Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 451947, 9 pages
http://dx.doi.org/10.1155/2015/451947

Research Article

An Efficient Kernel Learning Algorithm for Semisupervised Regression Problems

Chao Zhang and Shaogao Lv

Statistics School, Southwestern University of Finance and Economics, Chengdu 611130, China

Correspondence should be addressed to Shaogao Lv; lvsg716@swufe.edu.cn

Received 4 July 2015; Accepted 25 August 2015

Academic Editor: Igor Andrianov

Kernel selection is a central issue in kernel methods of machine learning. In this paper, we investigate the regularized learning schemes based on kernel design methods. Our ideal kernel is derived from a simple iterative procedure using large scale unlabeled data in a semisupervised framework. Compared with most of existing approaches, our algorithm avoids multioptimization in the process of learning kernels and its computation is as efficient as the standard single kernel-based algorithms. Moreover, large amounts of information associated with input space can be exploited, and thus generalization ability is improved accordingly. We provide some theoretical support for the least square cases in our settings; also these advantages are shown by a simulation experiment and a real data analysis.

1. Introduction

Kernel-based methods have been proved to be powerful for a wide range of different data analysis problems. Since support vector machine (SVM) was initially proposed by Vapnik [1], many other kernel-based methods have been proposed such as kernel PCA, kernel Fisher discriminant, and kernel CCA. In many cases, the performance of kernel methods greatly depends on the choice of kernel function (for the importance of specifying an appropriate kernel, see Chapter 13 of [2]). To choose an appropriate kernel, many kernel learning algorithms have been proposed in recent years such as [3–5]. Among them, there are two kinds of candidate kernel sets: the first one involves parameter selection from the candidate kernel collection including Gaussian kernels [6]; that is, $G_\sigma(x, y) = \exp(-\sigma\|x - y\|^2)$. The others mainly refer to the linear combination of certain prespecified kernels. It is known that the latter one is also called "multiple kernel learning" [4]. Recall that Lanckriet et al. [5] proposed a positive semidefinite programming to search the best linear combination automatically for SVM; however, this approach is time-consuming and only feasible for small sample cases. Sonnenburg et al. [7] relaxed this mentioned optimization problem to a semi-infinite linear program, which is capable of coping with a large spectrum of kernels and samples. However, these multiple kernel learning algorithms do not have better performance than traditional nonweighted kernel $K = \sum_k K_k$ in SVM sometimes, and Cortes [8] questioned that "can learning kernels help performance?". Recently Kloft and Blanchard [9] introduced a multikernel learning with l^q-norm ($q \geq 1$) approach, which has been shown effective in both theory and practice [10, 11]. Essentially, l^q-norm is a kind of minimizing empirical risk algorithm with kernel candidate set $\{K = \sum_{k=1}^{M} \theta_k K_k \mid \|\theta\|_{l^q} \leq 1, \theta \geq 0\}$. Kloft and Blanchard [9] provided an excess generation error utilizing local Rademacher complexity of l^q-norm multiple kernel learning. Although these mentioned learning kernel algorithms provide more flexibility than these one-kernel approaches, more complex computational problems induced by multiple kernel learning arise additionally. In addition, these above kernel learning algorithms are considered only under fully supervised learning settings. In practice, labeled instances however are often difficult, expensive, or time-consuming to obtain, as they require the efforts of experienced human annotators. Meanwhile unlabeled data may be relatively east to collect. In the machine learning literature, semisupervised learning addresses this problem by using large amount of

unlabeled data, together with the labeled data, to build better learners.

In this paper, we pursue the goal of kernel learning algorithms under semisupervised learning framework. For this sake, we resort to find a sequence of candidate kernels using an iterative procedure; then our regularized learning algorithms perform on the corresponding RKHS, which leads to a classical convex optimization program on training data. Finally, we apply the test data to select the optimal kernel function and regularization parameter. It is worth noting that the proposed method consists of two-step estimation stated as above. At the first step, we use large amount of information of unlabeled data to explore underlying data structure. Our optimization problem involved at the second step is as efficient as those classical single kernel approaches. More importantly, we provide sufficient theoretical support for our approach and we demonstrate the effectiveness of the proposed method by experiments.

The rest of the paper is organized as follows. In Section 2 we introduce some basic notations and our two-step estimation for kernel learning. We present main theoretical results for the proposed approach, which is achieved mainly by using advanced concentration inequalities stated in Section 3. Section 4 contains some proof details such as error decomposition and approximation error. We implement a simulation and a real data experiment in Section 5. Some proofs are relegated to the Appendix.

2. The Proposed Algorithm

We first describe the notations used in this paper. Suppose that our algorithm produces one learner $f : X \to Y$ from a compact metric space X to the output space $Y \subseteq \mathbb{R}$. Such a learner f yields for each point x the value $f(x) \in Y$, which is a prediction made for x. The goodness of estimation is usually assessed by some specified loss function denoted by $\nu : \mathbb{R}^2 \to \mathbb{R}^+$. The most commonly used loss function is the least square one; that is, $\nu(f(x), y) = (f(x) - y)^2$. Let (x, y) be the random variable on $X \times Y$ with the probability distribution ρ. Within statistical learning framework, the target function can be formulated as a minimizer of the following functional optimization:

$$f^* = \arg\min_{f \in \mathscr{F}} \left\{ \int_{X \times Y} \nu\left(f\left(x\right), y\right) d\rho\left(x \times y\right) \right\}. \tag{1}$$

In particular, in the case of the least square loss, we derive an explicit solution expressed as

$$f_\rho\left(x\right) = \int_Y y \, d\rho\left(y \mid x\right), \quad x \in X, \tag{2}$$

where $\rho(y \mid x)$ is conditional probability measure at x induced by ρ. Under fully supervised learning setting, based on available samples $S = \{(x_i, y_i)\}_{i=1}^m$, the main goal of learning is to design an efficient learning algorithm to obtain one learner f_S, which is capable of well approximating regression function f^* on the whole space. These popular regularized learning algorithms within a RKHS can be stated as

$$\inf_{f \in \mathscr{H}_K} \left\{ \frac{1}{m} \sum_{i=1}^m \nu\left(f\left(x_i\right), y_i\right) + \lambda \left\| f \right\|_K^2 \right\}, \tag{3}$$

where $\mathscr{H}_K$ is a specified RKHS and $0 < \lambda \leq 1$ is the regularization parameter, controlling empirical error and functional complexity of $\mathscr{H}_K$. Note that λ may rely on sample size; it satisfies $\lim_{m \to \infty} \lambda(m) = 0$.

