Chapter 4

Smoothing Spline ANOVA Models and their Applications in Complex and Massive Datasets

Jingyi Zhang, Honghe Jin, Ye Wang, Xiaoxiao Sun,
Ping Ma and Wenxuan Zhong

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.75861

Abstract

Complex and massive datasets can be easily accessed using the newly developed data acquisition technology. In spite of the fact that the smoothing spline ANOVA models have proven to be useful in a variety of fields, these datasets impose the challenges on the applications of the models. In this chapter, we present a selected review of the smoothing spline ANOVA models and highlight some challenges and opportunities in massive datasets. We review two approaches to significantly reduce the computational costs of fitting the model. One real case study is used to illustrate the performance of the reviewed methods.

Keywords: smoothing spline, smoothing spline ANOVA models, reproducing kernel Hilbert space, penalized likelihood, basis sampling

1. Introduction

Among the nonparametric models, smoothing splines have been widely used in many real applications. There has been a rich body of literature in smoothing splines such as the additive smoothing spline [1–6], the interaction smoothing spline [7–10], and smoothing spline ANOVA (SSANOVA) models [11–14].

In this chapter, we focus on studying the SSANOVA models. Suppose that the data (y_i, x_i) and $i = 1, 2, \ldots, n$ are independent and identically distributed (i.i.d.) copies of (Y, X), where $Y \in \mathcal{Y} \subset \mathbb{R}$ is the response variable and $X \in \mathcal{X} \subset \mathbb{R}^d$ is the covariate variable. We consider the regression model:

$$y_i = \eta(x_i) + \epsilon_i, \quad i = 1, 2, \ldots, n, \tag{1}$$

where y_i is the response, η is the nonparametric function varying in an infinite-dimensional space, $x_i = \left(x_{i\langle 1\rangle}, \ldots, x_{i\langle d\rangle}\right)^T$ is on the domain $\mathcal{X} \subset \mathbb{R}^d$, and $\epsilon_i \overset{\text{i.i.d.}}{\sim} N\left(0, \sigma^2\right)$. More general cases, in which the conditional distribution of Y given x, denoted as $Y|x$, which follows different distributions instead of the Gaussian distribution, will be discussed later. The nonparametric function η in (1) can be decomposed into

$$\eta(x) = \eta_c + \sum_{j=1}^{d} \eta_j\left(x_{\langle j\rangle}\right) + \sum_{k<j} \eta_{kj}\left(x_{\langle k\rangle}, x_{\langle j\rangle}\right) + \ldots$$

through the functional ANOVA, where η_c is a constant function, η_j is the main effect function of $x_{\langle j\rangle}$, η_{kj} is the interaction effect of $x_{\langle k\rangle}$ and $x_{\langle j\rangle}$, and so on.

In the model (1), η can be estimated by minimizing the following penalized likelihood functional:

$$L(\eta) + \lambda J(\eta), \tag{2}$$

where $L(\eta)$ is a log likelihood measuring the goodness of fit of η, $J(\eta)$ is a quadratic functional on η to quantify its smoothness, and λ is the smoothing parameter balancing trade-offs between the goodness of fit and the smoothness of η [11–13]. The computational complexity of estimating η by minimizing (2) is of the order $O\left(n^3\right)$ for the sample of size n. Therefore, the high computational costs render SSANOVA models impractical for massive datasets. In this chapter, we review two methods to lower the computational costs. One approach is through the adaptive basis selection algorithm [14]. By carefully sampling a smaller set of basic functions conditional on the response variables, the adaptive sampling reduces the computational costs to $O\left(nn^{*2}\right)$, where $n^* \ll n$ is the number of the sampled basis functions. The computational costs can also be reduced by the rounding algorithm [15]. This algorithm can significantly decrease the sample size to μ by rounding the data with a given precision, where $\mu \ll n$. After rounding, the computational costs can be dramatically reduced to $O\left(\mu^3\right)$.

The rest of the chapter is organized as follows. Section 2 provides a detailed introduction to SSANOVA models and the model estimation. The details of adaptive basis selection algorithm and rounding algorithm are reviewed in Section 3. In Appendix, we demonstrate the numerical implementations using the R software.

2. Smoothing spline ANOVA models

In this section, we first review smoothing spline models and the reproducing kernel Hilbert space. Second, we present how to decompose a nonparametric function on tensor product domains, which lays the theoretical foundation for SSANOVA models. In the end, we show the estimation of SSANOVA models and illustrate the model with a real data example.

2.1. Introduction of smoothing spline models

In the model (1), η is located in an infinite-dimensional space. One way to estimate it is to add some constraints and estimate η in a finite-dimensional space. With the smoothness constraint, we estimate η by minimizing the penalized likelihood functional (2), and the minimizer of (2) is called a smoothing spline.

Example 1. *Cubic smoothing splines*

Suppose that $Y|x$ follows a normal distribution, that is, $Y|x_i \sim N(\eta(x_i), \sigma^2)$. Then, the penalized likelihood functional (2) *can be reduced as the penalized least squares:*

$$\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \eta(x_i)\right)^2 + \lambda\int_{\mathcal{X}}(\ddot{\eta}(x))^2\,dx, \tag{3}$$

where $\ddot{\eta} = d^2\eta/dx^2$. The minimizer of (3) *is called a cubic smoothing spline* [16–18]. *In* (3), *the first term quantifies the fidelity to the data, and the second term controls the roughness of the function.*

Example 2. *Exponential family smoothing splines*

Suppose that $Y|x$ follows an exponential family distribution:

$$Y|x_i \sim \exp\left\{(y\eta(x_i) - b(\eta(x_i)))/a(\phi) + c(y,\phi)\right\},$$

where $a > 0$, b, and c are known functions and ϕ is either known or a nuisance parameter. Then, η can be estimated by minimizing the following penalized likelihood functional [19, 20]:

$$-\frac{1}{n}\sum_{i=1}^{n}\left\{y_i\eta(x_i) - b(\eta(x_i))\right\} + \lambda J(\eta). \tag{4}$$

Note that the cubic smoothing spline in Example 1 *is a special case of exponential family smoothing splines when $Y|x$ follows the Gaussian distribution.*

The smoothing parameter λ is sensitive to the estimation of η (see **Figure 1**). Therefore, it is crucial to implement some proper smoothing parameter selection methods to estimate λ. One of the most popular methods is the generalized cross validation (GCV) [21, 22]. More details will be discussed in Section 2.6.2.

2.2. Reproducing kernel Hilbert space

We assume that readers are familiar with Hilbert space, which is a complete vector space with an inner product well defined that allows length and angle to be measured [23]. In a general Hilbert space, the continuity of a functional, which is required in minimizing (2) on $\mathcal{H} = \{J(\eta) < \infty\}$, is not always satisfied. To circumvent the problem, we optimize (2) in a special Hilbert space named reproducing kernel Hilbert space [24].

