Econometrics Journal (2008), volume **00**, pp. 1–12.
Article No. ectj??????

K-nn Nonparametric Estimation Of Regression Functions In the Presence of Irrelevant Variables

Rui Li[†] and Guan Gong[‡]

[†]*School of Economics & Management, Beijing University of Aeronautics & Astronautics,China*
E-mail: lirui@buaa.edu.cn

[‡]*School of Economics, Shanghai University of Finance and Economics, China*
E-mail: ggong@mail.shufe.edu.cn

Received: First version received: December 2006; final version received: January 2008

Summary We show that when estimating a nonparametric regression model, the k-nearest-neighbor nonparametric estimation method has the ability to remove irrelevant variables provided one uses a product weight function with a vector of smoothing parameters, and the least squares cross validation method is used to select the smoothing parameters. Simulation results are consistent with our theoretical analysis and show that the performance of the k-nn estimator is comparable to the popular kernel estimator; and it dominates a nonparametric series (spline) estimator when there exist irrelevant regressors.

Keywords: *k Nearest Neighbor,Cross Validation, Irrelevant Variables, Simulations.*

1. INTRODUCTION

The nonparametric kernel estimation method is by far the most popular technique used to estimate a regression model nonparametrically. Recently, Hall, Li and Racine (2007) show that the kernel estimation method, coupled with the least squares cross validation method of selecting the smoothing parameters, has the amazing property that irrelevant (continuous or discrete) regressors can be automatically smoothed out. Li, Racine and Wooldridge (2007) further use a kernel-based propensity score estimator to estimate an average treatment effect. They defend the use of the kernel method by stating that, "It is not clear to us how to extend the property of kernel smoothing to other nonparametric estimation methods such as series methods (i.e., the ability to automatically remove irrelevant covariates). Therefore, we restrict our attention to nonparametric kernel methods in this paper."

Indeed, when one faces a mixture of continuous and discrete variables, the only known nonparametric series estimation method is to use indicator functions to split the sample into discrete cells, and then estimate a regression model using data from each cell. This sample split method becomes infeasible when the number of discrete cells is large. Therefore, the nonparametric series estimation method does not share the amazing 'removing irrelevant variable' property of the kernel-based estimator. In this paper we investigate the problem of whether or not the nonparametric k-nn method can have the property of automatically removing irrelevant variables in a regression model. We show that the k-nn estimator can indeed remove irrelevant variables from a regression model. However, special attention is needed in order for a k-nn estimator to possess this property. First, in order to remove irrelevant variables, a product weight function must be used with a vector of smoothing parameters so that each component of the regressor has a different

smoothing parameter. Second, only the uniform weight function has the property of *completely* removing irrelevant variables. All other weight functions, such as the Gaussian and Epanechnikov weight functions, do not have this property. However, a simple modification can lead to k-nn estimators with non-uniform weight functions to possess the property of removing irrelevant variables. This is quite different from the kernel-based estimators because Hall, Li and Racine (2007) show that only non-uniform kernel functions have the ability of removing irrelevant regressors.

In this paper we use simulation results to compare the finite sample performance of three popular nonparametric methods: the kernel method, the nearest neighborhood (k-nn) method, and the spline method. We are particularly interested in evaluating these estimators in the presence of irrelevant regressors. The remaining parts of the paper are organized as follows. In section 2 we first discuss the conventional k-nn estimator and compare it with the kernel estimation results of Hall et al (2007). We show in section 3 that with some modifications, the conventional k-nn estimation method can possess the ability of removing irrelevant variables. Section 4 reports the simulation results and examines the finite sample behavior of the k-nn estimator and compares it with the nonparametric kernel and series estimators. Finally, section 5 concludes the paper.

2. THE NONPARAMETRIC K-NN ESTIMATION METHOD

We will mainly focus on the case of a nonparametric regression model with continuous regressors. We will briefly discuss the mixed discrete and continuous regressor case at the end of this section. Considering the following nonparametric regression model:

$$Y_i = g(X_i) + u_i, \qquad i = 1, 2, ..., n, \tag{2.1}$$

where $X_i \in \mathcal{R}^q$ is a continuous variable of dimension q, the functional form of $g(\cdot)$ is unspecified. We only consider the case where $(Y_i, X_i)_{i=1}^n$ are independent and identically distributed (i.i.d); the results of the paper can be readily extended to the weakly dependent data case.

