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

Wesley Brooks

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 local regression model such that the regression coefficients vary over space and will be referred to as a spatially varying coefficient regression model (SVCR). Statistical inference for the coefficients as functions of location in an SVCR model is more complicated than estimating the coefficients in a global linear regression model where the coefficients are constant across the spatial domain. This document concerns the development of new methodologies 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 univariate $\{Y(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ and possibly multivariate $\{\boldsymbol{X}(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D}\}$ denote random fields of the response and the covariates, respectively. For $i = 1, \dots, n$, let $\boldsymbol{s}_i$ denote the location in $\mathcal{D}$ of the ith observation of the response and the covariates. Then the data are a realization of the random variables $\{Y(\boldsymbol{s}_i), \boldsymbol{X}(\boldsymbol{s}_i)\}$ for $i = 1, \dots, n$. Let the observed data be denoted $\{y(\boldsymbol{s}_i), \boldsymbol{x}(\boldsymbol{s}_i)\}$, $i = 1, \dots, n$.

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 $i = i, \ldots, n$.

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 regression 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

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

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

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

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. A random field is second-order stationary if the joint distribution at any two locations in the domain does not change when the locations are shifted by a fixed spatial lag.

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

$$Y(\boldsymbol{s}) = \boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta}(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \tag{4}$$

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 varying coefficient 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. Two methods have been proposed for global model selection in spline-based VCR models. Wang et al. (2008) applied 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 can be used for local variable selection.

Geographically weighted regression (GWR) is a kernel-based method of 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. In the context of ordinary least squares regression, Lasso regularization for variable selection (Tibshirani, 1996), while popular, does not generally produce consistent estimates of the relevant predictor variables (Leng et al., 2006). Regularization methods such as the adaptive Lasso (AL) (Zou, 2006) were developed and shown to have appealing properties for automating variable selection, sometimes including the "oracle" property of asymptotically selecting exactly the correct set of covariates for inclusion in a regression model.

The idea of using Lasso regularization for local variable selection in a GWR model appeared in the literature as the geographically weighted Lasso (GWL) (Wheeler, 2009). The GWL uses the Lasso with a jackknife criterion for selection of the tuning parameters. Because the jackknife criterion can only be computed at sampling locations where the response variable is observed, the GWL cannot be used for imputation of missing data nor for interpolation between sampling locations.

This paper introduces a new regularization method for local variable selection in GWR models that overcomes this limitation of the GWL by using a penalized-likelihood criterion to select the Lasso tuning parameters. In particular, a type of BIC is developed and used herein, 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) and a total BIC is based on an *ad hoc* calculation of the sample size and degrees of freedom for estimating the spatially-varying coefficient surfaces.

Three regularization methods were used in this work. The AL was implemented in two ways - once via the least angle regression (LARS) algorithm (Efron et al., 2004) which uses least squares, and once via the coordinate descent algorithm using the `R` package `glmnet` (Friedman et al., 2010). The third regularization method implemented here uses an adaptive elastic net (AEN) penalty (Zou and Zhang, 2009), also via the coordinate descent algorithm using the `glmnet` package.

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,

$$y(\boldsymbol{s}_i) = \boldsymbol{x}(\boldsymbol{s}_i)'\boldsymbol{\beta}(\boldsymbol{s}_i) + \varepsilon(\boldsymbol{s}_i). \tag{5}$$

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,

$$\varepsilon(\boldsymbol{s}_i) \overset{iid}{\sim} \mathcal{N}\left(0, \sigma^2\right). \tag{6}$$

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 (5) and (6) can now be rewritten as

$$y_i = \boldsymbol{x}_i'\boldsymbol{\beta}_i + \varepsilon_i \text{ and } \varepsilon_i \overset{iid}{\sim} \mathcal{N}\left(0, \sigma^2\right). \tag{7}$$

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:

$$\ell(\boldsymbol{\beta}) = -(1/2)\left\{ n \log\left(2\pi\sigma^2\right) + \left(\sigma^2\right)^{-1} \sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i'\boldsymbol{\beta}_i\right)^2 \right\}. \tag{8}$$

