spgen:

Creating spatially lagged variables in Stata

Keisuke Kondo[a, b]

[a] Research Institute of Economy, Trade and Industry

[b] Research Institute for Economics and Business Administration, Kobe University

July 20, 2023

2023 Stata Conference Stanford

RIETI Research Institute of Economy, Trade & Industry, IAA

KOBE UNIVERSITY 神戸大学

Outline

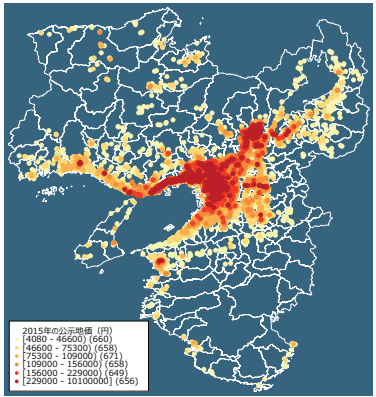
1. What is the spatial data?
2. How is the spgen command useful?
3. Application of the spgen command
4. Concluding remarks

Scan QR code to download this PDF file:

URL: https://keisukekondokk.github.io/software/doc/spgen_slide_stata_conference_2023.pdf

1. What is the spatial data?

- Spatial data analysis has gained attention from policy-makers, researchers, and data scientists.
- Spatial data is...(*)
  1. [data structures] Information about the locations and shapes of geographic features and the relationships between them, usually stored as coordinates and topology.
  2. [data models] Any data that can be mapped.
- Demand for spatial data analysis is continuously growing among Stata users.



Note: Author's creation based on 2015 land price data in the Osaka metropolitan area (MLIT, Japan)

* ESRI, spatial data, GIS Dictionary, `https://support.esri.com/en-us/gis-dictionary/spatial-data` (accessed July 13, 2023)

2. How is the `spgen` command useful?

In which situation is the `spgen` command useful?

☞ Examples of research questions:

- How do the neighboring regions affect one another?
- Whether densely populated regions have a higher risk of COVID-19 infection?
- How does local market size affect stores' sales?
- How does local market size affect firms' entry/exit decision?
- How many potential customers are there around stores?
- How many rival stores are there around stores?

2. How is the `spgen` command useful?

✣ Sp commands from Stata 15 ✣

- The Sp commands manage data and fit regressions accounting for spatial relationships (StataCorp, 2023).
- Spatial econometrics is officially supported since June 2017
- The `spgen` command provides similar functions to the `spgenerate` command, which is provided in the Sp commands.
- I would like to explain the motivation why I developed the `spgen` command and how different from the `spgenerate` command.

Note: The `spgen` command was firstly released in 2015 and is not the short abbreviation of the `spgenerate` command.

2. How is the spgen command useful?

☞ Comparison between spgenerate and spgen commands

spgenerate

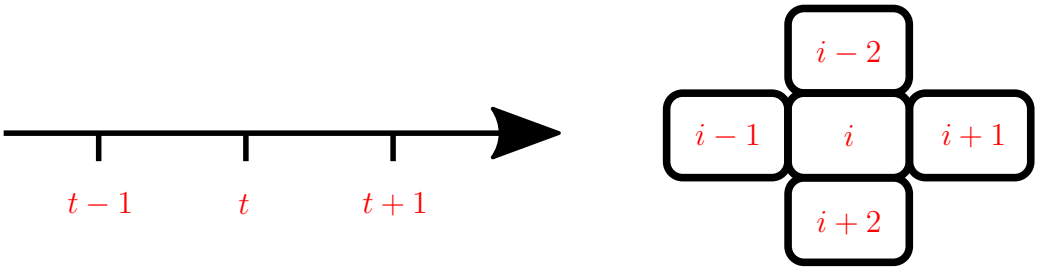
- It calculate the spatially lagged variable.
- It is specialized for the Sp commands (spatial econometrics).
- Users must prepare spatial weight matrix beforehand.
- It is infeasible for high dimensional spatial weight matrices. High-spec computer is necessary.

spgen

- It calculate the spatially lagged variable.
- It provides more flexible functions for spatial analysis.
- Users do not prepare spatial weight matrix beforehand.
- It is feasible for high dimensional spatial weight matrices. It works on low-spec computer.

