Local Variable Selection and Parameter Estimation of Spatially Varying Coefficient Regression Models

Wesley Brooks

Abstract

Spatially varying coefficient regression models allow the regression coefficients to vary across a spatial domain of interest . Geographically weighted regression, a kernel-based method for estimating the local regression coefficients in a spatially varying coefficient regression model, is considered here. A new method is introduced for local model selection and coefficient estimation in spatially varying coefficient regression models. The idea is to apply a penalty of the elastic net type to a local likelihood function, with a local elastic net tuning parameter and a global bandwidth parameter selected via information criteria. Simulations are used to evaluate the performance of the new method in model selection and coefficient estimation, and the method is applied to a real data example in spatial demography.

1. Introduction

Whereas the coefficients in traditional linear regression are scalar constants, the coefficients in a varying coefficient regression (VCR) model are functions - often *smooth* functions - of some effect modifying variable (Hastie and Tibshirani, 1993). When the effect modifying variable represents location in a spatial domain, a VCR model implies a spatially varying coefficient regression (SVCR) model wherein that the regression coefficients vary over space. Statistical inference for the coefficients as functions of location in an SVCR model is more complicated than estimating

the coefficients in a traditional linear regression model where the coefficients are constant across the spatial domain. My research concerns the development of new methodology for the analysis of spatial data using SVCR.

The methodology described herein is applicable to geostatistical data and areal data. Let $\mathcal{D}$ be a spatial domain on which data is collected. For geostatistical data, let $\boldsymbol{s}$ denote a location in $\mathcal{D}$. Let a univariate spatial process $\{Y(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ and a possibly multivariate spatial process $\{\boldsymbol{X}(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ denote random fields of the response and the covariates, respectively. For $i = 1, \ldots, n$, let $\boldsymbol{s}_i$ denote the sampling location in $\mathcal{D}$ of the ith observation of the response and the covariates. Let the observed data be denoted $\{y(\boldsymbol{s}_i), \boldsymbol{x}(\boldsymbol{s}_i)\}$, $i = 1, \ldots, n$. Then the data are a realization of the random fields at the sampling locations $\{Y(\boldsymbol{s}_i), \boldsymbol{X}(\boldsymbol{s}_i)\}$ for $i = 1, \ldots, n$.

For areal data, the spatial domain $\mathcal{D}$ is partitioned into n regions $\{D_1, \ldots, D_n\}$ such that $\mathcal{D} = \bigcup_{i=1}^{n} D_i$. In the case of areal data, the random variables $\{Y(D_i), \boldsymbol{X}(D_i)\}$ are defined for regions instead of for point locations; population and spatial mean temperature are examples of areal data. The analytical method described herein can be applied to areal data if they are recast as geostatistical data by assuming that the data are point-referenced to a representative location of each region, such as the centroid. That is, $\{\boldsymbol{X}(\boldsymbol{s}_i), Y(\boldsymbol{s}_i)\}$ where $\boldsymbol{s}_i$ is the centroid of D_i for $i = 1, \ldots, n$.

Common practice in the analysis of geostatistical and areal data is to model the response variable with a spatial linear regression model consisting of the sum of a fixed mean function, a spatial random effect, and random error all on domain $\mathcal{D}$, as in:

$$Y(\boldsymbol{s}) = \boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta} + W(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \tag{1}$$

where $\boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta}$ is the mean function consisting of a vector of covariates $\boldsymbol{X}(\boldsymbol{s})$, and a vector of regres-

sion coefficients $\boldsymbol{\beta}$. The random error $\varepsilon(\boldsymbol{s})$ denotes white noise such that the errors are independent and identically distributed with mean zero and variance σ^2, while the random component $W(\boldsymbol{s})$ denotes a mean-zero, second-order stationary random field that is independent of the random error. The mean function captures the large-scale systematic trend of the response, the spatial random field $W(\boldsymbol{s})$ can be thought of as a small-scale spatial random effect, and the error term $\varepsilon(\boldsymbol{s})$ captures micro-scale variation (Cressie, 1993).

It is common to pre-specify the form of a covariance function for the spatial random effect $W(\boldsymbol{s})$ (Diggle and Ribeiro, 2007). For example, the exponential covariance function (a special case of the Matérn class of covariance functions) has the form

$$\text{Cov}(W(\boldsymbol{s}), W(\boldsymbol{t})) = \exp\left\{-\phi^{-1}\delta(\boldsymbol{s}, \boldsymbol{t})\right\} \tag{2}$$

where ϕ denotes a range parameter and $\delta(\boldsymbol{s}, \boldsymbol{t})$ denotes the Euclidean distance between locations $\boldsymbol{s}$ and $\boldsymbol{t}$. The general form of a covariance function in the Matérn class is

$$\text{Cov}(W(\boldsymbol{s}), W(\boldsymbol{t})) = \left\{\Gamma(\nu)2^{\nu-1}\right\}^{-1} \left\{\delta(\boldsymbol{s}, \boldsymbol{t})\phi^{-1}\sqrt{2\nu}\right\}^{\nu} K_\nu \left(\delta(\boldsymbol{s}, \boldsymbol{t})\phi^{-1}\sqrt{2\nu}\right) \tag{3}$$

where ν denotes the degree of smoothness, K_ν denotes the modified Bessel equation of the second kind, and as before ϕ denotes a range parameter and $\delta(\boldsymbol{s}, \boldsymbol{t})$ the Euclidean distance between locations $\boldsymbol{s}$ and $\boldsymbol{t}$. The exponential covariance function corresponds to a Matérn class covariance function with $\nu = 1/2$.

A random field is said to be stationary if the joint distribution of a the response at a finite set of locations does not change when the set of locations are all shifted in space by a fixed spatial lag. That is, letting $\{T(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ be a random field on spatial domain $\mathcal{D}$ that takes value $T(\boldsymbol{s}_i)$ at location $\boldsymbol{s}_i \in \mathcal{D}$ for $i = 1, \ldots, n$, the random field $T(\boldsymbol{s})$ is stationary if $F_n\left(T(\boldsymbol{s}_1), \ldots, T(\boldsymbol{s}_n)\right) =$

$F_n\left(T(\boldsymbol{s}_1+\boldsymbol{h}),\ldots,T(\boldsymbol{s}_n+\boldsymbol{h})\right)$ where $F_n(\cdot)$ is the joint distribution of a length n sample from $T(\boldsymbol{s})$ and $\boldsymbol{h}$ is a fixed spatial lag. The random field $\{T(\boldsymbol{s}):\boldsymbol{s}\in\mathcal{D}\}$ is second-order stationary if the following are satisfied:

$$E\left\{T(\boldsymbol{s})\right\}=\mu \text{ for all } \boldsymbol{s}\in\mathcal{D}$$

$$\text{var}\left\{T(\boldsymbol{s})\right\}=\sigma^2<\infty \text{ for all } \boldsymbol{s}\in\mathcal{D}$$

$$\text{cov}\left\{T(\boldsymbol{s}),T(\boldsymbol{s}+\boldsymbol{h})\right\}=C(\boldsymbol{h}) \tag{4}$$

where the function $C(\cdot)$ depends only on the spatial lag $\boldsymbol{h}$ and not on the location $\boldsymbol{s}$.

The coefficient vector $\boldsymbol{\beta}$ in (1) is a fixed constant. The model can be made more flexible if the coefficients are described by a stationary random field. Such a model is written

$$Y(\boldsymbol{s})=\boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta}(\boldsymbol{s})+\varepsilon(\boldsymbol{s}) \tag{5}$$

where $\boldsymbol{\beta}(\boldsymbol{s})$ is a random coefficient field with a Matérn-class covariance function and the spatial random effect $W(\boldsymbol{s})$ included in the intercept $\beta_0(\boldsymbol{s})$. The random coefficient field $\boldsymbol{\beta}(\boldsymbol{s})$ can be estimated by Markov Chain Monte Carlo (MCMC) methods under the assumption that $\boldsymbol{\beta}(\boldsymbol{s})$ is stationary (Gelfand et al., 2003).

Alternatively, kernel-based and spline-based methods can be considered for fitting VCR models without assuming the coefficients are described by a stationary random field. For example, it is straightforward to modify a thin plate regression spline model into a traditional, non-spatial VCR model (Wood, 2006). A local likelihood can also be used to fit generalized linear models with varying coefficients using kernel smoothing (Loader, 1999). Fan and Zhang (1999) demonstrated that the optimal kernel bandwidth estimate for a VCR model can be found via a two-step technique.

Model selection in VCR models may be local or global. Global selection means including or excluding variables everywhere in the spatial domain, while local selection means including or excluding variables at individual locations within the spatial domain. For global model selection in spline-based VCR models, Wang et al. (2008) proposed a SCAD penalty (Fan and Li, 2001) for variable selection in spline-based VCR models with a univariate effect-modifying variable. Antoniadas et al. (2012) used the nonnegative Garrote penalty (Breiman, 1995) in P-spline-based VCR models having a univariate effect-modifying variable.