In the semisupervised learning framework, the first m samples are labeled as above and followed by n unlabeled samples $\widetilde{\mathbf{x}} = \{x_{m+1}, \ldots, x_{m+n}\}$. Denote by a weak kernel K^0 as the original kernel. Compared to those standard kernels, a weak kernel here means that its complexity is very large or it is less smooth. A learner with a weak kernel usually leads to overfitting, while it can approximate well more complicated functions and hence reduce estimation bias of the learner. Hence, selecting an appropriate kernel needs to trade off the functional complexity of various $\mathscr{H}_K$. Motivated by this observation, we propose an iterative procedure as our first step for constructing candidate kernels. For the kth step, the next candidate kernel is derived as follows:

$$K^{k+1}\left(x, u\right) = \frac{1}{m+n} \sum_{i=1}^{m+n} K^k\left(x, x_i\right) K^k\left(u, x_i\right), \quad x, u \in X. \tag{4}$$

The labeled samples are divided into training data denoted by $D = \{(x_i, y_i)\}_{i=1}^l$ and test data $T = S \setminus D$, respectively; then we establish our regularized learning algorithm based on the associated $\mathscr{H}_{K^k}$:

$$f_{\mathbf{z},\lambda}^k := \arg\min_{f \in \mathscr{H}_{K^k}} \left\{ \frac{1}{l} \sum_{i=1}^l \nu\left(\widehat{f}\left(x_i\right), y_i\right) + \lambda \left\| f \right\|_{K^k}^2 \right\}. \tag{5}$$

Given the total number N of iteration steps, we minimize $\widehat{f}(k, \lambda)$ on the test data:

$$f_{\mathbf{z},\lambda^*}^{k^*} = \min_{k \in \{1,2,\ldots,N\}, \lambda \in (0,1)} \left\{ \frac{1}{m-l} \sum_{(x_i, y_i) \in T} \left(f_{\mathbf{z},\lambda}^k\left(x_i\right) - y_i \right)^2 \right\}. \tag{6}$$

Thus, we take $f_{\mathbf{z},\lambda^*}^{k^*}$ as our final leaner in semisupervised learning setting. Note that we use the least square loss instead of ν in the final step, since we compute or approximate its solution easily following nice mathematical property of the least square one.

Our motivation of designing kernel as (4) is based on the following fact. By the Mercer Theorem [12], any given kernel K^k defined on a compact set can be expressed as $K^k(x, u) = \sum_{j=1}^\infty \lambda_j e_j(x) e_j(u)$, where (λ_j, e_j) is the corresponding eigenpairs of the integral operator L_{K^k}, which will be defined in (19) below. In general, the problem of selecting kernel corresponds to a suitable choice of the parameters λ_j, since the eigenvalue

λ_j has a close relationship with the functional complexity of $\mathscr{H}_{K^k}$ [13]. In our case, we use an iterative procedure to select an appropriate kernel. To be precise, we define a new candidate kernel by $K^{k+1}(x,u) := \int_X K^k(x,t)K^k(u,t)d\rho_X(t)$. Based on the observation $K^{k+1}(x,u) = \sum_{j=1}^{\infty}\lambda_j^2 e_j(x)e_j(u)$, it suffices to find an appropriate iteration step. Since ρ_X is often unknown, alternatively, we use an empirical estimator of K^{k+1} defined as (4) as our candidate kernel. Furthermore, in view of the slow rate with order $1/\sqrt{m}$, by which the empirical kernel in (4) converges to its corresponding population kernel, the large amounts of unlabeled data guarantee a smaller error generated by sampling randomly.

It is seen from the proposed program as above that this method avoids multioptimization in the process of learning kernels and its computation is as efficient as the standard single kernel-based algorithms up to some constant. Moreover, large amounts of information associated with input space have been made fully use of, so that some intrinsic data structure may be exploited.

3. Main Results

To highlight our idea by presenting more refined theoretical results, in what follows, we are primarily concerned with the least square setting, since the regularized least square algorithm has a closed-form solution. First of all, by the law of large number, with a high probability, we can replace the first step (4) with the following iterative procedure:

$$K^{k+1}(x,u) = \int_X K^k(x,t)\,K^k(u,t)\,d\rho_X(t), \quad x,u \in X, \quad (7)$$

where ρ_X is the marginal distribution induced by ρ. We denote by $f_{\mathbf{z},\lambda}^N$ the derived learner of (5) at the Nth iteration. For notational simplicity, we write $f_{\mathbf{z}} = f_{\mathbf{z},\lambda}^N$. In this paper, we focus on generalization error of the proposed algorithm; that is,

$$\left\| f_{\mathbf{z}} - f_\rho \right\|_{L^2_{\rho_X}}. \quad (8)$$

A small quantity of $\|f_{\mathbf{z}} - f_\rho\|_{L^2_{\rho_X}}$ implies a good prediction ability of $f_{\mathbf{z}}$. Different from those classical literatures under fixed kernel settings such as [13, 14], the main goal of this paper is to indicate some specific advantages theoretically compared with those fixed kernel approaches.

To simplify theoretical analysis, we assume that the conditional distribution $\rho(\cdot \mid x)$ has a support on $[-M, M]$, and it follows that $|f_\rho(x)| \le M$ almost everywhere. To this end, we introduce the projection operator as follows.

Definition 1. Define the projection operator π_M on measurable function $f : X \to \mathbb{R}$ as

$$\pi_M(f)(x) = \begin{cases} M, & \text{if } f(x) > M, \\ f(x), & \text{if } -M \le f(x) \le M, \\ -M, & \text{if } f(x) < -M. \end{cases} \quad (9)$$

Note that the error bound between the projections of $f_{\mathbf{z}}$ and f_ρ can be expressed as

$$\left\| \pi_M(f_{\mathbf{z}}) - f_\rho \right\|^2_{L^2_{\rho_X}} = \mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}(f_\rho), \quad (10)$$

where $\mathscr{E}(f) = \int_Z (f(x) - y)^2 d\rho$ denotes the population error of the function f.

