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

Wesley Brooks

1. Introduction

Varying coefficient regression (VCR) is a technique used to model non-stationary regression processes (Hastie and Tibshirani, 1993). Whereas the coefficients in traditional linear regression are scalar constants, the coefficients in a VCR model are functions - often *smooth* functions - of some effect modifying variable. When the effect modifying variable $\boldsymbol{s}$ represents location in a spatial domain, a VCR model implies that there is a local regression model ($y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta}(\boldsymbol{s}) + \varepsilon(\boldsymbol{s})$) at each location s. These local models are in contrast to a global linear regression model, where the coefficients are constant across the domain. Estimating the coefficient functions of a VCR model is therefore more complicated than estimating the coefficients in a global linear regression model.

Spatial association - meaning that nearby locations are more alike than distant locations - is a key concept in spatial statistics. Common practice in the analysis of geostatistical data is to write a spatial model as the sum of systematic and random components, as in:

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

where $\boldsymbol{x}(\boldsymbol{s})$ is the (possibly multivariate) spatial covariate process, $\boldsymbol{\beta}$ is a vector of regression coefficients, $\varepsilon(\boldsymbol{s})$ is a white noise process, and $W(\boldsymbol{s})$ is a second-order stationary mean-zero process that is independent of the white noise process, called the spatial random effect process (Cressie, 1993). Fitting such a model then proceeds by specifying a covariance function for the spatial random effect (Diggle and Ribeiro, 2007). For example, the exponential covariance function has the form

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

where ϕ is a bandwidth parameter and $\delta(\boldsymbol{s}, \boldsymbol{t})$ is the Euclidean distance between locations $\boldsymbol{s}$ and $\boldsymbol{t}$.

The spatial random effect describes how deviations from the systematic part of the model (i.e. $\boldsymbol{x}'(s)\boldsymbol{\beta}$) are spatially clustered. Though the model may be unbiased globally, clustered deviations may indicate that the global model achieves this global unbiasedness by allowing local biases to offset each other. In that case, a VCR model might eliminate the need for a spatial random effect by being unbiased both globally and locally.

There are reasons other than bias to make spatial models local. Analysts may be interested, e.g., in local coefficient estimates, local variable selection, local mean-squared error (MSE), and local variance, to name a few.

Both spline-based (Wood, 2006) and kernel-based (Fan and Zhang, 1999; Loader, 1999) methods are available for estimating varying coefficient functions. Wood (2006) demonstrates that it is straightforward to modify a thin-plate regression spline model into a VCR model; Wang et al. (2008) introduced the SCAD penalty of Fan and Li (2001) for global variable selection in spline-based VCR models with a univariate effect-modifying parameter, and Antoniadas et al. (2012) use the nonnegative garrote of Breiman (1995) for global variable selection in P-spline-based VCR models having a univariate effect-modifying parameter.

This document focuses on GWR, which is a kernel-based method of estimating the coefficients of a VCR model. GWR uses kernel-weighted regression with weights based on the distance between observation locations. The presentation of GWR in Fotheringham et al. (2002) follows the development of local likelihood in Loader (1999). GWR can be thought of as a kernel smoother for regression coefficients, and hence GWR coefficient estimates are likely to exhibit bias near the boundary of the region being modeled (Hastie and Loader, 1993). Modeling the coefficient surface as locally linear rather than locally constant (by including coefficient-by-location interactions) can reduce this boundary-effect bias (Hastie and Loader, 1993). Adding these interactions to the GWR model is analogous to a transition from kernel smoothing to local regression, and was introduced in Wang et al. (2008).

One reason to prefer GWR to spline-based VCR models for spatial data is the ability to do local variable selection. This paper describes local variable selection in GWR models using the adaptive lasso of Zou (2006). The idea first appears in the literature as the geographically-weighted lasso (GWL) of Wheeler (2009), which uses a jackknife criterion for selection of the lasso tuning parameters. Because the jackknife criterion can only be computed at locations where the response variable is observed, the GWL cannot be used for imputation of missing data nor for interpolation between observation locations. We avoid this limitation of the GWL by using a penalized-likelihood criterion to select the lasso tuning parameters (specifically the AIC, but in principle one could use the BIC, *et cetera*). The AIC allows us to easily adapt our method to the setting of a generalized linear model. The local AIC presented here is based on the local likelihood (Loader, 1999) and the total AIC is based on an *ad hoc* calculation of the sample size and degrees of freedom for estimating the spatially-varying coefficient surfaces.

2. Geographically Weighted Regression

2.1. Model

Geographically-weighted regression (GWR) (Brundson et al., 1998; Fotheringham et al., 2002) is a kernel-based technique for estimating VCR coefficient functions in the context of spatial data.

Consider n data observations, made 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 noise at location $\boldsymbol{s}_i$. That is,

$$y(\boldsymbol{s}_i) = \boldsymbol{x}(\boldsymbol{s}_i)'\boldsymbol{\beta}(\boldsymbol{s}_i) + \varepsilon(\boldsymbol{s}_i) \tag{1}$$

Further assume that the error term $\varepsilon(\boldsymbol{s}_i)$ is normally distributed with zero mean and a possibly spatially-varying variance $\sigma^2(\boldsymbol{s}_i)$

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

In order to simplify the notation, let $\boldsymbol{x}(\boldsymbol{s}_i) \equiv \boldsymbol{x}_i \equiv (1, x_{i1}, \ldots, x_{ip})'$, $\boldsymbol{\beta}(\boldsymbol{s}_i) \equiv \boldsymbol{\beta}_i \equiv (\beta_{i0}, \beta_{i1}, \ldots, \beta_{ip})'$, $y(\boldsymbol{s}_i) \equiv y_i$, and $\sigma^2(\boldsymbol{s}_i) \equiv \sigma_i^2$. Further, let $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)'$ and $\boldsymbol{y} = (y_1, \ldots, y_n)'$. Equations (1) and (2) can now be rewritten as

$$y_i = \boldsymbol{x}_i' \boldsymbol{\beta}_i + \varepsilon_i \text{ and } \varepsilon_i \sim \mathcal{N}\left(0, \sigma_i^2\right) \tag{3}$$

Assume that, given the design matrix $\boldsymbol{X}$, observations of the response variable at different locations are statistically independent of each other. Then the total log-likelihood of the observed data is the sum of the log-likelihood of each individual observation.

$$\ell\left(\boldsymbol{\beta}\right) = -\left(1/2\right) \sum_{i=1}^{n} \left\{ \log\left(2\pi\sigma_i^2\right) + \left(\sigma_i^2\right)^{-1} \left(y_i - \boldsymbol{x}_i'\boldsymbol{\beta}_i\right)^2 \right\} \tag{4}$$

Since there are a total of $n \times (p+1)$ free parameters for n observations the model is not identifiable, so 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 the local likelihood. In the setting of spatial data and with the kernel smoother based on the physical distance between observation locations, this is the traditional GWR.

2.2. Estimation

In geographically weighted regression, 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'. The bi-square kernel can be used to generate spatial weights based on the Euclidean distances and a bandwidth ϕ:

$$w_{ii'} = \begin{cases} \left[1 - \left(\phi^{-1}\delta_{ii'}\right)^2\right]^2 & \text{if } \delta_{ii'} < \phi \\ 0 & \text{if } \delta_{ii'} \geqslant \phi \end{cases} \tag{5}$$

For the purpose of estimation, define the local likelihood at each location (Fotheringham et al., 2002):

$$\mathcal{L}_i\left(\boldsymbol{\beta}_i\right) \quad = \quad \prod_{i'=1}^{n} \left[\left(2\pi\sigma_i^2\right)^{-1/2} \exp\left\{-\left(2\sigma_i^2\right)^{-1}\left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2\right\}\right]^{w_{ii'}} \tag{6}$$

Thus, the local log-likelihood function is:

$$\ell_i\left(\boldsymbol{\beta}_i\right) \quad \propto \quad -(1/2)\sum_{i'=1}^{n} w_{ii'}\left\{\log\sigma_i^2 + \left(\sigma_i^2\right)^{-1}\left(y_{i'} - \boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2\right\} \tag{7}$$

From which it is apparent that the GWR coefficient estimates $\hat{\boldsymbol{\beta}}_{i,\mathrm{GWR}}$, which maximize the local likelihood at location $\boldsymbol{s}_i$, can be calculated using weighted least squares. Letting the diagonal weight matrix $\boldsymbol{W}_i$ be:

$$\boldsymbol{W}_i = \operatorname{diag}\left\{w_{ii'}\right\}_{i'=1}^{n} \tag{8}$$

We have:

$$\hat{\boldsymbol{\beta}}_{i,\mathrm{GWR}} = \left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{Y} \tag{9}$$

And $\hat{\sigma}_i$, which maximizes (7), is:

$$\hat{\sigma}_i = \left(\mathbf{1}_n'\boldsymbol{w}_i\right)^{-1}\boldsymbol{w}_i'\left(\boldsymbol{Y} - \boldsymbol{X}\left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{Y}\right) \tag{10}$$

3. Model selection

3.1. Variable selection

Traditional GWR relies on *a priori* model selection to decide which variables should be included in the model. In the context of ordinary least squares regression, regularization methods such as the adaptive lasso (Zou, 2006) have been shown to have appealing properties for automating variable selection, sometimes including the "oracle" property of asymptotically selecting exactly the correct variables for inclusion in a regression model.