2. How is the `spgen` command useful?

- **Spatial lag** captures the inter-relationship between an observation and its neighboring observations



(a) Time (b) Space

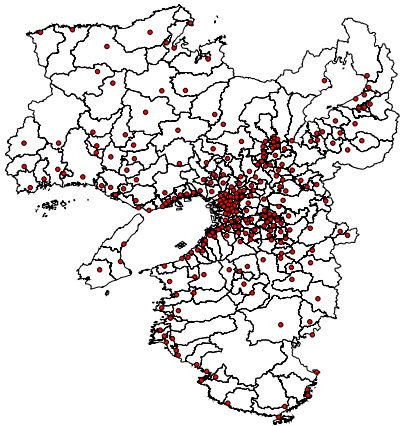
Fig: Lag in Time and Space

Source: Author's creation

2. How is the spgen command useful?

How are neighbors defined mathematically?

- There are R municipalities.
- Each municipality has one base location (red point).
- The distance is measured between location i and any other locations.
- Based on the bilateral distance between municipalities i and j, we define the neighbors of each municipality i.



Note: Author's creation.

2. How is the `spgen` command useful?

- The distance matrix $\boldsymbol{D}$ is obtained from the location information.
- The `spgen` command considers the four types of spatial weight matrix:

1. The power function: $d_{ij}^{-\delta}$
2. The exponential function: $\exp(-\delta d_{ij})$
3. The indicator function (threshold distance): $I(d_{ij} < d)$
4. The indicator function (k-nearest neighbor): $I(d_{ij} < d_{ij,(k)})$

$$\boldsymbol{D} = \begin{pmatrix} d_{11} & d_{12} & \cdots & d_{1R} \\ d_{21} & d_{22} & \cdots & d_{2R} \\ \vdots & \vdots & \ddots & \vdots \\ d_{R1} & d_{R2} & \cdots & d_{RR} \end{pmatrix}$$

2. How is the `spgen` command useful?

- Spatial weight matrix $\boldsymbol{W}$ is interpreted as the **weighted sum** or **local sum** operators.
- The diagonal elements w_{ii} take the value of 0 in the `spgen` command.
- Spatial weight matrix is generally row-standardized in the spatial weight matrix.
- Spatial weight matrix is not row-standardized for the market potential (Harris, 1954).
- The spatial lag of variable $\boldsymbol{x}$ is defined as $\boldsymbol{W}\boldsymbol{x}$.

$$\boldsymbol{W} = \begin{pmatrix} 0 & w_{12} & \cdots & w_{1R} \\ w_{21} & 0 & \cdots & w_{2R} \\ \vdots & \vdots & \ddots & \vdots \\ w_{R1} & w_{R2} & \cdots & 0 \end{pmatrix}$$

2. How is the spgen command useful?

spgen

```
spgen varlist, lat(varname) lon(varname) swm(swmtype) dist(#) dunit(km|mi)
```

Latitude | Longitude | SWM type | Dist. threshold | Dist. unit

```
    [optional settings]
```

☞ The spatial lag of `varlist` is stored in the dataset.

Note: See the manual for details: `https://github.com/keisukekondokk/spgen`

2. How is the `spgen` command useful?

☞ Row-standardized matrix returns the weighted sum of neighbors.

