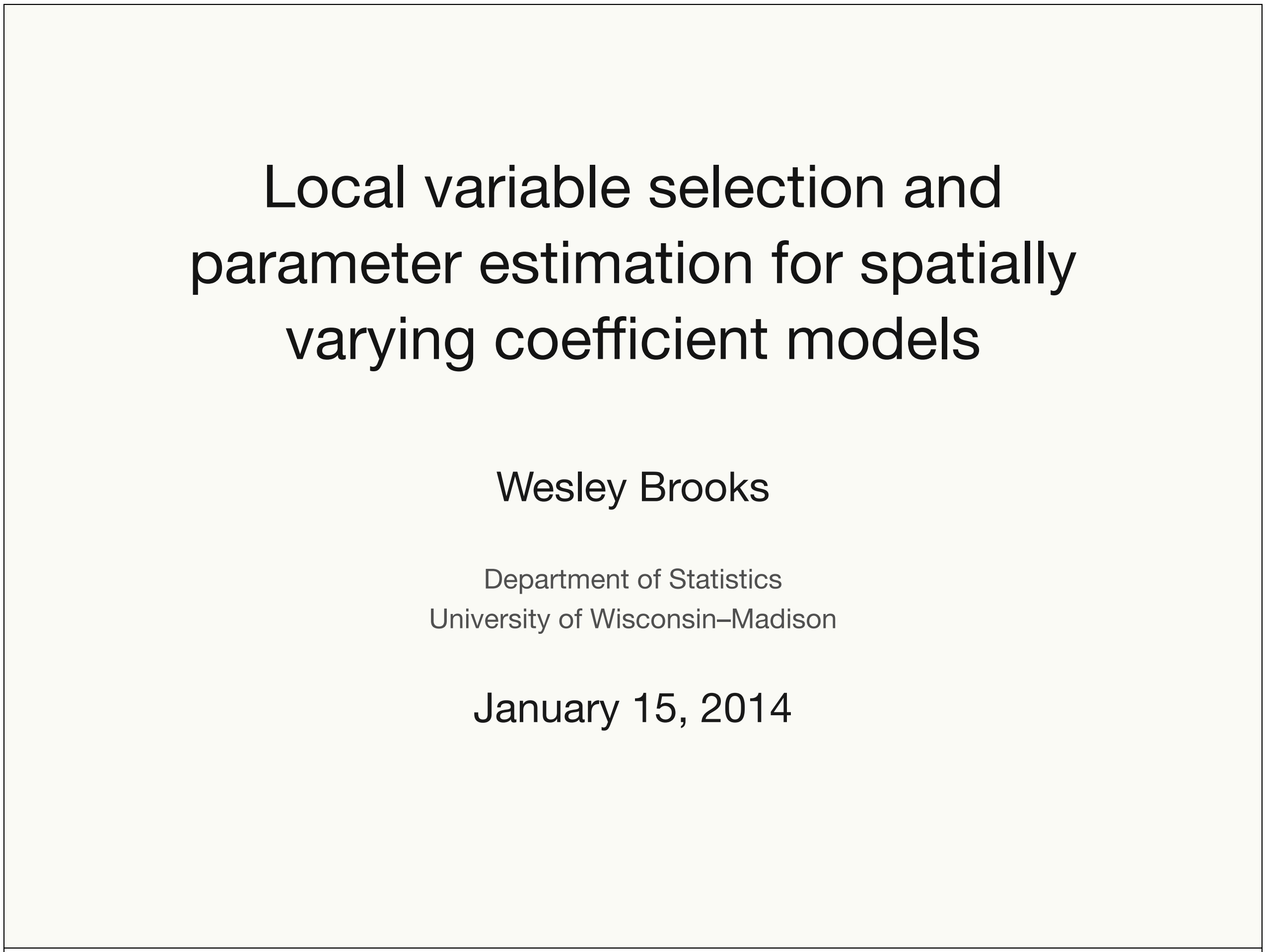


Local variable selection and parameter estimation for spatially varying coefficient models

Wesley Brooks

Department of Statistics
University of Wisconsin–Madison

January 15, 2014

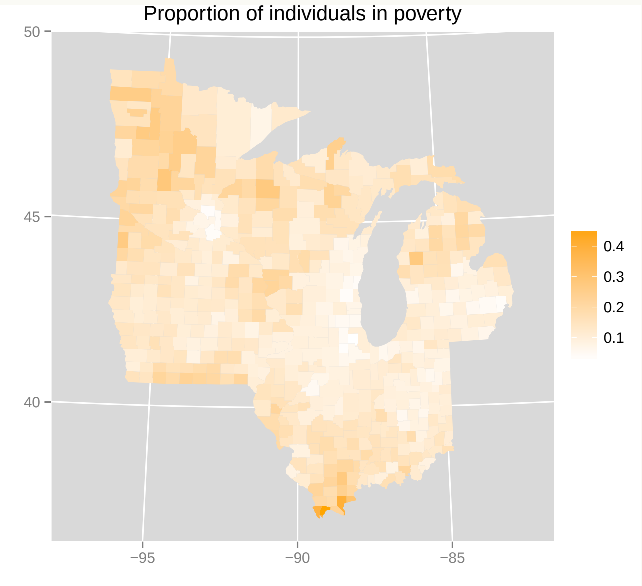
These slides were prepared for a practice version of my preliminary exam to advance to Ph.D candidacy in statistics at the University of Wisconsin–Madison.