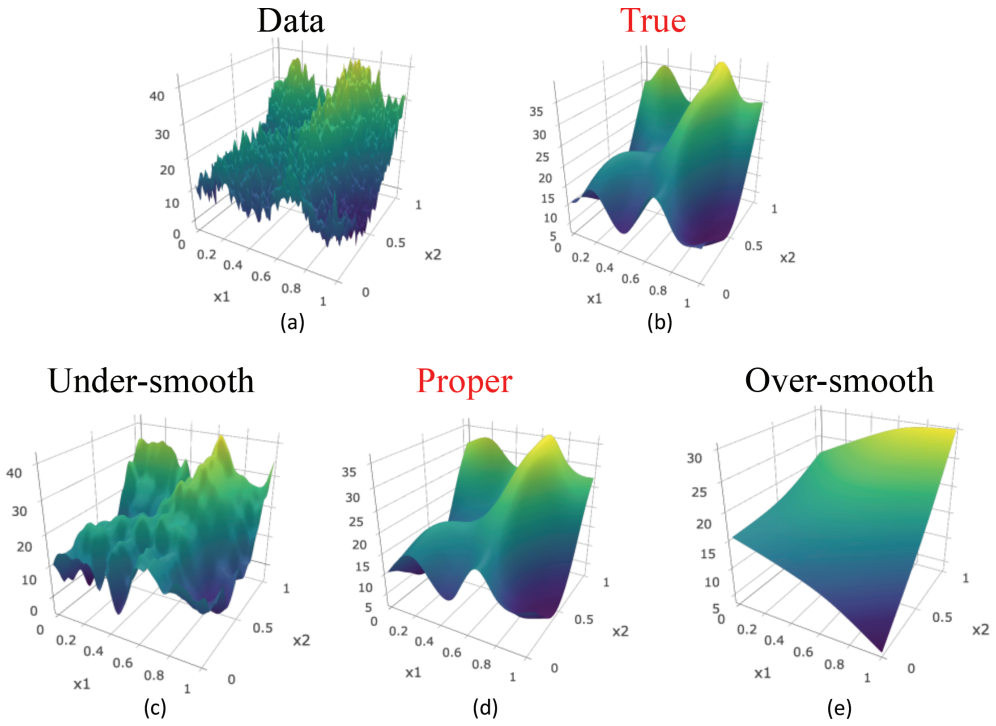

Figure 1. The data are generated by the model ϵ. $y = 5 + e^{3x_1} + 10^6 x_2^{11}(1-x_2)^6 + 10^4 x_2^2(1-x_2)^{10} + 5\cos(2\pi(x_1 - x_2)) + \epsilon$, where $\epsilon \sim N(0,1)$. Panels (a) and (b) show the data and the true function, respectively. The estimated functions depending on different smoothing parameters are shown in panels (c), (d), and (e). We set $\lambda \to 0$ in panel (c) and $\lambda \to \infty$ in panel (e). The proper λ selected by generalized cross validation (GCV) is used in panel (d).

For each $g \in \mathcal{H}$, there exists a corresponding continuous linear functional L_g such that $L_g(f) = \langle g, f \rangle$, where $f \in \mathcal{H}$ and $\langle \cdot, \cdot \rangle$ defines the inner product in $\mathcal{H}$. Conversely, an element $g_L \in \mathcal{H}$ can also be found such that $\langle g_L, f \rangle = L(f)$ for any continuous linear functional L of $\mathcal{H}$ [23]. This is known as the Riesz representation theorem.

Theorem 2.1. *Riesz representation*

Let $\mathcal{H}$ be a Hilbert space. For any functional L of $\mathcal{H}$, there uniquely exists an element $g_L \in \mathcal{H}$ such that

$$L(\cdot) = \langle g_L, \cdot \rangle,$$

where g_L is called the representer of L. The uniqueness is in the sense that g_1 and g_2 are considered as the same representer for any g_1 and g_2 satisfying $\|g_1 - g_2\| = 0$, where $\|\cdot\| = \langle \cdot, \cdot \rangle$ defines the norm in $\mathcal{H}$.

For a better construction of estimator minimizing (2), one needs the continuity of evaluation functional $[x]f = f(x)$. Roughly speaking, this means that if two functions f and g are close in

norm, that is, $\|f-g\|$ is small, then f and g are also pointwise close, that is, $|f(x)-g(x)|$ is small for all x.

Definition 1. *Reproducing kernel Hilbert space*

Consider a Hilbert space $\mathcal{H}$ consisting of functions on domain $\mathcal{X}$. For every element $x \in \mathcal{X}$, define an evaluation functional $[x]$ such that $[x]f = f(x)$. If all the evaluation functional $[x]$s are continuous, $\forall x \in \mathcal{X}$, then $\mathcal{H}$ is called a reproducing kernel Hilbert space.

By Theorem 2.1, for every evaluation functional $[x]$, there exists a corresponding function $R_x \in \mathcal{H}$ on $\mathcal{X}$ as the representer of $[x]$, such that $\langle R_x, f\rangle = [x]f = f(x)$ and $\forall f \in \mathcal{H}$. By the definition of evaluation functional, it follows

$$R_x(y) = \langle R_x, R_y\rangle = R_y(x). \tag{5}$$

The bivariate function $R(x,y) = \langle R_x, R_y\rangle$ is called the reproducing kernel of $\mathcal{H}$, which is unique if it exists. The essential meaning of the name "reproducing kernel" comes from its reproducing property

$$\langle R(x,\cdot), f\rangle = \langle R_x(\cdot), f\rangle = f(x)$$

for any $f \in \mathcal{H}$. In general, a reproducing kernel Hilbert space defines a reproducing kernel function that is both symmetric and positive definite. In addition, Moore-Aronszajn theorem states that every symmetric, positive definite kernel defines a unique reproducing kernel Hilbert space [25], and hence one can construct a reproducing kernel Hilbert space simply by specifying its reproducing kernel.

We now introduce the concept of tensor sum decomposition. Suppose that $\mathcal{H}$ is a Hilbert space and $\mathcal{G}$ is a linear subspace of $\mathcal{H}$. The linear subspace $\mathcal{G}^c = \{f \in \mathcal{H}, \langle f, g\rangle = 0, \forall g \in \mathcal{G}\}$ is called the orthogonal complement of $\mathcal{G}$. It is easy to verify that for any $f \in \mathcal{H}$, there exists a unique decomposition $f = f_{\mathcal{G}} + f_{\mathcal{G}^c}$, where $f_{\mathcal{G}} \in \mathcal{G}$ and $f_{\mathcal{G}^c} \in \mathcal{G}^c$. This decomposition is called a tensor sum decomposition, denoted by $\mathcal{H} = \mathcal{G} \oplus \mathcal{G}^c$. Suppose that $\mathcal{H}_1$ and $\mathcal{H}_2$ are two Hilbert spaces with inner products $\langle\cdot,\cdot\rangle_1$ and $\langle\cdot,\cdot\rangle_2$. If the only common element of these two spaces is $\mathbf{0}$, one can also define a tensor sum Hilbert space $\mathcal{H} = \mathcal{H}_1 \oplus \mathcal{H}_2$. For any $f, g \in \mathcal{H}$, one has unique decompositions $f = f_1 + f_2$ and $g = g_1 + g_2$, where $f_1, g_1 \in \mathcal{H}_1$ and $f_2, g_2 \in \mathcal{H}_2$. Moreover, the inner product defined on $\mathcal{H}$ would be $\langle f, g\rangle = \langle f_1, g_1\rangle_1 + \langle f_2, g_2\rangle_2$. The following theorem provides the rules in the tensor sum decomposition of a reproducing kernel Hilbert space.

