Breaking the Curse of Dimensionality

Mark Coppejans[1]
Department of Economics
Duke University
Durham NC, 27708-0097 USA

Phone: (919) 660-1804
Fax: (919) 684-8974
Email: mtc@econ.duke.edu

May 2000

[1]This and other papers by the author are available at http://www.econ.duke.edu/~mtc/papers.

ABSTRACT

This paper proposes a new nonparametric estimator for general regression functions with multiple regressors. The method used here is motivated by a remarkable result derived by Kolmogorov (1957) and later tightened by Lorentz (1966). In short, any continuous function $f(x_1, \ldots, x_d)$ has the representation $\sum_{k=1}^{2d+1} \tilde{g}(\lambda_1 \tilde{\phi}_k(x_1) + \cdots + \lambda_d \tilde{\phi}_k(x_d))$, where $\tilde{g}(\cdot)$ is a continuous function, $\tilde{\phi}_k(\cdot)$, $k = 1, \ldots, 2d+1$, is Lipschitz of order one and strictly increasing, and λ_j, $j = 1, \ldots, d$, is some constant. Generalizing this result, we propose the following estimator, $\sum_{k=1}^{2d+1} g_k(\lambda_{k,1}\phi_k(x_1) + \cdots + \lambda_{k,d}\phi_k(x_d))$, where both $g_k(\cdot)$ and $\phi_k(\cdot)$ are twice continuously differentiable and $\phi_k(\cdot)$ is non-decreasing. These functions are estimated using regression cubic B-splines, which have excellent numerical properties. One of the main contributions of this paper is that we develop a method for imposing monotonicity on the cubic B-splines, *a priori*, such that the estimator is dense in the set of all monotonic cubic B-splines. The method requires only $2(r+1)+1$ restrictions per each $\phi_k(\cdot)$, where r is the number of interior knots. Rates of convergence in L_2 are the same as the optimal rate for the one-dimensional case. A simulation experiment shows that the estimator works well when optimization is performed by using the back-fitting algorithm. The monotonic restriction has many other applications besides the one presented here, such as estimating a demand function. With only $r+2$ more constraints, it is also possible to impose concavity.

KEY WORDS: Curse of Dimensionality, Nonparametric Regression, B-splines, Monotonicity Constraints.

1 Introduction

In economics, there is a great need for a general method that estimates the conditional expectation of some dependent variable given a set of regressors. Economic theory provides insight into why this expectation is of interest, however its functional form is often left unspecified. Therefore, in order to meaningfully capture this relationship, we would like our estimator to "reasonably" flexible.

Because of its simplicity, practitioners usually restrict the conditional expectation to lie in some family of parametric models, such as linearity, but this form is very inflexible, which may lead to an incorrect analysis. As an alternative, multivariate kernels or B-spline regressions seem to be reasonable because they can consistently estimate a wide class of functions such as the class of twice continuously differentiable functions. This is more general than the standard parametric model, however there are serious problems with nonparametrically estimating high dimensional functions. From a theoretical standpoint, the rate of convergence decreases as the dimension increases—this is the curse of dimensionality. In addition, Silverman (1986; Table 4.2) provides some unsatisfactory findings on the small sample performance of a high-dimensional kernel estimator.

The main problem with the standard multivariate nonparametric approach is that for most moderate sample sizes, the estimator lies in class of functions which is too large. In other words, it is permitted to be too flexible. This might seem paradoxical because the increase in flexibility is usually why a nonparametric method is chosen over a parametric one. To overcome these potential shortcomings, the ideal estimator in many applications is one that allows for great flexibility while limiting the decrease in performance as the number of regressors increase. Such an estimator is proposed here.

To be more precise, let $y_i = f(x_{1,i}, \ldots, x_{d,i}) + \epsilon_i$, $i = 1, \ldots, n$, where $\{x_i', \epsilon\}'$ is iid, $x_i = \{x_{1,i}, x_{2,i}, \ldots, x_{d,i}\}'$, $y_i, x_{j,i}, \epsilon_i \in \Re$, $j = 1, \ldots, d$, $E[\epsilon|x] = 0$, $f(\cdot) \in C^q$, and C^q is the set of functions with q continuous derivatives defined in the usual way. Then under standard assumptions, rates of convergence of order $n^{-q/(2q+d)}$ are obtainable for the estimates of f. However, as stated above, the space C^q is typically too large to get reasonable estimates for moderate sample sizes. If we somehow restrict the class of functions so that it

resembles a set of one-dimensional functions, then many of the problems associated with the high-dimensional case disappear. Two popular examples, which will be outlined below, are additive models (Hastie and Tibshirani, 1986) and projection pursuit models (Friedman and Stuetzle, 1981).

This one-dimensional approach is taken here. To justify the estimator, an amazing result presented in Lorentz *et al.* (1996; p. 553), which is an extension of the work done by Kolmogorov (1957) and Lorentz (1966), is re-produced below.

THEOREM 1 Let $I = [0,1]$ and $I^d = [0,1]^d$ be the d-dimensional unit cube. Then there exist d constants $\lambda_j > 0$, $j = 1, \ldots, d$, $\sum_{j=1}^{d} \lambda_j \leq 1$, and $2d+1$ continuous strictly increasing functions $\tilde{\phi}_k$, $k = 1, \ldots, 2d+1$, which map I into itself, with the property that each function $f \in C(I^d)$ has a representation

$$f(x_1, \ldots, x_d) = \sum_{k=1}^{2d+1} \tilde{g} \left(\lambda_1 \tilde{\phi}_k(x_1) + \cdots \lambda_d \tilde{\phi}_k(x_d) \right), \tag{1}$$

with some $\tilde{g} \in C(I)$.

In words, the result states that functions of several variables can be represented as superpositions (e.g. functions of functions) of functions of one variable. Surprisingly, the $\tilde{\phi}_k$ are fixed functions (they do not depend on f) and Lipschitz of order one. It is crucial to note that there are a finite number, $2d+2$, of functions in (1) and that all of them are univariate.

Even though this is a remarkable theoretical result, it is of little direct use to practitioners because $\tilde{g}$ and $\tilde{\phi}_k$, given their lack of smoothness, are difficult to estimate. To overcome this difficulty and to add more flexibility, a variant of (1) is proposed,

$$\sum_{k=1}^{2d+1} g_k \left(\lambda_{1,k} \phi_k(x_1) + \cdots + \lambda_{d,k} \phi_k(x_d) \right), \tag{2}$$

where $g_k, \phi_k \in C^2(I)$ and ϕ_k is non-decreasing. We will estimate both g_k and ϕ_k using regression cubic B-splines, which have excellent numerical properties. The estimator will be termed *Lin-Supe*, where the "Supe" stands for superposition and the "Lin" stands for linear, highlighting the fact that the interior piece, in terms of the $\phi_k(\cdot)$, is linear.

For both theoretical and computational reasons, it is important to impose the restriction that ϕ_k is monotone, *a priori*. As Lorentz (1966) observed, if $\tilde{\phi}_k$ is not strictly increasing,

then there are two points in I^d such that $\lambda_{1,k}\tilde{\phi}_k(x_1) + \cdots + \lambda_{d,k}\tilde{\phi}_k(x_d)$, $k = 1, \ldots, 2d+1$, achieves the same value. This implies that the right hand side of (1) coincides at these points, and hence the representation on the left hand side is impossible for some functions. By the same reasoning, non-monotonicity would make the model difficult to estimate, creating a situation where there would be distinct choices of ϕ_k and g_k such that the objective function would not change.

We derive a simple method for imposing monotonicity. This is a new result and one of the main contributions of the paper. Previous results in the literature with regard to imposing monotonicity suffer from the fact that they are either not easy to implement (e.g. Wright and Wegman, 1980) and/or that they are not dense in the set of all monotonic cubic B-splines (e.g. Schumaker, 1983; Ramsay, 1988; Chen and Shen, 1998). In the Conclusion of this paper, a discussion will be given on how the methodology developed here can be extended to a variety of other important models, such as ones that impose concavity.

Under appropriate assumptions, it is shown that Lin-Supe converges in L_2 at a rate of $n^{-3/7}$. This is the optimal rate of convergence for three times continuously differentiable univariate functions. Because the rate does not depend on d, it is in this sense that we are claiming that we are "breaking" the curse of dimensionality.

The estimator defined in (2) may be also viewed as a generalization of additive and projection pursuit models, arguably two of the most common methods for circumventing the curse of dimensionality to date. An example of the former is $\sum_{k=1}^{d} g_k(x_k)$, and an example of the latter is $\sum_{k=1}^{p} g_k(\lambda_{1,k}x_1 + \cdots \lambda_{d,k}x_d)$ for some integer p. A potential pitfall for the additive model is that there is no interaction across the regressors. There are ways to arbitrarily overcome this, but then the rates of convergence decrease. In the projection pursuit case, if p is large, then there seems little to gain over standard multivariate nonparametric techniques (since the number of estimated parameters is quite large), and for $p \leq 2d+1$, projection pursuit falls under the class of models studied here.