Wavelet methods for fitting SVCR models were explored by Shang (2011) and Zhang and Clayton (2011). Sparsity in the wavelet coefficients is achieved either by ℓ_1-penalization (also known as the Lasso (Tibshirani, 1996)) (Shang, 2011) or by Bayesian variable selection (Zhang and Clayton, 2011). Sparsity in the wavelet domain does not imply sparsity in the covariates, though, so neither method is suitable for local variable selection.

Geographically weighted regression (GWR) is a kernel-based method for estimating the coefficients of an SVCR model where the kernel weights are based on the distance between sampling locations (Brundson et al., 1998; Fotheringham et al., 2002). At each sampling location, traditional GWR estimates the local regression coefficients by the local likelihood (Loader, 1999). As a kernel-based smoother for regression coefficients, traditional GWR tends to exhibit bias near the boundary of the region being modeled (Hastie and Loader, 1993). One way to reduce the boundary-effect bias is to model the coefficient surface as locally linear rather than locally constant by including coefficient-by-location interactions (Wang et al., 2008).

Traditional GWR relies on *a priori* global model selection to decide which variables should be included in the model. The idea of using Lasso regularization for local variable selection in a

GWR model appears in the literature as the geographically weighted Lasso (GWL) (Wheeler, 2009). The GWL applies the Lasso for local variable selection and uses a jackknife criterion for selection of the Lasso tuning parameters. Because the jackknife criterion can only be computed at sampling locations where the response variable is observed, the GWL cannot be used to impute missing values of the response variable nor to interpolate the coefficient surface and/or the response variable between sampling locations.

Lasso regularization for model selection, while popular, can leave relevant covariates out of the model when they are correlated with other covariates, and the predictive performance of the Lasso may be dominated in such a case by ridge regression, which does not allow for local model selection (Tibshirani, 1996). The elastic net is a regularization method that combines a ℓ_1 (Lasso) and a ℓ_2 (ridge) penalty on the estimated coefficients, overcoming these drawbacks of the Lasso (Zou and Hastie, 2005).

Additionally, Lasso regularization does not generally produce consistent estimates of the relevant covariates (Leng et al., 2006). The adaptive Lasso (AL) (Zou, 2006) is an improvement to the Lasso that does produce consistent estimates of the coefficients and has been shown to have appealing properties for automating variable selection, which under suitable conditions include the "oracle" property of asymptotically selecting exactly the correct set of covariates for inclusion in a regression model.

Combining these improvements to the Lasso, the adaptive elastic net (AEN) achieves an oracle property and performs better than other oracle-like methods when there is collinearity in the covariates (Zou and Zhang, 2009).

This document introduces a new regularization method, called the geographically weighted elastic

net (GWEN), for local variable selection in GWR models. Model selection under the GWEN uses the AEN. A penalized-likelihood criterion is used to select the local GWEN tuning parameters, which means that a GWEN can be fit at any location within the domain, whether or not data were observed at that location. The particular information criterion used to select the GWEN tuning parameters is a type of local BIC, but in principle another information criterion like the AIC is also possible. The local BIC presented here is based on the local likelihood (Loader, 1999).

The remainder of this document is organized as follows. The traditional GWR is presented in Section 2. The new GWEN is introduced in section 3. In Section 4, a simulation study is conducted to assess the performance of the GWEN in variable selection and coefficient estimation. An application of the GWEN to real data is presented in Section 5. Planned future improvements to the GWEN are discussed in Section 6.

2. Geographically weighted regression

2.1. Model

Consider n data observations, taken at sampling locations $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n$ in a spatial domain $D \subset \mathbb{R}^2$. For $i = 1, \ldots, n$, let $y(\boldsymbol{s}_i)$ and $\boldsymbol{x}(\boldsymbol{s}_i)$ denote the univariate response variable, and a $(p+1)$-variate vector of covariates measured at location $\boldsymbol{s}_i$, respectively. At each location $\boldsymbol{s}_i$, assume that the outcome is related to the covariates by a linear model where the coefficients $\boldsymbol{\beta}(\boldsymbol{s}_i)$ may be spatially-varying and $\varepsilon(\boldsymbol{s}_i)$ is random error at location $\boldsymbol{s}_i$. That is,

$$y(\boldsymbol{s}_i) = \boldsymbol{x}(\boldsymbol{s}_i)'\boldsymbol{\beta}(\boldsymbol{s}_i) + \varepsilon(\boldsymbol{s}_i). \tag{6}$$

Further assume that the error term $\varepsilon(\boldsymbol{s}_i)$ is normally distributed with zero mean and variance σ^2,

and that $\varepsilon(\boldsymbol{s}_i)$, $i = 1, \ldots, n$ are independent. That is,

$$\varepsilon(\boldsymbol{s}_i) \overset{iid}{\sim} \mathcal{N}\left(0, \sigma^2\right). \tag{7}$$

In order to simplify the notation, let $\boldsymbol{x}(\boldsymbol{s}_i) \equiv \boldsymbol{x}_i \equiv (1, x_{i1}, \ldots, x_{ip})'$, $\boldsymbol{\beta}(\boldsymbol{s}_i) \equiv \boldsymbol{\beta}_i \equiv (\beta_{i0}, \beta_{i1}, \ldots, \beta_{ip})'$, and $y(\boldsymbol{s}_i) \equiv y_i$. Equations (6) and (7) can now be rewritten as

$$y_i = \boldsymbol{x}_i'\boldsymbol{\beta}_i + \varepsilon_i \text{ and } \varepsilon_i \overset{iid}{\sim} \mathcal{N}\left(0, \sigma^2\right). \tag{8}$$

Further, let $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)'$ and $\boldsymbol{y} = (y_1, \ldots, y_n)'$. Thus, conditional on the design matrix $\boldsymbol{X}$, observations of the response variable at different locations are independent of each other. Then, a total log-likelihood of the observed data is the sum of the log-likelihood of each individual observation:

$$\ell\left(\boldsymbol{\beta}\right) = -\left(1/2\right)\left\{n \log\left(2\pi\sigma^2\right) + \left(\sigma^2\right)^{-1} \sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i'\boldsymbol{\beta}_i\right)^2\right\}. \tag{9}$$

Since there are a total of $n \times (p+1)$ free parameters for n observations, the model is not identifiable and it is not possible to directly maximize the total likelihood. One way to effectively reduce the number of parameters is to assume that the coefficients $\boldsymbol{\beta}(\boldsymbol{s})$ are smoothly varying over space, and use a kernel smoother to make pointwise estimates of the coefficients by maximizing a local likelihood. In the setting of spatial data and with the kernel smoother based on the physical distance between sampling locations, this is the traditional GWR.

2.2. Estimation

In the traditional GWR, the coefficient surface $\boldsymbol{\beta}(\boldsymbol{s})$ is estimated at each sampling location $\boldsymbol{s}_i$. First calculate the Euclidean distance $\delta_{ii'} \equiv \delta\left(\boldsymbol{s}_i, \boldsymbol{s}_{i'}\right) \equiv \|\boldsymbol{s}_i - \boldsymbol{s}_{i'}\|_2$ between locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_{i'}$ for all

$i, i' = 1, \ldots, n$. The bisquare kernel can be used to generate spatial weights based on the Euclidean distances and a bandwidth parameter ϕ:

$$w_{ii'} = \begin{cases} \left\{1 - \left(\phi^{-1}\delta_{ii'}\right)^2\right\}^2 & \text{if } \delta_{ii'} < \phi, \\ 0 & \text{if } \delta_{ii'} \geqslant \phi. \end{cases} \tag{10}$$

The bisquare kernel in (10) assigns the maximum weight of one where $\boldsymbol{s}_i = \boldsymbol{s}_{i'}$ (i.e. $\delta_{ii'} = 0$), is continuously differentiable, and assigns zero weight to observations at distances greater than one bandwidth from $\boldsymbol{s}_i$. For the purpose of estimation, define a local likelihood at each sampling location:

$$\mathcal{L}_i\left(\boldsymbol{\beta}_i\right) = \prod_{i'=1}^{n}\left[\left(2\pi\sigma_i^2\right)^{-1/2}\exp\left\{-\left(2\sigma_i^2\right)^{-1}\left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2\right\}\right]^{w_{ii'}}, \tag{11}$$

where σ_i^2 is a local approximation to the error variance σ^2 (Fotheringham et al., 2002). Thus, the local log-likelihood function is, up to an additive constant:

$$\ell_i\left(\boldsymbol{\beta}_i\right) = -(1/2)\sum_{i'=1}^{n} w_{ii'}\left\{\log\sigma_i^2 + \left(\sigma_i^2\right)^{-1}\left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2\right\}. \tag{12}$$

From (11) and (12), it is apparent that the GWR coefficient estimates $\hat{\boldsymbol{\beta}}_{i,\mathrm{GWR}}$ maximizing the local likelihood at location $\boldsymbol{s}_i$ can be obtained using weighted least squares.