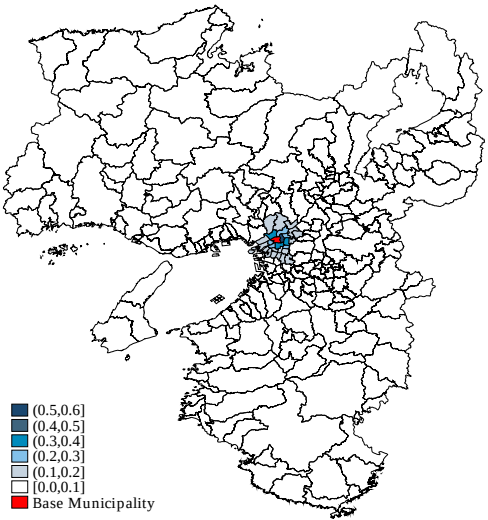
```
. spgen x, lon(_CX) lat(_CY) swm(pow 1) dist(.) dunit(km)
```

$$\boldsymbol{W}\boldsymbol{x} = \begin{pmatrix} 0 & w_{12} & \cdots & w_{1R} \\ w_{21} & 0 & \cdots & w_{2R} \\ \vdots & \vdots & \ddots & \vdots \\ w_{R1} & w_{R2} & \cdots & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_R \end{pmatrix} = \begin{pmatrix} \sum_{k=1}^{R} w_{1k}x_k \\ \sum_{k=1}^{R} w_{2k}x_k \\ \vdots \\ \sum_{k=1}^{R} w_{Rk}x_k \end{pmatrix}$$

Note: Spatial weight matrix $\boldsymbol{W}$ is often row-standardized (row-sum is equal to one) in the spatial econometrics.

2. How is the `spgen` command useful?

☞ Weights of Neighboring Municipalities



Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

2. How is the `spgen` command useful?

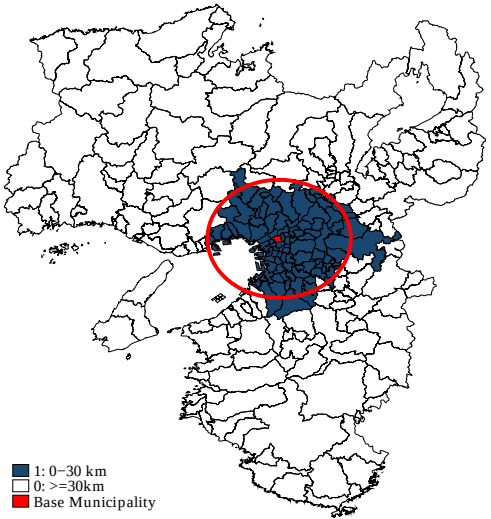
☞ Local sum operator with indicator elements (row is not standardized)

```
. spgen x, lon(_CX) lat(_CY) swm(bin) dist(30) dunit(km) nostd
```

$$\boldsymbol{W}\boldsymbol{x} = \begin{pmatrix} 0 & I(d_{ij} < 30) & \cdots & I(d_{ij} < 30) \\ I(d_{ij} < 30) & 0 & \cdots & I(d_{ij} < 30) \\ \vdots & \vdots & \ddots & \vdots \\ I(d_{ij} < 30) & I(d_{ij} < 30) & \cdots & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_R \end{pmatrix} = \begin{pmatrix} \sum_{k=1}^{R} I(d_{1k} < 30) x_k \\ \sum_{k=1}^{R} I(d_{2k} < 30) x_k \\ \vdots \\ \sum_{k=1}^{R} I(d_{Rk} < 30) x_k \end{pmatrix}$$

2. How is the `spgen` command useful?

☞ Indicator (1/0) of neighboring municipalities within 30 km

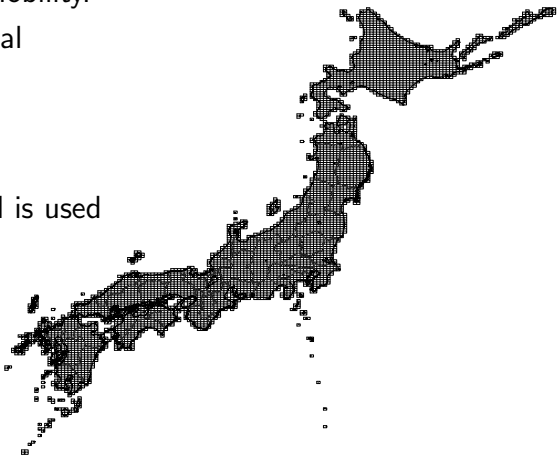


Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the `spgen` command

Research Question: Retail Sales and Local Markets Range

- Retail sales are constrained by human mobility.
- It is important to know the effect of local market range on the retail productivity.
- Our simple analysis attempts to identify geographical range of local markets.
- Mesh data at the 1km-by-1km grid level is used (391,451 obs.).

