Computer Science and Information Systems 21(1):143–166 https://doi.org/10.2298/CSIS230124001D

MK-MSVCR: An Efficient Multiple Kernel Approach to Multi-class Classification

Zijie Dong[1,2], Fen Chen[3], and Yu Zhang[4]

[1] School of Mathematics and Statistics,
Bigdata Modeling and Intelligent Computing research institute,
Hubei University of Education,
Second Gaoxin Road, Wuhan, 430205, China
[2] Hubei Key Laboratory of Applied Mathematics,
Faculty of Mathematics and Statistics,
Hubei University,
Wuhan 430062, China
zjdong07@163.com
[3] School of Finance, Hubei University of Economics,
Wuhan, 430205, China
fenfen_chen@163.com
[4] School of Mathematics and Statistics,
Hubei University of Education,
Second Gaoxin Road, Wuhan, 430205, China
romeozyu@163.com

Abstract. This paper introduces a novel multi-class support vector classification and regression (MSVCR) algorithm with multiple kernel learning (MK-MSVCR). We present a new MK-MSVCR algorithm based on two-stage learning (MK-MSVCR-TSL). The two-stage learning aims to make classification algorithms better when dealing with complex data by using the first stage of learning to generate "representative" or "important" samples. We first establish the fast learning rate of MK-MSVCR algorithm for multi-class classification with independent and identically distributed (i.i.d.) samples amd uniformly ergodic Markov chain (u.e.M.c.) smaples, and prove that MK-MSVCR algorithm is consistent. We show the numerical investigation on the learning performance of MK-MSVCR-TSL algorithm. The experimental studies indicate that the proposed MK-MSVCR-TSL algorithm has better learning performance in terms of prediction accuracy, sampling and training total time than other multi-class classification algorithms.

Keywords: multi-class classification, multiple kernel learning, learning rate, support vector classification and regression.

1. Introduction

Support vector machine (SVM) is an effective and famous algorithm with good generalization ability for classification. In practical problems, there are many multi-classification problems such as fault diagnosis problems, disease classification and so on. Many SVM-based methods are used to handle multi-class classification problems [2,27,18,32]. For multi-class SVM, there are two main frameworks: "all-together" method [27,18,9] and

"decomposition-reconstruction" method [3,10,15]. For the "all-together" method, we usually obtain a discrimination function by solving a single majorization problem such as AIO method [27,18,11,9]. For the "decomposition-reconstruction" method, the discrimination function is obtained by handling a series of binary classification problems, which consist of two classical approaches, "one-versus-rest" (OVR) method [3,10] and "one-versus-one" (OVO) method [15]. The disadvantage of OVR method is that almost all the binary problems are unbalanced and the shortcoming of OAO method is that for each binary category, the information of the remaining categories is neglected. Thus a new method, support vector classification and regression for multi-class classification problem, is proposed by Angulo et al. [1]. The information of all samples is used to classify by MSVCR algorithm. MSVCR has been regarded as a very important method to conquer the disadvantages of tradition multi-class classifications algorithms [1,8,7].

SVM solves nonlinear classification problems by introducing kernel functions, called kernel methods. Although the kernel method can be used to solve some complex problems, it brings many interdisciplinary challenges in statistics, optimization theory and applications [1]. Choosing the optimal kernel and its parameters often has to be decided by a human user with prior knowledge of the data. In other words, the classical classifier is based on a single kernel, while in practice, the ideal classifier is usually based on the combination of multiple kernels, i.e. multiple kernel learning.

Therefore, multiple kernel learning has been widely concerned and studied. For example, Lanckriet et al. [16] introduced the method which learns the kernel matrix with semi-definite programming to search the optimal of unrestricted kernel combination weights and showed that multiple kernel learning is comparable with the best soft margin SVM for radial basis function (RBF) kernel. Luo et al. [20] introduced a theoretically motivated and efficient online learning algorithm for the multiple kernel learning problem. In recent years, the multiple kernel learning method of iteratively updating kernel weights to obtain the optimal kernel combination has been successfully applied in many fields. For example, Chavaltada et al. [5] proposed a method of automated product categorisation by using multiple kernel learning to improve feature combination in e-commerce. Wilson et al. [28] applied multiple kernel learning to genomic data mining and prediction. Lauriola et al. [17] enhanced deep neural networks via multiple kernel learning, and the method proposed in [17] gave an effective way to design the output computation in deep networks. Wang et al. [26] propose a novel depth-width-scaling multiple kernel learning (DWS-MKL) algorithm that can adapt to data of different dimensions and sizes. In addition, machine learning algorithms are used in various fields, such as weather forecasting and smart city construction [4,33].

However, when the sample size is large, the complexity of the multi-core learning algorithm is very high. This means that although multiple kernel learning methods have good learning performance, and multiple kernel learning methods are usually very time-consuming and difficult to achieve even when the training sample size is large. Then a problem is posed:
How to reduce the algorithmic complexity of the multiple kernel learning method while maintaining the better classification accuracy?

To solve this problem, we present the idea of two-stage learning for multiple kernel algorithm in this paper. The reasons of introducing two-stage learning are as follows: First, the capacity of data is growing rapidly and the scale of data is getting larger and larger, so

the classical multiple kernel learning methods usually time-consuming and even difficult to implement in the case of a large training sample size. Second, the larger the amount of data, the lower the value density of data will be, which means that there are many noise samples in big data. A large number of machine learning experiments indicate that the noise samples will not only lead to the increase of storage space, but also affect the accuracy and efficiency of learning. By the statistical learning theory in [25], the samples that are closer (or the closest) to the interface of different classes datasets are "important" samples for classification problem. Thus, the two-stage learning aim to generate the "representative" or "important" samples by using the first stage learning. To our knowledge, this is the first algorithm of multiple kernel learning method to deal with multi-class classification non-i.i.d. samples. Therefore, in this paper we analyse the generalization of MK-MSVCR method for u.e.M.c. samples and i.i.d. observations, respectively. The main innovations of this paper can be stated as follows.

• The generalization bound of MK-MSVCR based on u.e.M.c. samples is obtained and the optimal learning rate is established.

• A new MK-MSVCR algorithm, MK-MSVCR-TSL is proposed. The numerical experiments show that the proposed algorithm has competitive performance.

The rest of this article is arranged as follows. Section 2 formulates the classical MSVCR with multiple kernel learning. Section 3 introduces a new MK-MSVCR algorithm based on two-stage learning (MK-MSVCR-TSL) and analyzes algorithmic complexity. The main theoretical results of the proposed MK-MSVCR with u.e.M.c. and i.i.d. samples are given in Section 4. The numerical experimental studies are presented in Section 5. Finally, we conclude this paper in Section 6.

2. MK-MSVCR learning machine

We assume that the training set $\mathbf{z} = \{(x_i, y_i)\}_{i=1}^m$ are drawn from an unknown probability distribution ρ defined on the space $\mathcal{Z} = X \times Y$, where X is the input space and Y is the corresponding output space. For multi-class problems, we usually assume that $y_i \in \{1, \ldots, k\}$, where k is the number of class.

The classifier of MSVCR algorithm depends on the reproducing kernel Hilbert space (RKHS) $\mathcal{H}_K$. Furthermore, for the given training set $\mathbf{z}$, a decision function $\phi(x)$ is found with outputs $\{-1, 0, +1\}$:

$$\phi(x_j) = \begin{cases} +1, j = 1, \ldots, m_1 \\ -1, j = m_1 + 1, \ldots, m_1 + m_2 \\ 0, \quad j = m_1 + m_2 + 1, \ldots, m. \end{cases}$$

Without loss of generality, $m_{12} = m_1 + m_2$ patterns corresponds the case of two classes to be separated, and $m_3 = m - m_{12}$ patterns corresponds the case of rest classes, which will be labeled 0. The multi-class classification framework of MSVCR can be stated as

follows:

$$
\begin{aligned}
f_{\mathbf{z}} = &\arg \min_{f\in\mathcal{H}_K} \lambda\|f\|_K^2 + \frac{1}{m_{12}}\sum_{j=1}^{m_{12}}\xi_j + \frac{1}{m_3}\sum_{j=m_{12}+1}^{m}(\varphi_j^+ + \varphi_j^-)\\
s.t.\ \ & y_j(\sum_{i=1}^{m}\alpha_i K(x_j,x_i)) \geq 1-\xi_j, j=1,\ldots,m_{12},\\
& -\varepsilon-\varphi_j^- \leq \sum_{i=1}^{m}\alpha_i K(x_j,x_i) \leq \varepsilon+\varphi_j^+, j=m_{12}+1,m_{12}+2,\ldots,m,\\
& \xi_j,\varphi_j^+,\varphi_j^- \geq 0, 0\leq\varepsilon<1.
\end{aligned}
$$

In practice, an ideal classifier is usually based on a combination of multiple kernels. Thus we also present the MSVCR algorithm based on multiple kernels learning as follows. We assume that there are n positive definite kernels $K_1,\ldots,K_n$, each RKHS $\mathcal{H}_p$ is associated with a Mercer kernel $K_p : X\times X\to\mathbb{R}, 1\leq p\leq n$. By the reproducing property of $\mathcal{H}_p$, we have $\langle K_{p,x}, g\rangle_{K_p} = g(x), \forall x\in X, \forall g\in H_p$. Let $\mathcal{C}(X)$ be the space of continuous functions with the norm $\|f_p\|_\infty = \sup_{x\in X}|f_p|$ and $\kappa = \sup_{x\in X}\sqrt{K_p(x,x)}$. It follows that $\|f_p\|_\infty \leq \kappa\|f_p\|_{K_p}, \forall f_p\in\mathcal{H}_p, 1\leq p\leq n$. We finally use the multiple kernel space $\bar{\mathcal{H}}_K = \mathcal{H}_{K_1}\times\cdots\times\mathcal{H}_{K_n}$. $\bar{\mathcal{H}}_K$ is an RKHS with the kernel $\bar{K}(\cdot,x)$, which has following form:

$$\bar{K}(\cdot,x) = \sum_{p=1}^{n} d_p K_p(\cdot,x),$$

where $\sum_{p=1}^{n} d_p = 1, d_p\geq 0$. Therefore, any $f\in\bar{\mathcal{H}}_K$ has the form $f=\sum_{p=1}^{n} d_p f_p, f_p\in\mathcal{H}_p$. The MK-MSVCR algorithm depends on RKHS $\bar{\mathcal{H}}_K$, which is defined as

$$
\begin{aligned}
f_{\mathbf{z}} = &\arg \min_{f\in\mathcal{H}_K} \lambda\sum_{p=1}^{n} d_p\|f_p\|_{K_p}^2 + \frac{1}{m_{12}}\sum_{j=1}^{m_{12}}\xi_j + \frac{1}{m_3}\sum_{j=m_{12}+1}^{m}(\varphi_j^+ + \varphi_j^-) \qquad (1)\\
s.t.\ \ & y_j(\sum_{i=1}^{m}\alpha_i \bar{K}(x_j,x_i)) \geq 1-\xi_j, j=1,\ldots,m_{12},\\
& -\varepsilon-\varphi_j^- \leq \sum_{i=1}^{m}\alpha_i \bar{K}(x_j,x_i) \leq \varepsilon+\varphi_j^+, j=m_{12}+1,m_{12}+2,\ldots,m,\\
& \xi_j,\varphi_j^+,\varphi_j^- \geq 0, 0\leq\varepsilon<1,\\
& \sum_{p=1}^{n} d_p = 1, d_p\geq 0.
\end{aligned}
$$

Here, λ controls the complexity of the function in the multiple kernel space. For simplicity, we take $d_p = 1/n$ in this paper. The corresponding decision function $\phi_{\mathbf{z}}(x)$ of MK-MSVCR algorithm (1) is defined as

$$
\phi_{\mathbf{z}}(x) = \begin{cases} +1, \text{ if } f_{\mathbf{z}}(x)\geq\varepsilon_0 \\ -1, \text{ if } f_{\mathbf{z}}(x)\leq-\varepsilon_0 \\ 0, \quad \text{otherwise,} \end{cases} \qquad (2)
$$

where ε_0 is a threshold value.

