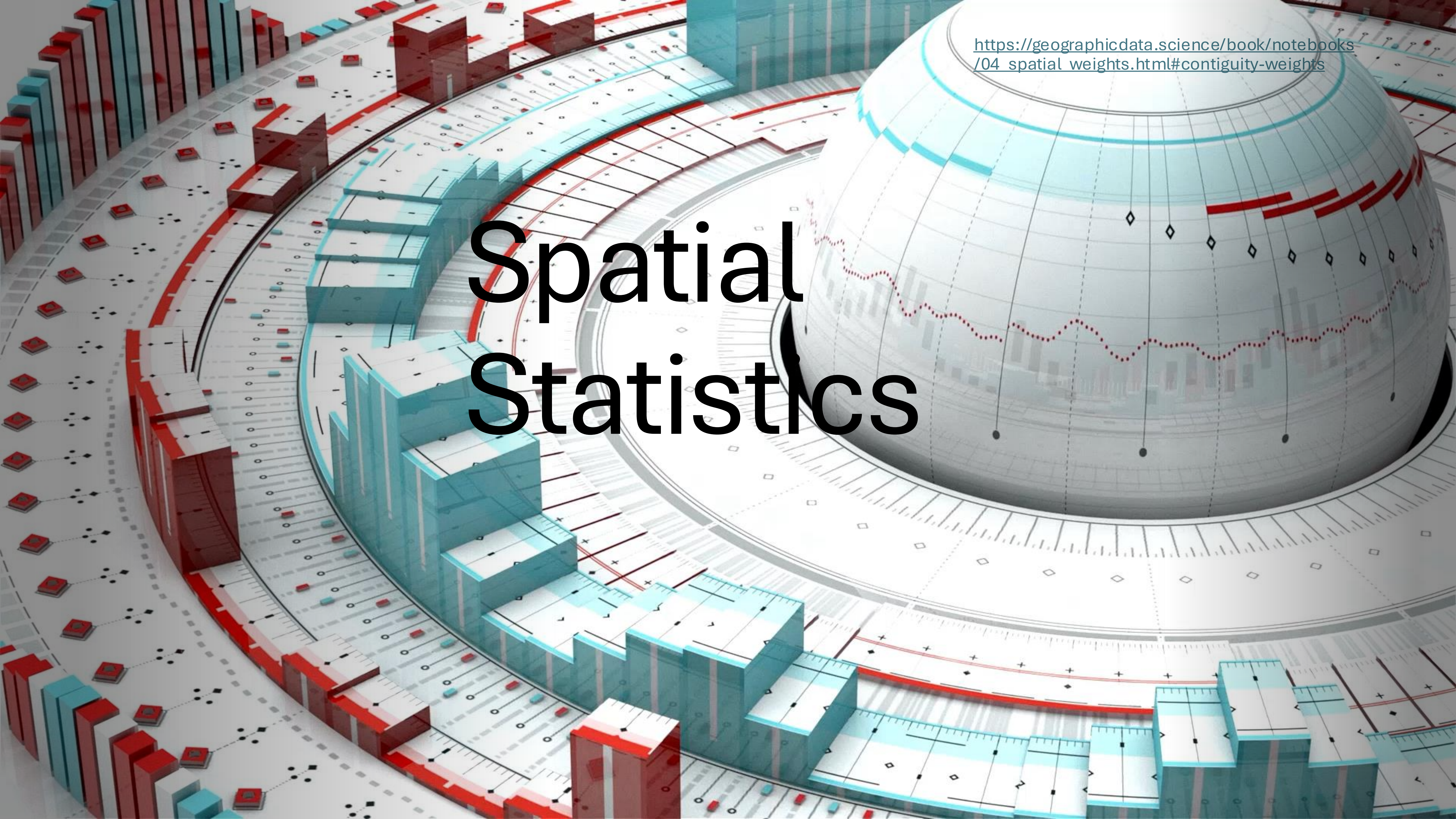


https://geographicdata.science/book/notebooks/04_spatial_weights.html#contiguity-weights