Theorem 2.2 *Suppose that R_1 and R_2 are the reproducing kernel Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$, respectively. If $\mathcal{H}_1 \cap \mathcal{H}_2 = \{\mathbf{0}\}$, then $\mathcal{H} = \mathcal{H}_1 \oplus \mathcal{H}_2$ has a reproducing kernel $R = R_1 + R_2$.*

Conversely, if the reproducing kernel R of $\mathcal{H}$ can be decomposed into $R = R_1 + R_2$, where both R_1 and R_2 are positive definite, and they are orthogonal to each other, that is, $\langle R_1(x_1,\cdot), R_2(x_2,\cdot)\rangle = 0$ for $\forall x_1, x_2 \in \mathcal{X}$, then the spaces $\mathcal{H}_1$ and $\mathcal{H}_2$ corresponding to the kernels R_1 and R_2 form a tensor sum decomposition $\mathcal{H} = \mathcal{H}_1 \oplus \mathcal{H}_2$.

2.3. Representer theorem

In (2), the smoothness penalty term $J(\eta) = J(\eta, \eta)$ is nonnegative definite, that is, $J(\eta, \eta) \geq 0$, and hence it is a squared semi-norm on the reproducing kernel Hilbert space $\mathcal{H} = \{\eta : J(\eta) < \infty\}$. Denote $\mathcal{N}_J = \{\eta \in \mathcal{H} : J(\eta) = 0\}$ as the null space of $J(\eta)$ and $\mathcal{H}_J$ as its orthogonal complement. By the tensor sum decomposition $\mathcal{H} = \mathcal{N}_J \oplus \mathcal{H}_J$, one may decompose the η into two parts: one in the null space $\mathcal{N}_J$ that has no contribution on the smoothness penalty and the other in $\mathcal{H}_J$ "reproduced" by the reproducing kernel $R(\cdot, \cdot)$ [12].

Theorem 2.3. *There exist coefficient vectors* $\mathbf{d} = (d_1, \ldots, d_M)^T \in \mathbb{R}^M$ *and* $\mathbf{c} = (c_1, \ldots, c_n)^T \in \mathbb{R}^n$ *such that*

$$\eta(x) = \sum_{k=1}^{M} d_k \xi_k(x) + \sum_{i=1}^{n} c_i R(x_i, x), \tag{6}$$

where $\{\xi_k, k = 1, .., M\}$ *is the basis of null space* $\mathcal{N}_J$ *and* $R(\cdot, \cdot)$ *is the reproducing kernel of* $\mathcal{H}_J$.

This theorem indicates that although the minimization problem is in an infinite-dimensional space, the minimizer of (2) lies in a data-adaptive finite-dimensional space.

2.4. Function decomposition

The decomposition of a multivariate function is similar to the classical ANOVA. In this section, we present the functional ANOVA which lays the foundation for SSANOVA models.

2.4.1. One-way ANOVA decomposition

We consider a classical one-way ANOVA model $y_{ij} = \mu_i + \epsilon_{ij}$, where y_{ij} is the observed data, μ_i is the treatment mean for $i = 1, \ldots, K$ and $j = 1, \ldots, J$, and ϵ_{ij}s are the random errors. The treatment mean μ_i can be further decomposed as $\mu_i = \mu + \alpha_i$, where μ is the overall mean and α_i is the treatment effect with the constraint $\sum_{i=1}^{K} \alpha_i = 0$.

Similar to the classical ANOVA decomposition, a univariate function $f(x)$ can be decomposed as

$$f = Af + (I - A)f = f_c + f_x, \tag{7}$$

where A is an averaging operator that averages the effect of x and I is an identity operator. The operator A averages a function f to a constant function f_c satisfying $A(I - A) = 0$. For example, one can take $Af = \int_0^1 f(x)\,dx$ in $\mathcal{L}_1[0,1] = \left\{f : \int_0^1 |f(x)|\,dx < \infty\right\}$. In (7), $f_c = Af$ is the mean function, and $f_x = (I - A)f$ is the treatment effect.

2.4.2. Multiway ANOVA decomposition

On a d-dimensional product domain $\mathcal{X} = \prod_{j=1}^{d} \mathcal{X}_j \in \mathbb{R}^d$, a multivariate function $f\left(x_{\langle 1 \rangle}, \ldots, x_{\langle d \rangle}\right)$ can be decomposed similarly to the one-way ANOVA decomposition. Let A_j, $j = 1, \ldots, d$, be the

average operator on $\mathcal{X}_j$, and then $A_j f$ is a constant function on $\mathcal{X}_j$. One can define the ANOVA decomposition on $\mathcal{X}$ as

$$\begin{aligned} f &= \left\{\prod_{j=1}^{d}(I - A_j + A_j)\right\} f \\ &= \sum_{\mathcal{S}} \left\{\prod_{j\in\mathcal{S}}(I - A_j)\prod_{j\notin\mathcal{S}} A_j\right\} f = \sum_{\mathcal{S}} f_{\mathcal{S}}, \end{aligned} \tag{8}$$

where $\mathcal{S}\subseteq\{1,\ldots,d\}$. The term $f_c = \prod_{j=1}^{d} A_j f$ is the constant function, $f_j = (I - A_j)\prod_{\alpha\neq j} A_\alpha f$ is the main effect term of $x_{\langle j\rangle}$, the term $f_{\mu\nu} = (I - A_\mu)(I - A_\nu)\prod_{\alpha\neq\mu,\nu} A_\alpha f$ is the interaction of $x_{\langle\mu\rangle}$ and $x_{\langle\nu\rangle}$, and so on.

2.5. Some examples of model conduction

Smoothing splines on $\mathcal{C}^{(m)}[0,1]$. If we define

$$J(\eta) = \int_0^1 \left(\eta^{(m)}\right)^2 dx$$

in the space $\mathcal{C}^{(m)}[0,1] = \left\{f : f^{(m)} \in \mathcal{L}_2[0,1]\right\}$, where $f^{(m)}$ denotes the mth differentiation of f, $\mathcal{L}_2 = \left\{f : \int_0^1 (f(x))^2 dx < \infty\right\}$, then the minimizer of (2) is called a polynomial smoothing spline.

Here, we use an inner product

$$\langle f, g\rangle = \sum_{\nu=0}^{m-1}\left(\int_0^1 f^{(\nu)}(x)dx\right)\left(\int_0^1 g^{(\nu)}(x)\,dx\right) + \int_0^1 f^{(m)}(x)g^{(m)}(x)\,dx. \tag{9}$$

One can easily check that (9) is a well-defined inner product in $\mathcal{C}^{(m)}[0,1]$ with $\mathcal{H}_0 = \left\{f : f^{(m)} = 0\right\}$ equipped with the inner product $\sum_{\nu=0}^{m-1}\left(\int_0^1 f^{(\nu)}(x)\,dx\right)\left(\int_0^1 g^{(\nu)}(x)\,dx\right)$ [21]. To construct the reproducing kernel, define

$$k_\nu(x) = -\left(\sum_{\mu=-\infty}^{-1} + \sum_{\mu=1}^{\infty}\right)\frac{\exp(2\pi\mathbf{i}\mu x)}{(2\pi\mathbf{i}\mu)^\nu}, \tag{10}$$

where $\mathbf{i} = \sqrt{-1}$. One can verify that $\int_0^1 k_\nu^{(\mu)}(x)\,dx = \delta_{\nu\mu}$ and $\nu, \mu = 0, 1, \ldots, m-1$, where $\delta_{\nu\mu}$ is the Kronecker delta [26]. Indeed, $\{k_0, \ldots, k_{m-1}\}$ forms an orthonormal basis of $\mathcal{H}_0$. Then, the reproducing kernel in $\mathcal{H}_0$ is

$$R_0(x,y) = \sum_{\nu=0}^{m-1} k_\nu(x)k_\nu(y).$$

For space

$$\mathcal{H}_1 = \left\{ f : \int_0^1 f^{(\nu)}(x)\,dx = 0, \nu = 0, 1, \ldots, m-1, f^{(m)} \in \mathcal{L}_2[0,1] \right\},$$

one can check that the reproducing kernel in $\mathcal{H}_1$ is

$$R_1(x,y) = k_m(x)k_m(y) + (-1)^{m-1}k_{2m}(x-y),$$

(See details in [11]; Section~2.3).

