Local variable selection and parameter estimation for spatially varying coefficient models

Wesley Brooks

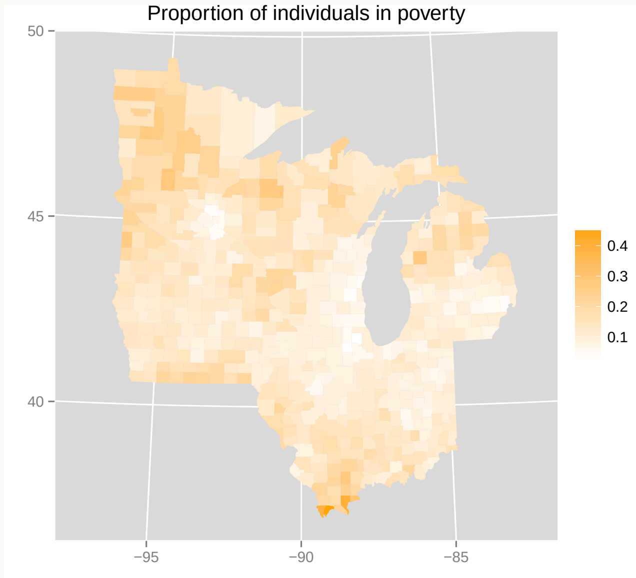
Department of Statistics
University of Wisconsin–Madison

`somesquares.org`

These slides were prepared for a practice version of my preliminary exam to advance to Ph.D candidacy in statistics at the University of Wisconsin–Madison.

Motivation

Take a look at some data

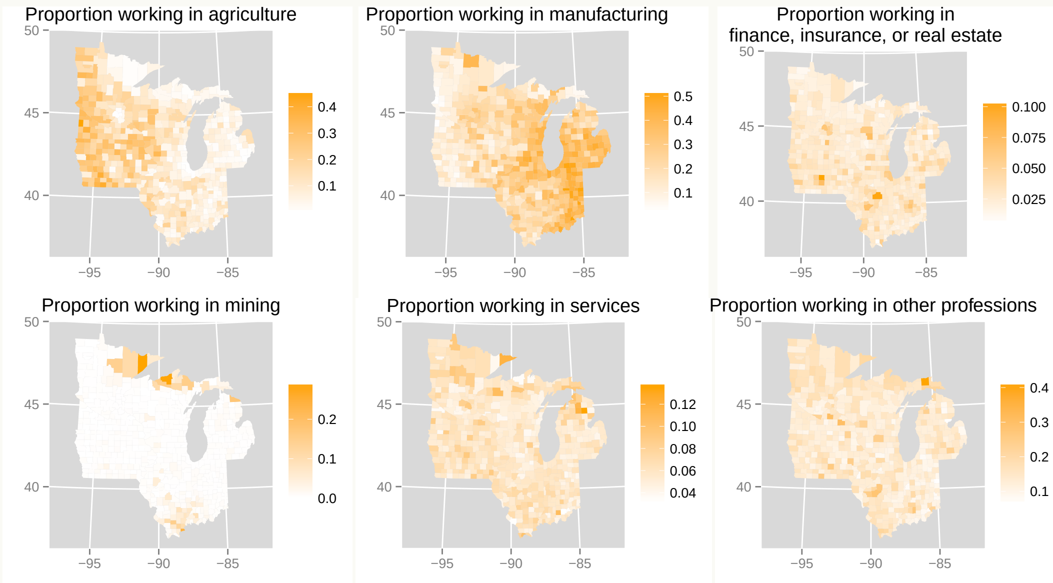


This is the county-level poverty rate from 1970, as well as the proportion of people who worked in manufacturing, agriculture, and services.

How is this data to be analyzed?

Motivation

Take a look at some data



This is the county-level poverty rate from 1970, as well as the proportion of people who worked in manufacturing, agriculture, and services.

How is this data to be analyzed?

Motivation

Sensible questions about the data

- Which of the economic-structure variables is associated with poverty rate?
- What are the sign and magnitude of that association?
- Is poverty rate associated with the same economic-structure variables across the entire region?
- Are the sign and magnitude of the associations constant across the region?

We're going to aim at answering these questions with the work I present today.

There are several other methods to answer at least some of these questions, which we'll cover next.

Introduction

A review of existing methods

- Spatial regression
- Varying coefficient regression
  - Splines
  - Kernels
  - Wavelets
- Model selection via regularization

Behind the methodology that I'm discussing is a wide range of literature.

Introduction

Some definitions

- Univariate spatial response process $\{Y(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$
- Multivariate spatial covariate process $\{\boldsymbol{X}(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$
- $n =$ number of observations
- $p =$ number of covariates
- Location (2-dimensional) $\boldsymbol{s}$
- Spatial domain $\mathcal{D}$

We'll use these variables throughout.