Define loss function $V(y_j, f(x_j)) = C_1(1-y_j f(x_j))_+ \cdot \mathbf{1}_{\{y_j \neq 0\}} + C_2(y_j - f(x_j))^2 \cdot \mathbf{1}_{\{y_j=0\}}$, where C_1, C_2 are two positive constants. The MK-MSVCR algorithm (1) can be written as

$$f_{\mathbf{z}} = \arg\min_{f\in\vec{\mathcal{H}}_K} \Big\{ \frac{1}{m}\sum_{i=1}^{m} V(y_j, f(x_j)) + \lambda \|f\|_K^2 \Big\}, \tag{3}$$

where $\|f\|_K^2 = \frac{1}{n}\sum_{p=1}^{n} \|f_p\|_{K_p}^2$ is regularization term and $\lambda > 0$ is a regularization parameter. The empirical risk and the corresponding generalization error are defined as $\mathcal{E}_{\mathbf{z}}(f) = \frac{1}{m}\sum_{i=1}^{m} V(y_i, f(x_i))$, $\mathcal{E}(f) = E[V(y, f(x))] = \int_Z V(y, f(x))d\rho$, then the MK-MSVCR algorithm (3) can be rewritten as $f_{\mathbf{z}} = \arg\min_{f\in\vec{\mathcal{H}}_K} \big\{\mathcal{E}_{\mathbf{z}}(f) + \lambda\|f\|_K^2\big\}$.

3. Algorithm and Algorithmic complexity

In this section, we present the MK-MSVCR algorithm with two-stage learning (MK-MSVCR-TSL) and then we analyze the algorithmic complexity of the MK-MSVCR-TSL algorithm.

3.1. MK-MSVCR-TSL algorithm

Inspired by the works in [22,21,13,34], We combine the MK-MSVCR algorithm that the multiple kernel multi-class classification algorithm for processing more complex data with two-stage learning (MK-MSVCR-TSL) to improve the classification rate without reducing the classification accuracy. Now the proposed MK-MSVCR-TSL algorithm can be stated as follows.

For simplicity, all the experimental results of this paper are based on $q = 1$. In the preprocessing step, all data are normalized to avoid the influence of numerical range on characteristic attributes and make numerical calculation easier [14,19]. We use a random process to divide each data sets into two different parts, where four quarters is divided into the training set D_{train}, one quarter is divided into the test set D_{test}. Let k be the number of classed, m be the total number of training samples. For MK-MSVCR algorithm based on randomly independent samples, we sample m training samples that are drawn randomly from the given training set D_{train} and denote it as $\mathbf{z}$. Training the sample set $\mathbf{z}$ by algorithm (3) and obtain a classifier $\phi_{\mathbf{z}}$. We test $\phi_{\mathbf{z}}$ on the given testing set, and calculate the corresponding misclassification rate. For MK-MSVCR-TSL algorithm, we state the algorithm as follows: (i) for the first stage, we sample randomly $N = m/2$ training samples that are drawn randomly from the given training set D_{train} and denote it as $\mathbf{z}^0$. Training $\mathbf{z}^0$ by algorithm (3) and obtain a classifier $\phi_{\mathbf{z}^0}$. (ii) for the second stage, we use $\phi_{\mathbf{z}^0}$ to define the acceptance probabilities and generate the Markovian chain samples $\mathbf{z}^1$, which consist of N samples. Training the sample set $\mathbf{z}^1$ by algorithm (3) and obtain a classifier $\phi_{\mathbf{z}^1}$. (iii) We test $\phi_{\mathbf{z}^1}$ on the given testing set, and calculate the corresponding misclassification rate.

Remark 1. To have a better understanding of Algorithm 1, we give the following remarks. Since we only have the training set D_{train}, to define the transition probabilities of Markovian resampling, we first draw randomly a training set $\mathbf{z}^0$ with N training samples, and

Algorithm 1: MK-MSVCR-TSL

Input: D_{train}, m, N.
Output: $\phi_{\mathbf{z}}(x)$.
For each sample $z \in D_{train}$, compartmentalize the training set into three parts: zero class, positive class, negative class;
(**the first stage**) generate randomly N samples $\mathbf{z}^0 := \{z_t\}_{t=1}^N$ from D_{train};
get a model $\mathbf{f}_{\mathbf{z}^0}$ by algorithm (3) with $\mathbf{z}^0$ and its corresponding classifier $\phi_{\mathbf{z}^0}$;
(**the second stage**)
generate randomly a sample z_a from D_{train}; $\mathbf{z}^{i+1} \leftarrow \{z_a\}$;
$count_{y_a} \leftarrow count_{y_a} + 1$;
repeat
  generate another random sample z_b from D_{train};
  $\mathrm{P} = e^{-\ell(\phi_{\mathbf{z}^i}, z_b)} / e^{-\ell(\phi_{\mathbf{z}^i}, z_a)}$;
  if $\mathrm{P} = 1$ *and* $y_a = y_b$ *and* $count_{y_b} < N/k$ **then**
    add z_b to $\mathbf{z}^{i+1}$; $count_{y_b} \leftarrow count_{y_b} + 1$;
  else if $(\mathrm{P} = 1$ *and* $y_a \neq y_b$ *and* $count_{y_b} < N/k)$ *or* $(\mathrm{P} < 1$ *and* $count_{y_b} < N/k)$ **then**
    add z_b to $\mathbf{z}^{i+1}$; $count_{y_b} \leftarrow count_{y_b} + 1$;
  end
  If ℓ candidate samples z_b can not be accepted continually, then accepting z_b; $z_a \leftarrow z_b$;
until $size(\mathbf{z}^{i+1}) \geq N$;
get the model with $\mathbf{z}^{i+1}$, and obtain the decision function $\phi_{\mathbf{z}^{i+1}}$ by equation (2);
return $\phi_{\mathbf{z}} = \phi_{\mathbf{z}^q}$;

obtain a preliminary learning classifier $\phi_{\mathbf{z}^0}$. Then we use the information of $\phi_{\mathbf{z}^0}$ to define the transition probabilities P (or $\mathrm{P}_1, \mathrm{P}_2, \mathrm{P}_3$ defined in Algorithm 1). Since these acceptance probabilities $\mathrm{P}, \mathrm{P}_1, \mathrm{P}_2, \mathrm{P}_3$ are positive, we can obtain an u.e.M.c sequence $\mathbf{z}^1$. Specially, if the acceptance probabilities $\mathrm{P}, \mathrm{P}_1, \mathrm{P}_2, \mathrm{P}_3$ are equal to 1, which is the case of random resampling. This reflects that the classical MSVCR algorithm based on i.i.d. samples can be regarded as the special case of Algorithm 1 with $q = 0, N = m$, and the acceptance probabilities $\mathrm{P}, \mathrm{P}_1, \mathrm{P}_2, \mathrm{P}_3$ are equal to 1. This implies Algorithm 1 extended the classical MSVCR algorithm introduced in [1] from i.i.d. samples to non-i.i.d. samples. In addition, since as the value $e^{-\ell(\phi_{\mathbf{z}^i})}$ of the current sample z_a is smaller, the acceptance probabilities will be smaller, which implies that the candidate sample z_b will be accepted with a smaller probability, which means that generating the Markovian samples $\mathbf{z}^j (1 \leq j \leq q)$ will be time-consuming. To quickly draw the Markovian samples $\mathbf{z}^1$, we use the technical parameters $\ell = 30$ in the following all the experimental results.

3.2. Explanations of algorithm

(i) Comparing Algorithm 1 with algorithm introduced in [7]

Dong et al. aim to extend the case of i.i.d. samples to non-i.i.d. samples, and study the theory and algorithm of non-i.i.d. multi-classification methods in [7]. However, we are now working to extend single kernel learning to multiple kernels learning, studying the theory and algorithm of non-i.i.d. multiple kernels multi-classification methods. SVM can solve nonlinear classification problem by kernel method. Choosing the optimal kernel from a set of candidates and its parameters is a central choice, which usually must be

made by a human user using the prior knowledge of data. In other words, the classical kernel-based classifiers are based on a single kernel. With the advent of the big data era, the data is diversified and the data characteristics are complicated. In practice, an ideal classifier is usually based on a combination of multiple kernels. For complex big data samples, we hope to use multiple kernels learning to improve classification efficiency without increasing classification time. Dong et al. pointed out in [7,9] that a proper q value can reduce the sampling and training total time of the algorithm without reducing the accuracy for Markovian resampling. If the total time is a concern, we should choose a smaller q value. Multiple kernels learning can effectively reduce the misclassification rates. However, because the combined kernel learning takes time, in order to effectively reduce the total time, we choose a smaller q value, let q=1 in this paper. So we propose MK-MSVCR-TSL method. The effectiveness of our method is also proved in the following experimental comparison.

(ii) Comparing Algorithm 1 with algorithm introduced in [1]

Comparing Algorithm 1 with algorithm introduced in [1], we can find that the differences are obvious: First, Algorithm 1 is a multiple kernels algorithm while the algorithm presented in [1] is a single kernel learning. This implies that Algorithm 1 extended the algorithm presented in [1] from a single kernel to the case of multiple kernels. In other words, the algorithm of [1] is a special case of Algorithm 1 proposed in this paper. Second, the algorithm presented in [1] is for the case that the samples are random and independent. However, our proposed Algorithm 1 is designed for not random and independent samples. This implies that Algorithm 1 improve algorithm of [1].

