Spatial Data II: Spatial Econometrics

Francisco Villamil

Applied Quantitative Methods II

MA in Social Sciences, Spring 2026

Today's goals

- Understand why spatial autocorrelation violates OLS assumptions
- Diagnose spatial dependence: Moran's I on residuals and LM tests
- Estimate the Spatial Error Model (SEM) and interpret λ
- Estimate the Spatial Lag Model (SLM/SAR) and interpret ρ
- Compute direct and indirect effects after SLM estimation
- Understand estimation methods: ML, IV/2SLS, and why OLS fails
- Choose between SEM, SLM, SDM, and extensions (SAC, SDEM)
- Compare models: LR tests for nested models, AIC for non-nested

Roadmap

A standard OLS regression

$$y_i = \alpha + \mathbf{x}_i \boldsymbol{\beta} + \varepsilon_i$$

- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i

A standard OLS regression

$$y_i = \alpha + \mathbf{x}_i \boldsymbol{\beta} + \varepsilon_i$$

- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i
- Spatial data violates this: observations are **not independent**

A standard OLS regression

$$y_i = \alpha + \mathbf{x}_i \boldsymbol{\beta} + \varepsilon_i$$

- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i
- Spatial data violates this: observations are **not independent**
  - $\rightarrow$ Neighboring units share unmeasured common factors

A standard OLS regression

$$y_i = \alpha + \mathbf{x}_i\boldsymbol{\beta} + \varepsilon_i$$

- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i
- Spatial data violates this: observations are **not independent**
  - → Neighboring units share unmeasured common factors
  - → Outcomes may genuinely spill over across borders

A standard OLS regression

$$y_i = \alpha + \mathbf{x}_i \boldsymbol{\beta} + \varepsilon_i$$

- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i
- Spatial data violates this: observations are **not independent**
  - → Neighboring units share unmeasured common factors
  - → Outcomes may genuinely spill over across borders
- Consequence: OLS standard errors are **biased**

A standard OLS regression

$$y_i = \alpha + \mathbf{x}_i \boldsymbol{\beta} + \varepsilon_i$$

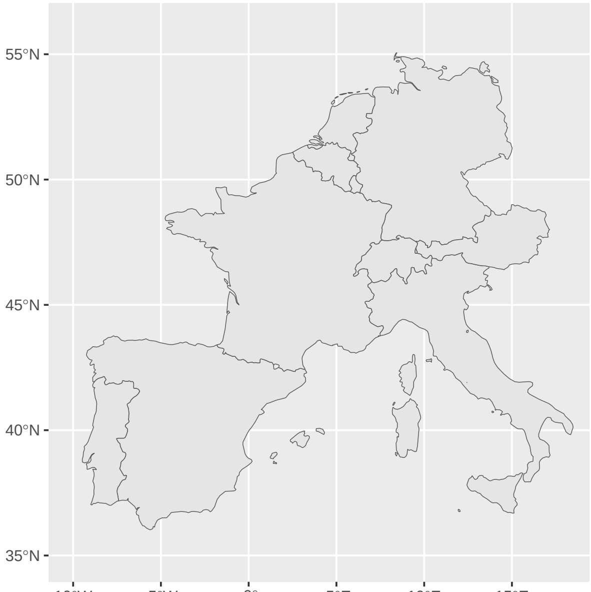
- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i
- Spatial data violates this: observations are **not independent**
  - → Neighboring units share unmeasured common factors
  - → Outcomes may genuinely spill over across borders
- Consequence: OLS standard errors are **biased**
  - → Same problem as with clustered or heteroskedastic errors

A standard OLS regression

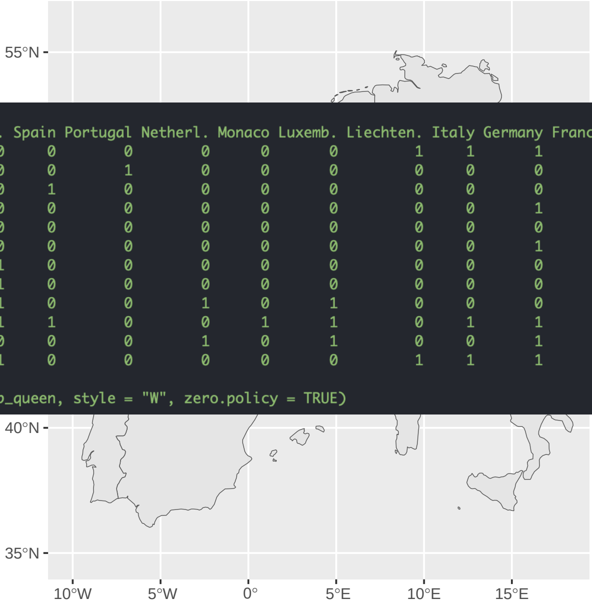
$$y_i = \alpha + \mathbf{x}_i\boldsymbol{\beta} + \varepsilon_i$$

- Classical assumption: $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent across i
- Spatial data violates this: observations are **not independent**
  - → Neighboring units share unmeasured common factors
  - → Outcomes may genuinely spill over across borders
- Consequence: OLS standard errors are **biased**
  - → Same problem as with clustered or heteroskedastic errors
  - → But the structure of dependence is explicitly *geographic*

Spatial weights: a brief recap

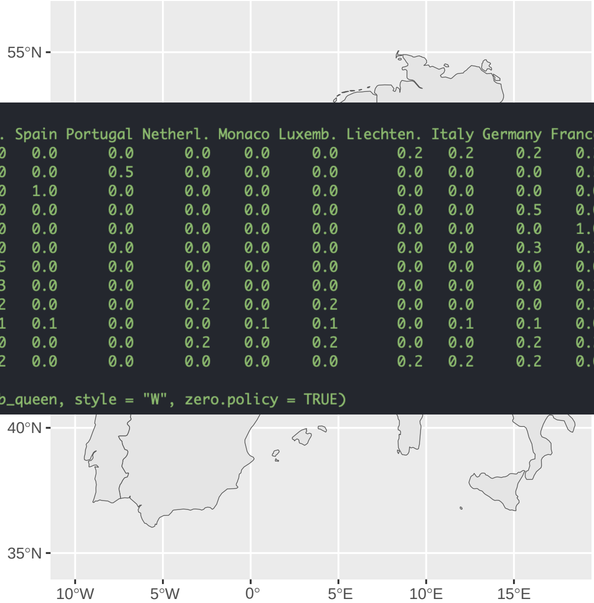


Spatial weights: a brief recap




```
> mat
              Switzer. Spain Portugal Netherl. Monaco Luxemb. Liechten. Italy Germany France Belgium Austria
Switzerland          0     0        0        0      0       0         1     1       1      1       0       1
Spain                0     0        1        0      0       0         0     0       0      1       0       0
Portugal             0     1        0        0      0       0         0     0       0      0       0       0
Netherlands          0     0        0        0      0       0         0     0       1      0       1       0
Monaco               0     0        0        0      0       0         0     0       0      1       0       0
Luxembourg           0     0        0        0      0       0         0     0       1      1       1       0
Liechtenstein        1     0        0        0      0       0         0     0       0      0       0       1
Italy                1     0        0        0      0       0         0     0       0      1       0       1
Germany              1     0        0        1      0       1         0     0       0      1       1       1
France               1     1        0        0      1       1         0     1       1      0       1       0
Belgium              0     0        0        1      0       1         0     0       1      1       0       0
Austria              1     0        0        0      0       0         1     1       1      0       0       0
attr(,"call")
nb2mat(neighbours = nb_queen, style = "W", zero.policy = TRUE)
```

Spatial weights: a brief recap




```
> mat
              Switzer. Spain Portugal Netherl. Monaco Luxemb. Liechten. Italy Germany France Belgium Austria
Switzerland        0.0   0.0      0.0      0.0    0.0     0.0       0.2   0.2     0.2    0.2     0.0     0.2
Spain              0.0   0.0      0.5      0.0    0.0     0.0       0.0   0.0     0.0    0.5     0.0     0.0
Portugal           0.0   1.0      0.0      0.0    0.0     0.0       0.0   0.0     0.0    0.0     0.0     0.0
Netherlands        0.0   0.0      0.0      0.0    0.0     0.0       0.0   0.0     0.5    0.0     0.5     0.0
Monaco             0.0   0.0      0.0      0.0    0.0     0.0       0.0   0.0     0.0    1.0     0.0     0.0
Luxembourg         0.0   0.0      0.0      0.0    0.0     0.0       0.0   0.0     0.3    0.3     0.3     0.0
Liechtenstein      0.5   0.0      0.0      0.0    0.0     0.0       0.0   0.0     0.0    0.0     0.0     0.5
Italy              0.3   0.0      0.0      0.0    0.0     0.0       0.0   0.0     0.0    0.3     0.0     0.3
Germany            0.2   0.0      0.0      0.2    0.0     0.2       0.0   0.0     0.0    0.2     0.2     0.2
France             0.1   0.1      0.0      0.0    0.1     0.1       0.0   0.1     0.1    0.0     0.1     0.0
Belgium            0.0   0.0      0.0      0.2    0.0     0.2       0.0   0.0     0.2    0.2     0.0     0.0
Austria            0.2   0.0      0.0      0.0    0.0     0.0       0.2   0.2     0.2    0.0     0.0     0.0
attr(,"call")
nb2mat(neighbours = nb_queen, style = "W", zero.policy = TRUE)
```