We first review the conventional k-nn estimation method and then discuss what modifications are needed in order for the k-nn estimator to be able to automatically remove irrelevant variables. We will use the same notation as in Ouyang et al (2006). For a fixed value $x \in \mathcal{R}^q$, define the k-nearest-neighbor distance, centered at x by

$$\begin{aligned} R_x \equiv R_n(x) \stackrel{def}{=} & \text{ the } k^{th} \text{ nearest Euclidean distance to } x \\ & \text{among all the } X_j\text{'s for } j = 1, ..., n. \end{aligned} \tag{2.2}$$

Also define the *k-nn* distance centered at X_i as:

$$\begin{aligned} R_i \equiv R_n(X_i) \stackrel{def}{=} & \text{ the } k^{th} \text{ nearest Euclidean distance to } X_i \\ & \text{among all the } X_j\text{'s for } j = 1, ..., n. \end{aligned} \tag{2.3}$$

Let $W(\cdot) : \mathcal{R}^q \to \mathcal{R}$ be a bounded non-negative weight function, $\int W(v)dv = 1$, $\int W(v)||v||^4 dv < \infty$, where $||v||$ denotes the Euclidean norm of v. For example, one can use the Epanechnikov weight function defined by $W(v) = (3/4)\left(1 - ||v||^2\right) \mathbf{1}(||v|| \leq 1)$, where $\mathbf{1}(A)$ is an indicator function which takes value 1 if event A holds true, and 0 otherwise. The local constant *k-nn* estimator of $g(x)$ is given by

$$\hat{g}(x) = \frac{1}{nR_x^q} \sum_{i=1}^{n} Y_i W\left(\frac{X_i - x}{R_x}\right) / \hat{f}(x) \tag{2.4}$$

where $\hat{f}(x) = \frac{1}{nR_x^q}\sum_{i=1}^{n} W\left(\frac{X_i - x}{R_x}\right)$ is the *k-nn* estimator of $f(x)$, $f(\cdot)$ is the probability density function of X_i.

Let $\hat{G}_k$ denote the $n \times 1$ vector with its i^{th} element being given by $\hat{g}(X_i)$. Then we have that $\hat{G}_k = M_n(k)Y$, where $M_n(k)$ is a $n \times n$ matrix with its $(i,j)^{th}$ element given by $W_{ij}/\sum_{l=1}^{n} W_{il}$, where $W_{ij} = W((X_i - X_j)/R_i)$.

The following three procedures for selecting k were studied by Li (1987).

(i) Mallows' C_L method (or C_p, Mallows (1973)):

One selects $\hat{k}$ to minimize the following objective function:

$$\hat{k}_C = \arg\min_k n^{-1}\sum_{i=1}^{n}[Y_i - \hat{g}(X_i)]^2 + 2\sigma^2 tr[M_n(k)]/n, \tag{2.5}$$

where σ^2 is the variance of u_i. We estimate σ^2 by $\hat{\sigma}^2 = n^{-1}\sum_{i=1}^{n}\hat{u}_i^2$ with $\hat{u}_i = Y_i - \hat{g}(X_i)$.

(ii) Generalized cross-validation method (Craven and Wahba (1979)):

One selects $\hat{k}$ to minimize the following objective function:

$$\hat{k}_{GCV} = \arg\min_k \frac{n^{-1}\sum_{i=1}^{n}[Y_i - \hat{g}(X_i)]^2}{(1 - n^{-1}tr[M_n(k)])^2}. \tag{2.6}$$

(iii) Leave-one-out cross-validation method (Stone (1974)):

One chooses $\hat{k}_{CV}$ to minimize

$$\hat{k}_{CV} = \arg\min_k CV(k) \equiv \arg\min_k \left\{\frac{1}{n}\sum_{i=1}^{n}[Y_i - \hat{g}_{-i}(X_i)]^2\right\}, \tag{2.7}$$

where $\hat{g}_{-i}(X_i) = \sum_{j\neq i}^{n} Y_j W_{ij}/\sum_{j\neq i}^{n} W_{ij}$ is the leave-one-out *k-nn* estimator of $g(X_i)$.

Li (1987) has shown that the above three procedures are asymptotically equivalent and all of them lead to optimal smoothing in the sense that

$$\frac{\int[\hat{g}_{\hat{k}}(x) - g(x)]^2 dF(x)}{\inf_k \int[\hat{g}_k(x) - g(x)]^2 dF(x)} \xrightarrow{p} 1, \tag{2.8}$$

where $F(\cdot)$ is the distribution function of X_i, and $\hat{g}_{\hat{k}}(x)$ is the *k-nn* estimator of $g(x)$ defined in (2.4) using one of the above procedures to select k, i.e., in (2.8), $\hat{k} = \hat{k}_C$, or $\hat{k} = \hat{k}_{GCV}$, or $\hat{k} = \hat{k}_{CV}$; and $\hat{g}_k(x) = \hat{g}(x)$ is the k-nn estimator of $g(x)$ with a generic k as defined in (2.4). Ouyang et al (2006) extend Li's (1987) result and derive the rate of convergence of the leave-one-out cross-validation $\hat{k}_{CV}$ to some non-stochastic optimal benchmark value of k, say k_0.

We now show that the above conventional k-nn estimator does not possess the property of removing irrelevant variables. We use X_{is} to denote the s^{th} component of X_i. Often in applied settings not all q regressors in X_i are relevant. Without loss of generality, assume that the first q_1 $(1 \leq q_1 \leq q)$ components of X are "relevant" in the sense defined below.