(iii) Comparing Algorithm 1 with algorithm of [31]

we can find that although Algorithm 1 has many steps similar to that of [31] and the two same technical parameters are adopted, the differences are obvious: First, Algorithm 1 is a multiple kernels algorithm while the algorithm presented in [31] is a SVM algorithm with a single kernel. Second, Algorithm 1 is a multi-class classification algorithm while [31] is a two-class classification algorithm. This implies that algorithm of [31] can be regarded as the special case of Algorithm 1 with $k = 2$ and $N = m/2$. Third, the total size of training samples for [31] is $2m$, which is double times of the size m for the classical algorithm. This implies that compared to the classical SVM, algorithm of [31] is time-consuming. While the learning process of Algorithm 1 can be seen as 2 times "batch learning", and the total size of training samples is $m = 2N$, which is same as the classical SVM.

3.3. Algorithmic complexity

There are m samples. Here k is the number of class. In Algorithm 1, the complexity of a single kernel MSVCR is $O(\frac{K(K-1)}{2}m^3)$. In this paper, we choose the mean weights as the kernel weights of MK-MSVCR. Therefore, the complexity of MK-MSVCR algorithm is about $O(\frac{K(K-1)}{2}m^3)$. But in Algorithm 1, we divided m training samples into $q+1$ pieces, thus, the complexity of Algorithm 1 is about $O(\frac{K(K-1)}{2}(\frac{m}{q+1})^3)$. If we assume $(q+1) \approx m^\gamma$ for any $\gamma > 0$, it is obvious that the complexity of our proposed MK-MSVCR-TSL algorithm in this paper is lower than that of the classical MK-MSVCR.

4. Estimating the Generalization Bounds

In this section, our target is to bound the generalization of MK-MSVCR-TSL algorithm. In this paper we first consider the case of uniformly ergodic Markov chain (u.e.M.c.) observations. As the special case of u.e.M.c., we also consider i.i.d. samples.

By the definitions of the acceptance probabilities in Algorithm 1, we can find that the acceptance probabilities are positive. In addition, the size m of training samples is finite. By the theory of Markov chain [24], we can conclude that the samples generated in Algorithm 1 is uniformly ergodic Markov chains (u.e.M.c.). Then we present the definition of u.e.M.c. as follows: Let $(\mathcal{Z}, \mathcal{A})$ be a measurable space, and $\{Z_t\}_{t\geq 1}$ be a Markov chain with transition probability measures $\mathrm{P}^n(S|z_j),\ S \in \mathcal{A},\ z_j \in \mathcal{Z}$. $\mathrm{P}^n(S|z_j)$ is defined as $\mathrm{P}^n(S|z_j) = \mathrm{P}\{z_{j+n} \in S|Z_i, i < j, Z_j = z_j\}$.

For any $S \in \mathcal{A}, z_j \in \mathcal{Z}$, if the transition probability $\mathrm{P}^n(S|z_j)$ satisfies $\mathrm{P}^n(S|z_j) = \mathrm{P}\{z_{j+n} \in S|Z_j = z_j\}$, which is the so called the Markov property of $\{Z_t\}_{t\geq 1}$. This property indicates that given the current state z_j, the past state $z_i, i < j$ is independent of the future state z_{n+j}. By these notations, the definition of u.e.M.c. is given as follows [23].

Definition 1. *A Markov chain $\{Z_t\}_{t\geq 1}$ is uniformly ergodic, if for some $\beta < 1, \varphi < \infty$, $||\mathrm{P}^n(\cdot|z) - \varpi(\cdot)||_{TV} \leq \varphi\beta^n$, for all $n = 1, 2, \ldots$, where $\varpi(\cdot)$ is the stationary distribution of $\{Z_t\}_{t\geq 1}$, $||\cdot||_{TV}$ is the total variation distance, which is defined as $||\mu_1 - \mu_2||_{TV} = \sup_{S\in\mathcal{A}} |\mu_1(S) - \mu_2(S)|$ for two measures μ_1, μ_2 defined on the space $(\mathcal{Z}, \mathcal{A})$.*

To measure the generalization ability of MK-MSVCR-TSL algorithm, we should bound the excess generalization error $\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B)$, where f_B is the minimizer of $\mathcal{E}(f)$ for all measurable function f. The corresponding best classifier ϕ_B is the Bayes rule, $\phi_B := \arg\min_{j\in Y} \sum_{y\in Y} P_y(x) \cdot \mathbf{1}_{\{y\neq j\}}, x \in X$. To estimate $\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B)$, we also present the following some definitions and assumptions.

Definition 2. *[29] We call the function f_B is approximated if there exists a constant C_q with an exponent $0 < q \leq 1$ such that $D(\lambda)$ satisfies $D(\lambda) \leq C_q\lambda^q$ for any $\lambda > 0$.*

For simplicity, we take $C_q = 1$ in this paper. Since the minimization (3) is taken for the discrete quantity $\mathcal{E}(f)$, bound the excess generalization error involves the capacity of $\bar{\mathcal{H}}_K$. In this paper the capacity of function set is managed by the covering number.

Definition 3. *For a subset $\mathcal{F}$ of a metric space and $\epsilon > 0$, the covering number $\mathcal{N}(\mathcal{F}, \epsilon)$ is the minimal $n_1 \in \mathbf{N}$ such that there exist n_1 disks with radius ϵ covering $\mathcal{F}$.*

For $R > 0$, $\mathcal{B}_R = \{f \in \bar{\mathcal{H}}_K : ||f||_K \leq R\}$. It is a subset of $\mathcal{C}(X)$ and the covering number is well defined [31,29,35]. For any $\epsilon > 0$, $\mathcal{N}(\epsilon) = \mathcal{N}(\mathcal{B}_1, \epsilon)$ is expressed as the covering number of $\mathcal{B}_1$.

Definition 4. *[29] The RKHS $\bar{\mathcal{H}}_K$ is said to have a polynomial complexity exponent $s > 0$ if there is some constant $C_s > 0$ such that $\ln\mathcal{N}(\epsilon) \leq C_s(1/\epsilon)^s, \forall\epsilon \geq 0$.*

Assumption 1 *For $\{\mathcal{H}_p\}_{p=1}^n$, $\mathcal{H}_p$ is separable with respect to the norm RKHS, and we set $\kappa = \sup_{x\in X}\sqrt{K_p(x,x)} \leq 1$. In this paper, we assume that there exists a constant $M \geq 0$, we have $f_B \leq M$ [12], and $C = \max\{C_1, C_2\} = 1$.*

Our main results on the generalization of MK-MSVCR-TSL algorithm are stated as follows.

Theorem 1. *Assume* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *is an u.e.M.c. sample. For any* $0 < \delta < 1$*, with probability at least* $1 - \delta$*, inequality*

$$\begin{aligned}\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq & 3D(\lambda) + \frac{224(\sqrt{D(\lambda)/\lambda} + R)^2 \|\Gamma_0\|^2 \ln(\frac{2}{\delta})}{m} \\ & + 2\Big(\frac{448\|\Gamma_0\|^2 C_s R^{s+2}}{m}\Big)^{\frac{1}{s+1}}\end{aligned}$$

is valid provided that $m \geq 448R\|\Gamma_0\|^2 \ln(2/\delta)(\ln(2/\delta)/C_s)^{1/s}$.

Corollary 1. *Let* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *be an u.e.M.c. sample. Taking* $\lambda = (\frac{1}{m})^{\frac{1}{1+s}}$*, we have that for any* $0 < \delta < 1$*, the following inequality is valid with probability at least* $1 - \delta$*,*

$$\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq \overline{C}(\frac{1}{m})^{\frac{1}{1+s}},$$

where $\overline{C} = 896\|\Gamma_0\|^2 R^2(4C_s{}^{\frac{1}{s+1}} + 4\ln(2/\delta) + 3)$.

Theorem 2. *Assume* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *is an i.i.d. sample. For any* $0 < \delta < 1$*, with probability at least* $1 - \delta$*, the inequality*

$$\begin{aligned}\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq & D(\lambda)\Big(2 + \frac{14\ln(\frac{2}{\delta})}{m\lambda}\Big) + \frac{14R^2\ln(\frac{2}{\delta})}{m} \\ & + \frac{\ln(\frac{2}{\delta})}{m} + 2\Big(\frac{300R^2C_s(4R)^s}{m}\Big)^{\frac{1}{s+1}}\end{aligned}$$

is valid provided that $m \geq 74R\ln(2/\delta)(\ln(2/\delta)/C_s)^{1/s}$.

Corollary 2. *Assume* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *is an i.i.d. sample. Let* $\lambda = (\frac{1}{m})^{\frac{1}{1+s}}$*. For any* $0 < \delta < 1$*, we have that the inequality*

$$\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B \leq \widehat{C}(\frac{1}{m})^{\frac{1}{1+s}}$$

is valid with probability at least $1-\delta$*, where* $\widehat{C} = 600R^2(4C_s{}^{\frac{1}{s+1}} + \ln(2/\delta))$ *is a constant.*

Theorems 1-2 and Corollaries 1-2 will be proved in Appendix B. Besides, in order to have better showing these theoretical results above, we give the following remarks.

Remark 2. By Corollaries 1-2, we have that $\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \to 0$*, as* $m \to \infty$*. This means that the MK-MSVCR algorithm based u.e.M.c. (or i.i.d.) samples is consistent. To our knowledge, all these results above are the first works on this topic.*

5. Numerical Studies

In this section, We compare Algorithm 1 with the support vector classification regression machine for multi-class classification (MSVCR) [1], the MSVCR algorithm with multiple kernel learning (MK-MSVCR), the mean weighted multiple kernel SVM [13] with OVO method (MKSVM-OVO), the mean weighted multiple kernel SVM [13] with OVA method (MKSVM-OVA). MKSVM-OVO is a algorithm that the mean weighted multiple kernel support vector machine is combined with one-versus-one strategy to handle multi-class classification, and MKSVM-OVA is a algorithm that the mean weighted multiple kernel support vector machine is combined with one-versus-rest strategy to handle multi-class classification.

5.1. Datasets and experimental setup

We use 9 public datasets from UCI[5] datasets: Connect4, Postures, Swarm, Twitter, Pavia, TV_News, Mnist, Proyecto, Kegg. For each dataset, we first divide randomly each data sets into the training set D_{train} and the test set D_{test}. The information of these datasets is summarized in Table 1. All experiments were run on Intel 2.80GHz E5-1603 v4 CPU with MATLAB 2018a.