Motivation

New Section

Motivation

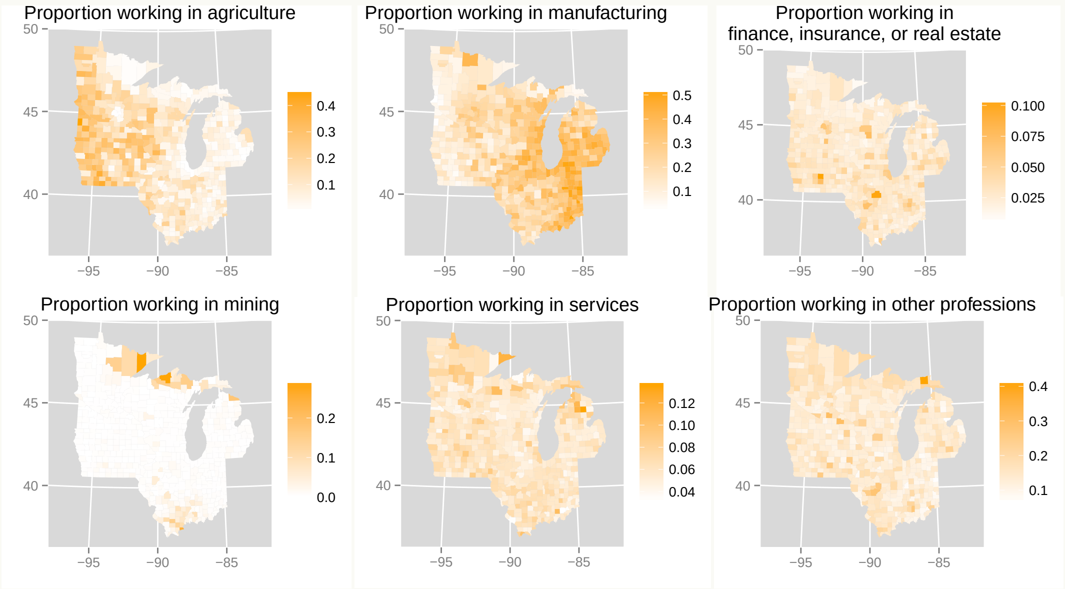
Response variable



This is the county-level poverty rate from 1970

Motivation

Covariates


Proportion working in agriculture

Proportion working in manufacturing

Proportion working in finance, insurance, or real estate

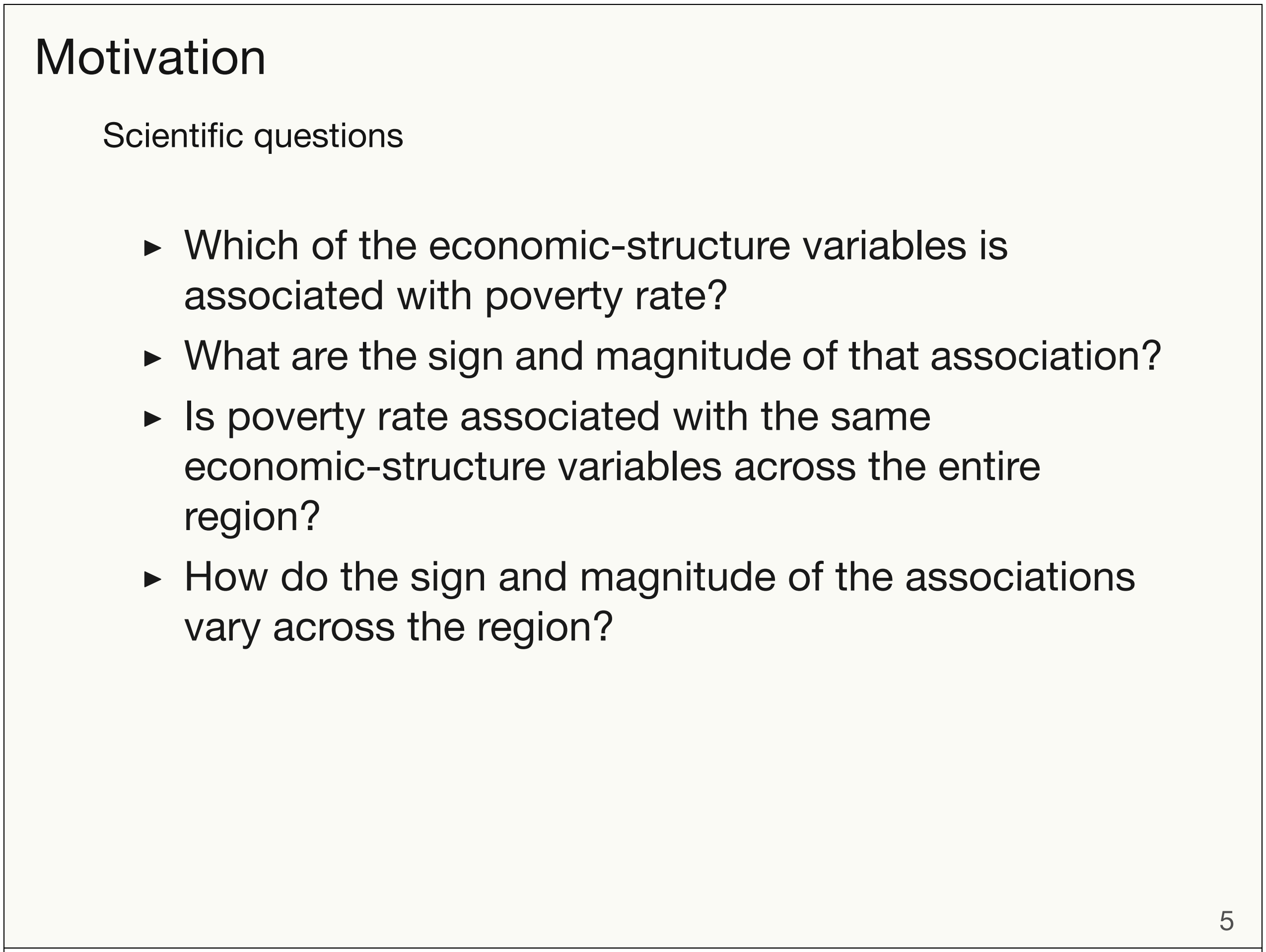
Proportion working in mining

Proportion working in services

Proportion working in other professions

Here we have the proportion of people in each county who worked in manufacturing, agriculture, and services in 1970.

How is this data to be analyzed?



Motivation

Scientific questions

- 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?
- How do the sign and magnitude of the associations vary across the region?

These are some sensible questions to ask about the county-level poverty rate. The work I'm presenting today attempts to answer these questions.

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

Introduction

New Section

Introduction

An overview

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

The existing methods to address the questions draw from these areas. Behind the methodology that I'm discussing is a wide range of literature.

Introduction

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.

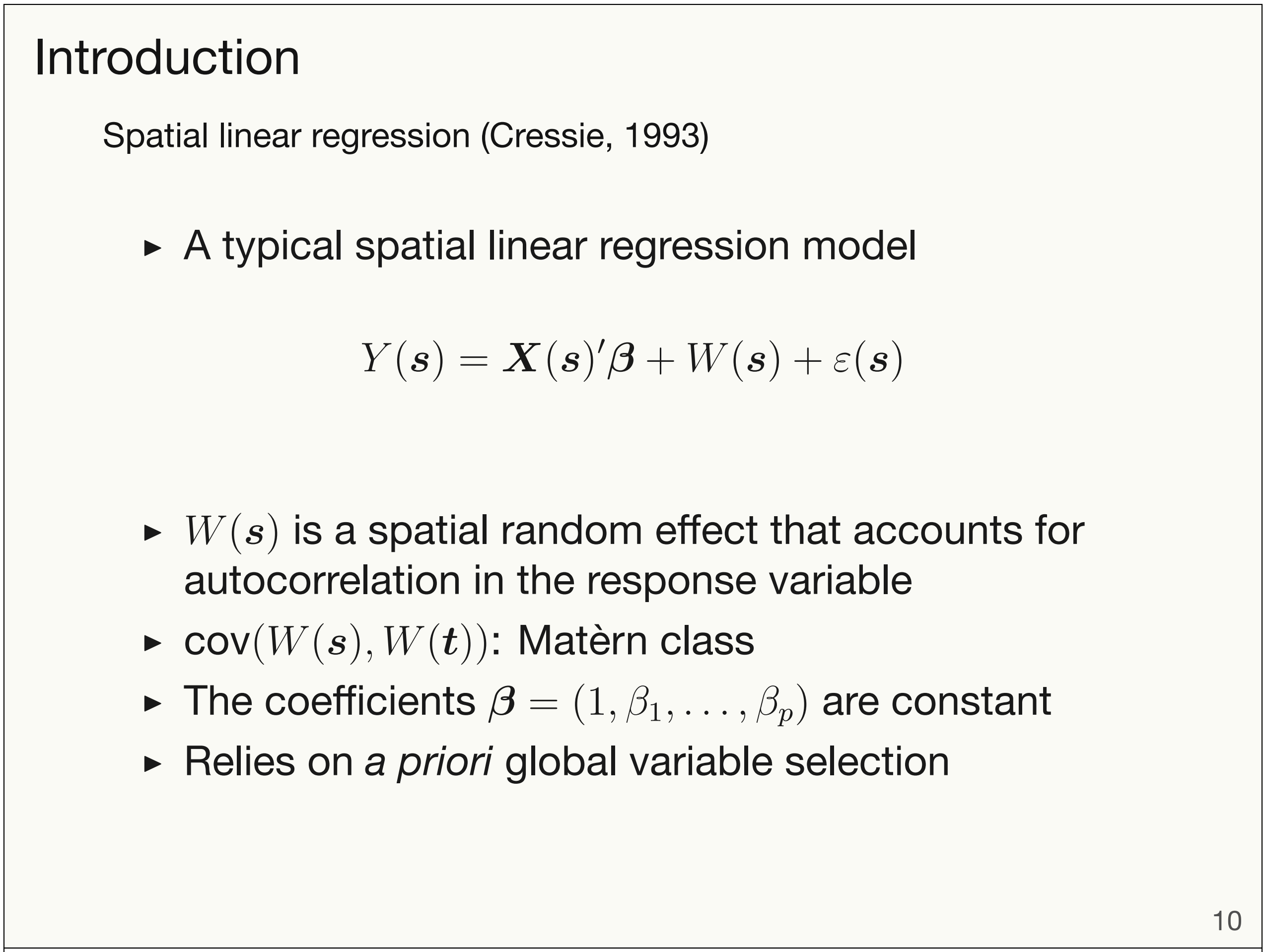
Introduction

Types of spatial data

- Geostatistical data:
  - Observations are made at sampling locations 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 I'll present later is based on simulated geostatistical data.



Introduction

Spatial linear regression (Cressie, 1993)

- A typical spatial linear regression model

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

- $W(\boldsymbol{s})$ is a spatial random effect that accounts for autocorrelation in the response variable
- cov$(W(\boldsymbol{s}), W(\boldsymbol{t}))$: Matèrn class
- The coefficients $\boldsymbol{\beta} = (1, \beta_1, \ldots, \beta_p)$ are constant
- Relies on *a priori* global variable selection

Here we have the usual spatial regression as described by Noel Cressie in his 1993 book.

This model assumes that the model coefficients are constant across the spatial domain and that the residuals can be separated into:

- The spatial random effect W that captures autocorrelation of the response, and - epsilon, which is iid white noise

The autocorrelation of the W's is from a Matérn class covariance function, like the exponential covariance function.

This model relies on a priori model selection.

Typically Bayesian methods are used to estimate the coefficients.

Introduction

Spatially varying coefficient model (Gelfand *et al.*, 2003)

- A more flexible model: coefficients in a spatial regression model can vary

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

- $\{\beta_0(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}, \ldots, \{\beta_p(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ are stationary spatial processes with Matèrn covariance functions
- Still relies on *a priori* global variable selection

The spatial regression model can be made more flexible by representing the coefficients as stationary spatial processes, rather than constants. The method was introduced by Gelfand in 2003.

The coefficient processes have matèrn class covariance functions, just like the autocorrelated errors W in the traditional spatial regression.

The autocorrelated errors W are now incorporated in the spatially varying intercept process.

This model also relies on a priori model selection and uses Bayesian methods to estimate the coefficients.

Introduction

Varying coefficients regression (VCR) (Hastie and Tibshirani, 1993)

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

- Assume an effect modifying variable $\boldsymbol{s}$
- Coefficients are functions of $\boldsymbol{s}$

The varying coefficient regression model was described by Hastie and Tibshirani in 1993. The form of this model looks like the spatially varying coefficient process, but this model is more general.

The coefficients are not necessarily spatial processes in this model. In fact, the effect-modifying variable s does not necessarily need to represent spatial location.

There are non-Bayesian methods to fit the model. We'll look at three.