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

3.1.1. Adaptive lasso

The adaptive lasso is applied to GWR by first multiplying the design matrix $\boldsymbol{X}$ by $\boldsymbol{W}_i^{1/2}$, the diagonal matrix of geographic weights centered at s_i. Since some of the weights $w_{ii'}$ may be zero, the matrix $\boldsymbol{W}_i^{1/2}\boldsymbol{X}$ is not of full rank. The matrices $\boldsymbol{Y}_i^*$, $\boldsymbol{X}_i^*$, and $\boldsymbol{W}_i^*$ are formed by dropping the rows of $\boldsymbol{X}$ and $\boldsymbol{W}_i$ that correspond to observations with zero weight in the regression model at location $\boldsymbol{s}_i$. Now, letting $\boldsymbol{U}_i^* = \boldsymbol{W}_i^{*1/2}\boldsymbol{X}_i^*$ and $\boldsymbol{V}_i^* = \boldsymbol{W}_i^{*1/2}\boldsymbol{Y}_i^*$, we seek the coefficients $\boldsymbol{\beta}_i$ of the regression model:

$$\boldsymbol{V}_i^* = \boldsymbol{U}_i^* \boldsymbol{\beta}_i + \boldsymbol{\varepsilon} \tag{11}$$

To apply the adaptive lasso for estimating these regression coefficients, each column of $\boldsymbol{U}_i^*$ is centered around zero and rescaled to have an L_2-norm of one. Let $\tilde{\boldsymbol{U}}_i^*$ be the centered-and-scaled version of $\boldsymbol{U}_i^*$. Adaptive weights are calculated using the OLS regression coefficients $\boldsymbol{\gamma}_i^*$ via ordinary least squares (OLS):

$$\boldsymbol{\gamma}_i^* = \left(\tilde{\boldsymbol{U}}_i^{*\prime} \tilde{\boldsymbol{U}}_i^* \right)^{-1} \tilde{\boldsymbol{U}}_i^{*\prime} \boldsymbol{V}_i^* \tag{12}$$

Now a final scaling step is done: for $j = 1, \ldots, p$, the jth column of $\tilde{\boldsymbol{U}}_i^*$ is multiplied by $(\gamma_i^*)_j$, the corresponding coefficient from (12). Call this rescaled matrix $\breve{\boldsymbol{U}}_i^*$.

Finally, the adaptive lasso coefficient estimates at location $\boldsymbol{s}_i$ are found, either by using the `lars` algorithm (Efron et al., 2004) to model $\boldsymbol{V}_i^*$ as a function of $\breve{\boldsymbol{U}}_i^*$ or by using the `glmnet` package to implement coordinate descent. Either way, the objective being minimized is the same:

$$\sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}_{i'}' \boldsymbol{\beta}_i \right)^2 \quad + \quad \lambda_i \sum_{j=1}^{p} |\beta_{ij} / \gamma_{ij}^*| \tag{13}$$

(14)

3.1.2. Adaptive elastic net

To implement the adaptive elastic net (Zou and Zhang, 2009), the adaptive weights $\boldsymbol{\gamma}_i^*$ are calculated as for the adaptive lasso, but there is an additional elastic net parameter α that controls the balance between the ℓ_1 and ℓ_2 penalties, so that the objective to be minimized is:

$$
\begin{aligned}
&\sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}'_{i'}\boldsymbol{\beta}_i\right)^2 + \alpha\lambda_i \sum_{j=1}^{p} |\beta_{ij}/\gamma_{ij}^*| + (1-\alpha)\lambda_i \sum_{j=1}^{p} \left(\beta_{ij}/\gamma_{ij}^*\right)^2 && (15)\\
= &\sum_{i'=1}^{n} w_{ii'} \left(y_{i'} - \boldsymbol{x}'_{i'}\boldsymbol{\beta}_i\right)^2 + \lambda_i \left(\alpha \sum_{j=1}^{p} |\beta_{ij}/\gamma_{ij}^*| + (1-\alpha) \sum_{j=1}^{p} \left[\beta_{ij}/\gamma_{ij}^*\right]^2\right) && (16)
\end{aligned}
$$

In the simulation study (Section 4), α is calculated from the maximum global (i.e. for all data without weighting) Pearson correlation between any two covariates, $\rho_{\max}$: $\alpha = 1 - \rho_{\max}$.

3.2. Tuning parameter selection

At each location $\boldsymbol{s}_i$, it is necessary to select the lasso tuning parameter λ_i. To compare different values of λ_i, we propose a locally-weighted version of the Akaike information criterion (AIC) (Akaike, 1974) which we call the local AIC, or AIC_{loc}. The local AIC is calculated by adding a penalty to the local likelihood, with the sum of the weights around 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).

$$\begin{aligned}
\mathrm{AIC}_{\mathrm{loc},i} &= -2\sum_{i'=1}^{n}\ell_{ii'} + 2\mathrm{df}_i & (17)\\
&= -2\times\sum_{i'=1}^{n}\log\left\{\left(2\pi\hat{\sigma}_i^2\right)^{-1/2}\exp\left[-\frac{1}{2}\hat{\sigma}_i^{-2}\left(y_{i'}-\boldsymbol{x}_{i'}'\hat{\boldsymbol{\beta}}_{i'}\right)^2\right]\right\}^{w_{ii'}} + 2\mathrm{df}_i & (18)\\
&= \sum_{i'=1}^{n} w_{ii'}\left\{\log(2\pi) + \log\hat{\sigma}_i^2 + \hat{\sigma}_i^{-2}\left(y_{i'}-\boldsymbol{x}_{i'}'\hat{\boldsymbol{\beta}}_{i'}\right)^2\right\} + 2\mathrm{df}_i & (19)\\
&= \hat{\sigma}_i^{-2}\sum_{i'=1}^{n} w_{ii'}\left(y_{i'}-\boldsymbol{x}_{i'}'\hat{\boldsymbol{\beta}}_i\right)^2 + 2\mathrm{df}_i + C_i & (20)
\end{aligned}$$

Where the estimated local variance $\hat{\sigma}_i^2$ is the variance estimate from the unpenalized local model (Zou et al., 2007), so C_i does not depend on the choice of tuning parameter and can be ignored.

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 AIC can be calculated at any location where we can calculate the local likelihood. As a practical matter this allows for variable selection and coefficient surface estimation to be done at locations where no data was observed (interpolation) and for imputation of missing values of the response variable.