SSANOVA models on product domains: A natural way to construct reproducing kernel Hilbert space on product domain $\prod_{j=1}^d \mathcal{X}_j$ is taking the tensor product of spaces constructed on the marginal domains $\mathcal{X}_j$s. According to the Moore-Aronszajn theorem, every nonnegative definite function R corresponds to a reproducing kernel Hilbert space with R as its reproducing kernel. Therefore, the construction of the tensor product reproducing kernel Hilbert space is induced by constructing its reproducing kernel.

Theorem 2.4. *Suppose that if* $R_{\langle 1\rangle}\left(x_{\langle 1\rangle}, y_{\langle 1\rangle}\right)$ *is nonnegative definite on* $\mathcal{X}_1$ *and* $R_{\langle 2\rangle}\left(x_{\langle 2\rangle}, y_{\langle 2\rangle}\right)$ *is nonnegative definite on* $\mathcal{X}_2$*, then* $R(x,y) = R_{\langle 1\rangle}\left(x_{\langle 1\rangle}, y_{\langle 1\rangle}\right)R_{\langle 2\rangle}\left(x_{\langle 2\rangle}, y_{\langle 2\rangle}\right)$ *is nonnegative definite on* $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2$.

Theorem 2.4 implies that a reproducing kernel R on tensor product reproducing kernel Hilbert space can be derived from the reproducing kernels on marginal domains. Indeed, let $\mathcal{H}_{\langle j\rangle}$ be the space on $\mathcal{X}_j$ with reproducing kernel $R_{\langle j\rangle}$, where $j = 1, 2$. Then, $R = R_{\langle 1\rangle}R_{\langle 2\rangle}$ is nonnegative definite on $\mathcal{X}_1 \times \mathcal{X}_2$. The space $\mathcal{H}$ corresponding to $R(\cdot,\cdot)$ is called the tensor product space of $\mathcal{H}_{\langle 1\rangle}$ and $\mathcal{H}_{\langle 2\rangle}$, denoted by $\mathcal{H} = \mathcal{H}_{\langle 1\rangle} \otimes \mathcal{H}_{\langle 2\rangle}$.

One can decompose each marginal space $\mathcal{H}_{\langle j\rangle}$ into $\mathcal{H}_{\langle j\rangle} = \mathcal{H}_{\langle j\rangle 0} \oplus \mathcal{H}_{\langle j\rangle 1}$, where $\mathcal{H}_{\langle j\rangle 0}$ denotes the averaging space and $\mathcal{H}_{\langle j\rangle 1}$ denotes the orthogonal complement. Then, by the discussion in Section 2.4, the one-way ANOVA decomposition on each marginal space can be generalized to a multidimensional space $\mathcal{H} = \otimes_{j=1}^d \mathcal{H}_{\langle j\rangle}$ as

$$\begin{aligned} \mathcal{H} &= \otimes_{j=1}^d \left(\mathcal{H}_{\langle j\rangle 0} \oplus \mathcal{H}_{\langle j\rangle 1}\right) \\ &= \oplus_{\mathcal{S}} \left\{ \left(\otimes_{j\in\mathcal{S}} \mathcal{H}_{\langle j\rangle 1} \right) \otimes \left(\otimes_{j\notin\mathcal{S}} \mathcal{H}_{\langle j\rangle 0} \right) \right\} \\ &= \oplus_{\mathcal{S}} \mathcal{H}_{\mathcal{S}}, \end{aligned} \tag{11}$$

where $\mathcal{S}$ denotes all the subsets of $\{1, \ldots, d\}$. The component $f_{\mathcal{S}}$ in (8) is in the space $\mathcal{H}_{\mathcal{S}}$. Based on the decomposition, the minimizer of (2) is called a tensor product smoothing spline. One can construct a tensor product smoothing spline following Theorem 2.3, in which the reproducing kernel term may be calculated in the same way as the tensor product (11).

In the following, we will give some examples of tensor product smoothing splines on product domains.

2.5.1. Smoothing splines on $\{1,\ldots,K\}\times[0,1]$

We construct the reproducing kernel Hilbert space by using

$$R_{\langle 1\rangle 0}=1/K \text{and} R_{\langle 1\rangle 1}=I_{[x_{\langle 1\rangle}=y_{\langle 1\rangle}]}$$

on $\{1,\ldots,K\}$. On $[0,1]$, assume that if $m=2$, then we have

$$R_{\langle 2\rangle 0}=1+k_1(x_{\langle 2\rangle})k_1\left(y_{\langle 2\rangle}\right)$$

and

$$R_{\langle 2\rangle 1}=k_2(x_{\langle 2\rangle})k_2\left(y_{\langle 2\rangle}\right)-k_4\left(x_{\langle 2\rangle}-y_{\langle 2\rangle}\right).$$

In this case, the space $\mathcal{H}$ can be further decomposed as

$$\mathcal{H}=\left(\mathcal{H}_{\langle 1\rangle 0}\oplus\mathcal{H}_{\langle 1\rangle 1}\right)\otimes\left(\mathcal{H}_{\langle 2\rangle 00}\oplus\mathcal{H}_{\langle 2\rangle 01}\oplus\mathcal{H}_{\langle 2\rangle 1}\right). \tag{12}$$

The reproducing kernels of tensor product cubic spline on $\{1,\ldots,K\}\times[0,1]$ are listed in **Table 1**.

On other product domains, for example, $[0,1]^2$, the tensor product reproducing kernel Hilbert space can be decomposed in a similar way. More examples are available in ([11], Section~2.4).

