Understanding Kernels Applied in Econometrics: A Comparison Perspective Bridging Econometrics and Machine-learning [⋆]

Yaohan Chen

School of Economics, Singapore Management University

Last version: February 23, 2021

This version: March 22, 2021

Comments are welcome, currently not for circulation

Abstract

Nonparametric econometrics constitutes the pivotal part of modern econometrics. By far the major nonparametric methodologies documented in literature stem from kernel or sieve techniques broadly studied in statistical and econometrics literature. The reason why nonparametric econometrics is important is not only for modelling flexibility along with it but also for the corresponding perspectives in terms of interpreting some long-standing issued statistics. Among which kernel-based methods justify many of the discussions and the way how kernel functions work is possible to be interpreted may vary from conventional econometrics perspective and modern machine-learning (rigorously speaking statistical learning) perspective. This note will review and compare Kernel techniques applied in nonparametric econometrics both from conventional econometrics perspective and modern machine-learning perspective.

Keywords: Kernels; Nonparametric econometrics; Mercer's theorem; Machine-learning; SVMs

[⋆] I am grateful for the discussion kindly shared literatures with Associate Professor Tao Zeng from Zhejiang University. All errors are my own. You may contact me at yaohan.chen.2017@phdecons.smu.edu.sg.

1 Introduction

Nonparametric econometrics constitutes one major part of modern econometrics and most conventional nonparametric modelling is either based upon kernel method or sieve method. Among which kernel method is relatively a more well-developed either in terms of the longer history of being discussed in literature or the corresponding backbone theories. Although various kernels (kernel functions) as one of the major nonparametric tool have been widely investigated and employed within academic studies or applications in practice, documented either in standard econometrics textbooks (see for instance, Pagan and Ullah, 1999; Li and Racine, 2006; Racine, 2019) or the related research papers, rarely is there any discussion about the mechanism through which kernels work jointly from the theoretical and practical perspective, especially the way to interpret kernels applied in econometrics from novel machine-learning (statistical leaning) perspective within the nonparametric regression context. Consequently, this note attempts to review kernels applied in econometrics, especially nonparametric econometrics for functional estimation and compare kernels both from conventional econometrics perspective and novel machine-learning perspective. Hopefully this piece of work would shed some light on our understanding, as econometricians, of kernels (kernel functions) applied in econometrics, especially for those who are interested in bridging conventional econometric modelling with novel machine-learning methodologies.

The following discussion of this note is going to focus on two parts: Section 2 will review kernels and the major mechanism through which kernels work from conventional econometrics perspective; Section 3 will alternatively provide way justifying the application of kernels from machine-learning (statistical learning) perspective along with some mathematical results (mainly about Mercer's theorem) that serves as the theoretical ground for the corresponding analysis. Related discussions about Monte Carlo experiments and practical applications will also be included each as one subsection of this part as well. Finally Section 4 concludes the discussion made in this note.

2 Kernels Interpreted from Conventional Econometrics Perspective

As commonly documented in statistics and econometrics literature, kernels (or kernel functions) serves as way for estimating density function (see Rosenblatt, 1956; Parzen, 1962; Fix and Hodges, 1989). Pagan and Ullah (1999) makes a relatively discussion summarizing some well-known density estimators, among which kernel density estimator is the best known not only for theoretical researchers but also for researchers focusing applied researches.

The pivotal question to be addressed in terms of density function estimation is as following (temporarily for discussion simplicity we may focus on the univariate case): suppose that the observed data $\{X_i\}_{i=1}^n$ follows distribution with cumulative distribution function (CDF) $F(x)$ along with the corresponding probability density function (PDF) $f(x)$. It is well-known from standard textbook (see Durrett, 2019) that under some regular conditions either $F(x)$ or $f(x)$ provides necessary information for describing distribution, which naturally motivates estimating $f(x)$ from data, i.e $\{X_i\}_{i=1}^n$. The way commonly adopted in literature for handing estimation of this kind via kernels relies on constructing estimator of $f(x)$ taking the following functional form

$$\begin{aligned} \widehat{f}(x) &= \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{X_i - x}{h}\right) \\ &= \frac{1}{n}\sum_{i=1}^{n} k_h\left(X_i - x\right) \end{aligned} \tag{1}$$

where $k(\cdot)$ refers to "kernel" function of this context, which is real-valued function over $\mathbb{R}$, and h commonly refers to "smoothing parameter" or "bandwidth" in literature. Obviously from density function estimator as in (1) using kernels, we may regard it as the sample average of kernels evaluated at different $X_i - x$. Interpreting density function estimator from this perspective actually sheds light on how kernels are applied in nonparametric-regression framework later to be discussed.

In analogy to the proceeding discussed density function estimation, a more general and more practical framework is nonparametric regression in which the pivotal modelling structure can be parsimoniously summarized as following

$$Y_i = m(X_i) + \epsilon_i, \quad i = 1, \ldots, n \tag{2}$$

where X_i as the $p \times 1$ vector collecting all the corresponding covariates, ϵ_i as usual collects error terms (or noise terms) which satisfy $\mathbb{E}[\epsilon_i] = 0$ and $\mathbb{E}\left[\epsilon_i^2 \mid X_i\right] = \sigma^2(X_i)$ [1]. Actually, (2) serves as a general representation for most of the statistical modelling. For instance, linear regression is an affine special case of $m(X_i)$ such that $m(X_i) = X_i^\top \boldsymbol{\beta}$, where $\boldsymbol{\beta}$ as the $p \times 1$ vector collecting all the regression coefficients. Hence, unravelling the functional form $m(\cdot)$, i.e., seeking for the appropriate estimator $\widehat{m}(\mathbf{x})$ of $m(\mathbf{x})$, is the central issue to be addressed within the nonparametic regression context. Where we employ the bold face letter $\mathbf{x}$ to denote the population counterpart of observed covariates X_i and for the degenerate case where $\mathbf{x}$ degenerates to a scalar x, the objective of econometricians is the same as that in the density function estimation.

Discussion corresponding to nonparametric regression using kernels primarily stems from the following constructed estimator (likewise we temporarily focus on the univariate case for the sake of discussion simplicity)

$$\widehat{m}(x) = \frac{\sum_{i=1}^n Y_i k_h\left(X_i - x\right)}{\sum_{i=1}^n k_h\left(X_i - x\right)} \tag{3}$$

and obviously (3) can be interpreted intuitively as the weighted average of observations $\{Y_i\}_{i=1}^n$ using the weights constructed using function value of kernels (kernel functions) evaluated at observed covariates $\{X_i\}_{i=1}^n$. As discussed in Nadaraya (1964) and Watson (1964), functional estimator of

[1] Here for generality concern, we introduce notation $\sigma^2(X_i)$ to emphasize either heteroskedasiticty or homoscedasticity are allowed, depending on whether $\sigma^2(X_i)$ varies across X_i or remains as a constant.

this form can be easily extended the multivariate case where for each i, the observed covariate X_i is a $p \times 1$ vector. In the following part, I will demonstrate why it should be the case heuristically under some implicit regular assumptions. Given the modelling structure specified as in (2), $m(\mathbf{x})$ is possible to be interpreted as the conditional mean of y given $\mathbf{x}$, which further implies that

$$m(\mathbf{x}) = \mathbb{E}(y \mid \mathbf{x}) = \frac{\int y f(y, \mathbf{x}) dy}{f(\mathbf{x})} \tag{4}$$

(4) motivates the construction of $\widehat{m}(\mathbf{x})$ by replacing $f(y, \mathbf{x})$ and $f(\mathbf{x})$ with their nonparametric kernel density estimator $\widehat{f}(y, \mathbf{x})$ and $\widehat{f}(\mathbf{x})$ in (4) respectively. This constructed estimator takes the following form, [2]

$$\widehat{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} Y_i K_{\mathbf{h}}(X_i - \mathbf{x})}{\sum_{i=1}^{n} K_{\mathbf{h}}(X_i - \mathbf{x})} \tag{5}$$

which is comparable to the univariate case as in (3). Detailed demonstration corresponding to why it should be the case is left in A.1.

