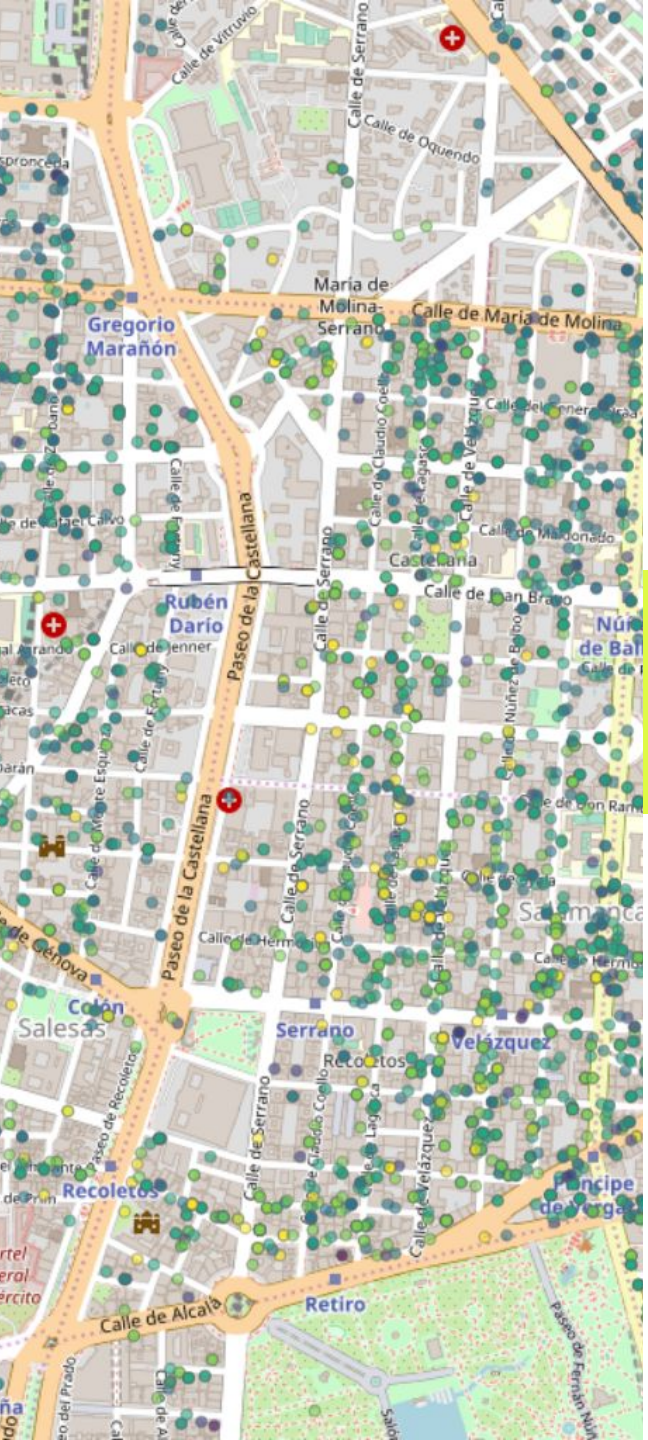
idealista/data

Spatial Autocorrelation is **EVERYWHERE**

Pelayo Arbués
Data Scientist
@pelayoarbues



AGENDA

What is Spatial Autocorrelation?

How to test it?

Quick note on Spatial Cross Validation

“Everything is related to everything else, but near things are more related than distant things”

Waldo Tobler (1970)

Spatial Autocorrelation: What is it? Why you should care

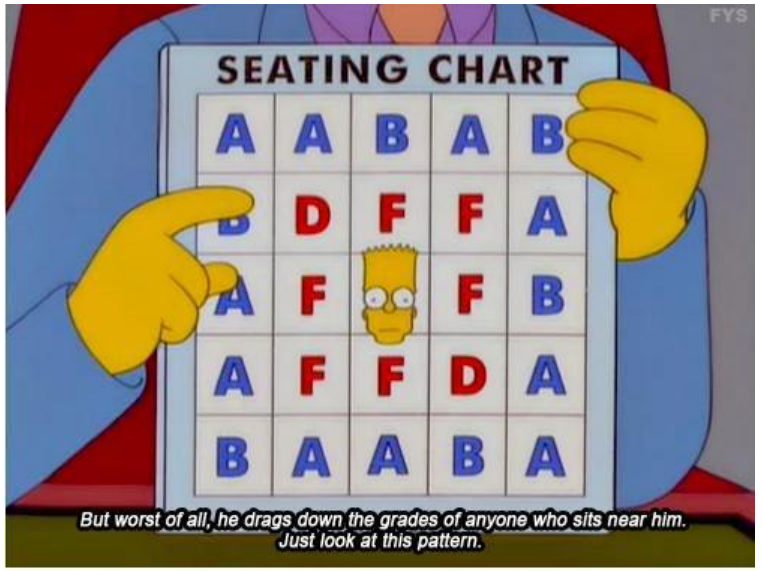
Relationship between nearby locations of the realization of a single variable

- Positive:
  - Similar values in Similar location. (Clustered)

- Negative:
  - Similar values further apart. (Checkerboard pattern)

