Kernel Stability for Model Selection in Kernel-Based Algorithms

Yong Liu, Shizhong Liao, Hua Zhang, Wenqi Ren, and Weiping Wang

***Abstract*—Model selection is one of the fundamental problems in kernel-based algorithms, which is commonly done by minimizing an estimation of generalization error. The notion of stability and cross-validation (CV) error of learning machines consists of two widely used tools for analyzing the generalization performance. However, there are some disadvantages to both tools when applied for model selection: 1) the stability of learning machines is not practical due to the difficulty of the estimation of its specific value and 2) the CV-based estimate of generalization error usually has a relatively high variance, so it is prone to overfitting. To overcome these two limitations, we present a novel notion of kernel stability (KS) for deriving the generalization error bounds and variance bounds of CV and provide an effective approach to the application of KS for practical model selection. Unlike the existing notions of stability of the learning machine, KS is defined on the kernel matrix; hence, it can avoid the difficulty of the estimation of its value. We manifest the relationship between the KS and the popular uniform stability of the learning algorithm, and further propose several KS-based generalization error bounds and variance bounds of CV. By minimizing the proposed bounds, we present two novel KS-based criteria that can ensure good performance. Finally, we empirically analyze the performance of the proposed criteria on many benchmark data, which demonstrates that our KS-based criteria are sound and effective.**

***Index Terms*—AutoML, cross-validation (CV), generalization error, kernel methods, kernel selection, model selection, stability.**

I. Introduction

MODEL selection is a key issue in statistical learning [1], and its target is to estimate the generalization error and the choice of an appropriate hypothesis space [2]. In kernel-based algorithms [3], [4], such as SVM [1], kernel ridge regression (KRR) [5] and least squares SVM [6], the reproducing kernel Hilbert spaces (RKHSs) are the candidate hypothesis spaces which are determined on the choice of the kernel function.

Manuscript received July 27, 2018; revised March 15, 2019; accepted June 12, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61703396, Grant 61673293, Grant 61602467, and Grant 61602464, in part by the Youth Innovation Promotion Association CAS, in part by the Excellent Talent Introduction of Institute of Information Engineering of CAS under Grant Y7Z0111107, and in part by the CCF-Tencent Open Research Fund. This paper was recommended by Associate Editor G. C. Anagnostopoulos. *(Corresponding author: Yong Liu.)*

Y. Liu, H. Zhang, W. Ren, and W. Wang are with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: liuyong@iie.ac.cn).

S. Liao is with the College of Intelligence and Computing, Tianjin University, Tianjin 300072, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2019.2923824

The common approach to estimate the generalization error is via testing on some unused data or via a theoretical bound [7]. To estimate the theoretical bound of generalization error of learning machines, some measures of complexity of space are introduced, such as cover number [8], [9]; VC dimension [1]; Rademacher complexity [10]; maximal discrepancy [11]; radius-margin bound [1], [12]; span bound [7], [13]; compression coefficient [14]; Bayesian regularization (BR) [15]; influence function [16]; eigenvalues perturbation [12]; local Rademacher complexity [17]–[20]; eigenvalues ratio [18], spectral measure [21]; etc. Minimizing an empirical estimation of the generalization error of learning machine is an alternative for model selection [22], [23]. Cross-validation (CV) is the most widely used empirical estimation. However, the CV-based estimates of generalization error have a relatively high variance, that is, the estimates of performance are highly variant, dependent on the training data. Thus, it is prone to overfitting [15], [24]–[26].

In recent years, several notions of stability of learning algorithm have been introduced for deriving theoretical generalization bounds to estimate the generalization error. Algorithmic stability was first introduced by Devroye and Wagner [27] for analyzing of the leave-one-out CV error. Kearns and Ron [28] studied it further and proposed a new notion of stability, called error stability, to derive sanity-check bounds. Bousquet and Elisseeff [29] introduced a stronger notion of uniform stability, and manifested that it could be used to obtain tight bounds. Moreover, they showed that it could be applied to large classes of algorithms. The relationship between stability and consistency was studied by Poggio *et al.* [30]. Cortes *et al.* [31] introduced the algorithmic stability, and further used it to prove generalization bounds. The stability-based bounds for kernel approximation were proposed by Cortes *et al.* [32], [33]. The connection between stability, uniform convergence, and learnability was studied by Shalev-Shwartz *et al.* [34]. Gao and Zhou [35] extended the above results, and studied the relationship between uniform convergence, stability, and learnability of ranking. Unfortunately, the focus of these stabilities is to obtain the theoretical generalization bounds, and are hard to estimate their specific values [36], making them unusable for practical model selection.

In this paper, we present a model selection method via a newly defined stability, called kernel stability (KS). Unlike the existing notions of stability of the learning machine, see [28]–[30], [32], [33], [36], [37], KS is defined via the kernel matrix, which can be easily estimated according to the

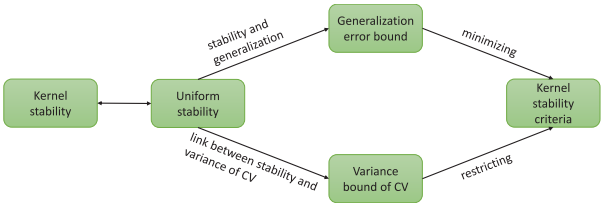


Fig. 1. Main work. We first establish the relationship between KS and the popular uniform stability of the learning algorithm for LSSVM, KRR, and SVM. Then, we derive several KS-based generalization error bounds and variance bounds of CV. Finally, we propose two novel KS-based criteria by minimizing the derived bounds.

available data. We manifest that KS can be used to prove upper bounds of generalization error and variance of CV for SVM, KRR, and LSSVM. Furthermore, we propose two novel KS-based model selection criteria: 1) by minimizing the bounds of generalization error, which can ensure good generalization performance and 2) by controlling the bounds of CV, which can avoid the high variance of CV.

Our proposed criteria have a sound theoretical foundation, and can also be validated by experimental results. The main work is shown in Fig. 1.

This paper is an extension of our previous work published in ECML [38], compared with [38], this paper contains many new contents, including:

1) a novel generalization theory with KS for kernel-based algorithms;
2) novel KS-based generalization bounds for SVM, KRR, and LSSVM;
3) a novel model selection criterion based on the generalization error bounds with KS;
4) the refined proofs of the theoretical results;
5) multiple experimental investigations of our KS-based criteria.

A. Related Work

1) Variance of CV: The CV has been studied and used for lots of years [24], [25], [39], [40], but exploring the variance of CV is tricky. Blum *et al.* [41] proved that the variance of a single holdout estimate is never smaller than that of the CV-based estimate. Bengio and Grandvalet [39] asserted that there exists no universal unbiased estimator of the variance of CV. Cawley and Talbot [15] investigated the use of BR [42] in model selection for reducing the effects of the high variance of leave-one-out CV for LSSVM. Kumar *et al.* [43] extended the result of [41], and used the algorithm stability to quantify the variance reduction. Different from the above works, we consider estimating the variance of CV via an appropriately defined stability on kernel matrix for model selection.

2) Stability: Most of the notions of stability, defined on learning machines, have been introduced to derive the generalization error bounds, though theoretically appealing, are not practical for model selection. Liu *et al.* [12] introduced a notion, called eigenvalues perturbation, which measures the difference between the eigenvalues of the integral operator and those of kernel matrix. Different from the notions of stability of the learning machine, eigenvalues perturbation is defined via the eigenvalues of integral operator induced by the kernel function. However, the eigenvalues of the integral operator are also hard to be estimated as the probability distribution is unknown. Liu and Liao [44] investigated it further and gave a notion of spectral perturbation stability (SPS). However, the eigenvalues of kernel matrix are very sensitive to the dataset. Moreover, it needs to calculate the eigenvalues of all the perturbed kernel matrices to estimate the value of SPS, which has a high computational cost. In this paper, we introduce a novel notion of stability, defined on kernel matrix, which has high computational efficiency and are practical for model selection.