Introduction

Further definitions

- Geostatistical data:
  - Observations are made at sampling locations $\boldsymbol{s}_i$ for $i = 1, \ldots, n$
  - E.g. elevation, temperature
- Areal data:
  - Domain is partitioned into n regions $\{D_1, \ldots, D_n\}$
  - The regions do not overlap, and they divide the domain completely: $\mathcal{D} = \bigcup_{i=1}^{n} D_i$
  - Sampling locations $\boldsymbol{s}_i$ for $i = 1, \ldots, n$ are the centroids of the regions
  - E.g. poverty rate, population, spatial mean temperature

The method I'm describing applies to geostatistical data, or to areal data when the observations are assumed to be located at the centroid.

The poverty data example is areal data, the simulation study is based on simulated geostatistical data.

Introduction

Existing approaches: spatial regression

- The typical spatial regression (**?**)

$$
\begin{aligned}
Y(\boldsymbol{s}) &= \boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta} + W(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \\
\text{cov}(W(\boldsymbol{s}), W(\boldsymbol{t})) &= \Gamma\left(\delta(\boldsymbol{s}, \boldsymbol{t})\right) \\
\delta(\boldsymbol{s}, \boldsymbol{t}) &= \sqrt{\|\boldsymbol{t} - \boldsymbol{s}\|_2} \\
\text{E.g. } \Gamma(\delta(\boldsymbol{s}, \boldsymbol{t})) &= \text{exp}\{-\phi^{-1}\delta(\boldsymbol{s}, \boldsymbol{t})\}
\end{aligned} \tag{1}
$$

- $W(\boldsymbol{s})$ is a spatial random effect that accounts for autocorrelation in the response variable
- The coefficients $\boldsymbol{\beta}$ are constant
- Relies on *a priori* global variable selection

This is the form of the usual spatial regression from e.g. Cressie (1993).

The spatial random effect W captures autocorrelation of the response, while the white noise is iid error

The Gamma function is a Matern-class covariance function, such as the exponential covariance function (listed here)

Introduction

Existing approaches: varying coefficients regression

$$Y(\boldsymbol{s}) = \boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta}(\boldsymbol{s}) + \varepsilon(\boldsymbol{s})$$

- Assume an effect modifying variable S
- Coefficients are functions of S
- Generally assume that the coefficient functions are smooth
- This is a varying coefficient regression (VCR) model (**?**)
- If $\boldsymbol{s}$ is a spatial location then we have a spatially varying coefficient regression (SVCR) model

Introduction

Existing approaches: spatially varying coefficient process

- Making model more flexible: coefficients in a spatial regression model can be allowed to vary (**?**)

$$Y(\boldsymbol{s}) = \boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta}(\boldsymbol{s}) + \varepsilon(\boldsymbol{s})$$

- The spatial random effect has been incorporated into the spatially varying intercept
- $\{\beta_1(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}, \ldots, \{\beta_p(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ are stationary spatial processes with Matèrn covariance functions
- Still relies on *a priori* global variable selection

Introduction

Existing approaches: spline-based VCR and SVCR models

- Splines are a way to parameterize smooth functions
- Splines can be incorporated into a generalized additive model (GAM):
  - $E\{Y(t)\} = f\{X_1(t)\} + \cdots + f\{X_p(t)\}$
- It is possible to parameterize a VCR model with splines for the coefficient functions:
  - $E\{Y(t)\} = \beta_1(t)X_1(t) + \cdots + \beta_p(t)X_p(t)$

Introduction

Existing approaches: Global selection in spline-based VCR models

Regularization methods for global variable selection in VCR models:

- The integral of a function squared (e.g. $\int \{f(t)\}^2 dt$) is zero if and only if the function is zero everywhere.
- Use regularization (maximize the likelihood plus a penalty) to encourage coefficient functions to be zero
- SCAD penalty (**?**) on the integral of the square of the coefficient function (**?**)
- Non-negative garrote penalty (**?**) on the integral of the square of the coefficient function (**?**)

These selection methods are all global - that is, they select variables for the entire domain simultaneously

Introduction

Existing approaches: wavelet methods for VCR models

Wavelet methods involve decomposing a function into local frequency components. Wavelet methods for VCR models include using Bayesian variable selection or the Lasso to estimate which local frequency components have nonzero coefficients (**??**).

These methods achieve sparsity in the local frequency components but not in the local covariates, and so are not suitable for local variable selection.

Introduction

Existing approaches: Local regression

Local regression uses a kernel function at each sampling location to weight observations based on their distance from the sampling location. An example is the bisquare kernel:

$$w_{ii'} = \begin{cases} \left[1 - (\phi^{-1}\delta_{ii'})^2\right]^2 & \text{if } \delta_{ii'} < \phi, \\ 0 & \text{if } \delta_{ii'} \geq \phi. \end{cases} \tag{2}$$

Where ϕ is a bandwidth parameter.
Given the weights, a local model is fit at each sampling location using the local likelihood (**?**)

Introduction

Existing approaches: Local likelihood

Calibrate the model by doing the following at each sampling location:

- Weight each observation's likelihood
- Weights are given by the kernel

$$L = \prod_{i'=1}^{n} (L_{i'})^{w_{ii'}}$$

$$\ell = \sum_{i'=1}^{n} w_{ii'} \left\{ \log \sigma_i^2 + \sigma_i^{-2} \left(y_{i'} - \boldsymbol{x}'_{i'} \boldsymbol{\beta}_i \right)^2 \right\}$$

Where $\boldsymbol{\beta}_i = \boldsymbol{\beta}(\boldsymbol{s}_i)$.