Spatial weights: a brief recap

Types of adjacency matrices (with spatial data)

Spatial weights: a brief recap

Types of adjacency matrices (with spatial data)

1. Based on **contiguity**
   → Queen or rook (or bishop, less common)

Spatial weights: a brief recap

Types of adjacency matrices (with spatial data)

1. Based on **contiguity**
   → Queen or rook (or bishop, less common)
2. Based on **distance**
   → Distance threshold
   → *k*-nearest neighbors (kNN)

Spatial weights: a brief recap

Types of adjacency matrices (with spatial data)

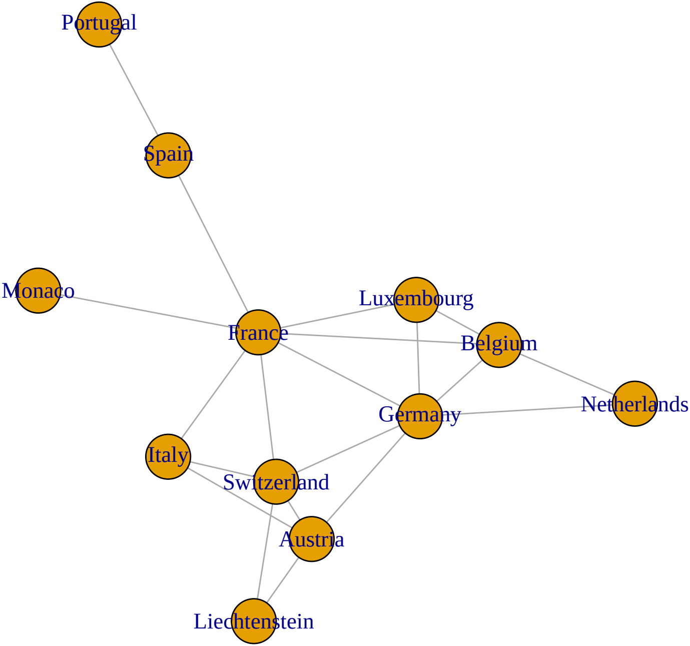
1. Based on **contiguity**
   → Queen or rook (or bishop, less common)
2. Based on **distance**
   → Distance threshold
   → *k*-nearest neighbors (kNN)
3. Non-binary: weighted adjacency
   → Based on distance (from border? from centroid? capital?), length of shared border, etc

Spatial weights: a brief recap

Types of adjacency matrices (with spatial data)

1. Based on **contiguity**
   → Queen or rook (or bishop, less common)
2. Based on **distance**
   → Distance threshold
   → *k*-nearest neighbors (kNN)
3. Non-binary: weighted adjacency
   → Based on distance (from border? from centroid? capital?), length of shared border, etc

- Conceptually, the same as a **network**, but just geographical


Portugal
Spain
Monaco
Luxembourg
France
Belgium
Netherlands
Germany
Italy
Switzerland
Austria
Liechtenstein

Spatial weights: a brief recap

Types of adjacency matrices (with spatial data)

1. Based on **contiguity**
   → Queen or rook (or bishop, less common)
2. Based on **distance**
   → Distance threshold
   → *k*-nearest neighbors (kNN)
3. Non-binary: weighted adjacency
   → Based on distance (from border? from centroid? capital?), length of shared border, etc

- Conceptually, the same as a **network**, but just geographical
  → e.g. difference with a network based on trade flows

Spatial weights: a brief recap

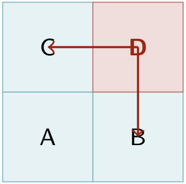
Types of adjacency matrices (with spatial data)

1. Based on **contiguity**
   → Queen or rook (or bishop, less common)
2. Based on **distance**
   → Distance threshold
   → *k*-nearest neighbors (kNN)
3. Non-binary: weighted adjacency
   → Based on distance (from border? from centroid? capital?), length of shared border, etc

- Conceptually, the same as a **network**, but just geographical
  → e.g. difference with a network based on trade flows
  (but you could use **the same models on network data**!)

Spatial weights: a brief recap

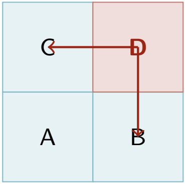
- **W**: the $n \times n$ spatial weights matrix



D's neighbors: B, C

Spatial weights: a brief recap

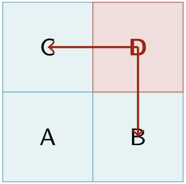
- **W**: the $n \times n$ spatial weights matrix
- $w_{ij} > 0$ if i and j are neighbors, 0 otherwise



D's neighbors: B, C

Spatial weights: a brief recap

- **W**: the $n \times n$ spatial weights matrix
- $w_{ij} > 0$ if i and j are neighbors, 0 otherwise
- (row-standardized matrix: rows sum to 1)

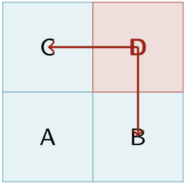


D's neighbors: B, C

Spatial weights: a brief recap

- **W**: the $n \times n$ spatial weights matrix
- $w_{ij} > 0$ if i and j are neighbors, 0 otherwise
- (row-standardized matrix: rows sum to 1)

- **Wy**: the **spatial lag** of **y**

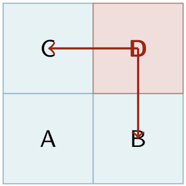


D's neighbors: B, C

Spatial weights: a brief recap

- **W**: the $n \times n$ spatial weights matrix
- $w_{ij} > 0$ if i and j are neighbors, 0 otherwise
- (row-standardized matrix: rows sum to 1)

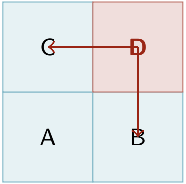
- **Wy**: the **spatial lag** of **y**
  - → For each unit i: weighted average of neighbors' y



D's neighbors: B, C

Spatial weights: a brief recap

- **W**: the $n \times n$ spatial weights matrix
- $w_{ij} > 0$ if i and j are neighbors, 0 otherwise
- (row-standardized matrix: rows sum to 1)



D's neighbors: B, C

- **Wy**: the **spatial lag** of **y**
  → For each unit i: weighted average of neighbors' y

- In R: `poly2nb()` → `nb2listw()`

Moran's I

$$I = \frac{n}{\sum_i \sum_j w_{ij}} \cdot \frac{\sum_i \sum_j w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\sum_i (y_i - \bar{y})^2}$$

- Ranges from -1 (perfect dispersion) to $+1$ (perfect clustering)

Moran's I

$$I = \frac{n}{\sum_i \sum_j w_{ij}} \cdot \frac{\sum_i \sum_j w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\sum_i (y_i - \bar{y})^2}$$

- Ranges from -1 (perfect dispersion) to $+1$ (perfect clustering)
- $I \approx 0$: no spatial autocorrelation (random)

Moran's I

$$I = \frac{n}{\sum_i \sum_j w_{ij}} \cdot \frac{\sum_i \sum_j w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\sum_i (y_i - \bar{y})^2}$$

- Ranges from -1 (perfect dispersion) to $+1$ (perfect clustering)
- $I \approx 0$: no spatial autocorrelation (random)
- Intuition: correlation between each unit's value and its **neighbors' values**

Moran's I

$$I = \frac{n}{\sum_i \sum_j w_{ij}} \cdot \frac{\sum_i \sum_j w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\sum_i (y_i - \bar{y})^2}$$

- Ranges from -1 (perfect dispersion) to $+1$ (perfect clustering)
- $I \approx 0$: no spatial autocorrelation (random)
- Intuition: correlation between each unit's value and its **neighbors' values**
  - → High I: similar values cluster together (hot spots and cold spots)

Moran's I

$$I = \frac{n}{\sum_i \sum_j w_{ij}} \cdot \frac{\sum_i \sum_j w_{ij}(y_i - \bar{y})(y_j - \bar{y})}{\sum_i (y_i - \bar{y})^2}$$

- Ranges from -1 (perfect dispersion) to $+1$ (perfect clustering)
- $I \approx 0$: no spatial autocorrelation (random)
- Intuition: correlation between each unit's value and its **neighbors' values**
  - → High I: similar values cluster together (hot spots and cold spots)
  - → Low I: high and low values alternate (checkerboard pattern)

The key diagnostic: Moran's I on residuals

- Run OLS first, compute residuals $\hat{\varepsilon}_i = y_i - \hat{y}_i$

```
ols_fit = lm(y ~ x, data = mydata)
moran.test(residuals(ols_fit), listw = my_listw)
```