2.5.1.1. General form

In general, a tensor product reproducing kernel Hilbert space can be specified as $\mathcal{H}=\oplus_j\mathcal{H}_j$, where $j\in B$ is a genetic index. Suppose that $\mathcal{H}_j$ is equipped with a reproducing kernel R_j and an inner product $\langle f,g\rangle_j$. Denote f_j as the projection of f onto $\mathcal{H}_j$. Then, an inner product in $\mathcal{H}$ can be defined as

Subspace	**Reproducing kernel**
$\mathcal{H}_{\langle 1\rangle 0}\otimes\mathcal{H}_{\langle 2\rangle 00}$	$1/K$
$\mathcal{H}_{\langle 1\rangle 0}\otimes\mathcal{H}_{\langle 2\rangle 01}$	$k_1(x_{\langle 2\rangle})k_1\left(y_{\langle 2\rangle}\right)/K$
$\mathcal{H}_{\langle 1\rangle 0}\otimes\mathcal{H}_{\langle 2\rangle 1}$	$\left[k_2(x_{\langle 2\rangle})k_2\left(y_{\langle 2\rangle}\right)-k_4\left(x_{\langle 2\rangle}-y_{\langle 2\rangle}\right)\right]/K$
$\mathcal{H}_{\langle 1\rangle 1}\otimes\mathcal{H}_{\langle 2\rangle 00}$	$I_{[x_{\langle 1\rangle}=y_{\langle 1\rangle}]}-1/K$
$\mathcal{H}_{\langle 1\rangle 1}\otimes\mathcal{H}_{\langle 2\rangle 01}$	$\left[I_{[x_{\langle 1\rangle}=y_{\langle 1\rangle}]}-1/K\right]k_1(x_{\langle 2\rangle})k_1\left(y_{\langle 2\rangle}\right)$
$\mathcal{H}_{\langle 1\rangle 1}\otimes\mathcal{H}_{\langle 2\rangle 1}$	$\left[I_{[x_{\langle 1\rangle}=y_{\langle 1\rangle}]}-1/K\right]\left[k_2(x_{\langle 2\rangle})k_2\left(y_{\langle 2\rangle}\right)-k_4\left(x_{\langle 2\rangle}-y_{\langle 2\rangle}\right)\right]$

Table 1. Reproducing kernels of (12) on $\{1,\ldots,K\}\times[0,1]$ when $m=2$.

$$J(f,g)=\sum_j \theta_j^{-1}\left\langle f_j,g_j\right\rangle_{j'} \tag{13}$$

where $\theta_j \geq 0$ are the tuning parameters. If a penalty J in (2) has the form (13), the SSANOVA models can be defined on the space $\mathcal{H}=\oplus_j \mathcal{H}_j$ with the reproducing kernel:

$$R=\sum_j \theta_j R_j. \tag{14}$$

2.6. Estimation

In this section, we show the procedure of estimating the minimizer $\widehat{\eta}$ of (2) under the Gaussian assumption and selecting the smoothing parameters.

2.6.1. Penalized least squares

We consider the same model shown in (1), and then the η can be estimated by minimizing the penalized least squares:

$$\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\eta(x_i)\right)^2+\lambda J(\eta). \tag{15}$$

Let S denote the $n\times M$ matrix with the (i,j)th entry $\xi_j(x_i)$ as in (6) and R denote the $n\times n$ matrix with the (i,j)th entry $R(x_i,x_j)$ with the form (14). Then, based on Theorem 2.3, $\boldsymbol{\eta}$ can be expressed as

$$\boldsymbol{\eta}=S\mathbf{d}+R\mathbf{c},$$

where $\boldsymbol{\eta}=(\eta(x_1),\ldots,\eta(x_n))^T$, $\mathbf{d}=(d_1,\ldots,d_M)^T$, and $\mathbf{c}=(c_1,\ldots,c_n)^T$. The least squares term in (15) becomes

$$\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\eta(x_i)\right)^2=\frac{1}{n}(\mathbf{y}-S\mathbf{d}-R\mathbf{c})^T(\mathbf{y}-S\mathbf{d}-R\mathbf{c}),$$

where $\mathbf{y}=\left(y_1,\ldots,y_n\right)^T$.

By the reproducing property (5), the roughness penalty term can be expressed as

$$J(\eta)=\sum_{i=1}^{n}\sum_{j=1}^{n}c_i R(x_i,x_j)c_j=\mathbf{c}^T R\mathbf{c}.$$

Therefore, the penalized least squares criterion (15) becomes

$$\frac{1}{n}(\mathbf{y}-S\mathbf{d}-R\mathbf{c})^T(\mathbf{y}-S\mathbf{d}-R\mathbf{c})+\lambda\mathbf{c}^T R\mathbf{c}. \tag{16}$$

The penalized least squares (16) is a quadratic form of both **d** and **c**. By differentiating (16), one can obtain the linear system:

$$\begin{pmatrix} S^TS & S^TR \\ R^TS & R^TR + n\lambda R \end{pmatrix} \begin{pmatrix} \mathbf{d} \\ \mathbf{c} \end{pmatrix} = \begin{pmatrix} S^T\mathbf{y} \\ R^T\mathbf{y} \end{pmatrix}. \tag{17}$$

Note that (17) only works for penalized least squares (15), and hence a normal assumption is needed in this case.

2.6.2. Selection of smoothing parameters

In SSANOVA models, properly selecting smoothing parameters is important to estimate η [9, 27, 28], as shown in **Figure 1**. Here, we introduce the generalized cross validation (GCV) method for the smoothing parameter selection.

For the multivariate predictors, the penalty term in (15) has the form

$$\lambda J(f) = \lambda \sum_{j=1}^{S} \theta_j^{-1} \left\langle f_j, f_j \right\rangle_j,$$

where S is the number of smoothing parameters, which is related to the functional ANOVA decomposition, and θ_j's are the extra smoothing parameters. To avoid overparameterization, we treat $\lambda = (\lambda/\theta_1, \cdots, \lambda/\theta_S)^T$ as the effective smoothing parameters.

A GCV score is defined as

$$V(\lambda) = \frac{n^{-1}\mathbf{y}^T(I - A(\lambda))^2\mathbf{y}}{[n^{-1}\mathrm{tr}(I - A(\lambda))]^2},$$

where $A(\lambda)$ is a symmetric matrix similar to the hat matrix in linear regression. We can select a proper λ by minimizing the GCV score [21].

2.7. Case study: Twitter data

Tweets in the contiguous United States were collected over five weekdays in January 2014. The dataset contains information of time, GPS location, and tweet counts (see **Figure 2**). To illustrate the application of SSANOVA models, we study the time and spatial patterns in this data.

The bivariate function $\eta(x_{\langle 1\rangle}, x_{\langle 2\rangle})$ is a function of time and location, where $x_{\langle 1\rangle}$ denotes the time and $x_{\langle 2\rangle}$ represents the longitude and latitude coordinates. We use the thin-plate spline for the spatial variable and cubic spline for the time variable. As a rotation-free method, the thin-plate spline is popular for modeling spatial data [29–31]. For a better interpretation, we decompose the function η as

$$\eta(x_{\langle 1\rangle}, x_{\langle 2\rangle}) = \eta_c + \eta_1(x_{\langle 1\rangle}) + \eta_2(x_{\langle 2\rangle}) + \eta_{12}(x_{\langle 1\rangle}, x_{\langle 2\rangle}),$$

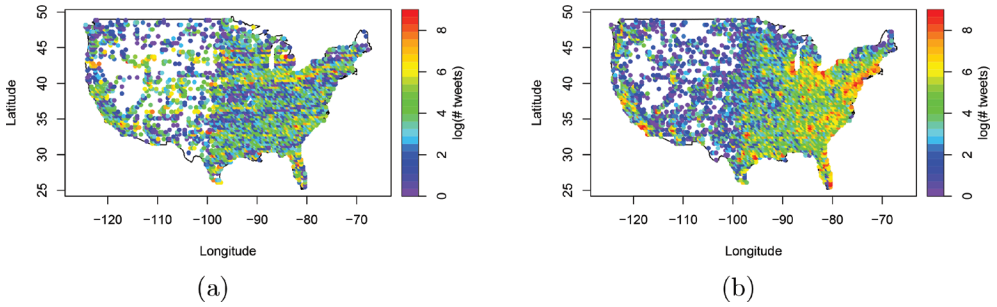

Figure 2. Heatmaps of tweet counts in the contiguous United States. (a) Tweet counts at 2:00 a.m. (b) Tweet counts at 6:00 p.m.

where η_c is a constant function; η_1 and η_2 are the main effects of time and location, respectively; and η_{12} is the spatial-time interaction effect.