Introduction

Spline-based VCR models (Wood, 2006)

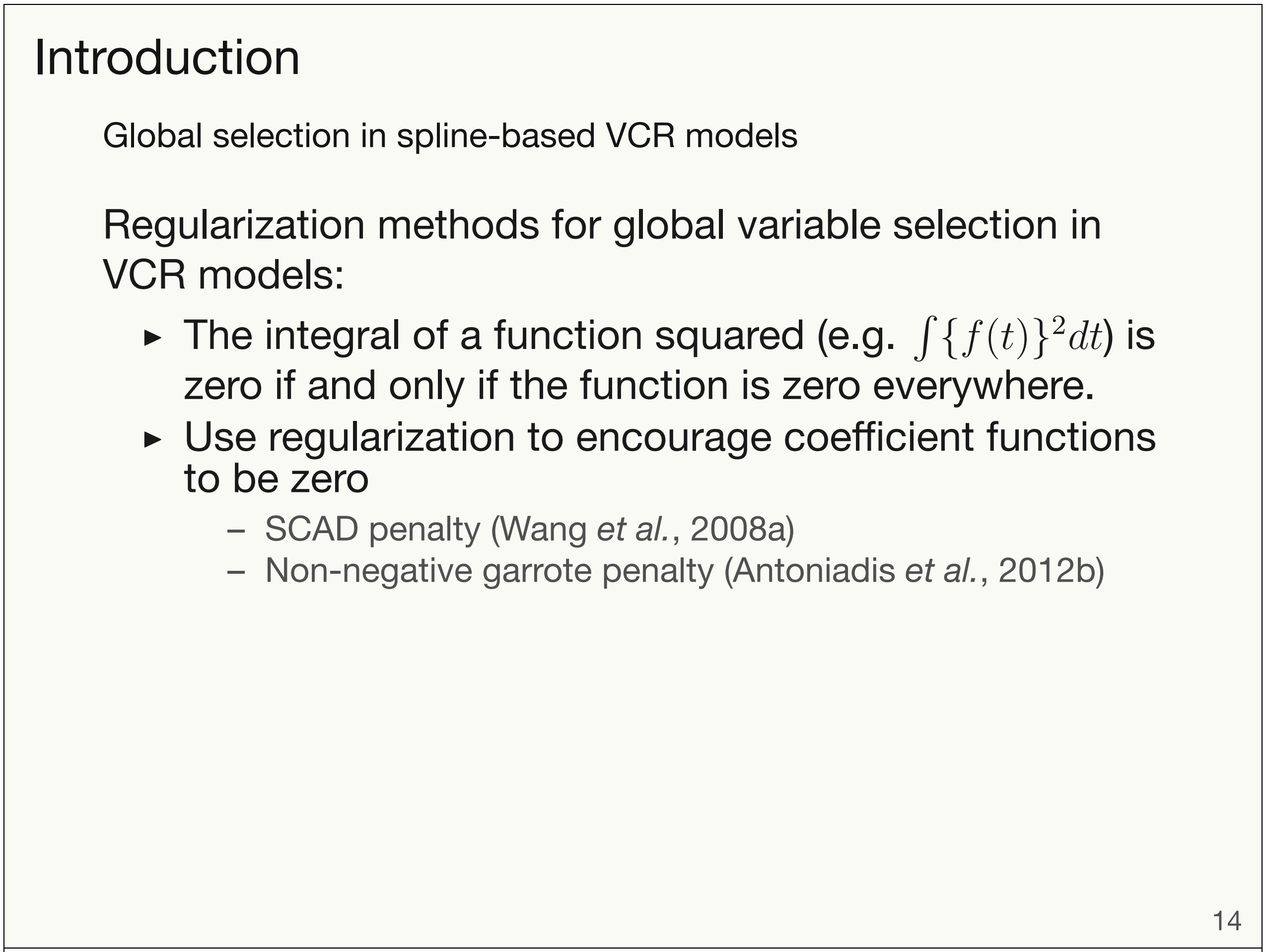
- Splines are a way to parameterize smooth functions
- Estimate the varying coefficients via splines:

$$E\{Y(t)\} = \beta_1(t)X_1(t) + \cdots + \beta_p(t)X_p(t)$$

There is a good overview in Simon Wood's 2006 book of how to use regression splines to fit a varying coefficients regression model.

Regression splines are a way to parameterize a smooth function. In this case, the coefficient is a smooth function of the spatial location

fitting a spline-based VCR requires a priori model selection(?).



Introduction

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 to encourage coefficient functions to be zero
  - SCAD penalty (Wang *et al.*, 2008a)
  - Non-negative garrote penalty (Antoniadis *et al.*, 2012b)

There are at least two references that describe how to select the covariates for a spline-based VCR model. Both rely on regularization.

The regularization penalizes the smooth coefficient function for being non-zero. Antoniadis et al. used a non-negative garrote penalty and Wang et al. used a SCAD penalty.

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

Introduction

Wavelet methods for VCR models

- Wavelet methods: decompose coefficient function into local frequency components
- Selection of nonzero local frequency components with nonzero coefficients:
  - Bayesian variable selection (Shang, 2011)
  - Lasso (Zhang and Clayton, 2011)
- Sparsity in the local frequency components; not in the local covariates

Another way to fit a VCR model is to use a wavelet decomposition, which decomposes the coefficient function into its local frequency components. Model selection is then used to identify which local frequency components to use in the model.

Murray Clayton's students Zuofeng Shang and Jun Zhang used Bayesian variable selection and the Lasso, respectively, to select the local frequency components.

However, these methods achieve sparsity in the wavelet coefficients, which does not imply sparsity in the covariates. So these methods don't achieve local model selection.

Now let's take a look at geographically weighted regression.

Geographically weighted regression

New Section

Geographically weighted regression

Brundson *et al.* (1998), Fotheringham *et al.* (2002)

- Consider observations at sampling locations $\boldsymbol{s}_1, \dots, \boldsymbol{s}_n$
- $y(\boldsymbol{s}_i) = y_i$ the univariate response at location $\boldsymbol{s}_i$
- $\boldsymbol{x}(\boldsymbol{s}_i) = \boldsymbol{x}_i$ the $(p+1)$-variate vector of covariates at location $\boldsymbol{s}_i$
- Assume $y_i = \boldsymbol{x}_i'\boldsymbol{\beta}_i + \varepsilon_i$ where $\varepsilon_i \overset{iid}{\sim} \mathcal{N}\left(0, \sigma^2\right)$

Geographically weighted regression is the method of using local regression to estimate the coefficients in a spatially varying coefficient regression model.

Our sampling locations are called s, the response is y and the covariates (which number p) are called x.

Assume that the errors are iid normal.

The notation β_i is used to indicate that he coefficients are specific to location i.

Geographically weighted regression

Brundson *et al.* (1998), Fotheringham *et al.* (2002)

- The total log likelihood is

$$\ell\left(\boldsymbol{\beta}\right) = -\left(1/2\right)\left\{n \operatorname{log}\left(2\pi\sigma^2\right) + \sigma^{-2}\sum_{i=1}^{n}\left(y_i - \boldsymbol{x}_i'\boldsymbol{\beta}_i\right)^2\right\}$$

- With n observations and $n(p+1)$ parameters, the model is not identifiable.
- Idea: to estimate parameters by borrowing strength from nearby observations

We have here the total log likelihood of the observed data.

Because each β_i is a p-vector of local coefficients, this model has n observations and n(p+1) free parameters, so the model is not identifiable.

We will estimate the parameters by borrowing strength from nearby observations

Geographically weighted regression

Local regression (Loader, 1999)

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

$$\mathcal{L}_i = \prod_{i'=1}^{n} \left(\mathcal{L}_{i'}\right)^{w_{ii'}}$$

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

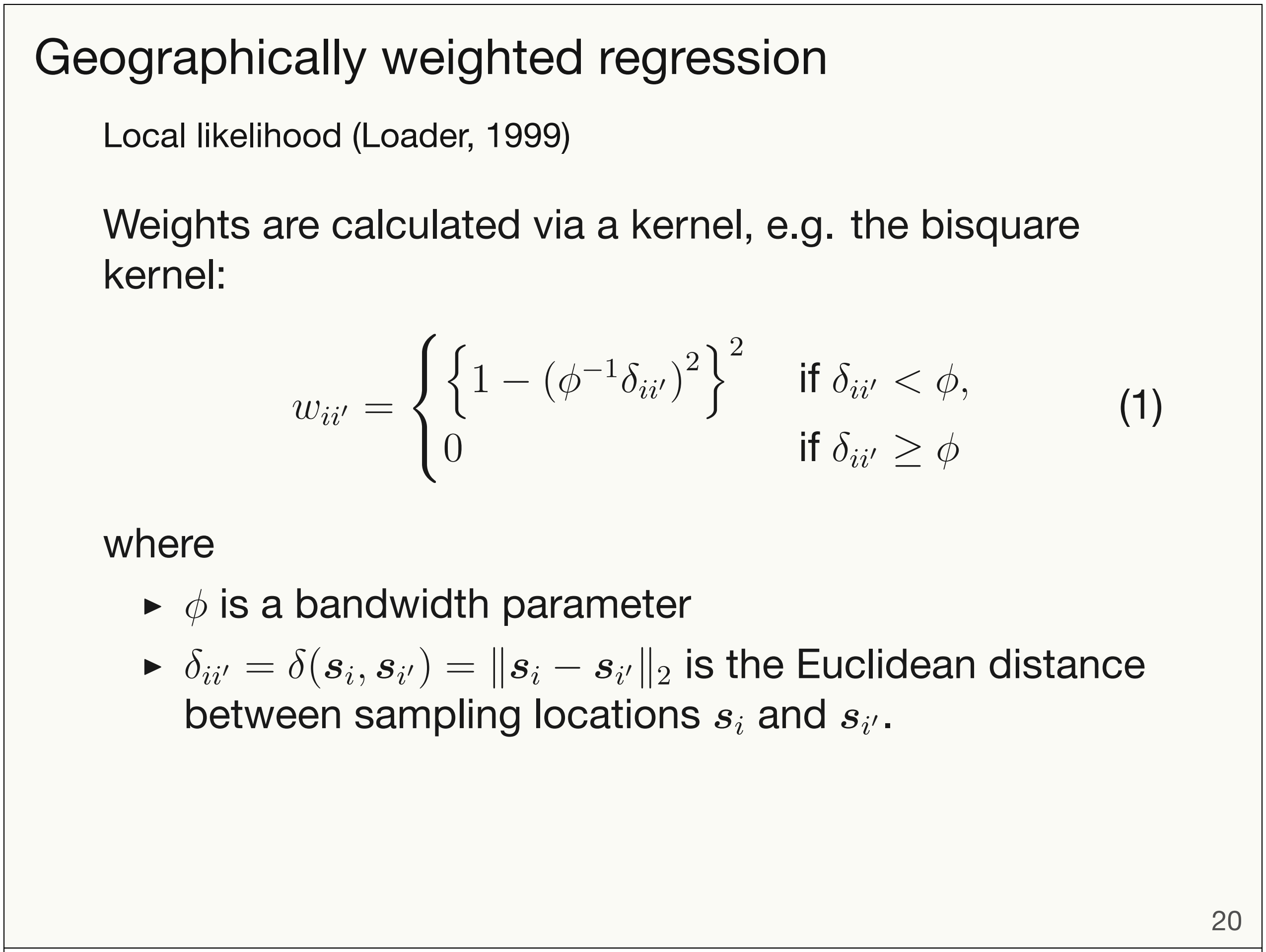
Given the weights, a local model is fit at each sampling location using the local likelihood

