I. RATIONALE FOR FUNCTION

Spatial autocorrelation illustrates how the locations and values of a sample or observation vary throughout space. This form of measurement follows the first law of geography; "Everything is related to everything else, but near things are more related than distant things" (Tobler 1970, p236). One of the measures of spatial autocorrelation is Moran's I; It is based on the cross-products of deviations of observations from their mean and is calculated by accounting for the location of the observations (Karun, Puranik and Binu, 2015). The function has a value range of -1 to 1, with values close to -1 illustrating negative autocorrelation, 0 indicating randomisation and 1 showing positive autocorrelation or clustering. Despite its popular use, Moran's I has certain limitations including the inability to identify and analyse local measures and its features (relationship between clusters). That said, the function created has adopted not only global spatial autocorrelation but also local spatial autocorrelations (both continuous and local indicators of spatial autocorrelations(LISA)) in order to demonstrate a detailed and comprehensive approach to spatial data analysis.

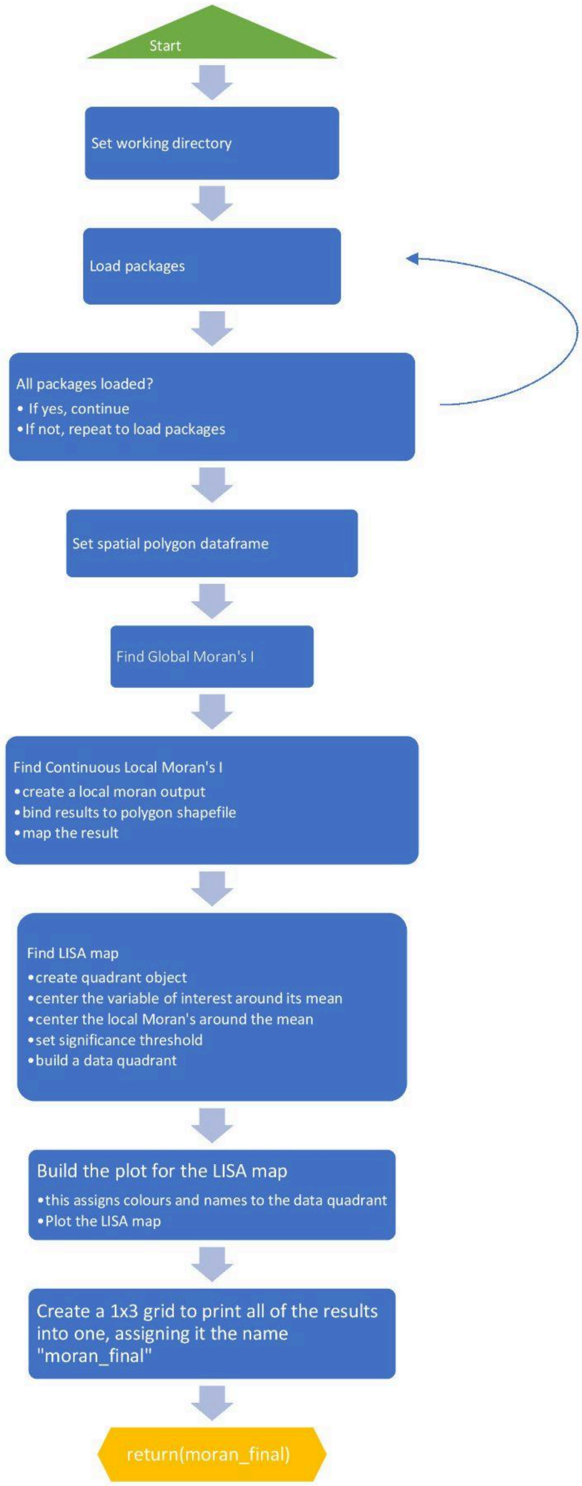
II. DESIGN OF FUNCTION

As previously said, this function will produce three information sets in the following order; a global moran statistic, a continuous local moran map and a LISA map. The following inputs are required for the function to be used efficiently; (1) Spatial data; a type of data that directly or indirectly references a specific geographical area or location (Zola and Fontecchio, 2021) (2) Attribute values; this can be any attributes (they are usually column names in a data frame) the user may want to measure for the Moran's I, (3) type of neighbours; there are several ways to measure neighbours including queen's case, rook's case and even by setting a distance boundary.