3) Outlines: Preliminaries and some notations are given in Section II. In Section III, we introduce the notion of KS and establish the relationship between KS and popular uniform stability. In Section IV, we use the notion of KS to prove the generalization bounds for SVM, KRR, and LSSVM. In Section V, we obtain the variance bounds of CV with KS. In Section VI, we propose two novel model selection criteria. The comparison of our criteria with the state-of-the-art model selection criteria is proposed in Section VII. In Section VIII, we give the discussion of this paper. We conclude in Section IX. Most of the proofs are given in the Appendix.

II. Preliminaries and Notations

Let $\mathcal{X} \subset \mathbb{R}^d$ be the input space and $\mathcal{Y}$ the output space. For classification, $\mathcal{Y} = \{+1, -1\}$, for regression $\mathcal{Y} \subseteq \mathbb{R}$

$$S = (z_1 = (\mathbf{x}_1, y_1), z_2 = (\mathbf{x}_2, y_2), \ldots, z_n = (\mathbf{x}_n, y_n))$$

is the training set of size n drawn independent identically distributed (i.i.d) from a probability distribution $\mathbb{P}$ on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. The ith removed set, $i \in \{1, 2, \ldots, n\}$, is denoted as

$$S^i = S \backslash z_i = (z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n).$$

A symmetric continuous function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel if $\forall \{\mathbf{x}_i\}_{i=1}^n \in \mathcal{X}^n$, the kernel matrix

$$\mathbf{K} = \left[K(\mathbf{x}_i, \mathbf{x}_j)\right]_{i,j=1}^n$$

is positive semidefinite. The RKHS $\mathcal{H}_K$ induced by K is the span of $\{K(\mathbf{x}, \cdot) : \mathbf{x} \in \mathcal{X}\}$, where the inner product satisfying

$$\left\langle K(\mathbf{x}, \cdot), K\left(\mathbf{x}', \cdot\right)\right\rangle_K = K\left(\mathbf{x}, \mathbf{x}'\right).$$

Assume that $|y| \leq M$ and $\forall \mathbf{x} \in \mathcal{X}$, $K(\mathbf{x}, \mathbf{x}) \leq \kappa$. In classification case, $M = 1$.

The learning machines, we focus on, are the kernel-based regularized algorithms [45]

$$f_S \coloneqq \arg\min_{f \in \mathcal{H}_K} \left\{ \frac{1}{|S|} \sum_{z \in S} \ell(f(\mathbf{x}), y) + \lambda \|f\|_K^2 \right\} \tag{1}$$

where $|S|$ is the size of the dataset S, $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ is a loss function, and λ is a tradeoff parameter.

1) *For KRR and LSSVM:* $\ell(t, y) = (t - y)^2$.
2) *For SVM:* $\ell(t, y) = \max(0, 1 - yt)$.

The performance of the kernel-based regularized algorithms is commonly measured by the *generalization error*

$$R(f_S) \coloneqq \int_{\mathcal{X} \times \mathcal{Y}} \ell(f_S(\mathbf{x}), y) \mathrm{d}\mathbb{P}(\mathbf{x}, y) \tag{2}$$

where f_S is the solution of the kernel-based regularized algorithm on S. The *empirical error* of f_S is denote as

$$R_{\text{emp}}(f_S) = \frac{1}{|S|}\sum_{z\in S}\ell(f_S(\mathbf{x}), y). \tag{3}$$

Let $S_1, \ldots, S_k$ be the folds of S, that is a random k-parts equipartition of S. We assume that n mod k for simplicity (note that for an arbitrary n and k, S can always be partitioned into k subdatasets, each with either $\lfloor (n/k) \rfloor$ or $\lceil (n/k) \rceil$). Let $f_{S\backslash S_i}$ be the hypothesis learned on $S\backslash S_i$

$$f_{S\backslash S_i} \coloneqq \arg\min_{f\in\mathcal{H}_K}\left\{\frac{1}{|S\backslash S_i|}\sum_{z\in S\backslash S_i}\ell(f(\mathbf{x}), y) + \lambda\|f\|_K^2\right\}.$$

Definition 1 (Uniform Stability [29]): An algorithm f is of β uniform stability with respect to the loss function $\ell(\cdot,\cdot)$ if the following holds: $\forall S \in \mathcal{Z}^n, i \in \{1, \ldots, n\}$, $z = (\mathbf{x}, y) \in \mathcal{Z}$

$$\left|\ell(f_S(\mathbf{x}), y) - \ell\left(f_{S^i}(\mathbf{x}), y\right)\right| \le \beta.$$

Remark 1: According to the above definition, one can see that the notion of uniform stability is defined on the learning algorithm. Therefore, if we want to estimate its value from the available empirical data, we should train the learning machine many times. Specifically, from Definition 1, one can see that

$$\hat{\beta} = \max_{i,j\in\{1,\ldots,n\}}\left|\ell\left(f_S(\mathbf{x}_j), y_j\right) - \ell\left(f_{S^i}(\mathbf{x}_j), y_j\right)\right|$$

is an empirical estimation of β from available empirical data S. To compute the empirical $\hat{\beta}$, we need to train the learning machine n^2 times, which is impracticable for kernel-based algorithms. To address this problem, we will introduce a novel stability defined on the kernel matrix in the next section for practical model selection.

III. Kernel Stability

In this section, we will give the definition of KS first, and then manifest the relationship between KS and uniform stability.

A. Definition of KS

The way of making the definition of KS is to start from the goal: to get bounds on generalization error or the variance of CV and want these bounds to be tight when the kernel function satisfies the KS.

It is well known that the kernel matrix contains most of the information needed by kernel methods. Therefore, we introduce a new notion of stability to quantify the perturbation of the kernel matrix with respect to the changes in the training set for kernel selection.

To this end, we let the ith removed kernel matrix $\mathbf{K}^i$ be

$$\begin{cases}[\mathbf{K}^i]_{jk} = 0 & \text{if } j \text{ or } k = i,\\ [\mathbf{K}^i]_{jk} = K\left(\mathbf{x}_j, \mathbf{x}_k\right) & \text{if } j \text{ and } k \ne i\end{cases} \tag{4}$$

that is, the elements of the ith row and column of $\mathbf{K}^i$ are 0, the other elements of $\mathbf{K}^i$ are the same as those of the kernel matrix $\mathbf{K}$.

Definition 2 (KS): A kernel function K is called a β-KS kernel if $\forall\{\mathbf{x}_i\}_{i=1}^n \in \mathcal{X}^n, \forall i \in \{1, \ldots, n\}$

$$\left\|\mathbf{K} - \mathbf{K}^i\right\|_2 \le \beta$$

where $\mathbf{K}$ is the kernel matrix, $\mathbf{K}^i$ is the ith removed kernel matrix defined in (4), and $\|\cdot\|_2$ is the 2-norm of matrix.

One can see that the notion of KS is used to quantify the perturbation of the kernel matrix when an arbitrary data point is removed. While for the popular uniform stability, it is used to quantify the perturbation of the learning machine when an arbitrary data point is removed. Therefore, loosely speaking, KS can be considered as an extension of the uniform stability to kernel matrix. Moreover, KS is defined on the kernel matrix; thus, we can estimate its value from empirical data easily (see Theorem 7 for detail), which makes this stability usable for model selection in practice.