Maximizing the local likelihood for a model of Gaussian data with iid errors can be done by weighted least squares.

Introduction

Existing approaches: geographically weighted regression

When the effect modifying variable s refers to spatial location, the method of local regression is called geographically weighted regression (GWR) (**??**)

Maximizing the local likelihood for a model of Gaussian data with iid errors can be done by weighted least squares.

Introduction

Existing approaches: bandwidth estimation for GWR

- Smaller bandwidth: less bias, more flexible coefficient surface
- Large bandwidth: less variance, less flexible coefficient surface
- Estimate the degrees of freedom used in estmating the coefficient surface (**?**):
  - $\hat{\boldsymbol{y}} = H\boldsymbol{y}$
  - $\nu = \text{tr}(H)$
- Then the corrected AIC for bandwidth selection is:
- $\text{AIC}_\text{c} = 2n \log \sigma + n \left\{ \frac{n+\nu}{n-2-\nu} \right\}$

Maximizing the local likelihood for a model of Gaussian data with iid errors can be done by weighted least squares.

Introduction

Existing approaches: geographically weighted Lasso

Within a GWR model, using the Lasso (**?**) for local variable selection is called the geographically weighted Lasso (GWL) (**?**).

- The GWL requires estimating a Lasso tuning parameter for each local model
- **?** estimates the local Lasso tuning parameter at location s_i by minimizing a jacknife criterion: $|y_i - \hat{y}_i|$
- The jacknife criterion can only be calculated where data are observed, making it impossible to use the GWL to impute missing data or to estimate the value of the coefficient surface at new locations
- Also, the Lasso is known to be biased in variable selection and suboptimal for coefficient estimation

GWL does local variable selection

Methodology

Geographically weighted elastic net (GWEN)

- Local variable selection in a GWR model using the adaptive elastic net (AEN) (**?**)
- Under suitable conditions, the AEN has an oracle property for selection

$$
\begin{aligned}
\mathcal{S}(\boldsymbol{\beta}_i) &= -2\ell_i(\boldsymbol{\beta}_i) + \mathcal{J}_2(\boldsymbol{\beta}_i) \\
&= \sum_{i'=1}^{n} w_{ii'} \left\{ \log \sigma_i^2 + \left(\sigma_i^2\right)^{-1} \left(y_{i'} - \boldsymbol{x}_{i'}' \boldsymbol{\beta}_i\right)^2 \right\} \\
&+ \alpha_i \lambda_i \sum_{j=1}^{p} |\beta_{ij}| / \gamma_{ij} \\
&+ (1-\alpha_i) \lambda_i^* \sum_{j=1}^{p} \left(\beta_{ij}/\gamma_{ij}\right)^2
\end{aligned}
$$

The adaptive weights $\boldsymbol{\gamma}_i = (\gamma_{i1}, \ldots, \gamma_{ip})'$ are defined in the same way as for the AL, and the elastic net parameter $\alpha_i \in [0, 1]$ controls the balance between ℓ_1 penalty $\lambda_i^* \sum\limits_{j=1}^{p} |\beta_{ij}|/\gamma_{ij}$ and ℓ_2 penalty $\lambda_i^* \sum_{j=1}^{p} (\beta_{ij}/\gamma_{ij})^2$.

Methodology

Geographically weighted elastic net (GWEN)

where $\sum_{i'=1}^{n} w_{ii'} (y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i)^2$ is the weighted sum of squares minimized by traditional GWR, and $\mathcal{J}_2(\boldsymbol{\beta}_i) = \alpha_i \lambda_i^* \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij} + (1 - \alpha_i) \lambda_i^* \sum_{j=1}^{p} (\beta_{ij}/\gamma_{ij})^2$ is the AEN penalty.

The adaptive weights $\boldsymbol{\gamma}_i = (\gamma_{i1}, \ldots, \gamma_{ip})'$ are defined in the same way as for the AL, and the elastic net parameter $\alpha_i \in [0, 1]$ controls the balance between ℓ_1 penalty $\lambda_i^* \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij}$ and ℓ_2 penalty $\lambda_i^* \sum_{j=1}^{p} (\beta_{ij}/\gamma_{ij})^2$.