FUNCTION

```
myfunction <- function(a, b, c){

neighbours <- c
neighbours_weight <- nb2listw(neighbours, style = "W")

moran_global <- moran.test(b, neighbours_weight)

moran_local <- localmoran(x = b, neighbours_weight)
moran_map <- a
moran_map@data <- cbind(a@data, moran_local)
moran_continuous <- tm_shape(moran_map) + tm_fill(col = "Ii", style = "quantile", title =
"local moran statistic")

quadrant <- vector(mode="numeric",length=nrow(moran_local))
m.variable <- b - mean(b)
m.local <- moran_local[,1] - mean(moran_local[,1])
signif <- 0.1
quadrant[m.variable <0 & m.local>0] <- 1
quadrant[m.variable <0 & m.local<0] <- 2
quadrant[m.variable >0 & m.local<0] <- 3
quadrant[m.variable >0 & m.local>0] <- 4
quadrant[moran_local[,5]>signif] <- 0
brks <- c(0,1,2,3,4)
colors <- c("white","blue",rgb(0,0,1,alpha=0.4),rgb(1,0,0,alpha=0.4),"red")
moran_lisa <-
plot(a,border="lightgray",col=colors[findInterval(quadrant,brks,all.inside=FALSE)])
box()
legend("bottomleft",legend=c("insignificant","low-low","low-high","high-low","high-high"),
    fill=colors,bty="n")


moran_final <- grid.newpage()
pushViewport(viewport(layout=grid.layout(1,3)))

print(moran_global, vp=viewport(layout.pos.col = 1, layout.pos.row =1))
print(moran_continuous, vp=viewport(layout.pos.col = 2, layout.pos.row =1))
print(moran_lisa, vp=viewport(layout.pos.col = 3, layout.pos.row =1))

return(moran_final)
}
```

III. FLOWCHART OF STEPS IN FUNCTION



IV. OUTLINE OF VARIABLES FORMAT, PARAMETERS AND R PACKAGES USED

FUNCTION myfunction(a,b,c)	
Required Variables Variables should be in the format of spatial data in order for the function to calculate the global moran’s I and continuous local moran’s I	
a	Spatial data The example data uses spatial polygons
b	SpatialData$AttributeValues Eg.Southwark$WhiteBritish
c	Type of neighbours - this could be in the following format: ● Queen’s case: poly2nb(a) ● Rook’s case: poly2nb(a, queen = FALSE) ● Specific distance eg. 2500 m: dnearneigh(coordinates(a),0,2500))
Required statistical packages	
‘sp’	Provides classes and methods for spatial data: points, lines, polygons and grids Our current data uses spatial polygons
‘rgdal’	Provides bindings to the 'Geospatial' Data Abstraction Library ('GDAL') (>= 1.11.4) and access to projection/transformation operations from the 'PROJ' library.
‘rgeos’	Implements functionality for the manipulation and querying of spatial geometries using the Geometry Engine — Open Source (GEOS)C library
‘tmap’	Produces thematic maps ● tm_shape() function ● tm_fill()
‘spdep’	Provides spatial autocorrelation functions such as: ● poly2nb() ● nb2listw() ● moran.test() ● localmoran() ● cbind()
‘RColorBrewer’	Provides a number of predefined colour maps
‘grid’	Grid adds an nx by ny rectangular grid to an existing plot
‘gridExtra’	Provides a number of user-level functions to work with "grid" graphics, notably to arrange multiple grid-based plots on a page, and draw tables.
Parameters for customization The user is able to customise a number of variables - especially when mapping the	

continuous local moran	
Type of neighbours Eg. poly2nb(a)	Types to consider: ● Queen's case: poly2nb(a) ● Rook's case: poly2nb(a, queen = FALSE) ● Specific distance eg. 2500 m: dnearneigh(coordinates(a),0,2500))
tm_fill()	● Choosing map colour by adjusting col = "(colour)" ● Setting colour intervals using style = "(type of intervals)" this could be quantile, equal, pretty, etc. ● Naming title using title = "(desired title)"
style = "W"	This is a standardisation parameter; style = "W" asks to standardise the weights. Other standardisation parameters include basic binary coding, "B", globally standardised, "C" or something in between, "S".
'grid.layout'	Specify column and row numbers. The function uses 3 columns and 1-row format.
'Significance threshold'	Sets significance threshold for the LISA map. The function uses a significance threshold of 0.1.
'CRS'	Sets the coordinate system (of a map) into a desired predefined system eg. in the example below, this function uses the British National Grid *(EPSG:27700)* to adjust to the British data that is being used.