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 $\delta_{ii'} \equiv \delta(\boldsymbol{s}_i, \boldsymbol{s}_{i'}) \equiv \|\boldsymbol{s}_i - \boldsymbol{s}_{i'}\|_2$ between locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_{i'}$ for all $i, i' = 1, \ldots, n$. The bi-square kernel can be used to generate spatial weights based on the Euclidean distances and a bandwidth ϕ:

$$w_{ii'} = \begin{cases} \left[1 - \left(\phi^{-1}\delta_{ii'}\right)^2\right]^2 & \text{if } \delta_{ii'} < \phi, \\ 0 & \text{if } \delta_{ii'} \geqslant \phi. \end{cases} \tag{9}$$

The bisquare kernel in (9) 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 location:

$$\mathcal{L}_i\left(\boldsymbol{\beta}_i\right) = \prod_{i'=1}^{n} \left[\left(2\pi\sigma_i^2\right)^{-1/2} \exp\left\{-\left(2\sigma_i^2\right)^{-1}\left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2\right\}\right]^{w_{ii'}}, \tag{10}$$

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:

$$\ell_i\left(\boldsymbol{\beta}_i\right) = -(1/2)\sum_{i'=1}^{n} w_{ii'}\left\{\log\sigma_i^2 + \left(\sigma_i^2\right)^{-1}\left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2\right\}. \tag{11}$$

The GWR coefficient estimates $\hat{\boldsymbol{\beta}}_{i,\mathrm{GWR}}$ maximize the local likelihood at location $\boldsymbol{s}_i$. From (10) and (11), it is apparent that $\hat{\boldsymbol{\beta}}_{i,\mathrm{GWR}}$ can be obtained using weighted least squares. Let $\boldsymbol{W}_i$ denote a diagonal weight matrix with diagonal entries $w_{ii'}$ for $i' = 1,\ldots,n$. That is,

$$\boldsymbol{W}_i = \mathrm{diag}\left\{w_{ii'}\right\}_{i'=1}^{n}. \tag{12}$$

It follows that, by weighted least squares,

$$\hat{\boldsymbol{\beta}}_{i,\mathrm{GWR}} = \left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{y}. \tag{13}$$

The estimate of σ_i^2 is obtained by maximizing (11), and is:

$$\begin{aligned}\hat{\sigma}_i^2 &= \left(\boldsymbol{1}_n'\boldsymbol{w}_i\right)^{-1}\left(\boldsymbol{y} - \boldsymbol{X}\left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{y}\right)'\boldsymbol{W}_i\left(\boldsymbol{y} - \boldsymbol{X}\left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{y}\right)\\ &= \left(\boldsymbol{1}_n'\boldsymbol{w}_i\right)^{-1}\left(\boldsymbol{y} - \hat{\boldsymbol{y}}\right)'\boldsymbol{W}_i\left(\boldsymbol{y} - \hat{\boldsymbol{y}}\right),\end{aligned} \tag{14}$$

where $\boldsymbol{1}_n$ is an n-variate vector of ones.

3. Model Selection

3.1. Local Variable Selection

Both the adaptive Lasso (AL) and the adaptive elastic net (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.

3.1.1. Local variable selection with the adaptive Lasso

The objective function for the local geographically weighted adaptive Lasso (GWAL) method at $\boldsymbol{s}_i$ is defined to be

$$\mathcal{S}(\boldsymbol{\beta}_i) = \sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}'_{i'} \boldsymbol{\beta}_i \right)^2 + \lambda_i \sum_{j=1}^{p} |\beta_{ij}| / \gamma_{ij}, \tag{15}$$

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 $\lambda_i \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij}$ 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 penalty 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 with the adaptive elastic net

The objective function for the local geographically weighted adaptive elastic net (GWAEN) method at $\boldsymbol{s}_i$ is defined to be

$$\begin{aligned}\mathcal{S}(\boldsymbol{\beta}_i) &= \sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}'_{i'}\boldsymbol{\beta}_i\right)^2 + \alpha_i\lambda_i \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij} + (1-\alpha_i)\lambda_i \sum_{j=1}^{p} \left(\beta_{ij}/\gamma_{ij}\right)^2 \\ &= \sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}'_{i'}\boldsymbol{\beta}_i\right)^2 + \lambda_i \left\{\alpha_i \sum_{j=1}^{p} |\beta_{ij}|/\gamma_{ij} + (1-\alpha_i) \sum_{j=1}^{p} \left(\beta_{ij}/\gamma_{ij}\right)^2\right\} \qquad (16)\end{aligned}$$

where the adaptive weights $\boldsymbol{\gamma}_i = (\gamma_{i1}, \ldots, \gamma_{ip})'$ are calculated as for the AL, and the elastic net parameter α_i controls the balance between the ℓ_1 and ℓ_2 penalties.