Methodology

Geographically weighted elastic net (GWEN)

It is necessary to estimate an AEN tuning parameter for each local model. Using the local BIC allows fitting a local model at any location within the domain

$$\begin{aligned}
\mathsf{BIC}_{\mathsf{loc},i} &= -2\sum_{i'=1}^{n} \ell_{ii'} + \log\left(\sum_{i'=1}^{n} w_{ii'}\right)\mathsf{df}_i \\
&= -2\sum_{i'=1}^{n} \log\left[\left(2\pi\hat{\sigma}_i^2\right)^{-1/2} \exp\left\{-\frac{1}{2}\hat{\sigma}_i^{-2}\left(y_{i'} - \boldsymbol{x}_{i'}'\hat{\boldsymbol{\beta}}_{i'}\right)^2\right\}\right]^{w} \\
&\quad + \log\left(\sum_{i'=1}^{n} w_{ii'}\right)\mathsf{df}_i
\end{aligned} \tag{3}$$

The adaptive weights $\boldsymbol{\gamma}_i = (\gamma_{i1}, \ldots, \gamma_{ip})'$ are defined in the same way as for the AL, and the elastic net parameter $\alpha_i \in [0, 1]$ controls the balance between ℓ_1 penalty $\lambda_i^* \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij}$ and ℓ_2 penalty $\lambda_i^* \sum_{j=1}^{p} (\beta_{ij}/\gamma_{ij})^2$.

Methodology

Geographically weighted elastic net (GWEN)

$$= \sum_{i'=1}^{n} w_{ii'} \left\{ \log(2\pi) + \log \hat{\sigma}_i^2 + \hat{\sigma}_i^{-2} \left(y_{i'} - \boldsymbol{x}_{i'}' \hat{\boldsymbol{\beta}}_{i'} \right)^2 \right\}$$

$$+ \log \left(\sum_{i'=1}^{n} w_{ii'} \right) \mathrm{df}_i$$

We treat the sum of the weights around the sampling location as the number of observations for the local BIC.

Simulation study

Simulating covariates

Five covariates $\tilde{X}_1, \ldots, \tilde{X}_5$ were simulated by Gaussian random fields on the domain $[0,1] \times [0,1]$ on a 30×30 grid of sampling locations:

$$\tilde{X}_j \sim N(0, \Sigma) \text{ for } j = 1, \ldots, 5$$
$$\{\Sigma\}_{i,i'} = \exp\{-\tau^{-1}\delta_{ii'}\} \text{ for } i, i' = 1, \ldots, n$$

Where the covariates were simulated with colinearity, the colinearity was induced by multiplying the design matrix by the square root of the colinearity matrix:

$$\begin{aligned} \text{diag}(\Omega_{5\times 5}) &= 1 \\ \text{off-diag}(\Omega_{5\times 5}) &= \rho \\ X &= \tilde{X}R \end{aligned} \tag{4}$$

Where $\Omega_{5\times 5} = R'R$ is the Cholesky decomposition.

Simulation study

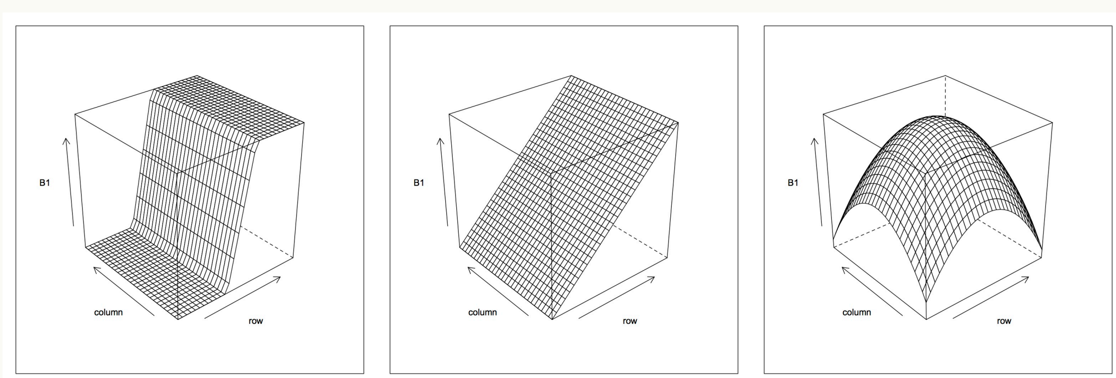
Simulating the response

- $Y(\boldsymbol{s}) = X(\boldsymbol{s})'\boldsymbol{\beta}(\boldsymbol{s}) = \sum_{j=1}^{5} \beta_j(\boldsymbol{s}) X_j(\boldsymbol{s}) + \varepsilon(\boldsymbol{s})$
- $\boldsymbol{\varepsilon} \sim \ iid \ \ N(0, \sigma^2)$
- $\beta_1(\boldsymbol{s})$, the coefficient function for X_1, is nonzero in part of the domain.
- Coefficients for $X_2, \ldots, X_5$ are zero everywhere

Simulation study

Coefficient functions

Call these functions step, gradient, and parabola:



Simulation study

Simulation settings

Setting	function	ρ	σ^2
1	step	0	0.25
2	step	0	1
3	step	0.5	0.25
4	step	0.5	1
5	gradient	0	0.25
6	gradient	0	1
7	gradient	0.5	0.25
8	gradient	0.5	1
9	parabola	0	0.25
10	parabola	0	1
11	parabola	0.5	0.25
12	parabola	0.5	1

Table : Simulation parameters for each setting.