Significant Moran's I on residuals $\Rightarrow$ OLS is insufficient

The key diagnostic: Moran's I on residuals

- Run OLS first, compute residuals $\hat{\varepsilon}_i = y_i - \hat{y}_i$
- Apply Moran's I to the **residuals**, not to y

```
ols_fit = lm(y ~ x, data = mydata)
moran.test(residuals(ols_fit), listw = my_listw)
```

Significant Moran's I on residuals $\Rightarrow$ OLS is insufficient

The key diagnostic: Moran's I on residuals

- Run OLS first, compute residuals $\hat{\varepsilon}_i = y_i - \hat{y}_i$
- Apply Moran's I to the **residuals**, not to y
  - $\rightarrow$ Spatial autocorrelation in y alone may be explained by $\mathbf{x}$

```
ols_fit = lm(y ~ x, data = mydata)
moran.test(residuals(ols_fit), listw = my_listw)
```

Significant Moran's I on residuals $\Rightarrow$ OLS is insufficient

The key diagnostic: Moran's I on residuals

- Run OLS first, compute residuals $\hat{\varepsilon}_i = y_i - \hat{y}_i$
- Apply Moran's I to the **residuals**, not to y
  - → Spatial autocorrelation in y alone may be explained by $\mathbf{x}$
  - → Autocorrelation in $\hat{\varepsilon}$ means the model has not removed the spatial structure

```
ols_fit = lm(y ~ x, data = mydata)
moran.test(residuals(ols_fit), listw = my_listw)
```

Significant Moran's I on residuals $\Rightarrow$ OLS is insufficient

The key diagnostic: Moran's I on residuals

- Run OLS first, compute residuals $\hat{\varepsilon}_i = y_i - \hat{y}_i$
- Apply Moran's I to the **residuals**, not to y
  - $\rightarrow$ Spatial autocorrelation in y alone may be explained by $\mathbf{x}$
  - $\rightarrow$ Autocorrelation in $\hat{\varepsilon}$ means the model has not removed the spatial structure
- In R:

```
ols_fit = lm(y ~ x, data = mydata)
moran.test(residuals(ols_fit), listw = my_listw)
```

Significant Moran's I on residuals $\Rightarrow$ OLS is insufficient

Two sources of spatial dependence

Moran's I on OLS residuals is significant

— why?

Nuisance correlation
Unmeasured spatially correlated factors

True spillover
Outcome in unit i affects neighbors j

⇒ Spatial Error Model

⇒ Spatial Lag Model

Same symptom, different model

Think about **your projects**: which mechanism is more plausible?

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence

```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence
- Two tests, evaluated jointly:

```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence
- Two tests, evaluated jointly:
  → **LMerr**: tests for spatial error dependence ($\lambda \neq 0$)

```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence
- Two tests, evaluated jointly:
  - → **LMerr**: tests for spatial error dependence ($\lambda \neq 0$)
  - → **LMlag**: tests for spatial lag dependence ($\rho \neq 0$)

```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence
- Two tests, evaluated jointly:
  - → **LMerr**: tests for spatial error dependence ($\lambda \neq 0$)
  - → **LMlag**: tests for spatial lag dependence ($\rho \neq 0$)
- Robust versions: **RLMerr** and **RLMlag**

```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence
- Two tests, evaluated jointly:
  - → **LMerr**: tests for spatial error dependence ($\lambda \neq 0$)
  - → **LMlag**: tests for spatial lag dependence ($\rho \neq 0$)
- Robust versions: **RLMerr** and **RLMlag**
  - → Control for the presence of the other type of dependence

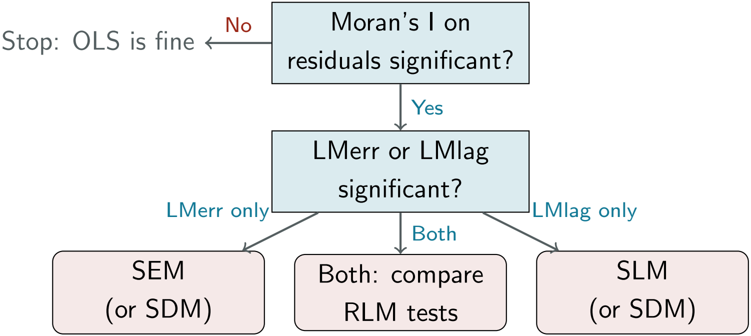
```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM tests: which model to choose?

- **LM tests** (Lagrange Multiplier): test for specific types of spatial dependence
- Two tests, evaluated jointly:
  - → **LMerr**: tests for spatial error dependence ($\lambda \neq 0$)
  - → **LMlag**: tests for spatial lag dependence ($\rho \neq 0$)
- Robust versions: **RLMerr** and **RLMlag**
  - → Control for the presence of the other type of dependence
- In R:

```
lm.LMtests(ols_fit, listw = my_listw,
  test = c("LMerr", "LMlag", "RLMerr", "RLMlag"))
```

LM decision rule



Roadmap

SEM: the model

Spatial Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- **W**: spatial weights matrix

SEM: the model

Spatial Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- **W**: spatial weights matrix
- λ (lambda): spatial autocorrelation in the **error term**

SEM: the model

Spatial Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- **W**: spatial weights matrix
- λ (lambda): spatial autocorrelation in the **error term**
  - $\rightarrow$ $\lambda = 0$: back to standard OLS

SEM: the model

Spatial Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u}$$

$$\mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$$

- **W**: spatial weights matrix
- λ (lambda): spatial autocorrelation in the **error term**
  - → $\lambda = 0$: back to standard OLS
  - → $\lambda > 0$: positive spatial autocorrelation in errors

SEM: the model

Spatial Error Model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- $\mathbf{W}$: spatial weights matrix
- λ (lambda): spatial autocorrelation in the **error term**
  - → $\lambda = 0$: back to standard OLS
  - → $\lambda > 0$: positive spatial autocorrelation in errors
- $\boldsymbol{\beta}$: same interpretation as OLS

SEM: the model

Spatial Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u}$$

$$\mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- **W**: spatial weights matrix
- λ (lambda): spatial autocorrelation in the **error term**
  - → $\lambda = 0$: back to standard OLS
  - → $\lambda > 0$: positive spatial autocorrelation in errors
- $\boldsymbol{\beta}$: same interpretation as OLS
- The spatial structure is in the **residuals**, not in the outcome

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:
  - → Country democracy regressed on GDP: unmeasured historical institutions cluster geographically

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:
  - → Country democracy regressed on GDP: unmeasured historical institutions cluster geographically
  - → Municipal health outcomes: regional infrastructure and public goods affect nearby areas

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:
  - → Country democracy regressed on GDP: unmeasured historical institutions cluster geographically
  - → Municipal health outcomes: regional infrastructure and public goods affect nearby areas
  - → Conflict event density: unobserved security capacity clusters across provinces

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:
  - $\rightarrow$ Country democracy regressed on GDP: unmeasured historical institutions cluster geographically
  - $\rightarrow$ Municipal health outcomes: regional infrastructure and public goods affect nearby areas
  - $\rightarrow$ Conflict event density: unobserved security capacity clusters across provinces
- $\lambda \neq 0$ does **not** mean y affects neighbors' y

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:
  - → Country democracy regressed on GDP: unmeasured historical institutions cluster geographically
  - → Municipal health outcomes: regional infrastructure and public goods affect nearby areas
  - → Conflict event density: unobserved security capacity clusters across provinces
- $\lambda \neq 0$ does **not** mean y affects neighbors' y
  - → It means the residuals are spatially structured

SEM: substantive interpretation

- The SEM says: **unmeasured factors** that affect y are geographically clustered
- Examples:
  - → Country democracy regressed on GDP: unmeasured historical institutions cluster geographically
  - → Municipal health outcomes: regional infrastructure and public goods affect nearby areas
  - → Conflict event density: unobserved security capacity clusters across provinces
- $\lambda \neq 0$ does **not** mean y affects neighbors' y
  - → It means the residuals are spatially structured
- Fix: correct SEs and improve efficiency — not model substantive spillover

SEM in R

```
library(spatialreg) # or spdep

# Fit Spatial Error Model
sem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata,
    listw = my_listw,
    zero.policy = TRUE)

summary(sem_fit)
```

- `errorsarlm()`: maximum likelihood estimation of SEM

SEM in R

```
library(spatialreg) # or spdep

# Fit Spatial Error Model
sem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata,
    listw = my_listw,
    zero.policy = TRUE)

summary(sem_fit)
```

- `errorsarlm()`: maximum likelihood estimation of SEM
- `zero.policy = TRUE`: allow units with no neighbors

SEM in R

```
library(spatialreg) # or spdep

# Fit Spatial Error Model
sem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata,
    listw = my_listw,
    zero.policy = TRUE)

summary(sem_fit)
```

- `errorsarlm()`: maximum likelihood estimation of SEM
- `zero.policy = TRUE`: allow units with no neighbors
- Key output: $\hat{\lambda}$ and its significance; $\hat{\beta}$ interpreted as OLS