Local regression uses a kernel function at each sampling location to weight the observations. For a GWR model, the kernel weights are based on an observation's distance from the sampling location.

Here we have the likelihood at one sampling location. Note that each observation is given a weight $w_{ii'}$

Given the weights, a local model is fit at each sampling location using the local likelihood

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



Geographically weighted regression

Local likelihood (Loader, 1999)

Weights are calculated via a kernel, e.g. the bisquare kernel:

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

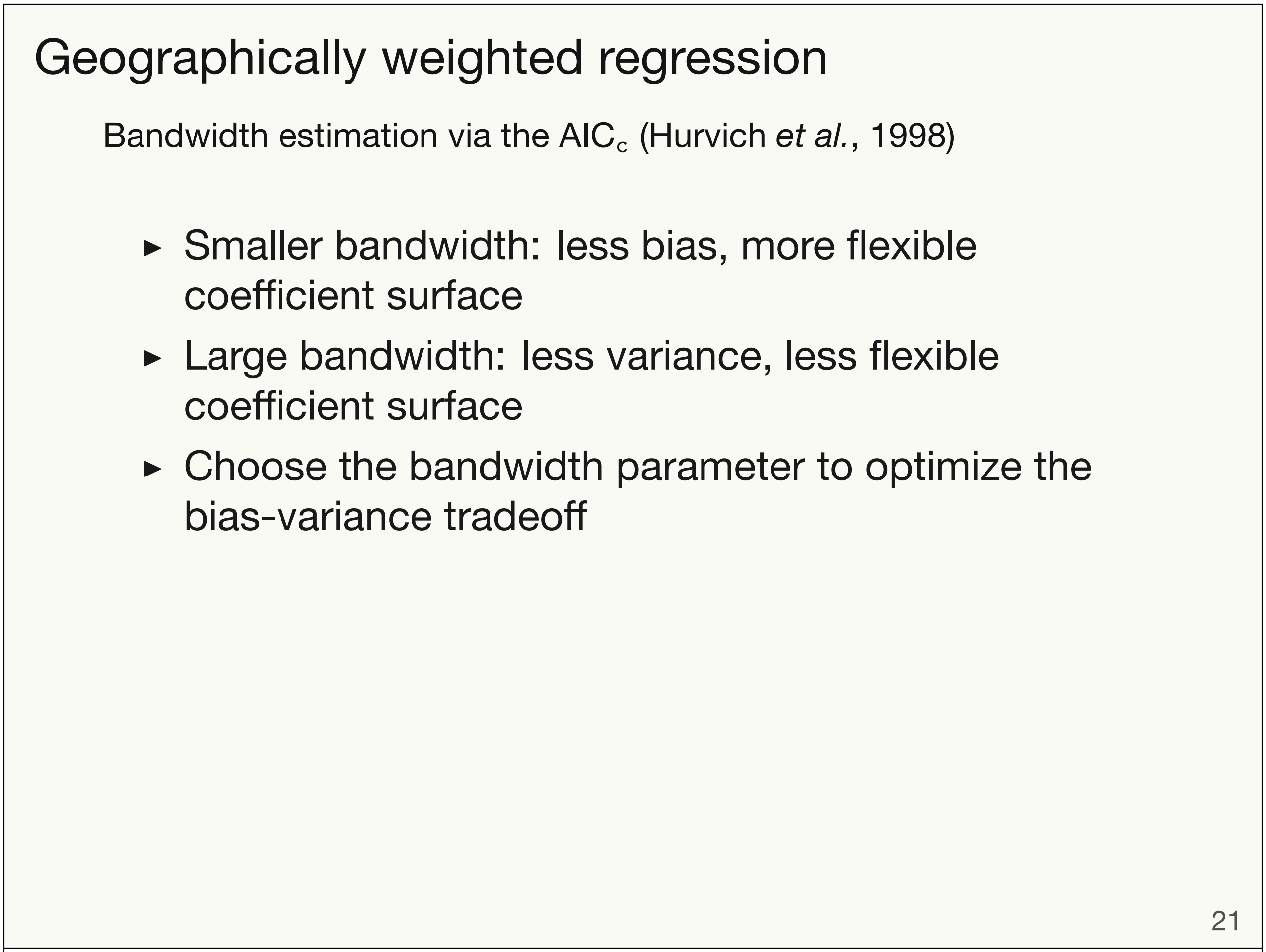
where

- ϕ is a bandwidth parameter
- $\delta_{ii'} = \delta(\boldsymbol{s}_i, \boldsymbol{s}_{i'}) = \|\boldsymbol{s}_i - \boldsymbol{s}_{i'}\|_2$ is the Euclidean distance between sampling locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_{i'}$.

The local weights $w_{ii'}$ from the previous slide are calculated from a kernel.

This is the form of the bisquare kernel, which is what I've used in this work.

ϕ is a bandwidth parameter and $\delta_{ii'}$ is the distance between points i and i'.



Geographically weighted regression

Bandwidth estimation via the AIC$_c$ (Hurvich *et al.*, 1998)

- Smaller bandwidth: less bias, more flexible coefficient surface
- Large bandwidth: less variance, less flexible coefficient surface
- Choose the bandwidth parameter to optimize the bias-variance tradeoff

To estimate a GWR model, it is necessary to estimate the bandwidth parameter, which involves a bias-variance tradeoff.

When the bandwidth is small, the coefficient surface is flexible and should have less bias but greater variance.

When the bandwidth is large, the coefficient surface is less flexible so it has less variance but potentially more bias.

Geographically weighted regression

Bandwidth estimation via the AIC_c (Hurvich *et al.*, 1998)

- Idea: to estimate the degrees of freedom used in estmating the coefficient surface:
- Then the corrected AIC for bandwidth selection is:

$$\text{AIC}_\text{c} = 2n \log \sigma + n \left\{ \frac{n+\nu}{n-2-\nu} \right\}$$

  - $\hat{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y}$
  - $\nu = \text{tr}(\boldsymbol{H})$
  - $\boldsymbol{H}_i = \left\{ \boldsymbol{W}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X} \right\}_i$
  - Where subscript i indicates the ith row of the matrix

One way to estimate the GWR bandwidth is via the corrected AIC of Hurvich et al..