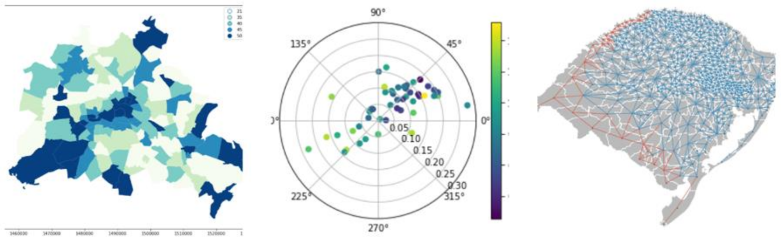
Spatial Statistics

Capturing neighborhoods

- Spatial statistics between geographic tables
  - Often requires expensive computation, e.g. what's nearby?
    - Brute force = Pairwise distance calculation
  - Spatial weight matrices capture "topology" so that computation is cheaper

PySAL: Python Spatial Analysis Library

PySAL is an open source cross-platform library for geospatial data science with an emphasis on geospatial vector data written in Python.



Housing Prices Berlin | Rose diagram (directional LISAs) | Visualizing Non Planar Neighbours

PySAL supports the development of high level applications for spatial analysis, such as

- detection of spatial clusters, hot-spots, and outliers
- construction of graphs from spatial data
- spatial regression and statistical modeling on geographically embedded networks
- spatial econometrics
- exploratory spatio-temporal data analysis

PySAL

PySAL components

- **explore** - modules to conduct exploratory analysis of spatial and spatio-temporal data, including statistical testing on points, networks, and polygonal lattices. Also includes methods for spatial inequality, distributional dynamics, and segregation.
- **viz** - visualize patterns in spatial data to detect clusters, outliers, and hot-spots.
- **model** - model spatial relationships in data with a variety of linear, generalized-linear, generalized-additive, and nonlinear models.
- **lib** - solve a wide variety of computational geometry problems:
  - graph construction from polygonal lattices, lines, and points.
  - construction and interactive editing of spatial weights matrices & graphs
  - computation of alpha shapes, spatial indices, and spatial-topological relationships
  - reading and writing of sparse graph data, as well as pure python readers of spatial vector data.

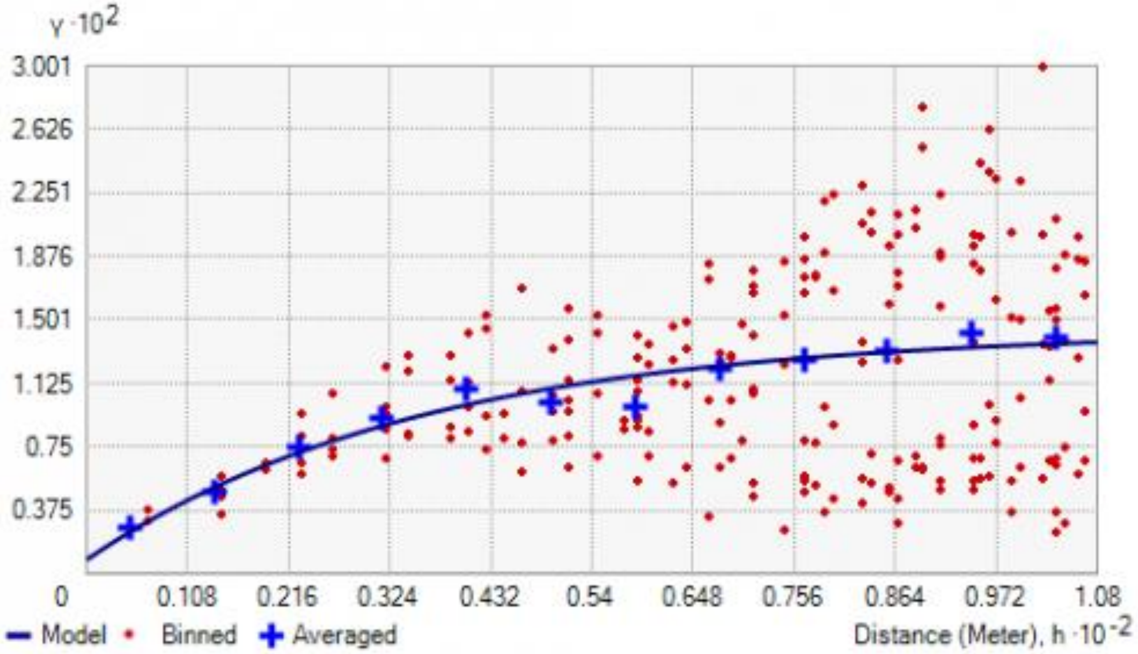
Tobler's First Law of Geography:

"Everything is related to everything else, but near things are more related than distant things."

Tobler, W. (1970) "A computer movie simulating urban growth in the Detroit region". Economic Geography, 46(2): 234-240.

Examples

Economic activity for shops nearby

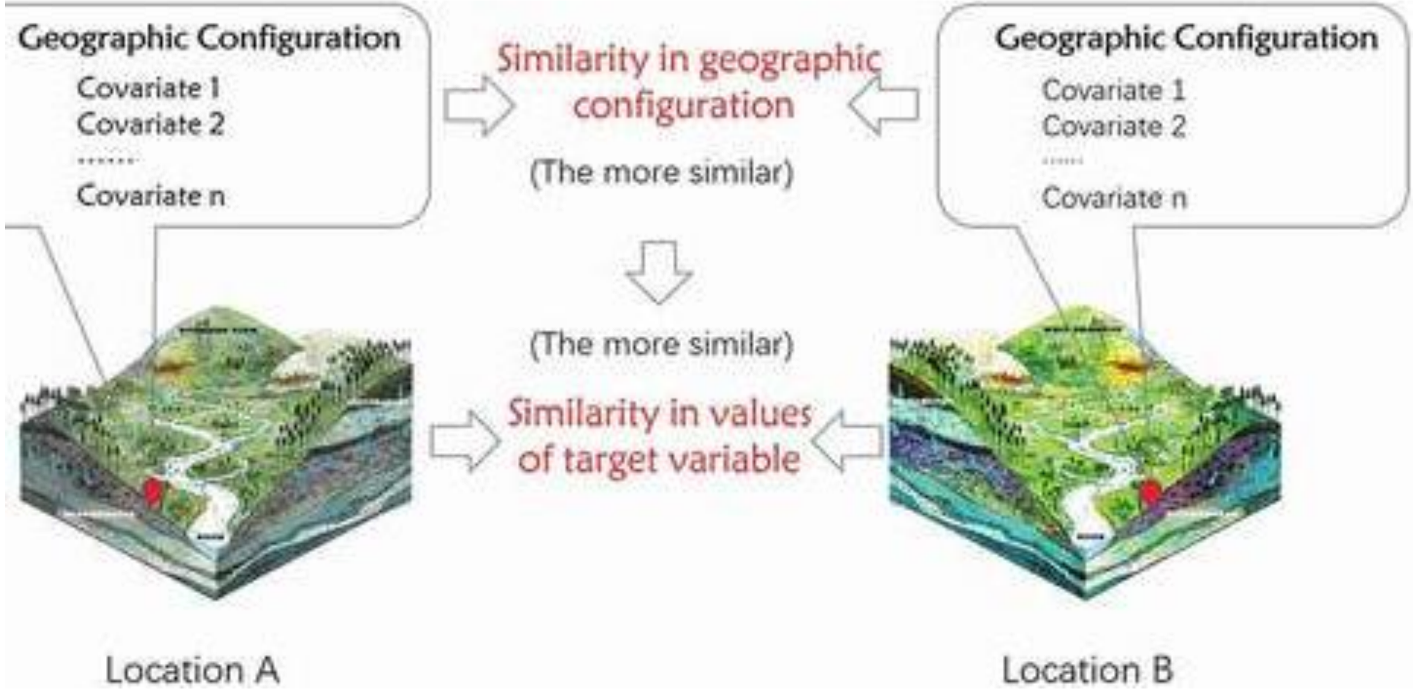


Terrain variability from a central point

The second law of geography

the phenomenon external to a geographic area of interest affects what goes on inside.