Fitting a SVCR model by the GWAEN 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 is required for the variable selection step of fitting each local model by the GWAL or GWAEN 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 (BIC_{loc}):

$$\begin{aligned}
\text{BIC}_{\text{loc},i} &= -2\sum_{i'=1}^{n} \ell_{ii'} + \left(\sum_{i'=1}^{n} w_{ii'}\right) \text{df}_i \\
&= -2 \times \sum_{i'=1}^{n} \log\left\{\left(2\pi\hat{\sigma}_i^2\right)^{-1/2} \exp\left[-\frac{1}{2}\hat{\sigma}_i^{-2}\left(y_{i'} - \boldsymbol{x}'_{i'}\hat{\boldsymbol{\beta}}_{i'}\right)^2\right]\right\}^{w_{ii'}} + \left(\sum_{i'=1}^{n} w_{ii'}\right) \text{df}_i \\
&= \sum_{i'=1}^{n} w_{ii'}\left\{\log\left(2\pi\right) + \log\hat{\sigma}_i^2 + \hat{\sigma}_i^{-2}\left(y_{i'} - \boldsymbol{x}'_{i'}\hat{\boldsymbol{\beta}}_{i'}\right)^2\right\} + \left(\sum_{i'=1}^{n} w_{ii'}\right) \text{df}_i \\
&= \hat{\sigma}_i^{-2}\sum_{i'=1}^{n} w_{ii'}\left(y_{i'} - \boldsymbol{x}'_{i'}\hat{\boldsymbol{\beta}}_{i}\right)^2 + \left(\sum_{i'=1}^{n} w_{ii'}\right) \text{df}_i + C_i \qquad (17)
\end{aligned}$$

where $C_i = \sum\limits_{i'=1}^{n} w_{ii'}\left\{\log 2\pi + \log\hat{\sigma}_i^2\right\}$.

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, C_i does not depend on the choice of tuning parameter and can be ignored (Zou et al., 2007).

Wheeler (2009) proposed selecting the tuning parameter for the Lasso at location $\boldsymbol{s}_i$ to minimize the jackknife prediction error $|y_i - \hat{y}_i^{(i)}|$. Because the jackknife prediction error is undefined everywhere except for at observation 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

After the variables are selected for inclusion in the local model, either by the GWAL or the GWAEN, the coefficient estimates are computed via weighted least squares on the selected variables without regularization. That is, letting $\boldsymbol{\Omega}_i = \text{diag}\left\{\boldsymbol{\omega}_i\right\}$ where $\boldsymbol{\omega}_i = (\omega_{i1}, \ldots \omega_{ip})'$ and $\omega_{ij} = I\left(\beta_{ij} \neq 0\right)$, the local coefficient estimates are:

$$\hat{\boldsymbol{\beta}}_i = \underset{\beta}{\text{argmin}} \sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}'_{i'} \boldsymbol{\Omega}_i \boldsymbol{\beta}'_i\right)^2 \tag{18}$$

$$= \left(\tilde{\boldsymbol{X}}' \boldsymbol{W}_i \tilde{\boldsymbol{X}}\right)^{-1} \tilde{\boldsymbol{X}}' \boldsymbol{W}_i \boldsymbol{y}. \tag{19}$$

and $\tilde{\boldsymbol{X}}_i$ is the design matrix $\boldsymbol{X}$ with columns corresponding to zeroes in $\boldsymbol{\omega}_i$ removed.

