Smoothing spline ANOVA for super-large samples: scalable computation via rounding parameters

Nathaniel E. Helwig* and Ping Ma

In the current era of big data, researchers routinely collect and analyze data of super-large sample sizes. Data-oriented statistical methods have been developed to extract information from super-large data. Smoothing spline ANOVA (SSANOVA) is a promising approach for extracting information from noisy data; however, the heavy computational cost of SSANOVA hinders its wide application. In this paper, we propose a new algorithm for fitting SSANOVA models to super-large sample data. In this algorithm, we introduce rounding parameters to make the computation scalable. To demonstrate the benefits of the rounding parameters, we present a simulation study and a real data example using electroencephalography data. Our results reveal that (using the rounding parameters) a researcher can fit nonparametric regression models to very large samples within a few seconds using a standard laptop or tablet computer.

AMS 2000 subject classifications: Primary 62G08, 65D07; secondary 65D10.
Keywords and phrases: Smoothing spline ANOVA, Rounding parameter, Scalable algorithm.

1. INTRODUCTION

In the current era of big data, it is common for researchers to collect super-large sample data ranging from hundreds of thousands to hundreds of millions of observations. The ambitious BRAIN Initiative of NIH is expected to bring a torrent of data, e.g., 100 terabytes of data per day from a single brain lab. These super-large datasets provide a wealth of information. To effectively extract the information, numerous data-oriented statistical learning methods have been developed. Among these methods, data-driven nonparametric regression models [see 20, 21] have achieved remarkable success in identifying subtle patterns and discovering functional relationships in large noisy data; such models require few assumptions about the observed data, but produce a powerful prediction.

For example, smoothing splines [see 21, 22] offer a powerful and flexible framework for nonparametric modeling. Smoothing spline analysis of variance (SSANOVA) models [3] further expand the research horizon of the smoothing spline; SSANOVAs can model multivariate data and provide nice interpretability of the modeling and prediction outcome. Furthermore, assuming that the smoothing parameters are selected via cross-validation, SSANOVA models have been shown to have desirable asymptotic properties [see 3, 12, 22]. The main drawback of the SSANOVA approach is its computational expense: the computational complexity of SSANOVA is on the order of $O(n^3)$, where n is sample size.

Over the years, many efforts have been made to design scalable algorithms for SSANOVA. Generalized additive models [GAMs; 5, 24] provide scalable computation at the price of eliminating or reparameterizing all interaction terms of an SSANOVA model. By collapsing similar subspaces, Helwig and Ma [10] provide an algorithm for modeling all interactions with affordable computational complexity. However, even using the most efficient SSANOVA approximation [11, 13] and algorithm [10], the computational burden grows linearly with the sample size, which makes the approach impractical for analyzing super-large datasets.

One possibility is to fit the model to a subset of the observed data. For example, when analyzing ultra large datasets, Ma, Mahoney and Yu [14] suggest fitting regression models to a randomly selected influential sample of the full dataset. This sort of smart-sampling approach works well, as long as a representative sample of observations is selected for analysis; however, the fitted model varies from time to time as the subsample is randomly taken. Furthermore, determining the appropriate size of the subsample could be difficult in some situations.

In this paper, we propose a new approach for fitting SSANOVA models to super-large samples. Specifically, we introduce user-tunable rounding parameters in the SSANOVA model, which makes it possible to control the precision of each predictor. As we demonstrate, fitting a nonparametric regression model to the rounded data can result in substantial computational savings without introducing much bias to the resulting estimate. In the following sections, we provide a brief introduction to SSANOVA (Section 2), develop the concept of rounding parameters for nonparametric regression (Section 3), present finite-sample and asymptotic results concerning the quality of the rounded SSANOVA estimator (Section 4), demonstrate the benefits of the rounding parameters with a simulation study (Section 5), and provide an example with real data to reveal the practical potential of the rounding parameters (Section 6).

*Corresponding author.

2. SMOOTHING SPLINES

2.1 Overview

A typical (Gaussian) nonparametric regression model has the form

$$(1) \qquad y_i = \eta(\mathbf{x}_i) + e_i$$

where $y_i \in \mathbb{R}$ is the response variable, $\mathbf{x}_i \equiv (x_{i1}, \ldots, x_{ip})$ is the predictor vector, η is the unknown smooth function relating the response and predictors, and $e_i \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$ is unknown, normally-distributed measurement error [see 3, 20, 22]. Typically, η is estimated by minimizing the penalized least-squares functional

$$(2) \qquad (1/n) \sum_{i=1}^{n} (y_i - \eta(\mathbf{x}_i))^2 + \lambda J(\eta)$$

where the nonnegative penalty functional J quantifies the roughness of η, and the smoothing parameter $\lambda \in (0, \infty)$ balances the trade-off between fitting the data and smoothing η.

Given fixed smoothing parameters and a set of selected knots $\{\breve{\mathbf{x}}_h\}_{h=1}^{q}$, the η_λ minimizing Equation (2) can be approximated using

$$(3) \qquad \eta_\lambda(\mathbf{x}) = \sum_{v=1}^{m} d_v \phi_v(\mathbf{x}) + \sum_{h=1}^{q} c_h \rho_{\mathrm{c}}(\mathbf{x}, \breve{\mathbf{x}}_h)$$

where $\{\phi_v\}_{v=1}^{m}$ are functions spanning the null space (i.e., $J(\phi_v) = 0$), ρ_{c} is the reproducing kernel (RK) of the contrast space (i.e., $J(\rho_{\mathrm{c}}) > 0$), and $\mathbf{d} = \{d_v\}_{m \times 1}$ and $\mathbf{c} = \{c_h\}_{q \times 1}$ are the unknown function coefficients [see 10, 11, 4]. Note that $\rho_{\mathrm{c}} = \sum_{k=1}^{s} \theta_k \rho_k^*$, where ρ_k^* denotes the RK of the k-th orthogonal contrast space, and $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_s)'$ are additional smoothing parameters with $\theta_k \in (0, \infty)$.

2.2 Estimation

Inserting the optimal representation in Equation (3) into the penalized least-squared functional in Equation (2) produces

$$(4) \qquad (1/n)\|\mathbf{y} - \mathbf{Kd} - \mathbf{J}_{\boldsymbol{\theta}}\mathbf{c}\|^2 + \lambda \mathbf{c}' \mathbf{Q}_{\boldsymbol{\theta}} \mathbf{c}$$

where $\| \cdot \|^2$ denotes the squared Frobenius norm, $\mathbf{y} \equiv \{y_i\}_{n \times 1}$, $\mathbf{K} \equiv \{\phi_v(\mathbf{x}_i)\}_{n \times m}$ for $i \in \{1, \ldots, n\}$ and $v \in \{1, \ldots, m\}$, $\mathbf{J}_{\boldsymbol{\theta}} = \sum_{k=1}^{s} \theta_k \mathbf{J}_k$ with $\mathbf{J}_k \equiv \{\rho_k^*(\mathbf{x}_i, \breve{\mathbf{x}}_h)\}_{n \times q}$ for $i \in \{1, \ldots, n\}$ and $h \in \{1, \ldots, q\}$, and $\mathbf{Q}_{\boldsymbol{\theta}} = \sum_{k=1}^{s} \theta_k \mathbf{Q}_k$ where $\mathbf{Q}_k \equiv \{\rho_k^*(\breve{\mathbf{x}}_g, \breve{\mathbf{x}}_h)\}_{q \times q}$ for $g, h \in \{1, \ldots, q\}$. Given a choice of $\boldsymbol{\lambda} \equiv (\lambda/\theta_1, \ldots, \lambda/\theta_s)$, the optimal function coefficients are given by

$$(5) \qquad \begin{pmatrix} \hat{\mathbf{d}} \\ \hat{\mathbf{c}} \end{pmatrix} = \begin{pmatrix} \mathbf{K}'\mathbf{K} & \mathbf{K}'\mathbf{J}_{\boldsymbol{\theta}} \\ \mathbf{J}_{\boldsymbol{\theta}}'\mathbf{K} & \mathbf{J}_{\boldsymbol{\theta}}'\mathbf{J}_{\boldsymbol{\theta}} + \lambda n \mathbf{Q}_{\boldsymbol{\theta}} \end{pmatrix}^{\dagger} \begin{pmatrix} \mathbf{K}' \\ \mathbf{J}_{\boldsymbol{\theta}}' \end{pmatrix} \mathbf{y}$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse.

The fitted values are given by $\hat{\mathbf{y}} = \mathbf{K}\hat{\mathbf{d}} + \mathbf{J}_{\boldsymbol{\theta}}\hat{\mathbf{c}} = \mathbf{S}_{\boldsymbol{\lambda}}\mathbf{y}$, where

$$(6) \qquad \mathbf{S}_{\boldsymbol{\lambda}} = \begin{pmatrix} \mathbf{K} & \mathbf{J}_{\boldsymbol{\theta}} \end{pmatrix} \begin{pmatrix} \mathbf{K}'\mathbf{K} & \mathbf{K}'\mathbf{J}_{\boldsymbol{\theta}} \\ \mathbf{J}_{\boldsymbol{\theta}}'\mathbf{K} & \mathbf{J}_{\boldsymbol{\theta}}'\mathbf{J}_{\boldsymbol{\theta}} + \lambda n \mathbf{Q}_{\boldsymbol{\theta}} \end{pmatrix}^{\dagger} \begin{pmatrix} \mathbf{K}' \\ \mathbf{J}_{\boldsymbol{\theta}}' \end{pmatrix}$$

is the smoothing matrix, which depends on $\boldsymbol{\lambda}$. The smoothing parameters are typically selected by minimizing Craven and Wahba's (1979) generalized cross-validation (GCV) score:

$$(7) \qquad \mathrm{GCV}(\boldsymbol{\lambda}) = \{n\|(\mathbf{I}_n - \mathbf{S}_{\boldsymbol{\lambda}})\mathbf{y}\|^2\}/\{[n - \mathrm{tr}(\mathbf{S}_{\boldsymbol{\lambda}})]^2\}.$$

The estimates $\hat{\lambda}$ and $\hat{\boldsymbol{\theta}}$ that minimize the GCV score have desirable properties [see 2, 3, 4, 12].