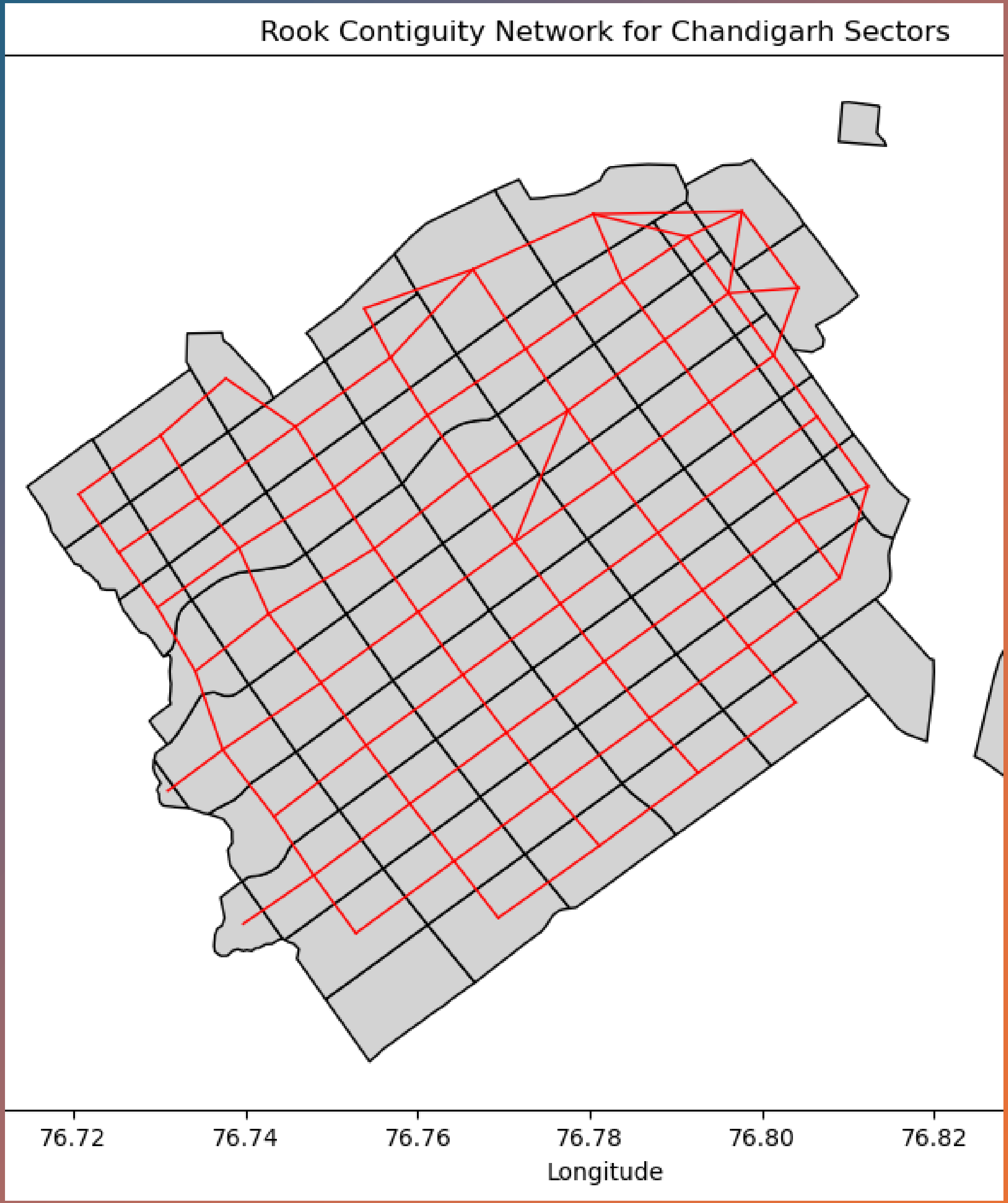
Third Law of Geography



How to weigh your neighbors?