Let $\bar{X}$ consist of the first q_1 relevant components of X let $\tilde{X} = X \setminus \{\bar{X}\}$ denote the remaining irrelevant components of X. Following Hall et al. (2007) we define $\bar{X}$ to be relevant and $\tilde{X}$ to be irrelevant by asking that

$$(\bar{X}, Y) \quad \text{is independent of} \quad \tilde{X} \tag{2.9}$$

Clearly, (2.9) implies that $E(Y|X) = E(Y|\bar{X})$. A weaker condition would be to ask that, conditional on $\bar{X}$, the irrelevant variables $\tilde{X}$ and Y are independent. Hall et al. (2007) show that under (2.9), the least squares cross validation method can automatically

remove irrelevant variables. However, simulations reported in Hall et al. (2007) reveal that the 'removing irrelevant variable' results also hold under weaker condition that $E(Y|X) = E(Y|\bar{X})$.

Hall et al. (2007) assume that the true regression model is given by $Y_i = \bar{g}(\bar{X}_i) + u_i$ where $\bar{g}(\cdot)$ is of unknown form, and $E(u_i|\bar{X}_i) = 0$. They assume that the relevant regressors are unknown ex ante, hence one estimates $\bar{g}(\cdot)$ using the superset of regressors $X = (\bar{X}, \tilde{X})$. The nonparametric kernel estimator of $g(x)$ is given by

$$\hat{g}_{kernel}(x) = \frac{\sum_{j=1}^{n} Y_j \prod_{s=1}^{q} w(\frac{X_{js}-x_s}{h_s})}{\sum_{j=1}^{n} \prod_{s=1}^{q} w(\frac{X_{js}-x_s}{h_s})}, \tag{2.10}$$

where $w(\cdot)$ is the univariate kernel function and h_s is the smoothing parameter associated with x_s, $s = 1, ..., q$.

Hall et al. (2007) suggest to choose the smoothing parameters by the least squares cross validation (CV) method. They show that the CV selected smoothing parameters, say $\hat{h}_s$, has the property that $\hat{h}_s \sim n^{-1/(4+q_1)}$ for $s = 1, ..., q_1$ and that $\hat{h}_s \to \infty$ for $s = q_1 + 1, ..., q$. Note that when $h_s = \infty$, we have $lim_{h_s \to \infty} w(\frac{X_{js}-x_s}{h_s}) = w(0)$. And $\hat{g}_{kernel}(x)$ becomes unrelated to x_s. Therefore, all irrelevant variables are smoothed out asymptotically.

The k-nn estimator $\hat{g}(x)$ defined in (2.4) does not possess the ability of removing irrelevant variables. This is because the k-nn distance R_x is a scalar. This is similar to the case of using a scalar smoothing parameter in the kernel estimation case, i.e., $h_1 = ... = h_q = h$. In this case, if there exists at least one relevant regressor, then $h \to 0$ as $n \to \infty$, and no variables will be smoothed out. On the other hand, if $h \to \infty$ as $n \to \infty$, then all variables are smoothed out. A scalar smoothing parameter h cannot have the flexibility of removing some (irrelevant) variables while keeping the remaining (relevant) variables.

For the same reason, the use of a scalar k-nn distance R_x cannot smooth out irrelevant regressors. Therefore, we suggest that one uses a product weight function and uses a different distance for each different component x_s in the nonparametric k-nn estimation. This is the topic of next section.

3. K-NN ESTIMATOR WITH PRODUCT WEIGHT FUNCTION

In order to use a product weight function and allow for a vector of smoothing parameters $(k_1, ..., k_q)$, we first need to introduce some notation. For $s = 1, ..., q$ define $R_{s,x}$ by

$$\begin{aligned} R_{s,x} \equiv R_n(x_s) \stackrel{def}{=} & \text{ the } k_s^{th} \text{ nearest Euclidean distance to } x_s \text{ among} \\ & \text{all the } X_{js}\text{'s for } j = 1, ..., n. \end{aligned} \tag{3.1}$$

Note that the range of k_s is $\{1, 2, ..., n\}$ for all $s = 1, ..., q$.

Also, we define the k_s-nn distance centered at X_{is} by:

$$\begin{aligned} & R_{s,i} \equiv R_n(X_{is}) \stackrel{def}{=} \text{ the } k_s^{th} \text{ nearest Euclidean distance to } X_{is} \\ & \text{among all the } X_{js}\text{'s for } j = 1, ..., n. \end{aligned} \tag{3.2}$$

Using a product weight function $W(\frac{X_j - x}{R_{n,x}}) \stackrel{def}{=} \prod_{s=1}^{q} w(\frac{X_{js}-x_s}{R_{s,x}})$, we define the k-nn

estimator by

$$\hat{g}(x) = \frac{\sum_{j=1}^{n} Y_j W((X_j - x)/R_{n,x})}{\sum_{j=1}^{n} W((X_j - x)/R_{n,x})}. \tag{3.3}$$

We say that the (irrelevant) regressor x_s is completely smoothed out from the regression model if $\hat{g}(x)$ is unrelated to x_s for all $x_s \in \mathcal{S}_s$, where $\mathcal{S}_s$ is the support for X_s. The next lemma shows that a k-nn estimator with a product uniform weight function possess the property of removing irrelevant variables, while a k-nn estimator with any other non-uniform weight function does not share this property. A uniform weight function is defined by $w(v) = (1/2)\mathbf{1}(|v| \leq 1)$, i.e., $w(v) = 1/2$ if $|v| \leq 1$; and 0 otherwise.