SEM: reading the output

	Estimate	**Std. Error**	***z*-value**	***p*-value**
(Intercept)	-2.14	0.48	-4.45	< 0.001
`gdp_pc`	0.63	0.07	9.11	< 0.001
`pop_l`	0.18	0.05	3.37	0.001
λ (Lambda)	0.54	0.09	5.80	< 0.001

- $\hat{\lambda} = 0.54$: strong positive spatial autocorrelation in errors

SEM: reading the output

	Estimate	**Std. Error**	***z*-value**	***p*-value**
(Intercept)	-2.14	0.48	-4.45	< 0.001
gdp_pc	0.63	0.07	9.11	< 0.001
pop_l	0.18	0.05	3.37	0.001
λ (Lambda)	0.54	0.09	5.80	< 0.001

- $\hat{\lambda} = 0.54$: strong positive spatial autocorrelation in errors
- $\hat{\boldsymbol{\beta}}$: interpreted exactly as in OLS

SEM: reading the output

	Estimate	**Std. Error**	*z*-**value**	*p*-**value**
(Intercept)	-2.14	0.48	-4.45	< 0.001
gdp_pc	0.63	0.07	9.11	< 0.001
pop_l	0.18	0.05	3.37	0.001
λ (Lambda)	0.54	0.09	5.80	< 0.001

- $\hat{\lambda} = 0.54$: strong positive spatial autocorrelation in errors
- $\hat{\boldsymbol{\beta}}$: interpreted exactly as in OLS
- Compare with OLS: coefficients may shift, SEs typically change

SEM: reading the output

	Estimate	**Std. Error**	***z*-value**	***p*-value**
(Intercept)	−2.14	0.48	−4.45	< 0.001
gdp_pc	0.63	0.07	9.11	< 0.001
pop_l	0.18	0.05	3.37	0.001
λ (Lambda)	0.54	0.09	5.80	< 0.001

- $\hat{\lambda} = 0.54$: strong positive spatial autocorrelation in errors
- $\hat{\boldsymbol{\beta}}$: interpreted exactly as in OLS
- Compare with OLS: coefficients may shift, SEs typically change
- Use AIC/LR test to confirm SEM improves on OLS

Roadmap

SLM: the model

Spatial Lag Model (SLM / SAR)

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- ρ (rho): spatial autoregressive parameter on the **lagged outcome**

SLM: the model

Spatial Lag Model (SLM / SAR)

$$\mathbf{y} = \rho\mathbf{W}\mathbf{y} + \mathbf{X}\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$$

- ρ (rho): spatial autoregressive parameter on the **lagged outcome**
  - → $\rho = 0$: standard OLS

SLM: the model

Spatial Lag Model (SLM / SAR)

$$\mathbf{y} = \rho \mathbf{W} \mathbf{y} + \mathbf{X} \boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- ρ (rho): spatial autoregressive parameter on the **lagged outcome**
  - → $\rho = 0$: standard OLS
  - → $\rho > 0$: positive spatial diffusion (clustering in y)

SLM: the model

Spatial Lag Model (SLM / SAR)

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- ρ (rho): spatial autoregressive parameter on the **lagged outcome**
  - $\rightarrow$ $\rho = 0$: standard OLS
  - $\rightarrow$ $\rho > 0$: positive spatial diffusion (clustering in y)
- $\mathbf{Wy}$: the spatial lag — average of neighbors' y

SLM: the model

Spatial Lag Model (SLM / SAR)

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- ρ (rho): spatial autoregressive parameter on the **lagged outcome**
  - → $\rho = 0$: standard OLS
  - → $\rho > 0$: positive spatial diffusion (clustering in y)
- $\mathbf{W}\mathbf{y}$: the spatial lag — average of neighbors' y
- Key difference from SEM: the spatial lag is on the **outcome**, not the error

SLM: the model

Spatial Lag Model (SLM / SAR)

$$\mathbf{y} = \rho \mathbf{W} \mathbf{y} + \mathbf{X}\beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

- ρ (rho): spatial autoregressive parameter on the **lagged outcome**
  - → $\rho = 0$: standard OLS
  - → $\rho > 0$: positive spatial diffusion (clustering in y)
- $\mathbf{Wy}$: the spatial lag — average of neighbors' y
- Key difference from SEM: the spatial lag is on the **outcome**, not the error
- y_i is partly determined by its neighbors' y_j

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:
  → **Democracy**: a country becomes more democratic partly because its neighbors are democratic

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:
  - → **Democracy**: a country becomes more democratic partly because its neighbors are democratic
  - → **Economic development**: city growth spills over to nearby cities (agglomeration)

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:
  - → **Democracy**: a country becomes more democratic partly because its neighbors are democratic
  - → **Economic development**: city growth spills over to nearby cities (agglomeration)
  - → **Conflict**: violence in one district raises conflict risk in adjacent districts

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:
  → **Democracy**: a country becomes more democratic partly because its neighbors are democratic
  → **Economic development**: city growth spills over to nearby cities (agglomeration)
  → **Conflict**: violence in one district raises conflict risk in adjacent districts
  → **Policy adoption**: states imitate neighboring states' welfare policies

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:
  - → **Democracy**: a country becomes more democratic partly because its neighbors are democratic
  - → **Economic development**: city growth spills over to nearby cities (agglomeration)
  - → **Conflict**: violence in one district raises conflict risk in adjacent districts
  - → **Policy adoption**: states imitate neighboring states' welfare policies
- $\rho > 0$: spatial diffusion (neighbors' high y raises your y)

SLM: substantive interpretation

- The SLM says: your outcome **depends on neighbors' outcomes**
- Examples of genuine spatial diffusion:
  - → **Democracy**: a country becomes more democratic partly because its neighbors are democratic
  - → **Economic development**: city growth spills over to nearby cities (agglomeration)
  - → **Conflict**: violence in one district raises conflict risk in adjacent districts
  - → **Policy adoption**: states imitate neighboring states' welfare policies
- $\rho > 0$: spatial diffusion (neighbors' high y raises your y)
- Warning: $\hat{\beta}$ from SLM is **not** the marginal effect of $\mathbf{x}$ on y

SLM in R

```
library(spatialreg)

# Fit Spatial Lag Model
slm_fit = lagsarlm(y ~ x1 + x2,
    data = mydata,
    listw = my_listw,
    zero.policy = TRUE)

summary(slm_fit)
```

- `lagsarlm()`: ML estimation of SLM

SLM in R

```
library(spatialreg)

# Fit Spatial Lag Model
slm_fit = lagsarlm(y ~ x1 + x2,
    data = mydata,
    listw = my_listw,
    zero.policy = TRUE)

summary(slm_fit)
```

- `lagsarlm()`: ML estimation of SLM
- Key output: $\hat{\rho}$ and its significance; $\hat{\beta}$ (not marginal effects!)

SLM in R

```
library(spatialreg)

# Fit Spatial Lag Model
slm_fit = lagsarlm(y ~ x1 + x2,
    data = mydata,
    listw = my_listw,
    zero.policy = TRUE)

summary(slm_fit)
```

- `lagsarlm()`: ML estimation of SLM
- Key output: $\hat{\rho}$ and its significance; $\hat{\beta}$ (not marginal effects!)
- Do not interpret $\hat{\beta}$ directly — use `impacts()` instead

SLM: reading the output

	Estimate	Std. Error	z-value	p-value
(Intercept)	-1.87	0.52	-3.59	< 0.001
gdp_pc	0.51	0.08	6.29	< 0.001
pop_l	0.14	0.05	2.85	0.004
ρ (Rho)	0.48	0.10	4.90	< 0.001

- $\hat{\rho} = 0.48$: strong positive spatial diffusion in y

SLM: reading the output

	Estimate	Std. Error	z-value	p-value
(Intercept)	-1.87	0.52	-3.59	< 0.001
gdp_pc	0.51	0.08	6.29	< 0.001
pop_l	0.14	0.05	2.85	0.004
ρ (Rho)	0.48	0.10	4.90	< 0.001

- $\hat{\rho} = 0.48$: strong positive spatial diffusion in y
- $\hat{\boldsymbol{\beta}}$ coefficients appear in output **but are not marginal effects**

SLM: reading the output

	Estimate	**Std. Error**	***z*-value**	***p*-value**
(Intercept)	-1.87	0.52	-3.59	< 0.001
gdp_pc	0.51	0.08	6.29	< 0.001
pop_l	0.14	0.05	2.85	0.004
ρ (Rho)	0.48	0.10	4.90	< 0.001

- $\hat{\rho} = 0.48$: strong positive spatial diffusion in y
- $\hat{\boldsymbol{\beta}}$ coefficients appear in output **but are not marginal effects**
- Marginal effects are obtained via `impacts(slm_fit, listw = my_listw)`

Roadmap

Introduction to spatial dependence

The Spatial Error Model (SEM)

The Spatial Lag Model (SLM/SAR)

Direct and Indirect Effects

The Spatial Durbin Model (SDM)

Estimation Methods

Model Selection and Practical Guidance

Model Relationships and Extensions

Beyond Cross-Sections: Spatial Panel Models

Changing x_i creates a ripple through the network.