V. TESTING THE FUNCTION

Description of the data

The function is demonstrated through an analysis of Southwark practical data - this measures the proportion of people who are White British, have qualifications, are unemployed and living in low occupancy in the area. This data is then merged with the output areas (OA) shapefile of Southwark in order to create a spatial polygon data frame named "Southwark". Moreover, the example will focus on (the proportion of) white British in the area and use the Queen's case as the type of neighbours to analyse the spatial data. In short, this function will analyse whether areas with high or low white British population in Southwark tend to neighbour each other.

The scientific basis and a brief description of the analysis

Before conducting the analysis, it is worth understanding the scientific basis of Moran's I and its local variant. Moran's I captures the relationship between spatial proximity and the variable similarity in a spatial weights matrix and the covariance, respectively (Wu, A.-M., and Kemp, K. K., 2019). The formula of Global Moran's I is written down below (Karun, Puranik and Binu, 2015); this only provides one statistical measure for the whole dataset whereas local Moran's I on the other hand analyses each individual observation. Moreover, local Moran's I can further analyse the data by understanding the relationship of spatial clusters; high-high, low-low, high-low, or low-high and mapping them into the LISA map. The aim of this analysis is to find all the three different moran's I.

Moran's I formula is:

$$I = \frac{n}{S_0} \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} (x_i - \bar{x})(x_j - \bar{x}) w_{ij}}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Testing

The first result of the function shows that the global Moran's I value is 0.6486052509 = 0.65. This indicates that there is a relatively high degree of clustering between areas with white British population. However, this result only shows an overall view of Southwark - there might be more clustering in certain areas of Southwark and less clustering in others. To understand this randomness, we will need to refer to the second result of the function which maps the continuous local Moran's I values.

```
                Moran I test under randomisation

data:  b
weights: neighbours_weight

Moran I statistic standard deviate = 31.886, p-value < 2.2e-16
alternative hypothesis: greater
sample estimates:
Moran I statistic      Expectation         Variance
     0.6486052509    -0.0011210762     0.0004152079
```

Figure 1. Global Moran's I result

The second result maps out the local continuous Moran's I; indicating how the Moran's I value varies for different parts of Southwark. As seen in the map below, the colours indicate 5 different ranges of Moran's I values; with darker colours corresponding to higher positive values - clustering, and lighter colours corresponding to lower to negative values of neighbours. The map shows that areas in the upper right side and middle left have more clustering in comparison to areas on the lower side and upper right-most corner. This confirms how local spatial autocorrelation is able to illustrate and analyse neighbouring data more comprehensively and analytically; despite the global moran's I being 0.55, there are areas that have values higher or lower than that.

However, there is a problem with the result as areas with higher values of Moran's I range from 1.268 to 5.801 - surpassing the Moran's I range of -1 to 1. This problem can be solved by changing the weighting of the data either using row standardisation (which the function has used) or other options including basic binary coding and globally standardised weighting. The user may also want to use a different neighbours type such as Rook's case or a specific distance case.