3. ROUNDING PARAMETERS

3.1 Overview

When fitting a nonparametric regression model to ultra large samples, we propose including user-tunable rounding parameters in the model [see 6, for preliminary work]. Assuming that all (continuous) predictors have been transformed to the interval [0,1], the rounding parameters $r_j \in (0, 1]$ are used to create locally-smoothed versions of the (continuous) predictor variables, such as

$$(8) \qquad z_{ij} = \mathrm{rd}(x_{ij}/r_j) r_j$$

for $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, p\}$, where the rounding function $\mathrm{rd}(\cdot)$ rounds the input value to the nearest integer. Note that the z_{ij} scores are formed simply by rounding the original x_{ij} scores to the precision defined by the rounding parameter for the j-th predictor variable, e.g., if $r_j = .02$, then each x_{ij} value is rounded to the nearest .02 to form z_{ij}.

Let $\mathbf{z}_i \equiv (z_{i1}, \ldots, z_{ip})'$ with z_{ij} defined according to Equation (8), and let $\{\breve{\mathbf{z}}_h\}_{h=1}^{q}$ denote the rounded knots; then, the penalized least-squares function in Equation (4) can be approximated as $(1/n)\|\mathbf{y} - \mathbf{K}_\star \mathbf{d}_\star - \mathbf{J}_{\boldsymbol{\theta}}^\star \mathbf{c}_\star\|^2 + \lambda \mathbf{c}_\star' \mathbf{Q}_{\boldsymbol{\theta}}^\star \mathbf{c}_\star$, where $\mathbf{K}_\star$, $\mathbf{J}_{\boldsymbol{\theta}}^\star$, and $\mathbf{Q}_{\boldsymbol{\theta}}^\star$ are defined according to Equation (4) with $\mathbf{z}_i$ replacing $\mathbf{x}_i$. Similarly, the optimal basis function coefficients corresponding to the rounded data (i.e., $\hat{\mathbf{d}}_\star$ and $\hat{\mathbf{c}}_\star$) can be defined according to Equation (5) with with $\mathbf{z}_i$ replacing $\mathbf{x}_i$. Finally, smoothing matrix corresponding to these coefficients (denoted by $\mathbf{S}_{\boldsymbol{\lambda}, r}$) can be defined according to Equation (6) with with $\mathbf{z}_i$ replacing $\mathbf{x}_i$.

One could calculate the fitted values using $\mathbf{S}_{\boldsymbol{\lambda}, r} \mathbf{y}$ (and this is what we recommend for the smoothing parameter estimation), however this could introduce a small bias to each predicted score. So, when interpreting specific $\hat{y}_i$ scores, we recommend using the $\hat{\mathbf{d}}_\star$ and $\hat{\mathbf{c}}_\star$ coefficients and basis function matrices with unrounded predictor variable scores

$$(9) \qquad \hat{\mathbf{y}}_\star = \mathbf{K}\hat{\mathbf{d}}_\star + \mathbf{J}_{\boldsymbol{\theta}}\hat{\mathbf{c}}_\star$$

where $\mathbf{K}$ and $\mathbf{J}_{\boldsymbol{\theta}}$ are defined according to Equation (4).

3.2 Computational benefits

Let $\{\tilde{\mathbf{z}}_t\}_{t=1}^u$ denote the set of unique observed $\mathbf{z}_i$ vectors with $u \geq q$, and note that u has an upper-bound that is determined by the rounding parameters and the predictor variables. For example, suppose that $\tilde{\mathbf{z}}_t \equiv (\tilde{z}_{t1}, \tilde{z}_{t2})$ with $\tilde{z}_{t1} \in [0,1]$ and $\tilde{z}_{t2} \in \{1, \ldots, f\}$; then, defining $r_1 = .01$, it is evident that $u \leq 101f$, given that z_{ij} can have a maximum of 101 unique values for the first predictor, and maximum of f unique values for the second predictor. As a second example, suppose that $\tilde{\mathbf{z}}_t \equiv (\tilde{z}_{t1}, \tilde{z}_{t2})$ with $\tilde{z}_{t1}, \tilde{z}_{t2} \in [0,1]$; then, defining $r_1 = r_2 = .01$, it is evident that $u \leq 101^2$, given that z_{ij} can have a maximum of 101 unique values for each predictor. Similar reasoning can be used to place an upper bound on u for different combinations of rounding parameters and predictor variable types.

Note that the inner-portion of $\mathbf{S}_{\lambda,r}$ can be written as

$$(10) \quad \begin{pmatrix} \mathbf{K}'_\star \mathbf{K}_\star & \mathbf{K}'_\star \mathbf{J}^\star_{\boldsymbol{\theta}} \\ (\mathbf{J}^\star_{\boldsymbol{\theta}})' \mathbf{K}_\star & (\mathbf{J}^\star_{\boldsymbol{\theta}})' \mathbf{J}^\star_{\boldsymbol{\theta}} + \lambda n \mathbf{Q}^\star_{\boldsymbol{\theta}} \end{pmatrix}^\dagger = \begin{pmatrix} \tilde{\mathbf{K}}'_\star \mathbf{W} \tilde{\mathbf{K}}_\star & \tilde{\mathbf{K}}'_\star \mathbf{W} \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} \\ (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \mathbf{W} \tilde{\mathbf{K}}_\star & (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \mathbf{W} \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} + \lambda n \mathbf{Q}^\star_{\boldsymbol{\theta}} \end{pmatrix}^\dagger$$

where $\tilde{\mathbf{K}}_\star \equiv \{\phi_v(\tilde{\mathbf{z}}_t)\}_{u \times m}$ for $t \in \{1, \ldots, u\}$ and $v \in \{1, \ldots, m\}$, $\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} = \sum_{k=1}^s \theta_k \tilde{\mathbf{J}}^\star_k$ where $\tilde{\mathbf{J}}^\star_k \equiv \{\rho_k^*(\tilde{\mathbf{z}}_t, \breve{\mathbf{z}}_h)\}_{u \times q}$ for $t \in \{1, \ldots, u\}$ and $h \in \{1, \ldots, q\}$, and $\mathbf{W} \equiv \mathrm{diag}(w_1, \ldots, w_u)$ with w_t denoting the number of $\mathbf{z}_i$ that are equal to $\tilde{\mathbf{z}}_t$ (for $t \in \{1, \ldots, u\}$). Next, define $\tilde{\mathbf{X}} = (\tilde{\mathbf{K}}_\star, \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})$ and define the reduced smoothing matrix $\tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}}$, such as

$$(11) \quad \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} = \tilde{\mathbf{X}} \begin{pmatrix} \tilde{\mathbf{K}}'_\star \mathbf{W} \tilde{\mathbf{K}}_\star & \tilde{\mathbf{K}}'_\star \mathbf{W} \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} \\ (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \mathbf{W} \tilde{\mathbf{K}}_\star & (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \mathbf{W} \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} + \lambda n \mathbf{Q}^\star_{\boldsymbol{\theta}} \end{pmatrix}^\dagger \tilde{\mathbf{X}}'.$$

Note that $\tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}}$ is a $u \times u$ matrix, and note that $u < n$ if there are replicate predictor vectors after the rounding (which is guaranteed if n is larger than u's upper bound).

Next, suppose that the $(y_i, \mathbf{z}_i)$ scores are ordered such that observations $1, \ldots, w_1$ have predictor scores $\tilde{\mathbf{z}}_1$, observations $w_1 + 1, \ldots, w_1 + w_2$ have predictor scores $\tilde{\mathbf{z}}_2$, and so on. Then $\mathbf{S}_{\lambda,r}$ can be written in terms of $\tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}}$, such as

$$(12) \quad \mathbf{S}_{\boldsymbol{\lambda},r} = \begin{pmatrix} (\mathbf{e}'_1 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_1) \mathbf{1}_{w_1} \mathbf{1}'_{w_1} & \cdots & (\mathbf{e}'_1 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_u) \mathbf{1}_{w_1} \mathbf{1}'_{w_u} \\ (\mathbf{e}'_2 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_1) \mathbf{1}_{w_2} \mathbf{1}'_{w_1} & \cdots & (\mathbf{e}'_2 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_u) \mathbf{1}_{w_2} \mathbf{1}'_{w_u} \\ \vdots & \ddots & \vdots \\ (\mathbf{e}'_u \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_1) \mathbf{1}_{w_u} \mathbf{1}'_{w_1} & \cdots & (\mathbf{e}'_u \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_u) \mathbf{1}_{w_u} \mathbf{1}'_{w_u} \end{pmatrix}$$

where $\mathbf{e}_t$ denotes a $u \times 1$ vector with a one in the t-th position and zeros elsewhere, and $\mathbf{1}_{w_t}$ denotes a $w_t \times 1$ vector of ones (for $t \in \{1, \ldots, u\}$). Furthermore, note that the fitted values corresponding to $\mathbf{S}_{\lambda,r}$ can be written as

$$(13) \quad \mathbf{S}_{\boldsymbol{\lambda},r} \mathbf{y} = \begin{pmatrix} (\mathbf{e}'_1 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_1) \mathbf{1}_{w_1} & \cdots & (\mathbf{e}'_1 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_u) \mathbf{1}_{w_1} \\ (\mathbf{e}'_2 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_1) \mathbf{1}_{w_2} & \cdots & (\mathbf{e}'_2 \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_u) \mathbf{1}_{w_2} \\ \vdots & \ddots & \vdots \\ (\mathbf{e}'_u \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_1) \mathbf{1}_{w_u} & \cdots & (\mathbf{e}'_u \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{e}_u) \mathbf{1}_{w_u} \end{pmatrix} \tilde{\mathbf{y}}$$

where $\tilde{\mathbf{y}} \equiv \{\tilde{y}_t\}_{u \times 1}$ with $\tilde{y}_t = \sum_{\mathcal{I}_t} y_i$ and $\mathcal{I}_t \subset \{1, \ldots, n\}$ denoting the set of indices such that $\mathbf{z}_i$ is equal to $\tilde{\mathbf{z}}_t$.