Table 1. 9 Public Datasets

Dataset	$\sharp D_{train}$	$\sharp D_{test}$	$\sharp Input\ Dimension$	$\sharp Class$
Connect4	50668	16889	126	3
Postures	58571	19524	36	5
Swarm	54036	18012	200	6
Twitter	169850	56616	77	9
Pavia	120000	28152	102	9
TV_News	97263	32422	132	10
Mnist	45000	15000	780	10
Proyecto	47010	15670	7	21
Kegg	49152	16384	25	25

All the experimental results are based on 50 times repeated experiments and the following 8 kernels: a linear kernel $K(a,b) = a'b$, three polynomial kernels $K(a,b) = (1 + a'b)^d$ with $d = \{2,3,4\}$ and four RBF kernels $K(a,b) = \exp(-\|a-b\|^2/2\sigma)$, where σ is chosen from the set $\{0.001, 0.01, 1, 10\}$. The other parameters of algorithms are also obtained through 10-fold cross-validation from $[10^{-3}, 10^{-2}, \ldots, 10^3]$. For experimental results of classical MSVCR algorithm with single kernel, we choose the best results between among 8 kernels.

Now, we state our experimental process as follows: (i) Train $\mathbf{z}$ and obtain a classifier $\phi_{\mathbf{z}}$ by Algorithm 1. (ii) Test the classifier on the given testing set and calculate the corresponding misclassification rates. (iii) Do process (i)-(ii) above under the same samples $\mathbf{z}$ by the above 4 multi-class classification algorithms, respectively. (iv) Repeat process

[5] http://archive.ics.uci.edu/ml/datasets.html

(ii)-(iii) above for 50 times and calculate the average misclassification rates of 5 algorithms, respectively. In this paper, we use "MSVCR", "MK-MSVCR" , "MK-OVO" and "MK-OVA" to refer the experimental results of MSVCR, MK-MSVCR, MKSVM-OVO and MKSVM-OVA algorithms based on randomly independent samples, respectively.

5.2. Comparison of misclassification rates

We show the average misclassification rates of 5 algorithms in Tables 2-3.

Table 2. Average misclassification rates ($\%$) with $m = 9000$

Dataset	MK-MSVCR-TSL	MK-MSVCR	MSVCR	MK-OVO	MK-OVA
Connect4	30.04±0.39	30.90±0.42	33.59±0.40	32.39±0.42	33.59±0.48
Postures	21.59±0.32	23.49±0.35	28.02±0.36	24.07±0.39	26.49±0.39
Swarm	35.13±0.36	36.93±0.38	39.15±0.41	38.45±0.47	39.48±0.49
Twitter	11.73±0.29	15.17±0.31	20.21±0.35	17.18±0.37	16.50±0.40
Pavia	20.97±1.09	22.78±1.61	29.18±1.61	27.36±1.01	26.52±1.40
TV_News	30.43±0.31	33.96±0.30	35.23±0.35	33.42±0.31	33.50±0.31
Mnist	11.02±0.32	13.96±0.41	16.66±0.35	16.06±0.51	15.54±0.42
Proyecto	33.44±0.31	34.93±0.35	36.13±0.29	35.43±0.39	35.45±0.45
Kegg	33.95±0.34	35.11±0.30	38.11±0.38	36.30±0.53	36.89±0.47

Table 3. Average misclassification rates ($\%$) with $m = 25000$

Dataset	MK-MSVCR-TSL	MK-MSVCR	MSVCR	MK-OVO	MK-OVA
Connect4	28.01±0.33	29.91±0.35	31.62±0.35	31.49±0.50	31.85±0.45
Postures	18.90±0.34	25.27±0.33	28.87±0.39	27.51±0.41	29.40±0.40
Swarm	33.11±0.32	35.92±0.34	38.16±0.41	36.51±0.51	36.53±0.34
Twitter	10.01±0.25	16.37±0.31	21.21±0.36	20.40±0.33	17.94±0.40
Pavia	19.90±1.01	26.15±1.55	30.01±1.44	32.58±0.84	32.61±1.50
TV_News	28.43±0.32	31.92±0.35	34.25±0.34	31.45±0.39	32.49±0.32
Mnist	10.98±0.34	13.86±0.37	16.06±0.39	14.58±0.34	15.47±0.36
Proyecto	31.50±0.34	34.92±0.35	35.15±0.30	34.38±0.43	34.36±0.45
Kegg	31.44±0.29	33.90±0.35	35.24±0.36	34.44±0.42	34.53±0.43

By Tables 2-3, we can find that for $m = 9000$ (or $m = 25000$), the means of misclassification rates of the proposed MK-MSVCR-TSL algorithm are less than that of other multi-class classification algorithms, and the standard deviations of misclassification rates for the proposed MK-MSVCR-TSL algorithm are also less than that of other multi-class classification algorithms. In detail, according to the experimental results of MSVCR and MK-MSVCR algorithms with randomly independent samples, we can find that the means of misclassification rates of the proposed MK-MSVCR algorithm are less than that of

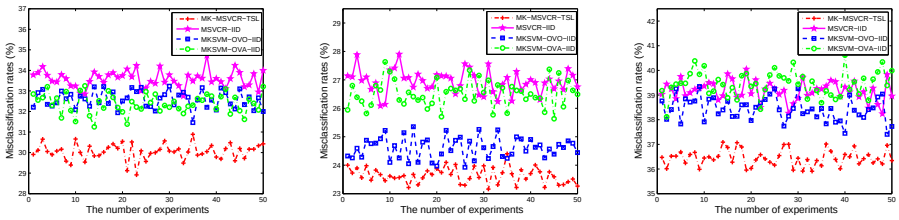
MSVCR algorithms, it is imply that for more complex and larger multi-classification data, multiple kernel learning can effectively improve the classification accuracy.

Table 4. Wilcoxon tests of average misclassification rates for 5 algorithms

Comparison	R^+	R^-	Hypothesis($\alpha = 0.05$)	Selected
MK-MSVCR-TSL vs. MK-MSVCR	0	45	Rejected	MK-MSVCR-TSL
MK-MSVCR-TSL vs. MSVCR	0	45	Rejected	MK-MSVCR-TSL
MK-MSVCR-TSL vs. MK-OVO	0	45	Rejected	MK-MSVCR-TSL
MK-MSVCR-TSL vs. MK-OVA	0	45	Rejected	MK-MSVCR-TSL

In Table 4, we apply the Wilcoxon signed-rank test ($\alpha = 0.05$)[**?**] to verify whether there exist statistical significance between the proposed MK-MSVCR-TSL algorithm and other 4 algorithms by using the mean of misclassification rates presented in Table 3. By Table 4, we can find that the proposed MK-MSVCR-TSL has better performance compared to other 4 multi-classification algorithms.

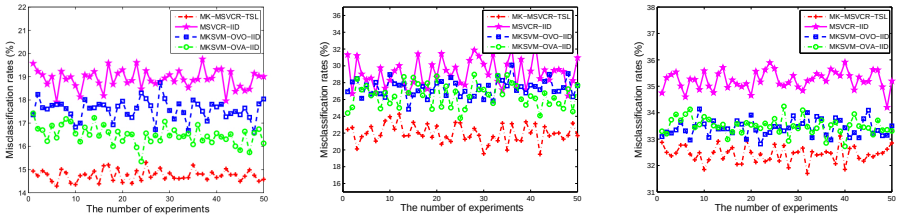
In order to have a better showing the learning performance of the proposed MK-MSVCR-TSL algorithm more clearer, we present Figures 1-6 to compare 50 times misclassification rates of MK-MSVCR-TSL algorithm with that of other algorithms. Here, we use "red cross", "blue square", "green circle" and "magenta asterisk" to denote the misclassification rates of MSVCR, MK-MSVCR, MKSVM-OVO and MKSVM-OVA algorithms based on randomly independent samples, respectively. Here m is the size of training sample, and the number of repeat experiments and the misclassification rates are represented on the horizontal axis and the vertical axis, respectively.



(a) (b) (c)

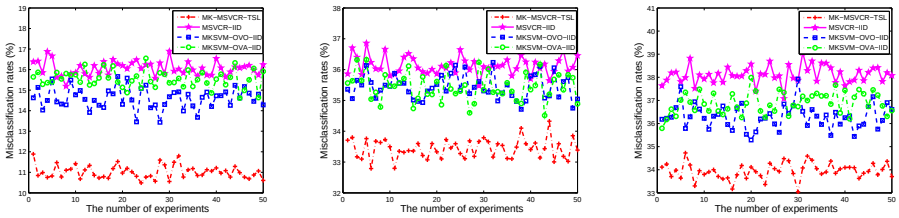
Fig. 1. 50 times experimental misclassification rates for $m = 10000$: (a) Connect4; (b) Postures; (c) Swarm

By Figures 1-6, we can find that for the same size ($m = 10000$ or $m = 20000$) of training sample, all the 50 times misclassification rates of MK-MSVCR-TSL are generally smaller than that of other multi-class classification algorithms. This means that in terms of classification accuracy, our algorithm has obvious advantages. And we can find that as the sample size increases, the advantages of our proposed algorithm are more obvious.



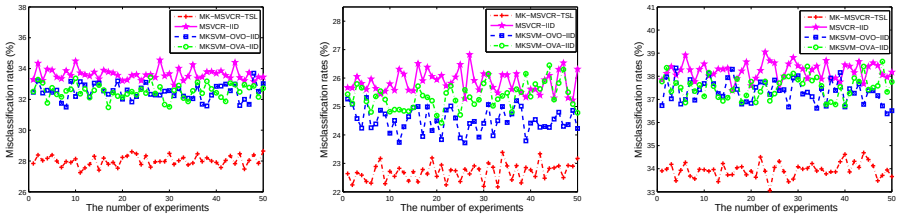
(a) (b) (c)

Fig. 2. 50 times experimental misclassification rates for $m = 10000$: (a) Twitter; (b) Pavia; (c) TV_News.



(a) (b) (c)

Fig. 3. 50 times experimental misclassification rates for $m = 10000$: (a) Mnist; (b) Proyecto; (c) Kegg



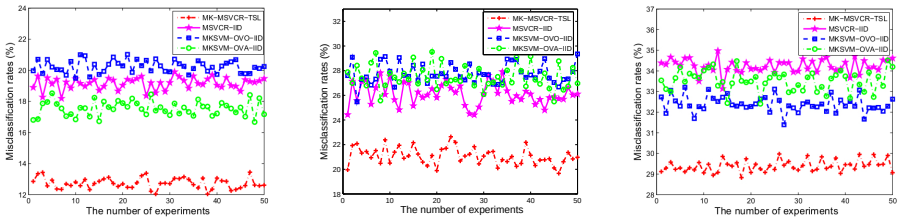
(a) (b) (c)

Fig. 4. 50 times experimental misclassification rates for $m = 20000$: (a) Connect4; (b) Postures; (c) Swarm

5.3. Comparison of total time

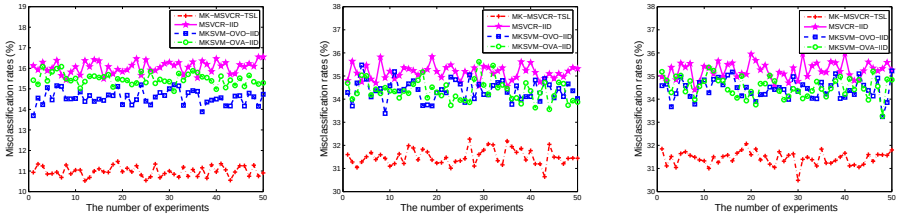
We show the sampling and training total time of 5 algorithms in Tables 5-6.