The coefficient $\hat{\beta}$ is not the marginal effect.

Why $\hat{\boldsymbol{\beta}}$ is not the marginal effect

Solving the SLM for $\mathbf{y}$:

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$
$$(\mathbf{I} - \rho \mathbf{W})\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$
$$\mathbf{y} = \underbrace{(\mathbf{I} - \rho \mathbf{W})^{-1}}_{\mathbf{B}}(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon})$$

- $\mathbf{B} = (\mathbf{I} - \rho \mathbf{W})^{-1}$: the **equilibrium effects matrix**

Why $\hat{\boldsymbol{\beta}}$ is not the marginal effect

Solving the SLM for **y**:

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$(\mathbf{I} - \rho \mathbf{W})\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$\mathbf{y} = \underbrace{(\mathbf{I} - \rho \mathbf{W})^{-1}}_{\mathbf{B}}(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon})$$

- $\mathbf{B} = (\mathbf{I} - \rho \mathbf{W})^{-1}$: the **equilibrium effects matrix**
- A change in x_i propagates through **B**: affects y_i **and** all y_j

Why $\hat{\boldsymbol{\beta}}$ is not the marginal effect

Solving the SLM for **y**:

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$
$$(\mathbf{I} - \rho \mathbf{W})\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$
$$\mathbf{y} = \underbrace{(\mathbf{I} - \rho \mathbf{W})^{-1}}_{\mathbf{B}}(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon})$$

- $\mathbf{B} = (\mathbf{I} - \rho \mathbf{W})^{-1}$: the **equilibrium effects matrix**
- A change in x_i propagates through **B**: affects y_i **and** all y_j
- $\partial y_i / \partial x_i \neq \hat{\beta}$

Why $\hat{\beta}$ is not the marginal effect

Solving the SLM for $\mathbf{y}$:

$$\mathbf{y} = \rho \mathbf{W} \mathbf{y} + \mathbf{X} \boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

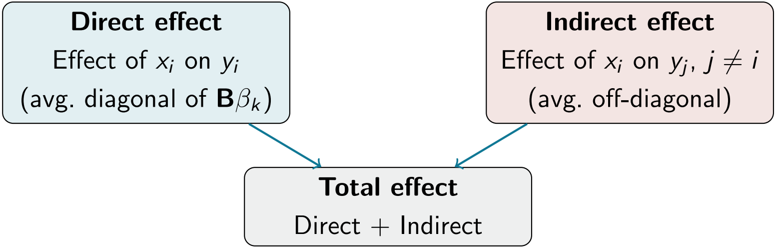
$$(\mathbf{I} - \rho \mathbf{W}) \mathbf{y} = \mathbf{X} \boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$\mathbf{y} = \underbrace{(\mathbf{I} - \rho \mathbf{W})^{-1}}_{\mathbf{B}} (\mathbf{X} \boldsymbol{\beta} + \boldsymbol{\varepsilon})$$

- $\mathbf{B} = (\mathbf{I} - \rho \mathbf{W})^{-1}$: the **equilibrium effects matrix**
- A change in x_i propagates through $\mathbf{B}$: affects y_i **and** all y_j
- $\partial y_i / \partial x_i \neq \hat{\beta}$

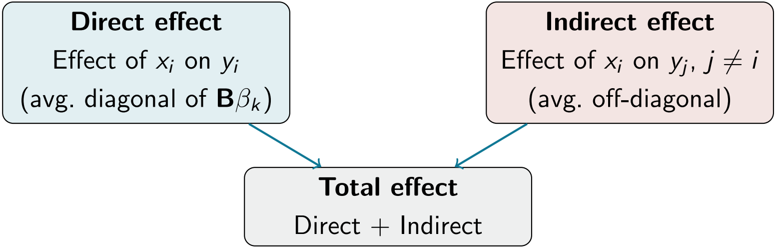
- **Question:** if you change x_i, how many units in the network are eventually affected?

Direct, indirect, and total effects



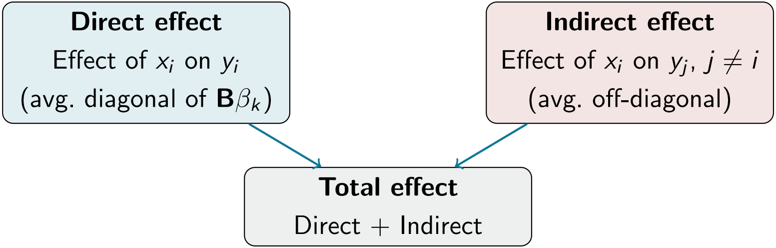
- Direct $\approx \hat{\beta}_k$ but slightly larger due to own-unit feedback

Direct, indirect, and total effects



- Direct $\approx \hat{\beta}_k$ but slightly larger due to own-unit feedback
- Indirect: captures spatial spillover — ignored by OLS and SEM

Direct, indirect, and total effects



- Direct $\approx \hat{\beta}_k$ but slightly larger due to own-unit feedback
- Indirect: captures spatial spillover — ignored by OLS and SEM
- Total: the full equilibrium effect of a uniform shock to x_k

Computing impacts in R

```
# Direct, indirect, total effects
imp = impacts(slm_fit, listw = my_listw)
summary(imp, zstats = TRUE)
```

```
## Impact measures (lag, exact):
##  Direct Indirect Total
# gdp_pc 0.574  0.531  1.105
# pop_l  0.161  0.149  0.310
```

- **Direct** (gdp_pc): 0.574 — unit's own GDP raises own y by 0.574

Computing impacts in R

```
# Direct, indirect, total effects
imp = impacts(slm_fit, listw = my_listw)
summary(imp, zstats = TRUE)

## Impact measures (lag, exact):
##  Direct Indirect Total
# gdp_pc 0.574  0.531  1.105
# pop_l  0.161  0.149  0.310
```

- **Direct** (gdp_pc): 0.574 — unit's own GDP raises own y by 0.574
- **Indirect**: 0.531 — unit's GDP raises neighbors' y by 0.531 on average

Computing impacts in R

```
# Direct, indirect, total effects
imp = impacts(slm_fit, listw = my_listw)
summary(imp, zstats = TRUE)

## Impact measures (lag, exact):
##  Direct Indirect Total
# gdp_pc 0.574  0.531  1.105
# pop_l  0.161  0.149  0.310
```

- **Direct** (gdp_pc): 0.574 — unit's own GDP raises own y by 0.574
- **Indirect**: 0.531 — unit's GDP raises neighbors' y by 0.531 on average
- **Total**: 1.105 — full equilibrium effect ($> \hat{\beta}_k = 0.51$)

Computing impacts in R

```
# Direct, indirect, total effects
imp = impacts(slm_fit, listw = my_listw)
summary(imp, zstats = TRUE)

## Impact measures (lag, exact):
##  Direct Indirect Total
# gdp_pc 0.574   0.531  1.105
# pop_l  0.161   0.149  0.310
```

- **Direct** (gdp_pc): 0.574 — unit's own GDP raises own y by 0.574
- **Indirect**: 0.531 — unit's GDP raises neighbors' y by 0.531 on average
- **Total**: 1.105 — full equilibrium effect ($> \hat{\beta}_k = 0.51$)
- Use `zstats = TRUE` to get z-statistics and p-values

Roadmap

SDM: extending the SLM

Spatial Durbin Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + \boldsymbol{\varepsilon}$$

- Adds spatially lagged covariates $\mathbf{WX}\boldsymbol{\theta}$ to the SLM

SDM: extending the SLM

Spatial Durbin Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + \boldsymbol{\varepsilon}$$

- Adds spatially lagged covariates $\mathbf{WX}\boldsymbol{\theta}$ to the SLM
- θ_k: effect of **neighbors'** x_k on unit i's y

SDM: extending the SLM

Spatial Durbin Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + \varepsilon$$

- Adds spatially lagged covariates $\mathbf{WX}\boldsymbol{\theta}$ to the SLM
- θ_k: effect of **neighbors'** x_k on unit i's y
- Most flexible model:

SDM: extending the SLM

Spatial Durbin Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\beta + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + \varepsilon$$

- Adds spatially lagged covariates $\mathbf{W}\mathbf{X}\boldsymbol{\theta}$ to the SLM
- θ_k: effect of **neighbors'** x_k on unit i's y
- Most flexible model:
  - $\rightarrow$ SLM is nested ($\boldsymbol{\theta} = 0$)

SDM: extending the SLM

Spatial Durbin Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\beta + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + \varepsilon$$

- Adds spatially lagged covariates $\mathbf{WX}\boldsymbol{\theta}$ to the SLM
- θ_k: effect of **neighbors'** x_k on unit i's y
- Most flexible model:
  - → SLM is nested ($\boldsymbol{\theta} = 0$)
  - → SEM is nested ($\boldsymbol{\theta} = -\rho\boldsymbol{\beta}$)