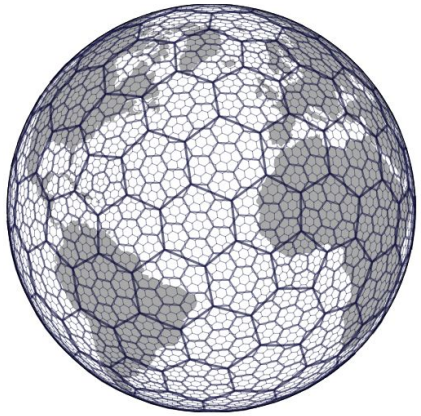
Uses of Spatial Autocorrelation (Getis, 2010):

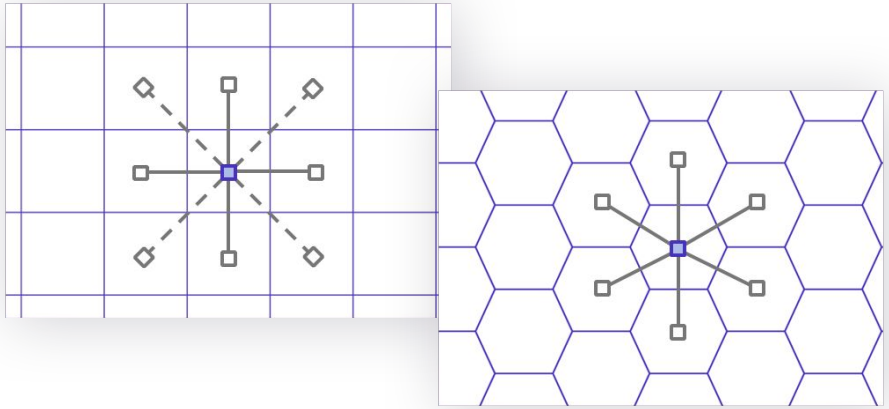
- A test on model misspecification
- A measure of spatial effects
- A test on spatial heterogeneity
- A means of identifying spatial clusters
- A way to understand Modifiable Areal Unit Problem (MAUP)
- A means of identifying outliers (spatial and non spatial)
- ...



idealista/data

Introducing Uber's H3





- Data points are bucketed in hexagons
- Hexagons have regular shapes vs Postal areas, Census tracts and other administrative polygons
- H3 supports sixteen resolutions. Each finer resolution has cells with one seventh the area of the coarser resolution.
- Square grids have two different neighbors: edge and vertex

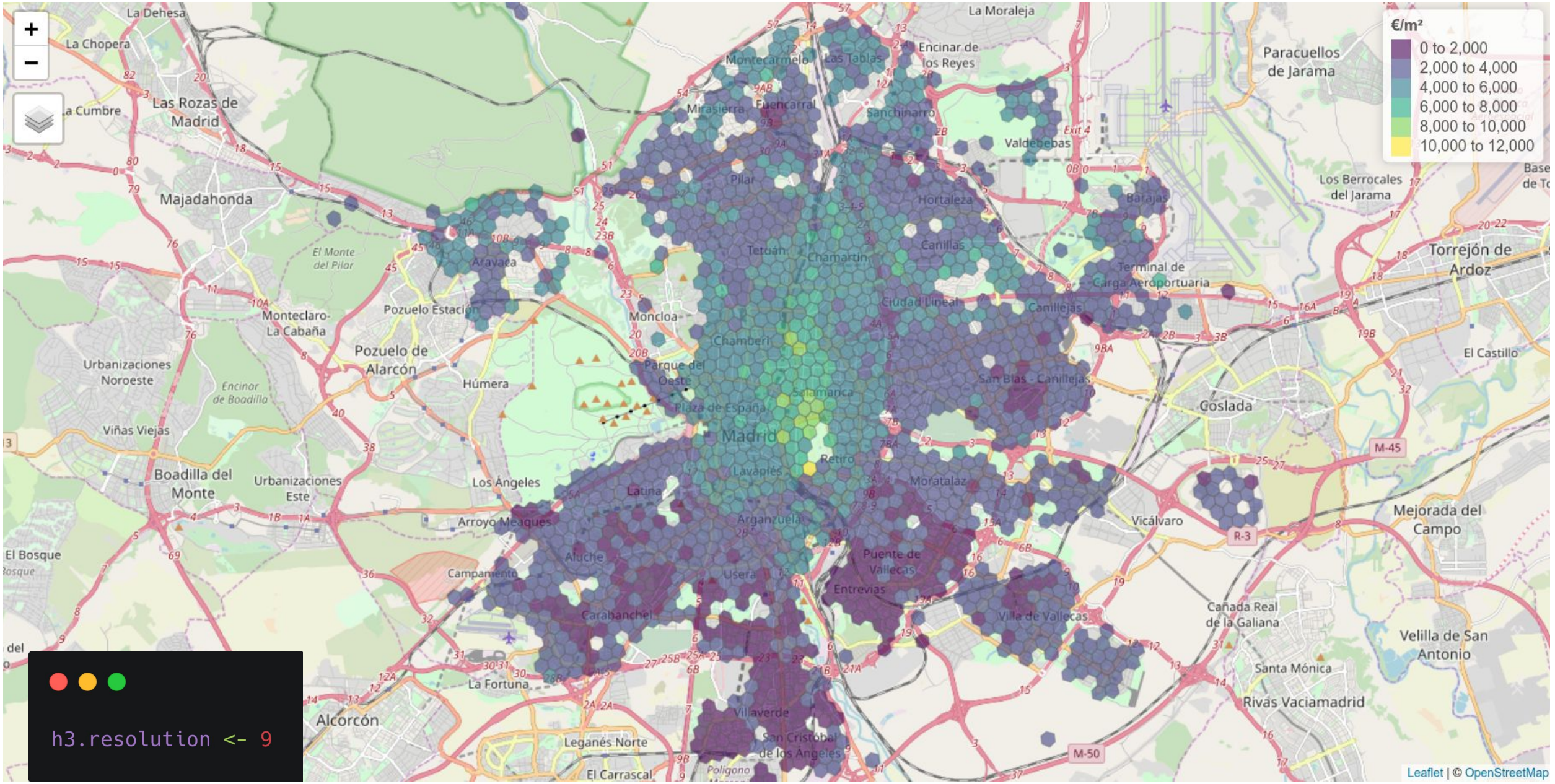
```
devtools::install_github("crazycapivara/h3-r")
```