Setting to zero the derivative of (12) with respect to σ_i^2, the maximum likelihood estimate (MLE) $\hat{\sigma}_i^2$ is found to be:

$$\hat{\sigma}_i^2 = \left(\sum_{i'=1}^{n} w_{ii'}\right)^{-1}\sum_{i'=1}^{n} w_{ii'}\left(y_{i'} - \boldsymbol{x}_{i'}'\hat{\boldsymbol{\beta}}_i\right)^2 \tag{13}$$

3. Local variable selection and parameter estimation

3.1. Local variable selection

Both the AL and the AEN are explored as penalty functions for local variable selection in GWR models.

The proposed local variable selection with AL penalty is an ℓ_1 regularization method for variable selection in regression models (Zou, 2006). Unlike the traditional Lasso penalty, which applies an equal penalty to each covariate in the local model at $\boldsymbol{s}_i$, the AL adjusts the penalty of each covariate based on the covariate's unpenalized local coefficient.

The proposed local variable selection with AEN penalty generalizes the AL penalty to include an additional ridge penalty (Zou and Zhang, 2009). Ridge regression is an ℓ_2 regularization technique that differs from the Lasso in that the ridge penalty is applied to the sum of the squared local regression coefficients (Hoerl and Kennard, 1970). The ridge penalty is used to estimate coefficients in regression models with correlated covariates because it stabilizes the inversion of the covariance matrix, which improves the robustness of the coefficient estimates (Hastie et al., 2009).

In fact, since the AL is an ℓ_1 regularization method while the AEN is a combined ℓ_1 and ℓ_2 regularization method, the AL can be viewed as a special case of the AEN where the ℓ_2 penalty is set to zero. When the ℓ_2 penalty is set to zero, the GWEN becomes the geographically weighted adaptive Lasso (GWAL).

3.1.1. Local variable selection and coefficient estimation with the adaptive Lasso

The objective function for the GWAL at $\boldsymbol{s}_i$ consists of the local log-likelihood and an additive penalty that is the weighted ℓ_1-norm of the coefficients, defined to be

$$\begin{aligned}\mathcal{S}(\boldsymbol{\beta}_i) &= -2\ell_i(\boldsymbol{\beta}_i) + \mathcal{J}_1(\boldsymbol{\beta}_i) \\ &= \sum_{i'=1}^{n} w_{ii'} \left\{ \log \sigma_i^2 + \left(\sigma_i^2\right)^{-1} \left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2 \right\} + \lambda_i^* \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij} \end{aligned} \tag{14}$$

where $\sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2$ is the weighted sum of squares minimized by traditional GWR, and $\mathcal{J}_1(\boldsymbol{\beta}_i) = \lambda_i \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij}$ is the AL penalty. With the vector of unpenalized local coefficients $\boldsymbol{\gamma}_i$, the AL penalty for the jth coefficient β_{ij} at location $\boldsymbol{s}_i$ is λ_i/γ_{ij}, where $\lambda_i > 0$ is a the local tuning parameter that applies to all coefficients at location $\boldsymbol{s}_i$ and $\boldsymbol{\gamma}_i = (\gamma_{i1}, \ldots, \gamma_{ip})'$ is the vector of adaptive weights at location $\boldsymbol{s}_i$.

3.1.2. Local variable selection and coefficient estimation with the adaptive elastic net

The objective function for the local geographically weighted elastic net (GWEN) method at $\boldsymbol{s}_i$ consists of the sum of the log-likelihood and an additive penalty that is a sum of weighted ℓ_1- and ℓ_2-norms of the coefficients, defined to be

$$\begin{aligned}\mathcal{S}(\boldsymbol{\beta}_i) &= -2\ell_i(\boldsymbol{\beta}_i) + \mathcal{J}_2(\boldsymbol{\beta}_i) \\ &= \sum_{i'=1}^{n} w_{ii'} \left\{ \log \sigma_i^2 + \left(\sigma_i^2\right)^{-1} \left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2 \right\} + \alpha_i \lambda_i \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij} + (1-\alpha_i)\lambda_i^* \sum_{j=1}^{p} \left(\beta_{ij}/\gamma_{ij}\right)^2 \end{aligned} \tag{15}$$

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

Fitting a SVCR model by the GWEN requires selecting the vector of elastic net parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)'$. In the simulation study (Section 4), the elastic net parameter is chosen globally ($\alpha_i \equiv \alpha$ for $i = 1, \ldots, n$). The global elastic net parameter is calculated as $\alpha = 1 - \rho_{\max}$ where $\rho_{\max}$ is the maximum global (i.e. for all data without weighting) Pearson correlation between any two covariates.

3.2. Tuning parameter selection

A local tuning parameter λ_i (or λ_i^*) is required for the variable selection step of fitting each local model by the GWAL (or GWEN) method. To select λ_i, we propose a locally-weighted version of the Bayesian Information Criterion (BIC) (Schwarz, 1978) which we call the local BIC ($\mathrm{BIC}_{\mathrm{loc}}$):

$$\begin{aligned}
\mathrm{BIC}_{\mathrm{loc},i} &= -2 \sum_{i'=1}^{n} \ell_{ii'} + \log \left(\sum_{i'=1}^{n} w_{ii'} \right) \mathrm{df}_i \\
&= -2 \times \sum_{i'=1}^{n} \log \left[\left(2\pi \hat{\sigma}_i^2 \right)^{-1/2} \exp \left\{ -\frac{1}{2} \hat{\sigma}_i^{-2} \left(y_{i'} - \boldsymbol{x}_{i'}' \hat{\boldsymbol{\beta}}_{i'} \right)^2 \right\} \right]^{w_{ii'}} + \log \left(\sum_{i'=1}^{n} w_{ii'} \right) \mathrm{df}_i \\
&= \sum_{i'=1}^{n} w_{ii'} \left\{ \log (2\pi) + \log \hat{\sigma}_i^2 + \hat{\sigma}_i^{-2} \left(y_{i'} - \boldsymbol{x}_{i'}' \hat{\boldsymbol{\beta}}_{i'} \right)^2 \right\} + \log \left(\sum_{i'=1}^{n} w_{ii'} \right) \mathrm{df}_i \qquad (16)
\end{aligned}$$

The local BIC is calculated by adding a penalty to the local likelihood, with the sum of the weights around $\boldsymbol{s}_i$, $\sum_{i'=1}^{n} w_{ii'}$, playing the role of the sample size and the "degrees of freedom" (df_i) at $\boldsymbol{s}_i$ given by the number of nonzero coefficients in $\boldsymbol{\beta}_i$ (Zou et al., 2007). Since the estimated variance $\hat{\sigma}_i^2$ is the variance estimate from the unpenalized local model, its value does not depend on the choice of tuning parameter; it is constant in (16) (Zou et al., 2007).

For the geographically weighted Lasso (GWL), Wheeler (2009) proposed selecting the local Lasso tuning parameters for local selection in a SVCR model at location $\boldsymbol{s}_i$ to minimize the local jackknife

prediction error $|y_i - \hat{y}_i^{(i)}|$. Because the jackknife prediction error is undefined everywhere except for at the sampling locations, this choice restricts coefficient estimation to occur at the locations where data has been observed. By contrast, the local BIC can be calculated at any location where the local log-likelihood can be obtained. As a practical matter this allows for variable selection and coefficient surface estimation to be done at locations where no data are observed and for imputation of missing values of the response variable.

3.3. Coefficient estimation

The locally linear coefficients maximize the unpenalized local likelihood:

$$\hat{\boldsymbol{\beta}}_i = \underset{\boldsymbol{\beta}_i}{\operatorname{argmax}} \left[-(1/2) \sum_{i'=1}^{n} w_{ii'} \left\{ \log \sigma_i^2 + \sigma_i^{-2} \left(y_{i'} - \boldsymbol{z}_{i'}' \boldsymbol{\beta}_i \right)^2 \right\} \right]. \tag{17}$$

where $Z_i = (z_1 \cdots z_n)'$ is the augmented local design design matrix at location $\boldsymbol{s}_i$, consisting of the original design matrix X reduced to the covariates selected locally by the GWEN or GWAL and augmented with covariate-by-location interactions. Letting $\tilde{X}_i$ be the reduced local design matrix consisting of the columns of X that correspond to covariates that are selected for inclusion in the local model at location $\boldsymbol{s}_i$, the augmented local design matrix is

$$\boldsymbol{Z}_i = \left(\tilde{X}_i \;\; L_i \tilde{X}_i \;\; M_i \tilde{X}_i \right) \tag{18}$$

where $L_i = \operatorname{diag}\{s_{i',x} - s_{i,x}\}$ and $M_i = \operatorname{diag}\{s_{i',y} - s_{i,y}\}$

3.4. Bandwidth parameter estimation

The fitted values from the model are

$$\hat{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y} \tag{19}$$

Where $\boldsymbol{H} = (H_1 \cdots H_n)'$ and H_i denotes the ith row of the matrix $\boldsymbol{W}_i^{1/2} Z_i \left(Z_i' \boldsymbol{W}_i \tilde{\boldsymbol{X}} \right)^{-1} Z_i' \boldsymbol{W}_i^{1/2}$.