By Tables 5-6, we can find that for $m = 9000$ (or $m = 25000$), the sampling and training total time of the proposed MK-MSVCR-TSL is shorter than that of other 4 multi-classification algorithms. Combined with Tables 2-3, for the experimental results of MK-MSVCR-TSL and MK-MSVCR algorithms, we can find that the sampling and training total time of the proposed MK-MSVCR-TSL algorithm is shorter than that of



(a) (b) (c)

Fig. 5. 50 times experimental misclassification rates for $m = 20000$: (a) Twitter; (b) Pavia; (c) TV_News



(a) (b) (c)

Fig. 6. 50 times experimental misclassification rates for $m = 20000$: (a) Mnist; (b) Proyecto; (c) Kegg

Table 5. Sampling and training total time (s) for $m = 9000$

Dataset	MK-MSVCR-TSL	MK-MSVCR	MSVCR	MK-OVO	MK-OVA
Connect4	3834.00	8270.45	5264.87	9930.60	7305.26
Postures	8053.02	23431.51	8914.51	18711.38	15302.13
Swarm	3809.33	9610.83	4088.25	10514.35	8484.36
Twitter	1543.39	5027.24	2532.39	4739.51	4243.14
Pavia	3163.18	9340.25	5231.44	8631.60	8369.17
TV_News	2038.28	5349.02	2454.38	4936.21	3696.07
Mnist	913.72	5381.88	1865.35	5691.39	8170.49
Proyecto	2935.07	8138.91	3519.49	9287.14	8036.25
Kegg	3836.14	13796.48	5103.23	12364.96	13505.16
Sum of Time	30126.14	88346.59	38973.91	84807.14	77112.03

MK-MSVCR algorithms, it is imply that for more complex and larger multi-classification data, our algorithm has obvious advantages in terms of sampling and training total time while ensuring certain classification accuracy.

In Table 7, we apply the Wilcoxon signed-rank test ($\alpha = 0.05$)[**?**] to verify whether there exist statistical significance between the proposed MK-MSVCR-TSL algorithm and other 4 algorithms by using the sampling and training total time presented in Table **??**.

Table 6. Sampling and training total time (s) for $m = 25000$

Dataset	MK-MSVCR-TSL	MK-MSVCR	MSVCR	MK-OVO	MK-OVA
Connect4	5009.91	10303.13	7041.91	12343.38	11069.20
Postures	11819.79	43427.13	13434.19	40741.38	36957.31
Swarm	9153.73	23429.15	13007.13	31245.58	29894.29
Twitter	3968.61	15353.13	7872.34	10013.18	8943.33
Pavia	8468.84	20311.17	12935.34	38031.12	35085.71
TV_News	2351.95	7360.40	5340.13	7671.83	7052.36
Mnist	2368.15	8689.35	3399.91	9400.62	10261.12
Proyecto	6054.01	12437.59	8695.15	14343.83	12118.36
Kegg	9846.13	30971.74	19678.04	35083.51	29836.58
Sum of Time	59041.13	172282.79	91404.15	198874.44	181218.27

By Table 7, we can find that the proposed MK-MSVCR-TSL has better performance compared to other 4 multi-classification algorithms.

Table 7. Wilcoxon tests of sampling and training total time for 5 algorithms

Comparison	R^+	R^-	Hypothesis($\alpha = 0.05$)	Selected
MK-MSVCR-TSL vs. MK-MSVCR	0	55	Rejected	MK-MSVCR-TSL
MK-MSVCR-TSL vs. MSVCR	0	55	Rejected	MK-MSVCR-TSL
MK-MSVCR-TSL vs. MK-OVO	0	55	Rejected	MK-MSVCR-TSL
MK-MSVCR-TSL vs. MK-OVA	0	55	Rejected	MK-MSVCR-TSL

6. Conclusions

In this paper we firstly considered the generalization bounds of MSVCR algorithm with multiple kernels based on u.e.M.c. samples. As its application, we also established the generalization bounds of MK-MSVCR algorithm for the case of i.i.d. samples and obtained the fast learning rate of MK-MSVCR algorithm for u.e.M.c. and i.i.d. samples, respectively. In addition, we also introduced a new algorithm, the MK-MSVCR-TSL, and showed that the experimental results of the proposed algorithm for public datasets. The experimental results shown that the means of misclassification rates of the MK-MSVCR-TSL and MK-MSVCR are smaller than the classical MSVCR introduced in [1], which implies that the proposed multiple kernel learning can obviously improve the learning performance of the classical MSVCR for large sample size. The experimental results also shown that not only the means of misclassification rates of the MK-MSVCR-TSL are smaller than other multi-class classification algorithms, but also the sampling and training total time of the MK-MSVCR is less than that of other multi-class classification algorithms, which implies that the proposed MK-MSVCR-TSL have obvious competitive strength for learning performance. In other words, the two-stage learning from the given

datasets is a new strategy of improving the learning performance of the classical MSVCR algorithm. Moreover, the experiments display that the proposed algorithm is valid and competitive compared to other multiple kernels multi-class classification methods.

Based on the existing work, there are still several open issues worth further research. For example, applying our method to deep neural networks, using the idea of distributed or parallel to accelerate our method. These problems mentioned above are under our current investigation.

Appendix A

In this section, we give the proof of the main results.

Proposition 1. *Let* $f_\lambda = \arg\min_{f\in\bar{\mathcal{H}}_K}\{\lambda\|f\|_K^2 + \mathcal{E}(f)\}$, $D(\lambda) = \mathcal{E}(f_\lambda) - \mathcal{E}(f_B) + \lambda\|f_\lambda\|_K^2$. *For any* $\mathbf{z}\in Z^m$, $\lambda\|f_{\mathbf{z}}\|_K^2 \geq 0$, *we have that inequality*

$$\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq \{T_1 + T_2\} + D(\lambda),$$

is valid, where

$$\begin{aligned} T_1 :=& \mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) - \mathcal{E}(f_B) + \mathcal{E}_{\mathbf{z}}(f_B),\\ T_2 :=& \mathcal{E}_{\mathbf{z}}(f_\lambda) - \mathcal{E}(f_\lambda) - \mathcal{E}_{\mathbf{z}}(f_B) + \mathcal{E}(f_B). \end{aligned}$$

Proof: Since for any $\mathbf{z}\in Z^m$, $\lambda\|f_{\mathbf{z}}\|_K^2 \geq 0$, we have the following error decomposition inspired by idea from [30],

$$\begin{aligned} \mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq& \mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) + \lambda\|f_{\mathbf{z}}\|_K^2\\ =& \{\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) + \mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) - \mathcal{E}(f_B) + \mathcal{E}_{\mathbf{z}}(f_B) - \mathcal{E}_{\mathbf{z}}(f_B)\\ &+ \mathcal{E}_{\mathbf{z}}(f_\lambda) - \mathcal{E}_{\mathbf{z}}(f_\lambda) - \mathcal{E}(f_\lambda) + \mathcal{E}(f_\lambda) + \mathcal{E}(f_B) - \mathcal{E}(f_B)\}\\ &+ \{\lambda\|f_{\mathbf{z}}\|_K^2 - \lambda\|f_\lambda\|_K^2 + \lambda\|f_\lambda\|_K^2\}\\ =& \{\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) - \mathcal{E}(f_B) + \mathcal{E}_{\mathbf{z}}(f_B)\}\\ &+ \{\mathcal{E}_{\mathbf{z}}(f_\lambda) - \mathcal{E}(f_\lambda) - \mathcal{E}_{\mathbf{z}}(f_B) + \mathcal{E}(f_B)\}\\ &+ \{\mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) - \mathcal{E}_{\mathbf{z}}(f_\lambda) + \lambda\|f_{\mathbf{z}}\|_K^2 - \lambda\|f_\lambda\|_K^2\}\\ &+ \{\mathcal{E}(f_\lambda) - \mathcal{E}(f_B) + \lambda\|f_\lambda\|_K^2\}\\ =& T_1 + T_2 + T_3 + D(\lambda) \leq T_1 + T_2 + D(\lambda) \end{aligned}$$

The last inequality above is follows from the fact that $T_3 \leq 0$ since by the definiton $f_{\mathbf{z}}$, we have $\mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) + \lambda||f_{\mathbf{z}}||_K^2 \leq \mathcal{E}_{\mathbf{z}}(f_\lambda) + \lambda||f_\lambda||_K^2$. $D(\lambda)$ is called the regularizing error, which is independent of the sample $\mathbf{z}$, but is dependent of the space $\mathcal{H}_K$. Then we complete the proof of Proposition 1.

To prove our results presented in Section 3, our main tools are as follows.

Lemma 1. *[29] Let* ξ *be a random variable on a probability space* Z *with mean* $E(\xi)$, *variance* $\sigma^2(\xi) = \sigma^2$, *and satisfying* $|\xi(z) - E(\xi)| \leq M_\xi$ *for almost all* $z\in Z$. *Then for all* $\varepsilon > 0$,

$$\mathrm{P}\Big\{\frac{1}{m}\sum_{i=1}^{m}\xi(z_i) - E(\xi) \geq \varepsilon\Big\} \leq \exp\Big\{-\frac{m\varepsilon^2}{2(\sigma^2 + \frac{1}{3}M_\xi\varepsilon)}\Big\}.$$

Lemma 2. *[29] Let $\mathcal{G}$ be a set of functions on Z such that for some $c_\rho \geq 0$, $|g - E(g)| \leq B$ almost everywhere and $E(g^2) \leq c_\rho E(g)$ for each $g \in \mathcal{G}$. Then for every $\varepsilon > 0$ and $0 < \alpha \leq 1$,*

$$\mathrm{P}\Big\{ \sup_{g\in\mathcal{G}} \frac{E(g) - \frac{1}{m}\sum_{i=1}^{m} g(z_i)}{\sqrt{E(g)+\varepsilon}} \geq 4\alpha\sqrt{\varepsilon} \Big\} \leq \mathcal{N}(\mathcal{G}, \alpha\varepsilon) \exp\Big\{ -\frac{\alpha^2 m\varepsilon}{2c_\rho + \frac{2}{3}B} \Big\}.$$

Lemma 3. *[31] Let $\mathcal{G}$ be a countable class of bounded measurable functions, and Z be a u.e.M.c. sample set. Assume that $0 \leq g(z) \leq C_{\mathcal{G}}$ for any $g \in \mathcal{G}$ and for any $z \in Z$. Then for any $\varepsilon > 0$, we have*

$$\mathrm{P}\Big\{ \frac{\frac{1}{m}\sum_{i=1}^{m} g(z_i) - E(g)}{\sqrt{E(g)+\varepsilon}} \geq \sqrt{\varepsilon} \Big\} \leq \exp\Big\{ \frac{-m\varepsilon}{56 C_{\mathcal{G}} \|\Gamma_0\|^2} \Big\},$$

where $||\Gamma_0|| = \sqrt{2}/(1-\beta_1^{1/2n_1})$, and β_1, n_1 are two positive constants independent of m.