Source: https://eng.uber.com/h3/

idealista/data

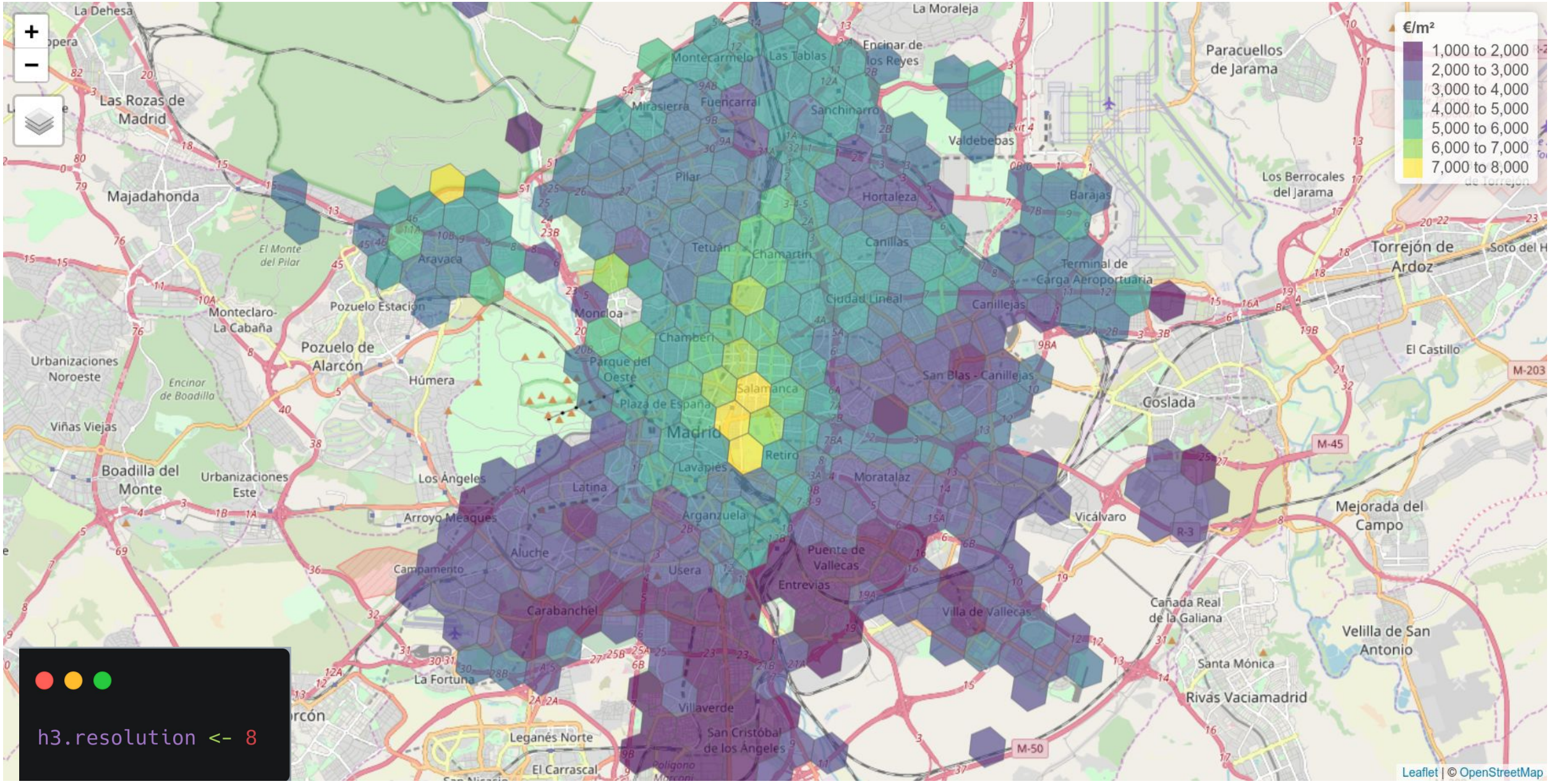

```r
#Define H3 Resolution
h3.resolution <- 8

df.h3 <- df %>%
  mutate(h3_index = geo_to_h3(df.sf,
                              h3.resolution)) %>%
  group_by(h3_index) %>%
  summarise(median.unitprice = median(UNITPRICE)) %>%
  ungroup()

hexagons <- h3_to_geo_boundary_sf(df.h3$h3_index)

st_geometry(df.h3) <- hexagons$geometry
```

idealista/data




```
h3.resolution <- 9
```

idealista/data



Testing for Spatial Autocorrelation:

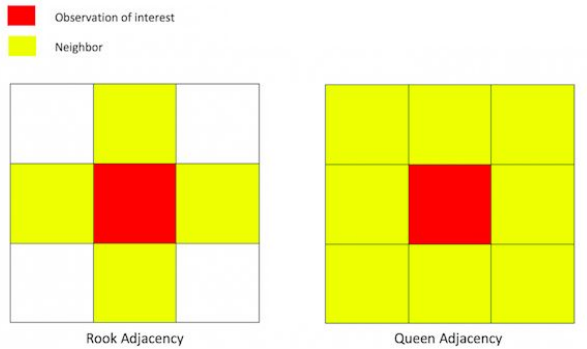
1. Choose a neighborhood criterion ? Which areas are linked?

2. Assign weights to the areas that are linked ? Create a spatial weights matrix

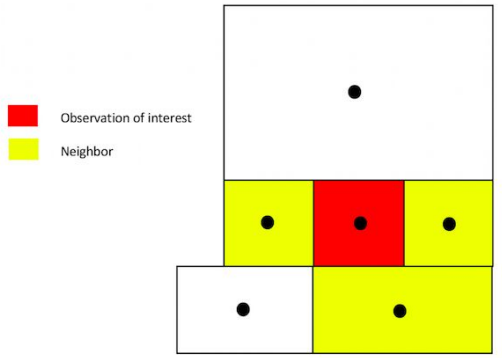
3. Run statistical test, using weights matrix, to examine spatial autocorrelation

Testing for Spatial Autocorrelation:

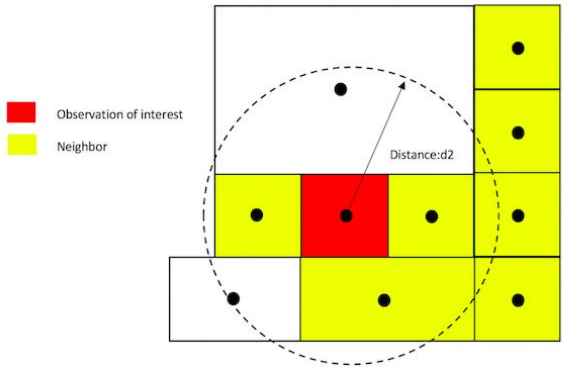
1. Choose a neighborhood criterion ? Which areas are linked?



2. Assign weights to the areas that are linked ? Create a spatial weights matrix



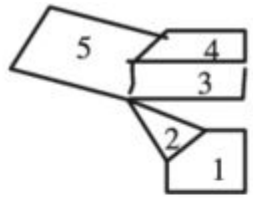
3. Run statistical test, using weights matrix, to examine spatial autocorrelation



Source: https://crd230.github.io/

Testing for Spatial Autocorrelation:

1. Choose a neighborhood criterion ? Which areas are linked?



2. Assign weights to the areas that are linked ? Create a spatial weights matrix (W)

$$B = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} \qquad W = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 1/2 & 1/2 & 0 \end{pmatrix}$$

3. Run statistical test, using weights matrix, to examine spatial autocorrelation

$$Y^* = WY = W \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix} = \begin{pmatrix} y_2 \\ y_1 \\ 1/2y_4 + 1/2y_5 \\ 1/2y_3 + 1/2y_5 \\ 1/2y_3 + 1/2y_4 \end{pmatrix}$$

Testing for Spatial Autocorrelation:

1. Choose a neighborhood criterion ? Which areas are linked?

2. Assign weights to the areas that are linked ? Create a spatial weights matrix

3. Run statistical test, using weights matrix, to examine spatial autocorrelation

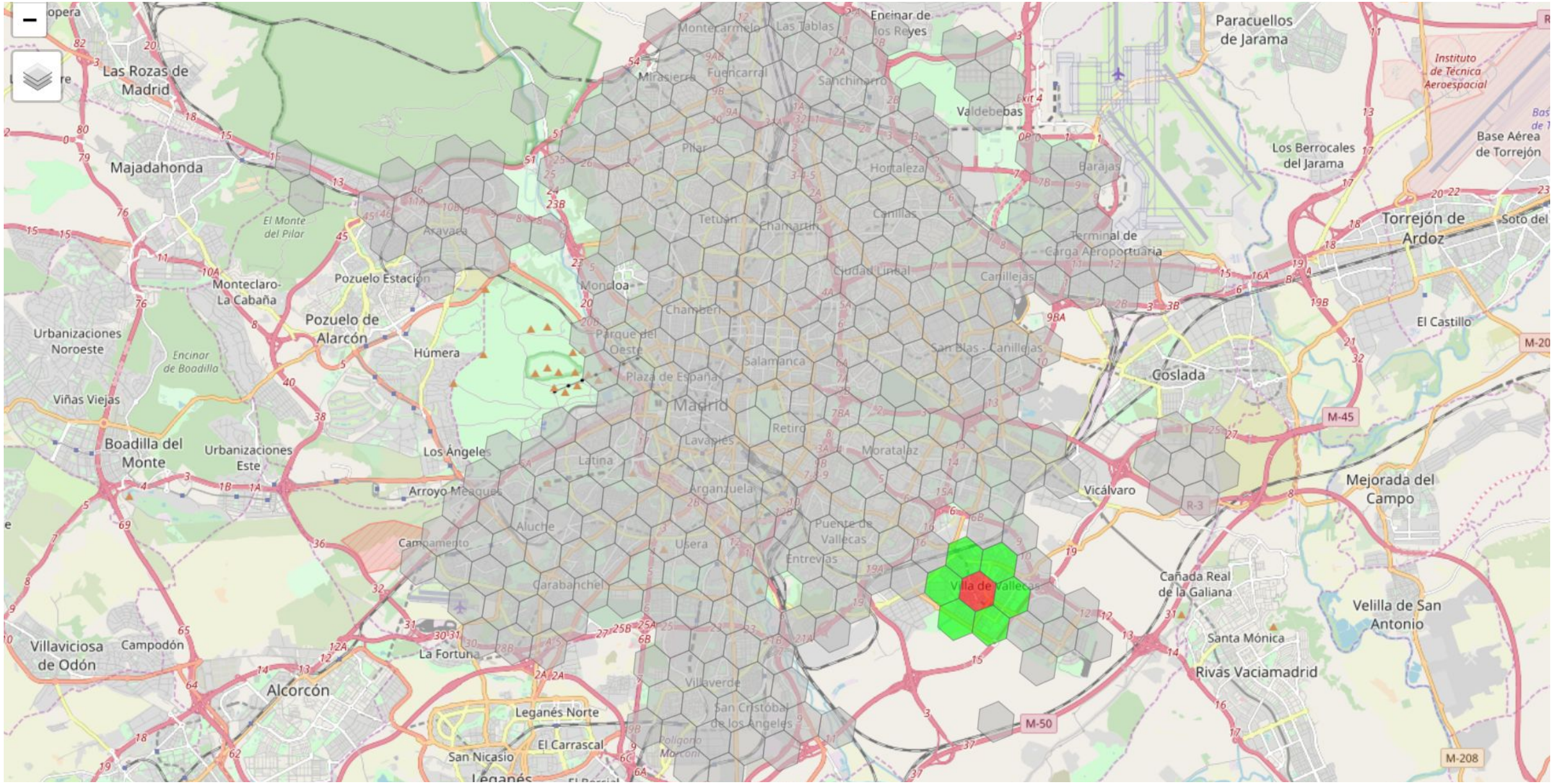
Global Tests: Moran's I:

$$I = \frac{N}{\sum_i \sum_j w_{ij}} \frac{\sum_i \sum_j w_{ij}(X_i - \bar{X})(X_j - \bar{X})}{\sum_i (X_i - \bar{X})^2}$$