The bandwidth parameter ϕ in (10) is selected globally by minimizing the corrected AIC:

$$\mathrm{AIC_c} = 2n \log \sigma + n \left\{ \frac{n+\nu}{n-2-\nu} \right\} \tag{20}$$

where ν is the trace of the linear smoothing matrix $\boldsymbol{H}$, and approximates the total degrees of freedom of the SVCR (Hurvich et al., 1998).

4. Simulation

4.1. Simulation setup

A simulation study was conducted to assess the performance of the method described in Sections 2–3.

Data were simulated on the domain $[0,1]^2$, which was divided into a 30×30 grid. Each of $p = 5$ covariates $X_1, \ldots, X_5$ was simulated by a Gaussian random field (GRF) with mean zero and exponential covariance function $\mathrm{Cov}\left(X_{ji}, X_{ji'}\right) = \sigma_x^2 \exp\left(-\tau_x^{-1} \delta_{ii'}\right)$ where $\sigma_x^2 = 1$ is the variance, $\tau_x = 0.1$ is the range parameter, and $\delta_{ii'}$ is the Euclidean distance $\|\boldsymbol{s}_i - \boldsymbol{s}_{i'}\|_2$.

Correlation was induced between the covariates by multiplying the matrix $\boldsymbol{X} = (X_1 \cdots X_5)$ by $\boldsymbol{R}$, where $\boldsymbol{R}$ is the Cholesky decomposition of the covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{R}'\boldsymbol{R}$. The covariance matrix $\boldsymbol{\Sigma}$ is a 5×5 matrix that has ones on the diagonal and ρ for all off-diagonal entries, where ρ is the between-covariate correlation.

The simulated response was $y_i = \boldsymbol{x}_i'\boldsymbol{\beta}_i + \varepsilon_i$ for $i = 1, \ldots, n$ where $n = 900$ and the ε_i's were iid Gaussian with mean zero and variance σ_ε^2. The simulated data included the response y and five covariates $x_1, \ldots, x_5$. The true data-generating model uses only x_1. The variables $x_2, \ldots, x_5$ are included to assess performance in model selection.

There were twelve simulation settings, each of which was simulated 100 times. For each of the twelve settings, $\beta_1(\boldsymbol{s})$, the true coefficient surface for x_1, was nonzero in at least part of the domain, with a minimum of zero and maximum of one. Three parameters were varied to produce the twelve settings: there were three functional forms for the coefficient surface $\beta_1(\boldsymbol{s})$, data was simulated both with ($\rho = 0.5$) and without ($\rho = 0$) correlation between the covariates, and simulations were made with low ($\sigma^2_\varepsilon = 0.25$) and high ($\sigma^2_\varepsilon = 1$) variance for the random error term. The twelve simulation settings are described in Table 1.

The three coefficient surfaces used to produce the response variable in the simulations are pictured in Figure 1. The first is a "step" function, which is equal to zero in 40% of the spatial domain, equal to one in a different 40% of the spatial domain, and increases linearly in the middle 20% of the domain. The second is a gradient function, which increases linearly from zero at one end of the domain to one at the other. The final coefficient function is a parabola taking its maximum value of 1 at the center of the domain and falling to zero at each corner of the domain.

The performance of the GWEN (with both the AL and the AEN used for model selection) was compared to that of the traditional GWR algorithm of Fotheringham et al. (2002) and that of "oracular" GWR, which is traditional GWR with oracular variable selection and locally linear fitting. Oracular selection means that exactly the correct set of covariates was used to fit the GWR model at each location in the simulation. Finally, there is a category of simulation results using the three penalized GWR methods for local variable selection and then ordinary GWR for coefficient estimation. The implementation of the AEN uses coordinate descent via the `R` package `glmnet` (Friedman et al., 2010).

Results of the simulation were summarized at five locations on the domain (Figure 2). Due to

edge effects, we expect biased estimation at locations one and five (which are at opposite corners of the domain) from traditional GWR, particularly when the coefficient surface has nonzero gradient at the boundary (which is the case for the gradient and parabola functions). Because the GWEN and oracular GWR use locally linear fitting, they are expected to exhibit less bias at the boundaries.

Locations two and four are at the 'corners' of the step function. Because the step function is undifferentiable at these locations, locally linear fitting is not expected to be as effective at reducing bias here as at the boundaries of the gradient and parabola functions.

Local variable selection is expected to be ambiguous at locations where the underlying coefficient surface transitions from zero to nonzero. In the simulations, that occurs at location four of the step function, location five of the gradient, and locations one and five of the parabola.

Unlike the other two functions, the gradient is actually constant across the domain in terms of the covariate-by-location interaction. As a result, the optimal kernel bandwidth ϕ is expected to be larger for estimating the gradient coefficient surface than for the step or the parabola. The result should be that the estimation is more accurate for the gradient function in terms of bias, variance, and MSE.

4.2. Simulation results

Variable selection. Table 2 lists the results of variable selection. The correct covariate was usually included in the local models, and the unimportant covariates were usually excluded. Ignore for now the ambiguous locations where the true β_1 surface transitions from zero to nonzero. Of the eighty simulated cases where $\beta_1(\boldsymbol{s})$ is unambiguously nonzero, more than half (59) saw no false negatives (over 100 simulations). The number with no false negatives and no false positives (i.e. exactly

the correct model was recovered in all 100 simulations) was 44. Of the 120 total simulated cases, 72 had no false positives (i.e. no variable whose true coefficient is zero was included in the model during any of the 100 simulation runs).

Selection performance was more affected by an increase in the noise variance from $\sigma_\varepsilon^2 = 0.25$ to $\sigma_\varepsilon^2 = 1$ than by an increase in colinearity from $\rho = 0$ to $\rho = 0.5$. Of the 44 cases where model selection recovered exactly the correct model in all 100 runs of the simulation, only five arose from cases where $\sigma_\varepsilon^2 = 1$, while 19 arose from cases where $\rho = 0.5$. The worst error rates that were observed in these unambiguous cases were a false positive rate of 6% (location one of the step function with $\sigma_\varepsilon^2 = 1$, $\rho = 0.5$, and selection via the elastic net) and a false negative rate of 16% (location three of the step function with $\sigma_\varepsilon^2 = 1$, $\rho = 0.5$, and selection via the lasso).

Model selection was ambiguous at locations where the true $\beta_1(\boldsymbol{s})$ transitions from zero to nonzero. At location four of the step function, the selection rate of β_1 ranged from 43% to 60% among the different simulation settings. At location five of the gradient, the range of selection rates was 63% to 82%, and the selection rate across locations one and five of the parabola ranged from 27% to 66%.

There is no indication that the GWEN performed better in selection than the GWAL, even in cases where the covariates were moderately correlated ($\rho = 0.5$).

Coefficient estimation. The mean squared error, bias, and variance of $\hat{\beta}_1$ (MSE(β_1), bias($\hat{\beta}_1$), var($\hat{\beta}_1$)) are listed in Tables 3, 4, and 5, respectively. The method of oracular selection led to the best MSE($\hat{\beta}_1$) in 41 of the 60 cases.

In terms of MSE($\hat{\beta}_1$), while oracular selection clearly was the most accurate estimation method in

most cases, the difference in accuracy between the estimation methods was modest in most cases. There were a few cases when the difference in MSE($\hat{\beta}_1$) between estimation methods amounted to at least an order of magnitude. At locations one and five of the parabola, oracular selection produces much more accurate estimation than GWEN, GWAL, or GWR because locations one and five are on the domain boundary where the parabola has a strong gradient, and those methods don't use locally linear fits to account for the boundary effect. This can also be seen from the fact that the bias($\boldsymbol{\beta}_1$) of GWEN, GWAL, and GWR is large at locations one and five of the parabola, where it is nearly zero for `u.lasso`, `u.enet`, and `oracular`.

A similar boundary effect is apparent at location five of the gradient, where GWEN, GWAL, and GWR produce a bias($\hat{\beta}_1$) and MSE($\hat{\beta}_1$) that are an order of magnitude or more greater than those of `u.lasso`, `u.enet`, and `oracular` (the differences in var($\hat{\beta}_1$) are smaller).

At location one of the step function, the MSE($\boldsymbol{\beta}_1$) and var($\boldsymbol{\beta}_1$) for GWR are much smaller than for the other estimation methods, including `oracular`, while the bias($\boldsymbol{\beta}_1$) doesn't vary much between estimation methods. It is not clear why this is the case.