3.3. Bandwidth selection

The bandwidth parameter is global and so we need a global statistic for comparing prospective bandwidths. The objective minimized by GWL is:

$$\sum_{i'=1}^{n} w_{ii'}\left(y_{i'}-\boldsymbol{x}_{i'}'\boldsymbol{\beta}_i\right)^2 + \sum_{j=1}^{p}\lambda_{ij}|\beta_{ij}| \tag{21}$$

Where $\lambda_{ij}, j = 1, \ldots, p$ are penalties from the adaptive lasso (Zou, 2006). Taking the derivatives with respect to β and setting to zero, we see that

$$\hat{\boldsymbol{\beta}}_{i,\mathrm{GWL}} = \left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{Y} - \frac{1}{2}\left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{\lambda}_i \tag{22}$$

$$\hat{y}_i = \boldsymbol{x}_i'\hat{\boldsymbol{\beta}}_{i,\mathrm{GWL}} = \boldsymbol{x}_i'\left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{Y} - \frac{1}{2}\boldsymbol{x}_i'\left(\boldsymbol{X}'\boldsymbol{W}_i\boldsymbol{X}\right)^{-1}\boldsymbol{\lambda}_i \tag{23}$$

Unlike in the case of ordinary geographically-weighted regression, the fitted values $\hat{\boldsymbol{Y}}$ are not a linear combination of the observations $\boldsymbol{Y}$. Because GWL is not a linear smoother it is not possible to calculate the AIC as in Fotheringham et al. (2002) (Zou, 2006). We propose a statistic called the total AIC ($\mathrm{AIC}_{\mathrm{tot}}$) for the purpose of selecting the bandwidth parameter. Because of the kernel weights and the application of the adaptive lasso, the sample size and the degrees of freedom are different at each location. The total AIC is found by taking the sum over all of the observed data:

$$\mathrm{AIC}_{\mathrm{tot}} = -2 \times \sum_{i=1}^{n} \ell_i + 2\mathrm{df} \tag{24}$$

$$= \sum_{i=1}^{n} \left\{ \log \hat{\sigma}_i^2 + \hat{\sigma}_i^{-2} \left(y_i - \boldsymbol{x}_i'\hat{\boldsymbol{\beta}}_i \right)^2 \right\} + 2\mathrm{df} \tag{25}$$

What remains is to calculate df, the number of degrees of freedom used by the model. Ordinary GWR, as developed in Loader (1999) and Fotheringham et al. (2002) calculates df using the trace of the "hat" matrix, but because the GWL is not a linear smoother, there is no "hat" matrix associated with GWL. Instead, notice that df can be pulled into the summation in (25):

$$\mathrm{df} = \sum_{i=1}^{n} \left(n^{-1}\mathrm{df} \right) \tag{26}$$

Now, because we are considering the sum of local weights to be the sample size for the local models, we estimate df by $\sum_{i=1}^{n} \left\{ \left(\sum_{i'=1}^{n} w_{ii'} \right)^{-1} \mathrm{df}_i \right\}$, and the total AIC is then:

$$\mathrm{AIC}_{\mathrm{tot}} = \sum_{i=1}^{n} \left\{ \log \hat{\sigma}_i^2 + \hat{\sigma}_i^{-2} \left(y_i - \boldsymbol{x}_i' \hat{\boldsymbol{\beta}}_i \right)^2 + 2 \left(\sum_{i'=1}^{n} w_{ii'} \right)^{-1} \mathrm{df}_i \right\} \tag{27}$$

The bandwidth that minimizes (27) is found by a line search.

3.4. Confidence interval construction

Confidence intervals for the GWL's coefficient estimates can be calculated either by the bootstrap (Efron and Tibshirani, 1986) or by exploiting an assumption of normally-distributed residuals. Then the, e.g., 95% confidence interval for each regression coefficient is defined by the (2.5, 97.5) percentiles of the coefficient estimates from the bootstrap replicates.

To compute coefficient confidence intervals via the bootstrap, the observations with non-zero geographic weights are resampled uniformly with replacement for each of n_B bootstrap replicates. For each bootstrap replicate, the GWL is used to estimate regression coefficients. The local likelihood of the bootstrap replicates may be different from that of the original sample, so the adaptive lasso tuning parameter may differ for each bootstrap replicate. Since the GWL is applied independently to each bootstrap replicate, the variables selected by GWL may be different for each replicate.

Unshrunk coefficient estimates are found by using the GWL at each location for variable selection only and then estimating the coefficients for the selected variables by weighted least squares. An unshrunk bootstrap confidence interval is found by estimating the unshrunk coefficients for each of the n_B bootstrap replicates and then calculating the percentiles as above.

A third way to estimate the coefficient confidence intervals is to use the GWL for variable selection only and then to use weighted least squares for both coefficient estimation and confidence interval construction:

$$\hat{\text{se}}_{\beta_i} \quad = \quad \left(\tilde{\boldsymbol{X}}_i' \boldsymbol{W}_i \tilde{\boldsymbol{X}}_i\right)^{-1} \tilde{\boldsymbol{X}}_i' \boldsymbol{W}_i \boldsymbol{Y} \tag{28}$$

where $\tilde{\boldsymbol{X}}_i$ is the model matrix including only those variables that are selected by GWL at location i.

4. Simulation

4.1. Simulation setup

A simulation study was conducted to assess the performance of the method described in Sections 2-3. There were twelve simulation settings, each of which was simulated 100 times. For each of the twelve settings, $\boldsymbol{\beta}_1$, the true coefficient surface for $\boldsymbol{Z}_1$, was nonzero in at least part of the simulation domain. There were four other simulated covariates, but their true coefficient surfaces were zero across the area under simulation.

Data was simulated on $[0,1] \times [0,1]$, which was divided into a 30×30 grid. Each of $p = 5$ covariates $Z_1, \ldots, Z_p$ was simulated by a Gaussian random field (GRF) with mean zero and exponential spatial covariance $Cov\left(Z_{ji}, Z_{ji'}\right) = \sigma_z^2 \exp\left(-\tau_z^{-1} \delta_{ii'}\right)$ where $\sigma_z^2 = 1$ is the variance, τ_z 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 $\boldsymbol{Z}$ matrix by $\boldsymbol{R}$, where $\boldsymbol{R}$ is the Cholesky decomposition of the covariance matrix $\Sigma = \boldsymbol{R}'\boldsymbol{R}$. 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 is $y_i = \boldsymbol{z}_i'\boldsymbol{\beta}_i + \varepsilon_i$ for $i = 1, \ldots, 900$ where for simplicity the vector of additive errors $\boldsymbol{\varepsilon}$ were iid Gaussian.

The simulated data include the output y and five covariates $Z_1, \ldots, Z_5$. The true data-generating model uses only Z_1, so $Z_2, \ldots, Z_5$ are included to test the variable-selection properties of GWL.

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 β_1 (step, gradient, and parabola - see Figure 1); data was simulated both with ($\rho = 0.5$) and without ($\rho = 0$) correlation between the covariates; and simulations were made with low ($\sigma^2 = 0.25$) and high ($\sigma^2 = 1$) variance for the random noise term.

The performance of the penalized GWR methods (adaptive lasso via `lars` and via `glmnet`, and the adaptive elastic net (`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 predictors 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.

[Table 1 about here.]

[Figure 1 about here.]

[Figure 2 about here.]

4.2. Results

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.

Selection: Tables 2 - 6

MSE of $\hat{Y}(s_i)$ $(i = 1, \ldots, 5)$: Tables 22 - 26

MSE of $\hat{\beta}_1(s_i)$ $(i = 1, \ldots, 5)$: Tables 7 - 11

Bias of $\hat{\beta}_1(s_i)$ $(i = 1, \ldots, 5)$: Tables 12 - 16

Variance of $\hat{\beta}_1(s_i)$ $(i = 1, \ldots, 5)$: Tables 17 - 21

4.3. Discussion

At locations where β_1 is nonzero, X_1 usually selected for inclusion in all or nearly all of the model runs. An exception is at location four for the step function, where X_1 was included in about half of the model runs. This is probably because location four is at the very point where β_1 transitions from zero to nonzero. Selection performance was relatively poor for the step function at location one, especially for data with $\sigma^2 = 1$. For those simulations, X_1 was correctly included in around 85% of the simulations. The bias, variance, and MSE of $\hat{\beta}_1$ under the same settings were also much larger than the baseline established by the standard `gwr` algorithm. The reason(s) for the poor performance under those particular conditions is currently unknown.

Otherwise, selection performance was good, with the rate of false positive selections for $X_2 - X_5$ (and for X_1 where its true coefficient was zero) usually below 0.10. Selection (also bias, variance, and MSE of $\hat{\beta}_1$) tended to suffer worse by the change from low to high error variance than by the change from low to high collinearity amongst the predictors.

There was not a clear and consistent difference in performance between the three selection methods. It might be expected that the adaptive elastic net would outperform the adaptive lasso under greater covariate collinearity, but if such effect is real it is not apparent from this simulation. The unshrunk coefficient-estimation methods tended to exhibit more bias than the selection-plus-shrinkage methods when the true coefficient value was near zero, and vice versa when the true coefficient was not near zero. The unshrunk methods were perhaps more consistent in their performance and for that reason they are probably preferable in practice.

Bias in coefficient estimation was greater and variance less for the standard `gwr` algorithm than for the methods described here. This is probably due to the fact that the methods described here show a preference for smaller bandwidths than those select by `gwr`. Accuracy (as measured by MSE) in fitting the true Y variables was comparable for all the methods.