The main effects of time and location are shown in **Figure 3**. Obviously, in panel (a), the number of tweets has the periodic effect, where it attains the maximum value at 8:00 p.m. and the minimum value at 5:00 a.m. The main effect of time shows the variations of Twitter usages in the United States. In addition, we can infer how the tweet counts vary across different locations based on panel (b) in **Figure 3**. There tend to be more tweets in the east than those in the west regions and more tweets in the coastal zone than those in the inland. We use the scaled dot product

$$\pi = (\widehat{\boldsymbol{\eta}}_{12})^T \widehat{\mathbf{y}} / \|\widehat{\mathbf{y}}\|^2$$

to quantify the percentage decomposition of the sum of squares of $\widehat{\mathbf{y}}$ [11], where $\widehat{\mathbf{y}} = (\widehat{y}_1, \ldots, \widehat{y}_n)^T$ is the predicted values of $\log(\#\text{Tweets})$, and $\widehat{\boldsymbol{\eta}}_{12} = (\eta_{12}(x_1), \ldots, \eta_{12}(x_n))^T$ is the estimated interaction effect term, where $\eta_{12}(x) = \eta_{12}(x_{\langle 1 \rangle}, x_{\langle 2 \rangle})$. In our fitted model,

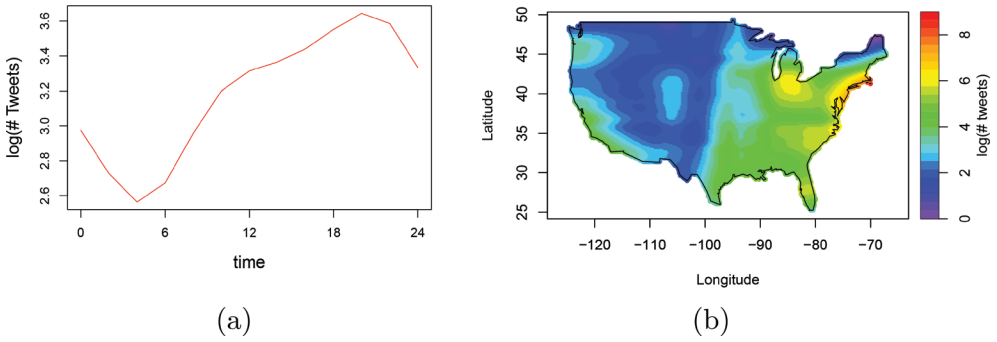

Figure 3. (a) The main effect function of time (hours). (b) The main effect function of location.

$\pi = 3 \times 10^{-16}$, which is so small that the interaction term is negligible. This indicates that there is no significant difference for the Twitter usages across time in the contiguous United States.

3. Efficient approximation algorithm in massive datasets

In this section, we consider SSANOVA models under the big data settings. The computational cost of solving (17) is of the order $O(n^3)$ and thus gives rise to a challenge on the application of SSANOVA models when the volume of data grows. To reduce the computational load, an obvious way is to select a subset of basis functions randomly. However, it is hard to keep the data features by uniform sampling. In the following section, we present an adaptive basis selection method and show its advantages over uniform sampling [14]. Instead of selecting basis functions, another approach to reduce the computational cost is shrinking the original sample size by rounding algorithm [15].

3.1. Adaptive basis selection

A natural way to select the basis functions is through uniform sampling. Suppose that we randomly select a subset $\breve{x} = \{\breve{x}_1, \ldots, \breve{x}_{\breve{n}}\}$ from $\{x_1, \ldots, x_n\}$, where $\breve{n}$ is the subsample size. Thus, the kernel matrix would be $R(\breve{x}_i, x)$, $i = 1, \ldots, \breve{n}$. Then, one minimizes (17) in the effective model space:

$$\mathcal{H}_E = \mathcal{N} \oplus \operatorname{span}\{R(\breve{x}_i, x), i = 1, 2, \ldots, \breve{n}\}.$$

The computational cost will be reduced significantly to $O(n\breve{n}2)$ if $\breve{n}$ is much smaller than n. Furthermore, it can be proven that the minimizer of (2), $\breve{\eta}$, by uniform sampling basis selection, has the same asymptotic convergence rate as the full basis minimizer $\widehat{\eta}$.

Although the uniform basis selection reduces the computational cost and the corresponding $\breve{\eta}$ achieves the optimal asymptotic convergence rate, it may fail to retain the data features occasionally. For example, when the data are not evenly distributed, it is hard for uniform sampling to capture the feature where there are only a few data points. In [14], an adaptive basis selection method is proposed. The main idea is to sample more basis functions where the response functions change largely and fewer basis functions on those flat regions. More details of adaptive basis selection method are shown in the following procedure:

Step 1 Divide the range of responses $\{y_i\}_{i=1}^n$ into K disjoint intervals, $S_1, \ldots, S_K$. Denote $|S_k|$ as the number of observations in S_k.

Step 2 For each S_k, draw a random sample of size n_k from this collection. Let $x^{*(k)} = \left(x_1^{*(k)}, \ldots, x_{n_k}^{*(k)}\right)$ be the predictor values.

Step 3 Combine $x^{*(1)}, \ldots, x^{*(K)}$ together to form a set of sampled predictor values $\{x_1^*, \ldots, x_{n*}^*\}$, where $n^* = \sum_{k=1}^K n_k$.

Step 4 Define

$$\mathcal{H}_E = \mathcal{H}_0 \oplus \operatorname{span}\left\{R\left(x_i^*, \cdot\right), i = 1, 2, \ldots, n^*\right\}$$

as the effective model space.

By adaptive basis selection, the minimizer of (2) keeps the same form as that in Theorem 2.3:

$$\eta_A(x) = \sum_{k=1}^{M} d_k \xi_k(x) + \sum_{i=1}^{n^*} c_i R\left(x_i^*, x\right).$$

Let R_* be an $n \times n^*$ matrix, and its (i,j)th entry is $R\left(x_i, x_j^*\right)$. Let R_{**} be an $n^* \times n^*$ matrix, and its (i,j)th entry is $R\left(x_i^*, x_j^*\right)$. Then, the estimator $\boldsymbol{\eta_A}$ satisfies

$$\boldsymbol{\eta_A} = S\mathbf{d_A} + R_*\mathbf{c_A},$$

where $\boldsymbol{\eta_A} = \left(\eta_A(x_1), \cdots, \eta_A(x_{n^*})\right)^T$, $\mathbf{d_A} = (d_1, \ldots, d_M)^T$, and $\mathbf{c_A} = (c_1, \ldots, c_{n^*})^T$. Similar to (17), the linear system of equations in this case is

$$\begin{pmatrix} S^TS & S^TR_* \\ R_*^TS & R_*^TR_* + n\lambda R_{**} \end{pmatrix} \begin{pmatrix} \mathbf{d_A} \\ \mathbf{c_A} \end{pmatrix} = \begin{pmatrix} S^T\mathbf{y} \\ R_*^T\mathbf{y} \end{pmatrix}. \tag{18}$$

The computational complexity of solving (18) is of the order $O\left(nn^{*2}\right)$, so the method decreases the computational cost significantly. It can also be shown that the adaptive sampling basis selection smoothing spline estimator η_A has the same convergence property as the full basis method. More details about the consistency theory can be found in [14]. Moreover, adaptive sampling basis selection method for exponential family smoothing spline models was developed in [32].