LEMMA 3.1. *Letting the k-nn estimator be defined using the product weight function given in (3.3),*

(i) if one uses a (product) uniform weight function, then x_s will be smoothed out from the regression model if $k_s = n$.

(ii) if one uses a non-uniform weight function, then x_s cannot be completely smoothed out for all values of $k_s = 1, ..., n$.

The proof of Lemma 3.1 is given in the Appendix.

Lemma 3.1 shows that the uniform weight function has the property of being able to smooth out irrelevant variables completely, while all other non-uniform weight functions such as the Gaussian or Epanechnikov weight functions do not possess this property. This is quite different from the kernel estimation result of Hall et al. (2007) who rule out the use of a uniform kernel in their analysis.

In practice one does not know which variable is relevant and which variable is irrelevant. Therefore, some data-driven methods are needed to select the smoothing parameter k_s optimally in practice. We suggest using the least squares cross validation method to select $k_1, ..., k_q$, i.e., we select $k_1, ..., k_q$ by minimizing the following objective function:

$$CV(k) = \frac{1}{n} \sum_{i=1}^{n} \left[Y_i - \hat{g}_{-i}(X_i) \right]^2, \tag{3.4}$$

where $\hat{g}_{-i}(X_i) = \sum_{j \neq i}^{n} Y_j W(\frac{X_j - X_i}{R_{n,i}}) / \sum_{j \neq i}^{n} W(\frac{X_j - X_i}{R_{n,i}})$ is the leave-one-out k-nn estimator of $g(X_i)$ with $W((X_j - X_i)/R_{n,i}) = \prod_{s=1}^{q} w((X_{js} - X_{is})/R_{s,i})$ being the product weight function.

When allowing for the use of different k_s, the asymptotic analysis for cross-validation selection of smoothing parameters is quite complex. To the best of our knowledge, no formal asymptotic results are available in this case. Therefore, we will resort to simulation studies to examine the finite sample performance of the k-nn estimator based cross validation selected smoothing parameters (with a vector of smoothing parameters).

According to the result of Lemma 3.1, it seems that one should always use a uniform (product) weight function in nonparametric k-nn estimation. However, removing irrelevant variables is only one part of the story. For relevant variables, a uniform weight function may not be the best choice. Other weight functions such as Gaussian or Epanechnikov may be more suitable to use in finite sample applications because these weight functions give more weight to observations closer to x, rather than a constant weight as the uniform weight function does. Therefore, it would be desirable if one could modify the

cross validation selection rule to smooth the irrelevant variables even with a non-uniform weight function. We show below that this is indeed possible.

From Lemma 3.1 we know that when a non-uniform weight function is used, such as the Gaussian or Epanechnikov weight function, then even when $k_s = n$ (taking the upper extreme value) x_s is not smoothed out completely. However, a simple modification can be made so that an irrelevant variable can be (completely) smoothed out even when one uses a non-uniform weight function. We suggest the following modifications in selecting k_s. If the cross validation method selects $k_s = n$, then one should add an extreme value for $R_{s,i} = \infty$ for all $i = 1, ..., n$. If the resulting $CV(k)$ (with $R_{s,i} = \infty$) has a smaller value than the case with $k_s = n$, then we remove the regressor x_s from the regression model. This is reasonable because $lim_{R_{n,i}\to\infty} w((X_{js} - W_{is})/R_{s,i}) = w(0)$. Hence, x_s is completely removed from $\hat{g}(x)$ if $R_{s,i} = \infty$. Therefore, we remove x_s from the regression model if removing x_s leads to a smaller cross validation function value.

Up to now we have assumed that all the regressors are continuous variables. Let's consider the case of mixed continuous and discrete regressors. Say $X_i = (X_i^c, X_i^d)$, where X_i^c is a continuous variable vector of dimension q, and X_i^d is a discrete variable vector of dimension r. Then one can use the same discrete weight function as suggested by Hall et al (2007) to deal with the discrete variable. Specifically, define $l(X_{is}^d, x_s^d, \lambda_s) = 1$ if $X_{is}^d = x_s^d$; and $l(X_{is}^d, x_s^d, \lambda_s) = \lambda_s$ if $X_{is}^d \neq x_s^d$. Then one can estimate $g(x)$ by

$$\hat{g}(x) = \frac{\sum_{j=1}^n Y_j W(\frac{X_j^c - x^c}{R_{n,x}}) L(X_j^d, x^d, \lambda)}{\sum_{j=1}^n W(\frac{X_j^c - x^c}{R_{n,x}}) L(X_j^d, x^d, \lambda)}, \tag{3.5}$$