Now, let $\hat{y}_t^\star = \mathbf{e}'_t \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \tilde{\mathbf{y}}$ denote the fitted value corresponding to $\tilde{\mathbf{z}}_t$ (for $t \in \{1, \ldots, u\}$), and note that the numerator of the GCV score in Equation (7) can be written as

$$(14)$$
$$n \sum_{t=1}^u \sum_{\mathcal{I}_t} (y_i - \hat{y}_t^\star)^2 = n \sum_{i=1}^n y_i^2 - 2n \sum_{t=1}^u \tilde{y}_t \hat{y}_t^\star + n \sum_{t=1}^u w_t (\hat{y}_t^\star)^2$$
$$= n \left[\|\mathbf{y}\|^2 - 2\tilde{\mathbf{y}}' \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \tilde{\mathbf{y}} + \tilde{\mathbf{y}}' \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \mathbf{W} \tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}} \tilde{\mathbf{y}} \right]$$

In addition, note that the denominator of the GCV score can be written as $[n - \mathrm{tr}(\mathbf{S}_{\boldsymbol{\lambda},r})]^2 = [n - \mathrm{tr}(\mathbf{W}\tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}})]^2$ using the relation in Equation (12).

The above formulas imply that, after initializing $\tilde{\mathbf{y}}$, $\|\mathbf{y}\|^2$, and $\mathbf{W}$, it is only necessary to calculate the reduced smoothing matrix $\tilde{\mathbf{S}}^\star_{\boldsymbol{\lambda}}$ to evaluate the GCV score. Furthermore, note that the optimal function coefficients can be estimated from the reduced smoothing matrix using

$$(15)$$
$$\begin{pmatrix} \hat{\mathbf{d}}_\star \\ \hat{\mathbf{c}}_\star \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{K}}'_\star \mathbf{W} \tilde{\mathbf{K}}_\star & \tilde{\mathbf{K}}'_\star \mathbf{W} \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} \\ (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \mathbf{W} \tilde{\mathbf{K}}_\star & (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \mathbf{W} \tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}} + \lambda n \mathbf{Q}^\star_{\boldsymbol{\theta}} \end{pmatrix}^\dagger \begin{pmatrix} \tilde{\mathbf{K}}'_\star \\ (\tilde{\mathbf{J}}^\star_{\boldsymbol{\theta}})' \end{pmatrix} \tilde{\mathbf{y}}$$

which implies that it is never necessary to construct the full $n \times n$ smoothing matrix to estimate η when using the rounding parameters.

3.3 Choosing rounding parameters

In many situations, a rounding parameter can be determined by the measurement precision of the predictor variable. For example, suppose we have one predictor x_i recorded with the precision of two decimals on the interval $[0,1]$, i.e., $x_i \in \{0, 0.01, 0.02, \ldots, 0.99, 1\}$ for $i \in \{1, \ldots, n\}$. In this case, setting $r = 0.01$ will produce the exact same solution as using the unrounded predictors (i.e., $z_i = x_i \forall i$) and can immensely reduce the computational burden. Note that $u \leq 101$ even if n is very large, and it is only necessary to evaluate the functions $\{\phi_v\}_{v=1}^m$ and ρ_c for the $u \ll n$ unique predictor scores to estimate η.

Now, for large n, note that a cubic smoothing spline is approximately a weighted moving average smoother [see 21, Section 3]. In particular, let $s_{i_1 i_2 (\lambda)}$ denote the entry in the i_1-th row and i_2-th column of $\mathbf{S}_\lambda$, and note that $s_{i_1 i_2 (\lambda)}$ asymptotically depends on a kernel function whose influence decreases exponentially as $|x_{i_1} - x_{i_2}|$ increases [see 21, equations 3.1–3.4]. Also, note that the rounding parameter proposed in this paper widens the peak of the kernel (see Figure 1). For relatively smooth functions (e.g., $\lambda \geq 10^{-3}$), the shape of the asymptotic kernel function is stable for $r \leq 0.05$; however, for more jagged functions (e.g., $\lambda \leq 10^{-7}$), the rounding parameter will need to be set smaller (e.g., $r = 0.01$) for the rounded kernel function to resemble the true asymptotic kernel (see Figure 1).
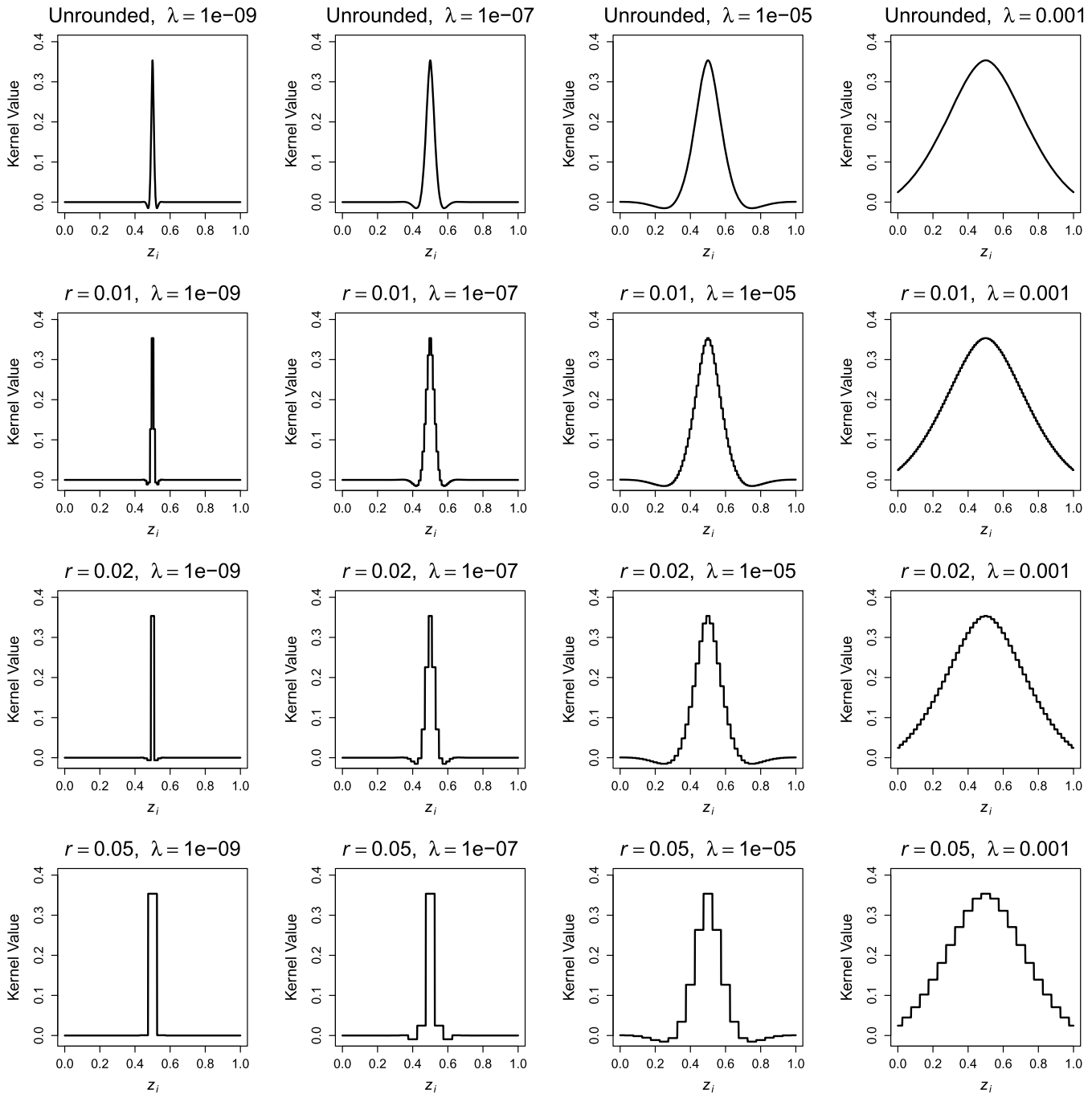
Figure 1. Asymptotic cubic spline kernel function for $z_i \in [0, 1]$ and $\breve{z} = 0.5$.