Since the regression function f_ρ may not be found within $\mathscr{H}_K$, the approximation error between $\mathscr{H}_K$ and f_ρ is needed. Considering the error induced by sampling and the approximation error, we introduce the following empirical error as

$$\mathscr{E}_{\mathbf{z}}(f) = \frac{1}{m}\sum_{i=1}^{m}(f(x_i) - y_i)^2, \quad (11)$$

and we also introduce an approximation error associated with the joint distribution ρ:

$$\mathscr{D}(\lambda) = \left\| f_\lambda - f_\rho \right\|^2_{L^2_{\rho_X}} + \lambda \left\| f_\lambda \right\|^2_{K^N}, \quad (12)$$

where f_λ is called the regularization function, given as

$$f_\lambda = \arg\min_{f \in \mathscr{H}_{K^N}} \left\{ \left\| f - f_\rho \right\|^2_{L^2_{\rho_X}} + \lambda \left\| f \right\|^2_{K^N} \right\}. \quad (13)$$

Remark 2. In the literature of learning theory, one usually assumes that there exist $c_\beta > 0$ and $0 < \beta \le 1$, such that $\mathscr{D}(\lambda) \le c_\beta \lambda^\beta$. In fact $\beta = 1$ means $f_\rho \in \mathscr{H}_{K^N}$ and vice versa [15–17]. Strictly speaking $\mathscr{D}(\lambda)$ is formally discussed in approximation theory.

To obtain convergence rates of (10), we decompose the term $\mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}(f_\rho)$ into two parts: the approximation error and the sample error; see [14, 15].

Proposition 3. *Let $f_{\mathbf{z}}$ be defined by (5); the following inequality holds*

$$\mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}(f_\rho) \le \mathscr{S}(\mathbf{z},\lambda) + \mathscr{D}(\lambda), \quad (14)$$

where

$$\begin{aligned} \mathscr{S}(\mathbf{z},\lambda) = {} & \left(\mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}_{\mathbf{z}}(\pi_M(f_{\mathbf{z}}))\right) \\ & + \left(\mathscr{E}_{\mathbf{z}}(f_\lambda) - \mathscr{E}(f_\lambda)\right). \end{aligned} \quad (15)$$

Proposition 3 shows that $\mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}(f_\rho)$ is bounded by $\mathscr{S}(\mathbf{z},\lambda) + \mathscr{D}(\lambda)$. We usually call $\mathscr{S}(\mathbf{z},\lambda)$ the sample error, since this quantity mainly involves random sampling and the complexity of $\mathscr{H}_K$.

Bounding the sample error $\mathscr{S}(\mathbf{z})$ is a standard technique in learning theory [13, 15, 18]. To this end, we need to introduce the notion of covering number to measure the complexity of $\mathscr{H}_K$.

Definition 4. Assume $(\mathscr{M}, d)$ is a pseudometric space with some metrics d and $S \subset \mathscr{M}$. Then, for any $\epsilon > 0$, we define

the covering number $\mathcal{N}(S,\epsilon,d)$ referring to ϵ and d as covering number of a ball with radius ϵ:

$$\mathcal{N}(S,\epsilon,d) = \min\left\{\ell \in \mathbb{N} : S \subset \bigcup_{j=1}^{\ell} B\left(s_j,\epsilon\right) \text{ for any sequence } \left\{s_j\right\}_{j=1}^{\ell} \subset \mathcal{M}\right\}. \tag{16}$$

Recall that a kernel function is called the Mercer kernel, if it is symmetric, positive definite, and continuous. Several properties concerning the Mercer kernel have been established well and can be found in [12, 15]. Suppose that $\kappa := \sup_{x\in X}\sqrt{K^0(x,x)} < \infty$.

Assumption 5. Suppose that the Mercer kernel K^N has a polynomial complexity with $s > 0$

$$\log \mathcal{N}\left(B_1,\eta\right) \le C_N\left(\frac{1}{\eta}\right)^s, \quad \forall \eta > 0, \tag{17}$$

where B_1 is the unit ball of $\mathcal{H}_K$ and C_N is some constant. For the Sobolev space H^h on $\mathbb{R}^p$ with the order h, it is known in [12] that $s = 2p/h$.

On the other hand, to quantify the approximation error $\mathcal{D}(\lambda)$ and characterize the regularity of f_ρ, we need to introduce the notion of fractional integral operator associated with K. Recall that a standard inner product on $L^2_{\rho_X}$ is defined as

$$\left\langle f,g\right\rangle_{\rho_X} = \int_X f(x)\, g(x)\, d\rho_X. \tag{18}$$

Then we can define an integral operator L_K on $L^2_{\rho_X}(X)$:

$$L_K(f)(x) = \int_X K(x,t)\, f(t)\, d\rho_X(t). \tag{19}$$

It has been verified in [16] that the integral operator L_K is a compact, self-adjoint, and positive definite operator from $L^2_{\rho_X}(X)$ to $L^2_{\rho_X}(X)$. So the fractional operator of L_K is well defined. Moreover, it is easy to check that $L_{K^{k+1}} = L^2_{K^k}$ and $L_{K^N} = L^{(2N)}_{K^0}$. Lemma 12 below will show that if a weak kernel with respect to the true function is used for learning, the approximation ability cannot be improved even if the true function is sufficiently smooth. This is why we propose the iterative procedure (4) for updating kernels.

With these preparations, we can state the main results depending on the capacity of $\mathcal{H}_{K^N}$ and the smoothness of target function as follows.

Theorem 6. *Let $f_{\mathbf{z}}$ be defined by (5) and Assumption 5 holds. If $L^{-r}_{K^0}(f_\rho) \in L^2_{\rho_X}(X)$ $(r > 0)$, then, for any $0 < \delta < 1$ and $m > (1360\log(1/\delta)/\widetilde{c})^{(1+s)/s}$, the following holds:*

$$\begin{aligned}\mathcal{S}(\mathbf{z},\lambda) &\le \frac{7\left(\kappa^N\lambda^{\min\{(r-N)/2N,0\}}\left\|L^{-r}_{K^0}f_\rho\right\|_{L^2_{\rho_X}} + 3M\right)^2\log(2/\delta)}{3m} \\ &\quad + \frac{1}{2}\lambda^{\min\{r/N,2\}}\left\|L^{-r}_{K^0}f_\rho\right\|^2_{L^2_{\rho_X}} + \frac{1}{2}\mathcal{E}\left(\pi_M\left(f_{\mathbf{z}}\right)\right) \\ &\quad - \mathcal{E}\left(f_\rho\right) + \widetilde{c}M^{s/(1+s)}\left(\frac{1}{\lambda}\right)^{s/2(1+s)}\left(\frac{1}{m}\right)^{1/(1+s)},\end{aligned} \tag{20}$$

with probability of at least $1-\delta$, where the constant $\widetilde{c}$ is given by Proposition 11.