where $L(X_j^d, x^d, \lambda) = \prod_{s=1}^r l(X_{js}^d, x_s^d, \lambda_s)$ is the discrete variable product weight function. The range of λ_s is $[0, 1]$ for all $s = 1, ..., r$. If $\lambda_s = 1$, x_s^d is deemed as an irrelevant variable since it is completely smoothed out from the regression model. The cross validation function is modified to:

$$CV(k, \lambda) = \frac{1}{n} \sum_{i=1}^n \left[Y_i - \hat{g}_{-i}(X_i) \right]^2, \tag{3.6}$$

where $\hat{g}_{-i}(X_i) = \sum_{j=1}^n Y_j W(\frac{X_j^c - X_i^c}{R_{n,i}}) L(X_j^d, X_i^d, \lambda) / \sum_{j=1}^n W(\frac{X_j^c - X_i^c}{R_{n,i}}) L(X_j^d, X_i^d, \lambda)$ is the leave-one-out estimator of $g(X_i)$. In practice one selects $(k_1, ..., k_q, \lambda_1, ..., \lambda_r)$ by minimizing the objective function $CV(k, \lambda)$ defined in (3.6).

We examine the finite sample performance of the k-nn estimator in the next section.

4. MONTE CARLO SIMULATIONS

In this section we report simulation results to examine the finite sample performances of the k-nn estimator with a product weight function and with the least squares cross validation method selecting the smoothing parameters. For comparison purposes we also report the results of the kernel estimator and the nonparametric series estimator. For the series estimation method we use the B-spline and the power series.

We consider the series estimators in which we can estimate $g(\cdot)$ using series approximating functions

$$\widehat{g}(x) = \sum_{j=1}^K \beta_j p_j^K(x), \tag{4.1}$$

where $p_j^K(x)$'s are called the base functions which can be power series, B-spline or some other type of series based function (e.g., wavelet).

In our simulations we consider both the power spline and the cardinal B-spline. We consider orders of $r = 2$, 3, and 4 for the B-spline base functions. The results are not sensitive to the order of the base functions. However, the estimator is sensitive to the order of the polynomials in the power spline series and the number of knots in the B-spline series. We will call the order of polynomials in power series and the number of knots in B-spline *the number of terms* in nonparametric series estimations.

Following Li (1987) we consider three popular criteria in choosing the optimal number of terms in the power spline and B-spline: (a) The Mallow's C_L method (or C_p, Mallow, 1973); (b) The generalized cross-validation (GCV) proposed by Craven and Wahba (1979);

(c) The leave-one-out cross-validation (Stone, 1974). See Li (1987), or Li and Racine (2007) for detailed description of these procedures.

We use the in-sample mean square error to measure the performance of an estimator:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}[g(x_i)^2 - \widehat{g}(x_i)]^2. \tag{4.2}$$

For 1,000 replications we obtain 1,000 sample MSE's as given by (4.2). We then sum over these 1,000 MSE's and divide the sum by 1,000; i.e., we compute the average MSE (AMSE) and report the AMSE, along with the median of the 1,000 MSEs.

4.1. DGPs

We consider a few different data generating processes (DGPs) in the Monte Carlo simulations which are all nested in the data generating process below.

The general DGP used in the simulations is given by

$$\begin{aligned} y_i = {} & \alpha_0 + \alpha_1 x_{i1} + \alpha_2 x_{i1}^2 + \alpha_3 \sin(4\pi x_{i1}) + \beta_1 x_{i2} + \beta_2 x_{i2}^2 + \beta_3 \sin(4\pi x_{i2}) + \gamma_1 x_{i3} \\ & + \gamma_2 \sin(4\pi x_{i3}) + \delta_1 x_{i1}^2 x_{i2} + \delta_2 \sin(x_{i1} x_{i2}) + \delta_3 z_i + u_i, \end{aligned} \tag{4.3}$$

where $x_{is} \sim$ uniform$(0, 1)$ for $s = 1, 2, 3$, $u_i \sim N(0, 1)$, z_i is a binary variable taking values in $\{0, 1\}$ with $P(z_i = 0) = p(z_i = 1) = 1/2$. Depending on the values of the parameters, (4.3) encompasses a linear model, a partially linear model, and a nonlinear model. We vary the values of the parameters as well as the number of observations and the number of replications in our experiments. It appears that the results are not sensitive to the number of replications. When the number of replications is more than 200, the results are stable. All the results reported in this paper are based on 1000 replications. The number of observations $n = 100$ and $n = 200$. The MSE in equation (4.2) is used to measure the performance of different estimators. We report the mean and median of the 1,000 MSEs in the tables.

We will estimate both a univariate and a bivariate nonparametric regression function. The true DGP, however, can be different from the estimated model. For example if the true regression is univariate and we estimate a bivariate regression model, we say that we estimate an over-specified model. Let $\hat{y}_i$ be the nonparametric fitted value of y_i, we compute the goodness-of-fit R^2 by $R^2 = 1 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 / \sum_{i=1}^{n}(y_i - \bar{y})^2$, where $\bar{y} = n^{-1}\sum_{i=1}^{n} y_i$. The R^2 for various DGPs considered in this paper range from 0.46 to 0.72.