3.4. Bandwidth selection

Let H_i denote the ith row of the matrix $\boldsymbol{W}_i^{1/2} \tilde{\boldsymbol{X}} \left(\tilde{\boldsymbol{X}}' \boldsymbol{W}_i \tilde{\boldsymbol{X}}\right)^{-1} \tilde{\boldsymbol{X}}' \boldsymbol{W}_i^{1/2}$, and let

$$\boldsymbol{H} = \left(H_1 \cdots H_n\right)'. \tag{20}$$

The fitted values from the model are

$$\hat{\boldsymbol{y}} = \boldsymbol{H} \boldsymbol{y} \tag{21}$$

The global bandwidth parameter ϕ in (9) is selected by minimizing an approximation to the global AIC:

$$\text{AIC} = 2n \log \sigma + n \left\{ \frac{n + \nu}{n - 2 - \nu} \right\} \tag{22}$$

where ν is the trace of the 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 spatial 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 spatial covariance $\text{Cov}\left(X_{ji}, X_{ji'}\right) = \sigma_x^2 \exp\left(-\tau_x^{-1}\delta_{ii'}\right)$ where $\sigma_x^2 = 1$ is the variance, $\tau_x = 0$ 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 $\boldsymbol{X}$ matrix 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 for simplicity 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, so $x_2, \ldots, x_5$ are included to assess performance in variable-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 spatial domain $[0, 1]^2$. There were four other simulated covariates, but their true coefficient surfaces were zero across the area under simulation. The twelve simulation settings are described in Table 1. 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 ($\sigma_\varepsilon^2 = 0.25$) and high ($\sigma_\varepsilon^2 = 1$) variance for the random error term.

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 0.535 at the center of the domain and falling to zero at each corner of the domain. The parabola is computed by finding the squared distance of each sampling location from the domain's center, multiplying by -1 and then adding an offset so that the corner points are equal to zero.

The performance of the penalized GWR methods (AL via `lars` and via `glmnet`, and the AEN via `enet`) was compared to that of oracular GWR (O-GWR), which is ordinary GWR with "oracular" variable selection, meaning that exactly the correct set of covariates was used to fit the GWR model at each location in the simulation. Also included in the comparison was the GWR algorithm of Fotheringham et al. (2002) without variable selection (`gwr`). 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.

Results from the simulation were summarized at five locations on the simulated grid (see Figure 2). The five key locations were chosen because they represent interesting regions of the β_1 coefficient surfaces. The results of variable selection and coefficient estimation are presented in the tables below.

4.2. Simulation results

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. Arguably the least-accurate selection was at locations one and five for the step function using the `lars` algorithm, where variables that do not appear in the true model were selected for inclusion at rates between 11% and 22%. The `enet` and `glmnet` algorithms, using the same data, had false-positive errors at rates between 0% and 8%, which are typical of the error rates for all other location/function/algorithm combinations.

Selection performance was more affected by an increase in the noise variance from $\sigma_\varepsilon = 0.5$ to $\sigma_\varepsilon = 1$ than by an increase in colinearity from $\rho = 0$ to $\rho = 0.5$. For instance, for the step function at location three, $\beta_1(\boldsymbol{s}_3) = 0.5$. Where $\sigma_\varepsilon = 0.5$, the `glmnet` algorithm selected $\beta_1(\boldsymbol{s}_3)$ for inclusion at a rate of 100% (when $\rho = 0$) and 99% (when $\rho = 0.5$). But when $\sigma_\varepsilon = 1$, the same algorithm selected $\beta_1(\boldsymbol{s}_3)$ for inclusion at a rate of 75% (when $\rho = 0$) and 68% (when $\rho = 0.5$).

The `enet` algorithm outperforms the others in selection but the difference is small - a roughly one percentage point improvement in the rate of true positives and true negatives when $\rho = 0.5$.There is no apparent difference between `glmnet` and `enet` when $\rho = 0$.

Coefficient Estimation. The MSE, bias, and variance of $\hat{\beta}_1$ are listed in Tables 3, 4, and **??**, respectively. The method of oracular selection led to the best MSE in 28 of the 60 cases, which is more than any other single method. In general, the methods that do local variable selection had lower MSE than traditional GWR. As was the case for selection, estimation accuracy (in terms of MSE) suffered more by an increase in σ_ε from 0.5 to 1 than from an increase in ρ from 0 to 0.5. Oracular selection was decisively superior to traditional GWR and to local variable selection for estimating the gradient β_1, turning in the best MSE for all combinations of location and simulation parameters.