- Continuity/Adjacency relations
- Distance based relations
- Hybrid weights

Continuity weights

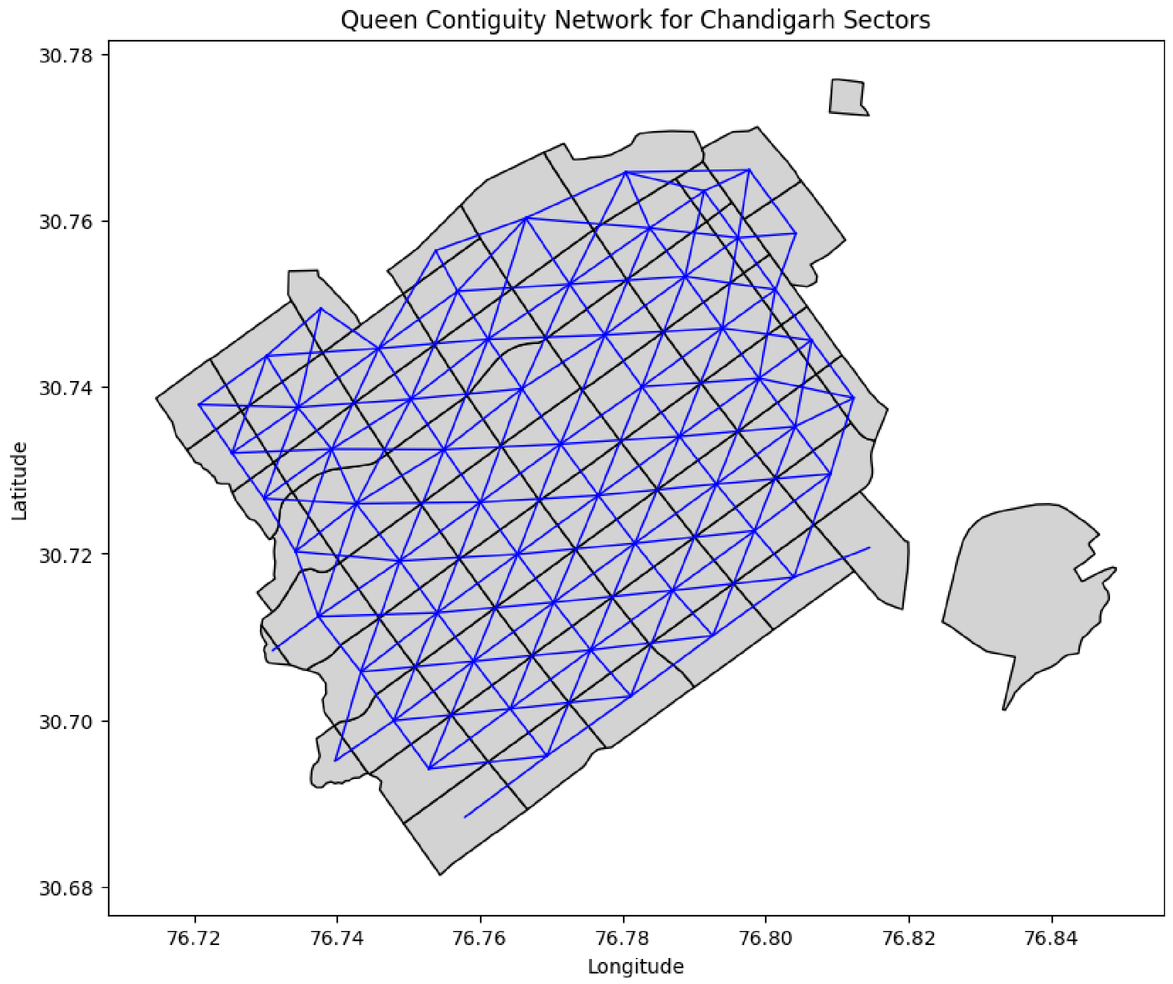


Rook Neighbors

- 4 islands
- Connections amongst sectors that share an edge
- See code example on notebook.
- Note, weights.full() gives matrix representation
- Useful for modeling more rigid phenomenon, e.g.,
  - Roads, zoning, water flows, etc.

Queen Contiguity

- Neighbors share an edge or a vertex
  - Diagonal movement is allowed
- Allows for a more flexible definition of neighborhood
  - Useful for modeling more ‘free’ phenomenon, e.g., trade, ecology, etc.