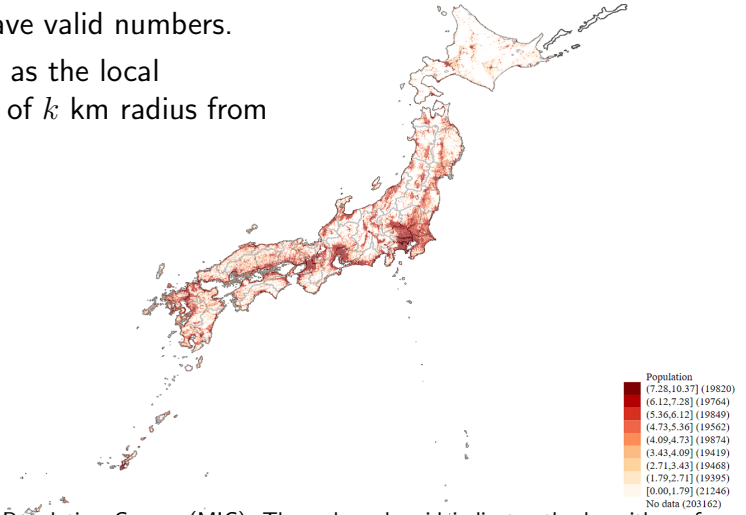


Source: Author's creation based on the shapefiles of mesh data obtained from the e-Stat (Statistical Bureau, MIC).

3. Application of the `spgen` command

Mesh Data: Population

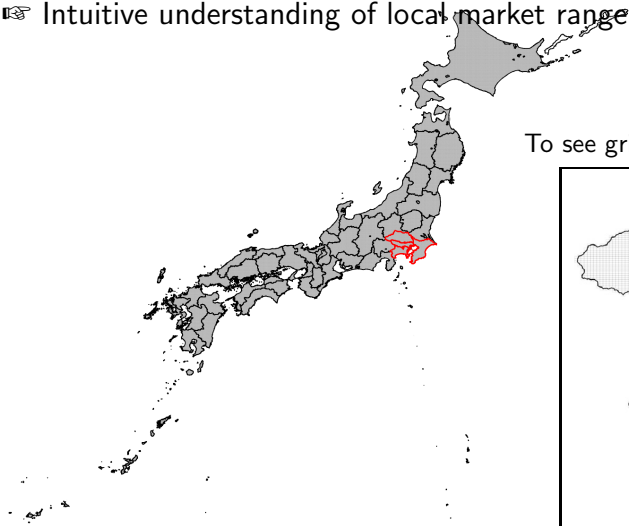
- 178,397 of 391,451 grids have valid numbers.
- Local market size is defined as the local population within the circle of k km radius from each mesh grid.



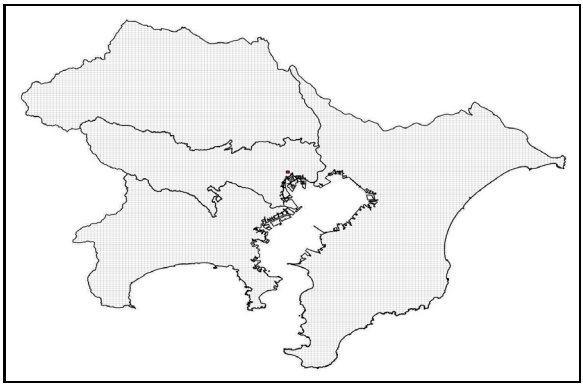
Source: Author's creation based on the 2015 Population Census (MIC). The red mesh grid indicates the logarithm of population.