Simulation results

Selection

	step				gradient				
	enet		lasso		enet		lasso		
location	β_1	β_2 - β_5	β_1	β_2 - β_5	β_1	β_2 - β_5	β_1	β_2 - β_5	β
1	0.99	0.00	0.99	0.00	1.00	0.00	1.00	0.00	0.3
	0.99	0.02	0.99	0.02	1.00	0.01	1.00	0.01	0.7
	0.99	0.00	1.00	0.00	1.00	0.00	1.00	0.00	0.2
	0.96	0.05	0.91	0.04	0.99	0.03	0.99	0.01	0.5
2	1.00	0.00	1.00	0.00	1.00	0.01	1.00	0.00	1.0
	1.00	0.03	1.00	0.03	1.00	0.02	1.00	0.02	1.0
	1.00	0.01	1.00	0.00	1.00	0.01	1.00	0.01	1.0
	0.99	0.05	0.97	0.04	1.00	0.02	0.99	0.01	0.9
3	0.91	0.01	0.91	0.00	1.00	0.01	1.00	0.01	1.0
	0.96	0.05	0.96	0.05	1.00	0.01	1.00	0.01	1.0
	0.92	0.05	0.95	0.02	1.00	0.02	1.00	0.01	1.0
	0.92	0.08	0.87	0.05	1.00	0.02	0.98	0.02	0.9
	0.48	0.01	0.43	0.01	1.00	0.01	1.00	0.01	1.0

Simulation results

MSE of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	0.026	*0.025*	0.057	0.057	0.062	**0.008**
		0.042	*0.040*	0.193	0.180	0.102	**0.016**
		0.036	**0.014**	0.055	0.067	0.080	*0.016*
		0.093	0.130	0.230	0.285	0.144	**0.030**
	2	0.063	0.058	0.043	*0.043*	**0.038**	0.055
		0.087	0.084	**0.064**	*0.064*	0.073	0.084
		0.068	0.049	0.045	*0.040*	**0.036**	0.052
		0.140	0.128	*0.082*	0.093	**0.074**	0.096
	3	0.025	0.025	0.019	0.019	**0.004**	*0.010*
		0.021	0.021	0.015	0.015	**0.007**	*0.011*
		0.027	0.021	0.018	*0.014*	**0.006**	0.019
		0.027	0.038	0.020	0.031	**0.007**	*0.016*
	4	*0.026*	0.026	0.028	**0.025**	0.034	0.054
		0.046	*0.050*	0.054	0.057	0.073	0.081
		0.025	0.030	0.030	*0.027*	0.036	0.063
		0.035	*0.036*	0.043	0.046	0.072	0.083
	5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.008
		0.002	0.002	0.001	*0.000*	**0.000**	0.014
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.021

Simulation results

MSE of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	0.026	*0.025*	0.057	0.057	0.062	**0.008**
		0.042	*0.040*	0.193	0.180	0.102	**0.016**
		0.036	**0.014**	0.055	0.067	0.080	*0.016*
		0.093	0.130	0.230	0.285	0.144	**0.030**
	2	0.063	0.058	0.043	*0.043*	**0.038**	0.055
		0.087	0.084	**0.064**	*0.064*	0.073	0.084
		0.068	0.049	0.045	*0.040*	**0.036**	0.052
		0.140	0.128	*0.082*	0.093	**0.074**	0.096
	3	0.025	0.025	0.019	0.019	**0.004**	*0.010*
		0.021	0.021	0.015	0.015	**0.007**	*0.011*
		0.027	0.021	0.018	*0.014*	**0.006**	0.019
		0.027	0.038	0.020	0.031	**0.007**	*0.016*
	4	*0.026*	0.026	0.028	**0.025**	0.034	0.054
		0.046	*0.050*	0.054	0.057	0.073	0.081
		0.025	0.030	0.030	*0.027*	0.036	0.063
		0.035	*0.036*	0.043	0.046	0.072	0.083
	5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.008
		0.002	0.002	0.001	*0.000*	**0.000**	0.014
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.021

Simulation results

MSE of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
		0.026	*0.025*	0.057	0.057	0.062	**0.008**
	1	0.042	*0.040*	0.193	0.180	0.102	**0.016**
		0.036	**0.014**	0.055	0.067	0.080	*0.016*
		0.093	0.130	0.230	0.285	0.144	**0.030**
		0.063	0.058	0.043	*0.043*	**0.038**	0.055
	2	0.087	0.084	**0.064**	*0.064*	0.073	0.084
		0.068	0.049	0.045	*0.040*	**0.036**	0.052
		0.140	0.128	*0.082*	0.093	**0.074**	0.096
		0.025	0.025	0.019	0.019	**0.004**	*0.010*
step	3	0.021	0.021	0.015	0.015	**0.007**	*0.011*
		0.027	0.021	0.018	*0.014*	**0.006**	0.019
		0.027	0.038	0.020	0.031	**0.007**	*0.016*
		0.026	0.026	0.028	**0.025**	0.034	0.054
	4	**0.046**	*0.050*	0.054	0.057	0.073	0.081
		0.025	0.030	0.030	*0.027*	0.036	0.063
		0.035	*0.036*	0.043	0.046	0.072	0.083
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.008
	5	0.002	0.002	0.001	*0.000*	**0.000**	0.014
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.021