From Proposition 3, we can deduce that $(1/2)\mathcal{E}(\pi_M(f_{\mathbf{z}})) - \mathcal{E}(f_\rho)$ in $\mathcal{S}(\mathbf{z},\lambda)$ can be ignored by studying the new equivalent sampling error given by $\widetilde{\mathcal{S}}(\mathbf{z},\lambda) = \mathcal{S}(\mathbf{z},\lambda) - (1/2)\mathcal{E}(\pi_M(f_{\mathbf{z}})) - \mathcal{E}(f_\rho)$. The following corollary provides an asymptotically optimal convergence rate of $f_{\mathbf{z}}$. The proof can be found in the Appendix.

Corollary 7. *Let $f_{\mathbf{z}}$ be defined by (5), and Assumption 5 holds true. If $L^{-r}_{K^0}f_\rho \in L^2_{\rho_X}$, then when $1 < N < r < 2N$, for any $0 < \delta < 1$ and $m > (1360\log(1/\delta)/\widetilde{c})^{(1+s)/s}$, the following holds:*

$$\widetilde{\mathcal{S}}(\mathbf{z},\lambda) = \log\left(\frac{2}{\delta}\right)\mathcal{O}\left(\frac{1}{m}\right)^{2r/(2r(1+s)+sN)}, \tag{21}$$

with probability of at least $1-\delta$, where $\lambda = (1/m)^{2N/(2r(1+s)+sN)}$. Particularly, for $K^N \in C^\infty(X)$, we have

$$\widetilde{\mathcal{S}}(\mathbf{z},\lambda) = \log\left(\frac{2}{\delta}\right)\mathcal{O}\left(\frac{1}{m}\right)^{1-\epsilon}, \tag{22}$$

where ϵ is an arbitrary positive number.

It is seen from Corollary 7 that the ideal choice of the regularization parameter λ depends on the two quantities r and s, which are often unknown in advance. Alternatively, cross-validation technique is one of the commonly used tools in practice. It is worth noting that our approach selects the ideal kernel and the regularization parameter simultaneously, which is of significant difference from those classical fixed kernel methods.

Next, we compare our rate with existing references. Recently, sharp learning rates have been established by advanced empirical process technique in [14]. Note that we use K^0 to replace the kernel K appearing in algorithm (3), where its covering number has polynomial decay index of p.

To be precise, an upper bound of sampling error in [14] was given as

$$\mathcal{S}(\mathbf{z},\lambda) \le 2\mathcal{D}(\lambda) + \frac{28\kappa\mathcal{D}(\lambda)}{3m\lambda}\log\left(\frac{2}{\delta}\right) + \frac{724M^2}{m}\log\left(\frac{2}{\delta}\right) + 2C_1\left(\frac{1}{m}\right)^{2/(2+p)}\left(\frac{1}{\lambda}\right)^{p/(2+p)}. \tag{23}$$

From formula (A.2) in the Appendix, we can achieve that $\mathcal{H}_{K^N} = L_{K^N}^{1/2}(L^2_{\rho_X}) = L^N_{K^0}(L^2_{\rho_X}) = L^{(N-1/2)}_{K^0}(\mathcal{H}_{K^0})$. Following the equivalent relationship between covering number and the spectrum of L_K (see Theorem 10 [13]), we see that sufficiently large N ensures that $s \ll p$. Additionally, when $1 < N < r < 2N$, $\lambda^{\min\{(r-N)/(2N),0\}}$ in Theorem 6 and $\mathcal{D}(\lambda)/\lambda$ above are both constants and hence it follows that $\lambda^{\min\{r/N,2\}} = \lambda^{r/N} < \lambda = c\mathcal{D}(\lambda)$ with some constant c, since $0 < \lambda < 1$. In summary, our derived sample error is sharper than that in [14]. Thus, if the regression function is smooth sufficiently, that is, which can also be approximated well by $\mathcal{H}_{K^N}$, we conclude that the corresponding learning rate of Theorem 6 outperforms that in [14]. This shows that if we know the smoothness (r) of the target function, we can choose a proper kernel K^N (which requires $N > r/2$) to improve the sampling error effectively. This provides us an excellent theoretical basis in choosing kernel function for real problems. Of course, considering noises with the samples in real problems, the chosen kernel also has to be smoother than the target function.

4. Error Analysis

The sampling error $\mathcal{S}(\mathbf{z},\lambda)$ is analyzed according to empirical process technique. The initial study on sampling error mainly applies the McDiarmid inequality without considering the conception of "space complexity." However, the McDiarmid inequality is not able to show the variance message of random variables. Afterwards, VC dimension was originally introduced into the literature and some other conceptions such as covering numbers formed in the Bernstein-type probability inequality significantly reduce the sampling error; see [19] for an explicit overview. To bound sample error, we split it into two parts again:

$$\begin{aligned}\mathcal{S}(\mathbf{z},\lambda) &= \left(\mathscr{E}_{\mathbf{z}}(f_\lambda) - \mathscr{E}_{\mathbf{z}}(f_\rho)\right) - \left(\mathscr{E}(f_\lambda) - \mathscr{E}(f_\rho)\right) \\ &\quad + \left\{\mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}(f_\rho) \right. \\ &\quad \left. - \left(\mathscr{E}_{\mathbf{z}}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}_{\mathbf{z}}(f_\rho)\right)\right\} := \mathcal{S}_1(\mathbf{z},\lambda) \\ &\quad + \mathcal{S}_2(\mathbf{z},\lambda).\end{aligned} \tag{24}$$

Note that $\mathcal{S}_1(\mathbf{z},\lambda)$ does not involve any functional complexity, which in turn can be estimated easily by the following one-side Bernstein probability inequality.

Lemma 8. *Define ξ as a random variable on the probability space Z, and there exists a constant M_ξ and it holds $|\xi - \mathbb{E}(\xi)| \le M_\xi$. Define the variance of ξ by σ^2; then, for any $0 < \delta < 1$, the following holds:*

$$\frac{1}{m}\sum_{i=1}^m \xi(z_i) - \mathbb{E}(\xi) \le \frac{2M_\xi \log(1/\delta)}{3m} + \sqrt{\frac{2\sigma^2\log(1/\delta)}{m}}, \tag{25}$$

with probability of at least $1-\delta$.

Proposition 9. *Define random variable $\xi_1(z) = (f_\lambda(x)-y)^2 - (f_\rho(x)-y)^2$. For any $0 < \delta < 1$ the following holds:*

$$\begin{aligned}\mathcal{S}_1(\mathbf{z},\lambda) &= \frac{1}{m}\sum_{i=1}^m \xi_1(z_i) - \mathbb{E}(\xi_1) \\ &\le \frac{7\left(\|f_\lambda\|_\infty + 3M\right)^2 \log(2/\delta)}{3m} \\ &\quad + \frac{1}{2}\|f_\lambda - f_\rho\|_\rho^2,\end{aligned} \tag{26}$$

with probability of at least $1-\delta/2$.