Geographically weighted regression

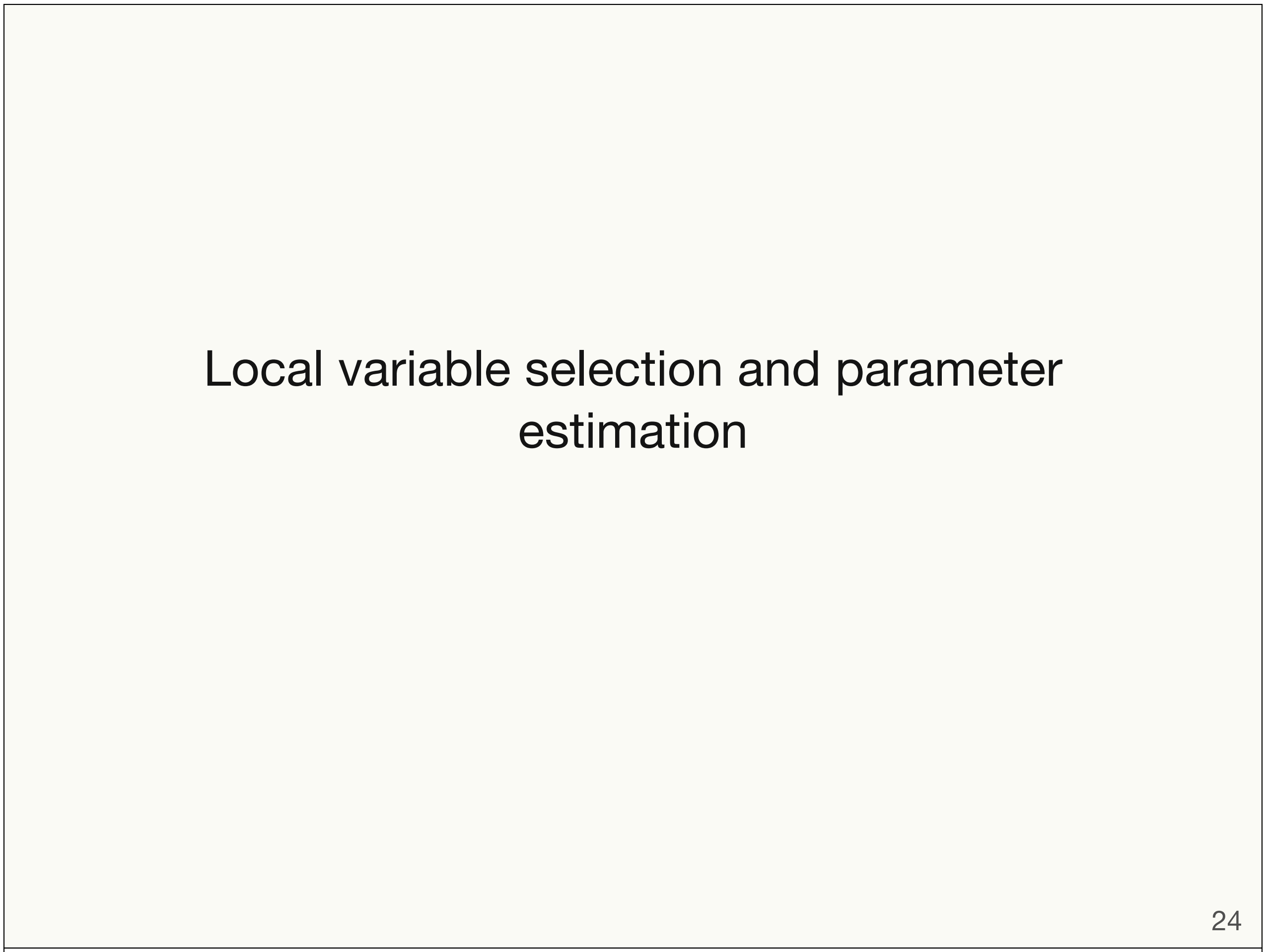
Bandwidth estimation via GCV (Wahba, 1990)

- Idea: to estimate the degrees of freedom used in estmating the coefficient surface:
- Then the corrected AIC for bandwidth selection is:

$$GCV = \frac{\sum_i = 1^n \left(y - \hat{y}\right)^2}{(n - \nu)^2}$$

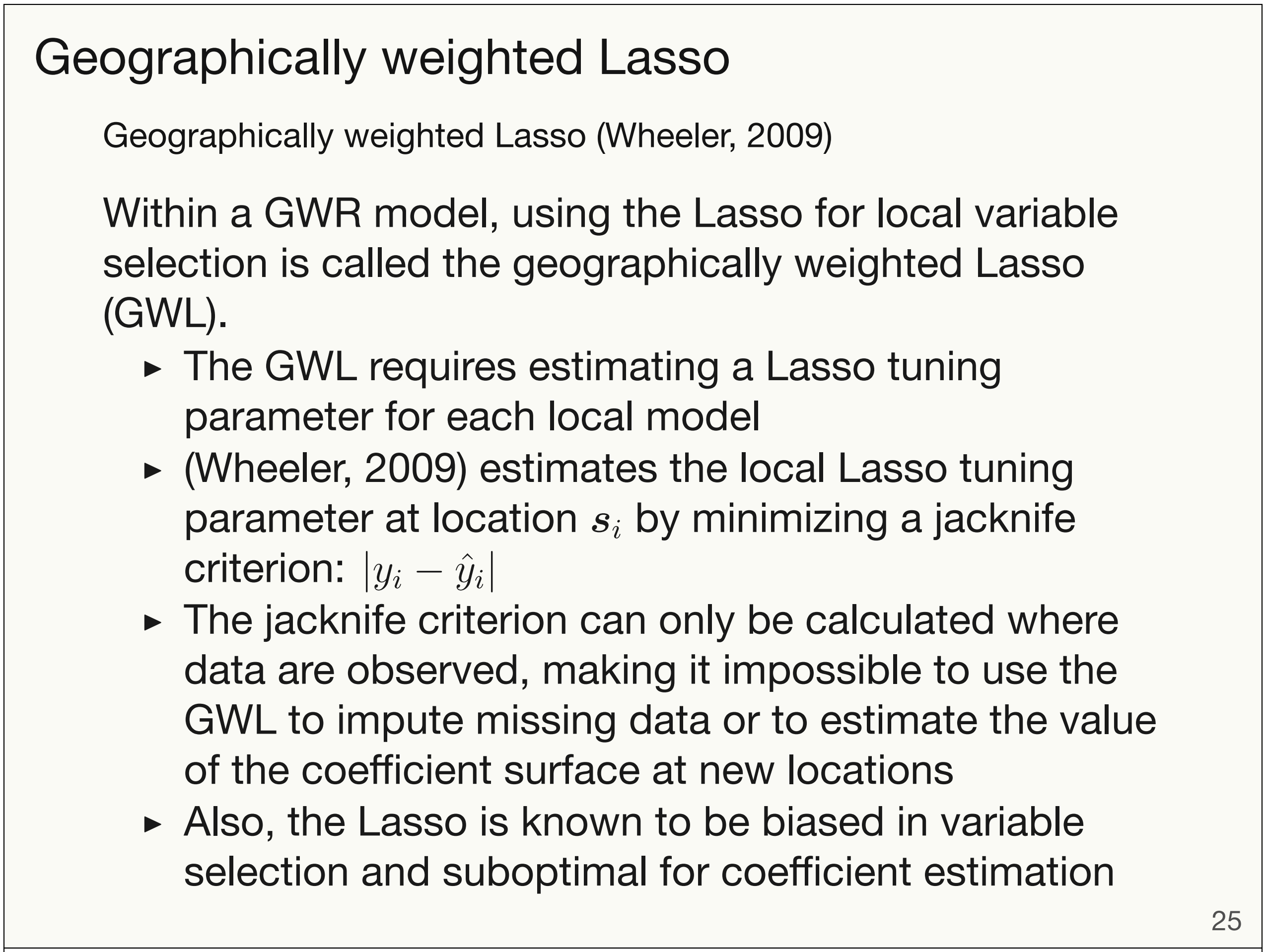
- $\hat{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y}$
- $\nu =$ tr($\boldsymbol{H}$)
- $\boldsymbol{H}_i = \left\{\boldsymbol{W}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}\right\}_i$
- Where subscript i indicates the ith row of the matrix

One way to estimate the GWR bandwidth is via the corrected AIC of Hurvich et al..



Local variable selection and parameter estimation

New Section



Geographically weighted Lasso

Geographically weighted Lasso (Wheeler, 2009)

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

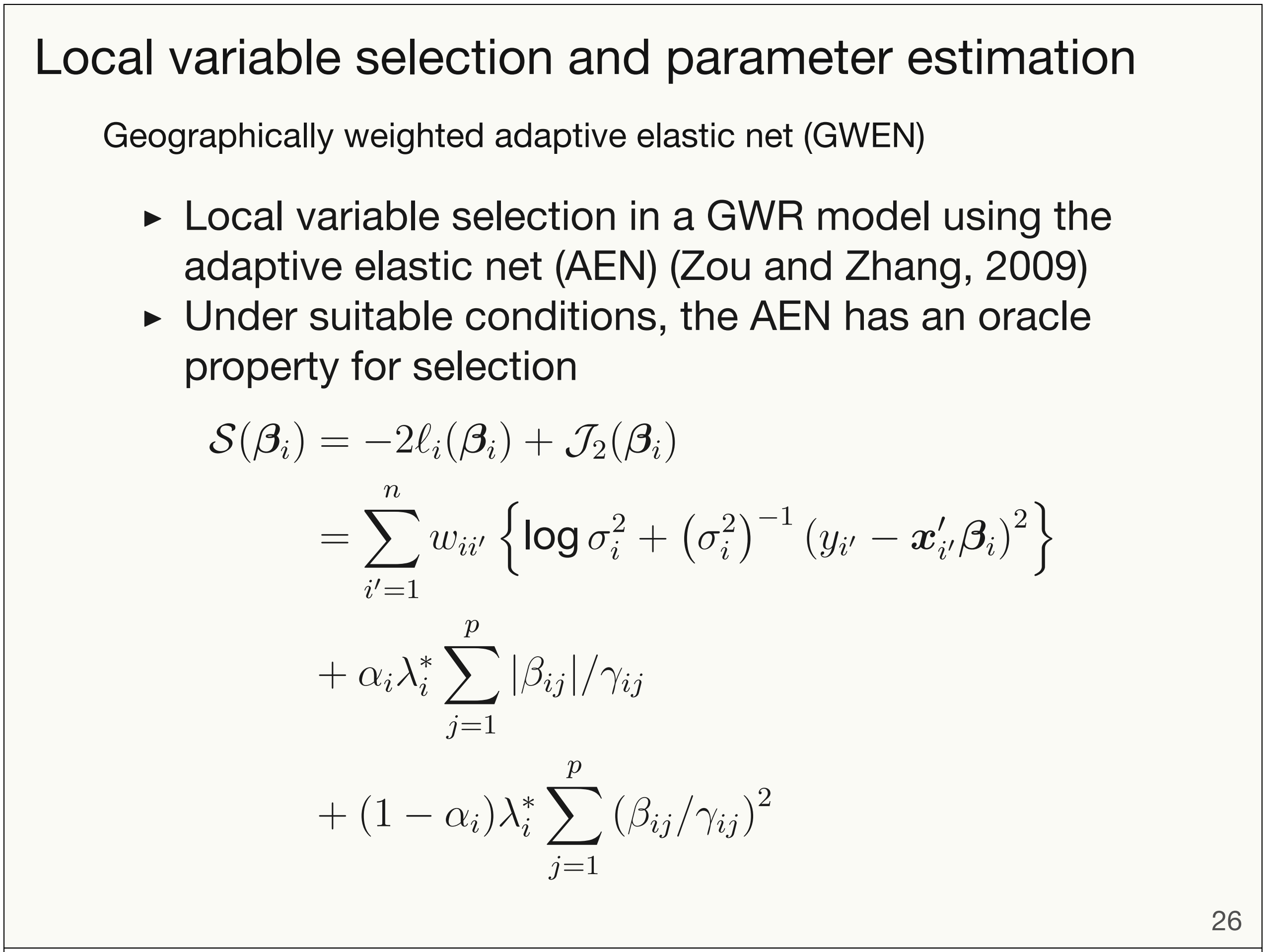
- The GWL requires estimating a Lasso tuning parameter for each local model
- (Wheeler, 2009) estimates the local Lasso tuning parameter at location $\boldsymbol{s}_i$ by minimizing a jacknife criterion: $|y_i - \hat{y}_i|$
- The jacknife criterion can only be calculated where data are observed, making 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

For local model selection in a GWR model, Wheeler proposed the geographically weighted lasso (GWL) in 2009.