As was the case for selection, accuracy in coefficient estimation seemed to suffer more from an increase in the noise variance than from increased correlation in the covariates. Once again, this effect is probably most apparent at location three of the step function.

Fitted Values. The MSE of the $\hat{Y}$, MSE $\left(\hat{Y}\right)$, is listed in Table 6. Nominally, MSE $\left(\hat{Y}\right)$ should be equal to the noise variance, σ_ε^2, which is 1 for odd-numbered rows and 0.25 for even numbered rows. Of the 60 simulation cases, GWR produced the minimal residual variance for 22, which is more than any other method. Where the residual noise

5. Data analysis

An example data analysis is presented to demonstrate application of the GWEN for local model selection in an SVCR model of how poverty is related to a list of socio-economic variables.

5.1. Poverty data for the Upper Midwest

An example demonstrates application of the GWEN to the identification of socio-economic covariates that are locally meaningful predictors of the county-level poverty rate in the Upper Midwest states of the U.S. (Minnesota, Iowa, Wisconsin, Illinois, Indiana, and Michigan). The data used in this example are aggregated at the level of counties, which are areal units. Each county's centroid is treated as its sampling location. The data are from the U.S. Census Bureau's decennial census in the year 1970.

Three kinds of covariates were considered as potential predictors of county-level poverty rate.

- Covariates that describe the county's employment structure (`pag`, the proportion of residents employed in agriculture, `pex`, the proportion of residents employed in mining, `man`, the proportion of residents employed in manufacturing, `pfire`, the proportion of residents employed in finance, insurance, and real estate, `pserve`, the proportion of residents employed in services, and `potprof`, the proportion of residents employed in other professions)

- Covariates that describe the county's racial makeup (`pwh`, the proportion of residents who are white, `pblk`, the proportion of residents who are black, and `phisp`, the proportion of residents who are hispanic)

- `pmetro`: an indicator of whether the county is in a metropolitan area.

The response variable (poverty rate) is a proportion, taking values in $[0, 1]$. To demonstrate the geographically-weighted Lasso in a linear regression context, we model the logit-transformed poverty rate. The predictor variables were not transformed - raw proportions were used.

The GWEN was used for variable selection, and then coefficients for the selected variables were estimated by weighted least squares without shrinkage. Traditional GWR was used to fit a model to the same data for the sake of comparison.

The coefficient estimates are plotted on maps of the upper midwest in Figure 3 (based on the GWEN) and Figure 4 (based on traditional GWR).

It is immediately apparent that the estimated coefficient surfaces are non-constant for most covariates. The same large-scale patterns appear in both figures, but with differences. First of all, the GWEN has selected a larger bandwidth than traditional GWR, so there is less variability in the coefficient estimates from the GWEN. This may be one reason that the GWEN coefficient estimates are less extreme than those for traditional GWR. In a model with a logit-transformed proportion as the output, the coefficients can be interpreted as log odds ratios, so, e.g., the estimate of -100 as the coefficient of `phisp` (albeit at the edge of the domain) seems unrealistic.

Assessing variable selection for this data is difficult, since the GWEN almost never removed any variables from the model. Indeed, some coefficients seem nearly constant across the domain. An exception is the coefficient surface for `pex` (mining employment). That surface indicates an interaction whereby the proportion of people working in mining in southern parts of the domain is associated with an increase in the poverty rate, while in northern parts of the domain it is associated with a decrease in the poverty rate.

6. Future work

The GWEN is presented here as a method of analysis for data where the response variable follows a gaussian distribution with independent additive errors. A key feature of spatial data, though, is that the errors are typically autocorrelated. Additionally, it is common to encounter data that does not follow a gaussian distribution but for which the GWEN would otherwise be a valuable tool for analysis.

Autocorrelated Errors. In order to get a sense of how the GWEN will perform when the assumption of independent errors is violated, the simulation study from Section 4 was repeated with the addition of a Matèrn-class spatial covariance structure in the noise (results not included here). Introducing autocorrelation in the errors causes a substantial degradation in the estimation accuracy of the GWAL and GWEN, accompanied by a tendency to prefer smaller kernel bandwidths. The likely explanation is that when the kernel bandwidth is small, autocorrelated errors are indistinguishable from a varying intercept. The residuals are reduced to the extent that the errors are incorporated in the intercept term. Since this effect is absent in the case of uncorrelated errors, greater autocorrelation in the error term will tend to mean a greater reduction in the errors - and therefore a greater increase in the log likelihood - as the kernel bandwidth decreases.

Since the optimal kernel bandwidth is balance between the log likelihood and the degrees of freedom consumed by the model, and because the effect of autocorrelated errors is an increase in log likelihood without an offsetting increase in the consumed degrees of freedom at a given kernel bandwidth, greater autocorrelation will tend to lead to a smaller optimal kernel bandwidth. One quick way to counter this effect is to increase the penalty that is added to the total log likelihood in (20). There is currently no rule to set the penalty based on the data, which would be necessary

before using this adjustment to analyze real data.

Generalized linear models. To date, validation of the GWEN has been for gaussian data. The extension to any exponential-family distribution involves generalizing the likelihood that is used to select AEN tuning parameters and the kernel bandwidth. Preliminary simulations of the generalized GWEN have been carried out with Poisson and binomial data.

7. References

References

Antoniadas, A., I. Gijbels, and A. Verhasselt (2012). Variable selection in varying-coefficient models using p-splines. *Journal of Computational and Graphical Statistics 21*, 638–661.

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics 51*, 373–384.

Brundson, C., S. Fotheringham, and M. Charlton (1998). Geographically weighted regression: a natural evolution of the expansion method for spatial data analysis. *Environment and Planning A 30*, 1905–1927.

Cressie, N. (1993). *Statistics for Spatial Data (Revised Edition)*. Wiley, New York.

Diggle, P. and P. Ribeiro (2007). *Model-Based Geostatistics*. Springer New York.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association 96*, 1348–1360.

Fan, J. and W. Zhang (1999). Statistical estimation in varying coefficient models. *Annals of Statistics 27*, 1491–1518.

Fotheringham, A., C. Brunsdon, and M. Charlton (2002). *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. Wiley, West Sussex, England.

Friedman, J., T. Hastie, and R. Tibshirani (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software 33*, 1–22.

Gelfand, A. E., H.-J. Kim, C. F. Sirmans, and S. Banerjee (2003). Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association 98*, 387–396.

Hastie, T. and C. Loader (1993). Local regression: automatic kernel carpentry. *Statistical Science 8*, 120–143.

Hastie, T. and R. Tibshirani (1993). Varying-coefficient models. *Journal of the Royal Statistical Society Series B 55*, 757–796.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York.

Hoerl, A. E. and R. W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics 12*, 55–67.

Hurvich, C. M., J. S. Simonoff, and C.-L. Tsai (1998). Smoothing parameter selection in nonparametric regression using an improved akaike information criterion. *Journal of the Royal Statistical Society Series B 60*, 271–293.

Leng, C., Y. Lin, and G. Wahba (2006). A note on the lasso and related procedures in model selection. *Statistica Sinica 16*, 1273–1284.

Loader, C. (1999). *Local Regression and Likelihood*. Springer, New York.

Schwarz, G. (1978). Estimating the dimensions of a model. *Annals of Statistics 6*, 461–464.

Shang, Z. (2011). *Bayesian Variable Selection*. Ph. D. thesis, University of Wisconsin-Madison.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B 58*, 267–288.

Wang, L., H. Li, and J. Z. Huang (2008). Variable selection in nonparametric varying-coefficient models for analysis of repeated measurements. *Journal of the American Statistical Association 103*, 1556–1569.

Wang, N., C.-L. Mei, and X.-D. Yan (2008). Local linear estimation of spatially varying coefficient models: an improvement on the geographically weighted regression technique. *Environment and Planning A 40*, 986–1005.

Wheeler, D. C. (2009). Simultaneous coefficient penalization and model selection in geographically weighted regression: the geographically weighted lasso. *Environment and Planning A 41*, 722–742.

Wood, S. (2006). *Generalized Additive Models: An Introduction With R*. Chapman and Hall, Boca Raton, FL.

Zhang, J. and M. K. Clayton (2011). Functional concurrent linear regression model for images. *Journal of Agricultural, Biological, and Environmental Statistics 16*, 105–130.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association 101*, 1418–1429.

Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B 67*, 301–320.

Zou, H., T. Hastie, and R. Tibshirani (2007). On the "degrees of freedom" of the lasso. *Annals of Statistics 35*, 2173–2192.

Zou, H. and H. Zhang (2009). On the adaptive elastic net with a diverging number of parameters. *Annals of Statistics 37*, 1733–1751.