Lemma 4. *With all notations as that in Lemma 3, then for $\forall \varepsilon > 0$, we have*

$$\mathrm{P}\Big\{ \sup_{g\in\mathcal{G}} \frac{\frac{1}{m}\sum_{i=1}^{m} g(z_i) - E(g)}{\sqrt{E(g)+\varepsilon}} \geq 4\sqrt{\varepsilon} \Big\} \leq \exp \mathcal{N}(\mathcal{G}, \varepsilon) \Big\{ \frac{-m\varepsilon}{56 C_{\mathcal{G}} \|\Gamma_0\|^2} \Big\}.$$

Lemma 5. *[6] Let $c_1, c_2 > 0$, and $p_1 > p_2 > 0$. Then, the equation $x^{p_1} - c_1 x^{p_2} - c_2 = 0$ has a unique positive zero x^*. In addition, we have $x^* \leq \max\{(2c_1)^{1/(p_1-p_2)}, (2c_2)^{1/p_1}\}$.*

Proposition 2. *Assume $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ is an u.e.M.c. sample. For any $0 < \delta < 1$, the following inequality is valid with confidence at least $1 - \delta/2$,*

$$T_1 \leq \frac{1}{2}[\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B)] + \varepsilon^*(m, \delta/2),$$

where $\varepsilon^*(m, \frac{\delta}{2}) = \max\Big\{ \frac{448\|\Gamma_0\|^2 R^2 \ln(\frac{2}{\delta})}{m}, (\frac{448\|\Gamma_0\|^2 C_s R^{s+2}}{m})^{\frac{1}{s+1}} \Big\}$.

Proof: Set $\xi_1 = V(y, f) - V(y, f_B)$. It is clear that ξ_1 varies among a set of functions in accordance with the varying sample $\mathbf{z}$. Let $\mathcal{G}_R = \big\{ g | g(\mathbf{z}) := V(y, f) - V(y, f_B), f \in \mathcal{B}_R \big\}$. We have

$$\begin{aligned} E(g) =& \mathcal{E}(f) - \mathcal{E}(f_B) \geq 0, \ \ \frac{1}{m}\sum_{i=1}^{m} g(z_i) = \mathcal{E}_{\mathbf{z}}(f) - \mathcal{E}_{\mathbf{z}}(f_B), \\ g(z) =& C_1[(1 - yf(x))_+ - (1 - yf_B(x))_+] \cdot \mathbf{1}_{\{y\neq 0\}} \\ & + C_2[(f(x) - f_B(x))(f(x) + f_B(x))] \cdot \mathbf{1}_{\{y=0\}}. \end{aligned}$$

Since $\|f\|_\infty \leq \|f\|_K \leq R$ and $|f_B(x)| \leq M$ almost everywhere, by the restriction $M \leq R$ and $C = \max\{C_1, C_2\} = 1$, we have

$$|g(z)| \leq C_1(R + M) + C_2(R + M)(R + M) \leq 4R^2.$$

It follows that $|g(z) - E(g)| \leq 8R^2$ almost everywhere, and

$$\begin{aligned} g^2 =&[C_1((1-yf(x))_+ - (1-yf_B(x))_+) \cdot \mathbf{1}_{\{y\neq 0\}} \\ &+ C_2(f^2(x) - f_B^2(x)) \cdot \mathbf{1}_{\{y=0\}}]^2 \\ =&C_1^2[(1-yf(x))_+ - (1-yf_B(x))_+]^2 \cdot \mathbf{1}_{\{y\neq 0\}} \\ &+ C_2^2[f^2(x) - f_B^2(x)]^2 \cdot \mathbf{1}_{\{y=0\}} \\ \leq&CC_1[(1-yf(x))_+ - (1-yf_B(x))_+](R+M) \cdot \mathbf{1}_{\{y\neq 0\}} \\ &+ CC_2[f^2(x) - f_B^2(x)](R^2+M^2) \cdot \mathbf{1}_{\{y=0\}} \\ \leq&2R^2 \cdot [C_1((1-yf(x))_+ - (1-yf_B(x))_+) \cdot \mathbf{1}_{\{y\neq 0\}} \\ &+ C_2(f^2(x) - f_B^2(x)) \cdot \mathbf{1}_{\{y=0\}}]. \end{aligned}$$

Thus $E(g^2) \leq 2R^2 E(g)$, and

$$\sup_{f\in\mathcal{B}_R} \frac{\mathcal{E}(f) - \mathcal{E}(f_B) - (\mathcal{E}_{\mathbf{z}}(f) - \mathcal{E}_{\mathbf{z}}(f_B))}{\sqrt{\mathcal{E}(f) - \mathcal{E}(f_B) + \varepsilon}} = \sup_{g\in\mathcal{G}_R} \frac{E(g) - \frac{1}{m}\sum_{i=1}^m g(z_i)}{\sqrt{E(g)+\varepsilon}}.$$

Applying Lemma 4 to the function set $\mathcal{G}_R$, we have that inequality

$$\sup_{f\in\mathcal{B}_R} \frac{\mathcal{E}(f) - \mathcal{E}(f_B) - (\mathcal{E}_{\mathbf{z}}(f) - \mathcal{E}_{\mathbf{z}}(f_B))}{\sqrt{\mathcal{E}(f) - \mathcal{E}(f_B) + \varepsilon}} = \sup_{g\in\mathcal{G}_R} \frac{E(g) - \frac{1}{m}\sum_{i=1}^m g(z_i)}{\sqrt{E(g)+\varepsilon}} \leq \sqrt{\varepsilon}$$

holds with probability at least $1 - \mathcal{N}(\mathcal{G}_R, \varepsilon)\exp\big\{-\frac{m\varepsilon}{56\cdot 4R^2\cdot\|\Gamma_0\|^2}\big\}$.

By Definition 4, we have

$$\begin{aligned} &\mathrm{P}\Big\{\sup_{f\in\mathcal{B}_R} \frac{\mathcal{E}(f) - \mathcal{E}(f_B) - (\mathcal{E}_{\mathbf{z}}(f) - \mathcal{E}_{\mathbf{z}}(f_B))}{\sqrt{\mathcal{E}(f) - \mathcal{E}(f_B) + \varepsilon}} \geq \sqrt{\varepsilon}\Big\} \\ \leq&\mathcal{N}(\mathcal{G}_R, \varepsilon)\exp\Big\{-\frac{m\varepsilon}{224R^2\|\Gamma_0\|^2}\Big\} \\ \leq&\exp\Big\{C_s(\frac{R}{\varepsilon})^s - \frac{m\varepsilon}{224R^2\|\Gamma_0\|^2}\Big\}. \end{aligned}$$

Let $\delta = \exp\big\{C_s(\frac{R}{\varepsilon})^s - \frac{m\varepsilon}{224R^2\|\Gamma_0\|^2}\big\}$. Solving this equation with respect to ε, by Lemma 5, we have

$$\varepsilon = \varepsilon^*(m,\delta) = \max\Big\{\frac{448R^2\|\Gamma_0\|^2\ln(\frac{1}{\delta})}{m}, (\frac{448\|\Gamma_0\|^2 C_s R^{s+2}}{m})^{\frac{1}{s+1}}\Big\}.$$

It follows that the following inequality holds with the probability at least $1-\delta$

$$\mathcal{E}(f) - \mathcal{E}(f_B) - (\mathcal{E}_{\mathbf{z}}(f) - \mathcal{E}_{\mathbf{z}}(f_B)) \leq \frac{1}{2}[\mathcal{E}(f) - \mathcal{E}(f_B)] + \varepsilon^*(m,\delta).$$

Replacing f by $f_{\mathbf{z}}$, we have that the following inequality

$$T_1 = \mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) - (\mathcal{E}_{\mathbf{z}}(f_{\mathbf{z}}) - \mathcal{E}_{\mathbf{z}}(f_B)) \leq \frac{1}{2}[\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B)] + \varepsilon^*(m,\delta/2)$$

is valid with probability at least $1 - \frac{\delta}{2}$. Therefore, we complete the proof of Proposition 2.

By making use of the similar proof method as that in Proposition 2, and Lemma 2, we have

Proposition 3. *Assume* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *is an i.i.d. sample. For any* $0 < \delta < 1$*, we have that the following inequality holds with confidence at least* $1 - \delta/2$*,*

$$T_1 \leq \frac{1}{2}[\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B)] + \bar{\varepsilon}(m, \delta/2),$$

where $\bar{\varepsilon}(m, 2/\delta) = \max\left\{\frac{300R^2\ln(2/\delta)}{m}, (\frac{300R^2C_s(4R)^s}{m})^{\frac{1}{s+1}}\right\}$.