Simulation results

Variance of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
		0.023	*0.023*	0.057	0.058	0.063	**0.009**
	1	*0.036*	0.036	0.195	0.180	0.098	**0.016**
		0.027	**0.013**	0.056	0.068	0.080	*0.016*
		0.059	0.100	0.226	0.276	0.145	**0.030**
		0.014	0.013	**0.006**	*0.006*	0.006	0.008
	2	0.017	0.017	0.011	*0.011*	**0.008**	0.013
		0.012	0.010	*0.005*	**0.004**	0.006	0.010
		0.021	0.033	0.021	0.037	**0.008**	*0.014*
		0.022	0.023	0.019	0.019	**0.004**	*0.009*
step	3	0.021	0.021	0.014	0.014	**0.005**	*0.008*
		0.024	0.021	0.018	*0.014*	**0.005**	0.016
		0.024	0.036	0.020	0.032	**0.005**	*0.014*
		0.022	0.023	0.023	0.022	**0.006**	*0.007*
	4	0.025	0.024	0.025	0.022	**0.006**	*0.008*
		0.021	0.025	0.024	0.023	**0.005**	*0.013*
		0.026	0.027	0.029	0.032	**0.009**	*0.015*
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.007
	5	0.002	0.002	0.001	*0.000*	**0.000**	0.014
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.021

Simulation results

Variance of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	0.023	*0.023*	0.057	0.058	0.063	**0.009**
		0.036	0.036	0.195	0.180	0.098	**0.016**
		0.027	**0.013**	0.056	0.068	0.080	*0.016*
		0.059	0.100	0.226	0.276	0.145	**0.030**
	2	0.014	0.013	**0.006**	*0.006*	0.006	0.008
		0.017	0.017	0.011	*0.011*	**0.008**	0.013
		0.012	0.010	*0.005*	**0.004**	0.006	0.010
		0.021	0.033	0.021	0.037	**0.008**	*0.014*
	3	0.022	0.023	0.019	0.019	**0.004**	*0.009*
		0.021	0.021	0.014	0.014	**0.005**	*0.008*
		0.024	0.021	0.018	*0.014*	**0.005**	0.016
		0.024	0.036	0.020	0.032	**0.005**	*0.014*
	4	0.022	0.023	0.023	0.022	**0.006**	*0.007*
		0.025	0.024	0.025	0.022	**0.006**	*0.008*
		0.021	0.025	0.024	0.023	**0.005**	*0.013*
		0.026	0.027	0.029	0.032	**0.009**	*0.015*
	5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.007
		0.002	0.002	0.001	*0.000*	**0.000**	0.014
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.021

Simulation results

Variance of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	0.023	*0.023*	0.057	0.058	0.063	**0.009**
		0.036	0.036	0.195	0.180	0.098	**0.016**
		0.027	**0.013**	0.056	0.068	0.080	*0.016*
		0.059	0.100	0.226	0.276	0.145	**0.030**
	2	0.014	0.013	**0.006**	*0.006*	0.006	0.008
		0.017	0.017	0.011	*0.011*	**0.008**	0.013
		0.012	0.010	*0.005*	**0.004**	0.006	0.010
		0.021	0.033	0.021	0.037	**0.008**	*0.014*
	3	0.022	0.023	0.019	0.019	**0.004**	*0.009*
		0.021	0.021	0.014	0.014	**0.005**	*0.008*
		0.024	0.021	0.018	*0.014*	**0.005**	0.016
		0.024	0.036	0.020	0.032	**0.005**	*0.014*
	4	0.022	0.023	0.023	0.022	**0.006**	*0.007*
		0.025	0.024	0.025	0.022	**0.006**	*0.008*
		0.021	0.025	0.024	0.023	**0.005**	*0.013*
		0.026	0.027	0.029	0.032	**0.009**	*0.015*
	5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.007
		0.002	0.002	0.001	*0.000*	**0.000**	0.014
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.021

Simulation results

Bias of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	-0.056	-0.049	**0.001**	*0.005*	0.015	-0.007
		-0.080	-0.069	*0.020*	0.040	0.072	**0.002**
		-0.093	-0.037	-0.010	-0.009	*-0.005*	**0.003**
		-0.185	-0.177	-0.075	-0.110	*0.032*	**-0.009**
	2	-0.222	-0.213	-0.193	*-0.191*	**-0.178**	-0.217
		-0.264	-0.259	**-0.231**	*-0.232*	-0.256	-0.268
		-0.236	-0.197	-0.199	*-0.188*	**-0.176**	-0.204
		-0.345	-0.309	*-0.248*	**-0.236**	-0.257	-0.286
	3	-0.057	-0.047	*-0.006*	**-0.006**	0.025	0.024
		-0.009	**0.004**	0.022	0.024	0.047	0.051
		-0.052	*-0.007*	**0.003**	0.020	0.039	0.055
		-0.062	-0.046	**-0.011**	*-0.014*	0.046	0.046
	4	0.066	*0.058*	0.071	**0.057**	0.168	0.218
		0.147	*0.165*	0.170	0.188	0.260	0.272
		0.062	0.071	0.077	*0.067*	0.174	0.223
		0.098	**0.098**	0.121	0.119	0.250	0.262
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	-0.022
		0.003	*0.001*	-0.005	-0.003	**0.000**	-0.018