Remark 2: In the definition of KS, we use the $\|\cdot\|_2$ to quantify the difference of $\mathbf{K}$ and $\mathbf{K}^i$. It is also feasible to use other matrix norm, such as trace norm $\|\cdot\|_*$ or Frobenius norm (also called Hilbert–Schmidt norm) $\|\cdot\|_F$, but the upper bounds based on $\|\cdot\|_*$ or $\|\cdot\|_F$ are looser than those of $\|\cdot\|_2$. Thus, we only consider $\|\cdot\|_2$ in this paper.

B. KS and Uniform Stability

In this section, we will give the relationship between KS and uniform stability for KRR, LSSVM, and SVM.

1) KRR and LSSVM:

Theorem 1: If K is a β-KS kernel, then the KRR and LSSVM algorithms are both of $\mathcal{O}(\beta/n)$-uniform stability.

The above theorem shows that the KS can measure the stability of KRR and LSSVM, which demonstrates the effectiveness of the application of KS.

Remark 3: In Theorem 1, for simple, the λ is considered as a constant, and is ignored. But, in fact, from the proof in the Appendix, we can find that uniform stability is inversely proportional to parameter λ.

2) SVM:

Theorem 2: If K is a β-KS kernel, then the SVM is $\mathcal{O}(\beta)$-uniform stability.

Remark 4: Uniform stability is a strong notion of stability [29]; therefore, if a learning algorithm is of β uniform stability, it is also of other weaker stability, such as hypothesis stability and pointwise hypothesis stability.

IV. Generalization Bounds With KS

In this section, we will use the notion of KS to obtain the generalization bounds for SVM, KRR, and LSSVM.

Theorem 3: If K is a β-KS kernel, then for the KRR and LSSVM, $\delta \in (0, 1)$, with probability at least $1 - \delta$, we have

$$R(f_S) \le R_{\text{emp}}(f_S) + \mathcal{O}\left(\frac{\beta}{n} + \beta\sqrt{\frac{\ln(1/\delta)}{n}}\right)$$

where $R(f_S)$ is the generalization error defined in (2) and $R_{\text{emp}}(S)$ is the empirical error defined in (3).

Remark 5: One can see that the convergence rate of $R(f_S) - R_{\text{emp}}(f_S)$ of our bound using KS is related to $\sqrt{\ln(1/\delta)}$. While

for the other popular notions of stability, such as hypothesis stability and pointwise hypothesis stability [29], the convergence rate of $R(f_S) - R_{\text{emp}}(f_S)$ is related to $\sqrt{1/\delta}$: if an algorithm f is of β_1 hypothesis stability or β_2 pointwise hypothesis stability, then with probability $1-\delta$, we have

$$R(f_S) \leq R_{\text{emp}}(f_S) + \sqrt{\frac{\beta_1}{\delta}}$$

and

$$R(f_S) \leq R_{\text{emp}}(f_S) + \sqrt{\frac{\beta_2}{\delta}}.$$

Note that the value of δ is usually very small, such as 0.01 or 0.005, thus $\sqrt{\ln(1/\delta)}$ is much smaller than $\sqrt{1/\delta}$, which demonstrates that our generalization bound is tighter than those of hypothesis stability and pointwise hypothesis stability.

For uniform stability, the convergence rate is related to $\ln(1/\delta)$ [29]: if an algorithm f is of β_3 uniform stability then with probability $1-\delta$, we have

$$R(f_S) \leq R_{\text{emp}}(f_S) + n\beta_3\sqrt{\frac{\ln(1/\delta)}{n}}.$$

From the relationship between the uniform stability and KS (see Theorem 1), we can find that the convergence rates of KS is the same as the uniform stability.

Remark 6: From the definition of KS, one can see that β is irrelevant to λ. However, to obtain the generalization bound, we use the uniform stability as a bridge. Since the uniform stability is inversely proportional to λ (see Remark 3), it is easy to verify that the second term of the generalization bound is proportional to β and irrelevant proportional to λ. Generally speaking, the more stable the algorithm is, the more likely underfitting it is. So, the empirical error (the first term of generalization bound) is usually a positive correlation to λ and negative correlation to β. Thus, only appropriate β and λ (not too big or too small) can lead better generalization bound, which shows that the derived bound is not trivial.

The above theorem demonstrates the effectiveness of the applying of KS to estimate the generalization error. Thus, to ensure good generalization performance, we can choose the kernel function from the following criterion:

$$K^* = \arg\min_{K\in\mathcal{K}} R_{\text{emp}}(S) + \frac{\eta}{n}\beta$$

where $\mathcal{K}$ is a candidate set of kernel functions and η is the tradeoff parameter.

Theorem 4: If K is a β-KS kernel, then for the SVM, with probability $1-\delta$, we have

$$R(f_S) \leq R_{\text{emp}}(f_S) + \mathcal{O}\left(\sqrt{\frac{\beta^{\frac{1}{2}}}{\delta}}\right).$$

One can see that the bound for SVM is weaker than the bounds for LSSVM and KRR, which is due to the difference in loss functions defining the optimization problems of these algorithms.

Remark 7: The performance of stability is very important to the learning algorithm, but most of the existing stability is defined on learning algorithm, making them unusable for practical model selection. Thus, the motivation of this paper is to design another notion of stability, which is simple and elegant, and can be used for practical model selection. Based on this motivation, we introduce the notion of KS. In [29], they adopted the tool of McDiarmid's concentration inequality to establish the relationship between uniform stability and generalization bound. Thus, to derive generalization bounds with KS, we first show the relationship between KS and uniform stability, and further derive KS-based generalization bounds. Although the proposed measure may bare similarity to the condition in McDiarmid's concentration inequality with bounded variation, which has been used extensively in the kernel community, such as [46]–[48], the motivation this paper is totally different from the above papers. Moreover, we prove the bound of the variance of CV, the McDiarmid's concentration inequality is not used to the proof of upper bound.

V. Variance Bounds of CV With KS

In this section, we will consider the use of KS to bound the variance of k-fold CV.

The k-fold CV hypothesis $f_{k\text{cv}}$ and the (empirical) loss of $f_{k\text{cv}}$ are defined as

$$f_{k\text{cv}} = \frac{1}{k}\sum_{i=1}^{k} f_{S\backslash S_i},\ \ell_{f_{k\text{cv}}}(S) = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{|S_i|}\sum_{z\in S_i}\ell\big(f_{S\backslash S_i}(\mathbf{x}), y\big)$$

respectively. The variance of k-fold CV hypothesis is defined as [43]

$$\operatorname*{var}_S\big(\ell_{f_{k\text{cv}}}(S)\big) = \mathop{\mathbb{E}}_{S\sim\mathcal{Z}^n}\left[\ell_{f_{k\text{cv}}}(S) - \mathop{\mathbb{E}}_{S\sim\mathcal{Z}^n}\big[\ell_{f_{k\text{cv}}}(S)\big]\right]^2$$

where $\mathbb{E}_{S\sim\mathcal{Z}^n}$ is the expectation when $S = \{z_i\}_{i=1}^n$ is sampled according to $\mathbb{P}$.

The CV is the popular criterion for model selection, but it is known to exhibit a relatively high variance. Thus, k-fold CV is prone to overfitting [15], [24]–[26]. To address this problem, we can choose the appropriate kernel function via the following criterion:

$$K^* = \arg\min_{K\in\mathcal{K}} \ell_{f_{k\text{cv}}}(S) + \eta\cdot\operatorname*{var}_S\big(\ell_{f_{k\text{cv}}}(S)\big)$$

where $\ell_{f_{k\text{cv}}}(S)$ can be considered as the bias of $R(f_S)$, and the second part $\text{var}_S(\ell_{f_{k\text{cv}}}(S))$ of this criterion is used to restrict the high variance of CV. Since probability distribution is unknown, $\text{var}_S(\ell_{f_{k\text{cv}}}(S))$ is not directly computable, In the following, we will give the upper bound of $\text{var}_S(\ell_{f_{k\text{cv}}}(S))$ with KS.