3. Application of the `spgen` command

☞ Intuitive understanding of local market range



To see grid lines, we zoom in Tokyo metropolitan area.

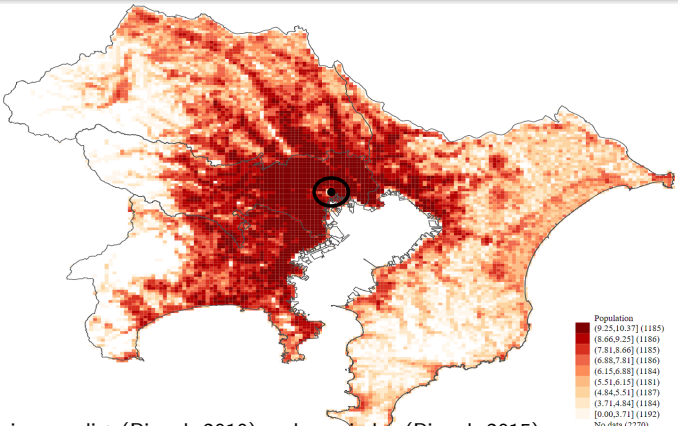


Source: Author's creation based on the shapefiles of mesh data obtained from the e-Stat (Statistical Bureau, MIC).

3. Application of the spgen command

☞ Local Market Range 0–5 km from each mesh grid

```
. gen rowif_sales_per_area = (sales_per_area > 0 & sales_per_area = .)
. spgen pop_total_all, lon(lon) lat(lat) swm(bin) nostd dist(5) dunit(km) rowif(rowif_sales_per_area)
largesize replace
```



Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the spgen command

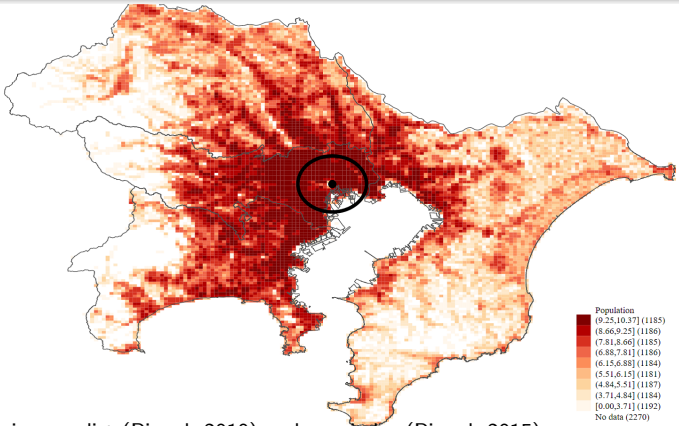
```
.  spgen pop_total_all, ///
>    lon(lon) lat(lat) swm(bin) nostd dist(5) dunit(km) rowif(rowif_sales_per_area) largesize replace
.  ROWIF option returns spatial lags for observations with rowif_sales_per_area = 1
.  Size of spatial weight matrix:  38334 * 381559
.  Calculating spatial lagged variable...
.  -----------------
.  |Completed:  10%|
.  |Completed:  20%|
.  |Completed:  30%|
.  |Completed:  40%|
.  |Completed:  50%|
.  |Completed:  60%|
.  |Completed:  70%|
.  |Completed:  80%|
.  |Completed:  90%|
.  |Completed:  100%|
.  -----------------
.  splag1_pop_total_all_b was generated in the dataset.
```

Notes: 39.28 minutes (OS: Windows 10 Pro 64bit, CPU: Intel Xeon W-2133, Memory: 64GB)

3. Application of the spgen command

☞ Local Market Range 0–10 km from each mesh grid

```
. gen rowif_sales_per_area = (sales_per_area > 0 & sales_per_area = .)
. spgen pop_total_all, lon(lon) lat(lat) swm(bin) nostd dist(10) dunit(km)
rowif(rowif_sales_per_area) largesize replace
```