Another popular estimator for circumventing the curse of dimensionality is neural networks (e.g. Barron, 1993). Under suitable assumptions, Chen and Shen (1998) show that a lower bound for the rate of convergence in L_2 is $n^{-1/4}$, which is quite slow, however.

The outline of this paper is as follows. We will describe the estimator in Section 2 and

then provide results on the rates of convergence. A simulation study is given in Section 3. Possible extensions are discussed in the last section.

2 Estimator

The key to constructing the estimator described in (2) is in the way the monotonicity restrictions are imposed on the ϕ_k's. With this in place, the construction is just a sum of superpositions of B-splines, which is straightforward to form. To denote a generic regressor out of the set of d possible choices, we will often use x, where it is understood that $x \in \Re$. When we need to distinguish this from the set of d regressors, we will sometimes use $\vec{x}$, $\vec{x} \in \Re^d$.

To begin, we will show that it is trivial to impose the monotonicity restriction on a polynomial of degree 3, $p(x) = \gamma_0 + \gamma_1 x + \gamma_2 x^2 + \gamma_3 x^3$, $x \in [a, b]$. The polynomial is increasing on $[a, b]$ if we force its derivative, $p'(x) = \gamma_1 + 2\gamma_2 x + 3\gamma_3 x^2$, to be positive for $x \in [a, b]$. Note that $p'(x)$ is minimized at $x = -\gamma_2/(3\gamma_3)$ with a minimum value of $\gamma_1 - \gamma_2^2/(3\gamma_3)$ if $\gamma_3 > 0$. Then the following three restrictions ensure monotonicity,

$$1\left(3\gamma_3 a \leq -\gamma_2 \leq 3\gamma_3 b,\ \gamma_3 > 0\right)\left(\gamma_1 - \frac{\gamma_2^2}{3\gamma_3}\right) \geq 0, \quad p'(a) \geq 0, \quad p'(b) \geq 0,$$

where $1(\cdot)$ denotes the indicator function. The last two restrictions, $p'(a), p'(b) \geq 0$, are only necessary when $\gamma_3 \leq 0$. So that the condition $\gamma_1 - \gamma_2^2/(3\gamma_3) \geq 0$ if $\gamma_3 > 0$ takes hold only when the minimum lies in $[a, b]$, we include an indicator over the region $3\gamma_3 a \leq -\gamma_2 \leq 3\gamma_3 b$. This permits the polynomial to be non-monotonic over the region $(-\infty, a)$ and (b, ∞). This is done so that these restrictions do not exclude any non-decreasing cubic polynomial on $x \in [a, b]$. In other words, it is easy to construct a cubic polynomial which is non-decreasing on $[a, b]$ and is decreasing somewhere on $(-\infty, a) \cup (b, \infty)$, and we do not want to exclude these types of polynomials from the set of all possible choices.

It is well known (Eubank, 1988) that a cubic spline, $s(x), x \in [a, b]$, with r interior knots, ξ_l, $l = 1, \ldots, r$, may be written as a sum of piecewise polynomials of degree 3,

$$s(x) = \sum_{v=0}^{3} \gamma_{v,l} x^l, \quad x \in [\xi_{l-1}, \xi_l),\ l = 1, \ldots, r, \quad \text{or} \quad x \in [\xi_r, \xi_{r+1}],\ l = r + 1,$$

such that $s(\xi_{l-1}) = s(\xi_l)$, $s'(\xi_{l-1}) = s'(\xi_l)$, $s''(\xi_{l-1}) = s''(\xi_l)$, $l = 2, \ldots, r$, and $\xi_0 = a, \xi_{r+1} = b$. Notice that $s(x)$ is twice continuously differentiable and its third derivative is a step-function.

Schumaker (1981) has shown that as the distance between adjacent knots tends to zero, there exists a corresponding cubic spline that will approximate any continuous function and its first three continuous derivatives, if they exist, arbitrarily well under the strong norm on $[a, b]$.[1] To ensure that $s(x)$ is also monotonic, we can impose the analogous restrictions as on the cubic polynomial,

$$\begin{aligned} 1\left(3\gamma_{3,l}\xi_{l-1} \leq -\gamma_{2,l} \leq 3\gamma_{3,l}\xi_l,\ \gamma_{3,l} > 0\right)\left(\gamma_{1,l} - \frac{\gamma_{2,l}^2}{3\gamma_{3,l}}\right) &\geq 0, \quad l = 1, \ldots, r+1, \\ \gamma_{1,l} + 2\gamma_{2,l}\xi_{l-1} + 3\gamma_{3,l}\xi_{l-1}^2 &\geq 0, \quad l = 1, \ldots, r+1, \\ \gamma_{1,r+1} + 2\gamma_{2,r+1}\xi_{r+1} + 3\gamma_{3,r+1}\xi_{r+1}^2 &\geq 0. \end{aligned} \tag{3}$$

This implies that the total number of restrictions is now $2(r+1)+1$. By the same argument as in the case of a cubic polynomial, there exists no other monotonic cubic spline on $[a, b]$ which does not satisfy these constraints given that it has the same placement of knots.

Theoretically, we could estimate the ϕ_k's using the cubic spline and the monotonicity constraints above. However, regression splines, even without these additional constraints, have poor numerical properties, most notably their tendency to be collinear. On the other hand, B-splines have superb numerical properties. They also span the same space as splines, which implies that they have identical approximating ability, theoretically. A brief description of B-splines is given below.[2]

The basis for the B-splines can be derived recursively. Denote 6 more knots as $\xi_{-3} = \xi_{-2} = \xi_{-1} = a$ and $\xi_{r+2} = \xi_{r+3} = \xi_{r+4} = b$. Then the basis for a B-spline of order 4, $\{B_{l,4}\}_{l=-3}^{r}$, is defined as

$$\begin{aligned} B_{l,m}(x) &= \frac{x - \xi_l}{\xi_{l+m-1} - \xi_l} B_{l,m-1}(x) + \frac{\xi_{l+m} - x}{\xi_{l+m} - \xi_{l+1}} B_{l+1,m-1}(x), \quad \text{if } m = 2, 3, 4, \\ B_{l,1} &= \begin{cases} 1, & \text{if } x \in [\xi_l, \xi_{l+1}) \\ 0, & \text{if otherwise.} \end{cases} \end{aligned}$$

Relevant properties of the basis are that $B_{l,4}(x) = 0$ if $x \notin [\xi_l, \xi_{l+4}]$, $\sum_{l=-3}^{r+4} B_{l,4}(x) = 1$, $B_{l,4}(x) \geq 0$ for all x, $l = -3, \ldots, r+4$, and $B_{-3,4}(a) = B_{r,4}(b) = 1$. The B-spline is defined

[1] For example, suppose that $g(x)$ is three times continuously differentiable on $[a, b]$ with $\sup_x |g(x)| \leq C^{(0)}$, $\sup_x |g'(x)| \leq C^{(1)}$, $\sup_x |g''(x)| \leq C^{(2)}$, and $\sup_x |g'''(x)| \leq C^{(3)}$, where $C^{(q)}$, $q = 0, 1, 2, 3$, are some constants. Then there exists a cubic spline with r uniformly placed knots, $s(x)$, such that $\sup_x |s(x) - g(x)| = O(r^{-3})$, $\sup_x |s'(x) - g'(x)| = O(r^{-2})$, $\sup_x |s''(x) - g''(x)| = O(r^{-1})$, and $\sup_x |s'''(x) - g'''(x)| \to 0$ as $r \to \infty$.

[2] See Eubank (1988) or de Boor (1978) for a more detailed introduction on B-splines.

as

$$b(x) = \sum_{l=1}^{r+4} \beta_l B_{l-4,4}(x), \tag{4}$$

for some set of coefficients, β_l, $l = 1, \ldots, r+4$. Even though it is not obvious from the recursion formula, $b(x)$ is twice continuously differentiable and its third derivative is a step-function. This is necessary since it spans the same space as the cubic spline.