At each model location, the Lasso is used to select the locally-relevant predictors

The GWL uses a jacknife criterion to select the local lasso tuning parameters, which means the GWL cannot be used at model locations other than sample locations.

That means the GWL cannot be used for interpolating the coefficient surface or for imputing missing values of the response variable.



Local variable selection and parameter estimation

Geographically weighted adaptive elastic net (GWEN)

- Local variable selection in a GWR model using the adaptive elastic net (AEN) (Zou and Zhang, 2009)
- 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 geographically weighted adaptive elastic net (GWEN) is similar to the GWAL but uses the elastic net for local model selection

The adaptive elastic net also has an oracle property under suitable conditions.

The adaptive elastic net consists of adding an L2 penalty to the regularization in addition to the L1 penalty of the adaptive lasso.

S here is the penalized likelihood for a local GWEN model

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 $\lambda_i^* \sum_{j=1}^{p} \left(\beta_{ij}/\gamma_{ij}\right)^2$.

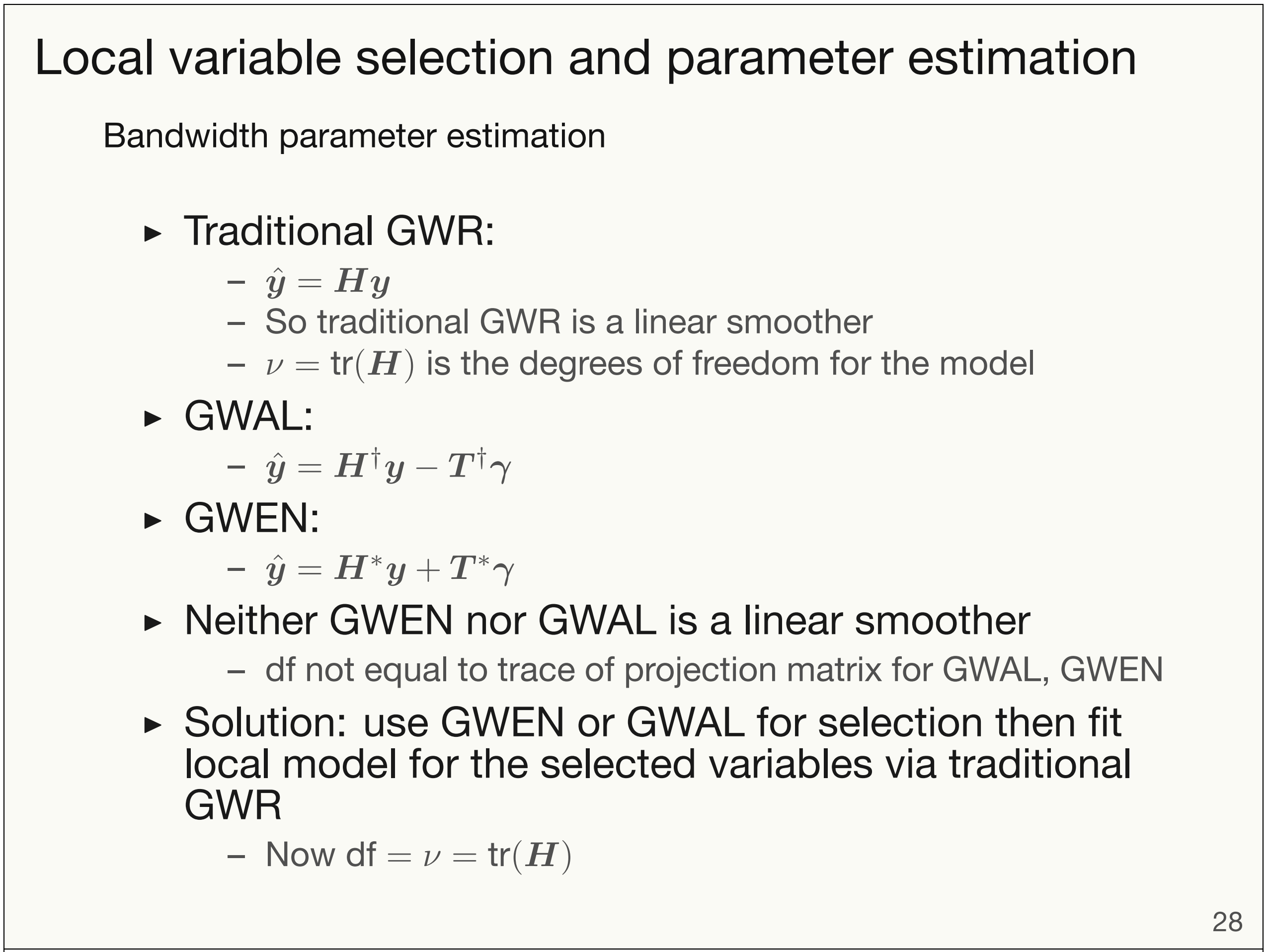
Local variable selection and parameter estimation

Geographically weighted adaptive elastic net (GWEN)

- The AEN penalty function is

$$\mathcal{J}_2(\boldsymbol{\beta}_i) = \alpha_i \lambda_i^* \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij} + (1-\alpha_i)\lambda_i^* \sum_{j=1}^{p} (\beta_{ij}/\gamma_{ij})^2$$

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 $\lambda_i^* \sum_{j=1}^{p} (\beta_{ij}/\gamma_{ij})^2$.



Local variable selection and parameter estimation

Bandwidth parameter estimation

- Traditional GWR:
  - $\hat{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y}$
  - So traditional GWR is a linear smoother
  - $\nu =$ tr$(\boldsymbol{H})$ is the degrees of freedom for the model
- GWAL:
  - $\hat{\boldsymbol{y}} = \boldsymbol{H}^{\dagger}\boldsymbol{y} - \boldsymbol{T}^{\dagger}\boldsymbol{\gamma}$
- GWEN:
  - $\hat{\boldsymbol{y}} = \boldsymbol{H}^{*}\boldsymbol{y} + \boldsymbol{T}^{*}\boldsymbol{\gamma}$
- Neither GWEN nor GWAL is a linear smoother
  - df not equal to trace of projection matrix for GWAL, GWEN
- Solution: use GWEN or GWAL for selection then fit local model for the selected variables via traditional GWR
  - Now df $= \nu =$ tr$(\boldsymbol{H})$