4. QUALITY OF ROUNDED SOLUTION

4.1 A Taylor heuristic

Note that the rounded predictor z_{ij} can be written as

$$(16) \qquad z_{ij} = x_{ij} + r_j v_{ij}$$

where $v_{ij} = (z_{ij} - x_{ij})/r_j$ by definition and $|z_{ij} - x_{ij}| \leq r_j/2$ so that $|v_{ij}| \leq 1/2$. This implies $\mathbf{z}_i = \mathbf{x}_i + \mathbf{R}\mathbf{v}_i$ where $\mathbf{v}_i = (v_{i1}, \ldots, v_{ip})'$ and $\mathbf{R} = \mathrm{diag}(r_1, \ldots, r_p)$. Consider the linear approximation of $\eta(\mathbf{z}_i)$ at the point $\mathbf{x}_i$

$$\eta(\mathbf{z}_i) = \eta(\mathbf{x}_i) + [\nabla \eta(\mathbf{x}_i)]' \mathbf{R}\mathbf{v}_i + o(\|\mathbf{R}\mathbf{v}_i\|)$$

where $\nabla \eta$ denotes the gradient of η. If the gradient of η were known, we could approximate the rounding error using

$$n^{-1} \sum_{i=1}^{n} [\eta(\mathbf{x}_i) - \eta(\mathbf{z}_i)]^2 \approx n^{-1} \sum_{i=1}^{n} \{[\nabla \eta(\mathbf{x}_i)]' \mathbf{R}\mathbf{v}_i\}^2$$

Linear Kernel

Cubic Kernel

Quintic Kernel

Linear Derivative

Cubic Derivative

Quintic Derivative
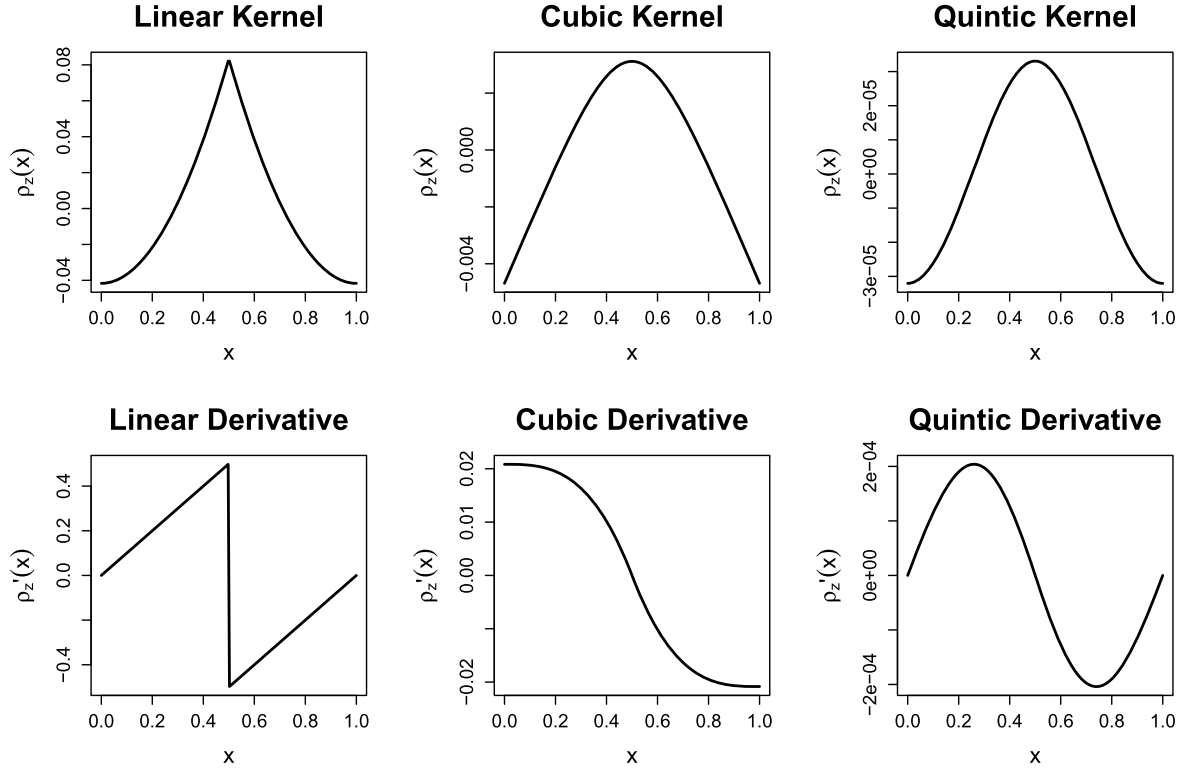
Figure 2. Top: contrast reproducing kernel $\rho_z(x)$ for linear spline ($m = 1$), cubic spline ($m = 2$), and quintic spline ($m = 3$) with $z = 0.5$ as the knot. Bottom: contrast reproducing kernel derivative $\rho_z'(x)$ for m-th order polynomial splines.

(17)
$$\leq n^{-1} \sum_{i=1}^{n} \|\nabla\eta(\mathbf{x}_i)\|^2 \|\mathbf{R}\mathbf{v}_i\|^2$$

$$\leq (4n)^{-1} \sum_{i=1}^{n} \|\nabla\eta(\mathbf{x}_i)\|^2 \|\mathbf{r}\|^2$$

where $\mathbf{r} = (r_1, \ldots, r_p)'$; note that the last line is due to the fact that $|v_{ij}| \leq 1/2$.

For example, using an m-th order polynomial smoothing spline with $x_i \in [0, 1]$ [see 2, 3] we have

$$\eta_\lambda(x) = \sum_{v=0}^{m-1} d_v k_v(x) + \sum_{h=1}^{q} c_h \rho_{\breve{x}_h}(x)$$

where $k_v(\cdot)$ are scaled Bernoulli polynomials, $\{\breve{x}_h\}_{h=1}^{q} \subset \{x_i\}_{i=1}^{n}$ are the selected knots, and

$$\rho_{\breve{x}_h}(x) = k_m(x) k_m(\breve{x}_h) + (-1)^{m-1} k_{2m}(|x - \breve{x}_h|)$$

is the reproducing kernel of the contrast space. Using the properties of Bernoulli polynomials we have

$$\eta_\lambda'(x) = \frac{\partial \eta_\lambda(x)}{\partial x} = \sum_{v=1}^{m-1} d_v k_{v-1}(x) + \sum_{h=1}^{q} c_h \rho_{\breve{x}_h}'(x)$$

where

$$\rho_{\breve{x}_h}'(x) = k_{m-1}(x) k_m(\breve{x}_h) + (-1)^{m-1} s_h k_{2m-1}(x - \breve{x}_h)$$

with $s_h = 1$ if $x \geq \breve{x}_h$ and $s_h = -1$ otherwise [see 2, 3].

Consequently, for polynomial splines we can approximate the rounding error using

$$n^{-1} \sum_{i=1}^{n} [\eta(x_i) - \eta(z_i)]^2 \approx n^{-1} \sum_{i=1}^{n} (r v_i)^2 [\eta_\lambda'(x_i)]^2$$

$$\leq r^2 (4n)^{-1} \|\mathbf{X}\mathbf{b}\|^2$$

where $\mathbf{X} = [\tilde{\mathbf{K}}, \tilde{\mathbf{J}}]$ with $\tilde{\mathbf{K}} = \{k_v(x_i)\}_{n \times m-1}$ for $v \in \{0, \ldots, m-2\}$ and $\tilde{\mathbf{J}} = \{\rho_{\breve{x}_h}'(x_i)\}_{n \times q}$ for $h \in \{1, \ldots, q\}$, and $\mathbf{b} = (d_1, \ldots, d_{m-1}, c_1, \ldots, c_q)'$. Note that the contrast space reproducing kernel $\rho_{\breve{x}_h}(x)$ is rather smooth for the classic cubic smoothing spline, and the magnitude of the derivatives are rather small (see Figure 2). This implies that setting $r \in \{0.01, 0.02, 0.05\}$ will not introduce much rounding error to the contrast kernel evaluation when using cubic smoothing splines on $x_i \in [0, 1]$.

The rounding error depends on the norm $\|\mathbf{X}\mathbf{b}\|$, so the relative impact of a particular choice of rounding parameters will depend on the (unknown) function coefficients $\mathbf{b}$. For practical use, we can approximate the rounding error relative to the norm of the coefficients, such as

$$\frac{1}{n\|\mathbf{b}\|^2} \sum_{i=1}^{n} [\eta(x_i) - \eta(z_i)]^2 \approx \frac{1}{n\|\mathbf{b}\|^2} \sum_{i=1}^{n} (r v_i)^2 [\eta_\lambda'(x_i)]^2$$

$$\leq r^2 (4n)^{-1} \lambda_1^*$$

where λ_1^* is the largest eigenvalue of $\mathbf{X}'\mathbf{X}$; note that we have $\|\mathbf{Xb}\|^2 \leq \|\mathbf{X}\|^2\|\mathbf{b}\|^2$ and $\|\mathbf{X}\|^2 = \lambda_1^*$ by definition. For practical computation, it is possible to estimate λ_1^*/n by taking a random sample of $\tilde{n} \ll n$ observations, and then approximate the relative rounding error as $r^2(\tilde{n}4)^{-1}\hat{\lambda}_1^*$. Clearly this sort of approach can be extended to assess the relative rounding error for tensor product smoothing splines, but the gradient formulas become a bit more complicated.

4.2 Finite sample performance

To quantify the finite-sample error introduced by rounding, define the loss function

$$(18) \qquad \begin{aligned} L(r) &= \frac{1}{n}\sum_{i=1}^n \left(\hat{\eta}_\lambda(\mathbf{x}_i) - \hat{\eta}_{\lambda,r}(\mathbf{z}_i)\right)^2 \\ &= n^{-1}\|(\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r})\mathbf{y}\|^2 \end{aligned}$$

where $\mathbf{S_\lambda}$ and $\mathbf{S}_{\lambda,r}$ are the smoothing matrices corresponding to the unrounded and rounded predictors (i.e., $\mathbf{x}_i$ and $\mathbf{z}_i$, respectively). Denote the risk function as

$$(19) \qquad \begin{aligned} R(r) &= E[L(r)] \\ &= n^{-1}\|(\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r})\boldsymbol{\eta}\|^2 + n^{-1}\sigma^2\mathrm{tr}\{(\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r})^2\} \end{aligned}$$

where $\boldsymbol{\eta} = \{\eta(\mathbf{x}_i)\}_{n\times 1}$ contains the realizations of the (unknown) true function η. Note that the first term of $R(r)$ corresponds to the (squared) bias difference between $\hat{\eta}_\lambda$ and $\hat{\eta}_{\lambda,r}$, and the second term is related to (but not equal to) the variance difference. Also note that we can write

$$(20) \qquad \begin{aligned} R(r) &\leq n^{-1}\|\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r}\|^2\|\boldsymbol{\eta}\|^2 + n^{-1}\sigma^2\sum_{i=1}^n \lambda_{i,r} \\ &\leq \lambda_{1,r}\left(n^{-1}\|\boldsymbol{\eta}\|^2 + \sigma^2\right) \end{aligned}$$

where $\lambda_{1,r} \geq \cdots \geq \lambda_{n,r}$ are the eigenvalues of $(\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r})^2$.