4.4. *Tables*

4.4.1. *Selection*

[Table 2 about here.]

[Table 3 about here.]

[Table 4 about here.]

[Table 5 about here.]

[Table 6 about here.]

4.4.2. *Estimation*

[Table 7 about here.]

[Table 8 about here.]

[Table 9 about here.]

[Table 10 about here.]

[Table 11 about here.]

[Table 12 about here.]

[Table 13 about here.]

[Table 14 about here.]

[Table 15 about here.]

[Table 16 about here.]

[Table 17 about here.]

[Table 18 about here.]

[Table 19 about here.]

[Table 20 about here.]

[Table 21 about here.]

[Table 22 about here.]

[Table 23 about here.]

[Table 24 about here.]

[Table 25 about here.]

[Table 26 about here.]

5. Data analysis

5.1. Census poverty data

An example data analysis is presented to demonstrate application of penalized GWR. In this example we use penalized GWR to do local variable selection and coefficient estimation for a varying-coefficients model of how poverty is related to a list of demographic and social variables. The data is from the U.S. Census Bureau's decennial census from 1970. This analysis looks specifically at the upper midwest states of Minnesota, Iowa, Wisconsin, Illinois, Indiana, and Michigan. This is areal data, aggregated at the county level.

Table 27 lists the variables that were considered as potential predictors of county-level poverty rate. The outcome of interest (poverty rate) is a proportion and so takes values on $[0, 1]$, but 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.

[Table 27 about here.]

5.2. Modeling

The adaptive elastic net was used for variable selection, and then coefficients for the selected variables were estimated by weighted least squares without shrinkage. The standard `gwr` algorithm was used to fit a model to the same data for the sake of comparison.

5.3. Figures

The coefficient estimates are plotted on maps of the upper midwest in Figure 3 (based on the adaptive elastic net) and Figure 4 (for standard GWR).

[Figure 3 about here.]

[Figure 4 about here.]

5.4. Discussion

It is immediately apparent that the estimated coefficient surfaces are non-constant for most variables. The same large-scale patterns appear in both figures, but with differences. First of all, the adaptive elastic net has selected a larger bandwidth than base GWR, so there is less variability in the coefficient estimates from the adaptive elastic net. This may be one reason that the adaptive elastic net coefficient estimates are less extreme than those for base 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 adaptive elastic net 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. References

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control 19*(6), 716–723.

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. Charlton (1998). Geographically weighted regression - modeling spatial nonstationarity. *The Statistician 47*(3), 431–443.

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.

Efron, B. and R. Tibshirani (1986). Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science 1*(1), 54–75.

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.

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.

Loader, C. (1999). *Local regression and likelihood.* Springer New York.

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.

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.

List of Figures

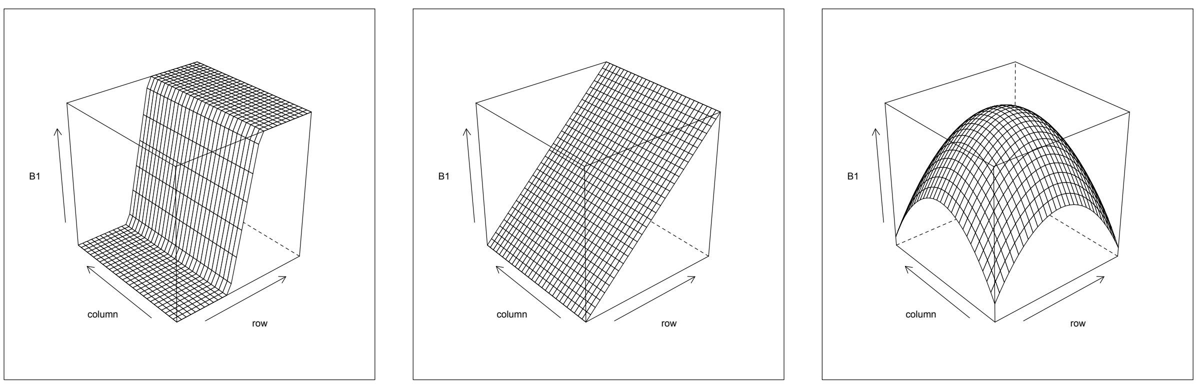

Figure 1: The actual β_1 coefficient surface used in the simulation.

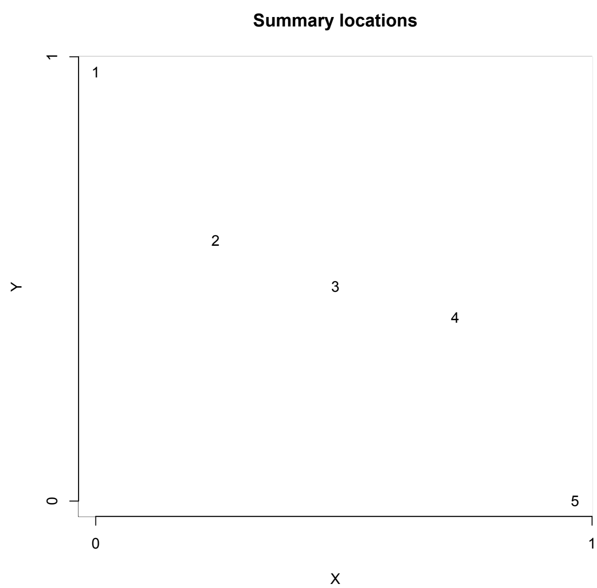

Figure 2: Locations where the variable selection and coefficient estimation of GWL were summarized.

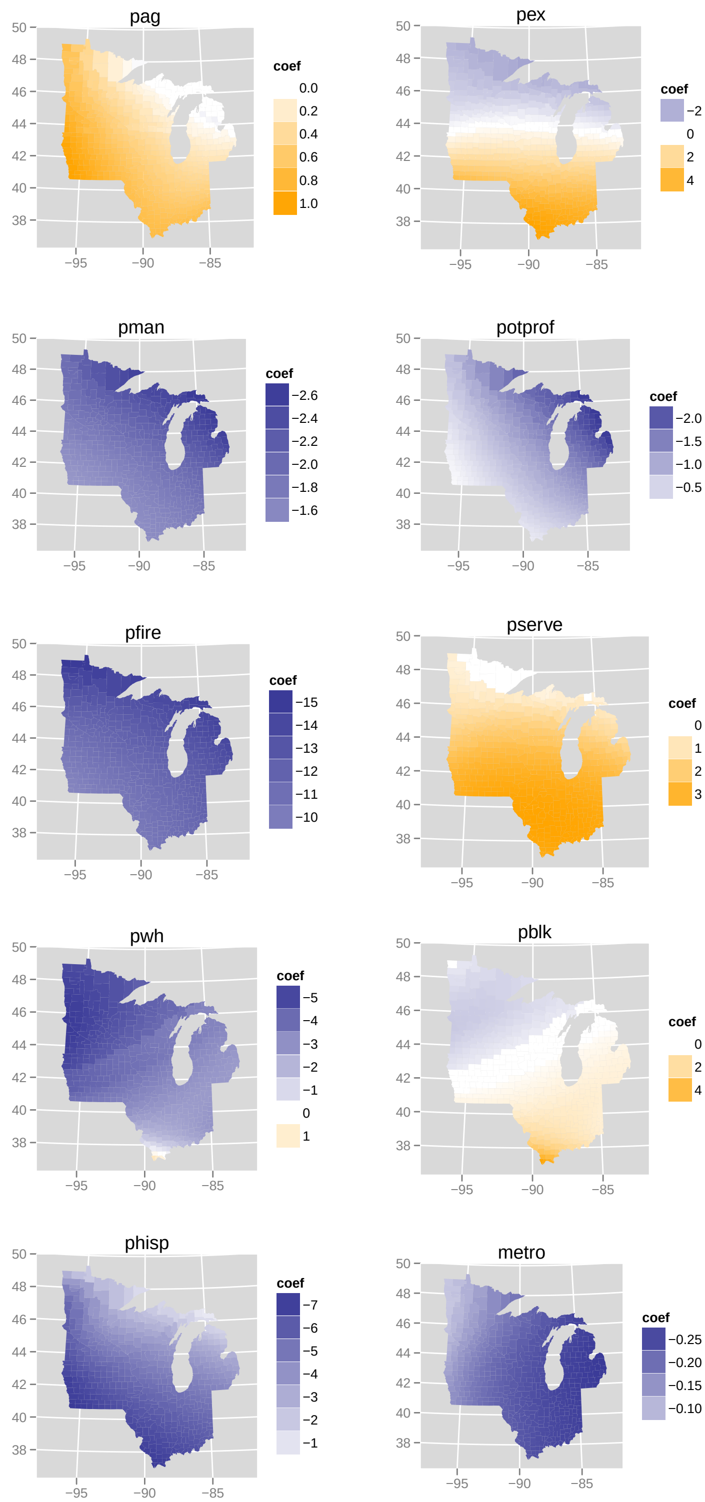

Figure 3: Coefficient surfaces for the logit of poverty rate, based on the 1970 census and estimated by the unshrunk adaptive elastic net.