3.2. Rounding algorithm

Other than sampling a smaller set of basis functions to save the computational resources, for example, the adaptive basis selection method presented previously, [15] proposed a new rounding algorithm to fit SSANOVA models in the context of big data.

Rounding algorithm: The details of rounding algorithm can be shown in the following procedure:

Step 1 Assume that all predictors are continuous.

Step 2 Convert all predictors to the interval $[0, 1]$.

Step 3 Round the raw data by using the transformation:

$$z_{i\langle j\rangle} = RD\left(x_{i\langle j\rangle}/r_{\langle j\rangle}\right) r_{\langle j\rangle}, \text{for } i \in \{1, \cdots, n\}, j \in \{1, \cdots, d\},$$

where the rounding parameter $r_{\langle j\rangle} \in (0, 1]$ and rounding function $RD(\cdot)$ transform input data to the nearest integer.

Step 4 After replacing $x_{i\langle j\rangle}$ with $z_{i\langle j\rangle}$, we redefine S and R in (16) and then estimate η by minimizing the penalized least squares (16).

Remark 1 In Step 3, if $r_{\langle j\rangle}$ is the rounding parameter for jth predictor and its value is 0.03, then each $z_{i\langle j\rangle}$ is formed by rounding the corresponding $x_{i\langle j\rangle}$ to its nearest 0.03.

Remark 2 It is evident that the value of rounding parameter can influence the precision of approximation. The smaller the rounding parameter, the better the model estimation and the higher the computational cost.

Computational benefits: We now briefly explain why the implementation of rounding algorithm can reduce the computational loads. For example, if the rounding parameter $r = 0.01$, it is obvious that $u \leq 101$, where u denotes the number of uniquely observed values. In conclusion, using user-tunable rounding algorithm can dramatically reduce the computational burden of fitting SSANOVA models from the order of $O(n^3)$ to $O(u^3)$, where $u \ll n$.

Case study: To illustrate the benefit of the rounding algorithm, we apply the algorithm to the electroencephalography (EEG) dataset. Note that EEG is a monitoring method to record the electrical activity of the brain. It can be used to diagnose sleep disorders, epilepsy, encephalopathies, and brain death.

The dataset [33] contains 44 controls and 76 alcoholics. Each subject was repeatedly measured 10 times by using visual stimulus at a frequency of 256 Hz. This brings about $n = 10$ replications $\times 120$ subjects $\times 256$ time points $= 307,200$ observations. There are two predictors, time and group (control vs. alcoholic). We apply the cubic spline to the time effect and the nominal spline to the group effect.

After applying the model to the unrounded data, rounded data with rounding parameter $r = 0.01$ and $r = 0.05$ for time covariate, we can obtain a summary table about GCV, AIC [34], BIC [35], and running time in **Table 2**.

Based on **Table 2**, we can easily see that there are no significant difference among the GCV scores and AIC/BIC. In addition using rounding algorithm reduces 92% CPU time compared to using unrounded dataset.

	GCV	AIC	BIC	CPU time (seconds)
Unrounded data	85.9574	2,240,019	2,240,562	15.65
Rounded data with $r = 0.01$	86.6667	2,242,544	2,242,833	1.22
Rounded data with $r = 0.05$	86.7654	2,242,893	2,243,089	1.13

Table 2. Fit statistics and running time for SSANOVA models.

4. Conclusion

Smoothing spline ANOVA (SSANOVA) models are widely used in applications [11, 20, 36, 37]. In this chapter, we introduced the general framework of the SSANOVA models in Section 2. In

Section 3, we discussed the models under the big data settings. When the volume of data grows, fitting the models is computing-intensive [11]. The adaptive basis selection algorithm [14] and rounding algorithm [15] we presented can significantly reduce the computational cost.

Acknowledgements

This work is partially supported by the NIH grants R01 GM122080 and R01 GM113242; NSF grants DMS-1222718, DMS-1438957, and DMS-1228288; and NSFC grant 71331005.

Conflict of interest

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interests; and expert testimony or patent-licensing arrangements) or nonfinancial interest (such as personal or professional relationships, affiliations, knowledge, or beliefs) in the subject matter or materials discussed in this manuscript.

Appendix

In this appendix, we use two examples to illustrate how to implement smoothing spline ANOVA (SSANOVA) models in R. The gss package in R, which can be downloaded on the CRAN https://cran.r-project.org/, is utilized.

We now load the gss package:

```
library(gss)
```

Example I: Apply the smoothing spline to a simulated dataset.

Suppose that the predictor x follows a uniform distribution on $[0,1]$, and the response y is generated based on $y = 5 + 2\cos(3\pi x) + \epsilon$, where $\epsilon \sim N(0,1)$.

```
x<-runif(100);y<-5+2*cos(3*pi*x)+rnorm(x)
```

Then, fit cubic smoothing spline model:

```
cubic.fit<-ssanova(y~x)
```

To evaluate the predicted values, one uses:

```
new<-data.frame(x=seq(min(x),max(x),len=50))
```

```
est<-predict(cubic.fit,new,se=TRUE)
```

The `se.fit` parameter indicates if one can get the pointwise standard errors for the predicted values. The predicted values and Bayesian confidence interval, shown in **Figure 4**, are generated by:

```
plot(x,y,col=1)

lines(new$x,est$fit,col=2)

lines(new$x,est$fit+1.96*est$se,col=3)

lines(new$x,est$fit-1.96*est$se,col=3)
```

Example II: Apply the SSANOVA model to a real dataset.

In this example, we illustrate how to implement the SSANOVA model using the gss package. The data is from an experiment in which a single-cylinder engine is run with ethanol to see how the nox concentration `nox` in the exhaust depends on the compression ratio `comp` and the equivalence ratio `equi`. The fitted model contains two predictors (`comp` and `equi`) and one interaction term.

```
data(nox)

nox.fit <- ssanova(log10(nox)~comp*equi,data=nox)
```

The predicted values are shown in **Figure 5**.

```
x=seq(min(nox$comp),max(nox$comp),len=50)

y=seq(min(nox$equi),max(nox$equi),len=50)

temp <- function(x, y){
```

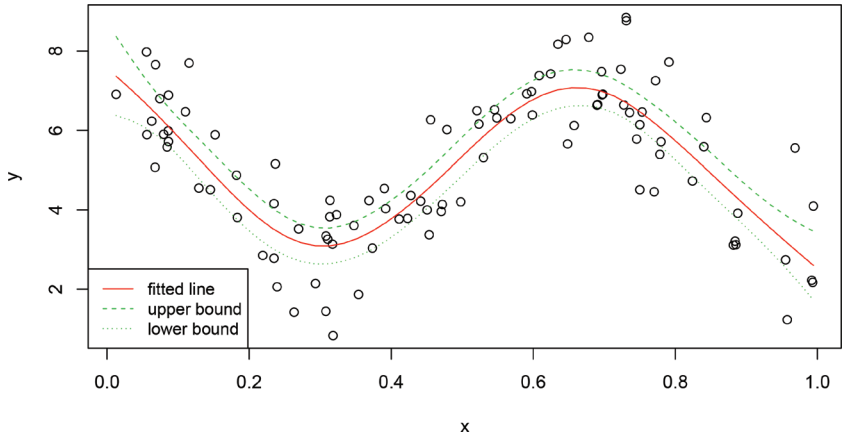

Figure 4. The solid red line represents the fitted values. The green lines represent the 95% Bayesian confidence interval. The raw data are shown as the circles.