3 Kernels Interpreted from Machine-learning Perspective

3.1 Mathematical ground

Mercer's theorem (Mercer, 1909; König, 1986) plays the fundamental role in justifying the application of kernels in either statistical or econometrics modelling, hence we formally state this theorem and the corresponding results as following.

Theorem 3.1 (Mercer (1909)) *Assume $(\mathcal{X}, \mu)$ is well-defined finite measure space. Suppose $k \in L_\infty(\mathcal{X}^2)$ is a symmetric real-valued function such that integral operator*

$$\begin{aligned} T_k \ &: \ L_2(\mathcal{X}) \to L_2(\mathcal{X}) \\ (T_k) f(x) \ &:= \ \int_{\mathcal{X}} k(x, x') f(x') d\mu(x') \end{aligned} \tag{6}$$

is positive definite; that is, for all $f \in L_2(\mathcal{X})$, we have [3]

$$\int_{\mathcal{X}^2} k(x, x') f(x) f(x') d\mu(x) d\mu(x') \geqslant 0 \tag{7}$$

Let $\psi_j \in L_2(\mathcal{X})$ be the normalized orthogonal eigenfunctions of T_k associated with eigenvalues $\lambda_j > 0$, sorted in non-increasing order. Then

[2] Here to emphasize the difference int he context of multivariate setting that kernels (kernel functions) employed are multivariate functions, I separately introduce the the notation $K_{\mathbf{h}}(\cdot)$ to denote kernels (kernel functions) as the multivariate functions where $\mathbf{h} = (h_1, \ldots, h_p)$ refers to the vector of dimension p collecting chosen bandwidths.

[3] It should be noted that here we do not use bold face letter $\mathbf{x}$ to emphasize that it is a vector for the multivariate case but just x to denote the element from the generally defined measure space $\mathcal{X}$. Hence both x a x' can either be vector for the multivariate case or scalar for the univariate case.

1. $\{\lambda_j\}_j \in \ell_2$

2. $k(x, x') = \sum_{j=1}^{N_{\mathcal{H}}} \lambda_j \psi_j(x) \psi_j(x')$. *Either* $N_{\mathcal{H}} \in \mathbb{N}$ *(where* $\mathbb{N}$ *refers to the set of positive integers) or* $N_{\mathcal{H}} = \infty$. *For the case* $N_{\mathcal{H}} = \infty$, *the series converges absolutely and uniformly for almost all* (x, x').

Implications from Theorem 3.1 are actually very rich (see in the standard textbooks, Vapnik, 1998; Hastie et al., 2001, for more comprehensive results) and among which the most important one is derived Mercer Kernel Map, which essentially stems from the second claim of Theorem 3.1. The following discussion summarizes definition of Mercer Kernel Map and the associated properties.

Definition 1 (Mercer Kernel Map) *Mercer Kernel Map, denoted by* Φ, *is defined as the map from measure space* $\mathcal{X}$ *to* $\ell_2^{N_{\mathcal{H}}}$ *such that* [4]

$$\begin{aligned} \Phi : \mathcal{X} &\to \ell_2^{N_{\mathcal{H}}} \\ x &\mapsto \{\sqrt{\lambda_j}\psi_j(x)\}_{j=1,\ldots,N_{\mathcal{H}}} \end{aligned} \tag{8}$$

Corollary 3.1 *If a kernel (kernel function) satisfy the conditions required in the Theorem 3.1, then Mercer Kernel Map is possible to constructed such that the positive-definite kernel* $k(\cdot,\cdot)$ *can be expressed as the dot-product in high-dimensional space. Thus*

$$\langle \Phi(x), \Phi(x') \rangle = k(x, x') \tag{9}$$

for almost $x, x' \in \mathcal{X}$. *Moreover, for any given* $\epsilon > 0$, *there exists a finite map* Φ^{N_ϵ} *to n-dimensional dot product space* [5] *such that*

$$\left| \langle \Phi(x), \Phi(x') \rangle - k(x, x') \right| < \epsilon$$

where Φ^{N_ϵ} *is as following*

$$\Phi^{N_\epsilon} : x \mapsto \left\{ \sqrt{\lambda_1}\psi_1(x), \ldots, \sqrt{\lambda_{N_\epsilon}}\psi_{N_\epsilon}(x) \right\}$$

Theorem 3.1 along with the Corallay 3.1 lays the foundation for the application of kernels in nonparametric regression setting, especially from the machine-learning (statistical-learning) perspective. In the following I will discuss how SVMs (Support Vector Machines), one pioneering but still prevalent machine-learning method, are theoretically-grounded with the basis as Mercer's theorem and how SVMs are connected with nonparametric regression.

[4] $\ell_2^{N_{\mathcal{H}}}$ refers to the space of square-summable sequences, that is Hilbert space of dimension N_ϵ.

[5] I want to emphasize here that $N_\epsilon \in \mathbb{N}$ for this setting refers to the dimension of corresponding dot product space and varies across ϵ.

3.2 Kernel-based methods as the classifier

Perhaps the most prevalent application of SVMs is classification. I will discuss how kernels (kernel functions) play the role in the classification problem and this would also shed light on the interpretation of kernels from machine-learning perspectives later within the more general nonparametric setting. Essentially speaking all the classification problem about pairwise comparison and hence binary classification constitutes the atom of the most of classification algorithms documented in literature, so is the kernel-based machine-learning algorithm. Extension from binary classification to the general multi-class classification is not seemingly hard and has been well discussed in some pioneering researches (Platt, 1999; Lin et al., 2007). Actually binary classification is easily to be embedded in the setting as in (2) by confining values of Y_i as binary variables such that $Y_i = \{\pm 1\}$. Accordingly within the nonparametric regression framework for classification, the general target of econometricians is to seek appropriate estimation of $\widehat{m}(\mathbf{x})$. It would be worthwhile to emphasize that in comparison to the kernel estimator constructed from conventional econometrics perspective, the kernel estimator constructed from machine-learning perspective does not necessarily require the used kernels (kernel functions) to be normalized. As suggested by Mercer's theorem, it is possible to map the feature vector (i.e. covariates $\mathbf{x} \in \mathcal{X}$) to another augmented feature vector $\mathbf{z} \in \mathcal{H}$, where $\mathcal{X}$ and $\mathcal{H}$ generally refers to well-defined measure space (specifically for most of the practical applications, $\mathcal{X}$ and $\mathcal{H}$ are Euclidean spaces of dimension $N_{\mathcal{X}}$ and $N_{\mathcal{H}}$ respectively). That is

$$\mathbf{z} = \Phi(\mathbf{x})$$

One key assumption (or alternatively it should be regarded as the implication from Mercer's theorem) for the kernels (kernel functions) to work in the context of classification is that data is able to be classified based on the feature vector $\mathbf{x} \in \mathcal{X}$ as long as it is able to be classified based on the augmented feature vector $\mathbf{z} \in \mathcal{H}$ and it is assumed that the augmented feature space features affine structure and hence classification in $\mathcal{H}$ relies on specific hyperplane equipped with parameters $\mathbf{w} \in \ell_2^{N_{\mathcal{H}}}$ and $b \in \mathbf{R}$, which keeps the following affine structure

$$\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b = 0 \tag{10}$$

With this readily specified hyperplane (namely the support vector), binary classification implemented in $\mathcal{H}$ is based on the following classification rule

Proposition 3.1 *For any observed data with feature vector* $\mathbf{x}$*, hyperplane (support vector) categorizes data into different classes based on the following decision function*

$$f(\mathbf{x}) = \operatorname{sign}\left(\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b\right) \tag{11}$$

where

$$\operatorname{sign}(u) = \begin{cases} 1 & \textit{if } u \geqslant 0 \\ -1 & \textit{if } u < 0 \end{cases}$$

Proposition 3.1 suggests the rule to be applied for binary classification using hyperplane but says nothing bout how the associated parameters $\mathbf{w}$ and b should be tuned. The following proposition how $\mathbf{w}$ and b are determined within optimization framework and from which it is possible see how kernels (kernel functions) apply.