The risk $R(r)$ depends on the squared norm of the unknown function η, so the practical relevance of a particular value of $R(r)$, e.g., $R(r) = 0.1$, differs depending on the situation, i.e., unknown true function. To overcome this practical issue, we can examine the risk relative to the squared norm of the unknown function, such as

$$(21) \qquad \begin{aligned} U(r) &= R(r)\|\boldsymbol{\eta}\|^{-2} \\ &\leq n^{-1}\lambda_{1,r}\left(1 + n\sigma^2\|\boldsymbol{\eta}\|^{-2}\right) \end{aligned}$$

where $n\sigma^2\|\boldsymbol{\eta}\|^{-2} = \sigma^2/(\|\boldsymbol{\eta}\|^2/n)$ relates to the noise-to-signal ratio, i.e., inverse of signal-to-noise ratio (SNR). Furthermore, for a fixed SNR and a large enough n, the second term in the upper-bound of the relative risk is negligible, and we have that $U(r) \lesssim n^{-1}\lambda_{1,r}$. Consequently, it is only necessary to know the largest eigenvalue of $\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r}$ to understand the expected performance of a given set of rounding parameters for a large sample size n.

In practice, calculating $\mathbf{S_\lambda} - \mathbf{S}_{\lambda,r}$ and $\lambda_{1,r}$ for various values of r is a computational challenge for large n. For practical computation, we recommend examining $R(r)$ and/or $U(r)$ using a random sample of $\tilde{n} \ll n$ observations. Using this approach, the unknown parameters (i.e., $\boldsymbol{\eta}$ and σ^2) can be estimated using the results of the unrounded solution. For example, the SNR can be estimated as $(\|\hat{\boldsymbol{\eta}}\|^2/\tilde{n})/\hat{\sigma}^2$ where $\hat{\boldsymbol{\eta}}$ and $\hat{\sigma}^2$ are the estimated function and error variance using the $\tilde{n}$ observations with unrounded predictors. Or, if the approximate SNR is known, Equation (21) can be used to place an upper-bound on the relative risk $U(r)$.

We demonstrate this approach in Figures 3–4, which plot functions with various degrees of smoothness (Figure 3) and the median estimated rounding risk $\hat{R}(r)$ across five samples of $\tilde{n} = 500$ observations (Figure 4). Note that Figure 4 illustrates that the expected difference between the unrounded and rounded solutions increases as the error variance increases. Furthermore, note that Figure 4 affirms that for $x \in [0,1]$ setting $r = 0.01$ can be expected to introduce minimal rounding error for a variety of functions and SNRs. Finally, Figure 4 reveals that setting $r \in \{0.01, 0.02, 0.05\}$ will not introduce much rounding error whenever the underlying function η is relatively smooth. For example, for the functions η_{A1} and η_{B1}, we should expect a negligible difference between the unrounded and rounded solutions using $r = 0.05$ for a variety of different SNRs.

4.3 Asymptotic bias and variance

To establish the asymptotic properties of the proposed estimate, we employ an equivalent kernel approach developed in [15]. The key idea is that a smoothing spline estimate can be written as kernel estimate

$$(22) \qquad \hat{\eta}_\lambda(x) = \frac{1}{n}\sum_{i=1}^n w(x_i, x)y_i$$

where the kernel function $w(x_i, x)$ can be well approximated by a Green's function. Then the asymptotic properties of $\hat{\eta}_\lambda$ can be established via the analytical properties of the Green's function.

Following [15], we establish the asymptotic properties of our rounding estimate for the one dimensional case. In addition, we assume that we use a full basis where all distinct rounded data are used as knots, i.e., $q = u$. Then our estimate $\hat{\eta}_{\lambda,r}$ is the minimizer of

$$(23) \qquad (1/n)\sum_{i=1}^n (y_i - \eta(z_i))^2 + \lambda\int_0^1 (\eta^{(m)})^2 dx.$$

Let $F_{n,r}$ denote the empirical distribution function for the rounded predictor z_i, $i = 1, \ldots, n$, let F be the limiting distribution of the original predictor x with a continuous and strictly positive density function f on $[0,1]$ and let

$$D_{n,r} = \sup_{x\in[0,1]} |F_{n,r} - F|,$$

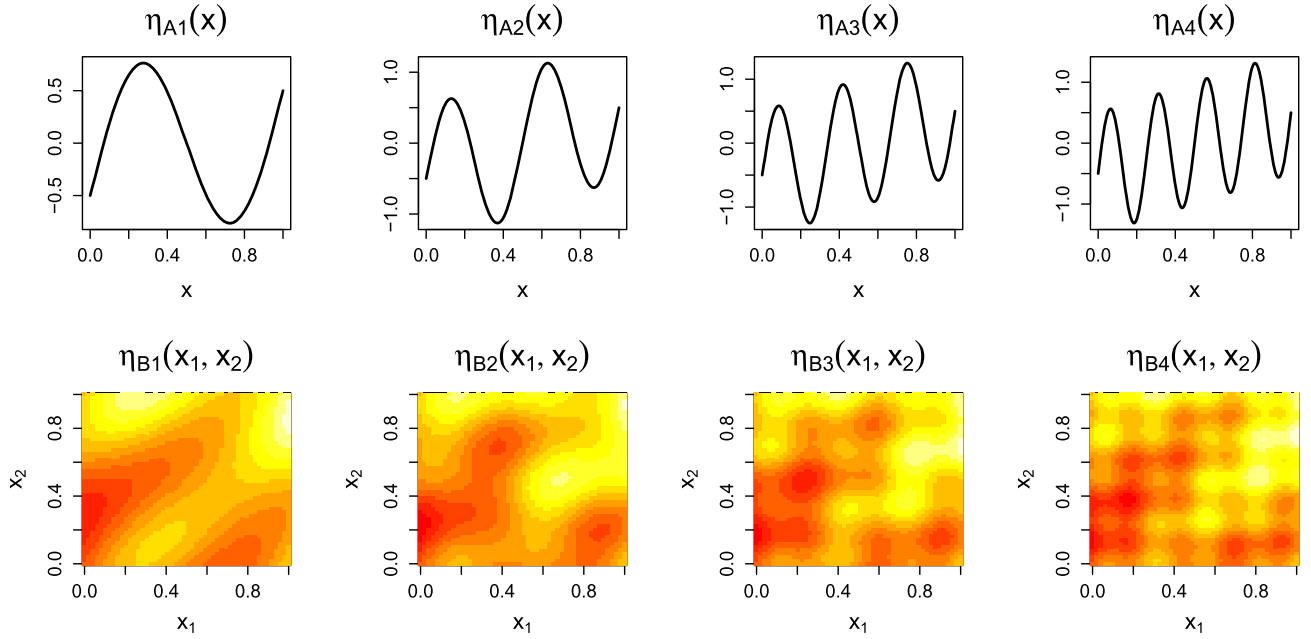and $\rho = \lambda^{1/2m}$. Then we have the following theorem.

Figure 3. Functions with various degrees of smoothness. $\eta_{Ak}(x) = x - 0.5 + \sin(2k\pi x)$ for $x \in [0,1]$ and
$\eta_{Bk}(x_1, x_2) = x_1 + x_2 - 1 + [\sin(2k\pi x_1) + \cos(2k\pi x_2) + 2\sin(2\pi(x_1 - x_2))]/4$ for $x_1, x_2 \in [0,1]$.
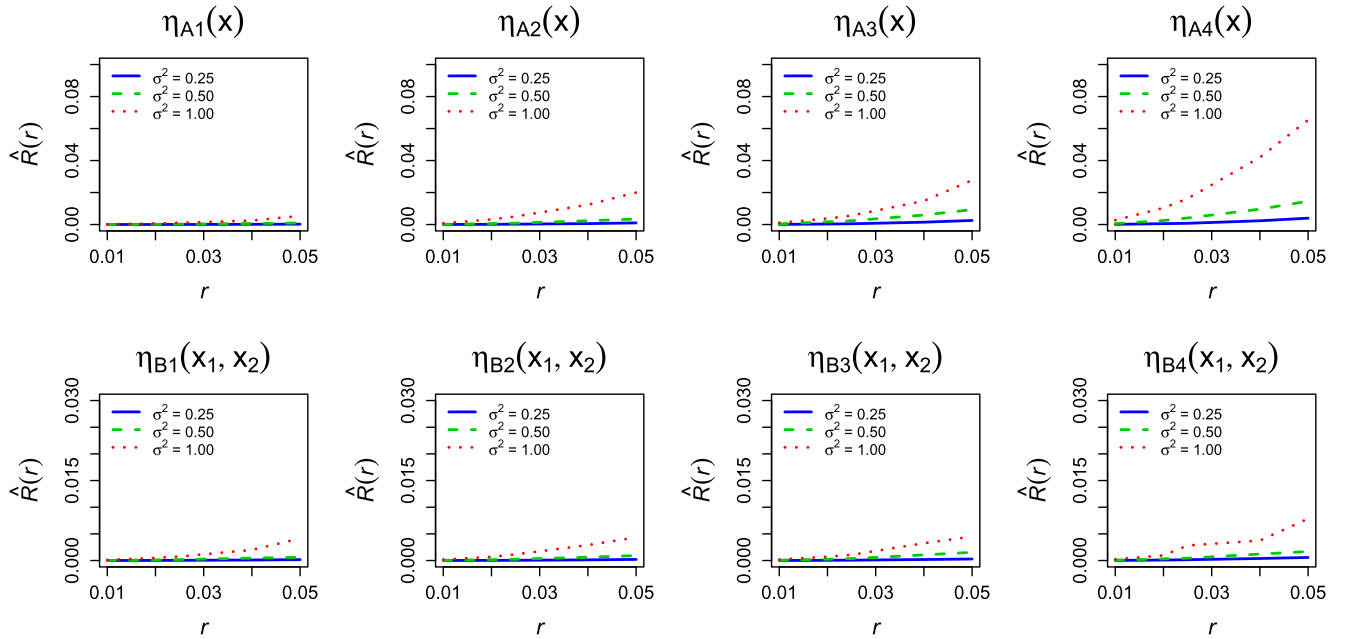