Unfortunately, imposing the monotonicity restrictions on $b(x)$ is no longer as straightforward as in the case of the cubic spline. The goal is to first derive a relationship between $s(x)$, the cubic spline, and $b(x)$, the B-spline. De Boor (1978) has shown that for $x \in [\xi_l, \xi_{l+1}]$, $B_{v,4}(x) = \alpha_{0,l,v} + \alpha_{1,l,v}x + \alpha_{2,l,v}x^2 + \alpha_{3,l,v}x^3$, $v = l-3, l-2, l-1$, or l, where the α's are coefficients depending only on the placement of the knots. These coefficients are easily obtained by least squares once $B_{v,4}(x)$ has been constructed, so throughout the remainder of the article, we will treat them as fixed. The relationship trivially holds elsewhere since $B_{v,4}(x) = 0$ for $v \neq l-3, l-2, l-1$, or l. Then for $x \in [\xi_l, \xi_{l+1}]$,

$$\begin{aligned}
b(x) &= \beta_{l+1}B_{l-3,4}(x) + \beta_{l+2}B_{l-2,4}(x) + \beta_{l+3}B_{l-1,4}(x) + \beta_{l+4}B_{l,4}(x) \\
&= (\beta_{l+1}\alpha_{0,l,l-3} + \beta_{l+2}\alpha_{0,l,l-2} + \beta_{l+3}\alpha_{0,l,l-1} + \beta_{l+4}\alpha_{0,l,l}) \\
&\quad (\beta_{l+1}\alpha_{1,l,l-3} + \beta_{l+2}\alpha_{1,l,l-2} + \beta_{l+3}\alpha_{1,l,l-1} + \beta_{l+4}\alpha_{1,l,l})\, x \\
&\quad (\beta_{l+1}\alpha_{2,l,l-3} + \beta_{l+2}\alpha_{2,l,l-2} + \beta_{l+3}\alpha_{2,l,l-1} + \beta_{l+4}\alpha_{2,l,l})\, x^2 \\
&\quad (\beta_{l+1}\alpha_{3,l,l-3} + \beta_{l+2}\alpha_{3,l,l-2} + \beta_{l+3}\alpha_{3,l,l-1} + \beta_{l+4}\alpha_{3,l,l})\, x^3 \\
&\equiv \delta_{0,l} + \delta_{1,l}x + \delta_{2,l}x^2 + \delta_{3,l}x^3.
\end{aligned} \tag{5}$$

Hence we can rewrite the B-spline as a cubic piecewise polynomial with

$$\delta_{v,l} = \sum_{p=1}^{4} \beta_{l+p}\alpha_{v,l,l-4+p}, \qquad v = 0,1,2,3, \qquad l = 1, \ldots, r+4. \tag{6}$$

This implies that we can use (3), with the δ's in place of the γ's, to impose monotonicity,

$$\begin{aligned}
1\,(3\delta_{3,l}\xi_{l-1} \leq -\delta_{2,l} \leq 3\delta_{3,l}\xi_l,\ \delta_{3,l} > 0)\left(\delta_{1,l} - \frac{\delta_{2,l}^2}{3\delta_{3,l}}\right) &\geq 0, \quad l = 1, \ldots, r+1, \\
\delta_{1,l} + 2\delta_{2,l}\xi_{l-1} + 3\delta_{3,l}\xi_{l-1}^2 &\geq 0, \quad l = 1, \ldots, r+1, \\
\delta_{1,r+1} + 2\delta_{2,r+1}\xi_{r+1} + 3\delta_{3,r+1}\xi_{r+1}^2 &\geq 0.
\end{aligned} \tag{7}$$

We have just proved the following result.

THEOREM 2 The cubic B-spline defined in (4) subject to the constraints in (7) is non-decreasing for $x \in [a, b]$. In addition, all non-decreasing cubic B-splines defined as in (4) satisfy (7) for $x \in [a, b]$.

In the case of B-splines of order 3, the monotonic restriction (which have the analogous property as in Theorem 2) takes a simpler form: $\delta_{v,l} \leq \delta_{v,l+1}$, $l = 1, \ldots, r+3$ (see de Boor, 1988, p. 163). The B-spline of order 4 is used here instead for several reasons. It is the most widely used of the splines, being often viewed as the paradigm case. We might also like to impose restrictions on the second derivative as well, and in such cases, an estimator that is twice continuously differentiable is preferable. In addition, faster rates of convergence are possible, as will be shown below, with an additional smoothness assumption. Lastly, many optimizing programs work better if the estimator is twice continuously differentiable.

We are now in a position to describe the general model. We will begin by restricting the class of functions to which the underlying function, f, can belong.

DEFINITION 1 Define the set of functions, $\mathcal{F}$, as

$$\left\{ \sum_{k=1}^{2d+1} g_k \left[\lambda_{1,k} \phi_k(x_1) + \cdots \lambda_{d,k} \phi_k(x_d) \right] \right\}$$

such that $g_k, \phi_k \in C^q([a, b])$, $1 \leq q \leq 3$,

$$\begin{aligned}
\sup_{a \leq x \leq b} |g_k^{(v)}(x)| &< C_g^{(v)} < \infty, \qquad v = 0, 1, \ldots, q, \qquad k = 1, \ldots, 2d+1, \\
\sup_{a \leq x \leq b} |\phi_k^{(v)}(x)| &< C_\phi^{(v)} < \infty, \qquad v = 1, \ldots, q, \qquad k = 1, \ldots, 2d+1, \\
\inf_{a \leq x \leq b} |\phi_k^{(1)}(x)| &> 0, \qquad k = 1, \ldots, 2d+1,
\end{aligned}$$

where $C_g^{(v)}$, $v = 0, 1, \ldots, q$, $C_\phi^{(v)}$, $v = 1, \ldots, q$, are constants, $h^{(0)}(x) \equiv h(x)$, $h^{(v)}(x) \equiv \partial^v h(x) / \partial x^v, v \geq 1$. In addition, $\phi_k(a) = a$, $\phi_k(b) = b$, $\sum_{j=1}^d \lambda_{j,k} = 1$, and $\lambda_{j,k} \geq 0$, $j = 1, \ldots, d$, $k = 1, \ldots, 2d+1$.

ASSUMPTION 1 The sequence $\{x_i', \epsilon_i\}_{i=1}^n$ is iid. Let $y_i = f(x_i) + \epsilon_i$, $E[\epsilon | x] = 0$, $f \in \mathcal{F}$, and $\epsilon_i \in \Re$. In addition, each regressor, $x_{j,i}$, $j = 1, \ldots, d$, has bounded support with bounds denoted by a and b, $a < b$.

The assumption that each regressor has the same support is made only for notational simplicity. It is innocuous since a rescaling can always achieve this, as long as the regressors have finite support. Estimation in the case of non-bounded support can be handled in a similar fashion as in Fenton and Gallant (1996); however, it is no longer clear if the general setup in (1) is still valid without additional structure imposed on $f(\cdot)$.

Setting $\phi_k(a) = a$ and $\phi_k(b) = b$ is made so that the functions map $[a, b]$ into itself. Clearly this is needed to separately identify $g_k(\cdot)$ and $\phi_k(\cdot)$. The restriction $\sum_{j=1}^{d} \lambda_{j,k} = 1$ is imposed instead of $\sum_{j=1}^{d} \lambda_{j,k} \leq 1$ for simplicity.

Note that $\mathcal{F} \subset C^q([a, b]^d)$, where $C^q([a, b]^d)$ is the set of functions with q continuous derivatives. In the case of most standard nonparametric regression estimators, such as kernels, convergence is with respect to functions in $C^q([a, b]^d)$, but as we noted in the Introduction, the cost is that the rates of convergence depend on d. Stone (1982) has shown that the convergence rate of $n^{-q/(2q+d)}$ is optimal. Therefore, to circumvent the curse of dimensionality, we must restrict the underlying class of functions in a non-trivial way. Additive models, projection pursuit models, and neural nets, for example, must also restrict the class of underlying functions in order to obtain rates that dominate Stone's (1982). If the underlying function is in fact in $C^q([a, b]^d)$ but not in $\mathcal{F}$, then there will be a bias which is not necessarily asymptotically negligible. In the Conclusion, we construct a method that will force this bias to tend to zero in probability.

The next assumption will make estimation easier, and it can be relaxed by using the methods described in Shen and Wong (1994). It is used to bound the *metric entropy*, which is the log of the number of ϵ balls it takes to cover the parameter space under the strong norm. The metric entropy is a measure of the size of the parameter space.[3]

ASSUMPTION 2 The bounds, $C_g^{(0)}$ and $C_g^{(1)}$, in Definition 1 are known.

As stated above, both the g_k's and the ϕ_k's will be estimated using B-splines, where the latter is restricted to be monotonic as described in (7). Again for notational simplicity, suppose that the number of knots, r, is the same across all splines and that the knots are placed uniformly on $[a, b]$. The parameter space, Ω_n, is defined below, and it is formally

[3]See Kolmogorov and Tihormirov (1961) for numerous examples.

called a *sieve*. Its dependence on n is explicitly stated since the number of parameters will increase with the sample size. The estimators for $\phi_k(x)$ and $g_k(z)$, subject to Ω_n, are denoted as

$$B\phi_k(x) = \sum_{l=1}^{r+4} \beta_{l,k,\phi} B_{l-4,4,k}(x), \tag{8}$$

$$Bg_k(z) = \sum_{l=1}^{r+4} \beta_{l,k,g} B_{l-4,4,k}(z), \tag{9}$$

where $x \in [a,b]$ and $z = B\phi_k(x) \in [a,b]$, $k = 1, \ldots, 2d+1$. Putting equations (8) and (9) together, the unconstrained estimator for functions in $\mathcal{F}$ is

$$B_n(\vec{x}) = \sum_{k=1}^{2d+1} \left\{ \sum_{l=1}^{r+4} \beta_{l,k,g} B_{l-4,4,k} \left[\sum_{j=1}^{d} \lambda_{j,k} \left(\sum_{l'=1}^{r+4} \beta_{l',k,\phi} B_{l'-4,4,k}(x_j) \right) \right] \right\}. \tag{10}$$

To impose the monotonicity constraints, we use the results from (5) and (6) to rewrite $B\phi_k(x)$ as a cubic polynomial,

$$\begin{aligned} B\phi_k(x) &= \delta_{0,l,k} + \delta_{1,l,k} x + \delta_{2,l,k} x^2 + \delta_{3,l,k} x^3, \qquad x \in [\xi_l, \xi_{l+1}], \\ \delta_{v,l,k} &= \sum_{p=1}^{4} \beta_{l+p,k,\phi} \alpha_{v,l,l-4+p}, \qquad v = 0,1,2,3. \end{aligned} \tag{11}$$

The following definition formally defines the constrained estimator.