Proposition 3.2 *Along with the suggested in classification rule in Proposition 3.1, hyperplane parameters* $\mathbf{w}$ *and* b *are determined as the solutions of the following optimization problem.*

$$\underset{\mathbf{w}\in\ell_2^{N_{\mathcal{H}}},b\in\mathbf{R}}{\text{minimize}} \quad \tau(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2 \tag{12}$$

$$\text{subject to} \quad Y_i * [\langle \mathbf{w}, \Phi(X_i)\rangle + b] \geqslant 1 \quad \textit{for all} \quad i = 1, \ldots, n \tag{13}$$

where $\ell_2^{N_{\mathcal{H}}}$ *as usual refers to the square-summable space equipped with norm* $\|\cdot\|_2$.

In the following part of this note, I will discuss why it is the case that $\mathbf{w}$ and b are determined as the way suggested from Proposition 3.2. Without loss of generality, we may assume that X_1 and X_2 are two points belonging to two different sets separated by hyperplane $\langle \mathbf{w}, \Phi(\mathbf{x})\rangle + b$. Moreover, for the sake of simplicity we may assume that X_1 and X_2 are the points closet to the hyperplane $\langle \mathbf{w}, \Phi(\mathbf{x})\rangle + b$ in dot product space $\mathcal{H}$. It should be kept in mind that for the temporarily discussed binary classification problem, the sign of values taken by $\langle \mathbf{w}, \Phi(X_i)\rangle + b$ determines which set into which data is to be categorized by this hyperplane and accordingly we may focus on the discussion such that $\langle \mathbf{w}, \Phi(X_i)\rangle + b = 1$ when $\langle \mathbf{w}, \Phi(X_i)\rangle + b$ takes positive values and likewise $\langle \mathbf{w}, \Phi(X_i)\rangle + b = -1$ when $\langle \mathbf{w}, \Phi(X_i)\rangle + b$ takes negative values. We summarizes these discussions as the following system of equations,

$$\langle \mathbf{w}, \Phi(X_1)\rangle + b = 1$$

$$\langle \mathbf{w}, \Phi(X_2)\rangle + b = -1$$

$$\langle \mathbf{w}, (\Phi(X_1) - \Phi(X_2))\rangle = 2 \tag{14}$$

$$\Downarrow$$

$$\left\langle \frac{\mathbf{w}}{\|\mathbf{w}\|_2}, (\Phi(X_1) - \Phi(X_2))\right\rangle = \frac{2}{\|\mathbf{w}\|_2} \tag{15}$$

where from (14) to (15), we just use the common affine property of Hilbert space. Moreover, the L.H.S. of (15) is nothing but the projection of $(\Phi(X_1) - \Phi(X_2))$ on the unitary normal vector $\frac{\mathbf{w}}{\|\mathbf{w}\|_2}$, hence it can be geometrically interpreted as Euclidean distance between X_1 and X_2 over the direction specified by normal vector $\mathbf{w}$ if we restrict the corresponding Hilbert space to be Euclidean space. This also suggests why $\tau(\mathbf{w})$ in (12) should be the objective function of the associated minimization problem as minimizing $\tau(\mathbf{w})$ is equivalent to maximizing the distance between two sets separated by the desired hyperplane.

Remark 3.1

1. *Constraint (13) plays the role guaranteeing that the decision function $f(X_i)$ will take exactly the value of $+1$ for $Y_i = +1$ and other -1 for $Y_i = -1$.*

2. *The lower bound claimed on the R.H.S. of specified constraint (13) does not necessarily take 1 but has to be a positive number since otherwise if this lower bound is specified as 0, it would make no sense to minimize $\tau(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2$. A quick discussion note on this is as following: suppose the the lower bound on the R.H.S. of (13) is replaced with 0 and the $(\mathbf{w}, b)$ serves as the solution of this optimization problem, then any scaled alternative combination such that $(\mathbf{w}', b') = \lambda(\mathbf{w}, b)$ with $0 < \lambda < 1$ suggests that $\|\mathbf{w}'\|_2 < \|\mathbf{w}\|_2$ while (13) is still satisfied, hence the corresponding optimization problem is not well-defined.*

To obtain the solution of optimization problem specified in (12) and (13) so that we may identify the functional form of the decision function $f(\mathbf{x})$, we set up the standard Lagrangian as following

$$\mathcal{L}(\mathbf{w}, b) := \frac{1}{2}\|\mathbf{w}\|_2^2 - \sum_{i=1}^{n} \alpha_i \left[Y_i * (\langle \mathbf{w}, \Phi(X_i) \rangle + b) - 1 \right] \tag{16}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ as the vector of n-dimension collecting Lagrangian-multipliers. The first order derivative taken with respect to $\mathbf{w}$ and b respectively yields the standard F.O.C. as following

$$\begin{cases} \dfrac{\partial}{\partial b}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) & = & 0 \\ \dfrac{\partial}{\partial \mathbf{w}}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) & = & 0 \end{cases} \Rightarrow \begin{array}{rcl} \sum_{i=1}^{n} Y_i \alpha_i & = & 0 \\ \mathbf{w} & = & \sum_{i=1}^{n} \alpha_i Y_i \Phi(X_i) \end{array} \tag{17}$$

and (17) implies that

$$\begin{aligned} & \langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b \\ = & \sum_{i=1}^{n} Y_i \alpha_i \langle \Phi(X_i), \Phi(\mathbf{x}) \rangle + b \end{aligned} \tag{18}$$

which is the desired hyperplane for classification. Moreover, the implications from (18) are summarized as following,

1. As suggested by Mercer's theorem, we may replace $\langle \Phi(X_i), \Phi(\mathbf{x}) \rangle$ with the corresponding kernel (kernel function), thus

$$\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b = \sum_{i=1}^{n} Y_i \alpha_i k(\Phi(X_i), \Phi(\mathbf{x})) + b$$

   which implies the role played by kernels in this setting and how it is comparable to the kernel-based functional estimator constructed from conventional econometrics perspective.

2. Application of standard Lagrangian techniques in solving optimization problem commonly

requires the $\alpha_i \geqslant 0$ $(i = 1, \ldots, n)$ along with the following complimentary conditions (see Rockafellar, 1970, which is referenced as Karush-Kuhn-Tucker, KKT conditions as well)),

$$\alpha_i \left[Y_i \langle \Phi(X_i), \Phi(\mathbf{x} \rangle) + b - 1 \right] = 0, \qquad i = 1, \ldots, n \tag{19}$$

KKT conditions as in (19) along with the F.O.C. constitutes the standard system of equations with $(n + N_{\mathcal{H}} + 1)$ equations and $(n + N_{\mathcal{H}} + 1)$ variables to be solved out, which is theoretically tractable under some regular conditions. Moreover, $\{\alpha_i\}_{i=1}^n$ as the Lagrangian multipliers is comparable to the normalized weights of kernel-based functional estimator constructed from conventional econometrics perspective (specifically the summation of kernel function values evaluated at observed data points).

3. Although the above specified system of equations is theoretically tractable, as suggested in (18), the desired hyperplane for classification is completely determined by $\{\alpha_i\}_{i=1}^n$ and b and hence how to equivalently obtain $\{\alpha_i\}_{i=1}^n$ and b would be of more interests. Fortunately it is justified by the dual theory by the modern convex analysis. Specifically, for the established Lagrangian as in (16), we may represent $\mathbf{w}$ and b in terms of $\{\alpha_i\}_{i=1}^n$ using the F.O.C. in (17) such that for any fixed $\boldsymbol{\alpha}$,

$$\begin{aligned} \mathcal{T}(\boldsymbol{\alpha}) = \mathcal{L} &= \frac{1}{2}\|\mathbf{w}\|_2^2 - \sum_{i=1}^n \alpha_i \left[Y_i * (\langle \mathbf{w}, \Phi(X_i) \rangle + b) - 1 \right] \\ &= \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \langle \Phi(X_i), \Phi(X_j) \rangle - \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \langle \Phi(X_i), \Phi(X_j) \rangle + \sum_{i=1}^n \alpha_i \\ &= \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \langle \Phi(X_i), \Phi(X_j) \rangle \end{aligned}$$

The we may follow the standard dual theory established in convex analysis to set up the associated dual optimization problem as following

$$\underset{\boldsymbol{\alpha} \in \mathbf{R}}{\text{maximize}} \qquad \mathcal{T}(\boldsymbol{\alpha}) = \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j \langle \Phi(X_i), \Phi(X_j) \rangle \tag{20}$$

$$\text{subject to} \qquad \alpha_i \geqslant 0 \quad i = 1, \ldots, n \tag{21}$$

$$\sum_{i=1}^n \alpha_i Y_i = 0 \tag{22}$$

which is standard constrained quadratic optimization problem. And once again we need to replace $\langle \Phi(X_i), \Phi(X_j) \rangle$ with appropriate kernels (kernel functions), which is arguably the place where kernels play the role in this setting.