In general, oracular selection and traditional GWR were quite similar in terms of var $\left(\hat{\beta}_1\right)$, with notably greater variance for the local selection methods. However, the local selection methods had less bias than traditional GWR, even exhibiting less bias than oracular selection in many settings. There was no simulation setting for which traditional GWR had the smallest or second-smallest bias.

It seems, therefore, that the local selection methods reduce bias and increase variance of the coefficient estimates, as compared to traditional GWR. Whether traditional GWR or local selection is better in terms of MSE of the coefficient estimates is not clear in all cases, but when the actual coefficient is equal to zero (or nearly so), local selection does seem to reduce the MSE over traditional GWR.

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^2_ε, which is 1 for odd-numbered rows and 0.25 for even numbered rows. There is not much difference in MSE $\left(\hat{Y}\right)$ between the various estimation methods, except that it is larger for the oracular and `gwr` methods where $\beta_1(\boldsymbol{s})$ is near or equal to zero.

4.3. Tables

4.3.1. Selection

4.3.2. Estimation

5. 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*(3), 638–661.

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

Brundson, C., S. Fotheringham, and M. Charltonn (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.* Wiley.

Diggle, P. and P. Ribeiro (2007). *Model-based geostatistics.* Springer New York.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least angle regression. *Annals of Statistics 32*(2), 407–499.

Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association 96*(456), 1348–1360.

Fan, J. and W. Zhang (1999). Statistical estimation in varying coefficient models. *The Annals of Statistics 27*(5), 1491–1518.

Fotheringham, A., C. Brunsdon, and M. Charlton (2002). *Geographically weighted regression: the analysis of spatially varying relationships.* Wiley.

Friedman, J., T. Hastie, and R. Tibshirani (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software 33*(1), 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*(462), 387–396.

Hastie, T. and C. Loader (1993). Local regression: automatic kernel carpentry. *Statistical Science 8*(2), 120–143.

Hastie, T. and R. Tibshirani (1993). Varying-coefficient models. *Journal of the Royal Statistical Society. Series B (Methodological) 55*(4), pp. 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*(1), 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 (Statistical Methodology) 60*(2), pp. 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. *The Annals of Statistics 6*(2), 461–464.

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

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological) 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*(484), 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.* Texts in statistical science. Chapman & Hall/CRC.

Zhang, J. and M. Clayton (2011). Functional concurrent linear regression model for images. *Journal of Agricultural, Biological, and Environmental Statistics 16*(1), 105–130.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association 101*(476), 1418–1429.

Zou, H., T. Hastie, and R. Tibshirani (2007). On the "degrees of freedom" of the lasso. *Annals of Statistics 35*(5), 2173–2192.

Zou, H. and H. Zhang (2009). On the adaptive elastic net with a diverging number of parameters. *The Annals of Statistics 37*(4), 1733–1751.

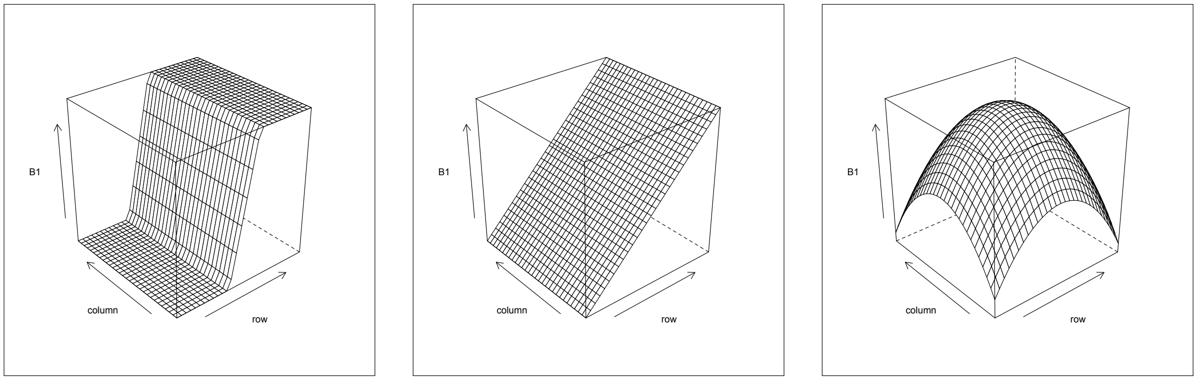

Figure 1: The actual β_1 coefficient surface used in the simulation.