Proof. Notice that

$$\xi_1(z) = \left(f_\lambda(x) - f_\rho(x)\right)\left(f_\lambda(x) + f_\rho(x) - 2y\right). \tag{27}$$

Since $|f_\rho(x)| \le M$ is true almost everywhere, thus

$$|\xi_1| \le c := \left(\|f_\lambda\|_\infty + M\right)\left(\|f_\lambda\|_\infty + 3M\right). \tag{28}$$

This yields that $|\xi_2 - \mathbb{E}(\xi_2)| \le M_{\xi_2} := 2c$.

Additionally, notice that $\mathbb{E}(\xi_1)^2$ satisfies

$$\begin{aligned}&\mathbb{E}\left(\left(f_\lambda(x) - f_\rho(x)\right)^2\left(f_\lambda(x) + f_\rho(x) - 2y\right)^2\right) \\ &\quad \le \left(\|f_\lambda\|_\infty + 3M\right)^2\|f_\lambda - f_\rho\|_\rho^2,\end{aligned} \tag{29}$$

which implies that $\sigma^2(\xi_1) \le \mathbb{E}(\xi_1^2) \le c\|f_\lambda - f_\rho\|_\rho^2$.

By Lemma 8 and the basic inequality $\sqrt{ab} \le (a+b)/2$ $(a, b \ge 0)$, we obtain

$$\frac{1}{m}\sum_{i=1}^m \xi_1(z_i) - \mathbb{E}(\xi_1) \le \frac{7c\log(2/\delta)}{3m} + \frac{1}{2}\|f_\lambda - f_\rho\|_\rho^2 \tag{30}$$

with probability of at least $1-\delta/2$. □

Bounding the sampling error $\mathcal{S}_2(\mathbf{z},\lambda)$ is more involved, since the estimator $f_{\mathbf{z}}$ will vary with the random sample. To handle it, some advanced uniform concentration inequality is required [14].

Lemma 10. *Define G as a function set on Z. If there exists a constant c_ρ, then $|g-\mathbb{E}(g)| \le B$ almost everywhere and $\mathbb{E}(g^2) \le c_\rho \mathbb{E}(g)$. Then for any positive number ε and $0 < \alpha \le 1$,*

$$\begin{aligned}&\mathrm{Prob}_{\mathbf{z}\in Z^m}\left\{\sup_{g\in G}\frac{\mathbb{E}(g)-(1/m)\sum_{i=1}^m g(z_i)}{\sqrt{\mathbb{E}(g)+\varepsilon}}\ge 4\alpha\sqrt{\varepsilon}\right\}\\&\le \mathcal{N}(G,\alpha\varepsilon)\exp\left\{-\frac{\alpha^2 m\varepsilon}{2c_\rho+(2/3)B}\right\}.\end{aligned}\tag{31}$$

Proposition 11. *Given $f_{\mathbf{z}}$ defined by (5), suppose that Assumption 5 holds. When $m > (1360\log(1/\delta)/\widetilde{c})^{(1+s)/s}$, then, for any $0 < \delta < 1$,*

$$\begin{aligned}\mathcal{S}_2(\mathbf{z},\lambda) \le{}& \frac{1}{2}\mathscr{E}(\pi_M(f_{\mathbf{z}})) - \mathscr{E}(f_\rho)\\&+\widetilde{c}M^{s/(1+s)}\left(\frac{1}{\lambda}\right)^{s/2(1+s)}\left(\frac{1}{m}\right)^{1/(1+s)}\end{aligned}\tag{32}$$

with probability of at least $1-\delta$, where $\widetilde{c} = (1360C_N(16)^s M^{2+s})^{1/1+s}$.

Proof. Introduce a function set $\mathscr{F}_R$ defined by Proposition 9:

$$\mathscr{F}_R = \left\{(\pi_M(f)(x)-y)^2 - (f_\rho(x)-y)^2 : f\in B_R\right\}.\tag{33}$$

Each function in $\mathscr{F}_R$ can be denoted as $g(z) = (\pi_M(f)(x)-y)^2 - (f_\rho(x)-y)^2$, where $f \in B_R$. It follows that $\mathbb{E}(g) = \mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho) \ge 0$, $(1/m)\sum_{i=1}^m g(z_i) = \mathscr{E}_{\mathbf{z}}(\pi_M(f)) - \mathscr{E}_{\mathbf{z}}(f_\rho)$, and

$$\begin{aligned}g(z) = {}&\left(\pi_M(f)(x)-f_\rho(x)\right)\\&\cdot\left(\pi_M(f)(x)+f_\rho(x)-2y\right).\end{aligned}\tag{34}$$

Also note that $\|\pi_M(f)\|_\infty \le M$ and $|f_\rho| \le M$ almost everywhere; we have

$$|g(z)| \le 8M^2.\tag{35}$$

This implies that

$$|g-\mathbb{E}(g)| \le B := 16M^2.\tag{36}$$

Additionally notice that

$$\begin{aligned}\mathbb{E}(g^2) = {}&\int_X \left(\pi_M(f)(x)-f_\rho(x)\right)^2\\&\cdot\left(\pi_M(f)(x)+f_\rho(x)-2y\right)^2 \le 16M^2\left\|\pi_M(f)\right.\\&\left.-f_\rho\right\|_\rho^2 = 16M^2\mathbb{E}(g).\end{aligned}\tag{37}$$

By Lemma 10 with $B = c_\rho = 16M^2$ and $\alpha = 1/4$, we obtain

$$\begin{aligned}&\frac{\left(\mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho)\right) - \left(\mathscr{E}_{\mathbf{z}}(\pi_M(f)) - \mathscr{E}_{\mathbf{z}}(f_\rho)\right)}{\sqrt{\mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho)+\varepsilon}}\\&\le\sqrt{\varepsilon},\end{aligned}\tag{38}$$

with probability at least

$$\begin{aligned}&1-\mathcal{N}\left(\mathscr{F}_R,\frac{1}{4}\varepsilon\right)\exp\left\{-\frac{m\varepsilon}{16\left(2c_\rho+(2/3)B\right)}\right\}\\&\ge 1-\mathcal{N}\left(\mathscr{F}_R,\frac{1}{4}\varepsilon\right)\exp\left\{-\frac{m\varepsilon}{680M^2}\right\}.\end{aligned}\tag{39}$$

Now we need to estimate covering number $\mathcal{N}(\mathscr{F}_R,(1/4)\varepsilon)$.