Theorem 5: If K is a β-KS kernel, then for KRR and LSSVM

$$\operatorname*{var}_S\big(\ell_{f_{k\text{cv}}}(S)\big) \leq \mathcal{O}\left(\frac{\beta^2}{m^2}\right)$$

where $m = ((k-1)n/k)$.

This theorem manifests that we can apply the β to control the value of $\text{var}_S(\ell_{f_{k\text{cv}}}(S))$. Therefore, it is reasonable to choose the appropriate kernel function that has a small β to prevent the high variance

$$K^* = \arg\min_{K\in\mathcal{K}} \ell_{f_{k\text{cv}}}(S) + \frac{\eta}{n}\beta$$

where β is used to restrict the high variance.

Theorem 6: If K is a β-KS kernel, then for SVM

$$\operatorname*{var}_{S}\left(\ell_{f_{\text{kcv}}}(S)\right) \leq \mathcal{O}(\beta).$$

VI. Model Selection

In this section, we will show how to use KS for practical model selection.

From the generalization error bounds and the upper bounds of the variance derived in the above sections, to guarantee good generalization performance, it is reasonable to consider the use of the following criteria:

$$\arg\min_{K\in\mathcal{K}} R_{\text{emp}}(S) + \frac{\eta}{n}\beta \text{ and } \arg\min_{K\in\mathcal{K}} \ell_{f_{\text{kcv}}}(S) + \frac{\eta}{n}\beta.$$

However, from the definition of KS, we should try all the possibilities of the S drawn from the probability distribution $\mathbb{P}$ to obtain β, which is impracticable. Thus, we should estimate its value from the available empirical data. One can see that

$$\hat{\beta} = \max_{i\in\{1,\ldots,n\}} \left\|\mathbf{K} - \mathbf{K}^i\right\|_2$$

is an empirical estimate of β. Therefore, we consider the use of the following KS-based criteria in practice:

$$\text{RKS}(\mathcal{K}) = \arg\min_{K\in\mathcal{K}} R_{\text{emp}}(S) + \frac{\eta}{n}\hat{\beta} \tag{5}$$

and

$$\text{CVKS}(\mathcal{K}, k) = \arg\min_{K\in\mathcal{K}} \ell_{f_{\text{kcv}}}(S) + \frac{\eta}{n}\hat{\beta}. \tag{6}$$

One can see that these two criteria includes two parts: 1) bias and 2) variance. $R_{\text{emp}}(S)$ or $\ell_{f_{\text{kcv}}}(S)$ can be considered as the bias of generation error, and $\hat{\beta}$ considered as the variance.

To apply these two criteria, we need to calculate the $\|\mathbf{K} - \mathbf{K}^i\|_2$, that is, the largest eigenvalue of $[\mathbf{K} - \mathbf{K}^i]$, $i = 1, \ldots, n$, which has a high computational cost. Fortunately, the closed form of $\|\mathbf{K} - \mathbf{K}^i\|_2$ exists, and can be effectively computed.

Theorem 7: $\forall\ S \in \mathcal{Z}^n$ and $i \in \{1, \ldots, n\}$

$$\hat{\beta} = \max_{i\in\{1,\ldots,n\}} \frac{\mathbf{K}_{ii} + \sqrt{\mathbf{K}_{ii}^2 + 4\sum_{j=1,j\neq i}^{n} \mathbf{K}_{ji}^2}}{2}.$$

Proof: According to the definitions of $\mathbf{K}$ and $\mathbf{K}^i$, one can see that its characteristic polynomial can be written as

$$\det\left(t\mathbf{I} - \left(\mathbf{K} - \mathbf{K}^i\right)\right) = t^{n-2}\left(t^2 - \mathbf{K}_{ii}t - \sum_{j=1,j\neq i}^{n} \mathbf{K}_{ji}^2\right).$$

Therefore, the eigenvalues of $\mathbf{K} - \mathbf{K}^i$ can be written as

$$\sigma\left(\mathbf{K} - \mathbf{K}^i\right) = \left\{\frac{\mathbf{K}_{ii} \pm \sqrt{\mathbf{K}_{ii}^2 + 4\sum_{j=1,j\neq i}^{n} \mathbf{K}_{ji}^2}}{2}, \overbrace{0, \ldots, 0}^{n-2}\right\}.$$

Thus, the biggest eigenvalue is

$$\frac{\mathbf{K}_{ii} + \sqrt{\mathbf{K}_{ii}^2 + 4\sum_{j=1,j\neq i}^{n} \mathbf{K}_{ji}^2}}{2}$$

which completes the proof. ■

Theorem 7 manifests that we only need $O(n^2)$ computational time to compute $\hat{\beta}$.

Remark 8: In this paper, we estimate the β based on the training data, and further uses it to model selection, which seems that it might lead to overfitting. In the following, we will consider the use of the popular polynomial kernel $K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^T\mathbf{x})^d$ as an example to clarify that our criterion can mitigate the overfitting of kernel-based algorithms (the similar analysis can easily be extended to the popular Gaussian kernel). Note that the larger value the d is, the more likely to be overfitting the kernel-based algorithm is. We can also find that when d increases, the estimation of β increases rapidly (see Theorem 7). Therefore, our criteria can avoid the large value of d, which can mitigate the overfitting of kernel-based algorithms.

Time Complexity: The time complexity of RKS(K) and CVKS(K, k) are $\mathcal{O}(n^2+J)$ and $\mathcal{O}(n^2+kF)$, respectively, where n is the size of the data, J and F are the time complexity of $R_{\text{emp}}(f_S)$, and the training on the size of $k-1$ folds of data.

In our previous work [22], [23], [49], we proposed a method to approximate the k-fold CV via Bouligand influence function [50]. The proposed approximate method requires training the algorithm only once. Therefore, the time complexity of CVKS(K, k) can be reduced to $O(F + n^2)$. Some other approximate methods, such as Nyström method [51], modified Nyström method [52], and random sample method, can be also used to reduce the computational complexity of $R_{\text{emp}}(f_S)$.

Remark 9: From Theorem 7, we know that we need $O(n^2)$ computational time to compute $\hat{\beta}$, the time cost is too high for large-scale problems. Therefore, for large-scale problems, we should approximate the $\hat{\beta}$. Nyström method and random projection are two available methods to approximate the $\hat{\beta}$. In this paper, we mainly want to verify the effectiveness of our KS criterion, thus we do not consider the use of the approximate KS for the large-scale problem.

VII. Experiments

We will give the empirical analysis of the performance of our proposed KS-based criteria in this section.

Eighteen public available benchmark datasets are used from LIBSVM Data[1] seen in Table I. We consider the use of the popular Gaussian kernel

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\tau}\right)$$

and polynomial kernel

$$K(\mathbf{x}, \mathbf{x}') = \left(1 + \mathbf{x}^T\mathbf{x}\right)^d$$

as our candidate kernels, $\tau \in \{2^i, i = -15, -14, \ldots, 15\}$ and $d \in \{1, 2, \ldots, 10\}$. We will compare our proposed RKS and k-CVKS (k is the fold of CV) with four popular model selection criteria:

1) k-fold CV (k-CV), $k = 5, 10$;
2) efficient leave-one-out CV (ELOO) [53];
3) BR [15];