3.3 Kernel-based methods as the regression

As we mentioned previously, kernel-based analysis discussed in subsection 3.2 is able to be extended to the general nonparametric regression framework, which constitutes the discussion of this part and this is also the what I want to emphasize the most as it nicely sets up the connection between kernels (kernel functions) and nonparametric regression and accordingly suggests the way we interpret the mechanism through which kernels (kernel functions) apply in econometrics (specifically nonparametric regression) from machine-learning (statistical learning) perspective. In terms of modelling framework, Extension from classification to regression just corresponds to relaxing restrictions that $Y_i = \{\pm 1\}$ so that in general Y_i can take any value from $\mathbf{R}$ as response variable. Likewise, we want to make our analysis embedded in the optimization framework but with slightly modified constraints as Y_i as response variable now is allowed to take any value from $\mathbf{R}$.

The idea that using Mercer Kernel Map to transfer the original data to augmented data in the corresponding Hilbert space $\mathcal{H}$ still applies. Then in analogy to the discussion corresponding binary classification, we set up the following optimization problem

$$\underset{\mathbf{w}\in\ell_2^{N_{\mathcal{H}}},b\in\mathbf{R}}{\text{minimize}} \quad \rho(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2 + \frac{C}{2n}\sum_{i=1}^{n}\epsilon_i^2 \tag{23}$$

$$\text{subject to} \quad Y_i = \langle \mathbf{w}, \Phi(X_i)\rangle + b + \epsilon_i \quad i = 1,\ldots,n \tag{24}$$

where C is a positive constant.

Remark 3.2 *As we can see from the above specified optimization problem, both the objective function and the associated constraints are slightly different from that in the previous discussion about kernel-based methods applied in classification, which are summarized as following respectively*

1. *In comparison the objective function $\tau(\mathbf{w})$ (12) as we discussed in the classification problems, the objective function $\rho(\mathbf{w})$ claimed in (23) includes one more added term $\frac{C}{2n}\sum_{i=1}^{n}\epsilon_i^2$ which can be interpreted as the average fitting error.*

2. *As for the associated constraints (24) in comparison to the counterpart in (13), the key difference stems from the introduced terms ϵ_i measuring the fitting error.*

The corresponding Lagrangian is established as following

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\epsilon}, \boldsymbol{\alpha}) \;\coloneqq\; \frac{1}{2}\|\mathbf{w}\|_2^2 + \frac{C}{2n}\sum_{i=1}^{n}\epsilon_i^2 - \sum_{i=1}^{n}\alpha_i\left[\langle \mathbf{w}, \Phi(X_i)\rangle + b + \epsilon_i - Y_i\right] \tag{25}$$

where I introduce the additional notation $\boldsymbol{\epsilon} = (\epsilon_1,\ldots,\epsilon_n)$ and $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_n)$ to denote the vectors collecting error terms and multipliers respectively.

As the above specified optimization problem does not involve inequality constraints, the corresponding F.O.C. is specified as following

$$\left\{\begin{array}{rcl} \dfrac{\partial}{\partial \mathbf{w}}\mathcal{L}\left(\mathbf{w}, b, \boldsymbol{\epsilon}, \boldsymbol{\alpha}\right) & = & 0 \\ \dfrac{\partial}{\partial b}\mathcal{L}\left(\mathbf{w}, b, \boldsymbol{\epsilon}, \boldsymbol{\alpha}\right) & = & 0 \\ \dfrac{\partial}{\partial \epsilon_i}\mathcal{L}\left(\mathbf{w}, b, \boldsymbol{\epsilon}, \boldsymbol{\alpha}\right) & = & 0 \\ \dfrac{\partial}{\partial \alpha_i}\mathcal{L}\left(\mathbf{w}, b, \boldsymbol{\epsilon}, \boldsymbol{\alpha}\right) & = & 0 \end{array}\right. \Rightarrow \begin{array}{rcll} \mathbf{w} & = & \sum_{i=1}^{n} \alpha_i \Phi(X_i) & \quad (26) \\ \sum_{i=1}^{n} \alpha_i & = & 0 & \quad (27) \\ \alpha_i & = & \dfrac{C}{n}\epsilon_i \qquad i = 1, \ldots, n & \quad (28) \\ \langle \mathbf{w}, \Phi(X_i)\rangle + b + \epsilon_i - Y_i & = & 0 \qquad i = 1, \ldots, n & \quad (29) \end{array}$$

This system of equations as specified in (26) to (29) is theoretically well-defined system as it essentially involves 4 sets of equations with 4 sets of unknowns. Specifically to see how the final result (desired hyperplane) takes the input as $\boldsymbol{\alpha}$ and b, we use (26) and (28) to eliminate $\mathbf{w}$ and $\boldsymbol{\epsilon}$. That is for each $i, \ldots, n$, substituting $\mathbf{w}$ and ϵ_i in (29) using (26) and (28) respectively yields

$$\sum_{j=1}^{n} \langle \Phi(X_i), \Phi(X_j)\rangle \alpha_j + b + \frac{n}{C}\alpha_i - Y_i = 0 \tag{30}$$

which can be parsimoniously represented in matrix form as following

$$\begin{bmatrix} 0 & \iota_n^\top \\ \iota_n & \mathcal{K} + \frac{n}{C}\mathbf{I}_n \end{bmatrix} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix} \tag{31}$$

where $\mathcal{K}$ is a $n \times n$ matrix with the (i, j) entry as $\langle \Phi(X_i), \Phi(X_j)\rangle$. Moreover, $\mathbf{y} = (y_1, \ldots, y_n)^\top$ with the i-th element y_i as the realized value of Y_i. Likewise to see how kernels (kernel functions) apply here, we need to replace $\langle \Phi(X_i), \Phi(X_j)\rangle$ with appropriate kernel function $k(\cdot, \cdot)$ evaluated at X_i and X_j as suggested from Mercer's theorem. Finally with $\hat{\boldsymbol{\alpha}}$ and $\hat{b}$ solved out from (31), the estimated functional form of $m(\mathbf{x})$ as following

$$\widehat{m}(\mathbf{x}) = \sum_{i=1}^{n} \hat{\alpha}_i k(\mathbf{x}, X_i) + \hat{b} \tag{32}$$

Remark 3.3

1. *It seems that the estimator constructed in (32) does not involve variable Y_i, but one should keep in mind that $\hat{\boldsymbol{\alpha}}$ is estimated from (31) and hence response variable is implicitly involved.*

2. *We may rewrite (5) and (32) in parallel as following for the sake of comparison,*

Conventional Econometrics	*Machine-learning*
$\widehat{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} Y_i K_{\mathbf{h}}(X_i - \mathbf{x})}{\sum_{i=1}^{n} K_{\mathbf{h}}(X_i - \mathbf{x})}$	$\widehat{m}(\mathbf{x}) = \sum_{i=1}^{n} \hat{\alpha}_i k(\mathbf{x}, X_i) + \hat{b}$

Comparisons made between these two estimated functional forms of $m(\mathbf{x})$ implies that in terms of functional structure, kernel-based estimator of $m(\mathbf{x})$ from conventional Econometrics perspective is similar to that constructed from machine-learning perspective, which can be essentially interpreted as the weighted average of kernel functions evaluated at observed data points. However as we have discussed in the main context, these two estimators are theoretically oriented in different way. Estimators constructed from conventional econometrics perspective is deeply rooted in density function estimation as the main intuition from this perspective is to construct weights using kernels (kernel functions) evaluated observed covariates and accordingly take the weighted average of response as the desired estimation of functional form; while estimators constructed from machine-learning perspective, as we have emphasized in the mathematical ground part, the key idea (alternatively the mathematical foundation) is that Mercer kernel map can map covariates in feature space to augmented Hilbert space in which the transformed data keeps the affine structure and dot product attached to this augmented Hilbert space can be approximately depicted well by corresponding kernels (kernel functions).