Table 1. In-Sample AMSE of Univariate Estimators

	B Spline			Power Spline			Kernel	k-nn	k-nn
	C_L	GCV	LSCV	C_L	GCV	LSCV		Uniform	Gaussian
$n = 100$									
Model is correctly specified. ($\alpha_3 = 2$)									
Mean	0.112	0.112	0.112	0.477	0.478	0.478	0.127	0.136	0.137
Median	0.108	0.110	0.105	0.474	0.474	0.478	0.117	0.136	0.128
Model is over-specified. (all parameters are zeros)									
Mean	0.045	0.043	0.038	0.038	0.036	0.037	0.018	0.029	0.029
Median	0.039	0.038	0.034	0.025	0.023	0.023	0.008	0.018	0.016
$n = 200$									
Model is correctly specified. ($\alpha_3 = 2$)									
Mean	0.089	0.091	0.092	0.181	0.182	0.182	0.827	0.073	0.078
Median	0.087	0.090	0.091	0.177	0.176	0.177	0.069	0.068	0.086
Model is over-specified. (all parameters are zeros)									
Mean	0.020	0.019	0.019	0.023	0.024	0.012	0.010	0.017	0.029
Median	0.015	0.017	0.017	0.014	0.013	0.012	0.011	0.010	0.009

4.2. Estimation of a Univariate Nonparametric Regression Model

In this subsection we consider the case of estimating an univariate regression model. The true model, however, can be univariate, or a white noise, or a bivariate regression model.

4.2.1. The true model is univariate ($\alpha_3 = 2$) We first consider the case where $\alpha_3 = 2$ and all other coefficients are equal to zero. Therefore, the true regression function is only a function of x_{1i} but we estimate the univariate regression model $E(y_i|x_{1i})$ nonparametrically. In this case the model is correctly specified, the performance of various methods are close to each other as we can see from Table 1. The B-spline estimator performs best in most cases, closely followed by the kernel and the k-nn estimators. In other DGPs, such as the exponential function, we do find that the power spline performs better than the B-spline sometimes. It appears that no method dominates the others. When the sample size n is doubled from 100 to 200, the AMSE of all estimators fall by 20% to 50%, showing the consistency of all estimators.

4.2.2. The true model is a white noise process (all parameters are zeros) When all parameters are set to be zero, we have $y_i = u_i$, a white noise process. However, we estimate the function $E(y_i|x_{i1})$ nonparametrically. Thus, we estimate an over-specified regression model, i.e., the DGP is white noise but we apply the univariate estimation method to estimate the conditional mean function, the k-nn (regardless of its weight function) and the kernel perform much better than the B-spline and the power spline, as we can see from Table 1. The superior performance of the k-nn and the kernel estimators over the series estimators is due to the fact that both the k-nn and the kernel methods can smooth out irrelevant regressors while the series estimators do not possess this property. Also, it appears that the k-nn estimator is not sensitive to its weight function being uniform or Gaussian.

Table 2. In-Sample AMSE of Bivariate Additive Estimators

	B Spline			Power Spline			Kernel	k-nn	k-nn
	C_L	GCV	LSCV	C_L	GCV	LSCV		Uniform	Gaussian
$n = 100$									
Model is correctly specified. ($\alpha_3 = -1, \beta_3 = 1$)									
Mean	0.327	0.329	0.331	0.344	0.341	0.342	0.199	0.256	0.236
Median	0.317	0.318	0.326	0.337	0.337	0.334	0.200	0.254	0.228
Model is over-specified. ($\alpha_3 = 2$)									
Mean	0.181	0.180	0.181	0.125	0.128	0.128	0.124	0.096	0.102
Median	0.172	0.172	0.172	0.123	0.142	0.148	0.149	0.081	0.117
$n = 200$									
Model is correctly specified. ($\alpha_3 = -1, \beta_3 = 1$)									
Mean	0.134	0.135	0.134	0.118	0.134	0.141	0.121	0.116	0.117
Median	0.133	0.133	0.132	0.117	0.147	0.148	0.120	0.119	0.112
Model is over-specified. ($\alpha_3 = 2$)									
Mean	0.251	0.250	0.251	0.259	0.258	0.258	0.149	0.211	0.190
Median	0.243	0.243	0.241	0.253	0.251	0.248	0.138	0.199	0.173

4.3. Estimation of a Bivariate Additive Model

In this subsection we estimate an additive model of the following form:

$$y_i = g_1(x_{i1}) + g_2(x_{i2}) + u_i.$$

However, the true regression model may be a univariate, or an additive model, or a bi-variate non-additive (with interaction terms) model.