SDM: extending the SLM

Spatial Durbin Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\beta + \mathbf{W}\mathbf{X}\boldsymbol{\theta} + \varepsilon$$

- Adds spatially lagged covariates $\mathbf{WX}\boldsymbol{\theta}$ to the SLM
- θ_k: effect of **neighbors'** x_k on unit i's y
- Most flexible model:
  - → SLM is nested ($\boldsymbol{\theta} = 0$)
  - → SEM is nested ($\boldsymbol{\theta} = -\rho\boldsymbol{\beta}$)
- LeSage & Pace (2009): SDM is often preferred as a conservative default

SDM in R and when to use it

```
# Spatial Durbin Model
sdm_fit = lagsarlm(y ~ x,
    data = mydata,
    listw = my_listw,
    Durbin = TRUE,
    zero.policy = TRUE)

# Direct, indirect, total as usual
impacts(sdm_fit, listw = my_listw)
```

- `Durbin = TRUE`: adds **WX** to `lagsarlm()`

SDM in R and when to use it

```
# Spatial Durbin Model
sdm_fit = lagsarlm(y ~ x,
    data = mydata,
    listw = my_listw,
    Durbin = TRUE,
    zero.policy = TRUE)

# Direct, indirect, total as usual
impacts(sdm_fit, listw = my_listw)
```

- `Durbin = TRUE`: adds **WX** to `lagsarlm()`
- Use SDM when: neighbors' **x** plausibly affects *y*

SDM in R and when to use it

```
# Spatial Durbin Model
sdm_fit = lagsarlm(y ~ x,
    data = mydata,
    listw = my_listw,
    Durbin = TRUE,
    zero.policy = TRUE)

# Direct, indirect, total as usual
impacts(sdm_fit, listw = my_listw)
```

- `Durbin = TRUE`: adds **WX** to `lagsarlm()`
- Use SDM when: neighbors' **x** plausibly affects *y*
- Also useful as robustness check for SLM estimates

Roadmap

How are these models estimated?

Model	**Estimation**	**Why not OLS?**
OLS	OLS	Baseline: valid if no spatial dependence
SEM	ML	OLS is unbiased but **inefficient**; SEs are wrong
SLM	ML (or IV/2SLS)	$\mathbf{Wy}$ is **endogenous**: OLS is biased
SDM	ML	Same endogeneity as SLM ($\rho\mathbf{Wy}$)

- **Maximum Likelihood** (ML): estimates $\boldsymbol{\beta}$ and spatial parameters (λ, ρ) jointly by maximizing the log-likelihood

How are these models estimated?

Model	**Estimation**	**Why not OLS?**
OLS	OLS	Baseline: valid if no spatial dependence
SEM	ML	OLS is unbiased but **inefficient**; SEs are wrong
SLM	ML (or IV/2SLS)	$\mathbf{Wy}$ is **endogenous**: OLS is biased
SDM	ML	Same endogeneity as SLM ($\rho\mathbf{Wy}$)

- **Maximum Likelihood** (ML): estimates $\boldsymbol{\beta}$ and spatial parameters (λ, ρ) jointly by maximizing the log-likelihood
- ML requires distributional assumption ($\varepsilon \sim \mathcal{N}$)

How are these models estimated?

Model	**Estimation**	**Why not OLS?**
OLS	OLS	Baseline: valid if no spatial dependence
SEM	ML	OLS is unbiased but **inefficient**; SEs are wrong
SLM	ML (or IV/2SLS)	$\mathbf{Wy}$ is **endogenous**: OLS is biased
SDM	ML	Same endogeneity as SLM ($\rho\mathbf{Wy}$)

- **Maximum Likelihood** (ML): estimates $\boldsymbol{\beta}$ and spatial parameters (λ, ρ) jointly by maximizing the log-likelihood
- ML requires distributional assumption ($\varepsilon \sim \mathcal{N}$)
- Alternative for SLM: **IV/2SLS** using $\mathbf{WX}$, $\mathbf{W}^2\mathbf{X}$ as instruments

How are these models estimated?

Model	**Estimation**	**Why not OLS?**
OLS	OLS	Baseline: valid if no spatial dependence
SEM	ML	OLS is unbiased but **inefficient**; SEs are wrong
SLM	ML (or IV/2SLS)	$\mathbf{Wy}$ is **endogenous**: OLS is biased
SDM	ML	Same endogeneity as SLM ($\rho\mathbf{Wy}$)

- **Maximum Likelihood** (ML): estimates $\boldsymbol{\beta}$ and spatial parameters (λ, ρ) jointly by maximizing the log-likelihood
- ML requires distributional assumption ($\varepsilon \sim \mathcal{N}$)
- Alternative for SLM: **IV/2SLS** using $\mathbf{WX}$, $\mathbf{W}^2\mathbf{X}$ as instruments
- Alternative for SEM: **Feasible GLS** or **GMM** (Kelejian & Prucha)

Roadmap

SEM vs SLM vs SDM: comparison

	SEM	**SLM**	**SDM**
What clusters	Errors ($\mathbf{u}$)	Outcome ($\mathbf{y}$)	Both
Spillover in y	No	Yes	Yes
Spillover from x	No	No	Yes
$\hat{\beta}$ **= marginal effect**	Yes	No	No
Use `impacts()`	No	Yes	Yes
Nuisance / design	Nuisance	Substantive	Both
R function	`errorsarlm()`	`lagsarlm()`	`lagsarlm( Durbin=TRUE)`

Choosing the right model: a guide

- **Statistical guide** (LM tests):

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - $\rightarrow$ Only LMerr significant $\rightarrow$ **SEM**

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - → Only LMerr significant → **SEM**
  - → Only LMlag significant → **SLM**

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - $\rightarrow$ Only LMerr significant $\rightarrow$ **SEM**
  - $\rightarrow$ Only LMlag significant $\rightarrow$ **SLM**
  - $\rightarrow$ Both significant $\rightarrow$ compare RLMerr vs. RLMlag $\rightarrow$ larger wins; or use SDM

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - → Only LMerr significant → **SEM**
  - → Only LMlag significant → **SLM**
  - → Both significant → compare RLMerr vs. RLMlag → larger wins; or use SDM
- **Conceptual guide** (theory):

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - → Only LMerr significant → **SEM**
  - → Only LMlag significant → **SLM**
  - → Both significant → compare RLMerr vs. RLMlag → larger wins; or use SDM
- **Conceptual guide** (theory):
  - → Is spatial dependence a *nuisance* from omitted variables? → **SEM**

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - → Only LMerr significant → **SEM**
  - → Only LMlag significant → **SLM**
  - → Both significant → compare RLMerr vs. RLMlag → larger wins; or use SDM
- **Conceptual guide** (theory):
  - → Is spatial dependence a *nuisance* from omitted variables? → **SEM**
  - → Is there genuine *diffusion* of the outcome across space? → **SLM / SDM**

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - → Only LMerr significant → **SEM**
  - → Only LMlag significant → **SLM**
  - → Both significant → compare RLMerr vs. RLMlag → larger wins; or use SDM
- **Conceptual guide** (theory):
  - → Is spatial dependence a *nuisance* from omitted variables? → **SEM**
  - → Is there genuine *diffusion* of the outcome across space? → **SLM / SDM**
  - → Does neighbors' x directly affect y? → **SDM**

Choosing the right model: a guide

- **Statistical guide** (LM tests):
  - → Only LMerr significant → **SEM**
  - → Only LMlag significant → **SLM**
  - → Both significant → compare RLMerr vs. RLMlag → larger wins; or use SDM
- **Conceptual guide** (theory):
  - → Is spatial dependence a *nuisance* from omitted variables? → **SEM**
  - → Is there genuine *diffusion* of the outcome across space? → **SLM / SDM**
  - → Does neighbors' x directly affect y? → **SDM**
- **Robustness**: estimate all three, report side by side

The full workflow in R

```
library(sf); library(spdep); library(spatialreg)

# 1.  Build weights
nb = poly2nb(data, queen = TRUE)
listw = nb2listw(nb, style = "W", zero.policy = TRUE)

# 2.  OLS + diagnostics
ols = lm(y ~ x1 + x2, data = data)
moran.test(residuals(ols), listw = listw)
lm.LMtests(ols, listw, test = c("LMerr","LMlag",
     "RLMerr","RLMlag"))

# 3.  Spatial models
sem = errorsarlm(y ~ x1 + x2, data, listw)
slm = lagsarlm(y ~ x1 + x2, data, listw)
impacts(slm, listw = listw)
```

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**
  - $\rightarrow$ Always use `impacts()` for marginal effects

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**
  - $\rightarrow$ Always use `impacts()` for marginal effects
- **Choosing the weights matrix arbitrarily**

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**
  - → Always use `impacts()` for marginal effects
- **Choosing the weights matrix arbitrarily**
  - → Results can be sensitive to queen vs. rook vs. k-NN; report robustness