3. *As implied from previous discussions, kernels (kernel functions) customarily adopted in standard econometrics analysis, $K_{\mathbf{h}}(\cdot)$, are closely connected with kernels (kernel functions) applied from machine-learning perspective. Specifically both $K_{\mathbf{h}}$ and $k(\cdot,\cdot)$ for the first place have to satisfy the symmetric conditions required. Secondly "smoothing parameter" or "bandwidth" $\mathbf{h}$ (vector $\mathbf{h}$ degenerates to scalar h as modelling setting changes from multivariate to univariate) associated with $K_{\mathbf{h}}(\cdot)$ is comparable to the tuning parameter σ attached to the Gaussian (Laplace) Radial Basis Function (RBF) kernels [6] in the context where kernels are applied to implement nonparametric regression as we discussed in the proceeding main context. Tuning for $\mathbf{h}$ or σ can either be customized specification or empirical estimation based on observed data. Usually σ lies in between 0.1 and 0.9 quantiles of $\|\mathbf{x}-\mathbf{x}'\|_2^2$ hence the thumb rule associated with the later empirically-determined configuration of σ for most of the empirical applications takes the median of 0.1 and 0.9 quantiles based on data (observed covariates in feature space).*

4. *The way we establish Lagrangian as in (25) implicitly suggests that applying kernels in fitting nonparametric regression automatically accommodate the heterogeneity captured by error term as for this case the F.O.C associated with error terms constitutes part of the optimal conditions as summarized from (26) to (29). Actually as the definition of σ suggests, σ^{-1} is comparable to the h for univariate case or the homogeneous choice of bandwidth $h_1 =, \dots, = h_p$ for multivariate case and accordingly configuration of σ determines "smoothness" or the in-sample goodness of fit and this "σ"-effect associated with the choice of kernels are to be demonstrated in the toy practical practical example contained the next subsection.*

[6] Gaussian (Laplace) Radial Basis Function (RBF) kernels are commonly adopted in literature takes the form $\exp(-\sigma\|\mathbf{x}-\mathbf{x}'\|_2^2)$, where σ as the tuning parameter usually refers to "inverse kernel width".

3.4 Monte Carlo and practical examples

For the Monte Carlo demonstration, I use the following DGP (data generating process) to simulate data used as in the SVMs. For the sake of visually demonstrating the learning results as the input from SVMs, I consider the case where $n = 200$ $p = 2$. Specifically, [7]

$$Y_i = m\left(X_i\right) + \epsilon_i = \sin\left(x_{i,1} + x_{i,2}^2\right) + \left(x_{i,1} + x_{i,2}^2\right) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}\left(0, \sigma^2\right), \quad 1 \leqslant i \leqslant n. \tag{33}$$

where $x_{i,j}$ is generated uniformly from $[-2, 2]$ and $\sigma = 0.1$. Obviously $m(\cdot)$ specified in (33) is non-linear and readily suitable for application of nonparametric regression. Data collected in $\{Y_i, X_i\}_{i=1}^n$ serve as the input of SVMs and plot 3-D surface using the learning predictions generated from SVMs. Corresponding results are demonstrated as following and this suggests that kernels commonly employed in conventional nonparametric econometrics studies are in line with kernel-based methodologies (specifically SVMs) but could be interpreted from different perspectives (as the comparison we have discussed in the previous context).

[Place Figure 1 about here]

With the development of modern computational tools, implementing kernel-based machine-learning algorithm within the setting of nonparametric regression is approachable on different platforms, among which `LIBSVM` developed along with Chang and Lin (2011) as one of the pioneering tools package (and by far the most influential one for its widely successful applications practice) provides efficient `C` and `C++` routines for handling required optimization problem using Sequential Minimization Optimization algorithm (SMO). Recently within `R` community (R Core Team, 2020), `kernlab` (Karatzoglou et al., 2004) provides by far the most comprehensive interfaces to `LIBSVM` for implementation of various kernel-based machine-learning algorithms.

3.5 Practical application examples

To be included more comprehensively. But currently here is a quickly demonstrated toy example using data collected in Holst et al. (1996). We demonstrate both the scatter plot of the original data and the fitting curve generated from kernel-based methods applied from machine-learning perspective. Gaussian (Laplace) Radial Basis Function (RBF) kernels are applied with *sigma* specified as $\sigma = 1$.

[Place Figure 2 about here]

[7] To rule out the possible confusion about notation, we emphasize it here that throughout this paper where the random variables are about R.H.S. covariates, we use capital X_i to denote the corresponding i-th observation. X_i could either be scalar (univariate case) for which X_i degenerates to x_i and x_i as a specific number referring to the realized value of X_i; or vector (multivariate case) for which $X_i = (x_{i,1}, \ldots, x_{i,p})$ and $x_{i,j}$ $(1 \leqslant j \leqslant p)$ as a specific number referring to realized value of the j-th covariate of X_i. The reason that we want to distinguish capital letter X from x as different notation is in light of modern views on random variables that random variables (or observed data) should be regarded as measurable functions defined over appropriate measure space with well-defined structure (see, Durrett, 2019).

As we have mentioned in Remark 3.3, the effect of the choice of σ is demonstrated well with this example. In the following example, I apply kernel-base methods on the same dataset but with different configuration of σ with σ specified as 0.1, 10 and the corresponding empirical estimation respectively.

[Place Figure 3 about here]

To demonstrate a relatively more practical application, we turn to focus on the discussion about expected return as the function of firm size (usually measured as the market capitalization, for instance Nard and Zhao (2020)). It corresponds closely to the long-standing discussion voluminously documented in cross-sectional asset pricing literature and firm size serves as one of the pioneering anomalies complementing standard CAPM theory (Fama and French, 1992, 1993, 1996). Ever since then the universe of anomalies documented in literature has expanded much and this induced dimensionality challenge motives the recent focus of academia community on applying new methodologies for handling this issue. Specifically, anomalies often referenced in literature are essentially different portfolios based on sorting on different cross-sectional characteristics (including the time-series dimension lead-lag effect such as anomalies of momentum class). Consequently for each cross-section, we may use the i as the index for cross-sectional stocks and all p characteristics constructed for firm i are collected in $X_i = (x_{i,1}, \ldots, x_{i,p})$, which is consistent with the notation we used in the previous discussion. Ever since Harvey et al. (2016), reproducible and comprehensive construction of anomalies has increasingly gained much attention and by far the most well-cited construction includes the one initially released in Green et al. (2017) and many other separately extended construction of firm-level characteristic data including (Freybergerk et al., 2019; Gu et al., 2019; Demiguel et al., 2020; Kozak et al., 2020) and recently by far the most comprehensive one released along with Chen and Zimmermann (2020a,b). For the setting in which our discussion relies, we will use $x_{i,\text{Size}}$ to denote realized value of Size for stock i while $x_{i,\text{-Size}}$ refers to the realized values of remained characteristics. Furthermore, we replace response variable Y_i with the adjusted return associated with firm i at month t ($R_{i,t}$), collected from Center for Research in Security Prices (CPSP) database and merge this asset return data with the characteristics data constructed in Chen and Zimmermann (2020a) [8] at monthly frequency with one month lag. [9] To circumvent the computational burden, we apply the kernel based algorithms on each cross-section and take time-series average as the finally fitted curve to demonstrate the Size (market value) effect on individual stock return. The final result is demonstrated as following,

[8] We appreciate the data construction work along with the kindly shared data by Andrew Y. Chen and Tom Zimmerman. They also kindly establish a website at `https://sites.google.com/site/chenandrewy/open-source-ap?authuser=0` for detailed description of this work.

[9] Given this panel data structure, rigorously speaking we need to introduce additional subscript to emphasize the time-series dependency such that $X_{i,t}$ refers to data in feature space (characteristics) and $R_{i,t}$ refers to response variable (adjusted return of stocks at monthly frequency). But temporarily for the sake of simplicity we just omit the subscript and moreover as theoretically it is possible for us to stack all the cross-sectional data for comprehensive analysis but surely will this increase computational burden.