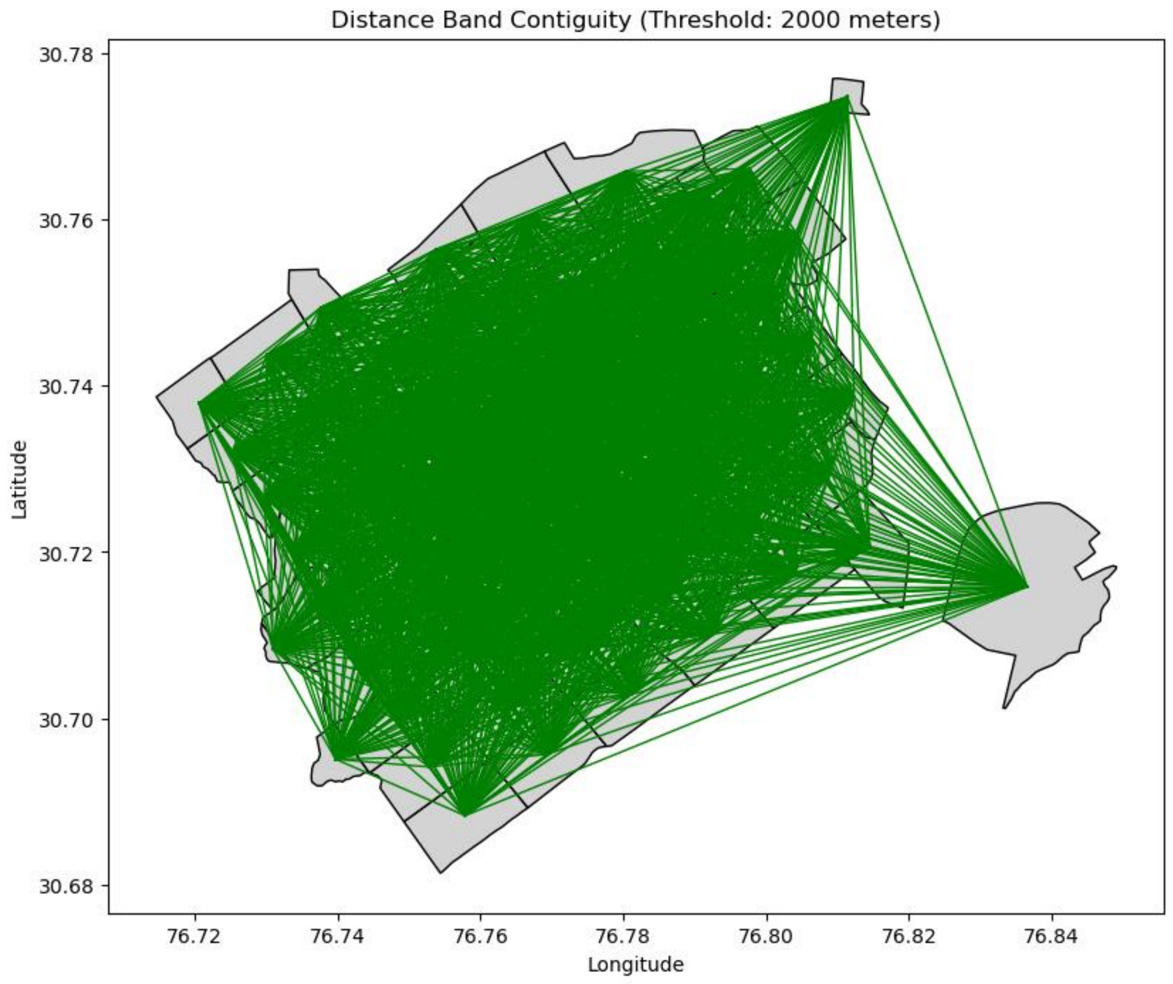
A note on islands

- Islands can create a problem in spatial statistics algorithms. It is thus preferred to add a single neighbor at least for further analysis.
  - Can be added by using sets (covered later) or
  - Can be added through manual editing of contiguity dictionary

Distance based relations

Define neighbors through a distance threshold

- Take all neighbors within a threshold distance
    - For example, blinkit deliveries