4.3.1. The true model is bivariate additive ($\alpha_3 = -1, \beta_3 = 1$) We choose $\alpha_3 = -1$ and $\beta_3 = 1$ and all other parameters are set to be zero, so that the true regression model is an additive model (additive in x_{i1} and x_{i2}). When the estimated model is correctly specified, the performance of various methods are close to each other, as we can see from Table 2. In general the kernel and the k-nn estimators perform slightly better than the series-based estimators.

4.3.2. The true model is univariate ($\alpha_3 = 1$) We choose $\alpha_3 = 1$ and all other parameters are zero. In this case we estimate an over-specified model, i.e., the DGP is univariate but we estimate a bivariate additive model. As can be seen from Table 2, the kernel and the k-nn estimators perform much better than the B-spline and the power spline estimators. The reason is that the kernel and the k-nn estimators can automatically remove irrelevant regressors (x_{i3}) while the series estimators cannot.

4.4. Estimation of a Bivariate Fully Nonparametric Model

In this subsection we estimate a bi-variate nonparametric regression model: $y_i = g(x_{i1}, x_{i2}) + u_i$. However, the true regression model can be a univariate, a bi-variate, or a tri-variate regression model.

Table 3. In-Sample AMSE of Bivariate Fully Nonparametric Estimators

	B Spline			Power Spline			Kernel	k-nn	k-nn
	C_L	GCV	LSCV	C_L	GCV	LSCV		(Uniform)	(Gaussian)
$n = 100$									
Model is correctly specified. ($\alpha_3 = -1, \beta_3 = 1, \delta_1 = 0.5$)									
Mean	0.337	0.337	0.338	0.335	0.335	0.338	0.262	0.259	0.236
Median	0.330	0.333	0.336	0.328	0.329	0.331	0.247	0.249	0.227
Model is over-specified. ($\alpha_3 = 2$)									
Mean	1.356	1.356	1.356	0.275	0.275	0.295	0.137	0.213	0.192
Median	1.373	1.373	1.373	0.192	0.192	0.192	0.124	0.207	0.179
$n = 200$									
Model is correctly specified. ($\alpha_3 = -1, \beta_3 = 1, \delta_1 = 0.5$)									
Mean	0.134	0.135	0.134	0.118	0.134	0.141	0.121	0.116	0.117
Median	0.133	0.133	0.132	0.117	0.147	0.148	0.120	0.119	0.112
Model is over-specified. ($\alpha_3 = 2$)									
Mean	1.113	1.112	1.113	0.127	0.115	0.125	0.077	0.079	0.082
Median	1.115	1.113	1.117	0.087	0.081	0.076	0.074	0.078	0.089

4.4.1. The true model is bivariate with interactions ($\alpha_3 = -1, \beta_3 = 1, \delta_1 = 0.5$) We choose $\alpha_3 = -1$, $\beta_3 = 1, \delta_1 = 0.5$ and all other parameters are set to be zero. Hence, the estimated model is correctly specified. From Table 3 we see that the k-nn and kernel methods perform better than the spline methods.

4.4.2. The true model is univariate ($\alpha_3 = 2$) With only $\alpha_3 = 2$ being the non-zero parameter, the estimated model is over-specified, i.e., the DGP is univariate but we estimate a bivariate fully nonparametric model. The kernel estimator performs best, followed by the k-nn (Gaussian and uniform weigth functions), the B-spline and the power spline, as can be seen from Table 3.

4.5. The Mixed Discrete and Continuous Regressors Case

In this section we consider estimating a nonparametric regression model with mixed discrete and continuous regressors. The model we will estimate is $y_i = g(x_{i1}, z_i) + u_i$. However, this may be the true model, or this may be an over-specified model, i.e., the true model might be a univariate regression model $y_i = g(x_{i1}) + u_i$.

4.5.1. The true model is a bi-variate mixed regressor model ($\alpha_3 = 2$, $\delta_3 = 1.5$) In this case the estimated model is correctly specified. Table 4 reports the results for $n = 100$ and $n = 200$. We observe that the performances of the B spline, the k-nn and the kernel estimators are all similar to each other. All of them perform much better than the power series estimators.

4.5.2. The true model is a univariate model ($\alpha_3 = 2$) In this case the true model is univariate but we estimate a bivariate regression model with one continuous regressor (the relevant variable x_{i1}) and one discrete regressor (the irrelevant variable z_i). Therefore, the true model is over-specified. The best performers are the k-nn and the kernel estimators, followed by the B spline, and finally, the power spline estimator. Again, the reason that