[1] http://www.csie.ntu.edu.tw/∼cjlin/libsvm

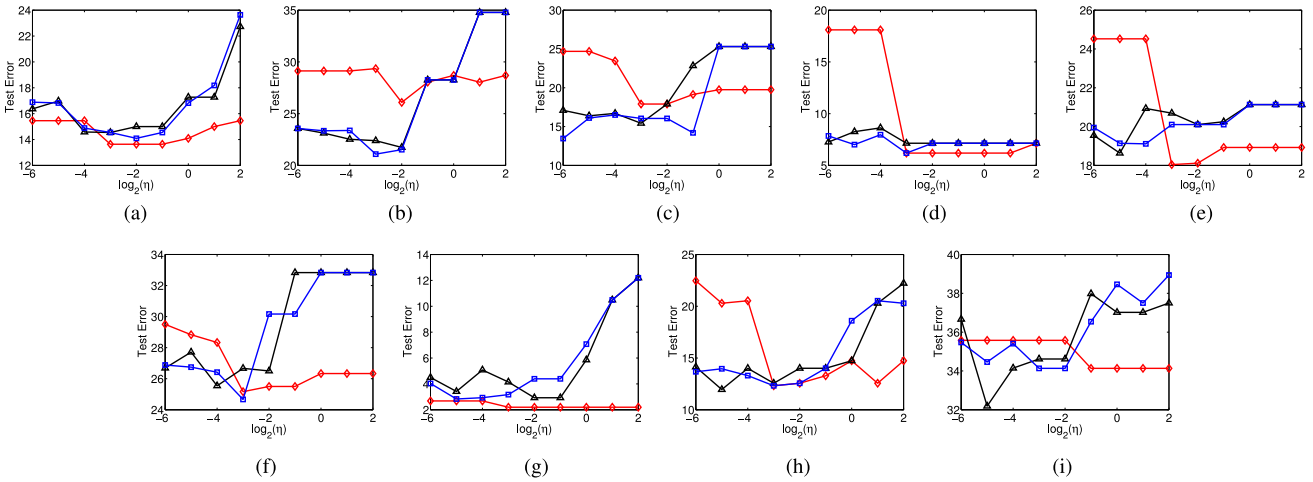


Fig. 2. Test errors of 5-CVKS (black line), 10-CVKS (blue line), and RKS (red line) with different η. In this experiment, we set $\lambda = 1/n$ (in Table II, we know that $n\lambda = 1$ can achieve good results on most datasets. Therefore, we only set $n\lambda = 1$). (a) sonar. (b) diabetes. (c) heart. (d) ionosphere. (e) a2a. (f) german.numer. (g) breast-cancer. (h) australian. (i) liver-disorders.

TABLE I
SUMMARY OF THE DATASETS

Dataset	Sample size	Feature	Type
australian	690	14	Classification
heart	270	13	Classification
ionosphere	351	34	Classification
breast	683	10	Classification
diabetes	768	8	Classification
german	1000	24	Classification
liver	345	5	Classification
sonar	208	60	Classification
a2a	2265	123	Classification
bodyfat	252	14	Regression
housing	506	13	Regression
mpg	392	7	Regression
pyrim	74	27	Regression
triazines	186	60	Regression
eunite	336	16	Regression
mg	1,385	6	Regression
space-ga	3,107	6	Regression
cpusmall	8,192	12	Regression

4) SPS [44]

$$\mathrm{SPS}(K) = R_{\mathrm{emp}}(S) + \delta \frac{1}{m^2} \sum_{j=1}^{m} \sum_{i=1}^{m} \left| \sigma_j(\mathbf{K}) - \sigma_j\left(\mathbf{K}^i\right) \right|$$

where $\sigma_j(\mathbf{K})$ is the eigenvalue of $\mathbf{K}$ (see detail in [44]) and δ is the tradeoff parameter.

We run all the algorithms 50 times with randomly selected 70% of data for training and the other 30% for testing on each dataset. All statements of statistical significance in the remainder refer to a 95% level of significance of t-test. The learning machine we considered for classification is SVM and for regression is KRR.

A. Classification

For each training set, we select the kernel parameter τ with each model selection criterion for each fixed regularized parameter $n\lambda \in \{0.1, 1, 10, 100\}$, and then we compute the test error on the testing set with the chosen optimal parameters. The optimal parameters $\eta \in \{2^i, i = -5, 0, 5, 10\}$ of CVKS and RKS, and the parameter $\delta \in \{2^i, i = -5, 0, 5, 10\}$ of SPS are determined by threefold CV on the training set.

The test errors are shown in Table II that can be summarized as follows.

1) On most datasets, k-CVKS gains better accuracy results than k-CV, $k = 5, 10$. In particular, for $n\lambda = 1$, k-KS is significantly better than k-CV on 4 out of 9 sets without being worse on the other five datasets. The results of the other values of $n\lambda$ are similar with that of $n\lambda = 1$. These results indicate that using the KS to control the high variance of CV can improve generalization performance.
2) RKS is significantly better than SPS on most datasets. For $n\lambda = 1$, RKS significant outperforms SPS on 7 out of 9 datasets. The results of the other values of $n\lambda$ are similar with that of $n\lambda = 1$. This can possibly be explained by the fact that SPS is defined based on the eigenvalues, which is very sensitive to the dataset and only includes a part information of kernel matrix, thus the model chosen by this criterion may not ensure good performance.
3) k-CVKS outperforms BR. In particular, for $n\lambda = 1$, k-CVKS is significantly better than BR on 4 (or more) out of 9 sets without being significantly worse on any of the remaining datasets. For other values of $n\lambda$, the results are similar with that of $n\lambda = 1$. Thus, it indicates that using the KS as the regularization term for restricting the high variance is better than that of BR.
4) On most datasets, BR is comparable or better than ELOO which manifests that using the BR can ameliorate the high variance of leave-one-out CV.
5) The 5-CVKS, 10-CVKS, and RKS give comparable results.

In this experiment, we explore the influence of η for CVKS and RKS. The test errors with different η is plotted in Fig. 2. On each fixed η, we select the τ by 5-CVKS, 10-CVKS, and RKS on the training set, and obtain the test errors with the chosen kernel parameters on the testing set.

TABLE II
COMPARISON OF MEAN TEST ERRORS (%) AMONG OUR KS CRITERIA: 5-CVKS, 10-CVKS, RKS, AND OTHER POPULAR ONES, INCLUDING ELOO, 5-CV, 10-CV, BR, AND SPS. WE BOLD THE BEST METHOD AND UNDERLINE THE OTHER METHODS THAT ARE NOT SIGNIFICANTLY WORSE THAN THE BEST ONE