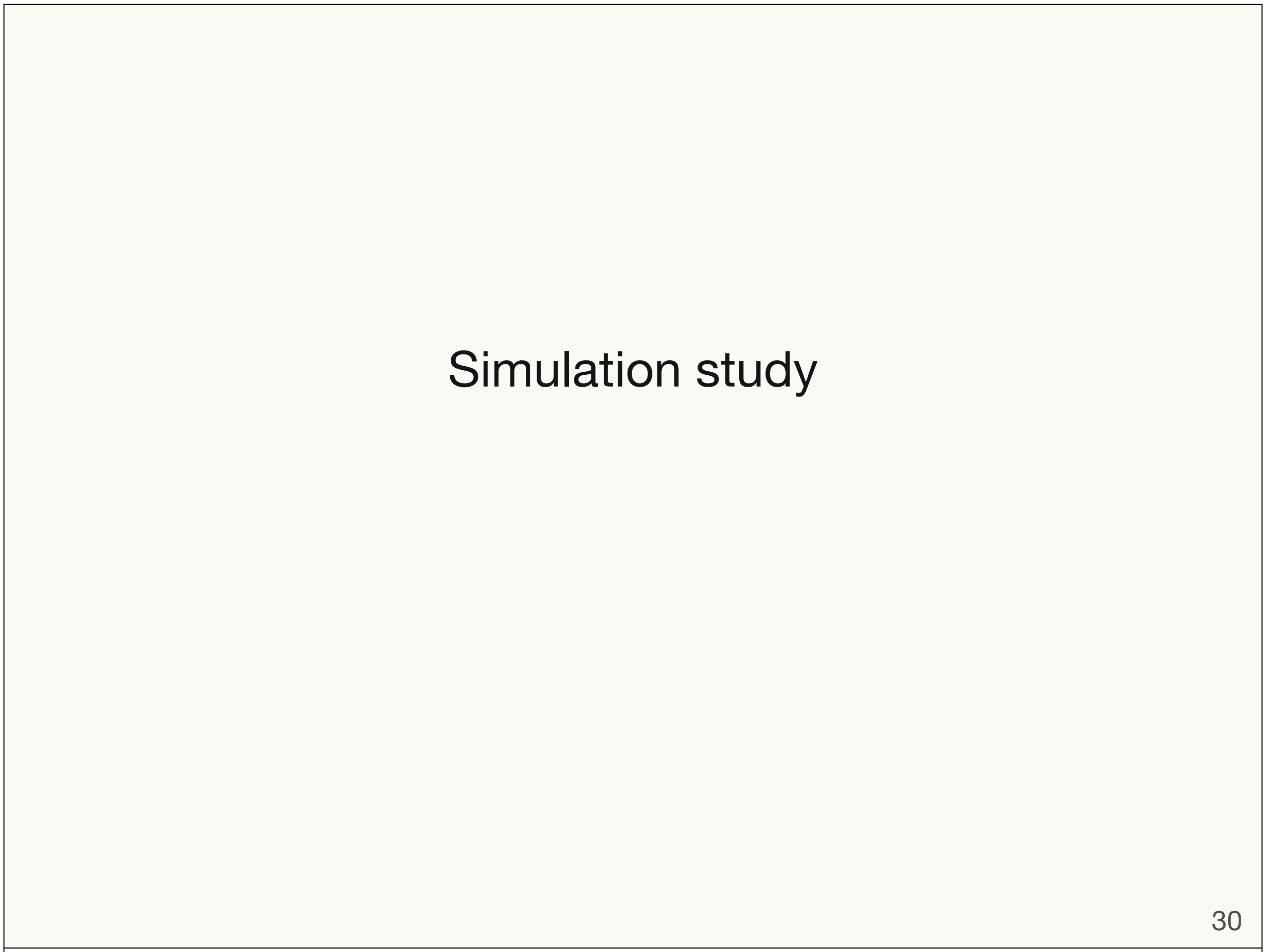
The

Local variable selection and parameter estimation

Locally linear coefficient estimation

- GWR, GWEN, GWAL: coefficients locally constant
  - as in Nadaraya-Watson kernel smoother
  - Leads to bias where there is a gradient at the boundary
- Solution: local polynomial modeling
  - First-order polynomial: locally linear coefficients
- Augment with covariate-by-location interactions
  - Two-dimensional
  - Augment with selected covariates only

note



Simulation study

New Section

Simulation study

Simulating covariates

- 30×30 grid on $[0,1] \times [0,1]$
- Five covariates $\tilde{X}_1, \ldots, \tilde{X}_5$
- Gaussian random fields:

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

- Colinearity: ρ

note

Simulation study

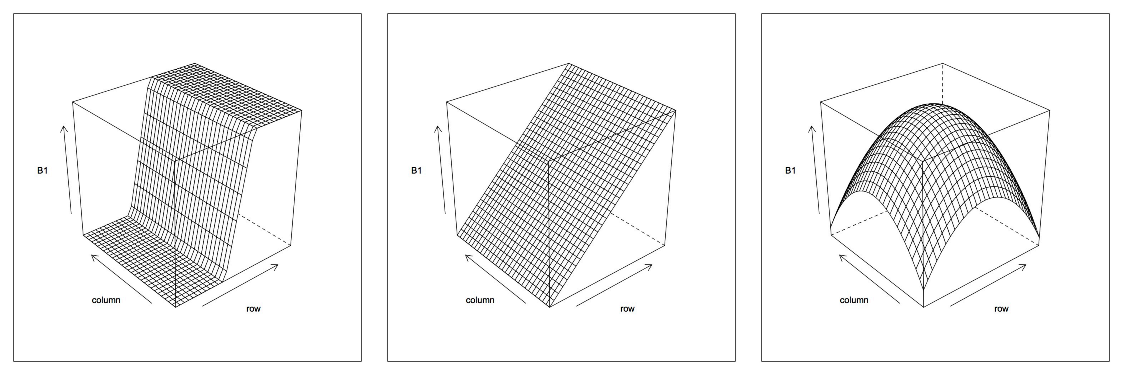
Simulating the response

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

note

Simulation study

Coefficient functions: step, gradient, and parabola



note

Simulation study

Simulation settings

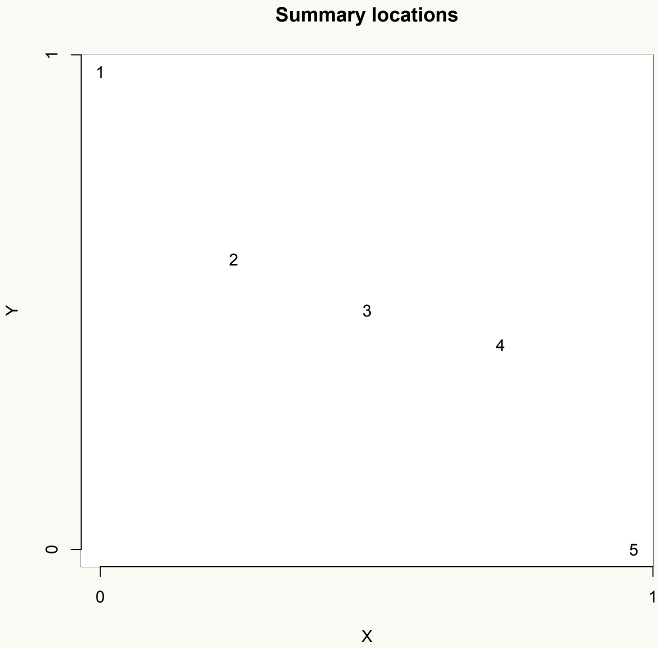
Each setting simulated 100 times:

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

note

Simulation results

Summary locations



note

Simulation results

Selection performance

- Non-ambiguous locations (80):
  - 52 saw no false negatives
  - 72 had no false positives
  - 26 neither false positives nor false negatives
- Incerased noise variance led to worse selection performance
- Increased colinearity in the covariates led to worse selection performance
- No difference between GWEN and GWAL

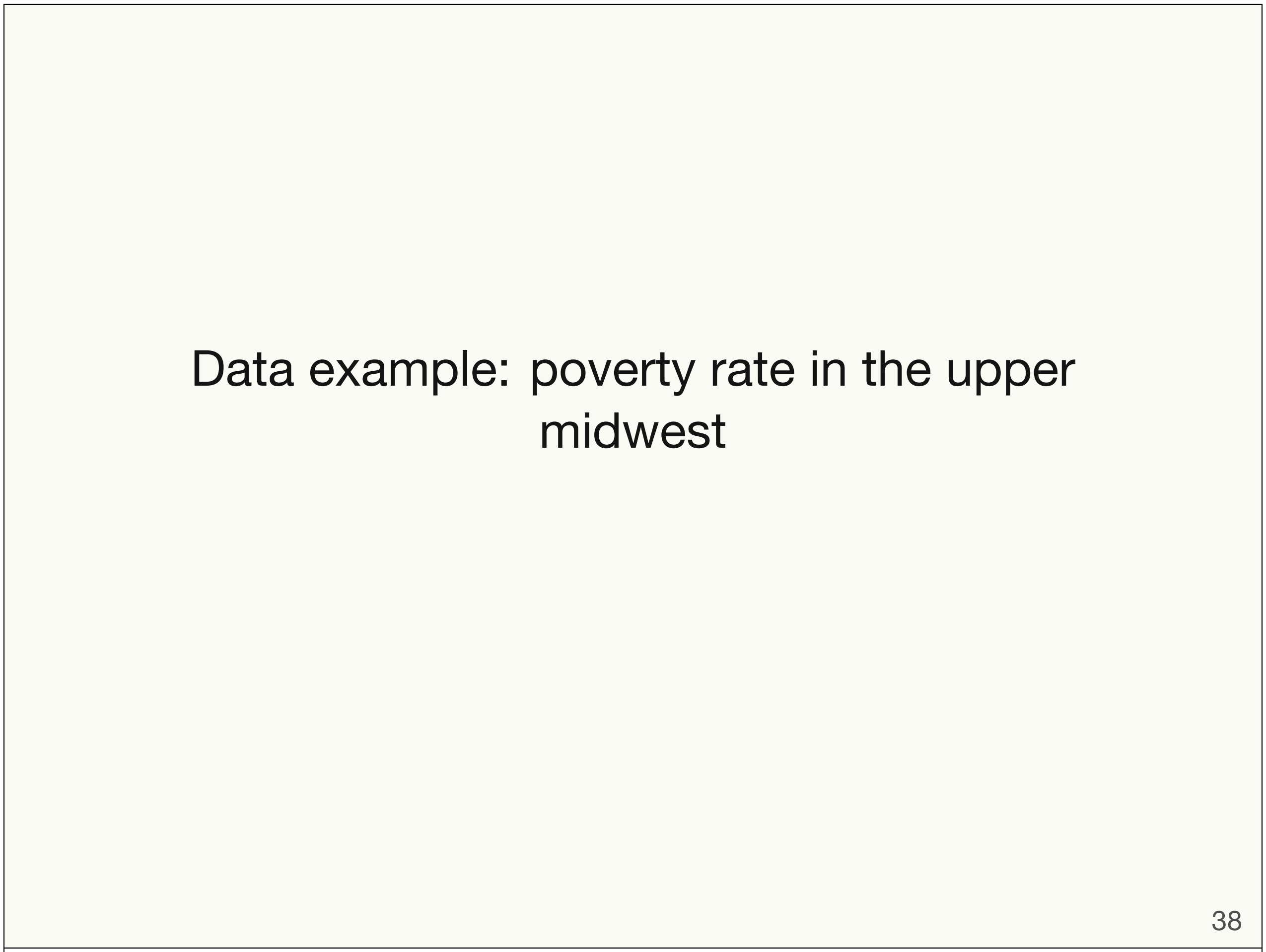
note

Simulation results

Estimation performance

- Oracular selection
  - best MSE($\hat{\beta}_1$) in 41 of the 60 cases
- Generally small difference between GWR, oracular, GWEN-LLE, and GWAL-LLE
- Incerased noise variance led to worse estimation accuracy
- Increased colinearity in the covariates led to worse estimation accuracy
- Fitting $\hat{y}$: best MSE split between GWAL-LLE, oracle, and GWR