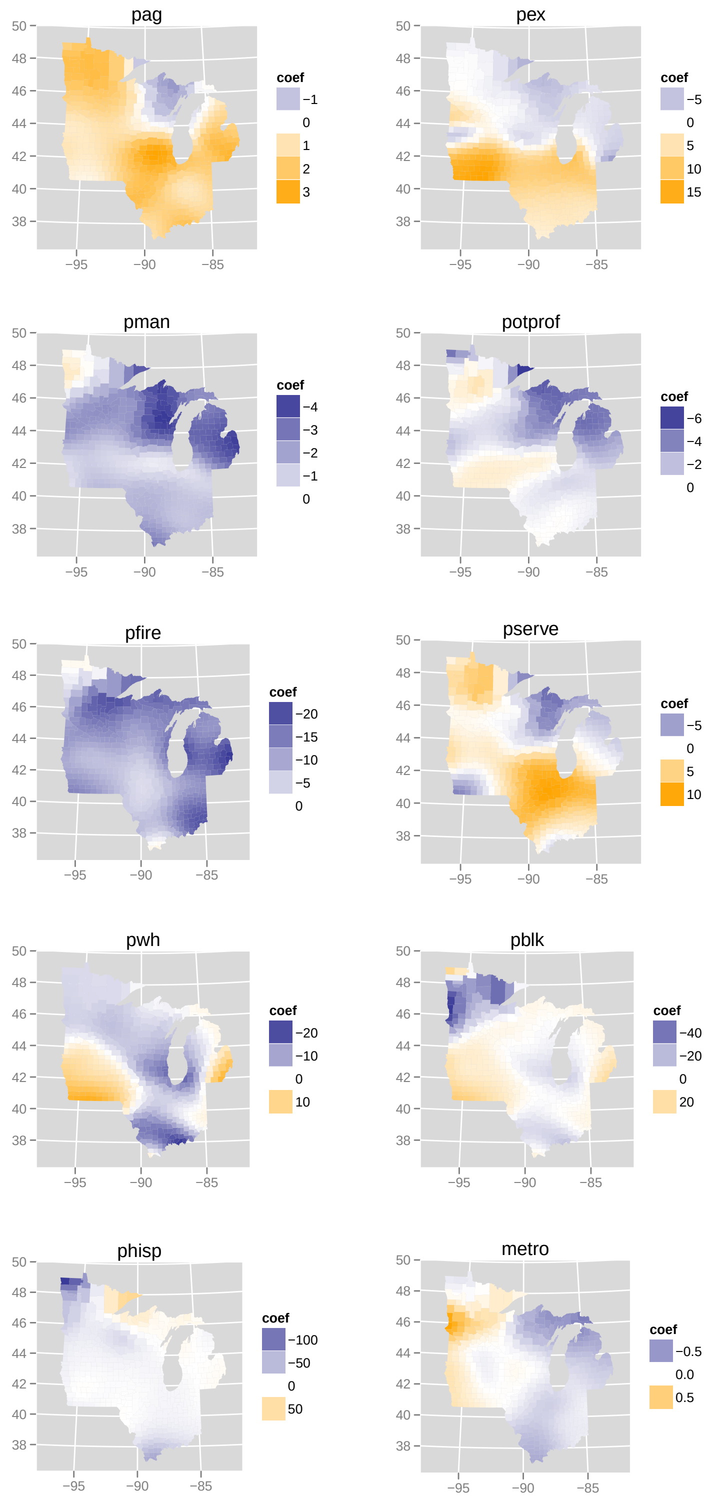

Figure 4: Coefficient surfaces for the logit of poverty rate based on the 1970 census and estimated by base GWR.

List of Tables

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.

	lars		enet		glmnet	
	β_1	β_4 - β_5	β_1	β_4 - β_5	β_1	β_4 - β_5
step	0.98	0.04	1.00	0.04	1.00	0.05
	0.89	0.09	0.86	0.09	0.82	0.07
	0.96	0.07	0.99	0.10	0.96	0.09
	0.84	0.04	0.84	0.07	0.88	0.05
gradient	1.00	0.04	1.00	0.03	1.00	0.03
	0.99	0.08	0.97	0.07	0.97	0.07
	1.00	0.07	1.00	0.06	1.00	0.04
	0.90	0.08	0.92	0.08	0.92	0.08
parabola	0.94	0.06	0.95	0.06	0.94	0.06
	0.80	0.06	0.81	0.07	0.80	0.06
	0.95	0.06	0.94	0.09	0.95	0.04
	0.78	0.12	0.79	0.12	0.80	0.12

Table 2: Selection frequency at location 1

	`lars`		`enet`		`glmnet`	
	β_1	β_4 - β_5	β_1	β_4 - β_5	β_1	β_4 - β_5
step	1.00	0.07	1.00	0.07	1.00	0.07
	1.00	0.06	1.00	0.06	1.00	0.07
	1.00	0.05	1.00	0.06	1.00	0.05
	0.99	0.03	1.00	0.07	0.99	0.04
gradient	1.00	0.10	1.00	0.08	1.00	0.07
	0.98	0.07	0.98	0.08	0.99	0.07
	1.00	0.07	1.00	0.06	1.00	0.05
	0.98	0.06	0.99	0.08	0.99	0.05
parabola	1.00	0.09	1.00	0.08	1.00	0.08
	0.97	0.12	0.98	0.11	0.98	0.10
	1.00	0.06	1.00	0.05	1.00	0.05
	0.94	0.08	0.94	0.10	0.94	0.08

Table 3: Selection frequency at location 2

	`lars`		`enet`		`glmnet`	
	β_1	β_4 - β_5	β_1	β_4 - β_5	β_1	β_4 - β_5
step	0.99	0.05	0.99	0.06	0.99	0.06
	0.84	0.08	0.84	0.08	0.82	0.07
	0.96	0.05	0.97	0.08	0.92	0.04
	0.78	0.08	0.81	0.11	0.80	0.08
gradient	1.00	0.09	1.00	0.08	1.00	0.07
	0.98	0.08	0.95	0.08	0.96	0.07
	1.00	0.07	1.00	0.06	1.00	0.04
	0.93	0.09	0.95	0.09	0.94	0.09
parabola	1.00	0.09	1.00	0.09	1.00	0.09
	0.96	0.10	0.97	0.09	0.97	0.10
	1.00	0.08	1.00	0.07	1.00	0.07
	0.93	0.10	0.94	0.10	0.96	0.10

Table 4: Selection frequency at location 3

	`lars`		`enet`		`glmnet`	
	β_1	β_4 - β_5	β_1	β_4 - β_5	β_1	β_4 - β_5
step	0.57	0.08	0.64	0.06	0.59	0.06
	0.48	0.07	0.48	0.07	0.49	0.07
	0.45	0.08	0.51	0.12	0.40	0.07
	0.53	0.08	0.52	0.07	0.51	0.07
gradient	1.00	0.06	1.00	0.06	1.00	0.06
	0.98	0.07	0.95	0.07	0.93	0.06
	1.00	0.09	1.00	0.08	1.00	0.10
	0.96	0.07	0.95	0.11	0.95	0.08
parabola	1.00	0.09	1.00	0.08	1.00	0.08
	0.93	0.07	0.92	0.08	0.94	0.08
	1.00	0.08	1.00	0.08	1.00	0.08
	0.96	0.08	0.96	0.09	0.96	0.09