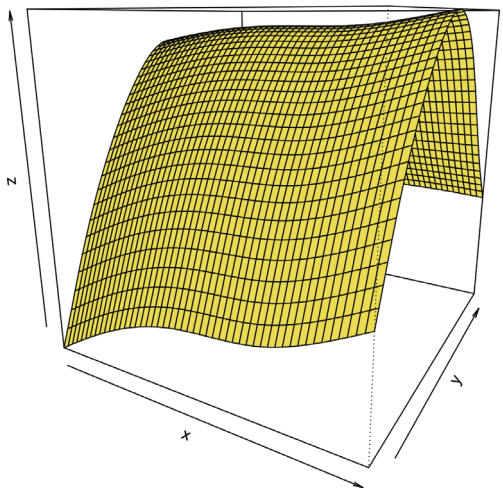

Figure 5. The x-axis, y-axis, and z-axis represent the compression ratio, the equivalence ratio, and the predicted values, respectively.

```
new=data.frame(comp=x,equi=y)
return(predict(nox.fit,new,se=FALSE))
}
z=outer(x, y, temp)
persp(x,y,z,theta = 30).
```

Author details

Jingyi Zhang, Honghe Jin, Ye Wang, Xiaoxiao Sun, Ping Ma* and Wenxuan Zhong

*Address all correspondence to: pingma@uga.edu

Department of Statistics, The University of Georgia, Athens, GA, USA

References

[1] Buja A, Hastie T, Tibshirani R. Linear smoothers and additive models. The Annals of Statistics. 1989;**17**:453-510

[2] Burman P. Estimation of generalized additive models. Journal of Multivariate Analysis. 1990;**32**(2):230-255

[3] Friedman JH, Grosse E, Stuetzle W. Multidimensional additive spline approximation. SIAM Journal on Scientific and Statistical Computing. 1983;**4**(2):291-301

[4] Hastie TJ. Generalized additive models. In: Statistical Models in S. Routledge; 2017. pp. 249-307

[5] Stone CJ. Additive regression and other nonparametric models. The Annals of Statistics. 1985;**13**:689-705

[6] Stone CJ. The dimensionality reduction principle for generalized additive models. The Annals of Statistics. 1986;**14**:590-606

[7] Barry D et al. Nonparametric bayesian regression. The Annals of Statistics. 1986;**14**(3):934-953

[8] Chen Z. Interaction Spline Models. University of Wisconsin–Madison; 1989

[9] Gu C, Wahba G. Minimizing GCV/GML scores with multiple smoothing parameters via the Newton method. SIAM Journal on Scientific and Statistical Computing. 1991;**12**(2):383-398

[10] Wahba G. Partial and Interaction Splines for the Semiparametric Estimation of Functions of Several Variables. University of Wisconsin, Department of Statistics; 1986

[11] Gu C, Smoothing Spline ANOVA. Models, Volume 297. In: Springer Science & Business Media; 2013

[12] Wahba G. Spline Models for Observational Data. SIAM; 1990

[13] Wang Y. Smoothing Splines: Methods and Applications. CRC Press; 2011

[14] Ma P, Huang JZ, Zhang N. Efficient computation of smoothing splines via adaptive basis sampling. Biometrika. 2015;**102**(3):631-645

[15] Helwig NE, Ma P. Smoothing spline ANOVA for super-large samples: Scalable computation via rounding parameters. Statistics and Its Interface, Special Issue on Statistical and Computational Theory and Methodology for Big Data. 2016;**9**:433-444

[16] Green PJ, Silverman BW. Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach. CRC Press; 1993

[17] Kimeldorf GS, Wahba G. A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. The Annals of Mathematical Statistics. 1970;**41**(2): 495-502

[18] Kimeldorf GS, Wahba G. Spline functions and stochastic processes. Sankhya: The Indian Journal of Statistics, Series A; 1970. pp. 173-180

[19] O'sullivan F, Yandell BS, Raynor WJ Jr. Automatic smoothing of regression functions in generalized linear models. Journal of the American Statistical Association. 1986;**81**(393):96-103

[20] Wahba G, Wang Y, Gu C, Klein R, Klein B. Smoothing spline ANOVA for exponential families, with application to the Wisconsin epidemiological study of diabetic retinopathy. The Annals of Statistics; 1995:1865-1895

[21] Craven P, Wahba G. Smoothing noisy data with spline functions. Numerische Mathematik. 1978;**31**(4):377-403

[22] Golub GH, Heath M, Wahba G. Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics. 1979;**21**(2):215-223

[23] Kreyszig E. Introductory Functional Analysis with Applications, Volume 1. New York: Wiley; 1989

[24] Berlinet A, Thomas-Agnan C. Reproducing Kernel Hilbert Spaces in Probability and Statistics. In: Springer Science & Business Media; 2011

[25] Aronszajn N. Theory of reproducing kernels. Transactions of the American Mathematical Society. 1950;**68**(3):337-404

[26] Abramowitz M, Stegun IA. Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables, Volume 55. Courier Corporation; 1964

[27] Hurvich CM, Simonoff JS, Tsai C-L. Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 1998;**60**(2):271-293

[28] Mallows CL. Some comments on Cp. Technometrics. 2000;**42**(1):87-94

[29] Duchon J. Splines minimizing rotation-invariant semi-norms in Sobolev spaces. Constructive Theory of Functions of Several Variables; 1977. pp. 85-100

[30] Meinguet J. Multivariate interpolation at arbitrary points made simple. Zeitschrift für Angewandte Mathematik und Physik (ZAMP). 1979;**30**(2):292-304

[31] Wahba G, Wendelberger J. Some new mathematical methods for variational objective analysis using splines and cross validation. Monthly Weather Review. 1980;**108**(8):1122-1143

[32] Ma P, Zhang N, Huang JZ, Zhong W. Adaptive basis selection for exponential family smoothing splines with application in joint modeling of multiple sequencing samples. Statistica Sinica, in press; 2017

[33] Lichman M. UCI Machine Learning Repository. 2013. URL. http://archive.ics.uci.edu/ml

[34] Akaike H. Information theory and an extension of the maximum likelihood principle. In: Parzen E, Tanabe K, Kitagawa G, editors. Selected Papers of Hirotugu Akaike, Springer Series in Statistics. New York: Springer; 1998. pp. 199-213

[35] Schwarz G. Estimating the dimension of a model. The Annals of Statistics. 1978;**6**(2):461-464. DOI: 10.1214/aos/1176344136

[36] Helwig NE, Shorter KA, Ma P, Hsiao-Wecksler ET. Smoothing spline analysis of variance models: A new tool for the analysis of cyclic biomechanical data. Journal of Biomechanics. 2016;**49**(14):3216-3222

[37] Lin X, Wahba G, Xiang D, Gao F, Klein R, Klein B. Smoothing spline ANOVA models for large data sets with Bernoulli observations and the randomized GACV. The Annals of Statistics. 2000;**28**:1570-1600