- Or just a few nearest neighbors,
    - E.g. kNN neighbors
    - Uses inter-centroid distances
- Shall we just use kernels?
    - Shape, e.g., gaussian, triangular,
    - Bandwidth, e.g., 500m. After this the weights are decayed.



Different kernel functions

$$z_{i,j} = d_{i,j}/h_i$$

triangular

$$K(z) = (1 - |z|)\ if |z| \leq 1$$

uniform

$$K(z) = 1/2\ if |z| \leq 1$$

quadratic

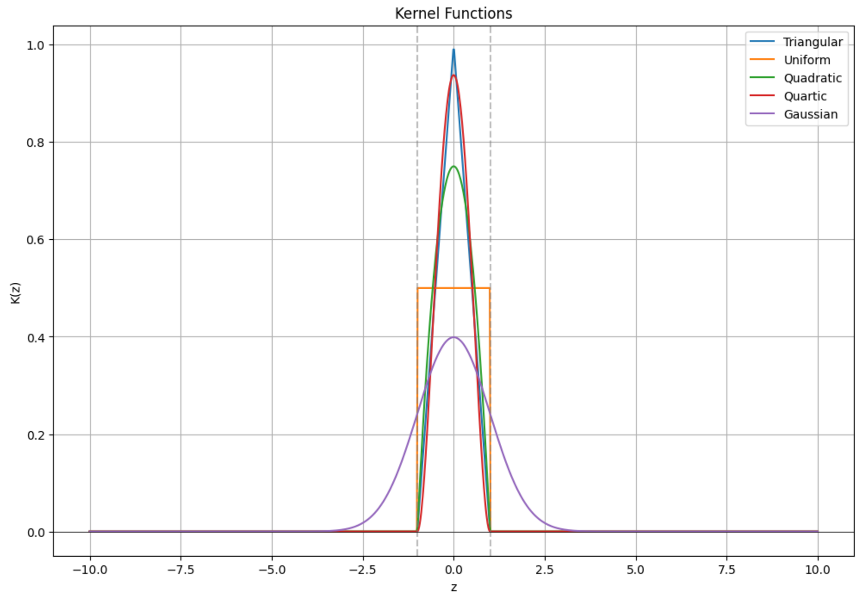
$$K(z) = (3/4)(1 - z^2)\ if |z| \leq 1$$