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**
  $\rightarrow$ Always use `impacts()` for marginal effects
- **Choosing the weights matrix arbitrarily**
  $\rightarrow$ Results can be sensitive to queen vs. rook vs. *k*-NN; report robustness
- **Treating $\hat{\lambda}$ or $\hat{\rho}$ as the quantity of interest**

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**
  $\rightarrow$ Always use `impacts()` for marginal effects
- **Choosing the weights matrix arbitrarily**
  $\rightarrow$ Results can be sensitive to queen vs. rook vs. *k*-NN; report robustness
- **Treating $\hat{\lambda}$ or $\hat{\rho}$ as the quantity of interest**
  $\rightarrow$ These are nuisance (SEM) or structural (SLM) parameters; focus on $\hat{\beta}$ and impacts

Common pitfalls

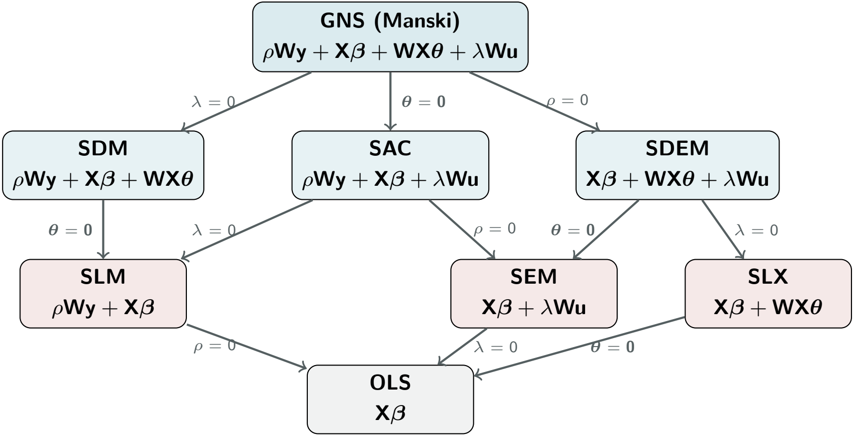
- **Interpreting SLM $\hat{\beta}$ directly**
  - $\rightarrow$ Always use `impacts()` for marginal effects
- **Choosing the weights matrix arbitrarily**
  - $\rightarrow$ Results can be sensitive to queen vs. rook vs. *k*-NN; report robustness
- **Treating $\hat{\lambda}$ or $\hat{\rho}$ as the quantity of interest**
  - $\rightarrow$ These are nuisance (SEM) or structural (SLM) parameters; focus on $\hat{\beta}$ and impacts
- **Skipping the diagnostic step**

Common pitfalls

- **Interpreting SLM $\hat{\beta}$ directly**
  → Always use `impacts()` for marginal effects
- **Choosing the weights matrix arbitrarily**
  → Results can be sensitive to queen vs. rook vs. k-NN; report robustness
- **Treating $\hat{\lambda}$ or $\hat{\rho}$ as the quantity of interest**
  → These are nuisance (SEM) or structural (SLM) parameters; focus on $\hat{\beta}$ and impacts
- **Skipping the diagnostic step**
  → Always run Moran's I on OLS residuals before estimating spatial models

Roadmap

The family of spatial regression models



Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - $\rightarrow$ OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - $\rightarrow$ OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)
  - $\rightarrow$ SEM and SLM are nested in SDM, SAC

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - $\rightarrow$ OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)
  - $\rightarrow$ SEM and SLM are nested in SDM, SAC

- **Likelihood Ratio test**: compares log-likelihoods

$$LR = 2\left[\ell(\text{unrestricted}) - \ell(\text{restricted})\right] \sim \chi^2(q)$$

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - $\rightarrow$ OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)
  - $\rightarrow$ SEM and SLM are nested in SDM, SAC

- **Likelihood Ratio test**: compares log-likelihoods

$$LR = 2\left[\ell(\text{unrestricted}) - \ell(\text{restricted})\right] \sim \chi^2(q)$$

$\rightarrow$ q = number of restrictions (extra parameters in unrestricted model)

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - $\rightarrow$ OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)
  - $\rightarrow$ SEM and SLM are nested in SDM, SAC

- **Likelihood Ratio test**: compares log-likelihoods

$$LR = 2\left[\ell(\text{unrestricted}) - \ell(\text{restricted})\right] \sim \chi^2(q)$$

- $\rightarrow$ q = number of restrictions (extra parameters in unrestricted model)
- $\rightarrow$ Reject H_0: the restricted model is sufficient

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - → OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)
  - → SEM and SLM are nested in SDM, SAC
- **Likelihood Ratio test**: compares log-likelihoods

$$LR = 2\big[\ell(\text{unrestricted}) - \ell(\text{restricted})\big] \sim \chi^2(q)$$

  - → $q =$ number of restrictions (extra parameters in unrestricted model)
  - → Reject H_0: the restricted model is sufficient
- In R: `anova(restricted_fit, unrestricted_fit)`

Comparing nested models: the LR test

- Two models are **nested** if one is a restricted version of the other
  - → OLS is nested in SEM ($\lambda = 0$), SLM ($\rho = 0$), SLX ($\boldsymbol{\theta} = \mathbf{0}$)
  - → SEM and SLM are nested in SDM, SAC
- **Likelihood Ratio test**: compares log-likelihoods

$$LR = 2\left[\ell(\text{unrestricted}) - \ell(\text{restricted})\right] \sim \chi^2(q)$$

  - → $q =$ number of restrictions (extra parameters in unrestricted model)
  - → Reject H_0: the restricted model is sufficient
- In R: `anova(restricted_fit, unrestricted_fit)`
  - → Or compare log-likelihoods manually with `logLik()`

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other
  - $\rightarrow$ Cannot use LR test to compare them directly

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other
  - $\rightarrow$ Cannot use LR test to compare them directly
- Use **AIC** (Akaike Information Criterion):

$$AIC = -2\ell + 2k$$

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other
  - $\rightarrow$ Cannot use LR test to compare them directly
- Use **AIC** (Akaike Information Criterion):

$$AIC = -2\ell + 2k$$

  - $\rightarrow$ ℓ = log-likelihood, k = number of parameters

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other
  - → Cannot use LR test to compare them directly
- Use **AIC** (Akaike Information Criterion):

$$AIC = -2\ell + 2k$$

→ ℓ = log-likelihood, k = number of parameters
→ Lower AIC = better fit (penalized for complexity)

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other
  - $\rightarrow$ Cannot use LR test to compare them directly

- Use **AIC** (Akaike Information Criterion):

$$AIC = -2\ell + 2k$$

  - $\rightarrow$ ℓ = log-likelihood, k = number of parameters
  - $\rightarrow$ Lower AIC = better fit (penalized for complexity)

- In R: `AIC(ols_fit, sem_fit, slm_fit, sdm_fit)`

Comparing non-nested models: AIC

- SEM and SLM are **not nested** in each other
  → Cannot use LR test to compare them directly

- Use **AIC** (Akaike Information Criterion):

$$AIC = -2\ell + 2k$$

  → ℓ = log-likelihood, k = number of parameters
  → Lower AIC = better fit (penalized for complexity)

- In R: `AIC(ols_fit, sem_fit, slm_fit, sdm_fit)`

- **Strategy**: use LR tests along nesting paths; use AIC across non-nested alternatives

The SAC model (SARAR)

Spatial Autoregressive Combined Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{u}, \quad \mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon$$

- Combines spatial lag (ρ) and spatial error (λ) in one model

```
sac_fit = sacsarlm(y ~ x1 + x2,
    data = mydata, listw = my_listw)
```

Useful when both diffusion in y **and** spatially structured errors are present

The SAC model (SARAR)

Spatial Autoregressive Combined Model

$$\mathbf{y} = \rho \mathbf{W}\mathbf{y} + \mathbf{X}\boldsymbol{\beta} + \mathbf{u}, \quad \mathbf{u} = \lambda \mathbf{W}\mathbf{u} + \varepsilon$$

- Combines spatial lag (ρ) and spatial error (λ) in one model
- Nests both SLM ($\lambda = 0$) and SEM ($\rho = 0$)

```
sac_fit = sacsarlm(y ~ x1 + x2,
    data = mydata, listw = my_listw)
```

Useful when both diffusion in *y* **and** spatially structured errors are present

The SAC model (SARAR)

Spatial Autoregressive Combined Model

$$\mathbf{y} = \rho\mathbf{W}\mathbf{y} + \mathbf{X}\beta + \mathbf{u}, \quad \mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \varepsilon$$

- Combines spatial lag (ρ) and spatial error (λ) in one model
- Nests both SLM ($\lambda = 0$) and SEM ($\rho = 0$)
- In R:

```
sac_fit = sacsarlm(y ~ x1 + x2,
   data = mydata, listw = my_listw)
```

Useful when both diffusion in y **and** spatially structured errors are present

The SDEM model

Spatial Durbin Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{W}\mathbf{X}\theta + \mathbf{u}, \quad \mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \varepsilon$$

- Neighbors' covariates ($\mathbf{W}\mathbf{X}\theta$) + spatially correlated errors (λ)

```
sdem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata, listw = my_listw,
    Durbin = TRUE)
```