DEFINITION 2 The parameter space, Ω_n, is defined as the set

$$\left\{ r, \{\beta_{l,k,\phi}, \beta_{l,k,g}\}_{l=1,k=1}^{r+4,2d+1}, \{\lambda_{l,k}\}_{l=1,k=1}^{r,2d+1}, B_n(\vec{x}) \right\}$$

such that

$$\begin{aligned} \beta_{1,k,\phi} &= a, \qquad k = 1, \ldots, 2d+1, \\ \beta_{r+4,k,\phi} &= b, \qquad k = 1, \ldots, 2d+1, \\ \sum_{j=1}^{r} \lambda_{j,k} &= 1, \qquad k = 1, \ldots, 2d+1, \\ \lambda_{j,k} &\geq 0, \qquad j = 1, \ldots, r, \qquad k = 1, \ldots, 2d+1, \\ \sup_{a \leq x \leq b} |Bg_k(x)| &\leq \tilde{C}_g^{(0)}, \qquad k = 1, \ldots, 2d+1, \\ \sup_{a \leq x \leq b} |\partial Bg_k(x)/\partial x| &\leq \tilde{C}_g^{(1)}, \qquad k = 1, \ldots, 2d+1, \end{aligned}$$

where the β's are as in equations (8) and (9), the λ's and $B_n(\vec{x})$ are as in (10), $Bg_k(x)$ is as in (8), $\tilde{C}_g^{(0)} \geq C_g^{(0)}$, $\tilde{C}_g^{(1)} \geq C_g^{(1)}$, and $C_g^{(0)}$ and $C_g^{(1)}$ are as in Definition 1. To ensure that $B\phi_k(x)$, $k = 1, \ldots, 2d+1$, is non-decreasing, we also require that

$$\begin{aligned} 1\left(3\delta_{3,l,k}\xi_{l-1} \leq -\delta_{2,l,k} \leq 3\delta_{3,l,k}\xi_l,\ \delta_{3,l,k} > 0\right)\left(\delta_{1,l,k} - \frac{\delta_{2,l,k}^2}{3\delta_{3,l,k}}\right) &\geq 0, \qquad l = 1, \ldots, r+1, \\ \delta_{1,l,k} + 2\delta_{2,l,k}\xi_{l-1} + 3\delta_{3,l,k}\xi_{l-1}^2 &\geq 0, \qquad l = 1, \ldots, r+1, \\ \delta_{1,r+1,k} + 2\delta_{2,r+1,k}\xi_{r+1} + 3\delta_{3,r+1,k}\xi_{r+1}^2 &\geq 0, \end{aligned}$$

where the δ's are as in equation (11).

The parameter space Ω_n is constructed so that $f \in \lim_{\{n\to\infty\}} \Omega_n$ given the conditions in Assumption 1. The dependence of Ω_n on n comes through r, which itself depends on n even though it is not explicitly stated here. The rate at which r increases with n is given in Theorem 3 below. The bounds on $Bg_k(x)$ and its derivative are used to bound the metric entropy of the parameter space, Ω_n; without it, the parameter space may be too large (e.g. too "wiggly") and as a result, consistency may not be obtainable. Since $B_{-3,4,k}(a) = 1, B_{v,4,k}(a) = 0, v \neq -3$ and $B_{r,4,k}(b) = 1, B_{v,4,k}(a) = 0, v \neq r$, the restrictions $\beta_{1,k,\phi} = a$, $\beta_{r+4,k,\phi} = b$ and the restrictions on the δ's ensure that $B\phi_k : [a,b] \to [a,b]$ is a monotonic mapping with $B\phi_k(a) = a$ and $B\phi_k(b) = b$.

An element of Ω_n is denoted as ω_n, $\omega_n \in \Omega_n$, which is itself a twice continuously differentiable function, $\omega_n \in C^2([a,b]^d)$. Likewise, $\omega_0 = f$, where f is as in Assumption 1. Optimization is performed by minimizing the sum of squares,

$$\hat{\omega}_n \ = \ \min_{\{\omega_n \in \Omega_n\}} \frac{1}{n}\sum_{i=1}^{n} \left[y_i - \omega_n(x_i)\right]^2 . \tag{12}$$

An algorithm for the actual implementation is discussed in the next section. Rates of convergence are given in the following theorem.

THEOREM 3 Suppose that Assumptions 1 and 2 hold. Let $\hat{\omega}_n$ be as in (12) subject to Ω_n given in Definition 2. Set $r = O(n^{1/(2q+1)})$, where q is as in Definition 1. Then

$$\rho(\hat{\omega}_n, f) \ = \ O_P\left(n^{-\frac{q}{2q+1}}\right),$$

where $\rho(\hat{\omega}_n, f) = \{E[\hat{\omega}_n(\vec{x}) - f(\vec{x})]^2\}^{1/2}$.

Proof The proof follows by applying the results of Shen and Wong (1994) Example 3 to their Theorem 1. They derived similar results for the standard regression B-spline, and we have to extend their case to that of superpositions of B-splines. Note that many of the bounds below are with respect to the strong norm rather than L_2. The reason is that the strong norm is easier to work with for theses specific calculations. Clearly (in their notation) $\alpha = 1$ and $\beta = 1$ as in their derivations in Example 3. The metric entropy for the individual B-spline estimators, $Bg_k(\cdot)$ and $B\phi_k(\cdot)$, is of order $r\log(1/\epsilon)$ because both estimators are bounded by $\tilde{C}_g^{(0)}$ and $\max\{|a|,|b|\}$, respectively, which implies that the analogous B-spline coefficients, $\beta_{l,k,g}$ and $\beta_{l,k,\phi}$, $l = 1,\ldots,r+4$, can not be greater than this bound, in absolute value, because $0 \leq B_{l-4,4,k}(x) \leq 1$ and $\sum_{l=-3}^{r+4} B_{l,4,k}(x) = 1$ for all x. By a mean value expansion, since the derivative of $B\phi_k(\cdot)$ is bounded by $\tilde{C}_g^{(1)}$, the metric entropy of $B_n(\vec{x})$ is also of order $r\log(1/\epsilon)$. Schumaker (1981, Corollary 6.21) has shown that the approximation error, given the conditions in Definition 1, for a qth continuously differentiable function is of order r^{-q} under the strong norm. In our setup, this corresponds to the approximation of some $B\phi_k(\cdot)$ and $Bg_k(\cdot)$ to $\phi_k(\cdot)$ and $g_k(\cdot)$, respectively. Schumaker also showed that the first derivative of the B-spline converges uniformly under the strong norm. Hence the monotonicity imposed on the $B\phi_k(\cdot)$'s does not affect the approximation error for large r. The sieve approximation error of $B_n(\vec{x})$ is also of order r^{-q}, which follows from a mean value expansion similar to the one used in the metric entropy calculation. □

This is the optimal rate of convergence for the univariate case. In this sense, we are circumventing the curse of dimensionality because our rates do not depend on d. The rate at which r increases, $O(n^{1/(2q+1)})$, is relatively slow, and it is due to the good approximating ability of the B-splines. For any given r and d, the total number of parameters, subject to the restrictions imposed on Ω_n, is $p = (2r+d+5)(2d+1)$. In terms of n, the number of parameters is of order $n^{1/(2q+1)}$, which is the same as in the univariate case. As a comparison, standard multivariate B-splines techniques require an order of r^d parameters, and in terms of n, the number of parameters is of order $n^{d^2/(2q+d)}$, which for moderate d and n, is quite large.[4] Given the large number, it is not surprising that in general the coefficients are not estimated

[4]The relationship between r and n is derived by equating the optimal rate of convergence, $n^{-q/(2q+d)}$, with the approximation error, $r^{-q/d}$ given in Schumaker(1981, Theorem 12.7).

very precisely. This explains why the variance component (of the bias^2 and variance tradeoff) is typically large for general high-dimensional nonparametric estimators.

3 Simulations

Given that our estimator is nonstandard, the two main goals of this section is to show that it is implementable and that its performance compares well with those of the standard kernel regression for a moderately large sample size. The kernel is a natural comparison since its use, especially for its theoretical properties, is ubiquitous throughout economics and statistics, even when the number of regressors is large.