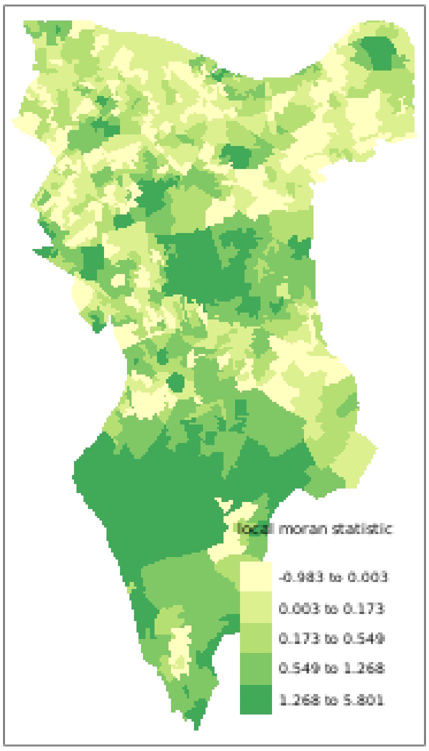


Figure 2. Local Continuous Moran's I

Another problem to consider is that the user is still unable to identify whether these clusters are between areas with a high white British population, low white British or both high and low. This is why the local indicators of spatial association (LISA) map is needed. The LISA map illustrates clusters with high-high, low-low, high-low and low-high white British population - enabling users to identify the actual qualities of the clusterings. For example, the third map shows that most of the clusterings in the lower side of Southwark are between a high proportion of white British whereas clusterings in the middle right of Southwark are among low values of white British.

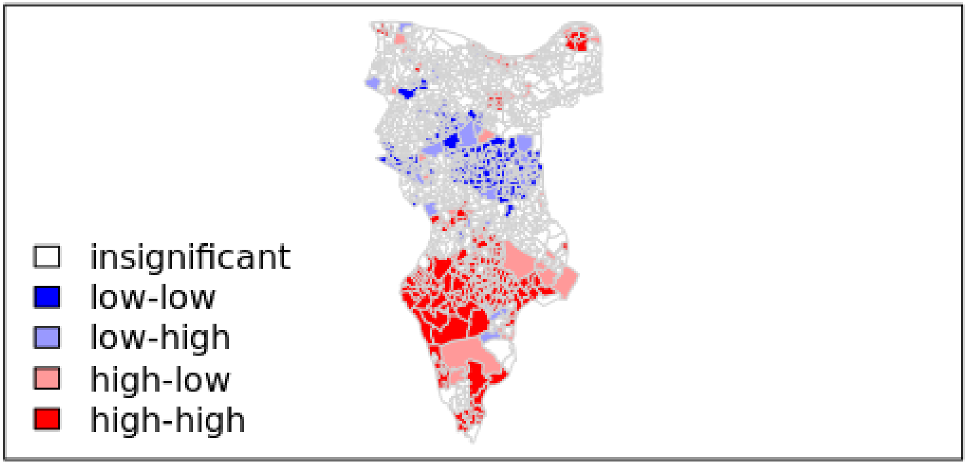


Figure 3. LISA map

Potential applications

The comprehensiveness of this function allows users to apply it with a variety of different datasets - whether it be politics (eg. voters turnout rate), healthcare (eg. COVID-19 vaccinations rate), environment (eg. rate of CO2 being produced), etc., this function is able to identify the global and local spatial autocorrelation measures of a given dataset - all while identifying the relationship between the clusters.

However, there are still limitations to this function such as the absence of spatial autocorrelation throughout time. The function is only able to process data in a particular time frame but real-life data shows that things change over time and this function is unable to illustrate that. The incorporation of spatiotemporal analysis would greatly advance this function to create a more extensive analysis on different datasets. This process has been done by Norou Diwara (2017); he did so by incorporating the Poisson point process.

References

- Diawara, N., Waller, L., King, R. and Lorio, J., 2018. Simulations of local Moran's index in a spatio-temporal setting. *Communications in Statistics - Simulation and Computation*, 48(6), pp.1849-1859.
- Karun, K., Puranik, A., and V.S., B, 2015, '*Easy way to understand the Global measures of Spatial Autocorrelations', Research Journal of Mathematical and Statistical Sciences, Vol. 3(6), 10-14,*
- Tobler W., (1970) "A computer movie simulating urban growth in the Detroit region". *Economic Geography*, 46(2): p236
- Wu, A.-M., and Kemp, K. K. (2019). Global Measures of Spatial Association. *The Geographic Information Science & Technology Body of Knowledge* (1st Quarter 2019 Edition), John P. Wilson (Ed.). DOI: 10.22224/gistbok/2019.1.12
- Zola, A. and Fontecchio, M., 2021. *What is spatial data and how does it work?*. [online] SearchSQLServer. Available at: <https://searchsqlserver.techtarget.com/definition/spatial-data> [Accessed 9 January 2022].