Proposition 4. *Assume* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *is a u.e.M.c. sample. For any* $0 < \delta < 1$*, we have that the following inequality holds with the probability at least* $1 - \delta/2$*,*

$$T_2 \leq \frac{1}{2}D(\lambda) + \frac{112(\sqrt{D(\lambda)/\lambda} + R)^2 \cdot \|\Gamma_0\|^2 \ln(\frac{2}{\delta})}{m}.$$

Proof: By the definitions of f_λ and $D(\lambda)$, we have $\lambda\|f_\lambda\|_K^2 \leq \mathcal{E}(f_\lambda) - \mathcal{E}(f_B) + \lambda\|f_\lambda\|_K^2 = D(\lambda)$. It follows that $|f_\lambda\|_\infty \leq \|f_\lambda\|_K \leq \sqrt{D(\lambda)/\lambda}$. Set $\xi_2 = V(y, f_\lambda) - V(y, f_B)$, we have

$$\begin{aligned}
\xi_2 =& C_1(1 - yf_\lambda(x))_+ \cdot \mathbf{1}_{\{y\neq 0\}} + C_2(y - f_\lambda(x))^2 \cdot \mathbf{1}_{\{y=0\}} \\
&- C_1(1 - yf_B(x))_+ \cdot \mathbf{1}_{\{y\neq 0\}} - C_2(y - f_B(x))^2 \cdot \mathbf{1}_{\{y=0\}} \\
=& C_1[(1 - yf_\lambda(x))_+ - (1 - yf_B(x))_+] \cdot \mathbf{1}_{\{y\neq 0\}} + C_2[f_\lambda^2(x) - f_B^2(x)] \cdot \mathbf{1}_{\{y=0\}},
\end{aligned}$$

then $T_2 = \frac{1}{m}\sum_{i=1}^m \xi_2(z_i) - E(\xi_2)$. Since $|f_B| \leq M \leq R$ almost everywhere, we have

$$\begin{aligned}
|\xi_2| \leq& |C_1[(1 - yf_\lambda(x))_+ - (1 - yf_B(x))_+] \cdot \mathbf{1}_{\{y\neq 0\}} \\
&+ C_2[(f_\lambda(x) - f_B(x))(f_\lambda(x) + f_B(x))] \cdot \mathbf{1}_{\{y=0\}}| \\
\leq& C(\sqrt{D(\lambda)/\lambda} + R) + C(\sqrt{D(\lambda)/\lambda} + R)(\sqrt{D(\lambda)/\lambda} + R) \\
\leq& 2(\sqrt{D(\lambda)/\lambda} + R)^2 := 2b.
\end{aligned}$$

Hence $|\xi_2 - E(\xi_2)| \leq M_{\xi_2} := 4b$, $|\xi_2| \leq 2(\sqrt{D(\lambda)/\lambda} + R)^2 := 2b$. Moreover, we have

$$\begin{aligned}
E(\xi_2^2) =& E[C_1((1 - yf_\lambda(x))_+ - (1 - yf_B(x))_+) \cdot \mathbf{1}_{\{y\neq 0\}} \\
&+ C_2(f_\lambda(x)^2 - f_B(x)^2) \cdot \mathbf{1}_{\{y=0\}}]^2 \\
=& E\{C_1[(1 - yf_\lambda(x))_+ - (1 - yf_B(x))_+] \cdot \mathbf{1}_{\{y\neq 0\}}\}^2 \\
&+ E\{C_2[(f_\lambda(x) - f_B(x))(f_\lambda(x) + f_B(x))] \cdot \mathbf{1}_{\{y=0\}}\}^2 \\
\leq& C^2\|f_\lambda(x) - f_B(x)\|_\rho^2 + C^2\|f_\lambda(x) - f_B(x)\|_\rho^2(\sqrt{D(\lambda)/\lambda} + R)^2 \\
\leq& C^2(\|f_\lambda(x) - f_B(x)\|_\rho^2 + \lambda\|f_\lambda\|_K^2) + C^2(\|f_\lambda(x) - f_B(x)\|_\rho^2 \\
&+ \lambda\|f_\lambda\|_K^2)(\sqrt{D(\lambda)/\lambda} + R)^2 \\
\leq& C^2D(\lambda) + C^2D(\lambda)(\sqrt{D(\lambda)/\lambda} + R)^2 \\
=& D(\lambda)(1 + (\sqrt{D(\lambda)/\lambda} + R)^2) = D(\lambda)(1 + b).
\end{aligned}$$

Applying Lemma 3, we have that for any $\varepsilon > 0$,

$$\mathrm{P}\left\{\frac{\frac{1}{m}\sum_{i=1}^m \xi_2(z_i) - E(\xi_2)}{\sqrt{E(\xi_2) + \varepsilon}} \geq \sqrt{\varepsilon}\right\} \leq \exp\left\{\frac{-m\varepsilon}{56 \cdot \|\Gamma_0\|^2 \cdot 2b}\right\}.$$

It follows that for any $0 < \delta < 1$, with probability at least $1 - \delta$, inequality

$$\frac{1}{m}\sum_{i=1}^{m}\xi_2(z_i) - E(\xi_2) \leq \frac{1}{2}D(\lambda) + \frac{112(\sqrt{D(\lambda)/\lambda} + R)^2 \cdot \|\Gamma_0\|^2 \ln(\frac{2}{\delta})}{m}$$

is valid. Then we accomplish the proof of Proposition 4.

Similar to the proof of Proposition 4, we obtain the following bound of T_2 for i.i.d. samples.

Proposition 5. *Assume* $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ *is an i.i.d. sample. For any* $0 < \delta < 1$*, the following inequality holds with the probability at least* $1 - \delta/2$*,*

$$T_2 \leq D(\lambda)\Big(1 + \frac{7\ln(\frac{2}{\delta})}{m\lambda}\Big) + \frac{7R^2\ln(\frac{2}{\delta})}{m} + \frac{\frac{1}{2}\ln(\frac{2}{\delta})}{m}.$$

Proof: Applying Lemma 1, by Proposition 4, we have that for any $t > 0$, $\frac{1}{m}\sum_{i=1}^{m}\xi_2(z_i) - E(\xi_2) \leq t$, with confidence at least

$$\begin{aligned}1 - \exp\Big\{-\frac{mt^2}{2(\sigma^2(\xi_2) + \frac{1}{3}M_{\xi_2}t)}\Big\} \geq & 1 - \exp\Big\{-\frac{mt^2}{2[D(\lambda)(1+b) + \frac{1}{3}\cdot 4bt]}\Big\} \\ = & 1 - \exp\Big\{-\frac{mt^2}{2D(\lambda)(1+b) + \frac{8}{3}bt}\Big\}.\end{aligned}$$

Select t^* as the only positive solution of the equation

$$-\frac{mt^2}{2D(\lambda)(1+b) + \frac{8}{3}bt} = \ln\delta.$$

So, $\frac{1}{m}\sum_{i=1}^{m}\xi_2(z_i) - E(\xi_2) \leq t^*$ holds with probability $1 - \delta$. Then

$$\begin{aligned}t^* = & \frac{\frac{4b}{3}\ln(\frac{1}{\delta}) + \sqrt{(\frac{4b}{3}\ln(\frac{1}{\delta}))^2 + 2D(\lambda)(1+b)m\ln(\frac{1}{\delta})}}{m} \\ \leq & \frac{8b\ln(\frac{1}{\delta})}{3m} + \sqrt{\frac{2D(\lambda)(1+b)\ln(\frac{1}{\delta})}{m}} \\ \leq & \frac{8b\ln(\frac{1}{\delta})}{3m} + D(\lambda) + \frac{(1+b)\ln(\frac{1}{\delta})}{2m}.\end{aligned}$$

Recall $b = (\sqrt{D(\lambda)/\lambda} + R)^2 \leq 2(D(\lambda)/\lambda + R^2)$. It follows that

$$t^* \leq D(\lambda)\Big(1 + \frac{7\ln(\frac{1}{\delta})}{m\lambda}\Big) + \frac{7R^2\ln(\frac{1}{\delta})}{m} + \frac{\frac{1}{2}\ln(\frac{1}{\delta})}{m}.$$

Then we accomplish the proof of Proposition 5.

Appendix B

Proof of Theorem 1: Assume $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ is a u.e.M.c. sample. Similar to the proof of Theorem 2, we have that for any $0 < \delta < 1$, with probability at least $1 - \delta$, the inequality

$$\begin{aligned}\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq & 3D(\lambda) + \frac{224 \cdot (\sqrt{D(\lambda)/\lambda} + R)^2 \cdot \|\Gamma_0\|^2 \ln(\frac{2}{\delta})}{m} \\ & + 2\Big(\frac{448\|\Gamma_0\|^2 C_s R^{s+2}}{m}\Big)^{\frac{1}{s+1}}\end{aligned}$$

is valid provided that $m \geq 448R\|\Gamma_0\|^2 \ln(2/\delta)(\ln(2/\delta)/C_s)^{1/s}$. Then, we accomplish the proof of Theorem 1.

Proof of Corollary 1: Assume $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ is a u.e.M.c. sample. According to Definition 2, $D(\lambda) \leq \lambda^q$. Then for any $0 < \delta < 1$, with probability at least $1 - \delta$, we have

$$\begin{aligned}& \mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \\ \leq & 3D(\lambda) + \frac{224 \cdot (\sqrt{D(\lambda)/\lambda} + R)^2 \cdot \|\Gamma_0\|^2 \ln(\frac{2}{\delta})}{m} + 2\varepsilon^*(m, \delta/2) \\ \leq & 3D(\lambda) + \frac{224 \cdot (\sqrt{D(\lambda)/\lambda} + R)^2 \cdot \|\Gamma_0\|^2 \ln(\frac{2}{\delta})}{m} \\ & + \frac{896\|\Gamma_0\|^2 R^2 \ln(\frac{2}{\delta})}{m} + 2(\frac{448\|\Gamma_0\|^2 C_s R^{s+2}}{m})^{\frac{1}{s+1}} \\ \leq & \overline{C}\Big(\lambda^q + \frac{\lambda^{q-1}}{m} + \frac{1}{m} + \frac{\lambda^{(q-1)/2}}{m} + \frac{1}{m} + (\frac{1}{m})^{\frac{1}{1+s}}\Big),\end{aligned}$$

where $\overline{C} = 896\|\Gamma_0\|^2 R^2 (4C_s{}^{\frac{1}{s+1}} + 4\ln(2/\delta) + 3)$.

Let $\lambda = (\frac{1}{m})^{\frac{1}{1+s}}$ and q close to 1, so the inequality

$$\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq \overline{C}\Big(\lambda^q + \frac{\lambda^{q-1}}{m} + \frac{1}{m} + \frac{\lambda^{(q-1)/2}}{m} + \frac{1}{m} + (\frac{1}{m})^{\frac{1}{1+s}}\Big) \leq \overline{C}(\frac{1}{m})^{\frac{1}{1+s}}$$

is valid with probability at least $1 - \delta$, where $\overline{C} = 896\|\Gamma_0\|^2 R^2 (4C_s{}^{\frac{1}{s+1}} + 4\ln(2/\delta) + 3)$ is a constant. Then, we finish the proof of Corollary 1.