Simulation results

Bias of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	-0.056	-0.049	**0.001**	*0.005*	0.015	-0.007
		-0.080	-0.069	*0.020*	0.040	0.072	**0.002**
		-0.093	-0.037	-0.010	-0.009	*-0.005*	**0.003**
		-0.185	-0.177	-0.075	-0.110	*0.032*	**-0.009**
	2	-0.222	-0.213	-0.193	*-0.191*	**-0.178**	-0.217
		-0.264	-0.259	**-0.231**	*-0.232*	-0.256	-0.268
		-0.236	-0.197	-0.199	*-0.188*	**-0.176**	-0.204
		-0.345	-0.309	*-0.248*	**-0.236**	-0.257	-0.286
	3	-0.057	-0.047	*-0.006*	**-0.006**	0.025	0.024
		-0.009	**0.004**	0.022	0.024	0.047	0.051
		-0.052	*-0.007*	**0.003**	0.020	0.039	0.055
		-0.062	-0.046	**-0.011**	*-0.014*	0.046	0.046
	4	0.066	*0.058*	0.071	**0.057**	0.168	0.218
		0.147	*0.165*	0.170	0.188	0.260	0.272
		0.062	0.071	0.077	*0.067*	0.174	0.223
		0.098	**0.098**	0.121	0.119	0.250	0.262
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	-0.022
		0.003	*0.001*	-0.005	-0.003	**0.000**	-0.018

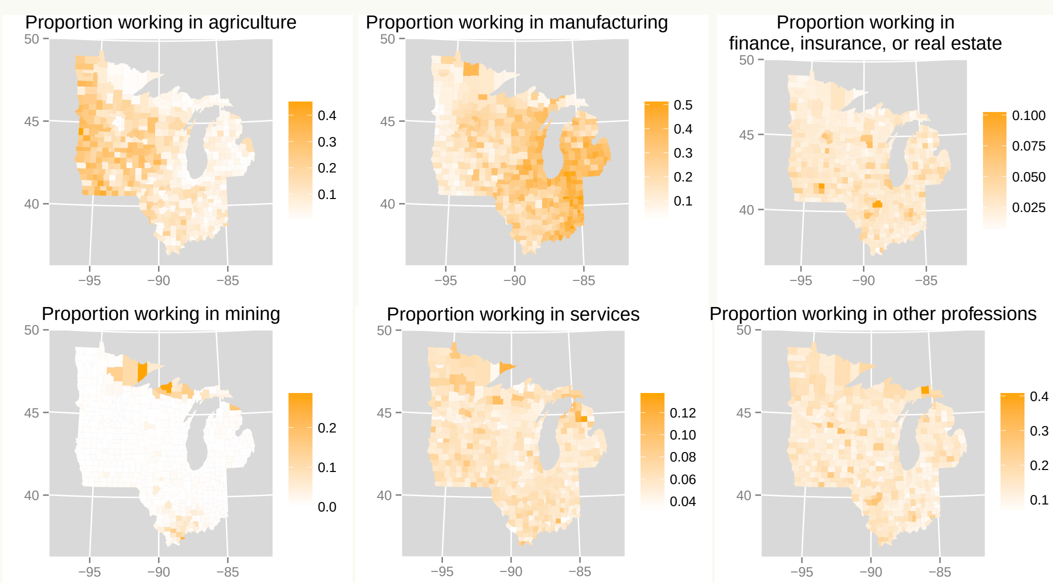
Simulation results

Bias of $\beta_1(\boldsymbol{s})$

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
		-0.056	-0.049	**0.001**	*0.005*	0.015	-0.007
	1	-0.080	-0.069	*0.020*	0.040	0.072	**0.002**
		-0.093	-0.037	-0.010	-0.009	*-0.005*	**0.003**
		-0.185	-0.177	-0.075	-0.110	*0.032*	**-0.009**
		-0.222	-0.213	-0.193	*-0.191*	**-0.178**	-0.217
	2	-0.264	-0.259	**-0.231**	*-0.232*	-0.256	-0.268
		-0.236	-0.197	-0.199	*-0.188*	**-0.176**	-0.204
		-0.345	-0.309	*-0.248*	**-0.236**	-0.257	-0.286
		-0.057	-0.047	*-0.006*	**-0.006**	0.025	0.024
step	3	*-0.009*	**0.004**	0.022	0.024	0.047	0.051
		-0.052	*-0.007*	**0.003**	0.020	0.039	0.055
		-0.062	-0.046	**-0.011**	*-0.014*	0.046	0.046
		0.066	*0.058*	0.071	**0.057**	0.168	0.218
	4	**0.147**	*0.165*	0.170	0.188	0.260	0.272
		0.062	0.071	0.077	*0.067*	0.174	0.223
		0.098	**0.098**	0.121	0.119	0.250	0.262
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	-0.022
		0.003	*0.001*	-0.005	-0.003	**0.000**	-0.018