Table 5: Selection frequency at location 4

	lars		enet		glmnet	
	β_1	β_4 - β_5	β_1	β_4 - β_5	β_1	β_4 - β_5
step	0.04	0.03	0.03	0.03	0.03	0.03
	0.07	0.05	0.06	0.04	0.04	0.05
	0.02	0.04	0.02	0.03	0.03	0.05
	0.05	0.04	0.04	0.03	0.06	0.06
gradient	0.92	0.05	0.93	0.05	0.94	0.04
	0.71	0.08	0.70	0.07	0.70	0.07
	0.93	0.10	0.95	0.14	0.95	0.10
	0.60	0.07	0.63	0.13	0.64	0.06
parabola	0.93	0.10	0.93	0.09	0.92	0.10
	0.80	0.05	0.81	0.05	0.79	0.05
	0.93	0.07	0.94	0.12	0.94	0.07
	0.81	0.09	0.81	0.11	0.83	0.08

Table 6: Selection frequency at location 5

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.046	0.025	*0.023*	0.151	0.127	0.124	0.082	**0.005**
	0.146	0.186	0.216	0.290	0.376	0.375	*0.134*	**0.009**
	0.072	*0.045*	0.073	0.172	0.134	0.205	0.101	**0.011**
	0.214	0.218	0.179	0.441	0.425	0.369	*0.154*	**0.022**
gradient	0.066	0.069	0.070	0.007	**0.007**	*0.007*	0.010	0.016
	0.084	0.094	0.096	0.161	0.078	0.085	*0.045*	**0.042**
	0.065	0.070	0.069	0.009	**0.007**	*0.008*	0.009	0.019
	0.161	0.149	0.144	0.149	0.123	0.121	**0.040**	*0.050*
parabola	0.074	0.075	0.074	0.020	*0.020*	**0.020**	0.022	0.105
	0.079	0.078	0.077	0.041	*0.040*	**0.039**	0.063	0.106
	0.077	0.069	0.076	0.024	**0.018**	0.023	*0.023*	0.099
	0.083	0.072	0.083	*0.048*	**0.044**	0.050	0.067	0.110

Table 7: Mean squared error of estimates for β_1 at location 1 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.024	0.024	0.024	**0.020**	0.021	*0.021*	0.021	0.042
	0.061	0.063	0.068	*0.050*	0.054	0.056	**0.042**	0.070
	0.022	0.027	0.021	**0.017**	0.021	*0.017*	0.018	0.044
	0.069	0.071	0.071	0.057	*0.056*	0.061	**0.043**	0.075
gradient	0.003	0.003	0.003	*0.001*	0.001	0.001	**0.001**	0.001
	0.014	0.013	0.008	0.015	0.013	0.009	*0.002*	**0.002**
	0.003	0.003	0.003	0.001	*0.001*	**0.001**	0.001	0.002
	0.015	0.012	0.012	0.014	0.011	0.011	**0.003**	*0.004*
parabola	*0.005*	0.005	0.005	0.007	0.007	0.007	**0.004**	0.007
	0.018	0.016	0.016	0.026	0.021	0.022	*0.008*	**0.007**
	0.007	0.007	*0.007*	0.011	0.010	0.009	**0.004**	0.008
	0.020	0.022	0.020	0.022	0.022	0.023	**0.007**	*0.009*

Table 8: Mean squared error of estimates for β_1 at location 2 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.011	0.011	0.010	0.007	0.007	0.007	**0.004**	*0.005*
	0.043	0.043	0.047	0.049	0.049	0.054	*0.009*	**0.008**
	0.016	0.014	0.022	0.013	0.011	0.021	**0.005**	*0.005*
	0.048	0.047	0.045	0.049	0.045	0.044	*0.008*	**0.008**
gradient	0.001	0.001	0.001	**0.001**	0.001	0.001	*0.001*	0.001
	0.007	0.017	0.015	0.007	0.019	0.017	*0.002*	**0.002**
	0.001	0.001	0.000	0.000	*0.000*	**0.000**	0.001	0.002
	0.022	0.017	0.019	0.023	0.017	0.021	**0.002**	*0.003*
parabola	0.015	0.015	0.015	0.015	0.015	*0.015*	**0.005**	0.022
	0.032	0.029	0.029	0.030	0.027	0.027	**0.012**	*0.023*
	0.019	0.018	0.019	0.018	*0.017*	0.018	**0.005**	0.024
	0.037	0.037	0.030	0.037	0.034	0.029	**0.012**	*0.024*

Table 9: Mean squared error of estimates for β_1 at location 3 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	*0.014*	0.014	**0.014**	0.017	0.019	0.018	0.021	0.042
	0.037	**0.036**	0.039	0.039	0.042	0.046	0.047	0.074
	0.010	0.012	*0.011*	0.013	0.016	0.014	0.020	0.044
	0.038	**0.028**	0.038	0.048	0.047	0.048	0.043	0.082
gradient	0.003	0.003	0.003	0.002	0.001	*0.001*	**0.001**	0.001
	0.009	0.014	0.016	0.007	0.012	0.015	**0.002**	*0.003*
	0.003	0.002	0.003	0.002	0.001	*0.001*	**0.001**	0.002
	0.013	0.015	0.014	0.013	0.014	0.014	**0.003**	*0.004*
parabola	0.006	0.006	*0.006*	0.009	0.009	0.009	**0.004**	0.008
	0.025	0.027	0.023	0.027	0.029	0.024	*0.010*	**0.009**
	0.008	0.008	0.008	0.011	0.011	0.011	**0.004**	0.010
	0.018	0.020	0.017	0.022	0.022	0.022	**0.009**	*0.010*

Table 10: Mean squared error of estimates for β_1 at location 4 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.002	*0.001*	0.002	0.006	0.004	0.004	**0.000**	0.007
	0.003	0.006	*0.002*	0.016	0.024	0.009	**0.000**	0.011
	0.002	0.002	0.003	0.009	0.009	0.009	**0.000**	0.010
	0.017	*0.004*	0.022	0.046	0.038	0.043	**0.000**	0.015
gradient	0.067	0.068	0.069	0.004	0.004	*0.004*	**0.000**	0.016
	0.054	0.051	0.052	*0.019*	0.019	0.019	**0.000**	0.044
	0.062	0.060	0.064	0.009	0.010	*0.007*	**0.000**	0.021
	0.050	0.047	0.053	*0.017*	0.020	0.017	**0.000**	0.051
parabola	0.074	0.075	0.075	**0.018**	0.020	0.021	*0.020*	0.104
	0.075	0.074	0.073	0.024	**0.022**	*0.023*	0.055	0.104
	0.077	0.069	0.076	*0.021*	0.023	**0.020**	0.025	0.099
	0.081	0.075	0.081	0.037	*0.036*	**0.035**	0.042	0.113

Table 11: Mean squared error of estimates for β_1 at location 5 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	-0.057	-0.029	-0.020	**-0.003**	0.038	0.033	0.034	*-0.004*
	-0.150	-0.195	-0.211	**-0.004**	-0.082	-0.075	0.053	*-0.017*
	-0.090	-0.077	-0.073	**0.002**	0.028	*-0.014*	0.050	-0.017
	-0.208	-0.214	-0.167	-0.053	-0.067	-0.068	*0.035*	**0.006**
gradient	-0.246	-0.248	-0.253	**0.008**	*0.010*	0.011	0.011	-0.111
	-0.233	-0.242	-0.248	0.026	*-0.014*	-0.022	**-0.007**	-0.182
	-0.245	-0.259	-0.255	-0.004	*-0.003*	**-0.000**	0.003	-0.112
	-0.306	-0.290	-0.287	-0.083	*-0.055*	-0.080	**0.017**	-0.197
parabola	0.252	0.254	0.252	0.091	0.090	*0.090*	**0.029**	0.323
	0.240	0.241	0.236	0.130	0.129	*0.126*	**0.070**	0.322
	0.262	0.245	0.259	0.091	*0.083*	0.101	**0.041**	0.313
	0.239	0.222	0.242	0.121	*0.095*	0.115	**0.068**	0.323

Table 12: Bias of estimates for β_1 at location 1 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	-0.117	-0.119	-0.119	**-0.106**	-0.110	*-0.110*	-0.124	-0.196
	-0.178	-0.178	-0.186	*-0.145*	**-0.145**	-0.150	-0.175	-0.253
	-0.103	-0.130	-0.099	**-0.083**	-0.103	*-0.083*	-0.110	-0.199
	-0.212	-0.221	-0.216	*-0.175*	**-0.167**	-0.184	-0.182	-0.263
gradient	-0.046	-0.047	-0.048	**0.000**	0.002	*0.001*	0.004	0.002
	-0.034	-0.044	-0.035	0.009	**0.003**	0.011	*0.008*	-0.011
	-0.039	-0.046	-0.043	0.003	0.003	0.003	**0.002**	*0.002*
	-0.058	-0.062	-0.056	-0.016	-0.011	*-0.009*	**-0.002**	-0.020
parabola	-0.062	-0.063	*-0.062*	-0.073	-0.074	-0.072	**-0.048**	-0.079
	-0.067	-0.066	**-0.063**	-0.069	-0.068	*-0.065*	-0.072	-0.078
	-0.073	-0.079	-0.076	-0.087	-0.091	-0.087	**-0.052**	-0.085
	-0.093	-0.104	-0.097	-0.101	-0.102	-0.101	**-0.065**	*-0.078*