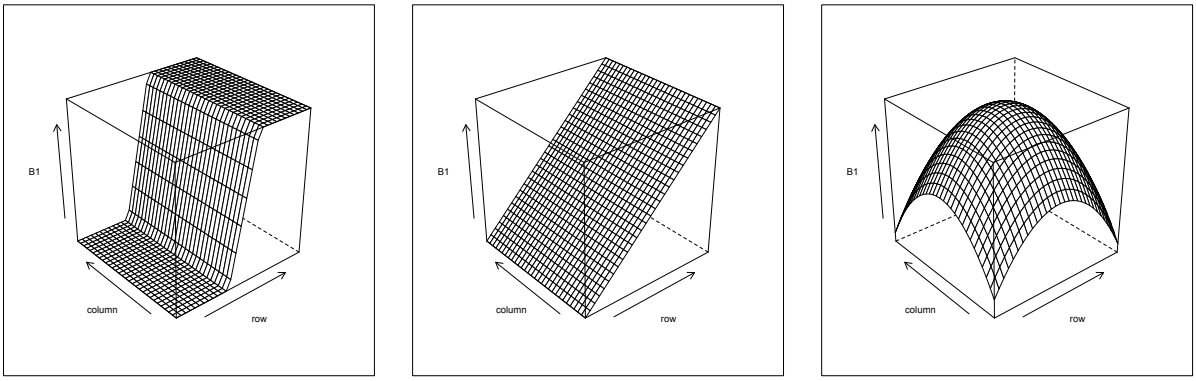

Figure 1: The actual β_1 coefficient surface used in the simulation.

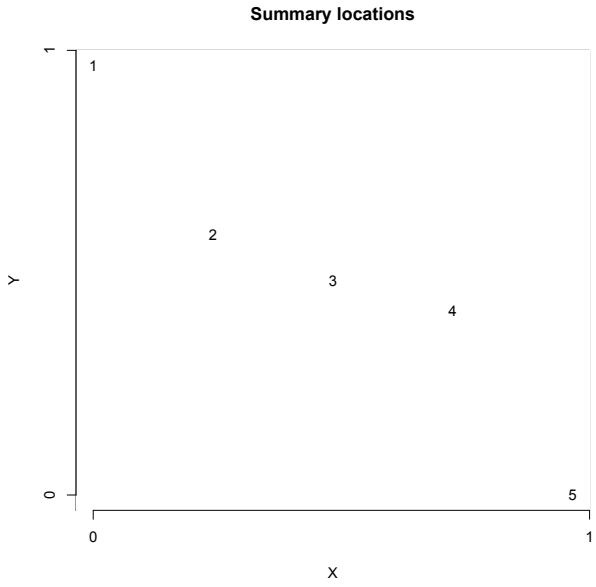

Figure 2: Locations where the variable selection and coefficient estimation of GWL were summarized.

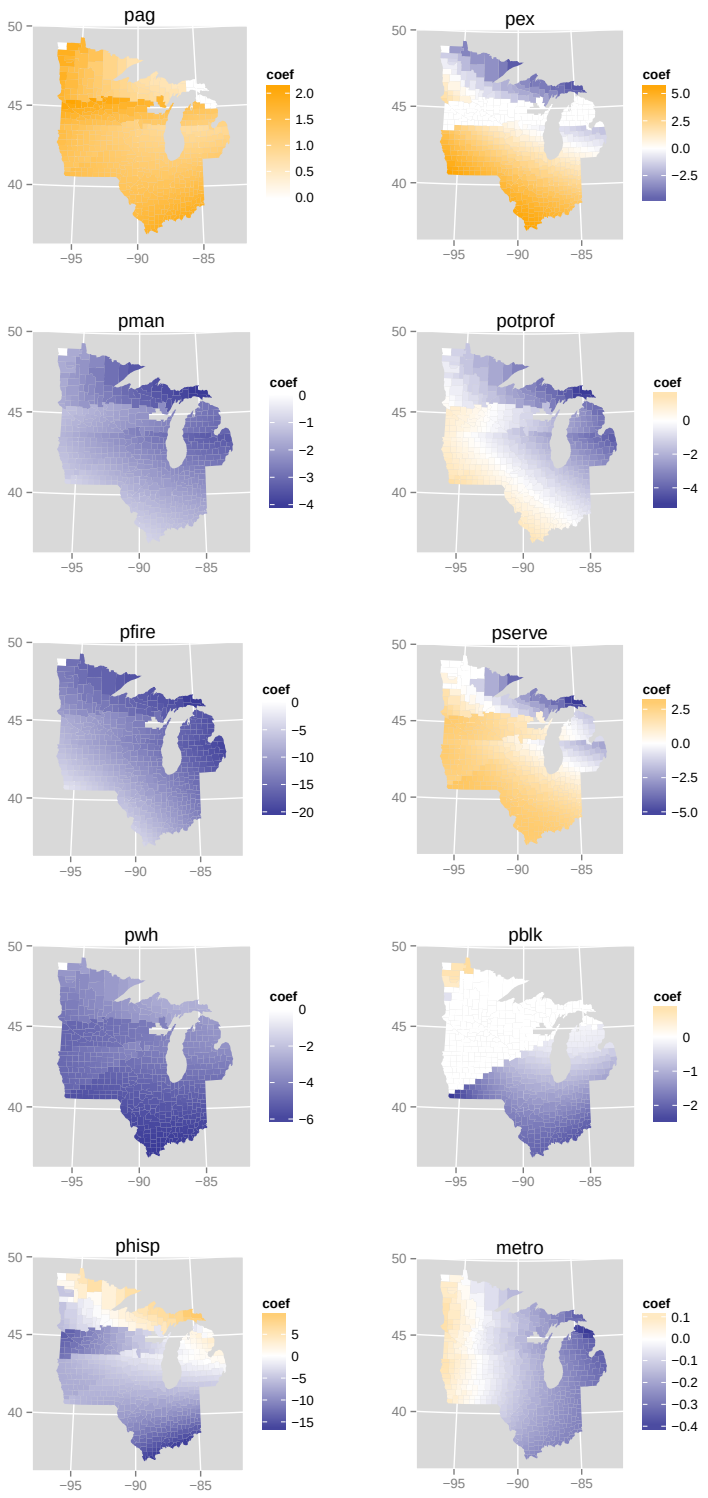

Figure 3: Coefficient surfaces for the logit of poverty rate, based on the 1970 census and selected by the GWEN.

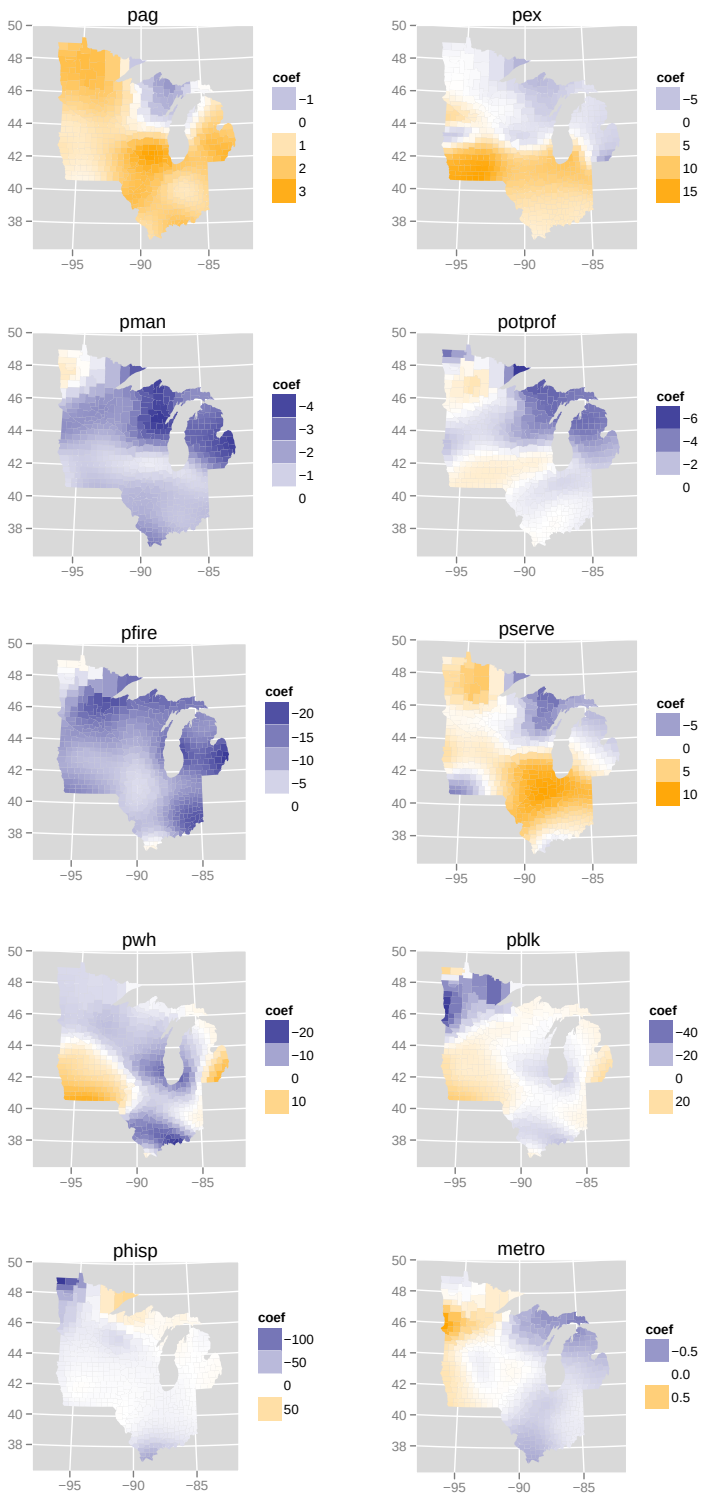

Figure 4: Coefficient surfaces for the logit of poverty rate based on the 1970 census and estimated by traditional GWR.