[Place Figure 4 about here]

As discussed in Fama and French (2008), microcaps (commonly referred to stocks associated relative small market values, specifically those stocks with market equity below the 20th percentile of NYSE stocks) represent only 3% of the total market capitalization of the NYSE-Amex-NASDAQ universe, but account for 60% of the number of stocks. This empirically documented result implies that the result graphically demonstrated in Figure 4 should be interpreted mainly from the left-hand-side of the kink point, which suggests the negative relationship between cross-sectional expected return and size. This implication is actually in line with the broadly documented empirical results in classical finance literature and can be further supported well via checking the cumulative return associated portfolio constructed from sorting on size (market value). [10]

[Place Figure 5 about here]

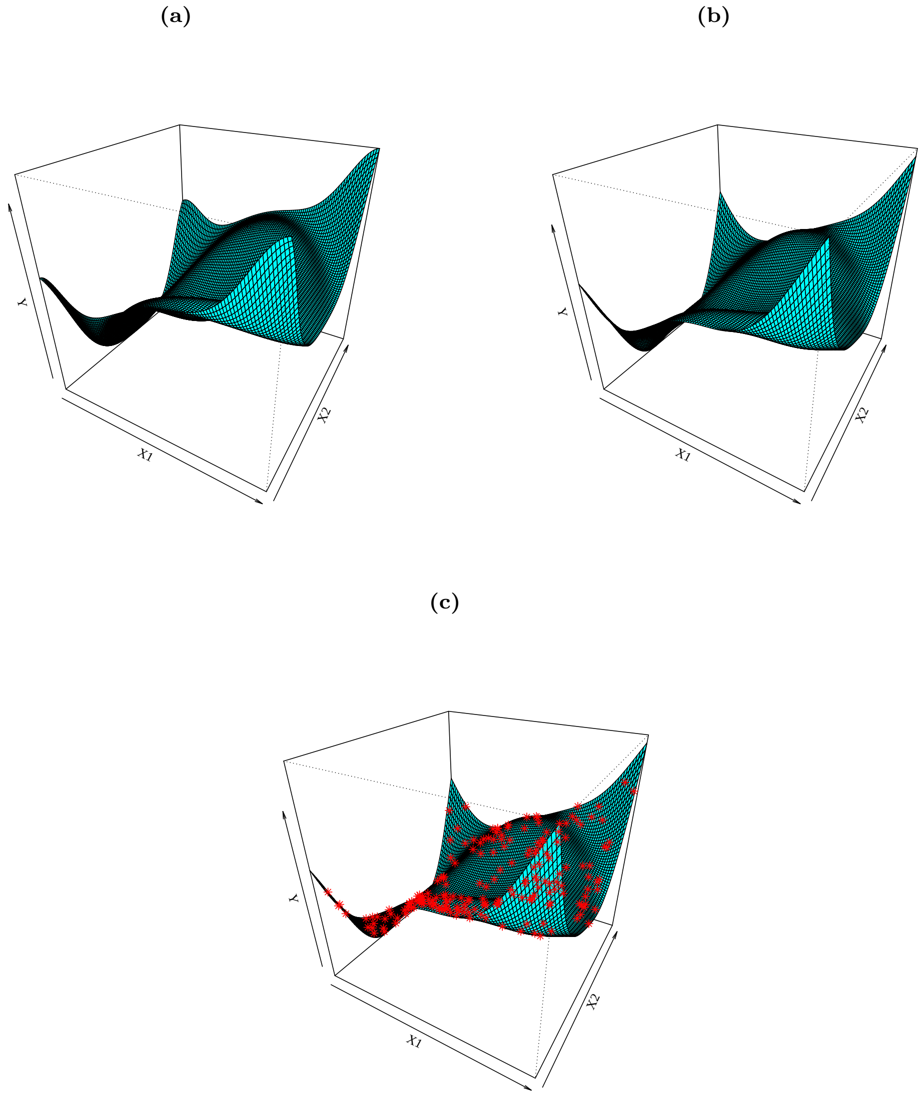
4 Conclusion

This paper as one remark note reviews and compares how kernels (kernel functions) are applied in nonparametric econometrics. One illuminating discussion done in this paper corresponds to the suggested way to interpret the mechanism through which kernels apply in nonparametric regression framework (which is arguably the backbone for most of the discussions corresponding to nonparametric econometrics) from machine-learning perspective and compare it with that interpreted from the perspective of conventional nonparametric econometrics (specifically the way to interpret kernels as tools for density function estimation). Although Mercer's theorem as one part of this note is a well-established result in both Mathematics and machine-learning literature, from my personal perspective, it is still worthwhile bridging the associated implications from this theorem along with the suggested framework for analysis with framework commonly adopted in conventional econometrics analysis as it would not only strengthen our understanding of kernels (kernel functions) but also shed light on thoughts about how conventional econometrics and modern machine-learning methodologies are inherently connected.

[10] Data construction and how this long-short portfolio is constructed are left in the appendix for more detailed discussions.

Figures and Tables

Figure 1


(a)

(b)

(c)

Note: In the above figure, we visually demonstrate the Monte Carlo examples discussed in the subsection 3.4 of main context. Specifically, (a) demonstrates the surface generated from $m\left(X_i\right)$ as specified in (33); (b) demonstrates the surface generated from fitting from SVMs; (c) demonstrates jointly the fitting from SVMs and observed data (red asterisk points). Obviously the kernel-based machine-learning algorithm SVMs, which as discussed in the main context serve as an alternative way interpreting the mechanism through which kernels (kernel functions) work in nonparametric regression setting, does a good job unravelling the non-linear data-generating mechanism but does not suffer from the over-fitting issue that much.

Figure 2

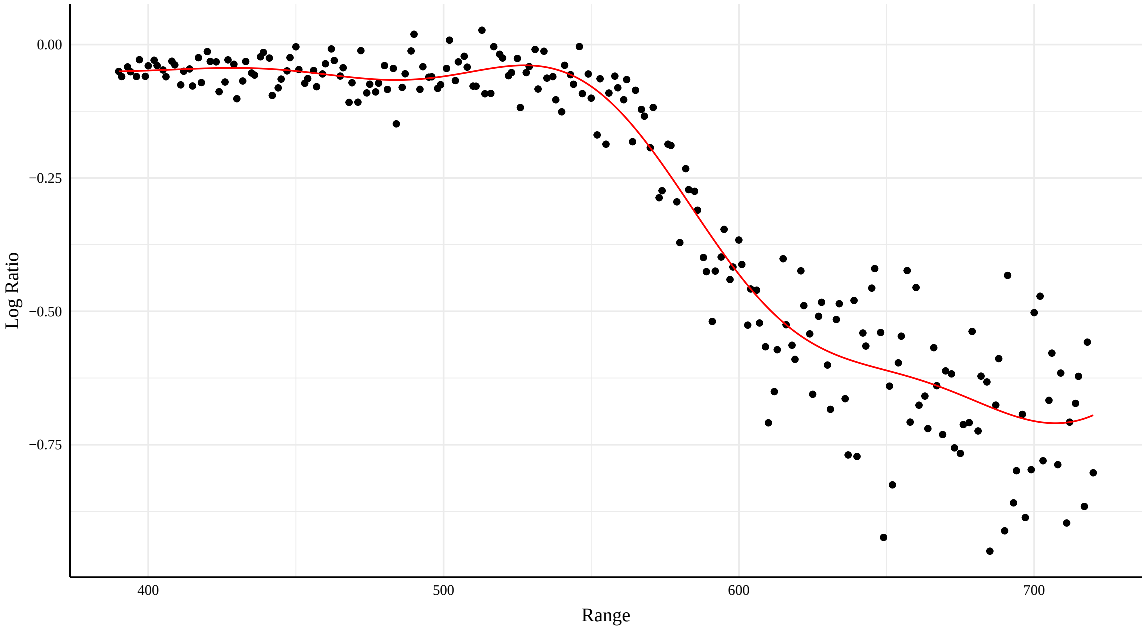
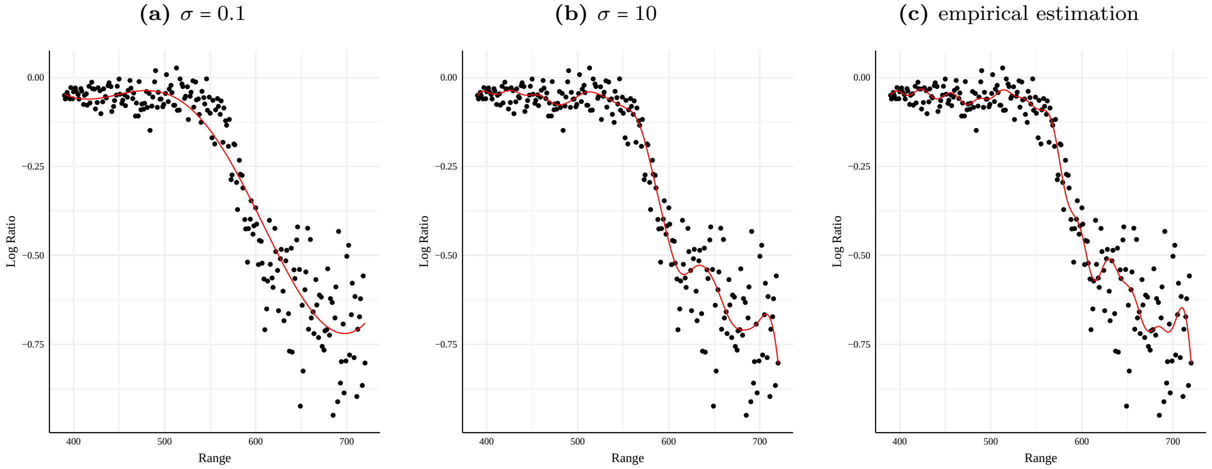