Figure 4. Median estimated risk $\hat{R}(r) = \tilde{n}^{-1}\|(\mathbf{S_\lambda} - \mathbf{S_{\lambda,r}})\hat{\boldsymbol{\eta}}\|^2 + \tilde{n}^{-1}\hat{\sigma}^2 \text{tr}\{(\mathbf{S_\lambda} - \mathbf{S_{\lambda,r}})^2\}$ for various functions, rounding
parameters, and error variances using five random samples of $\tilde{n} = 500$ observations.

Theorem 4.1. *Assume that $\hat{\eta}_{\lambda,r}$ is a smoothing spline estimate of (23) with $m = 1$ and z_i are not equally spaced. Suppose that $\eta \in C_2[0,1]$ and satisfies the Hölder condition $|\eta^{(2)}(x) - \eta^{(2)}(x')| \le M|x - x'|^\beta$ for some $\beta > 0$ and some $M < \infty$. Assume that f has a uniformly continuous derivative and $D_{n,r} \to 0$ as $n \to \infty$. Choose $0 < \Delta < 1$ and let $\lambda_n \to 0$ and $\Lambda_n \to 0$ as $n \to \infty$. Then*

$$E[\hat{\eta}_{\lambda,r}(x)] - \eta(x) = -\frac{\lambda}{f(x)}\eta^{(2)}(x) + o(\lambda) + O\left(\frac{D_{n,r}}{\rho}\right),$$

$$Var[\hat{\eta}_{\lambda,r}(x)] = \frac{\sigma^2}{8nf(x)}\left(\frac{f(x)}{\lambda}\right)^{1/2} + \sigma^2 O\left(\frac{D_{n,r}}{\rho}\right),$$

uniformly for $\lambda \in [\lambda_n, \Lambda_n]$ and $x \in [\Delta, 1 - \Delta]$ as $n \to \infty$.

The theorem is a direct result of Theorem 2.2 of [15]. For $m > 1$, a slightly more complicated version of our theorem can be shown using Theorem 2 of [23].

The theorem states that both the bias and variance of our estimate $\hat{\eta}_{\lambda,r}$ depend on $D_{n,r}$, which is required to be sufficiently small relative to ρ as $n \to \infty$. Consequently, the theorem reveals that the rounding parameter r will have to be set smaller when

(a) the true function η is rougher,
(b) the spline order m is larger,
(c) the predictor distribution f is rougher,
(d) the sample size n is larger.

These conclusions derive directly from the requirement that $D_{n,r}$ be sufficiently small relative to ρ as $n \to \infty$.

5. SIMULATION STUDY

5.1 Design and analyses

We conducted a simulation study to demonstrate the benefits of the rounding parameters. As a part of the simulation, we manipulated two conditions: (a) the function smoothness (8 levels: see Figure 3), and (b) the number of observations (3 levels: $n = 1000k$ for $k \in \{100, 200, 500\}$). Note that the functions are defined such that $J(\eta_{Aj}) < J(\eta_{Ak})$ and $J(\eta_{Bj}) < J(\eta_{Bk})$ for $j < k \in \{1, 2, 3, 4\}$, so the function smoothness is systematically manipulated. We generated y_i by (a) independently sampling the predictor(s) from a uniform distribution, (b) independently sampling e_i from a standard normal distribution, and (c) defining the observed response as $y_i = \eta(\mathbf{x}_i) + e_i$ for $i \in \{1, \ldots, n\}$.

Then, we fit a nonparametric regression model using six different methods: Method 1 is an SSANOVA using unrounded data [see 10], Method 2 is an SSANOVA with $r = .01$, Method 3 is an SSANOVA with $r = .02$, Method 4 is an SSANOVA with $r = .05$, Method 5 is standard GAM implemented through Wood's (2015) `gam.R` function, and Method 6 is batch-processed GAM implemented through Wood's (2015) `bam.R` function. Methods 1–4 are implemented through Helwig's (2015a) `bigspline.R` function (for η_{Ak}) and `bigssa.R` function (for η_{Bk}).

For the η_{Ak} functions we used $q = 21$ knots to fit the model, and for η_{Bk} functions we used $q = 100$ knots. For Methods 1–4, we used a bin-sampling approach to select knots spread throughout the covariate domain [9]; for Methods 5 and 6, we used the default `gam.R` and `bam.R` knot-selection algorithm [see 25]. For each method, we used cubic splines and selected the smoothing parameters that minimized the GCV score. Given the optimal smoothing parameters, we calculated the fitted values, and then defined the true mean-squared-error (MSE) as $(1/n) \sum_{i=1}^{n} (\eta(\mathbf{x}_i) - \hat{y}_i)^2$. Finally, we used 100 replications of the above procedure within each cell of the simulation design.

5.2 Results

The true MSE for each combination of simulation conditions is plotted in Figure 5. First, note that for each method, the true MSE decreased as n increased, which was expected. Next, note that all of the methods recovered η quite well (i.e., all MSEs smaller than 0.01). Comparing Methods 1–4, it is evident that setting $r \in \{.01, .02\}$ introduced minimal bias to the resulting solution. In contrast, setting $r = .05$ produced a more noticeable bias, particularly when analyzing the more jagged η_{Ak} and η_{Bk} functions, i.e., those with larger k. However, the bias introduced with $r = .05$ was small relative to the norm of η, so there is little practical difference between the solutions with $r \in \{.01, .02, .05\}$. Examining the true MSEs of Methods 5 and 6, it is apparent that the standard GAM performed almost identical to the batch-processed GAM throughout the simulation.

Comparing the true MSEs of Methods 1–4 to those of Methods 5 and 6, it apparent that the SSANOVAs performed similar to the GAMs in every simulation condition. In the one-dimensional case (η_{Ak} functions), the GAMs have slightly smaller true MSEs for $k \in \{3, 4\}$, but the difference is trivial compared to the norm of the η_{Ak} functions. In the two-dimensional case (η_{Bk} functions), the SSANOVAs have slightly smaller true MSEs for $k \in \{3, 4\}$. Differences between the SSANOVA and GAM solutions are most pronounced when analyzing the η_{B4} function; in this case, the median true MSE of the GAM solutions is over 10 times larger than the corresponding median of the SSANOVA solutions with $r \in \{NA, 0.01, 0.02\}$. However, the difference is still quite small compared to the norm of the η_{B4} function.

The median analysis runtimes (in seconds) for each simulation condition are displayed in Tables 1 and 2. First, note that for each method, the runtime increased as n increased, which was expected. Next, note that the runtimes for Methods 1, 5, and 6 were substantially larger than the corresponding runtimes of Methods 2–4. When analyzing the η_{Ak} functions, the median runtimes for Methods 2–4 were less than one-tenth of a second for all examined n, and were anywhere from 40–60 times faster than the median runtimes for Methods 5 and 6. When analyzing the η_{Bk} functions, the median runtimes for Methods 3–4 were less than one second for all examined n, and were anywhere from 10–20 times faster than the median runtimes for Methods 5 and 6.