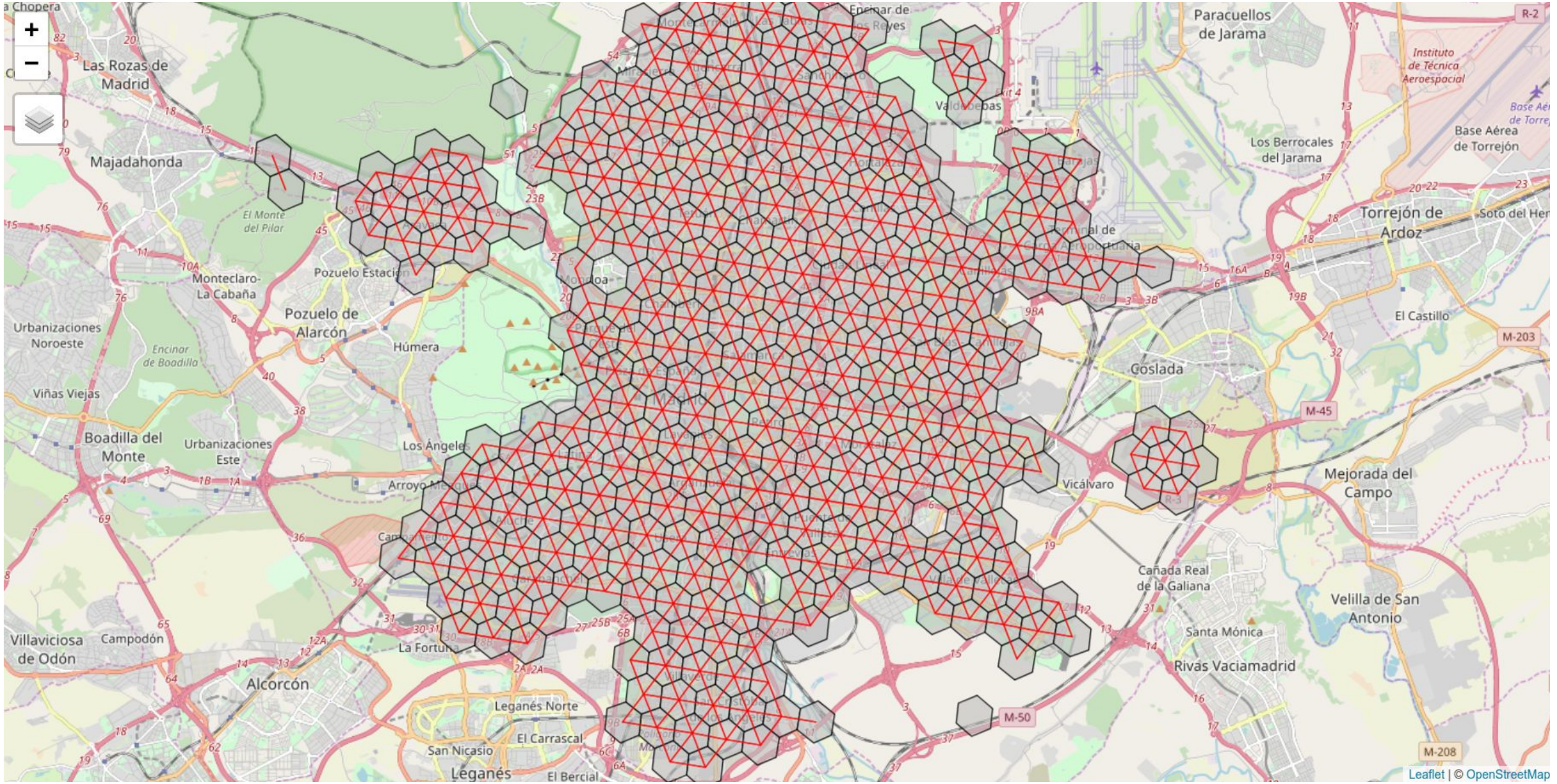
Local Tests: Local Indicators Of Spatial Associations (LISAs):