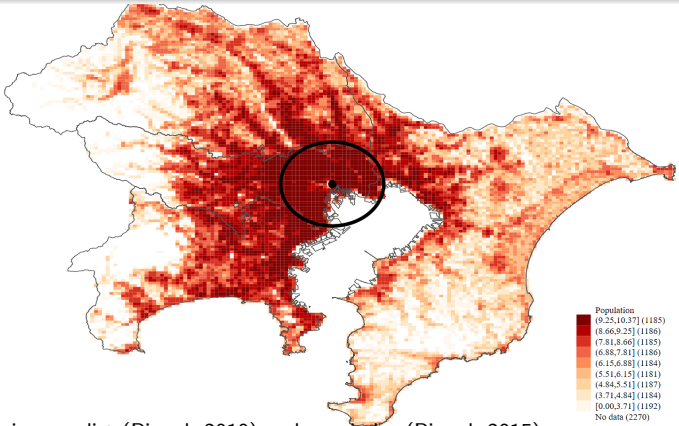


Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the spgen command

☞ Local Market Range 0–15 km from each mesh grid

```
.  gen rowif_sales_per_area = (sales_per_area > 0 & sales_per_area = .)
.  spgen pop_total_all, lon(lon) lat(lat) swm(bin) nostd dist(15) dunit(km)
rowif(rowif_sales_per_area) largesize replace
```

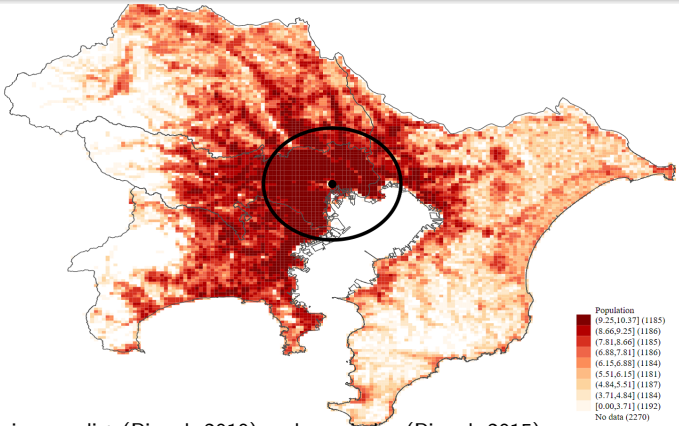


Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the spgen command

☞ Local Market Range 0–20 km from each mesh grid

```
. gen rowif_sales_per_area = (sales_per_area > 0 & sales_per_area = .)
. spgen pop_total_all, lon(lon) lat(lat) swm(bin) nostd dist(20) dunit(km)
rowif(rowif_sales_per_area) largesize replace
```

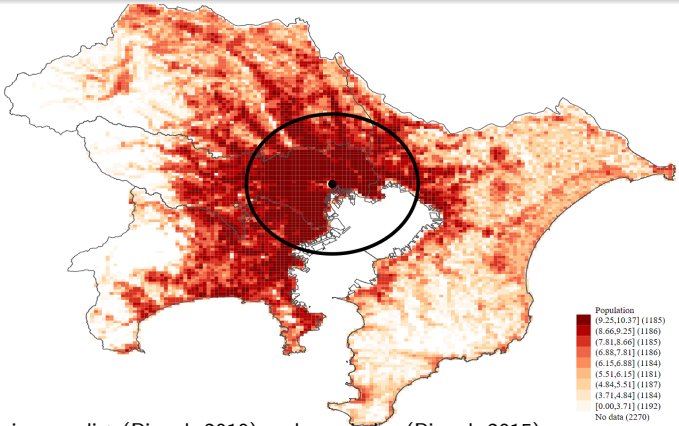


Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the spgen command

☞ Local Market Range 0–25 km from each mesh grid

```
.  gen rowif_sales_per_area = (sales_per_area > 0 & sales_per_area = .)
.  spgen pop_total_all, lon(lon) lat(lat) swm(bin) nostd dist(25) dunit(km)
rowif(rowif_sales_per_area) largesize replace
```