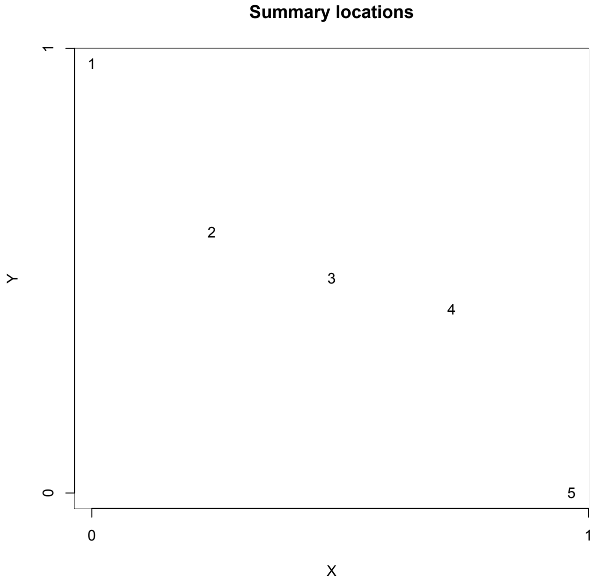

Figure 2: Locations where the variable selection and coefficient estimation of GWL were summarized.

Setting	function	ρ
1	step	0
4	step	0.5
5	gradient	0
8	gradient	0.5
10	parabola	0
11	parabola	0.5

Table 1: Simulation parameters for each setting.

	step				gradient				parabola			
	`enet`		`glmnet`		`enet`		`glmnet`		`enet`		`glmnet`	
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	1.00	0.04	1.00	0.05	0.99	0.10	0.96	0.08	1.00	0.03	1.00	0.02
	0.86	0.08	0.82	0.07	0.84	0.07	0.88	0.05	0.97	0.06	0.97	0.07
2	1.00	0.07	1.00	0.06	1.00	0.06	1.00	0.05	1.00	0.08	1.00	0.07
	1.00	0.06	1.00	0.06	1.00	0.07	0.99	0.04	0.98	0.08	0.99	0.07
3	0.99	0.06	0.99	0.06	0.97	0.08	0.92	0.04	1.00	0.08	1.00	0.07
	0.84	0.08	0.82	0.07	0.81	0.11	0.80	0.08	0.95	0.08	0.96	0.08
4	0.64	0.06	0.59	0.06	0.51	0.12	0.40	0.07	1.00	0.06	1.00	0.06
	0.48	0.07	0.49	0.07	0.52	0.07	0.51	0.07	0.95	0.07	0.93	0.06
5	0.03	0.03	0.03	0.03	0.02	0.03	0.03	0.05	0.93	0.05	0.94	0.04
	0.06	0.04	0.04	0.05	0.04	0.03	0.06	0.06	0.70	0.07	0.70	0.07

Table 2: Selection frequency for the simulation experiment

function	location	enet	glmnet	u.enet	u.glmnet	oracular	gwr
step	1	0.025	*0.023*	0.127	0.124	0.082	**0.005**
		0.186	0.216	0.376	0.375	*0.134*	**0.009**
	2	0.024	0.024	*0.021*	**0.021**	0.021	0.042
		0.063	0.068	*0.054*	0.056	**0.042**	0.070
	3	0.011	0.010	0.007	0.007	**0.004**	*0.005*
		0.043	0.047	0.049	0.054	*0.009*	**0.008**
	4	*0.014*	**0.014**	0.019	0.018	0.021	0.042
		0.036	*0.039*	0.042	0.046	0.047	0.074
	5	*0.001*	0.002	0.004	0.004	**0.000**	0.007
		0.006	*0.002*	0.024	0.009	**0.000**	0.011
gradient	1	*0.045*	0.073	0.134	0.205	0.101	**0.011**
		0.218	0.179	0.425	0.369	*0.154*	**0.022**
	2	0.027	0.021	0.021	**0.017**	*0.018*	0.044
		0.071	0.071	*0.056*	0.061	**0.043**	0.075
	3	0.014	0.022	0.011	0.021	**0.005**	*0.005*
		0.047	0.045	0.045	0.044	*0.008*	**0.008**
	4	*0.012*	**0.011**	0.016	0.014	0.020	0.044
		0.028	*0.038*	0.047	0.048	0.043	0.082
	5	*0.002*	0.003	0.009	0.009	**0.000**	0.010
		0.004	0.022	0.038	0.043	**0.000**	0.015
parabola	1	0.069	0.070	**0.007**	*0.007*	0.010	0.016
		0.094	0.096	0.078	0.085	*0.045*	**0.042**
	2	0.003	0.003	*0.001*	0.001	**0.001**	0.001
		0.013	0.008	0.013	0.009	*0.002*	**0.002**
	3	0.001	0.001	0.001	*0.001*	**0.001**	0.001
		0.017	0.015	0.019	0.017	*0.002*	**0.002**
	4	0.003	0.003	0.001	*0.001*	**0.001**	0.001
		0.014	0.016	0.012	0.015	**0.002**	*0.003*
	5	0.068	0.069	0.004	*0.004*	**0.000**	0.016
		0.051	0.052	0.019	*0.019*	**0.000**	0.044