6. REAL DATA EXAMPLE

6.1 Data and analyses

To demonstrate the practical benefits of the rounding parameters when working with real data, we use electroencephalography (EEG) data obtained from Bache and Lichman [1]. Note that EEG data consist of electrical activities that are recorded from various electrodes on the scalp, and EEG patterns are used to infer information about mental processing. The EEG data used in this example were
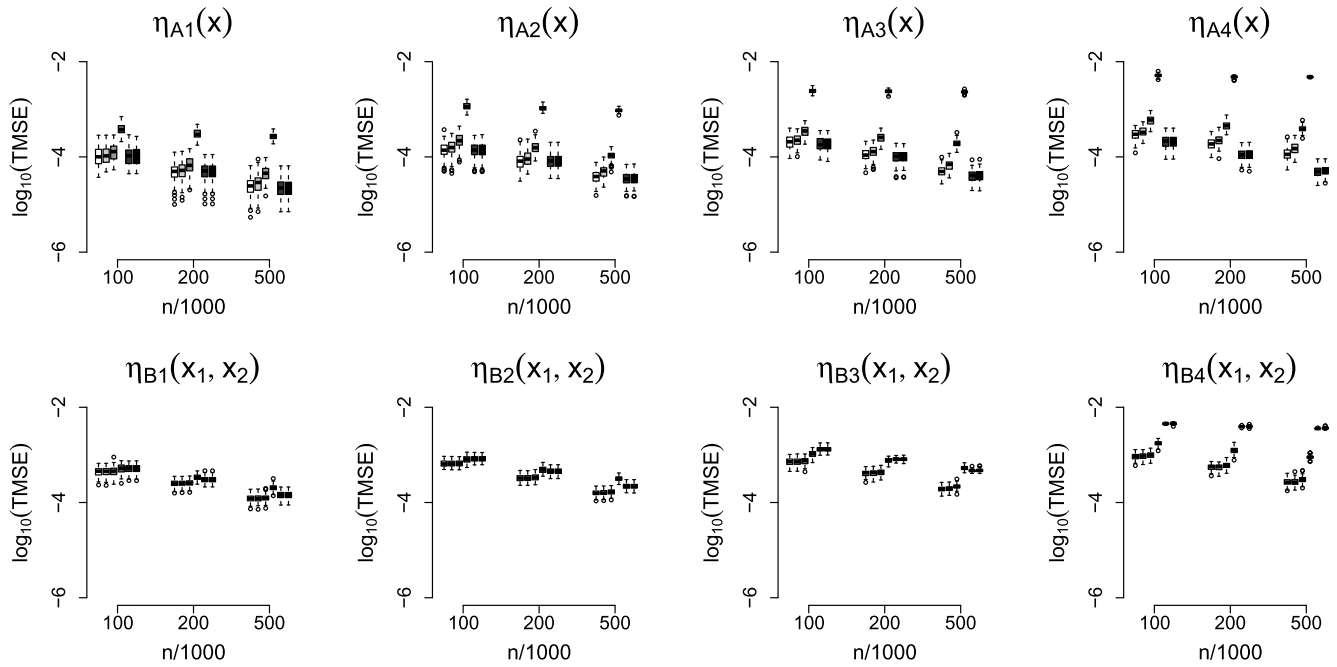
Figure 5. Simulation true MSEs on log-10 scale. Within each sample size, the six boxes correspond to Methods 1–6. Method 1 is SSANOVA with no rounding, Method 2 is SSANOVA with $r = .01$, Method 3 is SSANOVA with $r = .02$, Method 4 is SSANOVA with $r = .05$, Method 5 is gam.R, *and Method 6 is* bam.R.

Table 1. Median runtimes (seconds) for η_{Ak} functions

	η_{A1}			η_{A2}			η_{A3}			η_{A4}		
	100	200	500	100	200	500	100	200	500	100	200	500
Method 1 ($r = $ NA)	0.35	0.64	1.31	0.37	0.64	1.28	0.30	0.64	1.31	0.36	0.64	1.31
Method 2 ($r = 0.01$)	0.02	0.03	0.07	0.02	0.03	0.07	0.02	0.03	0.07	0.02	0.03	0.07
Method 3 ($r = 0.02$)	0.02	0.03	0.06	0.02	0.03	0.06	0.01	0.03	0.06	0.01	0.03	0.06
Method 4 ($r = 0.05$)	0.01	0.03	0.06	0.02	0.02	0.06	0.01	0.02	0.06	0.01	0.02	0.06
Method 5 (GAM)	1.44	2.24	4.05	1.40	2.11	4.03	1.47	2.12	4.06	1.40	2.11	4.06
Method 6 (BAM)	1.35	2.02	4.26	1.37	2.05	4.30	1.32	2.05	4.28	1.38	2.05	4.29

Table 2. Median runtimes (seconds) for η_{Bk} functions

	η_{B1}			η_{B2}			η_{B3}			η_{B4}		
	100	200	500	100	200	500	100	200	500	100	200	500
Method 1 ($r = $ NA)	3.80	6.60	14.84	3.80	6.60	14.82	3.81	6.61	14.85	3.81	6.60	14.85
Method 2 ($r = 0.01$)	0.85	0.80	1.35	0.85	0.80	1.34	0.85	0.80	1.35	0.85	0.80	1.35
Method 3 ($r = 0.02$)	0.34	0.51	0.99	0.34	0.51	0.99	0.34	0.51	0.99	0.34	0.51	0.99
Method 4 ($r = 0.05$)	0.28	0.43	0.90	0.28	0.43	0.90	0.28	0.43	0.90	0.28	0.43	0.90
Method 5 (GAM)	4.48	9.16	22.31	4.45	9.12	22.29	4.45	9.16	22.38	4.50	9.20	22.43
Method 6 (BAM)	4.75	7.81	18.55	4.73	7.78	18.55	4.74	7.80	18.61	4.77	7.85	18.65

recorded from both control and alcoholic subjects participating in an experiment at the Henri Begleiter Neurodynamic Lab at SUNY Brooklyn. The data were recorded during a standard visual stimulus event-related potential (ERP) experiment using a 61-channel EEG cap (see Figure 6). The data were recorded at a frequency of 256 Hz for one second following the presentation of the visual stimulus.

For the example, we analyzed data from the Pz electrode of 120 subjects (44 controls and 76 alcoholics), and we used 10 replications of the ERP experiment for each subject.[1] This resulted in $n = 307,200$ data points (120 subjects ×

---

[1]Note that data from subjects co2a0000425 and co2c0000391 were excluded from the analysis due to small amounts of data, and we used the first 10 replications for each subject.
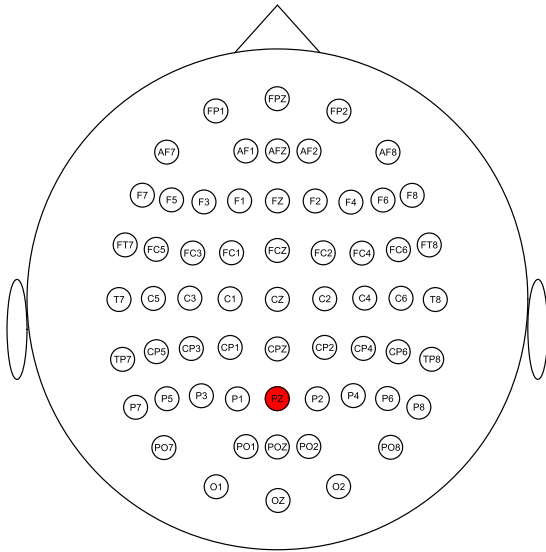
Figure 6. Depiction of the 61-channel EEG cap with the Pz electrode highlighted. Created using the `eegcap` function in the `eegkit` package [8].

256 time points × 10 replications). We analyzed the data using a two-way SSANOVA on the domain $[0,1] \times \{1,2\}$, where the first predictor is the time effect and the second predictor is the group effect (control vs. alcoholic); see the Appendix for an explanation of how the rounding parameter can be applied when working with continuous and nominal predictors. We used a cubic spline for the time effect, a nominal spline for the group effect, and $q = 50$ bin-sampled knots. Finally, we fit the model both with the unrounded data and with the time covariate rounded to the nearest .01 second (i.e., $r = .01$ on the interval $[0,1]$); note that setting $r = .01$ for the time covariate results in $u = 202$ unique covariate vectors, which is substantially less than the original $n = 307200$ data points.

6.2 Results

The predicted ERPs for the unrounded and rounded data are plotted in Figure 7. Note that there are no practical differences between the two solutions (c.f. Figure 7a,b). Furthermore, note that both solutions produced a GCV score of GCV=85.96 and variance-accounted-for value of $R^2 = 0.03$, suggesting that the rounded solution fits the data as well as the unrounded solution. It is also worth noting that the unrounded solution took over five times longer to fit compared to the rounded solution; furthermore, the unrounded solution required a substantial amount of RAM to fit the model, whereas the rounded solution is easily fittable on a standard laptop or tablet.

Comparing the estimated ERPs of the controls and alcoholics, there are obvious differences (see Figure 7). In particular, the alcoholic subjects are missing the P300 component of the ERP waveform (i.e., large positive peak occurring

about 300 ms after the stimulus). Note that the P300 component is thought to relate to a subject's internalization and/or categorization of stimuli, so these results suggest that alcoholic subjects have different information processing patterns for standard visual stimuli. This finding is consistent with previous findings regarding EEG patterns of alcoholic subjects [see 16, 19], and some research suggests that this sort of EEG pattern may predispose individuals to alcoholism [see 17, 18].

7. DISCUSSION

This paper proposes the use of rounding parameters to overcome the computational burden of fitting nonparametric regression models to super-large samples of data. By rounding each predictor to a given precision (e.g., 0.01), it is possible to estimate η using the $u \ll n$ unique rounded predictor variables. We have provided a simple Taylor heuristic that justifies the use of a small rounding parameter (e.g., $r = .01$) when using cubic smoothing splines for $x \in [0,1]$. Furthermore, we have provided methods for assessing the finite sample and asymptotic performance of the rounded SSANOVA estimator in various situations.

The simulation study and EEG example clearly demonstrate the benefits of the proposed rounding parameters. When fitting nonparametric regression models with large n, the simulation results reveal that setting $r_j \le .05$ can result in substantial computational savings without introducing much bias to the solution. Furthermore, the EEG data example reveals that there are no practical differences between the unrounded and rounded solutions (using $r = .01$) when analyzing real data. Thus, the rounding parameters offer a fast and stable method for fitting nonparametric regression models to very large samples.

In addition to providing a fast method for smoothing large datasets, the rounding parameters are also quite memory efficient. Because the rounding approach only uses the unique rounded-covariate values, it is never necessary to construct the full $n \times q$ model design matrix (or the $n \times n$ smoothing matrix). So, using the rounding parameters, it is possible to fit nonparametric regression models to very large samples using a standard laptop or tablet, e.g., all of the rounded SSANOVA models in this paper are easily fittable on a laptop with 4 GB of RAM. As a result, typical researchers now have the ability to discover functional relationships in super-large data sets without needing access to supercomputers or computing clusters.

