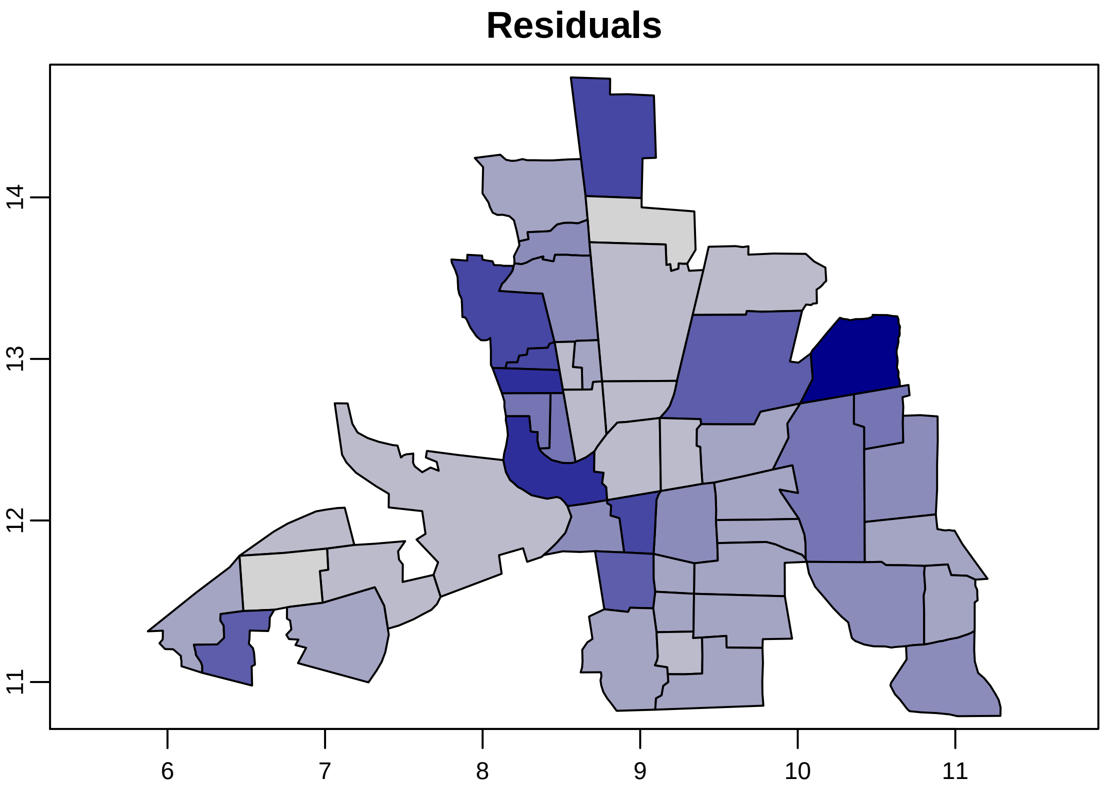
Table 1: 95% confidence intervals for model coefficients

	0.5	0.025	0.975
(Intercept)	3.7227	2.9525	4.4936
INC	0.0147	-0.0084	0.0371
CRIME	-0.0097	-0.0189	0.0000
OPEN	0.0201	0.0003	0.0400
DISCBD	-0.0002	-0.1345	0.1352
CP1	-0.1957	-0.5607	0.1764



The log likelihood of this model is 17.97 while the training root mean square error (RMSE) is 52.6. From the table of 95% confidence intervals for coefficients, we have

- *CRIME* and *OPEN* are the two variables that is significant since their confidence intervals don't have zero involved.
- Hold other predictors fixed, regions with **less crimes** tend to have higher *House values.*
- Hold other predictors fixed, regions with **higher household incomes** tend to have higher *House values.*
- Hold other predictors fixed, regions with **more open area** tend to have higher *House values.*
- Hold other predictors fixed, regions with **closer distance to CBD** tend to have higher *House values.*
- Hold other predictors fixed, **core** regions tend to have higher *House values.*



```
# Moran I
Moran.I(residuals(car_model_gaussian), ww)$p.value
```

```
## [1] 0.09593303
```

We also tested the Moran's I autocorrelation coefficient for the residuals. It shows there is **no** significant spatial correlation of the residuals since the p-value $p = 0.1 > 0.05$. Therefore, our CAR model fits the areal data nicely and leaves no significant spatial information in the residuals.

Conclusions

From our exploratory analysis and visualizing the data we know the variables of interest, with the exception of open space, have weak to moderate positive spatial autocorrelation, meaning neighboring polygons are more similar to each other than the dataset as a whole. All of the models we tried did not have significant ($\alpha = 0.05$) spatial autocorrelation.

We found that across all of our models average house value significantly increased with open space (units unknown). In some of the models (SAR, DURBIN, CAR) housing value decreased significantly with crime (per 1000 households). In the durbin log model and CAR model the effect of being located in an urban (core) neighborhood was also negative.

The best fit model by log likelihood was the CAR model. There was no significant difference in fit between the SAR lag and Durbin models.