Table 4. In-Sample AMSE of Nonparametric Estimators with Mixed Regressors

	B Spline			Power Spline			Kernel	k-nn	k-nn
	C_L	GCV	LSCV	C_L	GCV	LSCV		(Uniform)	(Gaussian)
$n = 100$									
Model is correctly specified. ($\alpha_3 = 2, \delta_3 = 1.5$)									
Mean	0.235	0.234	0.234	0.416	0.432	0.428	0.250	0.248	0.246
Median	0.225	0.224	0.224	0.358	0.365	0.349	0.241	0.241	0.242
Model is over-specified. ($\alpha_3 = 2$)									
Mean	0.223	0.223	0.224	0.409	0.428	0.420	0.177	0.183	0.191
Median	0.219	0.220	0.219	0.349	0.362	0.346	0.165	0.166	0.189
$n = 200$									
Model is correctly specified. ($\alpha_3 = 2, \delta_3 = 1.5$)									
Mean	0.132	0.133	0.133	0.347	0.348	0.345	0.124	0.129	0.146
Median	0.149	0.151	0.151	0.241	0.232	0.239	0.118	0.121	0.143
Model is over-specified. ($\alpha_3 = 2$)									
Mean	0.129	0.130	0.130	0.335	0.337	0.329	0.102	0.109	0.097
Median	0.143	0.149	0.148	0.235	0.229	0.234	0.091	0.098	0.089

the series-based estimators is inferior to either the k-nn or the kernel estimators is that, the series-based estimators cannot smooth out irrelevant discrete variables while both the k-nn and the kernel estimators have the ability of removing irrelevant discrete and continuous variables.

5. CONCLUSIONS

One conclusion from our theoretical analysis and the simulations is that both the k-nn and the kernel methods can automatically remove irrelevant variables and both perform well compared to the series method when the model is over-specified. Furthermore, our simulation results show that the k-nn method is not sensitive to the weight functions used. A referee suggested to us that the approach of Abramson (1984) might be useful in reducing the estimation bias in k-nn regression function estimations. Abramson considered the nonparametric density estimation problem and suggested some transformation methods so that the density function for the transformed data is relatively smoother than the density function for the original data, and therefore the estimation bias can be reduced. It is not clear to us whether Abramson's approach can be generalized to the regression model case because the unknown regression function may still possess the same degree of smoothness. We leave this as a future research topic. A more important future research topic is to develop new series-based nonparametric estimation methods that can automatically remove irrelevant variables.

ACKNOWLEDGEMENTS

The authors are grateful to two referees and a co-editor for their comments that helps to improve the paper. Rui Li's research is supported by the National Natural Science Foundation of China (project serial number: 70573057). Guan Gong received financial support from the Pujang Project sponsored by the City of Shanghai.

REFERENCES

Abramson, I.S. (1984), Adaptive density flatting: A metric distortion principle for combating bias in nearest neighbor methods. *Annals of Statistics* 12, 880-886.

Craven, P., and G. Wahba (1979), Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik* 31: 377-403

Hall, P., Q. Li, and J. Racine (2007), *Nonparametric Estimation of Regression Functions in the Presence of Irrelevant Variables*, forthcoming in *Review of Economics and Statistics* 89, 784-789.

Li, K.C. (1987), Asymptotic optimality for C_p, C_L, cross-validation, and generalized cross-validation: discrete index set, *Annals of Statistics* 15: 958-975.

Li, Q., and J.S. Racine (2007), *Nonparametric Econometrics: Theory and Practice*, Princeton University Press.

Li, Q., J. Racine, and J. Wooldridge *Efficient Estimation of Average Treatment Effects With Mixed Categorical and Continuous Data*, forthcoming in *Journal of Business and Economics Statistics*.

Mallow, C.L. (1973), Some comments on C_p, *Technometrics* 15: 661-675.

Ouyang, D., D. Li, and Q. Li (2006), Cross-Validation and Nonparametric K nearest neighbor Estimation, *Econometrics Journal* 9, 448-471.

Stone, C.J. (1974), Cross-validatory choices and assessment of statistical predictions, *Journal of the Royal Statistical Society*, Series B 36: 111-147.

APPENDIX: PROOF OF LEMMA 3.1

Proof of Lemma 3.1 (i): In order to smooth out x_s from $\hat{g}(x)$, one must have $w((X_{is} - x_s)/R_{x,s}) = c$ for all values of X_{is}, $x_s \in \mathcal{S}_s$. Given that $\mathcal{S}_s$ is a compact set and without loss of generality we can assume that $\mathcal{S}_s = [0, 1]$ (the unit interval). Then we know if $k_s = n$, then $|(X_{js} - x_s)/R_{n,s}| \leq 1$ for all $j = 1, ..., n$. Thus, we have $w((X_{js} - x_s)/R_{n,s}) = (1/2)$, a constant, for all values of X_{js} and x_s. Therefore, $\hat{g}(x)$ is unrelated to x_s.

Proof of Lemma 3.1 (ii): If $w(\cdot)$ is not a uniform weight function, then for $k_s = n$, we know that $R_{x,s}$ is finite. Hence, $0 \leq |(X_{js} - x_s)/R_{x,s}| \leq 1$ and that $w((X_{js} - x_s)/R_{x,s})$ is not a constant function because $|(X_{js} - x_s)/R_{x,s}|$ is not a constant and $w(\cdot)$ is not a uniform weight function. Hence, $\hat{g}(x)$ must depend on x_s, i.e., x_s cannot be completely smoothed for $k_s = n$. Similarly, one can show that for all $k_s \in \{2, ..., n-1\}$, x_s cannot be completely smoothed out from $\hat{g}(x)$.