Figure 3

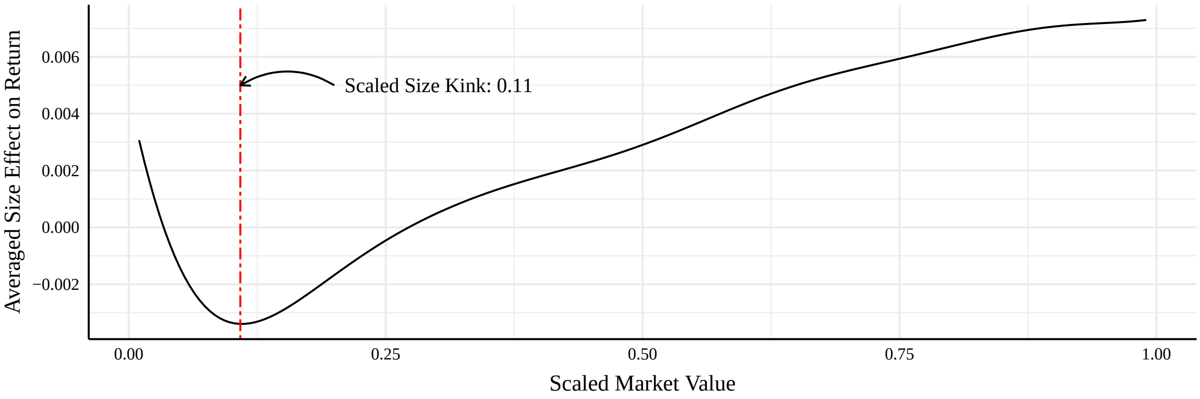

(a) σ = 0.1

(b) σ = 10

(c) empirical estimation

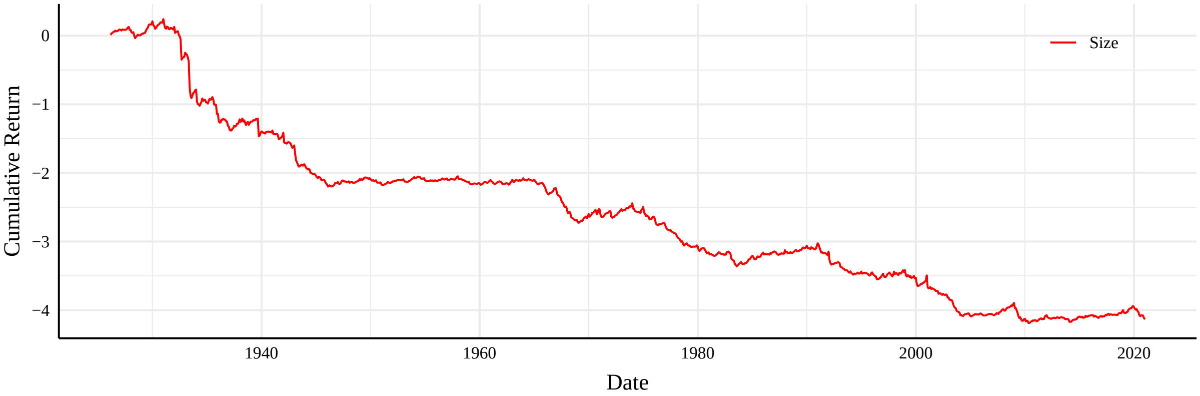
Note: In the above figure, we visually demonstrate the fitting effect associated with different σ configuration for kernels (kernel functions) applied. Specifically from the left panel to the right panel, the σ is configured as σ = 0.1, 10 and empirically estimated using the average quantiles of observed data, which is estimated as $\hat{\sigma}$ = 14.73. The main point to be emphasized here is that as σ increases, weights are assigned more on the neighbouring area of each data point in feature space and this is basically the reason why visually the in-sample goodness of fit increases as σ increase. However, as well discussed in standard statistical learning literature, the goodness of fit does not necessarily imply out-of-sample prediction accuracy and configuration of σ varies across different practical applications.

Figure 4. Size Effect on Expected Return



Note: In the above figure, we visually demonstrate the average (over time-series dimension) fitted effects of size (measured by market value and and scaled to $[0, 1]$ interval) on expected return. That is we want to approximately pin down the suggested functional form of return against scaled size using the time-series average of each cross-section.

Figure 5



Note: In the above figure, we visually demonstrate the cumulative return of long-short portfolio constructed from sorting on size (market value) in the way as we discussed in the main context.

References

Chang, C.-C. and C.-J. Lin (2011): "LIBSVM: A Library for Support Vector Machines," *ACM Trans. Intell. Syst. Technol.*, 2. [Cited on page 12.]

Chen, A. Y. and T. Zimmermann (2020a): "Open Source Cross-Sectional Asset Pricing," Working paper, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3604626. [Cited on pages 13 and A-2.]

——— (2020b): "Publication Bias and the Cross-Section of Stock Returns," *The Review of Asset Pricing Studies*, 10, 249–289. [Cited on page 13.]

Demiguel, V., A. Martín, F. J. Nogales, and R. Uppal (2020): "A Transaction-Cost Perspective on the Multitude of Firm Characteristics," *The Review of Financial Studies*, 33, 2180–2122. [Cited on page 13.]

Durrett, R. (2019): *Probability: Theory and Examples*, Cambridge University Press, 415–420, Cambridge Series in Statistical and Probabilistic Mathematics, 5 ed. [Cited on pages 2 and 12.]

Fama, E. F. and K. R. French (1992): "The Cross-Section of Expected Stock Returns," *The Journal of Finance*, 47, 427–465. [Cited on pages 13 and A-2.]

——— (1993): "Common Risk Factors in the Returns on Stocks and Bonds," *Journal of Financial Economics*, 33, 3–56. [Cited on pages 13 and A-2.]

——— (1996): "Multifactor Explanations of Asset Pricing Anomalies," *The Journal of Finance*, 51, 55–84. [Cited on page 13.]

——— (2008): "Dissecting Anomalies," *The Journal of Finance*, 63, 1653–1678. [Cited on page 14.]

Fix, E. and J. L. Hodges (1989): "Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties," *International Statistical Review / Revue Internationale de Statistique*, 57, 238–247. [Cited on page 1.]

Freybergerk, J., A. Neuhierl, and M. Weber (2019): "Dissecting Characteristics Nonparametrically," Working paper, forthcoming in *The Review of Financial Studies.* [Cited on pages 13 and A-2.]

Green, J., J. R. M. Hand, and X. F. Zhang (2017): "The Characteristics that Provide Independent Information about Average U.S. Monthly Stock Returns," *The Review of Financial Studies*, 30, 4389–4436. [Cited on page 13.]