For arbitrary $g_1, g_2 \in \mathscr{F}_R$, we obtain

$$\begin{aligned}&|g_1(z)-g_2(z)|\\&\le |f_1(x)-f_2(x)|\left|\pi_M(f_1)(x)+\pi_M(f_2)(x)-2y\right|.\end{aligned}\tag{40}$$

Since $|\pi_M(f)(x)| \le M$ is true for any $x \in X$ and π_M is contraction mapping, we have

$$\begin{aligned}&|g_1(z)-g_2(z)| \le 4M|f_1(x)-f_2(x)|,\\&\qquad\qquad\forall f_1, f_2 \in B_R.\end{aligned}\tag{41}$$

Hence

$$\mathcal{N}\left(\mathscr{F}_R,\frac{1}{4}\varepsilon\right) \le \mathcal{N}\left(B_R,\frac{\varepsilon}{16M}\right) \le \mathcal{N}\left(B_1,\frac{\varepsilon}{16MB}\right).\tag{42}$$

According to Assumption 5, suppose that

$$C_N(16MB)^s\left(\frac{1}{\varepsilon}\right)^s - \left\{\frac{m\varepsilon}{680M^2}\right\} = \log\delta,\tag{43}$$

where $b = C_N(16MB)^s$ and $a = m/680M^2$. Then it can be written as

$$\varepsilon^{1+s} - \frac{\log(1/\delta)}{a}\varepsilon^s - \frac{b}{a} = 0.\tag{44}$$

According to Lemma 7 given in [16]

$$\varepsilon \le \max\left\{\frac{2\log(1/\delta)}{a}, \left(\frac{2b}{a}\right)^{1/(1+s)}\right\}.\tag{45}$$

Substituting it into (38) and noticing that $\sqrt{\mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho)+\varepsilon(\sqrt{\varepsilon})} \le (1/2)\mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho) + \varepsilon$, we further see that $(\mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho)) - (\mathscr{E}_{\mathbf{z}}(\pi_M(f)) - \mathscr{E}_{\mathbf{z}}(f_\rho))$ can be bounded by

$$\begin{aligned}&\frac{1}{2}\mathscr{E}(\pi_M(f)) - \mathscr{E}(f_\rho)\\&+\max\left\{\frac{1360\log(1/\delta)}{m}, \widetilde{c}B^{s/(1+s)}\left(\frac{1}{m}\right)^{1/(1+s)}\right\}.\end{aligned}\tag{46}$$

Based on Lemma 4.1 given in [14], for each $\mathbf{z} \in Z^m$ we have

$$\|f_{\mathbf{z}}\|_{K^N} \le \frac{M}{\sqrt{\lambda}}.\tag{47}$$

Hence, if

$$m > \left(\frac{1360\log(1/\delta)}{\widetilde{c}}\right)^{(1+s)/s},\tag{48}$$

we have

$$\frac{1360\log(1/\delta)}{m} \le \tilde{c}B^{s/(1+s)}\left(\frac{1}{m}\right)^{1/(1+s)}. \quad (49)$$

Proposition 11 is completed by taking $B = M/\sqrt{\lambda}$. □

According to the conclusion of Proposition 9, two important quantities $\|f_\lambda\|_\infty$ and $\|f_\lambda - f_\rho\|_\rho^2$ involved in $\mathcal{S}_1(\mathbf{z},\lambda)$ need to be bounded.

Lemma 12. *Let f_λ be defined as (13). If $L_{K^0}^{-r} f_\rho \in L^2_{\rho_X}$, the following holds:*

$$\begin{aligned} \|f_\lambda\|_{K^N} &\le \lambda^{\min\{(r-N)/2N,0\}} \left\|L_{K^0}^{-r} f_\rho\right\|_{L^2_{\rho_X}}, \\ \left\|f_\lambda - f_\rho\right\|_{L_{\rho_X}} &\le \lambda^{\min\{r/2N,1\}} \left\|L_{K^0}^{-r} f_\rho\right\|_{L^2_{\rho_X}}. \end{aligned} \quad (50)$$

It makes sense that the estimation of f_λ is extended from Lemma 4.3 in [18], where K^0 takes place of K^N. Now we discuss the second quantity. In the classical algorithm (3), when $r > 1$, the increase of smoothness of f_ρ is not able to improve the error $\|f_\lambda - f_\rho\|_{L_{\rho_X}}$, which is called the "saturation" phenomenon in the literature of inverse problems. While, for the algorithm we study, only when $r > 2N$, the saturation will happen. This shows specific advantages of using K^N instead of the original K^0 from the perspective of approximation theory.

Proof of Lemma 12. According to [17], $f_\lambda = (\lambda I + L_{K^N})^{-1} L_{K^N} f_\rho$. Notice that

$$L_{K^N} = L_{K^0}^{2N}, \quad (51)$$

and this yields

$$\begin{aligned} f_\lambda &= \left(\lambda I + L_{K^0}^{2N}\right)^{-1} L_{K^0}^{2N} f_\rho \\ &= \left(\lambda I + L_{K^0}^{2N}\right)^{-1} L_{K^0}^{2N} L_{K^0}^{r} L_{K^0}^{-r} f_\rho \\ &= \sum_{k=1}^{\infty} \frac{\lambda_k^{2N+r}}{\lambda + \lambda_k^{2N}} \left\langle L_{K^0}^{-r} f_\rho, e_k \right\rangle_{L^2_{\rho_X}} e_k, \end{aligned} \quad (52)$$

where $\{\lambda_k, e_k\}_k$ is the corresponding spectrum of the integral operator L_{K^0}. Thus

$$\begin{aligned} \|f_\lambda\|_{K^N}^2 &= \sum_{k=1}^{\infty} \frac{\lambda_k^{2N+2r}}{\left(\lambda + \lambda_k^{2N}\right)^2} \left\langle L_{K^0}^{-r} f_\rho, e_k \right\rangle_{L^2_{\rho_X}}^2 \\ &\le \lambda^{\min\{(r-N)/N,0\}} \left\|L_{K^0}^{-r} f_\rho\right\|_{L^2_{\rho_X}}. \end{aligned} \quad (53)$$

On the other hand, noting the fact that $f_\lambda - f_\rho = \lambda(\lambda I + L_{K^N})^{-1} f_\rho$ and the assumption $L_{K^0}^{-r} f_\rho \in L^2_{\rho_X}$, we have

$$\begin{aligned} \left\|f_\lambda - f_\rho\right\|_{L^2_{\rho_X}} &= \lambda \left\|\left(\lambda I + L_{K^0}^{2N}\right)^{-1} L_{K^0}^{r} L_{K^0}^{-r} f_\rho\right\|_{L^2_{\rho_X}} \\ &= \lambda \left\|\sum_{k=1}^{\infty} \alpha_k \frac{\lambda_k^r}{\lambda_k^{2N} + \lambda} e_k\right\|_{L^2_{\rho_X}} \\ &\le \lambda^{\min\{r/2N,1\}} \left\|L_{K^0}^{-r} f_\rho\right\|_{L^2_{\rho_X}} \end{aligned} \quad (54)$$

where we used the fact that $\|\alpha\|_{l^2} = \left\|L_{K^0}^{-r} f_\rho\right\|_{L^2_{\rho_X}}$.