Proof of Theorem 2: Assume $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ is i.i.d. sample. With the bounds of T_1(Prop.3), T_2(Prop.5) and $D(\lambda)$(Def.3), we have that with confidence $1 - \delta$,

$$\begin{aligned}& \mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq T_1 + T_2 + D(\lambda) \\ \leq & \frac{1}{2}[\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B)] + D(\lambda)\Big(1 + \frac{7\ln(\frac{2}{\delta})}{m\lambda}\Big) + \frac{7R^2 \ln(\frac{2}{\delta})}{m} + \frac{\frac{1}{2}\ln(\frac{2}{\delta})}{m} + \overline{\varepsilon}(m, \delta/2).\end{aligned}$$

For $\overline{\varepsilon}(m, \delta/2)$, the inequality $(\frac{300R^2 C_s (4R)^s}{m})^{\frac{1}{s+1}} \geq \frac{300R^2 \ln(2/\delta)}{m}$ is valid with $m \geq 74R\ln(2/\delta)(\ln(2/\delta)/C_s)^{1/s}$, we get $\overline{\varepsilon}(m, \delta/2) = (\frac{300R^2 C_s (4R)^s}{m})^{\frac{1}{s+1}}$. Thus, for any $0 <$

$\delta < 1$, with probability at least $1-\delta$, we have

$$\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq D(\lambda)\Big(2 + \frac{14\ln(\frac{2}{\delta})}{m\lambda}\Big) + \frac{14R^2\ln(\frac{2}{\delta})}{m} + \frac{\ln(\frac{2}{\delta})}{m} + 2\Big(\frac{300R^2C_s(4R)^s}{m}\Big)^{\frac{1}{s+1}}.$$

Then, we finish the proof of Theorem 2.

Proof of Corollary 2: Assume $\mathbf{z} = \{z_i\}_{i=1}^m \in Z^m$ is i.i.d. sample. According to Definition 2, $D(\lambda) \leq \lambda^q$. Then for any $0 < \delta < 1$, with probability at least $1-\delta$, we have

$$\begin{aligned}
&\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \\
\leq& D(\lambda)\Big(2 + \frac{14\ln(\frac{2}{\delta})}{m\lambda}\Big) + \frac{14R^2\ln(\frac{2}{\delta})}{m} + \frac{\ln(\frac{2}{\delta})}{m} + 2\overline{\varepsilon}(m, \delta/2) \\
\leq& \lambda^q(2 + \frac{14\ln(\frac{2}{\delta})}{m\lambda}\Big) + \frac{14R^2\ln(\frac{2}{\delta})}{m} + \frac{\ln(\frac{2}{\delta})}{m} \\
&+ \frac{600R^2\ln(\frac{2}{\delta})}{m} + 2(\frac{300R^2C_s(4R)^s}{m})^{\frac{1}{s+1}} \\
\leq& \widehat{C}\Big(\lambda^q + \frac{\lambda^q}{m\lambda} + \frac{1}{m} + \frac{1}{m} + \frac{1}{m} + (\frac{1}{m})^{\frac{1}{1+s}}\Big),
\end{aligned}$$

where $\widehat{C} = 600R^2(4C_s{}^{\frac{1}{s+1}} + \ln(2/\delta))$. Let $\lambda = (\frac{1}{m})^{\frac{1}{1+s}}$ and q close to 1, so the inequality

$$\mathcal{E}(f_{\mathbf{z}}) - \mathcal{E}(f_B) \leq \widehat{C}\Big(\lambda^q + \frac{\lambda^q}{m\lambda} + \frac{3}{m} + (\frac{1}{m})^{\frac{1}{1+s}}\Big) \leq \widehat{C}(\frac{1}{m})^{\frac{1}{1+s}}$$

is valid with probability at least $1-\delta$, where $\widehat{C} = 600R^2(4C_s{}^{\frac{1}{s+1}} + \ln(2/\delta))$ is a constant. Then, we accomplish the proof of Corollary 2.

Acknowledgments. This work is supported by Scientific Research Foundation of Hubei University of Education for Talent Introduction (No.ESRC20230008), and Open Foundation of Hubei Key Laboratory of Applied Mathematics (Hubei University) (HBAM202304).

References

1. Angulo, C., Parra, X., Català, A.: K-svcr: A support vector machine for multi-class classification. Neurocomputing 55(1¨C2), 57–77 (2003)
2. Bennett, K., Mangasarian, O.L.: Combining support vector and mathematical programming methods for induction. Advances in Kernel Methods-SV Learning pp. 307–326 (1999)
3. Bottou, L., Cortes, C., Denker, J.S., Drucker, H., Guyon, I., Jackel, L.D., LeCun, Y., Muller, U.A., Sackinger, E., Simard, P.: Comparison of classifier methods: A case study in handwritten digit recognition. In: Proceedings of the 12th IAPR International Conference on Pattern Recognition. pp. 77–82 (1994)
4. Chao, Z.: Machine learning-based intelligent weather modification forecast in smart city potential area. Computer Science and Information Systems (20), 631–656 (2023)
5. Chavaltada, C., Pasupa, K., Hardoon, D.R.: Combining multiple features for product categorisation by multiple kernel learning. International Conference on Computing and Information Technology pp. 3–12 (2018)

6. Cucker, F., Smale, S.: Best choices for regularization parameters in learning theory: On the bias-variance problem. Foundations of Computational Mathematics 2(4), 413–428 (2002)
7. Dong, Z., Gong, J., Zou, B., Wang, Y., Xu, J.: Generalization and learning rate of multi-class support vector classification and regression. International Journal of Wavelets, Multiresolution and Information Processing (20), 2250017 (2022)
8. Dong, Z., Qin, Y., Zou, B., Xu, J., Tang, Y.Y.: Lmsvcr: Novel effective method of semi-supervised multi-classification. Neural Computing and Application (34), 3857–3873 (2022)
9. Dong, Z., Xu, C., Xu, J., Zou, B., Zeng, J., Tang, Y.Y.: Generalization capacity of multi-class svm based on markovian resampling. Pattern Recognition (142), 109720 (2023)
10. Duan, Y., Zou, B., Xu, J., Chen, F., Wei, J., Tang, Y.Y.: Oaa-svm-ms: A fast and efficient multi-class classification algorithm. Neurocomputing (454), 448–460 (2021)
11. Duğan, U., Glasmachers, T., Igel, C.: A unified view on multi-class support vector classification. Journal of Machine Learning Research 17(45), 1–32 (2016)
12. Feng, Y., Yang, Y., Zhao, Y., Lv, S., Suykens, J.A.: Learning with Kernelized Elastic Net Regularization. KU Leuven, Leuven, Belgium (2014)
13. Gönen, M., Alpaydin, E.: Multiple kernel learning algorithms. Journal of Machine Learning Research (12), 2211–2268 (2011)
14. Huang, C.L., Dun, J.F.: A distributed pso¨csvm hybrid system with feature selection and parameter optimization. Applied Soft Computing 8(4), 1381–1391 (2008)
15. Krebel, U.H.G.: Pairwise classification and support vector machines. Advances in kernel methods: support vector learning pp. 255–268 (1999)
16. Lanckriet, G.R.G., Cristianini, N., Bartlett, P.L., Ghaoui, L.E., Jordan, M.I.: Learning the kernel matrix with semi-definite programming. Journal of Machine Learning Research (5), 27–72 (2004)
17. Lauriola, I., Gallicchio, C., Aiolli, F.: Enhancing deep neural networks via multiple kernel learning. Pattern Recognition (101), 107194 (2020)
18. Lee, Y., Lin, Y., Wahba, G.: Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. Journal of the American Statistical Association 99(465), 67–81 (2004)
19. Lin, S.W., Ying, K.C., Chen, S.C., Lee, Z.J.: Particle swarm optimization for parameter determination and feature selection of support vector machines. Expert Systems with Applications 35(4), 1817–1824 (2008)
20. Luo, J., Orabona, F., Fornoni, M., Caputo, B., Cesa-Bianchi, N.: Om-2: An online multi-class multi-kernel learning algorithm. IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops pp. 43–50 (2010)
21. Lv, S.G., Zhou, F.Y.: Optimal learning rates of l^p-type multiple kernel learning under general conditions. Information Sciences (294), 255–268 (2015)
22. Lv, S.G., Zhu, J.D.: Error bounds for l^p-norm multiple kernel learning with least square loss. Abstract and Applied Analysis pp. 1–18 (2012)
23. Meyn, S.P., Tweedie, R.L.: Markov Chains and Stochastic Stability. Springer Science & Business Media (2012)
24. Qian, M., Nie, F., Zhang, C.: Efficient multi-class unlabeled constrained semi-supervised svm. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management. pp. 1665–1668 (2009)
25. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
26. Wang, T., Su, H., Li, J.: Dws-mkl: Depth-width-scaling multiple kernel learning for data classification. Neurocomputing (411), 455–467 (2020)
27. Weston, J., Watkins, C.: Support vector machines for multi-class pattern recognition. Esann pp. 219–224 (1999)
28. Wilson, C.M., Li, K.Q., Yu, X.Q., Kuan, P.F., Wang, X.F.: Multiple-kernel learning for genomic data mining and prediction. BMC Bioinformatics 20(1), 1–7 (2019)

29. Wu, Q., Ying, Y., Zhou, D.X.: Learning rates of least-square regularized regression. Foundations of Computational Mathematics 6(2), 171–192 (2006)
30. Wu, Q., Zhou, D.X.: Svm soft margin classifiers: Linear programming versus quadratic programming. Neural Computation 17(5), 1160–1187 (2005)
31. Xu, J., Tang, Y.Y., Zou, B., Xu, Z., Li, L., Lu, Y., Zhang, B.: The generalization ability of svm classification based on markov sampling. IEEE Transactions on Cybernetics 45(6), 1169–1179 (2015)
32. Yang, Y., Guo, Y., Chang, X.: Angle-based cost-sensitive multicategory classification. Computational Statistics & Data Analysis (156), 107107 (2021)
33. Yao, B., Liu, S., Wang, L.: Using machine learning approach to construct the people flow tracking system for smart cities. Computer Science and Information Systems (20), 679–700 (2023)
34. Yi, Z.H., Etemadi, A.H.: Line-to-line fault detection for photovoltaic arrays based on multiresolution signal decomposition and two-stage support vector machine. IEEE Transactions on Industrial Electronics 64(11), 8546–8556 (2017)
35. Zou, B., Li, L., Xu, Z.: The generalization performance of erm algorithm with strongly mixing observations. Machine Learning 75(3), 275–295 (2009)

Zijie Dong received the Ph.D. degree from the Faculty of Mathematics and Statistics of Hubei University, China, in 2022. She is currently working at School of Mathematics and Statistics, Hubei University of Education, Wuhan, 430205, China. Her current research interests include statistical learning theory, machine learning, and pattern recognition.

Fen Chen received the Ph.D. degree from the Faculty of Mathematics and Statistics of Hubei University, China, in 2019. She is currently working at School of Finance, Hubei University of Economics, Wuhan, 430205, China.

Yu Zhang received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, China, in 2021. He is currently working at School of Mathematics and Statistics, Hubei University of Education, Wuhan, 430205, China.

Received: January 24, 2023; Accepted: December 10, 2023.