$n\lambda = 0.1$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	SPS
australian	13.14(1.0)	15.51(1.4)	**12.52(1.4)**	14.78(1.6)	15.46(2.6)	14.30(2.4)	14.30(2.9)	14.15(2.5)
heart	18.52(3.1)	**18.17(2.3)**	19.54(3.9)	19.01(3.7)	18.27(3.4)	18.52(3.3)	18.52(6.1)	18.83(4.5)
ionosphere	4.38(1.5)	**4.35(1.3)**	5.33(1.5)	4.76(1.7)	4.67(2.2)	5.33(2.1)	5.13(3.9)	6.76(2.8)
breast	3.41(0.7)	3.45(0.6)	**3.02(0.8)**	3.61(0.6)	3.51(0.9)	3.41(0.9)	3.41(1.4)	5.27(1.6)
diabetes	23.65(3.7)	23.91(2.9)	24.70(3.7)	23.72(3.3)	23.83(2.5)	23.04(2.7)	**22.96(2.3)**	30.26(5.6)
german	**24.17(1.2)**	24.67(1.3)	24.27(1.2)	24.60(1.1)	25.80(1.4)	24.67(1.3)	24.60(3.1)	29.67(3.2)
liver	27.12(2.0)	**26.15(1.0)**	28.46(1.3)	26.92(2.5)	27.50(2.6)	26.55(2.0)	26.73(4.7)	30.08(5.2)
sonar	13.23(3.5)	**12.58(2.4)**	13.87(2.6)	14.19(4.0)	13.87(4.5)	12.90(4.9)	12.90(6.7)	17.74(6.8)
a2a	17.14(0.9)	**15.20(0.6)**	17.11(0.8)	18.44(1.0)	17.94(1.0)	17.91(1.0)	17.34(1.5)	18.92(1.8)
$n\lambda = 1$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	EP
australian	12.19(1.1)	12.30(1.2)	**11.88(1.3)**	14.30(1.6)	13.30(1.7)	14.43(1.9)	13.43(1.9)	16.29(3.4)
heart	15.80(4.0)	**15.31(4.2)**	16.54(3.8)	18.27(6.7)	17.80(4.3)	14.83(4.8)	14.07(6.3)	20.77(7.3)
ionosphere	**3.81(1.9)**	4.38(1.0)	4.95(1.2)	5.33(1.9)	5.33(1.9)	6.48(2.2)	6.48(2.2)	5.38(4.6)
breast	3.42(0.8)	**3.22(0.6)**	3.32(0.8)	3.61(0.8)	3.51(0.9)	3.32(0.8)	3.32(0.8)	5.56(1.1)
diabetes	23.48(1.8)	23.45(2.2)	24.35(2.1)	23.65(2.4)	23.83(2.3)	24.22(1.9)	**23.22(1.8)**	26.52(0.4)
german	24.60(2.4)	**23.87(0.9)**	24.00(2.3)	25.07(1.1)	23.93(2.3)	24.60(2.4)	24.67(2.4)	25.13(2.1)
liver	28.46(3.3)	**26.54(1.8)**	27.88(2.4)	29.04(4.6)	27.12(2.7)	27.12(2.7)	26.82(3.3)	28.46(3.1)
sonar	13.87(4.5)	**11.55(5.7)**	12.90(5.7)	14.84(6.7)	11.61(5.9)	13.55(6.7)	12.83(6.8)	13.90(7.3)
a2a	**15.51(0.9)**	16.11(1.0)	16.82(0.9)	17.23(1.1)	17.35(1.0)	16.91(0.9)	16.94(0.9)	19.71(1.1)
$n\lambda = 10$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	EP
australian	13.82(1.3)	13.72(1.4)	**12.27(1.4)**	14.59(3.3)	14.98(3.4)	14.01(2.6)	14.01(3.1)	16.54(4.5)
heart	17.78(4.2)	18.02(4.3)	18.30(4.3)	18.27(6.3)	18.27(5.6)	**17.28(5.3)**	17.78(6.2)	19.74(7.3)
ionosphere	**4.38(2.1)**	5.14(1.3)	6.67(2.4)	4.95(3.2)	4.95(3.2)	5.14(2.2)	5.14(2.2)	9.52(4.2)
breast	3.46(0.7)	3.80(0.6)	**3.32(0.9)**	3.51(0.9)	3.80(1.0)	3.75(0.9)	3.41(1.1)	7.02(1.3)
diabetes	**22.30(1.6)**	23.83(2.1)	25.39(2.3)	24.00(2.4)	23.48(2.6)	23.83(2.8)	23.83(2.8)	25.83(3.0)
german	26.33(2.3)	**24.93(1.2)**	25.33(2.1)	26.40(3.1)	26.47(3.2)	26.87(2.8)	26.25(4.2)	28.67(3.4)
liver	**25.27(2.1)**	28.65(2.4)	30.77(2.3)	28.85(2.8)	30.00(2.6)	28.46(2.5)	28.65(2.9)	29.42(3.2)
sonar	13.55(4.3)	**12.23(4.1)**	12.90(4.2)	14.52(5.3)	13.23(5.6)	12.58(6.3)	12.94(6.7)	14.74(7.3)
a2a	17.76(0.8)	17.82(0.8)	**17.41(0.7)**	18.88(0.8)	18.97(0.9)	18.76(1.1)	18.41(1.0)	20.15(1.2)
$n\lambda = 100$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	EP
australian	13.53(1.4)	14.20(1.6)	**12.66(1.5)**	14.30(1.8)	13.53(1.7)	13.91(1.6)	13.41(1.5)	14.54(2.1)
heart	19.26(4.1)	118.26(3.9)	**17.28(4.3)**	19.51(4.2)	19.26(5.6)	19.51(5.8)	19.26(5.3)	22.6(8.3)
ionosphere	8.38(2.4)	8.28(2.1)	**6.48(1.8)**	12.76(2.7)	8.38(2.9)	9.33(2.3)	9.14(3.2)	12.56(3.6)
breast	3.22(0.5)	**3.12(0.7)**	3.51(0.8)	3.92(1.1)	3.22(1.3)	3.41(0.9)	3.21(0.8)	4.98(1.2)
diabetes	29.83(2.1)	29.48(2.6)	**24.87(2.7)**	29.65(3.1)	29.83(3.3)	29.57(3.2)	29.57(3.5)	35.65(4.1)
german	27.40(2.5)	26.51(2.2)	28.20(2.1)	31.21(2.6)	27.40(2.8)	**25.40(3.2)**	29.40(3.0)	31.40(3.8)
liver	**31.05(2.5)**	32.42(2.4)	37.50(2.5)	33.46(3.5)	31.08(3.4)	31.85(3.6)	38.65(4.1)	33.08(4.1)
sonar	**26.06(5.6)**	27.10(5.8)	27.97(5.5)	27.81(7.2)	**26.06(6.3)**	26.77(6.7)	26.77(7.7)	27.10(5.5)
a2a	22.18(1.2)	22.68(1.3)	**18.67(0.9)**	24.68(1.7)	22.18(1.5)	24.68(2.1)	24.31(1.8)	23.21(2.1)

One can see that the performance is stable with respect to $\eta \in [2^{-3}, 2^{1}]$ and $\eta \in [2^{-4}, 2^{-1}]$ on most datasets for CVKS and RKS, respectively. Moreover, it turns out that $\eta \in [2^{-3}, 2^{1}]$ and $\eta \in [2^{-4}, 2^{-1}]$ are good choices for CVKS and RKS, respectively. The robustness property of the parameter η shows that we can randomly select $\eta \in [2^{-3}, 2^{1}]$ and $\eta \in [2^{-4}, 2^{-1}]$ for CVKS and RKS without sacrificing much accuracy.

TABLE III
COMPARISON OF THE TEST MEAN SQUARE ERRORS AMONG OUR KS CRITERIA: 5-CVKS, 10-CVKS, RKS, AND OTHER POPULAR ONES, INCLUDING ELOO, 5-CV, 10-CV, BR, AND SPS. WE BOLD THE BEST METHOD, AND UNDERLINE THE OTHER METHODS THAT ARE NOT SIGNIFICANTLY WORSE THAN THE BEST ONE