Data analysis

Revisiting the introductory example

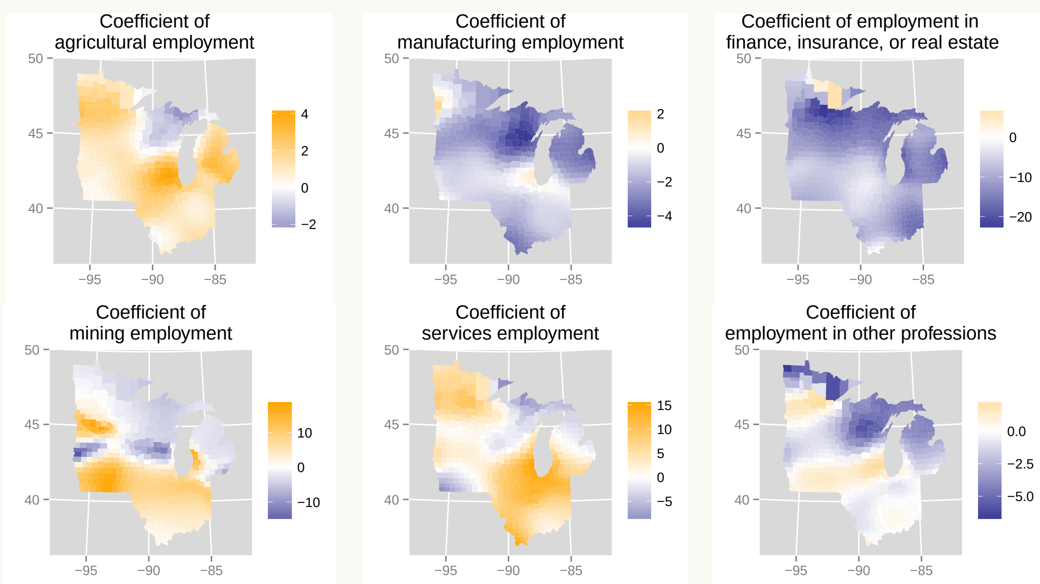


This is the county-level poverty rate from 1970, as well as the proportion of people who worked in manufacturing, agriculture, and services.

How is this data to be analyzed?

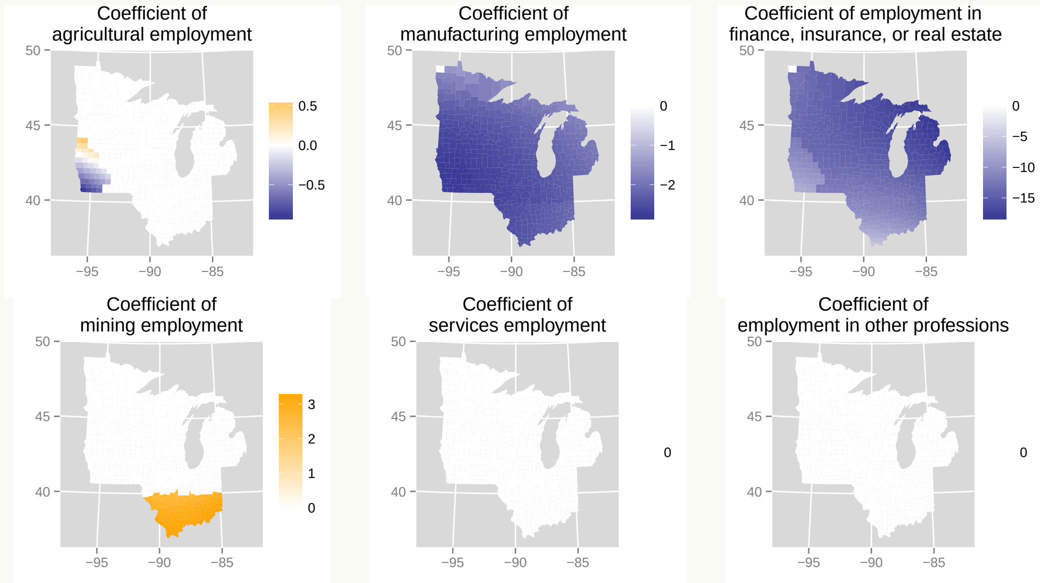
Data analysis

Results from traditional GWR



Data analysis

Results from GWEN



Future work

- Apply the GWEN to data with non-Gaussian response variable
- Incorporate spatial autocorrelation in the model (simulated errors were iid)

References I