This is the end of proving Lemma 12. □

Together with Propositions 9 and 11, Lemma 12, and the fact that $\|f_\lambda\|_\infty \le \kappa^N \|f_\lambda\|_{K^N}$, Theorem 6 is proved easily.

5. Numerical Experiments

5.1. Simulated Example. Although this paper mainly focuses on theoretical analysis, we can take some experiments to show its efficiency in practice. A simulated example is considered, where the true regression is an additive model. That is,

$$f^*(x_i) = 5f_1(x_{i1}) + 3f_2(x_{i2}) + 4f_3(x_{i3}) + 6f_4(x_{i4}) \quad (55)$$

with $f_1(u) = 2\exp(-0.1|u|)$, $f_2(u) = (2u-1)^2$, $f_3(u) = \sin(\pi u)/(2-\sin(\pi u))$, and $f_4(u) = 0.2\sin(\pi u) + 0.1\cos(\pi u) + 0.3\sin^2(\pi u) + 0.5\cos^3(\pi u) + 0.5\sin^3(\pi u)$. Firstly, generate x_{ij} which are independent from $U(-0.5, 0.5)$; then generate y_i by $y_i = f^*(x_i) + \epsilon_i$ with $\epsilon_i \sim N(0, 0.1)$.

For the above example, different scenarios are taken into account, with $(m = 200, n = 30)$, $(m = 400, n = 100)$, and $(m = 500, n = 120)$, and each scenario is repeated 50 times. We here use the widely used Gaussian kernel, $K_\sigma(x,u) = \exp(-\|x-u\|^2/\sigma^2)$, where parameter σ will be specified by conducted 10-fold cross-validation on each data set. Besides, as we mentioned before, we start with a weak kernel K and search a better one somehow iteratively. A standard weak kernel is defined as follows: $K_{\text{weak}}(x,y) = e^{-\lambda\|x-y\|}$, where λ is an adjustable parameter, where we specify it as $\lambda = 0.1$.

The performance of various methods is measured by the MSE, where MSE represents the relative mean squared error for each kernel-based regression. The averaged performance measures are summarized in Table 1. Note that SKM represents the single kernel method with the Gaussian kernel, UKM represents the proposed method without using any unlabeled data, and SEKM represents the proposed method with using all the unlabeled data.

From Table 1 we find that using the proposed kernel learning method on the data sets generates better prediction accuracy than using a single kernel. Probably, the true function is more complicated, and in this case the Gaussian kernel has a limited learning ability. Thus, learning to start with a weak kernel implies that the hypothesis space is much larger than that induced by the Gaussian kernel, so that

Table 1: Performance obtained using various kernel-based methods.

Method/sample Size	($m = 200$, $n = 30$)	($m = 400$, $n = 100$)	($m = 500$, $n = 120$)
SKM	0.090 ± 0.032	0.093 ± 0.022	0.095 ± 0.018
UKM	0.085 ± 0.052	0.082 ± 0.037	0.080 ± 0.041
SEKM	0.080 ± 0.062	0.077 ± 0.042	0.075 ± 0.036

Table 2: On test set obtaining best accuracy for Boston housing.

Method/sample Size	($m = 300$, $n = 30$)	($m = 350$, $n = 40$)	($m = 426$, $n = 50$)
SKM	1.774 ± 0.0931	1.712 ± 0.0835	1.675 ± 0.0733
UKM	1.785 ± 0.851	1.684 ± 0.737	1.633 ± 0.841
SEKM	1.764 ± 0.851	1.705 ± 0.752	1.625 ± 0.804

the true function can be learnt well by our algorithm. Moreover, from the last row of Table 1, we see that using the unlabeled data by our algorithm can further reduce the prediction error, as we expect in theory.

5.2. Real Example. The proposed method is also applied to a real example, the Boston housing data, which is publicly available. The Boston housing data concerns the median value of owner-occupied homes in each of the 506 census tracts in the Boston Standard Metropolitan Statistical Area in 1970. It consists of 13 variables, including per capita crime rate by town (CRIM), proportion of residential land zoned for lots over 25,000 square feet (ZN), proportion of nonretail business acres per town (INDUS), Charles River dummy variable (CHAS), nitric oxides concentration (NOX), average number of rooms per dwelling (RM), proportion of owner-occupied units built prior to 1940 (AGE), weighted distances to five Boston employment centers (DIS), index of accessibility to radial highways (RAD), full-value property-tax rate per $10000 (TAX), pupil-teacher ratio by town (PTRATIO), the proportion of blacks by town (B), and lower status of the population (LSTAT), which may affect the housing price.

In our analysis, all the variables are standardized. To compute the averaged prediction error, each dataset is randomly split into two parts: training data and testing data with the number 30. To show the performance our method compared with a single kernel method, we split the training data with three different scenarios with ($m = 300, n = 30$), ($m = 350, n = 40$), and ($m = 426, n = 50$), and each scenario is repeated 50 times. Besides, parameter σ will be specified by conducted 10-fold cross-validation on each data set. The prediction performance of the single kernel method via the proposed method is summarized in Table 2.

As shown in the table, the proposed method is significant in terms of prediction accuracy, except that one of six results in Table 2 is of poor performance compared to the single kernel method. This practical result may be acceptable, since we do not know the underlying rule for this real data, and it is hard to ensure a perfect performance in various settings. Totally, to some extent, the proposed method is a simple but efficient kernel learning method in the family of kernel methods.

6. Conclusions and Discussions

This paper mainly discussed the kernel learning problems within semisupervised learning setting. Our candidate kernel sequence is generated by a simple iterative procedure using large amounts of unlabeled data. Under mild assumptions on target function, it is shown that we can match a kernel theoretically to outperform efficiently the sample error induced by one-kernel learning. This also shows that the learning kernel function outperforms traditional kernel-based learning algorithms in our case. Moreover, a simulation example and a real data experiment are implemented, respectively, to show the effectiveness of our proposed method.