The SDEM model

Spatial Durbin Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{W}\mathbf{X}\theta + \mathbf{u}, \quad \mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \varepsilon$$

- Neighbors' covariates ($\mathbf{WX}\boldsymbol{\theta}$) + spatially correlated errors (λ)
- **No** spatial lag of $y \Rightarrow$ no endogeneity $\Rightarrow \hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\theta}}$ are marginal effects

```
sdem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata, listw = my_listw,
    Durbin = TRUE)
```

The SDEM model

Spatial Durbin Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{W}\mathbf{X}\theta + \mathbf{u}, \quad \mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \varepsilon$$

- Neighbors' covariates ($\mathbf{WX}\boldsymbol{\theta}$) + spatially correlated errors (λ)
- **No** spatial lag of $y \Rightarrow$ no endogeneity $\Rightarrow \hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\theta}}$ are marginal effects
- Nests SEM ($\boldsymbol{\theta} = \mathbf{0}$) and SLX ($\lambda = 0$)

```
sdem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata, listw = my_listw,
    Durbin = TRUE)
```

The SDEM model

Spatial Durbin Error Model

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{W}\mathbf{X}\theta + \mathbf{u}, \quad \mathbf{u} = \lambda\mathbf{W}\mathbf{u} + \varepsilon$$

- Neighbors' covariates ($\mathbf{WX}\boldsymbol{\theta}$) + spatially correlated errors (λ)
- **No** spatial lag of $y \Rightarrow$ no endogeneity $\Rightarrow \hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\theta}}$ are marginal effects
- Nests SEM ($\boldsymbol{\theta} = \mathbf{0}$) and SLX ($\lambda = 0$)
- In R:

```
sdem_fit = errorsarlm(y ~ x1 + x2,
    data = mydata, listw = my_listw,
    Durbin = TRUE)
```

Summary: nesting and comparison strategy

Comparison	Nested?	Test
OLS vs. SEM	Yes	LR test ($\lambda = 0$)
OLS vs. SLM	Yes	LR test ($\rho = 0$)
SLM vs. SDM	Yes	LR test ($\boldsymbol{\theta} = \mathbf{0}$)
SEM vs. SDM	Yes	LR test ($\boldsymbol{\theta} = -\rho\boldsymbol{\beta}$)
SLM vs. SAC	Yes	LR test ($\lambda = 0$)
SEM vs. SAC	Yes	LR test ($\rho = 0$)
SEM vs. SDEM	Yes	LR test ($\boldsymbol{\theta} = \mathbf{0}$)
SEM vs. SLM	No	AIC / BIC
SDM vs. SAC	No	AIC / BIC
SDM vs. SDEM	No	AIC / BIC

LR test for nested pairs; AIC/BIC across non-nested alternatives

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction
- **Diagnose first**: Moran's I on residuals → LM tests → model choice

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction
- **Diagnose first**: Moran's I on residuals $\rightarrow$ LM tests $\rightarrow$ model choice
- **SEM** (λ): spatial structure is nuisance; $\hat{\beta}$ interpreted as OLS

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction
- **Diagnose first**: Moran's I on residuals $\rightarrow$ LM tests $\rightarrow$ model choice
- **SEM** (λ): spatial structure is nuisance; $\hat{\beta}$ interpreted as OLS
- **SLM** (ρ): genuine spillover in y; use `impacts()` for MEs

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction
- **Diagnose first**: Moran's I on residuals $\rightarrow$ LM tests $\rightarrow$ model choice
- **SEM** (λ): spatial structure is nuisance; $\hat{\beta}$ interpreted as OLS
- **SLM** (ρ): genuine spillover in y; use `impacts()` for MEs
- **SDM**: most flexible; nests SEM and SLM; good conservative choice

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction
- **Diagnose first**: Moran's I on residuals $\rightarrow$ LM tests $\rightarrow$ model choice
- **SEM** (λ): spatial structure is nuisance; $\hat{\beta}$ interpreted as OLS
- **SLM** (ρ): genuine spillover in y; use `impacts()` for MEs
- **SDM**: most flexible; nests SEM and SLM; good conservative choice
- **Extended models**: SAC (lag + error), SDEM (lagged $\mathbf{x}$ + error)

Key takeaways

- **Spatial autocorrelation in OLS residuals**: biases SEs, needs correction
- **Diagnose first**: Moran's I on residuals $\rightarrow$ LM tests $\rightarrow$ model choice
- **SEM** (λ): spatial structure is nuisance; $\hat{\beta}$ interpreted as OLS
- **SLM** (ρ): genuine spillover in y; use `impacts()` for MEs
- **SDM**: most flexible; nests SEM and SLM; good conservative choice
- **Extended models**: SAC (lag + error), SDEM (lagged $\mathbf{x}$ + error)
- **Comparison**: LR test for nested models; AIC/BIC for non-nested pairs

Roadmap

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:
  - $\rightarrow$ Add unit fixed effects and a spatial lag $\rightarrow$ estimate with `plm` + `lagsarlm`?

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:
  - $\rightarrow$ Add unit fixed effects and a spatial lag $\rightarrow$ estimate with `plm` + `lagsarlm`?
- This does not work properly:

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:
  - $\rightarrow$ Add unit fixed effects and a spatial lag $\rightarrow$ estimate with `plm` + `lagsarlm`?
- This does not work properly:
  - $\rightarrow$ Fixed effects with a spatial lag create **incidental parameters** bias

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:
  - $\rightarrow$ Add unit fixed effects and a spatial lag $\rightarrow$ estimate with `plm` + `lagsarlm`?
- This does not work properly:
  - $\rightarrow$ Fixed effects with a spatial lag create **incidental parameters** bias
  - $\rightarrow$ Time dependence and spatial dependence interact

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:
  - $\rightarrow$ Add unit fixed effects and a spatial lag $\rightarrow$ estimate with `plm` + `lagsarlm`?
- This does not work properly:
  - $\rightarrow$ Fixed effects with a spatial lag create **incidental parameters** bias
  - $\rightarrow$ Time dependence and spatial dependence interact
  - $\rightarrow$ Standard cross-sectional ML ignores the time dimension

What if you have panel data?

- All models so far assume a **single cross-section** (n units, one time period)
- With panel data (n units $\times$ T periods), a natural instinct:
  - $\rightarrow$ Add unit fixed effects and a spatial lag $\rightarrow$ estimate with `plm` + `lagsarlm`?
- This does not work properly:
  - $\rightarrow$ Fixed effects with a spatial lag create **incidental parameters** bias
  - $\rightarrow$ Time dependence and spatial dependence interact
  - $\rightarrow$ Standard cross-sectional ML ignores the time dimension
- Dedicated **spatial panel models** handle both dimensions jointly

Spatial panel models: where to look

- R package `splm` (Millo & Piras):

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  $\rightarrow$ Spatial panel models with fixed and random effects

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  - → Spatial panel models with fixed and random effects
  - → ML and GM estimation for SEM, SLM, SAC with panel structure

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  - → Spatial panel models with fixed and random effects
  - → ML and GM estimation for SEM, SLM, SAC with panel structure
  - → `spml()`, `spgm()` functions

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  - → Spatial panel models with fixed and random effects
  - → ML and GM estimation for SEM, SLM, SAC with panel structure
  - → `spml()`, `spgm()` functions
- R package `plm` + `spatialreg`:

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  - → Spatial panel models with fixed and random effects
  - → ML and GM estimation for SEM, SLM, SAC with panel structure
  - → `spml()`, `spgm()` functions
- R package `plm` + `spatialreg`:
  - → Some integration, but limited compared to `splm`

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  - → Spatial panel models with fixed and random effects
  - → ML and GM estimation for SEM, SLM, SAC with panel structure
  - → `spml()`, `spgm()` functions
- R package `plm` + `spatialreg`:
  - → Some integration, but limited compared to `splm`
- Key reference: Elhorst (2014), *Spatial Econometrics: From Cross-Sectional Data to Spatial Panels*

Spatial panel models: where to look

- R package `splm` (Millo & Piras):
  - → Spatial panel models with fixed and random effects
  - → ML and GM estimation for SEM, SLM, SAC with panel structure
  - → `spml()`, `spgm()` functions
- R package `plm` + `spatialreg`:
  - → Some integration, but limited compared to `splm`
- Key reference: Elhorst (2014), *Spatial Econometrics: From Cross-Sectional Data to Spatial Panels*
- **Bottom line**: if your data has a spatial *and* temporal dimension, use purpose-built tools — do not improvise with cross-sectional spatial models

For the lab session

- Complete the diagnostics workflow on a provided dataset
  - → Run OLS; compute Moran's I on residuals
  - → Run LM tests; select the appropriate model
- Estimate SEM and SLM; compare with OLS
  - → Interpret $\hat{\lambda}$ and $\hat{\rho}$
  - → Use `impacts()` for SLM marginal effects
- Reflect: what mechanism drives spatial dependence in your data?

Questions?