As a final point, it should be noted that in some cases (e.g., large p) the number of unique rounded-covariate values may be very large. In such cases, forming the $u \times q$ model design matrix may require a substantial amount of memory (because u is so large). However, as is noted in Helwig [6] and Helwig and Ma [10], fitting an SSANOVA model only depends on various crossproduct vectors and matrices. So, if u is too large to form the full $u \times q$ model design matrix, then the needed crossproduct statistics can be formed in a
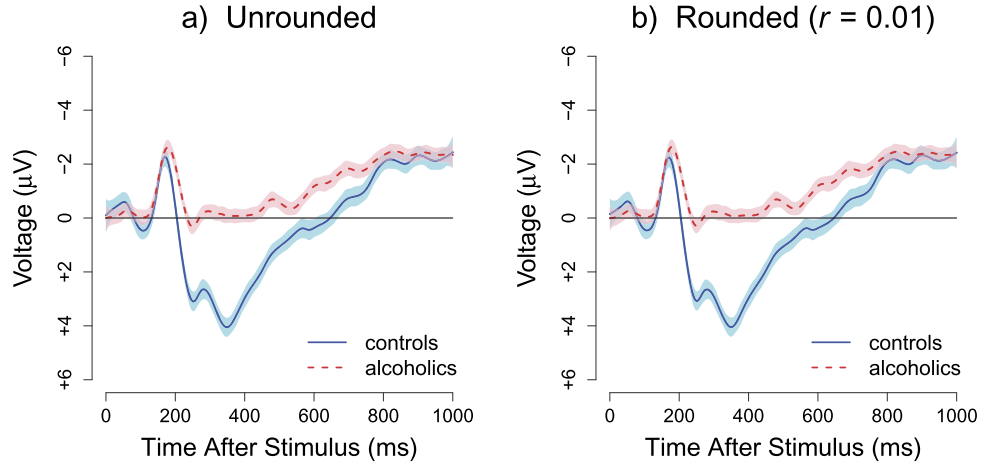
Figure 7. Predicted ERPs using the unrounded data (a) and rounded data (b). Shaded regions give a 99% Bayesian confidence interval around $\hat{\eta}$. Created using the `eegtime` function in the `eegkit` R package [8].

batch-processing manner similar to the approach used by Wood's (2015) `bam.R` function.

APPENDIX: ROUNDING ALGORITHM

In this section, we provide algorithms for rounding SSANOVA predictors and obtaining the sufficient statistics for the SSANOVA estimation. The first algorithm assumes that all of the covariates are continuous; extensions for nominal covariates will be discussed after the presentation of the initial algorithm.

First, let $r_j \in (0, 1]$ denote the rounding parameter for the j-th predictor, let $\tilde{\mathbf{x}}_j$ denote the $n \times 1$ vector containing the j-th predictor's scores, and let $x_{(i)j}$ denote the i-th order statistic of the j-th predictor. Next, initialize $\mathbf{g} \equiv \{1\}_{n \times 1}$ and $h \equiv 1$, and then calculate

for $j \in \{1, \ldots, p\}$
    1. $\mathbf{g} \leftarrow \mathbf{g} + h[\mathrm{rd}\{(1/r_j)(\tilde{\mathbf{x}}_j - x_{(1)j})/(x_{(n)j} - x_{(1)j})\}]$
    2. $h \leftarrow \mathrm{rd}(1 + 1/r_j)h$
end

where the rounding function $\mathrm{rd}\{\cdot\}$ rounds the input to the nearest integer. After running the for loop, we have $g_i \in \{1, \ldots, u\}$, where g_i denotes the i-th element of $\mathbf{g}$, and u is the total possible number of unique covariate vectors; thus, the vector $\mathbf{g}$ indexes the multi-dimensional rounded-covariate score for each observation.

The above result implies that the unique rounded-covariate scores (i.e., $\tilde{\mathbf{z}}_t$) can be obtained by sorting the predictors according to the g_i values, and then sampling one observation's covariate vector from each unique g_i value. Similarly, once the data is sorted according to the g_i values, the sum of the response at each unique covariate (i.e., $\tilde{y}_t$) and the number of observations at each unique covariate (i.e., w_t) can be easily calculated. Lastly, after calculating

$\|\mathbf{y}\|$, the SSANOVA model can be fit using the sufficient statistics from the rounded solution, i.e., $\tilde{\mathbf{z}}_t$, $\tilde{y}_t$, and w_t.

As we previously mentioned, the above algorithm can be modified to include nominal covariates as well. When working with nominal covariates, the algorithm assumes that all nominal covariates are of the form $x_{ij} \in \{1, \ldots, f_j\}$ where f_j is the number of factor levels of the j-th covariate. Assuming that $x_{ij} \in \{1, \ldots, f_j\}$, both steps of the rounding algorithm need to be slightly modified:

for $j \in \{1, \ldots, p\}$
    If x_{ij} is continuous
        1. $\mathbf{g} \leftarrow \mathbf{g} + h[\mathrm{rd}\{(1/r_j)(\tilde{\mathbf{x}}_j - x_{(1)j})/(x_{(n)j} - x_{(1)j})\}]$
        2. $h \leftarrow \mathrm{rd}(1 + 1/r_j)h$
    Else if x_{ij} is nominal
        1. $\mathbf{g} \leftarrow \mathbf{g} + h(\tilde{\mathbf{x}}_j - 1)$
        2. $h \leftarrow f_j h$
end

Using this simple modification, the rounding algorithm can be efficiently applied to any combination of continuous and nominal covariates.

REFERENCES

[1] BACHE, K. and LICHMAN, M. (2013). UCI Machine Learning Repository.

Smoothing spline ANOVA for super-large samples 443

[2] CRAVEN, P. and WAHBA, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik* **31** 377–403. MR0516581

[3] GU, C. (2013). *Smoothing Spline ANOVA Models*, Second ed. Springer-Verlag, New York. MR3025869

[4] GU, C. and WAHBA, G. (1991). Minimizing GCV/GML scores with multiple smoothing parameters via the Newton method. *SIAM Journal on Scientific and Statistical Computing* **12** 383–398. MR1087766

[5] HASTIE, T. and TIBSHIRANI, R. (1990). *Generalized Additive Models*. Chapman and Hall/CRC, New York. MR1082147

[6] HELWIG, N. E. (2013). Fast and stable smoothing spline analysis of variance models for large samples with applications to electroencephalography data analysis. PhD thesis, University of Illinois at Urbana-Champaign. MR3211368

[7] HELWIG, N. E. (2015a). bigsplines: Smoothing Splines for Large Samples R package version 1.0-6.

[8] HELWIG, N. E. (2015b). eegkit: Toolkit for Electroencephalography Data R package version 1.0-2.

[9] HELWIG, N. E. and MA, P. (in prep.). Stable smoothing spline approximation via bin-sampled knots.

[10] HELWIG, N. E. and MA, P. (in press). Fast and stable multiple smoothing parameter selection in smoothing spline analysis of variance models with large samples. *Journal of Computational and Graphical Statistics*.

[11] KIM, Y.-J. and GU, C. (2004). Smoothing spline Gaussian regression: More scalable computation via efficient approximation. *Journal of the Royal Statistical Society, Series B* **66** 337–356. MR2062380

[12] LI, K.-C. (1987). Asymptotic optimality for C_p, C_L, cross-validation and generalized cross-validation: Discrete index set. *The Annals of Statistics* **15** 958–975. MR0902239

[13] MA, P., HUANG, J. and ZHANG, N. (2015). Efficient computation of smoothing splines via adaptive basis sampling. *Biometrika* **102** 00–00.

[14] MA, P., MAHONEY, M. and YU, B. (2014). A statistical perspective on algorithmic leveraging. *JMLR: Workshop and Conference Proceedings* **32** 91–99.

[15] NYCHKA, D. (1995). Splines as local smoothers. *Annals of Statistics* **23** 1175–1197. MR1353501

[16] PORJESZ, B., BEGLEITER, H. and GAROZZO, R. (1980). Visual evoked potential correlates of information deficits in chronic alcoholics. In *Biological Effects of Alcohol* (H. Begleiter, ed.) 603–623. Plenum Press.

[17] PORJESZ, B. and BEGLEITER, H. (1990a). Event-related potentials for individuals at risk for alcoholism. *Alcohol* **7** 465–469.

[18] PORJESZ, B. and BEGLEITER, H. (1990b). Neuroelectric processes in individuals at risk for alcoholism. *Alcohol & Alcoholism* **25** 251–256.

[19] PORJESZ, B., BEGLEITER, H., BIHARI, B. and KISSIN, B. (1987). The N2 component of the event-related brain potential in abstinent alcoholics. *Electroencephalography and Clinical Neurophysiology* **66** 121–131.

[20] RUPPERT, D., WAND, M. P. and CARROLL, R. J. (2003). *Semiparametric Regression*. Cambridge University Press, Cambridge. MR1998720

[21] SILVERMAN, B. W. (1985). Aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society, Series B* **47** 1–52. MR0805063

[22] WAHBA, G. (1990). *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, Philadelphia. MR1045442

[23] WANG, X., DU, P. and SHEN, J. (2013). Smoothing splines with varying smoothing parameter. *Biometrika* **100** 955–970. MR3142343

[24] WOOD, S. N. (2006). *Generalized Additive Models: An Introduction with R*. Chapman & Hall, Boca Raton. MR2206355

[25] WOOD, S. N. (2015). mgcv: Mixed GAM Computation Vehicle with GCV/AIC/REML smoothness estimation and GAMMs by REML/PQL R package version 1.8-5.

Nathaniel E. Helwig
Department of Psychology & School of Statistics
University of Minnesota (Twin Cities)
USA
E-mail address: helwig@umn.edu

Ping Ma
Department of Statistics
University of Georgia
USA
E-mail address: pingma@uga.edu