$n\lambda = 0.1$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	SPS
bodyfat	**8.46(1.1)e-6**	8.73(1.3)e-6	9.72(1.4)e-6	8.73(1.4)e-6	9.46(1.5)e-6	33.4(4.1)e-6	33.4(7.1)e-6	9.62(1.4)e-6
housing	11.72(2.1)	11.09(2.3)	**9.75(1.9)**	11.75(2.4)	11.72(2.6)	11.34(2.7)	11.34(2.7)	12.25(2.4)
mpg	5.96(0.6)	**5.83(0.5)**	6.64(0.6)	6.90(0.8)	6.31(0.8)	6.96(1.0)	6.96(1.0)	8.81(1.3)
pyrim	1.53(0.2)e-2	1.53(0.3)e-2	2.62(0.7)e-3	2.92(0.6)e-3	2.24(0.5)e-2	2.75(0.8)e-3	2.73(0.9)e-3	**1.01(0.1)e-3**
triazines	**1.43(0.1)e-2**	1.75(0.1)e-2	1.85(0.1)e-2	1.98(0.3)e-2	1.53(0.2)e-2	2.35(0.2)e-2	1.95(0.2)e-2	2.22(0.4)e-2
eunite	372.34(68.0)	**364.72(55.7)**	454.70(65.3)	424.72(73.5)	374.72(65.8)	431.49(66.1)	411.49(74.2)	438.05(112.4)
mg	1.86(0.3)e-2	1.47(0.3)e-2	**1.31(0.2)e-2**	1.68(0.4)e-2	1.38(0.2)e-2	1.46(0.5)e-2	1.37(0.6)e-2	1.52(0.3)e-2
space-ga	1.32(0.3)e-2	**1.35(0.2)e-2**	1.53(0.2)e-2	1.62(0.5)e-2	1.73(0.5)e-2	1.41(0.3)e-2	1.41(0.3)e-2	1.45(0.3)e-2
cpusmall	**9.51(2.3)**	9.61(1.9)	10.17(2.1)	14.32(2.3)	9.56(1.9)	14.74(2.5)	14.74(2.3)	15.54(2.6)
$n\lambda = 1$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	SPS
bodyfat	7.14(0.9)e-6	**7.05(0.8)e-6**	7.45(1.0)e-6	7.31(0.9)e-6	7.13(0.9)e-6	1.47(0.2)e-5	1.47(0.2)e-5	1.36(0.2)e-5
housing	9.24(1.8)	10.45(1.5)	**8.03(1.1)**	10.34(2.1)	11.72(2.6)	9.37(2.3)	9.37(2.3)	22.42(4.5)
mpg	**5.23(0.5)**	5.73(0.6)	6.76(0.7)	7.04(1.1)	6.31(0.9)	7.04(1.3)	7.04(1.3)	9.15(1.5)
pyrim	**2.84(0.3)e-3**	3.63(0.3)e-3	2.99(0.3)e-3	3.20(0.4)e-3	1.55(0.1)e-2	3.23(0.4)e-3	3.34(0.6)e-3	6.23(0.6)e-3
triazines	2.06(0.3)e-2	**1.64e-2(0.2)**	1.91(0.3)e-2	2.07(0.4)e-2	1.75(0.3)e-2	2.10(0.3)e-2	2.08(0.4)e-2	2.22(0.4)e-2
eunite	**363.35(58)**	372.32(68)	470.0(74)	408.8(65)	372.34(72)	408.8(65)	404.0(61)	600.49(73)
mg	**1.12(0.2)e-2**	1.34(0.2)e-2	1.25(0.1)e-2	1.36(0.2)e-2	1.41(0.2)e-2	1.40(0.2)e-2	1.42(0.2)e-2	1.44(0.2)e-2
space-ga	9.24(2.4)e-3	**8.24(2.1)e-3**	1.25(0.1)e-2	1.57(0.1)e-2	1.22(0.1)e-2	1.24(0.1)e-2	1.19(0.1)e-2	1.75(0.2)e-2
cpusmall	**8.25(1.4)**	**8.25(1.4)**	11.00(1.7)	11.14(1.8)	9.48(1.5)	11.14(1.8)	11.14(1.8)	12.09(2.2)
$n\lambda = 10$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	SPS
bodyfat	**9.25(1.1)e-6**	9.62(1.2)e-6	1.32(0.35)e-5	1.32(0.43)e-5	1.11(0.32)e-5	9.55(1.2)e-5	9.55(1.2)e-5	1.15(0.36)e-5
housing	15.02(2.1)	14.51(1.9)	**14.04(1.7)**	15.38(2.4)	14.72(1.8)	15.98(2.3)	15.98(2.3)	18.92(2.8)
mpg	5.96(0.5)	**5.65(0.5)**	6.85(0.6)	7.01(0.8)	7.52(1.0)	7.29(0.9)	7.29(0.9)	7.42(1.0)
pyrim	2.43(0.2)e-3	2.84(0.2)e-3	**2.42(0.2)e-3**	2.69(0.4)e-3	2.68(0.3)e-3	2.62(0.3)e-3	2.59(0.4)e-3	1.59(0.2)e-2
triazines	1.99(0.3)e-2	1.92(0.3)e-2	**1.82(0.2)e-2**	2.05(0.4)e-2	2.01(0.4)e-2	1.93(0.3)e-2	1.89(0.2)e-2	2.08(0.4)e-2
eunite	425.52(67)	**413.72(58)**	471.12(73)	473.80(85)	442.8(74)	494.3(94)	494.3(94)	792.84(124)
mg	1.53(0.3)e-2	1.50(0.3)e-2	**1.36(0.3)e-2**	1.45(0.3)e-2	1.32(0.2)e-2	1.44(0.4)e-2	1.72(0.4)e-2	2.11(0.5)e-2
space-ga	**9.67(2.6)e-3**	1.02(0.2)e-2	1.72(0.3)e-2	1.65(0.4)e-2	1.83(0.4)e-2	1.82(0.4)e-2	1.84(0.5)e-2	3.65(0.8)e-2
cpusmall	**19.06(3.2)**	19.61(2.9)	20.52(2.8)	21.28(3.2)	20.14(2.8)	21.28(3.3)	21.28(3.3)	24.52(3.5)
$n\lambda = 100$								
Method	5-CVKS	10-CVKS	RKS	5-CV	10-CV	ELOO	BR	SPS
bodyfat	**2.51(0.3)e-5**	3.23(0.4)e-5	3.01(0.4)e-5	2.82(0.3)e-5	3.11(0.5)e-5	1.94(0.2)e-4	1.94(0.2)e-4	2.04(0.2)e-4
housing	21.52(2.8)	**20.41(2.4)**	21.58(2.7)	23.58(3.1)	22.14(3.0)	23.10(3.3)	23.10(3.3)	28.33(4.2)
mpg	8.13(1.2)	**7.84(0.9)**	8.15(1.3)	8.52(1.5)	9.92(1.8)	9.35(1.6)	9.35(1.6)	13.92(2.1)
pyrim	1.40(0.1)e-3	**1.26(0.0)e-3**	2.51(0.2)e-3	2.51(0.2)e-3	1.98(0.1)e-3	2.75(0.3)e-3	2.74(0.4)e-3	1.25(0.1)e-2
triazines	**1.89(0.3)e-2**	1.90(0.3)e-2	1.95(0.3)e-2	2.02(0.4)e-2	2.46(0.6)e-2	2.02(0.4)e-2	2.05(0.4)e-2	2.48(0.7)e-2
eunite	605.23(123)	601.35(103)	623.55(112)	613.01(132)	604.23(143)	**592.51(128)**	892.59(153)	932.24(174)
mg	1.73(0.4)e-2	1.51(0.3)e-2	**1.24(0.2)e-2**	1.64(0.3)e-2	1.52(0.2)e-2	1.52(0.2)e-2	1.52(0.2)e-2	1.52(0.2)e-2
space-ga	**1.20(0.2)e-2**	**1.20(0.2)e-2**	1.94(0.3)e-2	2.11(0.4)e-2	2.69(0.4)e-2	2.05(0.4)e-2	2.07(0.5)e-2	2.23e-2(0.6)
cpusmall	36.73(6.4)	**34.73(6.7)**	41.34(7.3)	46.62(7.8)	42.56(7.4)	56.3(8.4)	46.62(7.7)	38.34(6.8)

B. Regression

The test mean square errors reported in Table III. From this table we can find the following.

1) CVKS gains better accuracy results than CV. In particular, CVKS significantly outperforms KS on 4 (or more) out of 9 sets for each λ without being significantly worse on the other datasets.
2) CVKS and RKS are significant better than SPS on almost all datasets.
3) On most datasets, CVKS outperforms CV, LOO, and BR. For each λ, KS is significant better than LOO and

BR on 3 (or more) out of 9 sets, and outperforms CV on 4 out of 9 sets.