Gu, S., B. Kelly, and D. Xiu (2019): "Empirical Asset Pricing via Maching Learning," Working paper, forthcoming in the *The Review of Financial Studies.* [Cited on page 13.]

Harvey, C. R., Y. Liu, and H. Zhu (2016): "…and the Cross-Section of Expected Returns," *The Review of Financial Studies*, 29, 5–68. [Cited on page 13.]

Hastie, T., R. Tibshirani, and J. Friedman (2001): *The Elements of Statistical Learning*, Springer Series in Statistics, New York, NY, USA: Springer New York Inc. [Cited on page 4.]

Holst, U., O. Hössjer, C. Bjöklund, P. Ragnarson, and H. Edner (1996): "Locally Weighted Least Squares Kernel Regression ans Statistical Evaluation," *Environmetrics*, 7, 401–416. [Cited on page 12.]

Karatzoglou, A., A. Smola, K. Hornik, and A. Zeileis (2004): "kernlab - An S4 Package for Kernel Methods in R," *Journal of Statistical Software, Articles*, 11, 1–20. [Cited on page 12.]

König, H. (1986): *Eigenvalue Distribution of Compact Operators*, Springer Basel AG. [Cited on page 3.]

Kozak, S., S. Nagel, and S. Santosh (2020): "Shrinking the Cross Section," *Journal of Financial Economics*, 135, 271–292. [Cited on page 13.]

Li, Q. and J. Racine (2006): *Nonparametric Econometrics: Theory and Practice*, Princeton University Press, 1 ed. [Cited on pages 1 and A-1.]

Lin, H.-T., C.-J. Lin, and R. C. Wang (2007): "A note on Platt's probabilistic outputs for support vector machines," *Machine Learning*, 68. [Cited on page 5.]

Mercer, J. (1909): "Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 209, 415–446. [Cited on page 3.]

Nadaraya, E. A. (1964): "On Estimating Regression," *Theory of Probability & Its Applications*, 9, 141–142. [Cited on pages 2 and A-1.]

Nard, G. D. and Z. Zhao (2020): "A Large-Dimensional Test for Cross-Sectional Anomalies: Efficient Sorting Revisited," Working paper, New York University. [Cited on page 13.]

Pagan, A. and A. Ullah (1999): *Nonparametric Econometrics*, Cambridge University Press. [Cited on pages 1 and A-1.]

Parzen, E. (1962): "On Estimation of a Probability Density Function and Mode," *The Annals of Mathematical Statistics*, 33, 1065–1076. [Cited on page 1.]

Platt, J. C. (1999): "Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods," in *Advances in Large Margin Classifiers*, Cambridge, MA, MIT Press, 61–74. [Cited on page 5.]

R Core Team (2020): *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing. URL: https://www.r-project.org/. [Cited on page 12.]

Racine, J. S. (2019): *An Introduction to the Advanced Theory and Practice of Nonparametric Econometrics: A Replicable Approach Using R*, Cambridge University Press. [Cited on page 1.]

Rockafellar, R. T. (1970): *Convex Analysis*, Princeton University Press. [Cited on page 8.]

Rosenblatt, M. (1956): "Remarks on Some Nonparametric Estimates of a Density Function," *The Annals of Mathematical Statistics*, 27, 832–837. [Cited on page 1.]

Vapnik, V. (1998): *Statistical Learning Theory*, Wiley, New York. [Cited on page 4.]

Watson, G. S. (1964): "Smooth Regression Analysis," *Sankhyā: The Indian Journal of Statistics, Series A (1961-2002)*, 26, 359–372. [Cited on pages 2 and A-1.]

Appendix

A Auxiliary Proofs

A.1 Nonparametric kernel functional estimator

Proof. As suggested in Nadaraya (1964) and Watson (1964) and some standard nonparametric textbooks (Pagan and Ullah, 1999; Li and Racine, 2006), kernels (kernel functions) chosen within this context are usually associated with bandwidth determining smoothness. For the multivariate case to be discussed, let $(h_y, \mathbf{h}) = (h_y, h_1, \dots, h_p)$ and $\mathbf{h} = (h_1, \dots, h_p)$ denote the vectors collecting chosen bandwidths for $\widehat{f}(y, \mathbf{x})$ and $\widehat{f}(\mathbf{x})$ respectively. Thus as we have discussed in the main context, $\widehat{f}(y, \mathbf{x})$ and $\widehat{f}(\mathbf{x})$ take the following forms respectively

$$\begin{aligned}
\widehat{f}(y, \mathbf{x}) &= \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - x\right) k_{h_y}\left(Y_i - y\right) \\
&= \frac{1}{nh_y h_1 \dots h_q} \sum_{i=1}^{n} K\left(\frac{X_{i1} - x_1}{h_1}, \dots, \frac{X_{iq} - x_q}{h_q}\right) k\left(\frac{Y_i - y}{h_y}\right)
\end{aligned}$$

and

$$\widehat{f}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - x\right) = \frac{1}{nh_1 \dots h_q} \sum_{i=1}^{n} K\left(\frac{X_{i1} - x_1}{h_1}, \dots, \frac{X_{iq} - x_q}{h_q}\right)$$

To obtain the functional form of (5), it suffices to derive the numerator of (5), which is demonstrated as following

$$\begin{aligned}
\int f(y, \mathbf{x}) dy &= \frac{1}{nh_y} \sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - \mathbf{x}\right) \int yk\left(\frac{Y_i - y}{h_y}\right) dy \\
&= \frac{1}{nh_y} \sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - \mathbf{x}\right) \int yk\left(\frac{y - Y_i}{h_y}\right) dy \\
&= \frac{1}{n} \sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - \mathbf{x}\right) \int \left(Y_i + h_y u\right) k(u) du \\
&= \frac{1}{n} \sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - \mathbf{x}\right) Y_i
\end{aligned}$$

where from the first equation to the second equation we use symmetric property of kernels (kernel) functions and from the second to third equation we just the apply the standard change of variables trick and the fourth equation stems from the corresponding properties of kernels such that $\int k(u) = 1$

and $\int uk(u)du = 0$. With numerator of this form derived, it is straightforward to have

$$\widehat{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} Y_i K_{\mathbf{h}}\left(X_i - \mathbf{x}\right)}{\sum_{i=1}^{n} K_{\mathbf{h}}\left(X_i - \mathbf{x}\right)}$$

as demonstrated in the main context. □

B Data and Long-short Portfolio Construction

The firm-level characteristic-data constructed in Chen and Zimmermann (2020a, henceforth CZ2020a) does provide by far the most comprehensive universe of firm-level characteristics of U.S. stock market. However as it is relatively more standard to construct firm size (measured by market value) from database like `COMPUTAT` and `CRSP`, CZ2020a does not publish the corresponding data on the website, Open Source Asset Pricing . Consequently we follow the standard procedure implemented in Fama and French (1992, 1993) to construct firms' market values via `me` = `prc` × `shrout`, where `prc` refers to the acronym of stock price or Bid/Ask average while `shrout` refers to the shares outstanding following the standard labelling scheme in `CRSP`. One point to be emphasized here is for the case when different `permnos` are identified by the same `CRSP` permanent company number `permco`, we need to aggregate market values attached to these `permnos` as the corresponding market values shared by all these assets identified by these `permnos`. Once the size data (measured by market value) is constructed, it is normalized to lie in between 0 and 1 as in Freybergerk et al. (2019) such that

$$rc_{i,t}^{\text{size}} = \frac{\text{rank}\left(c_{i,t}^{\text{size}}\right)}{n_t + 1} \tag{B.1}$$

where $c_{\text{size},t}^{i}$ refers to the originally unscaled size values associated with stock i and n_t refers to the total number of firms available for observations at time t. Then the corresponding portfolio weights used to construct long-short portfolios are established as following

$$z_{i,t}^{\text{size}} = \frac{\left(rc_{i,t}^{\text{size}} - \overline{rc}_t^{\text{size}}\right)}{\sum_{i=1}^{n_t} \left|rc_{i,t}^{\text{size}} - \overline{rc}_t^{\text{size}}\right|} \tag{B.2}$$

where

$$\overline{rc}_t^{\text{size}} = \frac{1}{n_t}\sum_{i=1}^{n_t} rc_{i,t}^{\text{size}}.$$