Table 13: Bias of estimates for β_1 at location 2 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	-0.014	*-0.014*	**-0.010**	0.018	0.017	0.015	0.021	0.040
	-0.026	-0.027	-0.031	*0.006*	0.009	**0.004**	0.050	0.059
	-0.044	-0.056	-0.056	*-0.013*	**-0.009**	-0.030	0.017	0.034
	-0.083	-0.094	-0.077	-0.059	-0.056	-0.056	**0.017**	*0.055*
gradient	0.007	0.005	0.005	0.003	*0.002*	**0.001**	0.003	0.002
	-0.004	-0.017	-0.012	**-0.003**	-0.013	-0.007	*0.003*	0.006
	0.006	0.005	0.006	0.001	*0.000*	0.000	0.003	**0.000**
	-0.019	-0.029	-0.018	-0.012	-0.022	-0.013	**-0.002**	*0.003*
parabola	-0.107	-0.107	-0.106	-0.103	-0.103	*-0.102*	**-0.057**	-0.148
	-0.141	-0.136	-0.132	-0.129	-0.122	*-0.121*	**-0.090**	-0.147
	-0.125	-0.127	-0.125	-0.123	*-0.121*	-0.121	**-0.060**	-0.154
	-0.147	-0.156	-0.131	-0.137	-0.136	*-0.121*	**-0.092**	-0.147

Table 14: Bias of estimates for β_1 at location 3 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	**0.047**	0.059	*0.049*	0.058	0.074	0.065	0.129	0.196
	0.072	*0.075*	0.076	0.080	0.088	0.090	0.193	0.263
	0.014	0.027	**0.010**	0.027	0.043	0.020	0.129	0.199
	0.091	**0.073**	*0.089*	0.108	0.105	0.105	0.189	0.275
gradient	0.047	0.043	0.045	0.009	*0.006*	0.007	**0.004**	0.008
	0.021	0.006	**0.002**	*-0.003*	-0.014	-0.023	0.004	0.020
	0.039	0.038	0.039	*0.001*	-0.001	-0.001	**0.000**	-0.001
	0.000	-0.009	0.003	-0.009	-0.021	-0.013	*-0.003*	0.014
parabola	*-0.066*	-0.070	-0.069	-0.078	-0.083	-0.081	**-0.051**	-0.088
	-0.113	-0.119	-0.110	-0.119	-0.126	-0.115	**-0.081**	*-0.088*
	-0.080	-0.085	*-0.078*	-0.092	-0.095	-0.090	**-0.055**	-0.095
	-0.088	-0.099	-0.088	-0.094	-0.099	-0.094	**-0.079**	*-0.086*

Table 15: Bias of estimates for β_1 at location 4 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	-0.009	*-0.006*	-0.006	-0.015	-0.009	-0.010	**0.000**	-0.006
	0.001	-0.009	*-0.000*	-0.018	-0.025	-0.008	**0.000**	-0.011
	-0.006	*-0.005*	-0.009	-0.010	-0.009	-0.012	**0.000**	-0.009
	-0.012	-0.011	-0.011	-0.026	-0.036	-0.021	**0.000**	*-0.007*
gradient	0.246	0.249	0.253	0.002	*0.002*	0.003	**0.000**	0.113
	0.191	0.182	0.186	0.007	*-0.001*	0.004	**0.000**	0.187
	0.234	0.234	0.243	0.007	*-0.001*	0.008	**0.000**	0.115
	0.168	0.165	0.179	0.030	*0.024*	0.029	**0.000**	0.190
parabola	0.253	0.256	0.253	*0.085*	0.091	0.089	**0.022**	0.321
	0.234	0.233	0.228	0.095	*0.088*	0.089	**0.004**	0.319
	0.257	0.243	0.257	0.093	*0.079*	0.088	**0.052**	0.313
	0.243	0.232	0.246	0.120	*0.096*	0.110	**0.069**	0.328

Table 16: Bias of estimates for β_1 at location 5 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.043	0.024	*0.023*	0.152	0.127	0.124	0.081	**0.005**
	0.125	0.149	0.173	0.293	0.373	0.373	0.133	**0.009**
	0.064	*0.040*	0.068	0.173	0.134	0.207	0.099	**0.011**
	0.173	0.174	*0.153*	0.443	0.424	0.368	0.154	**0.022**
gradient	*0.006*	0.007	0.006	0.008	0.007	0.007	0.010	**0.004**
	0.030	0.035	0.035	0.162	0.079	0.085	0.046	**0.009**
	0.005	**0.003**	*0.004*	0.009	0.007	0.008	0.009	0.007
	0.068	0.066	0.062	0.143	0.121	0.116	*0.040*	**0.011**
parabola	0.010	*0.010*	0.010	0.012	0.012	0.012	0.022	**0.001**
	0.021	*0.020*	0.021	0.024	0.024	0.024	0.058	**0.002**
	0.009	0.009	0.009	0.015	0.011	0.013	0.021	**0.001**
	0.026	*0.023*	0.025	0.034	0.035	0.037	0.064	**0.006**

Table 17: Variance of estimates for β_1 at location 1 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.010	0.010	0.010	0.009	0.009	0.009	*0.006*	**0.003**
	0.029	0.032	0.034	0.029	0.033	0.034	*0.012*	**0.006**
	0.011	0.011	0.012	0.010	0.010	0.010	*0.006*	**0.005**
	0.024	0.022	0.025	0.026	0.028	0.028	*0.010*	**0.006**
gradient	*0.001*	0.001	0.001	0.001	0.001	0.001	**0.001**	0.001
	0.013	0.011	0.007	0.016	0.013	0.009	*0.002*	**0.002**
	0.001	*0.001*	0.001	0.001	0.001	**0.001**	0.001	0.002
	0.011	0.008	0.009	0.014	0.011	0.011	**0.003**	*0.003*
parabola	*0.001*	0.001	0.001	0.002	0.002	0.002	0.001	**0.000**
	0.014	0.012	0.012	0.021	0.017	0.018	*0.003*	**0.001**
	0.001	0.001	*0.001*	0.003	0.002	0.002	0.001	**0.000**
	0.012	0.011	0.011	0.012	0.012	0.013	*0.003*	**0.003**

Table 18: Variance of estimates for β_1 at location 2 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.011	0.011	0.010	0.007	0.007	0.007	*0.004*	**0.003**
	0.043	0.043	0.047	0.050	0.050	0.055	*0.007*	**0.004**
	0.014	0.011	0.019	0.013	0.011	0.020	*0.004*	**0.004**
	0.041	0.039	0.039	0.046	0.043	0.042	*0.008*	**0.005**
gradient	0.001	0.001	0.001	**0.001**	0.001	0.001	*0.001*	0.001
	0.007	0.017	0.015	0.007	0.019	0.017	*0.002*	**0.002**
	0.001	0.001	0.000	0.000	*0.000*	**0.000**	0.001	0.002
	0.022	0.016	0.019	0.023	0.016	0.021	**0.002**	*0.003*
parabola	0.004	0.004	0.004	0.005	0.005	0.005	*0.002*	**0.000**
	0.012	0.010	0.012	0.014	0.013	0.013	*0.004*	**0.001**
	0.003	0.002	0.003	0.003	0.003	0.003	*0.002*	**0.000**
	0.016	0.013	0.013	0.018	0.016	0.015	*0.004*	**0.002**