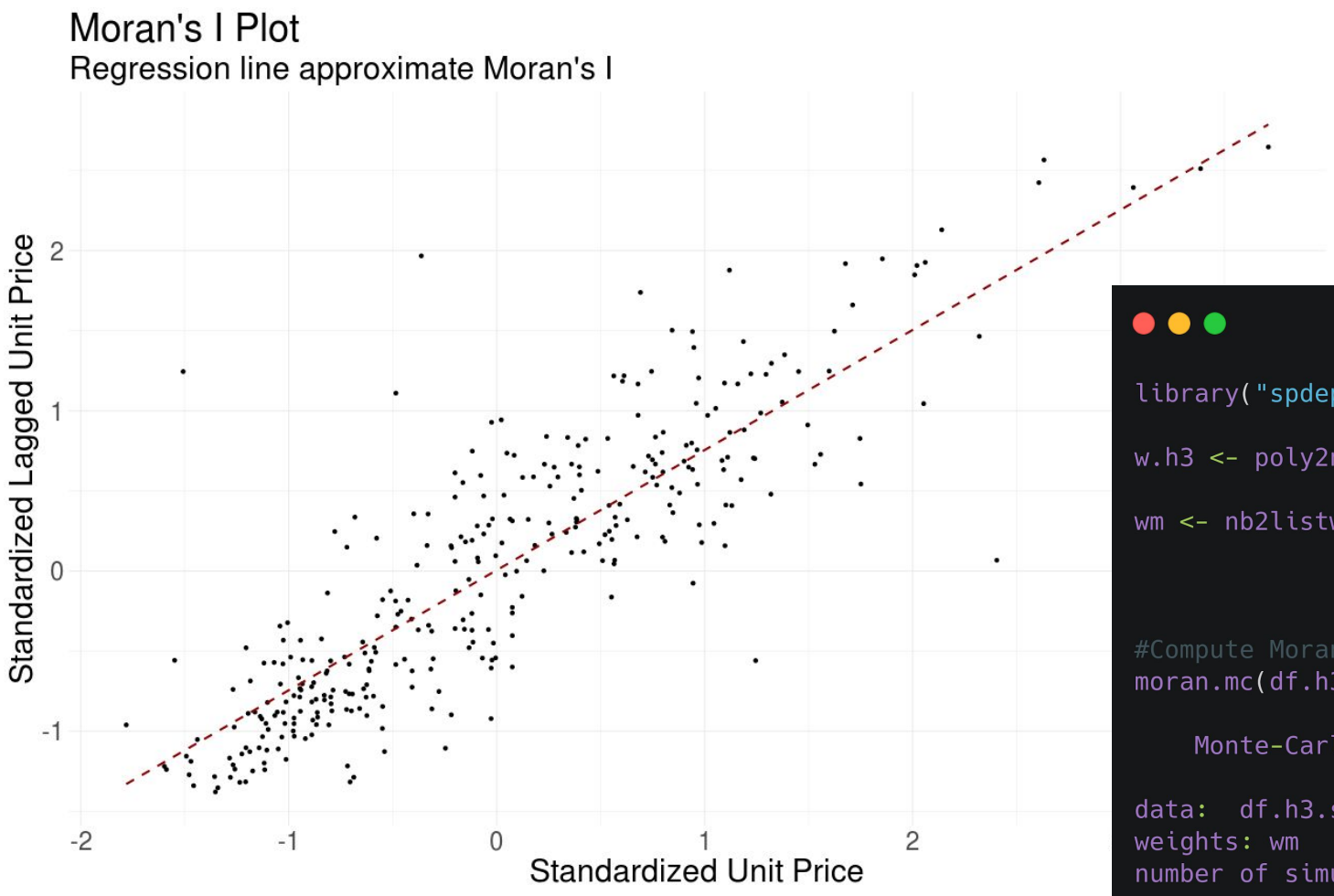
- Local Moran's I
- Getis-Ord G

idealista/data



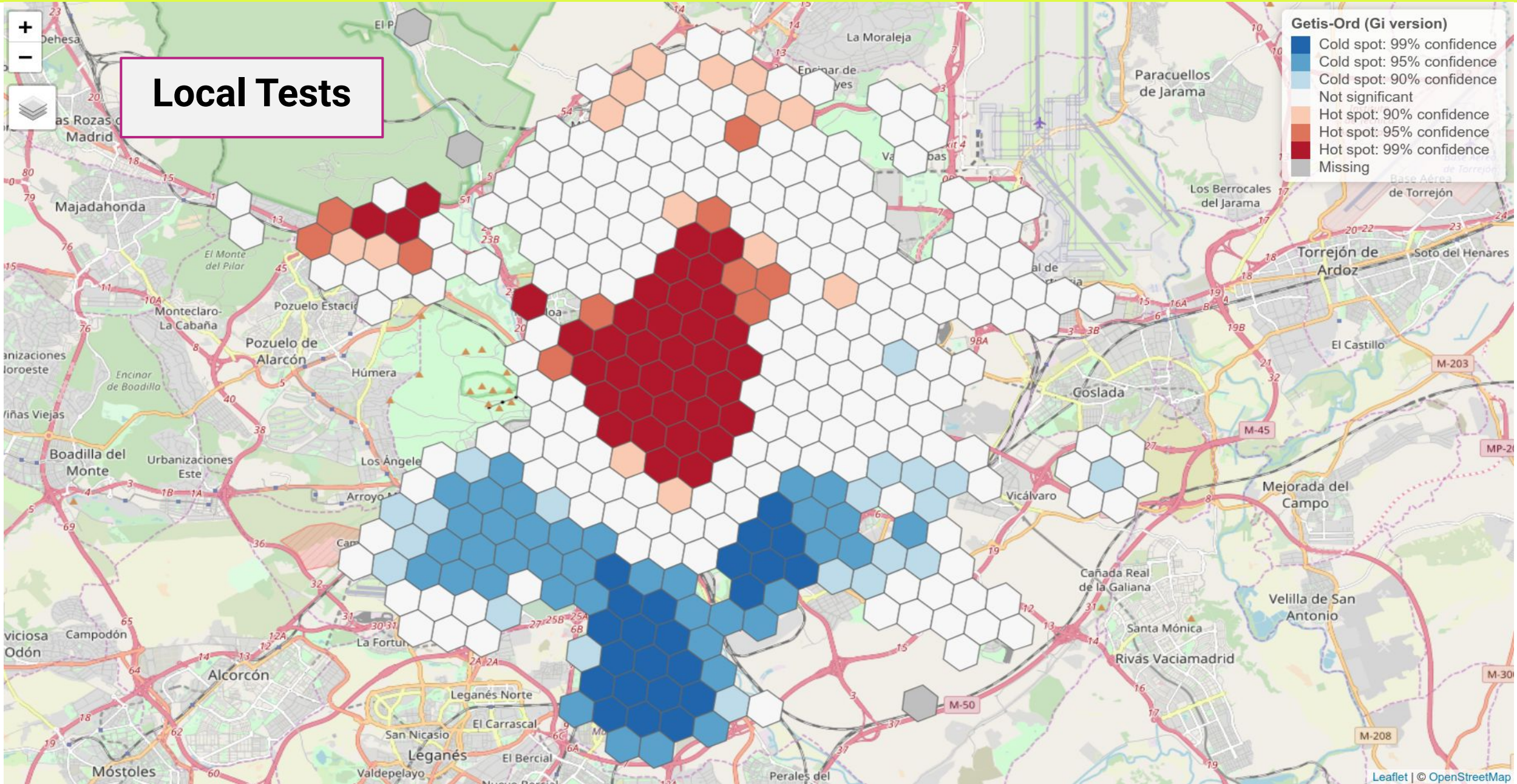
idealista/data



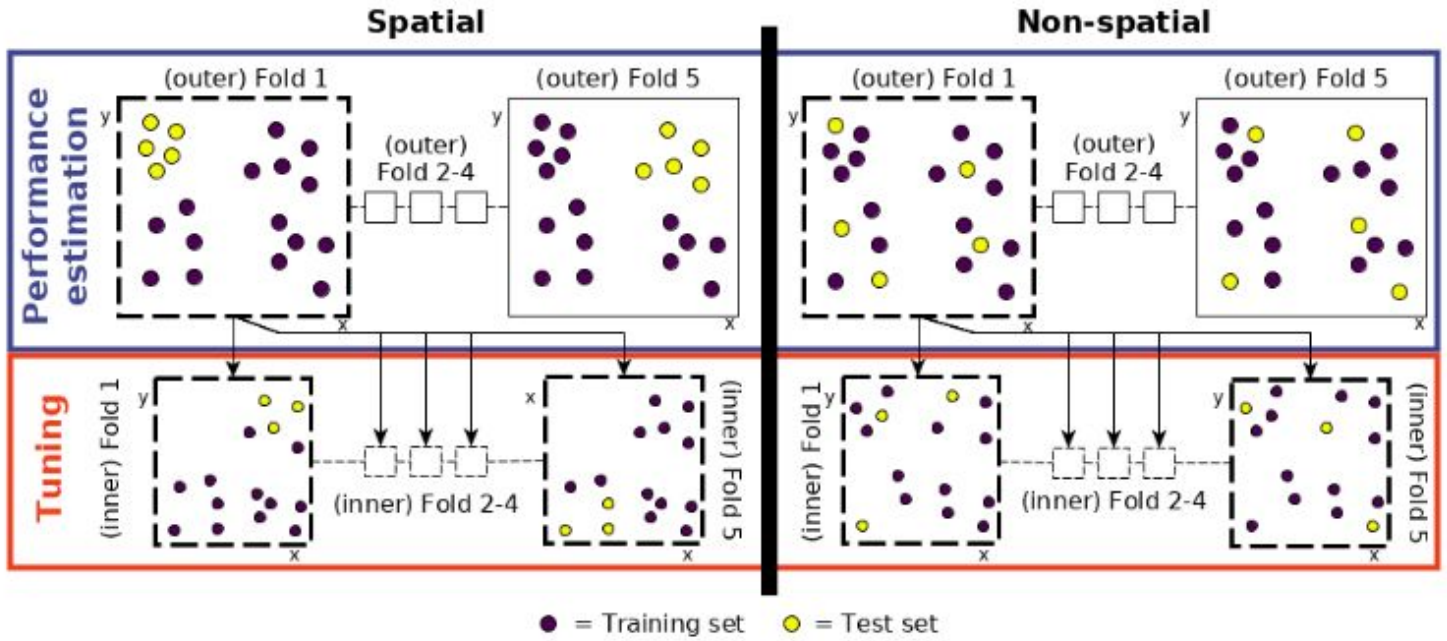


Global Tests

```
library("spdep")

w.h3 <- poly2nb(df.h3.sp,
                row.names=df.h3$h3_index) #Polygon to neighbor
wm <- nb2listw(w.h3,
               style='W',
               zero.policy=TRUE) #Neighbor object to listw

#Compute Moran's I test
moran.mc(df.h3.sp$median.unitprice, wm, nsim=999, zero.policy = T)

    Monte-Carlo simulation of Moran I

data:  df.h3.sp$median.unitprice
weights: wm
number of simulations + 1: 1000

statistic = 0.74251, observed rank = 1000, p-value = 0.001
alternative hypothesis: greater
```


Local Tests
Getis-Ord (Gi version)
Cold spot: 99% confidence
Cold spot: 95% confidence
Cold spot: 90% confidence
Not significant
Hot spot: 90% confidence
Hot spot: 95% confidence
Hot spot: 99% confidence
Missing
Madrid
Majadahonda
Pozuelo de Alarcón
Boadilla del Monte
Alcorcón
Móstoles
Leganés
Leganés Norte
Humera
Goslada
Vicálvaro
Paracuellos de Jarama
Torrejón de Ardoz
Mejorada del Campo
Velilla de San Antonio
Rivas Vaciamadrid
Cañada Real de la Galiana
Santa Mónica
El Castillo
Los Berrocales del Jarama
Soto del Henares
M-50
M-45
M-203
M-208
Leaflet | © OpenStreetMap

Cross Validation under Spatial Autocorrelation




```r
library("mlr")

task <- makeRegrTask(data = data,
                     target = target.var,
                     coordinates = coords)

#spatial cv
spatial_cv <- makeResampleDesc(method = "SpRepCV",
                               folds = 5,
                               reps = 100)

# Fit Spatial CV XgBoost
xgboost_sp_cv_fit <-  mlr::resample(
                          learner = xgboost,
                          task = task,
                          resampling = spatial_cv,
                          measures = mlr::mae)
```

Source: https://geocompr.robinlovelace.net/

References

Arribas-Bel, Daniel (2019): Geographic Data Science Course by @darribasbel: http://darribas.org/gds19/

Uber's H3: https://eng.uber.com/h3/

Spatial Weights Matrix: https://crd230.github.io/lab5.html#spatial_weights_matrix

Geocomputation with R: https://geocompr.robinlovelace.net/

Machine Learning for Spatial Data: http://www.opengeohub.org/machine-learning-spatial-data

Fischer, M. M., & Getis, A. (Eds.). (2010). *Handbook of Applied Spatial Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-03647-7

idealista/data

We  data