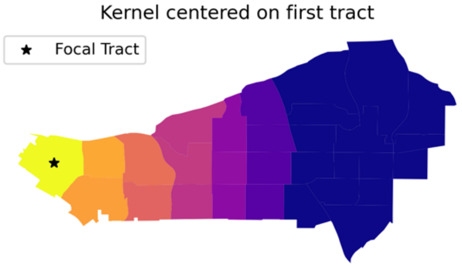
quartic

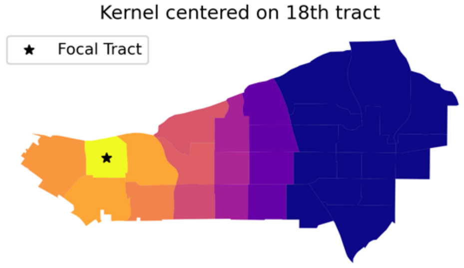
$$K(z) = (15/16)(1 - z^2)^2\ if |z| \leq 1$$

gaussian

$$K(z) = (2\pi)^{(-1/2)} exp(-z^2/2)$$







Summary

Spatial Weights Type	**Best When...**	**Pros**	**Cons**
Rook Contiguity	Influence spreads via **shared borders** (e.g., political units)	Simple, interpretable	Ignores diagonal connections
Queen Contiguity	Influence spreads via **borders & corners**	More inclusive than Rook	Still ignores distance
Distance Band	You have a **fixed range of influence** (e.g., air pollution, traffic impact)	Ensures all close units are neighbors	Sharp cutoff—no gradual influence
Inverse Distance	Influence **decays with distance** (e.g., economic interactions)	Accounts for gradual influence loss	Needs careful choice of `power`
K-Nearest Neighbors (KNN)	You need **a fixed number of neighbors per unit**	Ensures every unit has neighbors	No distance-based weighting
Kernel Weights	Influence spreads **smoothly over distance**	Most flexible, smooth decay	Harder to interpret

Hybrid

Hybrid weights

- Decaying distance within a threshold
  - w_bdb = weights.distance.DistanceBand.from_dataframe(gdf, 1.5, binary=True)
  - DistanceBand uses inverse distance relationship

Incorporating earth’s curvature

```python
# ignore curvature of the earth
knn4_bad = weights.distance.KNN.from_shapefile(
    "../data/texas/texas.shp", k=4
)
```

```python
radius = geometry.sphere.RADIUS_EARTH_MILES
radius
```

```python
knn4 = weights.distance.KNN.from_shapefile(
    "../data/texas/texas.shp", k=4, radius=radius
)
```

Block Weights

Block weights

- Use a list as proxy for near-ness, e.g., all sectors with a waste treatment plant

Combining block and distance?

- Define neighbors as *all schools in a ward within a distance of 2 km*
  - Use **Sets**
    - Sets allow you to do set operations - union, intersection, etc. of neighbors from different weight matrices

```
w_fixed_sets = weights.set_operations.w_union(w_rook, wk1)
```

In Research

Jia, Yuhao, Zile Wu, Shengao Yi, and Yifei Sun. "GeoTransformer: Enhancing Urban Forecasting with Geospatial Attention Mechanisms." *arXiv preprint arXiv:2408.08852* (2024).

- https://arxiv.org/abs/2408.08852

Zhu, Di, Yu Liu, Xin Yao, and Manfred M. Fischer. "Spatial regression graph convolutional neural networks: A deep learning paradigm for spatial multivariate distributions." *GeoInformatica* 26, no. 4 (2022): 645-676.

https://link.springer.com/article/10.1007/s10707-021-00454-x

Liu, Pengyuan, and Filip Biljecki. "A review of spatially-explicit GeoAI applications in Urban Geography." *International Journal of Applied Earth Observation and Geoinformation* 112 (2022): 102936.

https://www.sciencedirect.com/science/article/pii/S1569843222001339

Zhao, Tianhong, Xiucheng Liang, Wei Tu, Zhengdong Huang, and Filip Biljecki. "Sensing urban soundscapes from street view imagery." *Computers, Environment and Urban Systems* 99 (2023): 101915.