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

function	location	enet	glmnet	u.enet	u.glmnet	oracular	gwr
step	1	-0.029	*-0.020*	0.038	0.033	0.034	**-0.004**
		-0.195	-0.211	-0.082	-0.075	*0.053*	**-0.017**
	2	-0.119	-0.119	*-0.110*	**-0.110**	-0.124	-0.196
		-0.178	-0.186	**-0.145**	*-0.150*	-0.175	-0.253
	3	*-0.014*	**-0.010**	0.017	0.015	0.021	0.040
		-0.027	-0.031	*0.009*	**0.004**	0.050	0.059
	4	*0.059*	**0.049**	0.074	0.065	0.129	0.196
		0.075	*0.076*	0.088	0.090	0.193	0.263
	5	*-0.006*	-0.006	-0.009	-0.010	**0.000**	-0.006
		-0.009	*-0.000*	-0.025	-0.008	**0.000**	-0.011
gradient	1	-0.077	-0.073	0.028	**-0.014**	0.050	*-0.017*
		-0.214	-0.167	-0.067	-0.068	*0.035*	**0.006**
	2	-0.130	*-0.099*	-0.103	**-0.083**	-0.110	-0.199
		-0.221	-0.216	**-0.167**	-0.184	*-0.182*	-0.263
	3	-0.056	-0.056	**-0.009**	-0.030	*0.017*	0.034
		-0.094	-0.077	-0.056	-0.056	**0.017**	*0.055*
	4	0.027	**0.010**	0.043	*0.020*	0.129	0.199
		0.073	*0.089*	0.105	0.105	0.189	0.275
	5	*-0.005*	-0.009	-0.009	-0.012	**0.000**	-0.009
		-0.011	-0.011	-0.036	-0.021	**0.000**	*-0.007*
parabola	1	-0.248	-0.253	**0.010**	0.011	*0.011*	-0.111
		-0.242	-0.248	*-0.014*	-0.022	**-0.007**	-0.182
	2	-0.047	-0.048	*0.002*	**0.001**	0.004	0.002
		-0.044	-0.035	**0.003**	0.011	*0.008*	-0.011
	3	0.005	0.005	*0.002*	**0.001**	0.003	0.002
		-0.017	-0.012	-0.013	-0.007	**0.003**	*0.006*
	4	0.043	0.045	*0.006*	0.007	**0.004**	0.008
		0.006	**0.002**	-0.014	-0.023	*0.004*	0.020
	5	0.249	0.253	*0.002*	0.003	**0.000**	0.113
		0.182	0.186	*-0.001*	0.004	**0.000**	0.187