References

Bivand, Roger. 2021. *Spdep: Spatial Dependence: Weighting Schemes, Statistics.* https://github.com/rspatial/spdep/.

Bivand, Roger, Giovanni Millo, and Gianfranco Piras. 2021. "A Review of Software for Spatial Econometrics in R." *Mathematics* 9 (11). https://doi.org/10.3390/math9111276.

Bivand, Roger, Jakub Nowosad, and Robin Lovelace. 2021. *SpData: Datasets for Spatial Analysis.* https://nowosad.github.io/spData/.

Bivand, Roger S., Edzer Pebesma, and Virgilio Gomez-Rubio. 2013a. *Applied Spatial Data Analysis with R, Second Edition.* Springer, NY. https://asdar-book.org/.

———. 2013b. *Applied Spatial Data Analysis with R, Second Edition.* Springer, NY. https://asdar-book.org/.

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.

Furrer, Reinhard, and the Applied Statistics Group. 2021. "Modeling Dependent Data: An Excursion." http://user.math.uzh.ch/furrer/download/sta330/script_sta330.pdf.

Henry, Lionel, and Hadley Wickham. 2020. *Purrr: Functional Programming Tools.* https://CRAN.R-project.org/package=purrr.

Hijmans, Robert J. 2021. *Terra: Spatial Data Analysis.* https://rspatial.org/terra/.

Hothorn, Torsten, Achim Zeileis, Richard W. Farebrother, and Clint Cummins. 2019. *Lmtest: Testing Linear Regression Models.* https://CRAN.R-project.org/package=lmtest.

Lee, Duncan. 2013. "CARBayes: An R Package for Bayesian Spatial Modeling with Conditional Autoregressive Priors." *Journal of Statistical Software* 55 (13): 1–24.

Müller, Kirill, and Hadley Wickham. 2020. *Tibble: Simple Data Frames.* https://CRAN.R-project.org/package=tibble.

Paradis, Emmanuel, Simon Blomberg, Ben Bolker, Joseph Brown, Santiago Claramunt, Julien Claude, Hoa Sien Cuong, et al. 2020. *Ape: Analyses of Phylogenetics and Evolution.* http://ape-package.ird.fr/.

Paradis, E., and K. Schliep. 2019. "Ape 5.0: An Environment for Modern Phylogenetics and Evolutionary Analyses in R." *Bioinformatics* 35: 526–28.

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.

———. 2021. *Sf: Simple Features for R.* https://CRAN.R-project.org/package=sf.

Pebesma, Edzer, and Roger Bivand. 2020. *Sp: Classes and Methods for Spatial Data.* https://CRAN.R-project.org/package=sp.

Pebesma, Edzer J., and Roger S. Bivand. 2005. "Classes and Methods for Spatial Data in R." *R News* 5 (2): 9–13. https://CRAN.R-project.org/doc/Rnews/.

R Core Team. 2020. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Schloerke, Barret, Di Cook, Joseph Larmarange, Francois Briatte, Moritz Marbach, Edwin Thoen, Amos Elberg, and Jason Crowley. 2021. *GGally: Extension to Ggplot2.* https://CRAN.R-project.org/package=GGally.

Ver Hoef, Jay M, Ephraim M Hanks, and Mevin B Hooten. 2018. "On the Relationship Between Conditional (Car) and Simultaneous (Sar) Autoregressive Models." *Spatial Statistics* 25: 68–85.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. https://ggplot2.tidyverse.org.

———. 2019a. *Stringr: Simple, Consistent Wrappers for Common String Operations.* https://CRAN.R-project.org/package=stringr.

———. 2019b. *Tidyverse: Easily Install and Load the Tidyverse*. https://CRAN.R-project.org/package=tidyverse.

———. 2020. *Forcats: Tools for Working with Categorical Variables (Factors)*. https://CRAN.R-project.org/package=forcats.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Winston Chang, Lionel Henry, Thomas Lin Pedersen, Kohske Takahashi, Claus Wilke, Kara Woo, Hiroaki Yutani, and Dewey Dunnington. 2021. *Ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*. https://CRAN.R-project.org/package=ggplot2.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2020. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Wickham, Hadley, and Lionel Henry. 2020. *Tidyr: Tidy Messy Data*. https://CRAN.R-project.org/package=tidyr.

Wickham, Hadley, Jim Hester, and Romain Francois. 2018. *Readr: Read Rectangular Text Data*. https://CRAN.R-project.org/package=readr.

Zeileis, Achim, and Gabor Grothendieck. 2005. "Zoo: S3 Infrastructure for Regular and Irregular Time Series." *Journal of Statistical Software* 14 (6): 1–27. https://doi.org/10.18637/jss.v014.i06.

Zeileis, Achim, Gabor Grothendieck, and Jeffrey A. Ryan. 2020. *Zoo: S3 Infrastructure for Regular and Irregular Time Series (Z's Ordered Observations)*. http://zoo.R-Forge.R-project.org/.

Zeileis, Achim, and Torsten Hothorn. 2002. "Diagnostic Checking in Regression Relationships." *R News* 2 (3): 7–10. https://CRAN.R-project.org/doc/Rnews/.