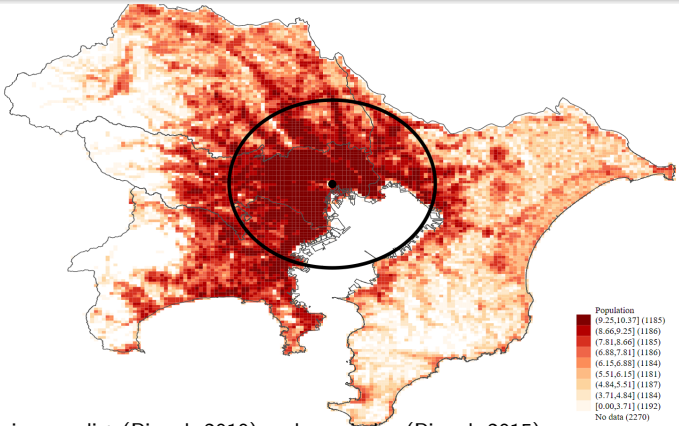


Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the spgen command

☞ Local Market Range 0–30 km from each mesh grid

```
. gen rowif_sales_per_area = (sales_per_area > 0 & sales_per_area = .)
. spgen pop_total_all, lon(lon) lat(lat) swm(bin) nostd dist(30) dunit(km)
rowif(rowif_sales_per_area) largesize replace
```

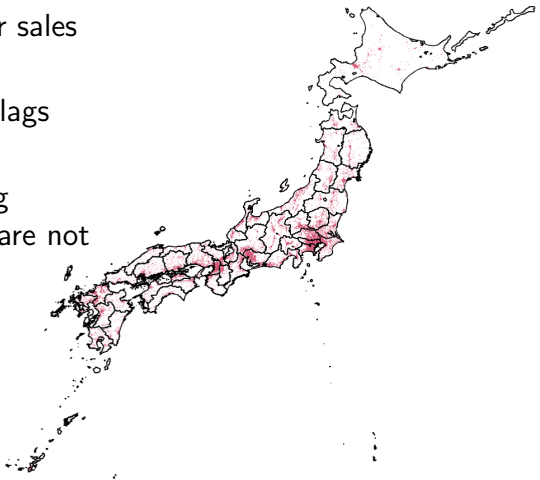


Source: Author's creation using geodist (Picard, 2010) and geocircles (Picard, 2015).

3. Application of the `spgen` command

Mesh Data: Sales per floor area

- The 38,334 of 391,451 grids are valid for sales per floor area.
- The `rowif()` option returns the spatial lags only for the selected observations.
- We can save calculation time by skipping observations with missing values, which are not used in the regression.



Source: Author's creation based on the 2014 Census of Commerce (METI). The red mesh grid indicates that the sales per floor area take valid values. The 35,936 mesh grids have anonymized data due to the small number of establishments in the corresponding mesh grids.