An appropriate place to start is to test how well the monotonicity restrictions work in terms of a nonlinear optimizer. To do this, the following model is selected,

Model 1

$$y = \sin\left(\frac{1}{d}\sum_{j=1}^{d}\exp(x_j)\right) + \epsilon,$$

where throughout this section, $x_j, j = 1, \ldots, d$, is uniformly distributed on $[0, \pi]$ and ϵ is distributed as a standard normal. These distributional choices on the regressors and errors are standard, and in doing so, the main focus here can remain of the estimates of the regression function. A graph of the regression function in the case of two regressors is given in Figure 1. The model is very oscillatory, and it is an unlikely representative of most economic relationships. Nonetheless, the estimator proposed here should do well under these circumstances with k in (2) held fixed at $k^* = 1$, $g_1(\cdot) = \sin(\cdot)$, $\phi_1(\cdot) = \exp(\cdot)$, and $\lambda_{j,1} = 1/d, j = 1, \ldots, d$, and thereby being an adequate test. A moderately large sample size, at least for most economic settings, is chosen here to be $n = 10,000$. Estimation is performed with $d = 2, 3$, and 5 regressors. In terms of nonparametrics, five regressors is typically considered large with only 10,000 observations. On the other hand, the kernel estimator should do well for the case $d = 2$. Setting $d = 3$ can be viewed as an intermediate choice.

Optimization is performed using NPSOL by Gill *et al.* (1986) because it allows for constrained optimization.[5] The optimizer performs better if the constraints are smooth, but the constraint $\gamma_{1,l} - \gamma_{2,l}^2/(3\gamma_{3,l}) \geq 0$ is only binding if $3\gamma_{3,l}\xi_{l-1} \leq -\gamma_{2,l}$ or $3\gamma_{3,l}\xi_l \geq -\gamma_{2,l}$ and $\gamma_{3,l} > 0$. To smooth this out, we construct a strictly increasing fourth order B-spline, $BI(\xi)$, on $[(\xi_{l-2} + \xi_{l-1})/2, \xi_{l-1}]$ with $BI((\xi_{l-2} + \xi_{l-1})/2) = 0$ and $BI(\xi_{l-1}) = 1$ such that for any ξ in this interval, the constraint becomes $BI(\xi)[\gamma_{1,l} - \gamma_{2,l}^2/(3\gamma_{3,l})] \geq 0$. An analogous strategy is constructed on $[\xi_l, (\xi_l + \xi_{l+1})/2]$ with a strictly decreasing B-spline, $BD(\xi)$, such that $BD(\xi_l) = 1$ and $BD((\xi_l + \xi_{l+1})/2) = 0$.[6] The other constraints do not pose this problem. A good optimizer should work well, nonetheless, even without this added smoothness, as long as the initial values are chosen well. This implies that in many cases, one will need to search over a set of initial values, which is very time expensive in the case of simulations. Instead we chose to include $BI(\xi)$ and $BD(\xi)$ and use a single set of initial values, with $\beta_{j,k,g} = 0$, $\lambda_{j,k} = 1/d$, and the $\beta_{j,k,\phi}$'s where chosen so that $B\phi_k(x) = x/\pi$.

Throughout this section, the bounds on $Bg_k(\cdot)$ and its derivative, $\tilde{C}_g^{(0)}$ and $\tilde{C}_g^{(1)}$, are set to 100 and 1,000, respectively. The bounds on $Bg_k(\cdot)$ are easy to impose, requiring only that the associated B-spline coefficients are less than 100. For the derivative, we checked if the bounds were satisfied after estimation, and in all cases they were.[7] We also assume that the underlying function, f, is three times differentiable ($q = 3$).

For each B-spline, the optimal number of knots is of order $10{,}000^{1/7} \approx 4$. However, better results are often obtained if the search is done over several possible choices, and this is the approach taken here. We allow the number of knots to vary (keeping the number the same for both the estimator of $\phi_1(\cdot)$ and $g_1(\cdot)$) from zero to eleven.[8] The model is chosen

[5] One can use any optimizer that allows for nonlinear constrained optimization, such as GQOPT.

[6] With two interior knots uniformly placed,

$$\begin{aligned} BI(\xi) &= 0.25B_{0,4}(\xi) + 0.75B_{1,4}(\xi) + B_{2,4}(\xi) + B_{3,4}(\xi) \\ BD(\xi) &= B_{0,4}(\xi) + B_{1,4}(\xi) + 0.75B_{2,4}(\xi) + 0.25B_{3,4}(\xi). \end{aligned}$$

[7] A method to impose these types of constraints, *a priori*, is to construct additional constraints at the corresponding knots (or the troubled areas) using the recursive formula for B-spline derivatives (i.e. see de Boor, 1978; p.139). Then do another check after estimation to ensure that the bound is satisfied everywhere; if it is not satisfied, add more constraints and repeat the above process. This strategy has worked well in the past, but it is not imposed here because of compilation time considerations.

[8] The theory in the last section is for the case when r is a predetermined sequence such as $r = \lceil n^{1/7} \rceil$, where $\lceil \cdot \rceil$ is the smallest integer no less than $\cdot$, but the theory easily extends to the case where the search

as the one that minimizes the generalized cross-validation (GCV), as suggested by Eubank (1988, p. 363),

$$\frac{n \cdot \text{obj}(r)}{[1 - p(r)/n]^2},$$

where obj(r) is the value of the objection function evaluated at r knots and $p(r) = 2r + d + 5$ is the corresponding number of parameters. Even though this selection method is based on the standard B-spline, it works well here too.

We also chose the knot sequence in two ways. The first was to uniformly place the knots along the interval $[0, \pi]$ for both the estimators of $\phi_1(\cdot)$ and $g_1(\cdot)$. Since $B\phi_1(\cdot)$ is monotonic, knot placement for it is not as important as knot placement for $Bg_1(\cdot)$, which can be more variable. So in the second case, we also estimated the corresponding knots used in constructing $Bg_1(\cdot)$. Adjusting for the increase in parameters in the second case (i.e. $p(n) = 3r + d + 4$), the two methods were evaluated by GCV, and the variable knot method outperformed the fixed case, which probably has to due with the fact that the regression function is very oscillatory.

As a method of comparison, standard nonparametric regression is also implemented. We use two different kernels, one with a gaussian density, which is called Kernel-1, and the other, called Kernel-2, is a gaussian density times $1.5 - 0.5x^2$. Kernel-1 is a second order kernel and Kernel-2 is a fourth order kernel, where if the underling function is twice continuously differentiable, a second and fourth order kernel converge in L_2 at a rate $n^{-2/(4+d)}$, and if the underlying function is four times continuously differentiable, a rate of $n^{-4/(8+d)}$ is obtained if a fourth order kernel is used, but the rate for the second order kernel is still $n^{-2/(4+d)}$. It is possible to construct a third order kernel, which would be consistent with the assumptions imposed on Lin-Supe, but this would imply that the kernel would have the undesirable property of being asymmetric. Hence in terms of a relevant comparison with Lin-Supe, one can take Kernel-1 as a lower bound and Kernel-2 as an upper bound on the kernel's overall performance. Even though kernels of order higher than four are often used in theory, they are rarely used in practice. The bandwidth is chosen by cross-validation (CV), which is optimal in terms of minimizing the MSE, over the set $\{0.01, 0.02, \ldots 1.0\}$.

is done over the set $\{\max(\lceil n^{1/7} \rceil - C_1, 0), \ldots, \min(\lceil n^{1/7} \rceil + C_2, \lceil n \rceil)\}$, where C_1, C_2 are positive constants. The reason why the theory carries over is that the order of the metric entropy remains the same.

A summary of these results is reported in Table 1. The root mean squared error (RMSE) is the same as the pseudo-metric used in Theorem 3, and it is calculated as[9] $\{E[\hat{\omega}_n(\vec{x}) - f(\vec{x})]^2\}^{1/2}$. Because of time considerations, only 10 simulations were performed. However, the standard deviation of the RMSE's is also reported. This statistic is usually small, suggesting that further simulations would not substantially change the results. Given this, it is quite clear that relative to the kernel, the model does extremely well, even in the case when there are only two regressors.[10] The fact that the RMSE for Lin-Supe decreases as d increases is an indicator that the estimator is working well—that it is not greatly affected by the number of regressors. This can be seen by observing that all else equal, the MSE is the integration of the squared error times the uniform density, π^{-d}. For ease of comparison, the root integrated squared error (RISE), $\{\int[\hat{\omega}_n(\vec{x}) - f(\vec{x})]^2\, d\vec{x}\}^{1/2}$, is also reported. Across the ten simulations, the median number of knots and their range is denoted as Med(r) and Range(r).