Table 19: Variance of estimates for β_1 at location 3 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.012	0.011	0.012	0.014	0.014	0.014	*0.004*	**0.003**
	0.032	0.030	0.033	0.033	0.035	0.038	*0.009*	**0.005**
	0.010	0.011	0.011	0.013	0.014	0.013	**0.003**	*0.004*
	0.029	0.023	0.031	0.037	0.037	0.037	*0.007*	**0.006**
gradient	0.001	0.001	0.001	0.002	0.001	*0.001*	**0.001**	0.001
	0.008	0.014	0.017	0.007	0.012	0.014	**0.002**	*0.002*
	0.001	0.001	0.001	0.002	0.001	*0.001*	**0.001**	0.002
	0.013	0.015	0.014	0.013	0.014	0.014	**0.003**	*0.004*
parabola	0.002	*0.001*	0.001	0.003	0.002	0.002	0.001	**0.000**
	0.012	0.013	0.011	0.013	0.013	0.011	*0.003*	**0.001**
	0.001	*0.001*	0.002	0.003	0.001	0.003	0.001	**0.001**
	0.011	0.010	0.010	0.014	0.012	0.013	*0.003*	**0.003**

Table 20: Variance of estimates for β_1 at location 4 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.002	*0.001*	0.002	0.006	0.004	0.004	**0.000**	0.007
	0.003	0.006	*0.002*	0.016	0.024	0.009	**0.000**	0.011
	0.002	0.002	0.003	0.009	0.009	0.009	**0.000**	0.010
	0.017	*0.004*	0.022	0.045	0.037	0.043	**0.000**	0.015
gradient	0.007	0.006	0.005	0.004	0.004	0.004	**0.000**	*0.003*
	0.018	0.018	0.018	0.019	0.020	0.020	**0.000**	*0.009*
	0.007	0.005	*0.005*	0.009	0.010	0.007	**0.000**	0.008
	0.022	0.020	0.021	0.016	0.020	0.016	**0.000**	*0.015*
parabola	0.010	*0.010*	0.011	0.011	0.012	0.013	0.019	**0.001**
	0.021	0.019	0.021	0.016	*0.014*	0.015	0.056	**0.003**
	0.011	0.010	*0.010*	0.013	0.017	0.012	0.023	**0.001**
	0.022	0.021	*0.021*	0.023	0.027	0.023	0.038	**0.005**

Table 21: Variance of estimates for β_1 at location 5 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.130	**0.100**	0.101	0.130	*0.100*	0.101	0.111	0.118
	0.483	0.594	0.564	*0.483*	0.594	0.564	0.694	0.850
	0.196	*0.151*	0.169	0.196	**0.151**	0.169	0.213	0.247
	0.563	0.559	**0.552**	0.563	0.559	*0.552*	0.757	0.895
gradient	0.235	0.224	0.232	0.235	0.224	0.232	*0.223*	**0.222**
	0.693	**0.669**	0.671	0.693	*0.669*	0.671	0.723	0.757
	0.257	0.258	0.260	0.257	0.258	0.260	*0.237*	**0.210**
	0.724	0.733	0.731	*0.724*	0.733	0.731	0.815	0.784
parabola	0.145	0.142	**0.140**	0.145	0.142	*0.140*	0.157	0.248
	1.275	*1.257*	1.266	1.275	1.257	1.266	**1.153**	1.466
	0.299	*0.285*	0.295	0.299	0.285	0.295	**0.270**	0.434
	0.835	**0.801**	0.806	0.835	*0.801*	0.806	0.862	0.986

Table 22: Mean squared error of estimates for Y at location 1 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	**0.193**	0.196	0.194	*0.193*	0.196	0.194	0.225	0.244
	1.023	1.019	**1.001**	1.023	1.019	*1.001*	1.171	1.123
	0.270	0.275	0.273	*0.270*	0.275	0.273	0.311	0.332
	0.973	**0.897**	0.953	0.973	*0.897*	0.953	1.000	1.048
gradient	0.218	0.216	0.218	0.218	*0.216*	0.218	0.221	**0.210**
	0.828	**0.814**	0.836	0.828	*0.814*	0.836	0.863	0.832
	0.257	0.257	0.257	0.257	0.257	0.257	*0.257*	**0.247**
	0.795	0.819	0.803	*0.795*	0.819	0.803	0.822	0.799
parabola	*0.192*	0.195	0.193	**0.192**	0.195	0.193	0.204	0.199
	1.139	1.193	1.152	*1.139*	1.193	1.152	1.204	1.214
	0.248	0.254	0.250	0.248	0.254	0.250	**0.246**	0.257
	1.165	**1.150**	1.181	1.165	*1.150*	1.181	1.180	1.199

Table 23: Mean squared error of estimates for Y at location 2 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	0.238	*0.232*	0.233	0.238	**0.232**	0.233	0.255	0.262
	0.852	0.850	**0.833**	0.852	0.850	*0.833*	1.025	1.020
	0.246	0.257	0.246	*0.246*	0.257	0.246	0.275	0.265
	0.622	**0.620**	0.652	0.622	*0.620*	0.652	0.673	0.664
gradient	0.241	0.241	*0.241*	0.241	0.241	0.241	0.249	**0.229**
	1.113	**1.094**	1.096	1.113	*1.094*	1.096	1.135	1.117
	0.311	0.311	0.313	0.311	*0.311*	0.313	0.314	**0.305**
	1.256	**1.244**	1.252	1.256	*1.244*	1.252	1.289	1.259
parabola	0.214	0.214	*0.213*	0.214	0.214	**0.213**	0.221	0.233
	1.022	1.024	1.029	*1.022*	1.024	1.029	1.075	1.081
	0.241	0.241	0.243	*0.241*	0.241	0.243	**0.238**	0.252
	0.982	0.977	*0.975*	0.982	0.977	**0.975**	0.990	1.006

Table 24: Mean squared error of estimates for Y at location 3 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	**0.234**	0.241	0.250	*0.234*	0.241	0.250	0.269	0.288
	0.984	0.950	**0.950**	0.984	0.950	*0.950*	1.045	1.053
	0.260	0.293	**0.259**	0.260	0.293	*0.259*	0.304	0.333
	0.715	0.748	0.743	*0.715*	0.748	0.743	0.815	0.802
gradient	0.277	0.276	0.277	0.277	*0.276*	0.277	0.281	**0.262**
	0.874	0.882	0.875	0.874	0.882	0.875	0.885	**0.870**
	0.204	0.204	0.202	0.204	0.204	*0.202*	0.206	**0.201**
	0.776	0.785	**0.776**	0.776	0.785	*0.776*	0.807	0.810
parabola	0.249	0.246	0.247	0.249	*0.246*	0.247	0.247	**0.245**
	1.417	1.405	**1.378**	1.417	1.405	*1.378*	1.387	1.383
	0.306	0.306	0.304	0.306	0.306	0.304	**0.297**	*0.303*
	1.031	**0.999**	1.022	1.031	*0.999*	1.022	1.072	1.058

Table 25: Mean squared error of estimates for Y at location 4 (**minimum**, *next best*).

	lars	enet	glmnet	unshrunk.lars	unshrunk.enet	unshrunk.glmnet	oracular	gwr
step	**0.219**	0.231	0.224	*0.219*	0.231	0.224	0.293	0.234
	0.701	**0.675**	0.697	0.701	*0.675*	0.697	0.782	0.716
	0.206	0.259	**0.203**	0.206	0.259	*0.203*	0.278	0.238
	0.889	0.961	0.915	**0.889**	0.961	0.915	1.127	0.972
gradient	0.198	**0.197**	0.202	0.198	*0.197*	0.202	0.222	0.202
	1.245	1.257	1.256	*1.245*	1.257	1.256	1.289	1.275
	0.216	0.219	0.220	0.216	0.219	0.220	0.231	**0.204**
	0.877	0.919	0.884	*0.877*	0.919	0.884	1.068	0.996
parabola	**0.223**	0.230	0.225	*0.223*	0.230	0.225	0.223	0.328
	0.950	0.952	*0.948*	0.950	0.952	**0.948**	0.963	1.037
	0.199	*0.192*	0.201	0.199	0.192	0.201	**0.190**	0.282
	0.842	0.861	0.848	*0.842*	0.861	0.848	0.870	1.016

Table 26: Mean squared error of estimates for Y at location 5 (**minimum**, *next best*).

Variable name	Description
`pag`	Proportion working in agriculture
`pex`	Proportion working in extraction (mining)
`pman`	Proportion working in manufacturing
`pserve`	Proportion working in services
`pfire`	Proportion working in finance, insurance, and real estate
`potprof`	Proportion working in other professions
`pwh`	Proportion who are white
`pblk`	Proportion who are black
`phisp`	Proportion who are hispanic
`metro`	Is the county in a metropolitan area?

Table 27: Description of the variables used in the census-data example