note

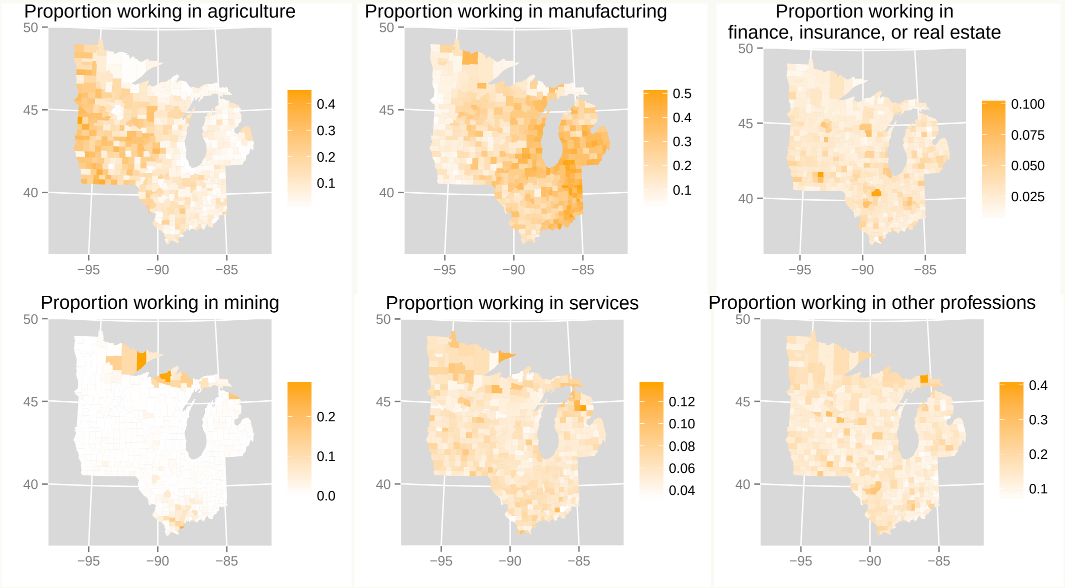


Data example: poverty rate in the upper midwest

New Section

Data example: poverty rate in the upper midwest

Revisiting the motivating example


Proportion working in agriculture
50
45
40
−95
−90
−85
0.4
0.3
0.2
0.1
Proportion working in manufacturing
50
45
40
−95
−90
−85
0.5
0.4
0.3
0.2
0.1
Proportion working in
finance, insurance, or real estate
50
45
40
−95
−90
−85
0.100
0.075
0.050
0.025
Proportion working in mining
50
45
40
−95
−90
−85
0.2
0.1
0.0
Proportion working in services
50
45
40
−95
−90
−85
0.12
0.10
0.08
0.06
0.04
Proportion working in other professions
50
45
40
−95
−90
−85
0.4
0.3
0.2
0.1

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 example: poverty rate in the upper midwest

Data description

- Response: logit-transformed poverty rate in the Upper Midwest states of the U.S.
  - Minnesota, Iowa, Wisconsin, Illinois, Indiana, Michigan
- Covariates: employment structure (raw proportion employed in:)
  - agriculture
  - finance, insurance, and real estate
  - manufacturing
  - mining
  - services
  - other professions
- Data source: U.S. Census Bureau's decennial census of 1970

note

Data example: poverty rate in the upper midwest

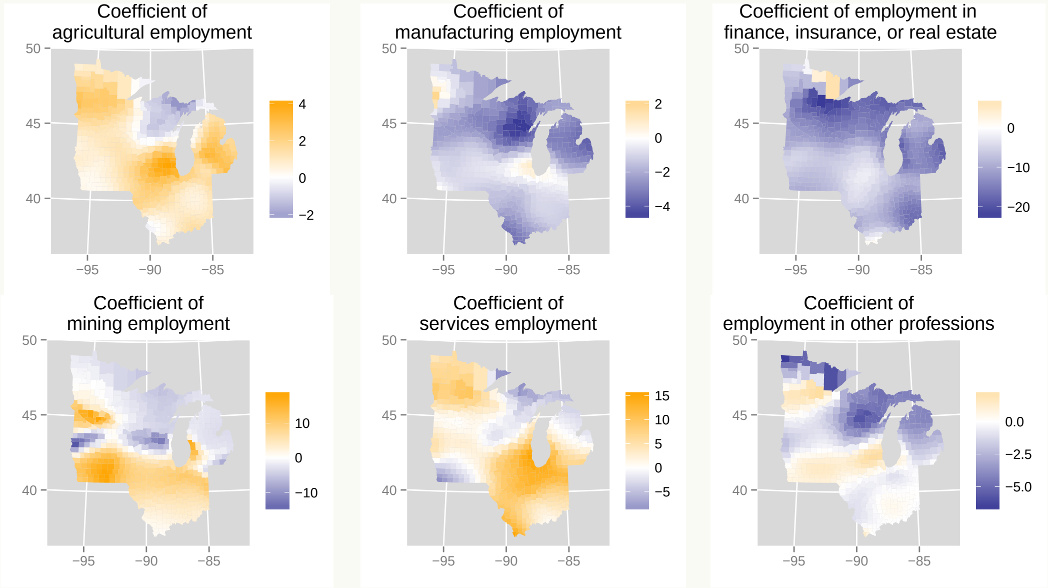
Data description

- Data aggregated to the county level
  - counties are areal units
- county centroid treated as sampling location

note

Data example: poverty rate in the upper midwest

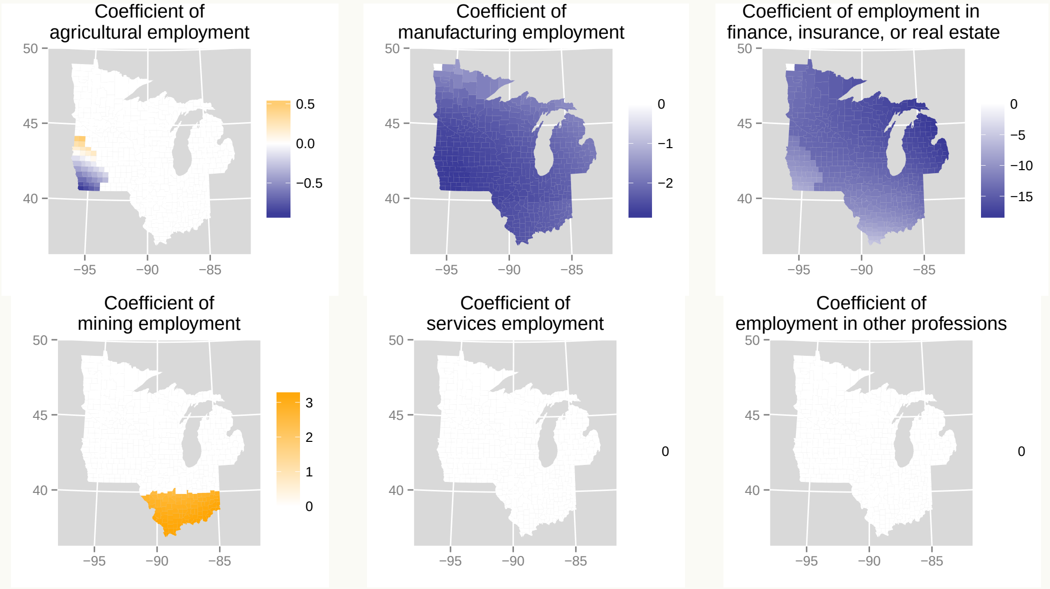
Results from traditional GWR



note

Data example: poverty rate in the upper midwest

Results from GWEN



Coefficient of agricultural employment

Coefficient of manufacturing employment

Coefficient of employment in finance, insurance, or real estate

Coefficient of mining employment

Coefficient of services employment

Coefficient of employment in other professions

note

Data example: poverty rate in the upper midwest

Results from GWEN-LLE

- Relatively constant compared to GWR
- Services, "other professions" do not affect the poverty rate
- Manufacturing: negative coefficient everywhere
- Finance, insurance, and real estate negative coefficient everywhere
  - Largest magnitude (min: -20, next-largest: -3)
  - GWR comparable to GWEN-LLE
- Manufacturing: negative coefficient everywhere
  - GWR: coefficient greater than zero near Chicago and in NW Minnesota
- Agriculture: nonzero in western Iowa
  - North-south gradient to coefficient
  - ranges positive to negative
- Mining: nonzero in parts south
  - Associated with increased poverty rate
  - Comparable to GWR within far southern range

note

Future work

New Section

Future work

- Apply the GWEN to models for non-Gaussian response variable
- Incorporate spatial autocorrelation in the model
- PalEON project: modeling and mapping tree biomass in the upper midwest

note

Acknowledgements