We also would like to evaluate Lin-Supe in a more realistic type of economic setting and, at the same time, still be as general as possible. This rules out very oscillatory functions, but unfortunately leaves little other guidelines. There are also other limitations. Most of the nonlinear functions that researchers use in simulations, because of their familiarity and because the functions are built into most computer programs, are of the form of $\sin(\cdot)$, $\cos(\cdot)$, $\log(\cdot)$, or $\exp(\cdot)$, and it is hard escaping this. The usual alternative is to use some type of polynomial, but that poses problems here because B-splines are themselves piecewise polynomials. Contrary to Model 1, we also do not want the regression functions to obviously be of the form of something like in (2). Finally, we need the model to be comparable across different sets of regressors. With this in mind, the following two experiments are

[9]The MSE was calculated using a grid search. For the case of d = 2,3, the computation was constructed using 50^d uniformly placed points, $(\pi/100, \pi/100, \ldots, \pi/100)$, $(3\pi/100, \pi/100, \ldots, \pi/100)$, $\ldots, (99\pi/100, 99\pi/100, \ldots, 99\pi/100)$. In the case $d = 5$, 10^5 uniformly placed points where chose. The reason for the reduction is the length of time it took to complete the statistic in the case of kernels. Even running a Fortran program on a Sun Ultra 2, the statistic took over 2 hours per sample to compute.

[10]To see if the difference in performance is due to the grid size used in the CV construction, for the first sample with $d = 5$, the RMSE for Kernel-1 was also computed on a grid with a difference of only 0.0001 between points. The bandwidth and RMSE for the coarser grid are 0.30 and 0.40601, and the corresponding calculations for the finer grid are 0.2979 and 0.40584. The difference, especially relative to the performance of the Lin-Supe, is negligible, however. The time it takes to compute a CV estimate at a single bandwidth is about 15 minutes.

Table 1: Model 1

$f(\vec{x}) = \sin[\sum_{j=1}^{d} \exp(x_j)/d]$

	RMSE	SD	RISE	Med(r)	Range(r)	k^*
			$d = 2$			
Lin-Supe	0.057	0.004	0.179	7	7-10	1
Kernel-1	0.171	0.010	0.538	—	—	—
Kernel-2	0.164	0.010	0.516	—	—	—
			$d = 3$			
Lin-Supe	0.055	0.006	0.307	8	7-11	1
Kernel-1	0.270	0.003	1.506	—	—	—
Kernel-2	0.264	0.005	1.472	—	—	—
			$d = 5$			
Lin-Supe	0.048	0.005	0.838	8	6-11	1
Kernel-1	0.408	0.005	7.140	—	—	—
Kernel-2	0.401	0.004	7.020	—	—	—

Note: RMSE is the average root mean squared error, SD is the standard deviation of the RMSE across the different estimates, RISE is the average root integrated squared error, Med(r) and Range(r) is the median value and range, respectively, for the number of knots across the estimates, which are chosen by generalized cross validation, and k^* is the number of different g_k's used in Lin-Supe. Kernel-1 is the kernel estimator with the standard normal as a kernel, and Kernel-2 is a fourth order kernel, $1.5 - 0.5x^2$ times a univariate normal. For both kernel estimators, the bandwidth is chosen by cross-validation.

constructed,[11]

Model 2

$$y = -5 + \frac{10 \cdot \sum_{j=1}^{d} \exp(z_j)}{1 + \sum_{j=1}^{d} \exp(z_j)} + \epsilon, \qquad z_j = 1.5x_j + x_j^2 - 2.9x_j^3 + 1.25x_j^4 - 0.15x_j^5,$$

and

Model 3

$$y = \exp\left[\sin\left(\sum_{j=1}^{d} x_j\right) - c \cdot \sum_{j=1}^{d} j \cdot \cos(x_j)\right] + \epsilon, \qquad c = \frac{1}{\sum_{j=1}^{d} j},$$

and the corresponding plots for the case of two regressors are given in Figures 2 and 3. The constants in the regression functions are used so that the three models, in terms of height, are comparable across each other and across the different sets of regressors. The fifth degree polynomial in Model 2 looks like an inverted "S" lying on its side. Model 3 is constructed so that there is some asymmetry with respect to each regressor.

In this more general setup, the shear number of parameters in our model make direct optimization difficult. We instead use the back-fitting algorithm, proposed by Friedman and Stuetzle (1981), to solve the optimization problem. Theoretical convergence properties are given in Buja, Hastie, and Tibshirani (1989).

We can view the estimator, $B_n(\vec{x})$, in (10) as having $2d+1$ terms, where each term is denoted by $A_{n,k}(\vec{x})$, $k = 1, \ldots, 2d+1$,

$$B_n(\vec{x}) \;=\; \sum_{k=1}^{2d+1} A_{n,k}(\vec{x}).$$

The idea is to optimize each of the $A_{n,k}$ separately, treating the other $A_{n,k'}$, $k' = 1, \ldots, k-1, k+1, \ldots, 2d+1$, as fixed, and this is repeated until some threshold is reached. Note that the parameter restrictions imposed in Ω_n only apply to each $A_{n,k}$; there are no restrictions across the $A_{n,k}$'s. Hence optimization will be performed subject to only those constraints in Ω_n which affect $A_{n,k}$. For each $A_{n,k}$, we will denote the relevant subset of Ω_n as $\Omega_{n,k}$. We

[11] Donoho and Johnstone (1989) constructed a comparison of projection pursuit and kernels, but the functions from which they simulated, harmonic and radial, suffer the same lack of economic realism as in Model 1.

will also add another subscript and "hat" to $A_{n,k}$, $\hat{A}_{n,k,m}$, to denote the estimated function from the mth step. The back-fitting method is outlined below (note that there are $2d+1$ steps to each part).

Step 1

1) $\min_{\Omega_{n,1}} \frac{1}{n}\sum_{i=1}^{n}[y_i - A_{n,1}]^2$ to get $\hat{A}_{n,1,1}$.

2) Construct $\hat{y}_{i,2,1} = y_i - \hat{A}_{n,1,1}(x_i)$ and then $\min_{\Omega_{n,2}} \frac{1}{n}\sum_{i=1}^{n}[\hat{y}_{i,2,1} - A_{n,2}]^2$ to get $\hat{A}_{n,2,1}$.

$\vdots$

$2d+1$) Construct $\hat{y}_{i,2d+1,1} = y_i - \sum_{k'=1}^{2d}\hat{A}_{n,k',1}(x_i)$ and then $\min_{\Omega_{n,2d+1}} \frac{1}{n}\sum_{i=1}^{n}[\hat{y}_{i,2d+1,1} - A_{n,2d+1}]^2$ to get $\hat{A}_{n,2d+1,1}$.

Step m

1) Construct $\hat{y}_{i,1,m} = y_i - \sum_{k'=2}^{2d+1}\hat{A}_{n,k',m-1}(x_i)$ and then $\min_{\Omega_{n,1}} \frac{1}{n}\sum_{i=1}^{n}[\hat{y}_{i,1,m} - A_{n,1}]^2$ to get $\hat{A}_{n,1,m}$.

$\vdots$

k) Construct $\hat{y}_{i,k,m} = y_i - \sum_{k'=1}^{k-1}\hat{A}_{n,k',m}(x_i) - \sum_{k'=k+1}^{2d+1}\hat{A}_{n,k',m-1}(x_i)$ and then $\min_{\Omega_{n,k}} \frac{1}{n}\sum_{i=1}^{n}[\hat{y}_{i,k,m} - A_{n,k}]^2$ to get $\hat{A}_{n,k',m}$.

$\vdots$

$2d+1$) Construct $\hat{y}_{i,2d+1,m} = y_i - \sum_{k'=1}^{2d}\hat{A}_{n,k',m}(x_i)$ and then $\min_{\Omega_{n,2d+1}} \frac{1}{n}\sum_{i=1}^{n}[\hat{y}_{i,2d+1,m} - A_{n,2d+1}]^2$ to get $\hat{A}_{n,2d+1,m}$.

Step $m+1$

Repeat Step m until each $||\hat{A}_{n,k,m+1} - \hat{A}_{n,k,m}||$, $k = 1, \ldots, 2d+1$, does not change by more than some predetermined threshold.

Initial values in the first step are analogous to those used in Model 1. Like before, the model is chosen as the one which minimizes the GVC, and in both of cases, fixed knots is selected. A summary of the results for Model 2 is reported in Table 2. One of the most surprising results is that k^* is only two, where k^* is the number of $A_{n,k}(\cdot)$ terms, $k = 1, \ldots, k^*$, defined above. Lin-Supe also performs well especially for the cases of five regressors, in which the relative improvement in the RMSE is about 60 percent. To get a feel for the size of this difference, suppose $C_1 n^{-2/9}$ and $C_2 n^{-4/13}$ are good approximations to the RMSE, where

Table 2: Model 2

$$f(\vec{x}) = -5 + 10\sum_{j=1}^{d}\exp(z_j)/[1+\sum_{j=1}^{d}\exp(z_j)]$$
$$z_j = 1.5x_j + x_j^2 - 2.9x_j^3 + 1.25x_j^4 - 0.15x_j^5$$

	RMSE	SD	RISE	Med(r)	Range(r)	k^*
			$d=2$			
Lin-Supe	0.112	0.008	0.351	6	4-8	2
Kernel-1	0.123	0.009	0.386	—	—	—
Kernel-2	0.121	0.010	0.379	—	—	—
			$d=3$			
Lin-Supe	0.121	0.008	0.673	7.5	3-9	2
Kernel-1	0.178	0.005	0.991	—	—	—
Kernel-2	0.176	0.006	0.975	—	—	—
			$d=5$			
Lin-Supe	0.081	0.006	1.420	3	3-7	2
Kernel-1	0.202	0.005	3.527	—	—	—
Kernel-2	0.202	0.004	3.534	—	—	—

Note: RMSE is the average root mean squared error, SD is the standard deviation of the RMSE across the different estimates, RISE is the average root integrated squared error, Med(r) and Range(r) is the median value and range, respectively, for the number of knots across the estimates, which are chosen by generalized cross validation, and k^* is the number of different g_k's used in Lin-Supe. Kernel-1 is the kernel estimator with the standard normal as a kernel, and Kernel-2 is a fourth order kernel, $1.5 - 0.5x^2$ times a univariate normal. For both kernel estimators, the bandwidth is chosen by cross-validation.