3. Application of the `spgen` command

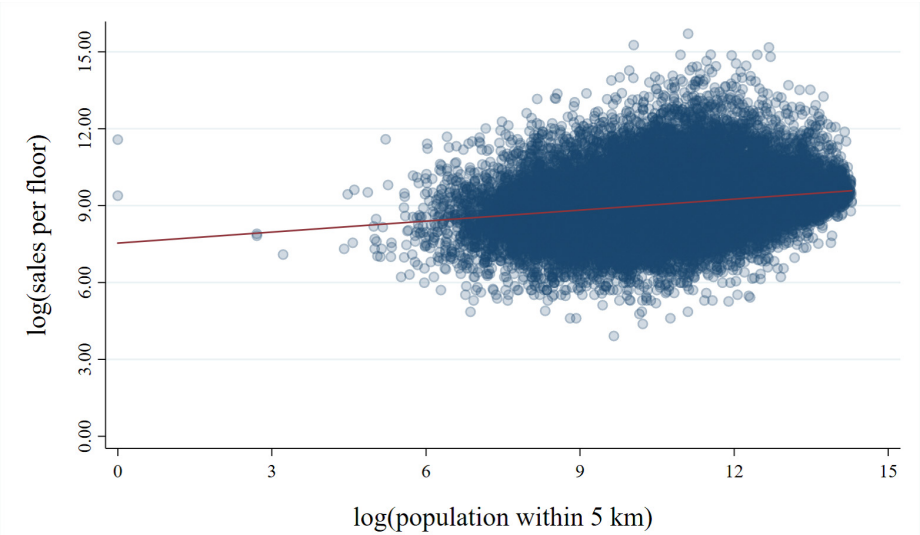


Fig: Scatter plot of sales per floor area and local market size within 5 km circle

3. Application of the spgen command

$$\log(SalesPerFloor_i) = \beta_{0\text{-}k\text{km}} \log\left(LocalPopulation_{i,0\text{-}k\text{km}}\right) + Controls_i + u_i$$

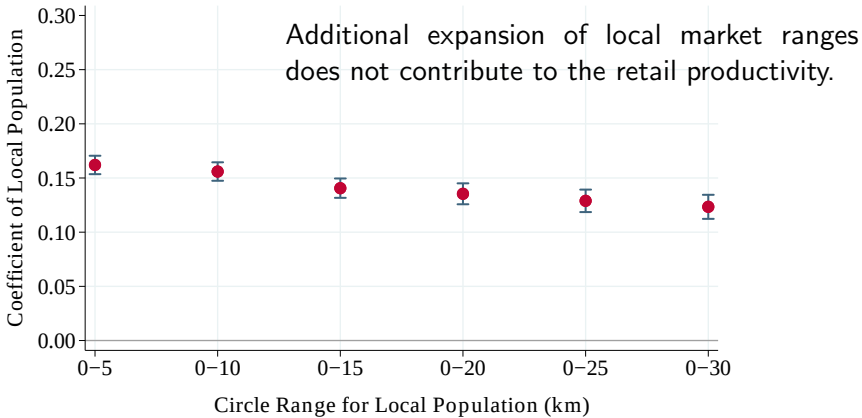


Fig: Coefficient Plot

Source: Author's creation. Controls include prefecture dummies.

4. Concluding remarks

- The spgen command is useful for the following two reasons:

spgenerate

- Spatial econometrics

spgen

- Spatial econometrics
- Geocoded microdata + Geospatial data (big data)

- Beyond the conventional spatial econometrics, the spgen command helps researchers who conduct a spatial analysis using geocoded microdata and high-dimensional mesh data (big data).
- A future challenge is the parallel computing for speeding up the calculation.

References

- Harris, Chauncy D. (1954) "The Market as a Factor in the Localization of Industry in the United States," *Annals of the Association of American Geographers*, 44(4), pp. 315–348.
- Kondo, Keisuke (2015) "SPGEN: Stata module to generate spatially lagged variables," Statistical Software Components S458105, Boston College Department of Economics, revised 17 Jun 2021.
- Picard, Robert (2010) "GEODIST: Stata module to compute geographical distances," Statistical Software Components S457147, Boston College Department of Economics, revised 24 Jun 2019.
- Picard, Robert (2015) "GEOCIRCLES: Stata module to create circles defined by geographic coordinates," Statistical Software Components S457991, Boston College Department of Economics, revised 16 Aug 2015.
- StataCorp LLC (2023) *Spatial Autoregressive Models Reference Manual*, Release 18, College Station: Stata Press.

Thank you for listening.

Any comments and suggestions would be greatly appreciated.

To install the spgen command, type:

```
ssc install spgen
```

kondo-keisuke@rieti.go.jp
https://keisukekondokk.github.io/