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 1: Simulation parameters for each setting.

	step				gradient				parabola			
	`enet`		`lasso`		`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	β_2 - β_5	β_1	β_2 - β_5
1	0.99	0.00	0.99	0.00	1.00	0.00	1.00	0.00	0.36	0.00	0.38	0.00
	0.99	0.02	0.99	0.02	1.00	0.01	1.00	0.01	0.71	0.02	0.70	0.02
	0.99	0.00	1.00	0.00	1.00	0.00	1.00	0.00	0.28	0.00	0.33	0.00
	0.96	0.05	0.91	0.04	0.99	0.03	0.99	0.01	0.56	0.02	0.55	0.02
2	1.00	0.00	1.00	0.00	1.00	0.01	1.00	0.00	1.00	0.00	1.00	0.00
	1.00	0.03	1.00	0.03	1.00	0.02	1.00	0.02	1.00	0.02	0.99	0.01
	1.00	0.01	1.00	0.00	1.00	0.01	1.00	0.01	1.00	0.00	1.00	0.00
	0.99	0.05	0.97	0.04	1.00	0.02	0.99	0.01	0.98	0.02	0.97	0.01
3	0.91	0.01	0.91	0.00	1.00	0.01	1.00	0.01	1.00	0.00	1.00	0.00
	0.96	0.05	0.96	0.05	1.00	0.01	1.00	0.01	1.00	0.00	1.00	0.00
	0.92	0.05	0.95	0.02	1.00	0.02	1.00	0.01	1.00	0.00	1.00	0.00
	0.92	0.08	0.87	0.05	1.00	0.02	0.98	0.02	0.99	0.01	0.99	0.01
4	0.48	0.01	0.43	0.01	1.00	0.01	1.00	0.01	1.00	0.00	1.00	0.00
	0.72	0.04	0.78	0.03	1.00	0.01	1.00	0.01	1.00	0.00	1.00	0.00
	0.49	0.02	0.46	0.02	1.00	0.02	1.00	0.00	1.00	0.00	1.00	0.00
	0.60	0.05	0.56	0.04	1.00	0.03	0.98	0.02	1.00	0.01	0.98	0.02
5	0.00	0.00	0.00	0.00	0.83	0.00	0.82	0.00	0.32	0.00	0.32	0.00
	0.03	0.01	0.02	0.00	0.70	0.00	0.66	0.00	0.68	0.02	0.73	0.02
	0.00	0.00	0.00	0.00	0.87	0.01	0.87	0.00	0.37	0.00	0.42	0.00
	0.06	0.02	0.01	0.02	0.61	0.01	0.62	0.02	0.61	0.04	0.58	0.03

Table 2: Selection frequency for the indicated variables.

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
		0.006	*0.004*	0.016	0.009	**0.000**	0.029
gradient	1	0.126	0.125	**0.005**	*0.005*	0.006	0.024
		0.105	0.102	*0.026*	0.027	**0.019**	0.042
		0.136	0.132	0.005	*0.005*	**0.005**	0.029
		0.135	0.119	*0.043*	0.044	**0.023**	0.055
	2	0.006	0.006	0.001	*0.001*	**0.001**	0.002
		0.006	0.006	0.002	*0.002*	**0.002**	0.003
		0.008	0.006	*0.001*	0.001	**0.001**	0.003
		0.009	0.011	*0.004*	0.008	**0.003**	0.006
	3	0.002	0.002	**0.000**	0.000	*0.000*	0.002
		0.003	0.003	0.001	*0.001*	**0.001**	0.003
		0.002	0.002	*0.000*	0.000	**0.000**	0.003
		0.006	0.010	*0.002*	0.007	**0.001**	0.007
	4	0.005	0.005	*0.000*	**0.000**	0.001	0.002
		0.007	0.006	0.002	*0.002*	**0.002**	0.004
		0.004	0.005	*0.000*	0.000	**0.000**	0.002
		0.009	0.010	*0.003*	0.006	**0.002**	0.009
	5	0.108	0.110	0.002	*0.002*	**0.000**	0.022
		0.084	0.084	0.011	*0.010*	**0.000**	0.044
		0.107	0.119	*0.003*	0.003	**0.000**	0.028
		0.065	0.076	*0.008*	0.010	**0.000**	0.056
parabola	1	0.050	0.054	0.019	*0.017*	**0.001**	0.123
		0.145	0.151	0.053	*0.053*	**0.001**	0.248
		0.029	0.046	0.016	*0.015*	**0.001**	0.133
		0.105	0.125	*0.065*	0.082	**0.001**	0.248
	2	**0.103**	0.104	0.105	0.106	*0.104*	0.105
		0.088	0.091	*0.085*	0.086	**0.079**	0.086
		0.092	0.100	*0.099*	0.100	0.100	0.104
		0.085	0.097	0.091	0.103	**0.077**	0.094
	3	*0.148*	0.150	0.156	0.157	0.156	**0.144**
		0.110	0.114	0.121	0.126	*0.108*	**0.086**
		0.139	*0.143*	0.150	0.150	0.152	0.144
		0.111	0.122	0.130	0.139	0.117	**0.101**
	4	**0.110**	0.112	0.115	0.116	0.115	*0.111*
		0.092	0.093	0.094	0.095	*0.085*	**0.085**
		0.104	0.111	0.113	0.114	0.114	*0.109*
		0.080	0.100	0.094	0.108	*0.088*	0.097
	5	0.044	0.047	*0.014*	0.016	**0.001**	0.123
		0.155	0.153	0.102	*0.101*	**0.001**	0.250
		0.040	0.060	*0.012*	0.018	**0.001**	0.136
		0.111	0.126	*0.055*	0.061	**0.001**	0.234

Table 3: Mean squared error of $\hat{\beta}_1$ (**minimum**, *next best*).

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
	5	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	-0.022
		0.003	*0.001*	-0.005	-0.003	**0.000**	-0.018
		0.000	**0.000**	**0.000**	**0.000**	**0.000**	0.003
		0.016	*0.006*	-0.007	0.010	**0.000**	0.012
gradient	1	-0.342	-0.341	*0.007*	0.007	**0.002**	-0.141
		-0.305	-0.299	**0.019**	*0.021*	0.024	-0.174
		-0.357	-0.352	-0.009	*-0.008*	**-0.003**	-0.151
		-0.340	-0.311	-0.026	*-0.019*	**-0.007**	-0.197
	2	-0.061	-0.061	0.001	**0.000**	*-0.001*	-0.003
		-0.049	-0.046	0.004	**0.004**	*0.004*	-0.011
		-0.073	-0.062	*-0.002*	**-0.001**	-0.002	-0.003
		-0.064	-0.054	*0.004*	0.004	**0.003**	-0.008
	3	0.003	0.006	*-0.001*	-0.002	-0.002	**0.001**
		-0.000	0.003	-0.005	-0.003	-0.003	*0.001*
		-0.001	0.009	0.002	0.002	*0.001*	0.005
		-0.023	-0.013	**-0.002**	-0.009	*0.003*	0.011
	4	0.047	0.050	-0.003	*-0.003*	-0.004	**0.002**
		0.028	0.032	-0.010	*-0.008*	**-0.007**	0.011
		0.043	0.055	*0.003*	**0.003**	0.003	0.010
		0.008	0.025	**-0.002**	-0.003	*-0.002*	0.029
	5	0.293	0.296	*-0.000*	0.000	**0.000**	0.126
		0.235	0.228	*0.013*	0.017	**0.000**	0.182
		0.298	0.318	0.003	*0.002*	**0.000**	0.137
		0.189	0.208	-0.004	*-0.001*	**0.000**	0.197
parabola	1	0.108	0.118	**0.014**	*0.020*	-0.034	0.336
		0.299	0.303	0.082	*0.081*	**-0.034**	0.479
		0.059	0.097	**0.002**	*0.017*	-0.034	0.339
		0.214	0.233	*0.052*	0.090	**-0.034**	0.466
	2	**0.316**	*0.318*	0.319	0.321	0.318	0.319
		0.288	0.283	0.281	*0.273*	**0.271**	0.286
		0.299	0.313	*0.311*	0.313	0.313	0.317
		0.248	0.266	*0.264*	0.273	0.267	0.290
	3	*0.382*	0.385	0.391	0.393	0.391	**0.378**
		0.316	0.319	0.331	0.335	0.317	**0.281**
		0.369	0.376	0.384	0.384	0.386	*0.375*
		0.298	0.310	0.326	0.336	0.326	*0.299*
	4	**0.329**	0.331	0.336	0.337	0.336	*0.330*
		0.294	0.294	0.295	0.295	*0.284*	**0.282**
		0.318	0.329	0.333	0.335	0.335	*0.327*
		0.262	*0.281*	0.290	0.290	0.285	0.297
	5	0.090	0.094	**0.001**	*0.006*	-0.034	0.329
		0.289	0.300	0.135	*0.125*	**-0.034**	0.479
		0.092	0.133	**0.012**	*0.019*	-0.034	0.342
		0.229	0.241	0.059	*0.058*	**-0.034**	0.449

Table 4: Bias of $\hat{\beta}_1$ (**minimum**, *next best*).