We note that the space complexity of function space in the paper is described by covering number, which was a straightaway conception. Yet it is not a perfect choice theoretically. Combining with the way in which the kernel function was formed in the text, we can replace Assumption 5 with the eigenvalue asymptotic behavior assumption on the integration operator L_{K^0}. Based on its relationship among entropy and the Rademacher complexity, a better theoretical results may be achieved. This will be our subsequent work in the future. We attempt to explore some intrinsic structure of input space by selecting an appropriate kernel. Perhaps, there are other more effective ways to explore these underlying structures.

Appendix

Integral operator L_K has the following properties which were proved in [15].

(1) L_K is a positive, self-adjoint, and compact operator from $L^2_{\rho_X}(X)$ to $L^2_{\rho_X}(X)$. Consequently, according to classical spectral theorem, its standard eigenfunctions $e_1, e_2 \ldots$ consist of a family of orthogonal bases on $L^2_{\rho_X}$. The discrete operator spectrum and the corresponding eigenvalues $\lambda_1, \lambda_2 \ldots$ are finite or monotonically decreasing; $\lim_{i\to\infty}\lambda_i = 0$.

(2) For each $\eta > 0$, $L^\eta_K : L^2_{\rho_X}(X) \to L^2_{\rho_X}(X)$ is defined as

$$L^\eta_K(f)(x) = \sum_{k=1}^{\infty} \lambda_k^\eta \langle f, e_k\rangle e_k, \quad \forall f \in L^2_{\rho_X}(X). \tag{A.1}$$

Denote $\Gamma = \{i : \lambda_i > 0\}$; $\{\sqrt{\lambda_i} e_i : i \in \Gamma\}$ forms a set of orthogonal bases of $\mathcal{H}_K$. Furthermore $L^{1/2}_K$ is isomorphic mapping between $\mathcal{H}_K$ and L^2_μ. Particularly, the following holds:

$$\|f\|_{L^2_{\rho_X}} = \left\|L^{1/2}_K f\right\|_K. \tag{A.2}$$

Proof of Corollary 7. Notice that when $1 < N < r < 2N$, $\lambda^{\min\{(r-N)/2N,0\}}$ in the conclusion of Proposition 3 is a constant, and $\lambda^{\min\{r/N,2\}} = \lambda^{r/N}$. By taking

$$\lambda^{r/N} = \left(\frac{1}{\lambda}\right)^{s/2(1+s)} \left(\frac{1}{m}\right)^{1/(1+s)}, \tag{A.3}$$

we obtain $\lambda = (1/m)^{2N/(2r(1+s)+sN)}$. Thus

$$\widetilde{\mathcal{S}}(\mathbf{z}, \lambda) = \log\left(\frac{2}{\delta}\right) \mathcal{O}\left(\frac{1}{m}\right)^{2r/(2r(1+s)+sN)}. \quad (A.4)$$

In addition, if $K^N \in C^\infty(X)$, the corresponding s infinitely approaches 0 by the classic conclusion described in [14]. Thus, we complete the proof of Corollary 7. □

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The second author's research is supported partially by National Natural Science Foundation of China (no. 11301421) and Fundamental Research Funds for the Central Universities of China (Grant nos. JBK141111, 14TD0046, and JBK151134).

References

[1] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, NY, USA, 2nd edition, 1998.

[2] B. Scholkopf and A. Smola, *Learning with Kernels*, MIT Press, Cambridge, Mass, USA, 2002.

[3] O. Chapelle, J. Weston, and B. Schölkopf, "Cluster kernels for semi-supervised learning," in *Proceedings of the 16th Annual Neural Information Processing Systems Conference (NIPS '02)*, New York, NY, USA, December 2002.

[4] M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien, "l_p-norm multiple kernel learning," *Journal of Machine Learning Research*, vol. 12, pp. 953–997, 2011.

[5] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semi-definite programming," *Journal of Machine Learning Research*, vol. 5, pp. 27–72, 2004.

[6] Q. Wu, Y. M. Ying, and D. X. Zhou, "Multi-kernel regularized classifiers," *Journal of Complexity*, vol. 23, no. 1, pp. 108–134, 2007.

[7] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf, "Large scale multiple kernel learning," *Journal of Machine Learning Research*, vol. 7, pp. 1531–1565, 2006.

[8] C. Cortes, "Invited talk: can learning kernels help performance," in *Proceedings of the 26th Annual ICML*, New York, NY, USA, 2009.

[9] M. Kloft and G. Blanchard, "The local Rademacher complexity of l_p-norm multiple kernel learning," *Journal of Machine Learning Research*, vol. 13, pp. 2465–2501, 2012.

[10] S.-G. Lv and J.-D. Zhu, "Error bounds for l^p-norm multiple kernel learning with least square loss," *Abstract and Applied Analysis*, vol. 2012, Article ID 915920, 18 pages, 2012.

[11] S. Lv and F. Y. Zhou, "Optimal learning rates of l_p-type multiple kernel learning under general conditions," *Information Sciences*, vol. 294, pp. 255–268, 2015.

[12] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American Mathematical Society*, vol. 68, pp. 337–404, 1950.

[13] I. Steinwart, D. Hush, and C. Scovel, "Optimal rates for regularized least squares regression," in *Proceedings of the 22nd Annual Conference on Learning Theory (COLT '09)*, pp. 79–93, Montreal, Canada, June 2009.

[14] Q. Wu, Y. M. Ying, and D.-X. Zhou, "Learning rates of least-square regularized regression," *Foundations of Computational Mathematics*, vol. 6, no. 2, pp. 171–192, 2006.

[15] F. Cucker and D.-X. Zhou, *Learning Theory: An Approximation Theory Viewpoint*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, UK, 2007.

[16] F. Cucker and S. Smale, "On the mathematical foundations of learning," *Bulletin of the American Mathematical Societ*, vol. 39, no. 1, pp. 1–49, 2002.

[17] S. Smale and D.-X. Zhou, "Estimating the approximation error in learning theory," *Analysis and Applications*, vol. 1, no. 1, pp. 17–41, 2003.

[18] H. W. Sun and Q. Wu, "Regularized least square regression with dependent samples," *Advances in Computational Mathematics*, vol. 32, no. 2, pp. 175–189, 2010.

[19] U. V. Luxburg and B. Schölkopf, "Statistical learning theory: models, concepts, and results," *Handbook of the History of Logic*, vol. 10, pp. 651–706, 2011.