$n^{-2/9}$ and $n^{-4/13}$ are the corresponding rates of convergence for Kernel-1 and Kernel-2 when $d = 5$ and C_1 and C_2 are some constants. Then extrapolating C_1 and C_2 for a RMSE of 0.202 and $n = 10,000$, and then projecting the sample size it takes for the kernels to obtain a RMSE of 0.081, we see that the estimated number of observations is around 610,777 and 194,901, respectively. In this light, Lin-Supe's improvement is quite substantial.

A summary of the results for Model 3 is reported in Table 3. Relative to the kernel, Lin-Supe again performs very well. In the case of five regressors, there is about an 65 and 59 percent improvement, respectively, in the RMSE. Like before, sample sizes of 1,121,998 and 184,719 are projected to equate the RMSE's across Lin-Supe and the kernels. The size

Table 3: Model 3

$f(\vec{x}) = \exp\{\sin(\sum_{j=1}^{d} x_j) - [\sum_{j=1}^{d} j \cos(x_j)]/(\sum_{j=1}^{d} j)\}$

	RMSE	SD	RISE	Med(r)	Range(r)	k^*
			$d = 2$			
Lin-Supe	0.076	0.008	0.238	4	2-7	2
Kernel-1	0.074	0.008	0.232	—	—	—
Kernel-2	0.069	0.008	0.216	—	—	—
			$d = 3$			
Lin-Supe	0.095	0.010	0.531	6	4-8	3
Kernel-1	0.159	0.007	0.886	—	—	—
Kernel-2	0.143	0.007	0.796	—	—	—
			$d = 5$			
Lin-Supe	0.117	0.012	2.054	9	7-11	3
Kernel-1	0.334	0.004	5.837	—	—	—
Kernel-2	0.287	0.004	5.028	—	—	—

Note: RMSE is the average root mean squared error, SD is the standard deviation of the RMSE across the different estimates, RISE is the average root integrated squared error, Med(r) and Range(r) is the median value and range, respectively, for the number of knots across the estimates, which are chosen by generalized cross validation, and k^* is the number of different g_k's used in Lin-Supe. Kernel-1 is the kernel estimator with the standard normal as a kernel, and Kernel-2 is a fourth order kernel, $1.5 - 0.5x^2$ times a univariate normal. For both kernel estimators, the bandwidth is chosen by cross-validation.

of the model, as determined by k^*, is relatively small for this model too, suggesting that the general form of Lin-Supe is remarkably flexible.

4 Conclusion

This paper introduces a nonparametric regression estimator, Lin-Supe, for the case of multivariate regressions. The core construction rests on the theoretical result that any continuous function of several variables can be represented as sums of superpositions of functions of one variable. Lin-Supe removes some of the arbitrariness of the other high dimensional estimators, such linear additive models, where functional restrictions are made more for statistical reasons than economic ones. A simulation study shows that the estimator performs quite

well as compared to the kernel even when there are only three regressors and the sample size is 10,000 observations. One of the most surprising findings is that the number of terms, as measured by k^*, is very small, suggesting that Lin-Supe is very adaptable.

The methodology developed here has many further extensions, both within and outside of economics, especially for univariate functions. The monotonicity restriction alone has other applications. An example, specific to economics, is estimating a demand function, and an application outside of economics is treatment responses to, say, some drug (e.g. see Bloch and Silverman 1998).

Concavity is another useful restriction. Again the method to do this is clear once we are able to show how to impose it on a cubic polynomial, $p(x) = \gamma_0 + \gamma_1 x + \gamma_2 x^2 + \gamma_3 x^3$, $x \in [a, b]$. Observe that the second derivative, $p''(x) = 2\gamma_2 + 6\gamma_3 x$, is linear, thus to ensure $p''(x) \leq 0$, we need $2\gamma_2 + 6\gamma_3 a \leq 0$ and $2\gamma_2 + 6\gamma_3 b \leq 0$. This is simpler than monotonicity, requiring only $r + 2$ constraints for the B-spline. To impose both monotonicity and concavity, just combine both sets of constraints to give a total of $3(r + 1) + 2$ restrictions. Using the same methods as for the case of monotonicity alone, it is easy to show that the concave restriction and the monotonicity and concave restriction together are dense in the space of cubic B-splines with these properties. Enforcing linear homogeneity, as in Matzkin (1994), at the knots, for example, is also possible. Judd (1998) lists several other uses for these types of "shape preserving" estimators.

As a first cut to the data, the researcher may want to run the following simpler model. Instead of imposing the restrictions on the δ's in Definition 2, impose $\beta_{l,k,\phi} \leq \beta_{l+1,k,\phi}$, $l = 1, \ldots, r + 3$, $k = 1, \ldots, 2d + 1$. The benefit is that all of the restrictions, assuming that the estimate satisfies the derivative bounds, are now linear (and hence easier to program) while the estimator for $\phi_k(\cdot)$ is still monotone. The cost is that the estimator is not dense in the set of all monotonic cubic B-splines, implying that the fit will tend to be not as good. Another simplification would be to fix k at 1, thus avoiding the need for the back-fitting algorithm.

Unfortunately, if $f \notin \mathcal{F}$, where $\mathcal{F}$ is as in Definition 1, but $f \in C^q([a, b]^d)$, there does not appear to be any straightforward method for bounding the asymptotic bias. At least an analogous result could not be found for additive models, projection pursuit models, or neural

nets.[12] The following estimator, nonetheless, will asymptotically drive the bias to zero,

$$\tilde{B}_n(\vec{x}) \;=\; \tau_n B_n(\vec{x}) \;+ (1-\tau_n)\bar{B}_n(\vec{x}),$$

where B_n is as in Section 2, $\bar{B}_n$ is some general multivariate estimator (such as a kernel or B-spline) whose bias tends to zero, and τ_n is a sequence of constants which tend to zero. Of course $\tilde{B}_n(\vec{x})$ will no longer have the faster rate of convergence, asymptotically. To tie this into the paper, one could argue that in general, by the results in Section 3, B_n works better than $\bar{B}_n$ for general $f \in C^q([a,b]^d)$ when the sample size is not extremely large. But when $f \notin \mathcal{F}$, standard theory indicates that $\bar{B}_n$ will eventually outperform B_n; however, the results in Section 3 suggest that this will only hold, in general, for very large sample sizes if d is moderately large.

Finally, it is noted that the results here also will hold for certain types of temporal dependency. The relevant theory is given in Chen and Shen (1998).

[12] Diaconis and Shahshahani (1984) have shown that, in this case, the asymptotic bias for the projection pursuit model can be made arbitrarily small if there are a suitably large number of g_k terms. Likewise, Hornik (1991) develops a similar result for neural nets. In the same vein, Schumaker (1981, Corollary 6.21) shows that, under the strong norm, there exists a cubic B-spline that approximates any bounded continuos function on $[a,b]$ arbitrarily well, and there also exists a cubic B-spline that approximates a Lipschitz of order one function on $[a,b]$ at the rate $O(r^{-1})$ as $r \to \infty$. This suggests that the asymptotic bias for Lin-Supe can also be made to tend to zero. Nonetheless, these results are not strong enough to establish rates of convergence, especially if we want outperform the standard nonparametric case. On the other hand, the above results for B-splines imply that the asymptotic bias well be small as long as $\tilde{C}_g^{(1)}$ is relatively large.

5 References

Barron, A. R. (1993), "Universal Approximation Bounds for Superpositions of a Sigmoidal Function," *IEEE Transactions on Information Theory*, 39, 930–945.

Bloch, D. A., and B. W. Silverman (1998), "Monotone Discriminant Functions and Their Applications in Rheumatology," *Journal of the American Statistical Society*, 92, 144–153

Buja, A., T. Hastie, and R. Tibshirani (1989), "Linear Smoothers and Additive Models," *Annals of Statistics*, 17, 453–555.

Chen X., and X. Shen (1998), "Sieve Extremum Estimates for Weakly Dependent Data," *Econometrica*, 66, 289–314.

de Boor, C. (1978), *A Practical Guide to Splines*, Springer-Verlag.

Diaconis, P., and M. Shahshahani (1984), "On Nonlinear Functions of Linear Combinations," *SIAM Journal on Scientific and Statistical Computing*, 5, 175–191.