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

function	location	enet	glmnet	u.enet	u.glmnet	oracular	gwr
step	1	0.024	*0.023*	0.127	0.124	0.081	**0.005**
		0.149	0.173	0.373	0.373	*0.133*	**0.009**
	2	0.010	0.010	0.009	0.009	*0.006*	**0.003**
		0.032	0.034	0.033	0.034	*0.012*	**0.006**
	3	0.011	0.010	0.007	0.007	*0.004*	**0.003**
		0.043	0.047	0.050	0.055	*0.007*	**0.004**
	4	0.011	0.012	0.014	0.014	*0.004*	**0.003**
		0.030	0.033	0.035	0.038	*0.009*	**0.005**
	5	*0.001*	0.002	0.004	0.004	**0.000**	0.007
		0.006	*0.002*	0.024	0.009	**0.000**	0.011
gradient	1	*0.040*	0.068	0.134	0.207	0.099	**0.011**
		0.174	*0.153*	0.424	0.368	0.154	**0.022**
	2	0.011	0.012	0.010	0.010	*0.006*	**0.005**
		0.022	0.025	0.028	0.028	*0.010*	**0.006**
	3	0.011	0.019	0.011	0.020	*0.004*	**0.004**
		0.039	0.039	0.043	0.042	*0.008*	**0.005**
	4	0.011	0.011	0.014	0.013	**0.003**	*0.004*
		0.023	0.031	0.037	0.037	*0.007*	**0.006**
	5	*0.002*	0.003	0.009	0.009	**0.000**	0.010
		0.004	0.022	0.037	0.043	**0.000**	0.015
parabola	1	0.007	*0.006*	0.007	0.007	0.010	**0.004**
		0.035	0.035	0.079	0.085	0.046	**0.009**
	2	0.001	*0.001*	0.001	0.001	**0.001**	0.001
		0.011	0.007	0.013	0.009	*0.002*	**0.002**
	3	0.001	0.001	0.001	*0.001*	**0.001**	0.001
		0.017	0.015	0.019	0.017	*0.002*	**0.002**
	4	0.001	0.001	0.001	*0.001*	**0.001**	0.001
		0.014	0.017	0.012	0.014	**0.002**	*0.002*
	5	0.006	0.005	0.004	0.004	**0.000**	*0.003*
		0.018	0.018	0.020	0.020	**0.000**	*0.009*

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

function	location	enet	glmnet	u.enet	u.glmnet	oracular	gwr
step	1	**0.100**	0.101	*0.100*	0.101	0.111	0.118
		0.594	**0.564**	0.594	*0.564*	0.694	0.850
	2	0.196	**0.194**	0.196	*0.194*	0.225	0.244
		1.019	**1.001**	1.019	**1.001**	1.171	1.123
	3	*0.232*	0.233	**0.232**	0.233	0.255	0.262
		0.850	**0.833**	0.850	**0.833**	1.025	1.020
	4	**0.241**	0.250	*0.241*	0.250	0.269	0.288
		0.950	**0.950**	0.950	**0.950**	1.045	1.053
	5	0.231	**0.224**	0.231	*0.224*	0.293	0.234
		0.675	0.697	**0.675**	0.697	0.782	0.716
gradient	1	*0.151*	0.169	**0.151**	0.169	0.213	0.247
		0.559	**0.552**	0.559	*0.552*	0.757	0.895
	2	0.275	**0.273**	0.275	*0.273*	0.311	0.332
		0.897	0.953	**0.897**	0.953	1.000	1.048
	3	0.257	**0.246**	0.257	*0.246*	0.275	0.265
		0.620	0.652	*0.620*	0.652	0.673	0.664
	4	0.293	**0.259**	0.293	**0.259**	0.304	0.333
		0.748	*0.743*	0.748	**0.743**	0.815	0.802
	5	0.259	**0.203**	0.259	**0.203**	0.278	0.238
		0.961	*0.915*	0.961	**0.915**	1.127	0.972
parabola	1	0.224	0.232	0.224	0.232	*0.223*	**0.222**
		0.669	0.671	*0.669*	0.671	0.723	0.757
	2	0.216	0.218	*0.216*	0.218	0.221	**0.210**
		0.814	0.836	*0.814*	0.836	0.863	0.832
	3	0.241	*0.241*	0.241	0.241	0.249	**0.229**
		1.094	1.096	**1.094**	1.096	1.135	1.117
	4	0.276	0.277	*0.276*	0.277	0.281	**0.262**
		0.882	*0.875*	0.882	0.875	0.885	**0.870**
	5	**0.197**	0.202	**0.197**	0.202	0.222	0.202
		1.257	*1.256*	1.257	**1.256**	1.289	1.275

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