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
		0.006	*0.004*	0.016	0.009	**0.000**	0.029
gradient	1	0.009	0.009	*0.005*	0.005	0.006	**0.004**
		0.012	0.013	0.026	0.027	0.019	**0.012**
		0.008	0.008	0.005	*0.005*	**0.005**	0.007
		0.020	0.023	0.043	0.044	0.023	**0.016**
	2	0.003	0.002	0.001	*0.001*	**0.001**	0.002
		0.004	0.004	0.002	*0.002*	**0.002**	0.003
		0.003	0.003	*0.001*	0.001	**0.001**	0.003
		0.005	0.008	*0.004*	0.008	**0.003**	0.006
	3	0.002	0.002	*0.000*	0.000	**0.000**	0.002
		0.003	0.003	0.001	*0.001*	**0.001**	0.003
		0.002	0.002	*0.000*	0.000	**0.000**	0.003
		0.006	0.010	*0.002*	0.007	**0.001**	0.007
	4	0.003	0.003	*0.000*	**0.000**	0.000	0.002
		0.006	0.006	0.002	*0.002*	**0.002**	0.004
		0.003	0.002	*0.000*	0.000	**0.000**	0.002
		0.009	0.010	*0.003*	0.006	**0.002**	0.008
	5	0.022	0.023	0.003	*0.002*	**0.000**	0.006
		0.029	0.032	0.011	*0.010*	**0.000**	0.011
		0.018	0.019	*0.003*	0.003	**0.000**	0.009
		0.030	0.033	*0.008*	0.011	**0.000**	0.017
parabola	1	0.039	0.040	0.019	0.017	**0.000**	*0.010*
		0.057	0.059	0.047	0.047	**0.000**	*0.019*
		0.026	0.037	0.016	*0.015*	**0.000**	0.018
		0.060	0.071	0.063	0.074	**0.000**	*0.031*
	2	0.003	0.003	0.003	**0.003**	*0.003*	0.003
		0.005	0.011	0.006	0.011	0.005	**0.004**
		0.003	0.002	*0.002*	0.002	**0.002**	0.004
		0.024	0.027	0.021	0.029	**0.006**	*0.010*
	3	*0.002*	0.002	0.002	0.002	0.003	**0.002**
		0.010	0.013	0.011	0.014	**0.007**	*0.008*
		0.002	**0.002**	0.003	0.003	0.003	0.003
		0.023	0.026	0.024	0.026	**0.011**	*0.012*
	4	0.002	0.002	**0.002**	*0.002*	0.002	0.002
		0.006	0.006	0.007	0.008	**0.005**	*0.005*
		0.003	0.002	**0.002**	*0.002*	0.002	0.003
		0.012	0.021	0.010	0.024	**0.007**	*0.009*
	5	0.036	0.038	*0.014*	0.016	**0.000**	0.015
		0.072	0.063	0.084	0.086	**0.000**	*0.021*
		0.031	0.043	*0.012*	0.017	**0.000**	0.020
		0.060	0.069	0.052	0.058	**0.000**	*0.032*

Table 5: Variance of $\hat{\beta}_1$ (**minimum**, *next best*).

function	location	GWEN	GWAL	u.enet	u.lasso	oracle	GWR
step	1	*0.164*	0.165	0.164	0.165	**0.159**	0.170
		0.779	0.799	*0.779*	0.799	0.974	0.949
		0.193	0.189	0.193	0.189	*0.181*	**0.172**
		0.839	*0.705*	0.839	**0.705**	0.949	0.984
	2	0.228	0.230	*0.228*	0.230	**0.226**	0.239
		0.888	*0.874*	0.888	0.874	0.916	**0.869**
		0.263	0.269	**0.263**	0.269	0.264	0.273
		1.177	**1.140**	1.177	**1.140**	1.229	1.177
	3	*0.241*	0.245	**0.241**	0.245	0.254	0.275
		1.238	**1.237**	1.238	*1.237*	1.270	1.325
		0.238	0.242	*0.238*	0.242	**0.238**	0.239
		0.896	**0.893**	0.896	**0.893**	0.926	0.959
	4	0.260	*0.253*	0.260	**0.253**	0.283	0.289
		1.044	1.068	**1.044**	1.068	1.123	1.178
		0.260	**0.249**	0.260	**0.249**	0.259	0.262
		1.020	**1.002**	1.020	**1.002**	1.017	1.109
	5	0.244	0.244	0.244	0.244	*0.242*	**0.192**
		0.879	0.962	**0.879**	0.962	0.993	0.909
		0.311	0.309	0.311	0.309	*0.306*	**0.258**
		0.857	0.880	**0.857**	0.880	0.913	0.867
gradient	1	0.236	*0.235*	0.236	0.235	0.237	**0.226**
		0.820	0.818	0.820	*0.818*	0.839	**0.725**
		0.227	0.229	*0.227*	0.229	0.232	**0.197**
		0.999	1.020	**0.999**	1.020	1.047	1.026
	2	*0.257*	0.257	0.257	0.257	0.257	**0.253**
		0.950	0.951	0.950	0.951	*0.947*	**0.940**
		0.179	0.179	0.179	0.179	0.179	**0.173**
		0.960	0.960	*0.960*	0.960	0.973	**0.955**
	3	**0.295**	0.295	**0.295**	0.295	0.296	0.301
		0.971	*0.968*	0.971	**0.968**	0.969	0.974
		0.263	**0.262**	0.263	*0.262*	0.264	0.265
		0.938	0.938	*0.938*	0.938	0.947	0.980
	4	0.272	*0.272*	0.272	**0.272**	0.275	0.275
		1.001	1.009	*1.001*	1.009	1.016	1.010
		0.248	0.249	0.248	0.249	0.250	**0.248**
		1.255	1.254	1.255	*1.254*	1.266	**1.231**
	5	0.270	**0.267**	0.270	**0.267**	0.296	0.271
		0.850	0.857	*0.850*	0.857	0.866	**0.845**
		0.219	0.221	0.219	0.221	*0.214*	**0.205**
		1.046	*1.040*	1.046	*1.040*	**1.011**	1.107
parabola	1	**0.300**	0.307	**0.300**	0.307	0.336	0.332
		0.951	**0.942**	0.951	*0.942*	1.109	1.165
		0.200	**0.189**	0.200	*0.189*	0.221	0.230
		1.078	*1.070*	1.078	**1.070**	1.182	1.196
	2	0.190	**0.189**	0.190	*0.189*	0.192	0.190
		0.737	0.739	*0.737*	0.739	0.764	0.751
		0.200	*0.198*	0.200	*0.198*	0.199	**0.190**
		1.045	1.057	*1.045*	1.057	1.079	**1.021**
	3	*0.263*	0.263	0.263	0.263	0.266	**0.252**
		0.797	*0.785*	0.797	*0.785*	0.786	**0.779**
		0.301	0.301	0.301	0.301	*0.299*	**0.294**
		1.098	*1.073*	1.098	**1.073**	1.091	1.099
	4	0.253	*0.252*	0.253	**0.252**	0.253	0.255
		0.871	*0.859*	0.871	*0.859*	0.869	**0.819**
		0.236	0.238	*0.236*	0.238	0.239	**0.228**
		0.977	*0.883*	0.977	**0.883**	0.985	0.997
	5	**0.213**	0.214	**0.213**	0.214	0.226	0.265
		0.728	0.739	0.728	0.739	**0.704**	0.883
		0.247	0.253	**0.247**	0.253	0.280	0.284
		0.958	*0.933*	0.958	**0.933**	1.050	1.250

Table 6: Mean squared error of $\hat{Y}$ (**minimum**, *next best*).