Donoho, D. L., and I. M. Johnstone (1989), "Projection-Based Approximation and a Duality with Kernel Methods," *The Annals of Statistics*, 17, 58–106.

Eubank, R. L. (1988), *Spline Smoothing and Nonparametric Regression*, Marcel Kedder, Inc.

Fenton, V. M., and A. R. Gallant (1996), "Convergence Rates of SNP Density Estimators," *Econometrica*, 64, 719-727.

Friedman, J. H., and Stuetzle, W. (1981), "Projection Pursuit Regression," *Journal of the American Statistical Society*, 76, 817–823.

Gill, P. E., W. Murray, M. A. Saunders, and M. H. Wright (1986), "User's Guide for NPSOL (Version 4.0): A Fortran Package for Nonlinear Programming," Technical Report, Stanford University.

Hastie, T., and Tibshirani, R. (1986), "Generalized Additive Models," *Statistical Science*, 1, 297–318.

Hornik, K. (1991), "Approximation Capabilities of Multilayer Feedforward Networks," *Nueral Networks*, 4, 251-257.

Judd, K. (1998), *Numerical Methods in Economics*, The MIT Press

Kolmogorov, A. N. (1957), "On the Representation of Continuous Functions of Several Variables by Superpositions of Continuous Functions of One Variable and Addition," *Doklady*, 114, 679–681.

Kolmogorov, A. N. and V. M. Tihormirov (1961): "ϵ-Entropy and ϵ-Capacity of Sets in Functional Spaces," *American Mathematical Society Translations*, Series 2, 17, 277–364.

Lorentz, G. G. (1966), *Approximation of Functions* , Holt, Rinehart and Winston.

Lorentz, G. G., M. v. Golitschek, and Y. Makovoz (1996), *Constructive Approximation Advanced Problems*, Springer-Verlag.

Matzkin, R. L. (1994), "Restrictions of Economic Theory in Nonparametric Methods," in: R. Engle and D. L. McFadden, eds., *Handbook of Econometrics*, vol. 4, North-Holland.

Ramsay, J. O. (1988), "Monotone Regression Splines in Action," *Statistical Science*, 3, 425–461.

Schumaker, L. L. (1981), *Spline Functions Basic Theory*, John Wiley & Sons.

Schumaker, L. L. (1983), "On Shape Preserving Quadratic Spline Interpolation", *SIAM Journal of Numerical Analysis*, 20, 854–864.

Shen, X., and W. H. Wong (1994), "Convergence Rates of Sieve Estimates," *Annals of Statistics*, 22,580–615.

Silverman, B. W. (1986), *Density Estimation for Statistics and Data Analysis* , Chapman and Hall.

Stone, C. (1982), "Optimal Global Rates of Convergence for Nonparametric Regression," *Annals of Statistics*, 18, 907–924.

Wright, I. W., and E. J. Wegman (1980), "Isotonic, Convex and Related Splines," *Annals of Statistics*, 8, 1023–1035.

Figure 1: MODEL 1

$$\sin\left(\exp(x)/2 + \exp(y)/2\right))$$

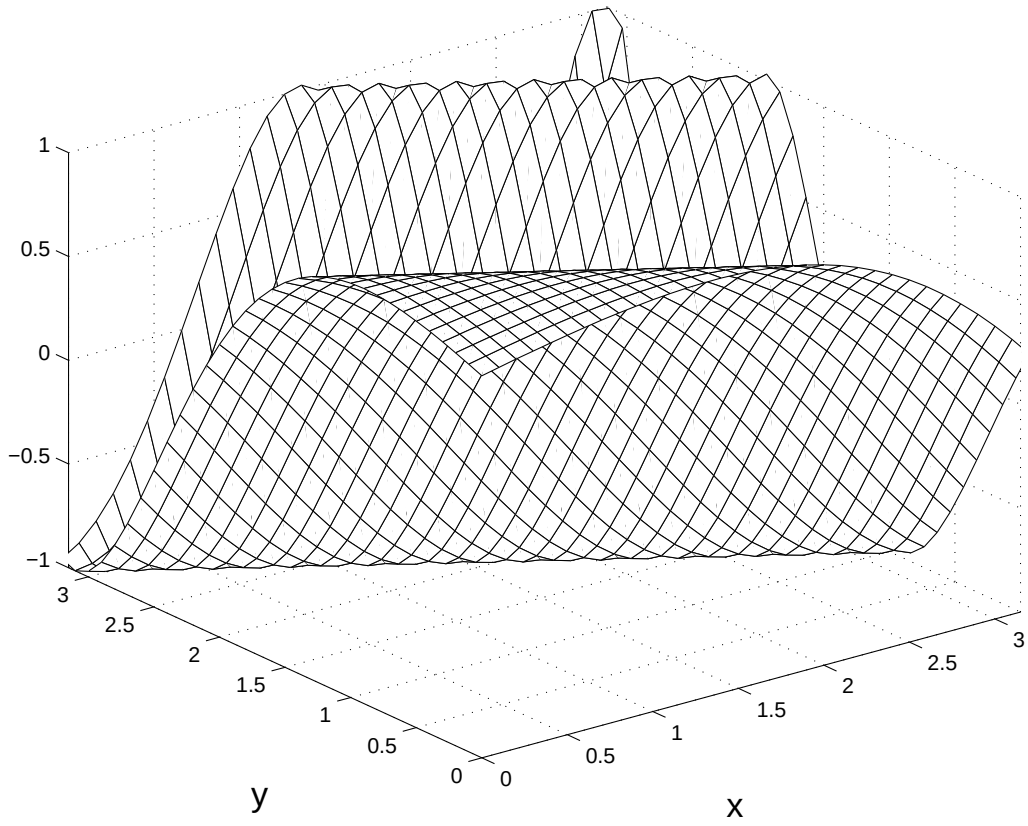

Figure 2: MODEL 2

$$-5 + 10\left[\exp(z_1) + \exp(z_2)\right] / \left[1 + \exp(z_1) + \exp(z_2)\right],$$
$$z_1 = 1.5x + x^2 - 2.9x^3 + 1.25x^4 - 0.15x^5,$$
$$z_2 = 1.5y + y^2 - 2.9y^3 + 1.25y^4 - 0.15y^5$$

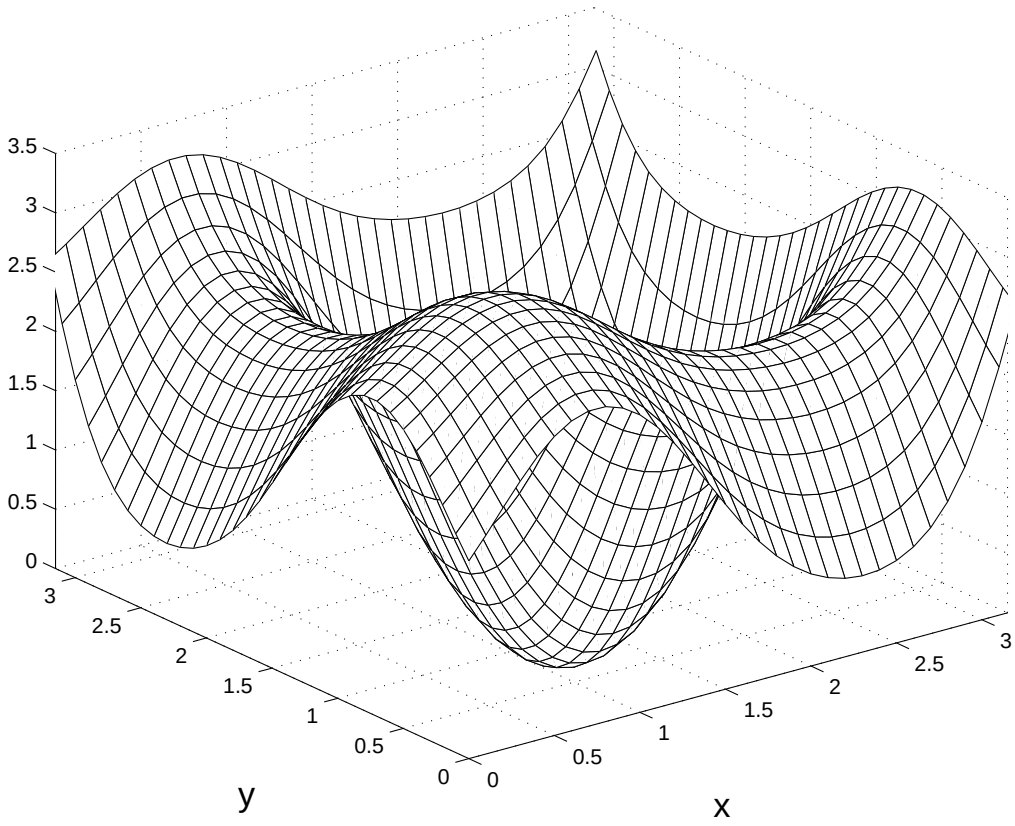

Figure 3: MODEL 3

$$\exp\left[\sin\left(x+y\right)-\cos(x)/3-2\cos(y)/3\right]$$