“How can I calculate Moran’s I in Stata?”

Authored by
stats writer

July 1, 2024

Moran's I is a statistical measure used to detect spatial autocorrelation in a dataset. It quantifies the similarity of values between neighboring observations and can be calculated using the statistical software Stata. To calculate Moran's I in Stata, one can use the "moran" command, which requires the input of a variable representing the spatial attribute and a weight matrix. The resulting Moran's I value ranges from -1 to 1, where positive values indicate positive spatial autocorrelation and negative values indicate negative spatial autocorrelation. This measure can help researchers understand the spatial patterns and relationships within their data, providing valuable insights for further analysis.

How can I calculate Moran's I in Stata? | Stata FAQ

Note: The commands shown in this page are user-written Stata commands that must be downloaded. To install the package of spatial analysis tools, type search spatgsa in the command window.

Moran's I is a measure of spatial autocorrelation-how related the values of a variable are based on the locations where they were measured. Using a set of user-written Stata commands, we can calculate Moran's I in Stata. We will be using the spatwmat command to generate a matrix of weights based on the locations in our data and the spatgsa command to calculate Moran's I or other spatial

autocorrelation measures.

Let's look at an example. Our dataset, ozone, contains ozone measurements from
thirty-two locations in the Los Angeles area aggregated over one month. The dataset
includes the station number (station), the latitude and longitude of the station
(lat and lon), and the average
of the highest eight hour daily averages (av8top). This data, and other spatial datasets, can be
downloaded from the University of Illinois's Spatial Analysis Lab. We can look
at a summary of our location variables to see the range of locations under
consideration.

use https://stats.idre.ucla.edu/stat/stata/faq/ozone.dta, clear
summarize lat lon

Variable | Obs Mean Std. Dev. Min Max
-------------+--
lat | 32 34.0146 .2228168 33.6275 34.69012

lon | 32 -117.7078 .5683853 -118.5347 -116.2339

Based on the minimum and maximum values of these variables, we can calculate the greatest Euclidean distance we might measure between two points in our dataset.

display sqrt((34.69012 - 33.6275)^2 + (-116.2339 - -118.5347)^2)

2.5343326

Knowing this maximum distance between two points in our data, we can generate a matrix based on the distances between points. In the spatwmat command, we name the weights matrix to be generated, indicate which of our variables are the x- and y-coordinate variables, and provide a range of distance values that are of interest in the band option. All of the distances are of interest in this example, so we create a band with an upper bound greater than our

largest possible distance. If we did not care about distances greater than 2, we could indicate this in the band option.

spatwmat, name(ozoneweights) xcoord(lon) ycoord(lat) band(0 3)

The following matrix has been created:

1. Inverse distance weights matrix ozoneweights
Dimension: 32x32
Distance band: 0

As described in the output, the command above generated a matrix with 32 rows and 32 columns because our data includes 32 locations. Each off-diagonal entry in the matrix is equal to 1/(distance between point i and point j).
Thus, the matrix entries for pairs of points that are close together are higher than for pairs of points that are far apart. If you wish to look at the matrix, you can display it with the matrix list command.

With our matrix
of weights, we can now calculate Moran's I.

spatgsa av8top, weights(ozoneweights) moran

Measures of global spatial autocorrelation

Weights matrix
--
Name: ozoneweights
Type: Distance-based (inverse distance)
Distance band: 0.0

Based on these results, we can reject the null hypothesis that there is zero spatial autocorrelation present in the variable av8top at alpha = .05.

Variations

Binary Matrix: If there exists some threshold distance d such that pairs with distances less than d are neighbors and pairs
with distances greater than d are not, you can create a binary neighbors
matrix with the spatwmat command (indicating bin and setting band

to have an upper bound of d) and use this weights matrix for calculating Moran's I. We could do this for d = 1:

spatwmat, name(ozoneweights) xcoord(lon) ycoord(lat) band(0 1) bin

The following matrix has been created:

1. Distance-based binary weights matrix ozoneweights
Dimension: 32x32
Distance band: 0
spatgsa av8top, weights(ozoneweights) moran

Measures of global spatial autocorrelation

Weights matrix

Name: ozoneweights
Type: Distance-based (binary)
Distance band: 0.0

In this example, the binary formulation of distance yields a similar result.

**We can reject the null hypothesis that there is zero spatial autocorrelation
present in the variable av8top at alpha = .05.**

**Using an existing matrix: If you have calculated a weights matrix
according to some other metric than those available in spatwmat and wish
to use it in calculating Moran's I, spatwmat allows you to read in a
Stata dataset of the required dimensions and format it as a
distance matrix that can be used by spatgsa. If altweights.dta is a
dataset with 32 columns and 32 rows, it could be converted to a weighted matrix
aweights to be used in spatgsa analyzing av8top:**

spatwmat using "C:altweights.dta", name(aweights)