4) RKS is comparable to CVKS.

The above results demonstrate that KS-based criteria are good choices for model selection.

VIII. Discussion

In this part, we will discuss the relationship between this paper with the existing work.

A. Stability

The notion of stability is an important tool for studying the generalization performance of learning machines [54]. The uniform stability [29] is the most popular notion of stability, and the generalization bound for kernel methods is established. However, the uniform stability is defined on the learning machine, so it is difficult to estimate its value (see Remark 1). Unlike the uniform stability, KS is defined on the kernel matrix, and it is easy to compute its value according to empirical data (see in Theorem 7). Moreover, we show the relationship between KS and uniform stability for SVM, KRR, and LSSVM, so we can also use KS to establish exponential generalization bounds.

In our previous work [44], we give a notion of SPS that defined on the eigenvalues of the kernel matrix. Although the SPS is not defined on learning machines, the estimation of this stability is also difficult, and the generalization error bounds for LSSVM and SVM are not established. Moreover, the experimental results also imply that the proposed criteria are better than the SPS-based criterion (see in Tables II and III).

B. Cross-Validation

The CV is the most popular model selection method, but it usually exhibits a high variance. Cawley and Talbot [15] investigated the use of BR to model selection to ameliorate the high variance of leave-one-out CV for LSSVM. However, this method is only usable for leave-one-out CV on LSSVM, and the relationship between BR and the variance of leave-one-out CV is not established. Kumar *et al.* [43] introduced a new stability of learning machine called loss stability, and showed that this notion serves as an additive factor to the optimal variance reduction obtained by CV. Different from the above works, in this paper, we derive variance bounds of CV with KS for model selection. Furthermore, our method can be used for general k-fold CV on several kernel-based machines.

IX. Conclusion

We developed a novel stability-based model selection method via a newly defined stability (KS) from a new perspective of the kernel matrix. We manifested that the generalization error and variance of CV of SVM, LSSVM, and KRR can be bounded with KS, so we can apply this stability for model selection to guarantee the generalization performance. Our model selection criteria are theoretically justified and experimentally validated.

In our future work, we will extend our method to other kernel based methods, and using some approximate methods to reduce the computational complexity of our proposed criteria for large-scale data.

Appendix A
Proof of Theorem 1

We first give the following theorem to prove Theorem 1.

Theorem 8: If K is a β-KS kernel, then for the KRR and LSSVM, $\forall S \in \mathcal{Z}^n, \forall i \in \{1, \ldots, n\}, \forall \mathbf{x} \in \mathcal{X}$, we have

$$\left|f_S(\mathbf{x}) - f_{S^i}(\mathbf{x})\right| \leq \frac{C_1\beta + C_2}{n-1}$$

where $C_1 = (\kappa M/\lambda^2)$ and $C_2 = (2\kappa M/\lambda)$.

Proof: Note that the solutions of KRR (or LSSVM) on S and S^i can be written as

$$f_S(\mathbf{x}) = \mathbf{k}^T(\mathbf{K} + n\lambda\mathbf{I})^{-1}\mathbf{y}$$
$$f_{S^i}(\mathbf{x}) = \mathbf{k}_i^T(\mathbf{K}_i + (n-1)\lambda\mathbf{I}_i)^{-1}\mathbf{y}_i$$

where

$$\mathbf{k} = (K(\mathbf{x}, \mathbf{x}_1), K(\mathbf{x}, \mathbf{x}_2), \ldots, K(\mathbf{x}, \mathbf{x}_n))^T$$
$$\mathbf{k}_i = (K(\mathbf{x}, \mathbf{x}_1), \ldots, K(\mathbf{x}, \mathbf{x}_{i-1}), K(\mathbf{x}, \mathbf{x}_{i+1}), \ldots, K(\mathbf{x}, \mathbf{x}_n))^T$$
$$\mathbf{y}_i = (y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)^T$$
$$\mathbf{K}_i = \left[K\left(\mathbf{x}_j, \mathbf{x}_k\right)\right]_{j,k}, \mathbf{x}_j, \mathbf{x}_k \in S^i$$
$$\mathbf{I}_i \text{ is the } (n-1) \times (n-1) \text{ identity matrix.}$$

Let $\mathbf{H}^i = \mathbf{K}^i + (n-1)\lambda\mathbf{I}$, $\mathbf{H}_i = \mathbf{K}_i + (n-1)\lambda\mathbf{I}_i$, $\mathbf{H} = \mathbf{K} + n\lambda\mathbf{I}$, $\mathbf{K}^i$ is the ith removed matrix defined in (4). Without loss of generality, we assume $i = n$. Taking into account the block matrix inversion formula, we have

$$\mathbf{H}^{i^{-1}} = \begin{bmatrix} \mathbf{H}_i^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \mathbf{A}_i \tag{7}$$

where $\mathbf{A}_i$ is the diagonal matrix, the ith diagonal element is $(1/(n-1)\lambda)$, others 0. So the f_{S^i} can be represented as

$$f_{S^i}(\mathbf{x}) = \mathbf{k}_i^T\mathbf{H}_i^{-1}\mathbf{y}_i = \mathbf{k}^T\left(\mathbf{H}^{i^{-1}} - \mathbf{A}_i\right)\mathbf{y}.$$

Therefore, we can obtain that

$$\begin{aligned} f_S(\mathbf{x}) - f_{S^i}(\mathbf{x}) &= \mathbf{k}^T\mathbf{H}^{-1}\mathbf{y} - \mathbf{k}^T\mathbf{H}^{i^{-1}}\mathbf{y} + \mathbf{k}^T\mathbf{A}_i\mathbf{y} \\ &= \mathbf{k}^T\left(\mathbf{H}^{-1} - \mathbf{H}^{i^{-1}}\right)\mathbf{y} + \mathbf{k}^T\mathbf{A}_i\mathbf{y} \end{aligned} \tag{8}$$

where $\mathbf{H} = \mathbf{K} + n\lambda\mathbf{I}$.

For any invertible matrices $\mathbf{M}$, $\mathbf{M}'$, $\mathbf{M}'^{-1} - \mathbf{M}^{-1} = -\mathbf{M}'^{-1}(\mathbf{M}' - \mathbf{M})\mathbf{M}^{-1}$ is valid, so we can obtain that

$$\mathbf{H}^{-1} - \mathbf{H}^{i^{-1}} = -\mathbf{H}^{-1}\left(\mathbf{K} - \mathbf{K}^i + \lambda\mathbf{I}\right)\mathbf{H}^{i^{-1}}. \tag{9}$$

From (9), it is easy to verity that

$$\begin{aligned} &\left\|\mathbf{k}^T\left(\mathbf{H}^{-1} - \mathbf{H}^{i^{-1}}\right)\mathbf{y}\right\| \\ &\leq \left\|\mathbf{k}^T\right\|\left\|\mathbf{H}^{-1}\right\|_2\left\|\mathbf{K} - \mathbf{K}^i + \lambda\mathbf{I}\right\|_2\left\|\mathbf{H}^{i^{-1}}\right\|_2\|\mathbf{y}\| \\ &= \frac{\left\|\mathbf{k}^T\right\|\left\|\mathbf{K} - \mathbf{K}^i + \lambda\mathbf{I}\right\|_2\|\mathbf{y}\|}{\lambda_{\min}(\mathbf{H}) \cdot \lambda_{\min}\left(\mathbf{H}^i\right)} \end{aligned} \tag{10}$$

where $\lambda_{\min}(\mathbf{H})$ and $\lambda_{\min}(\mathbf{H}^i)$ are the smallest eigenvalue of $\mathbf{H}$ and $\mathbf{H}^i$, respectively. Since $\mathbf{K}$ and $\mathbf{K}^i$ are positive semidefinite, we have

$$\lambda_{\min}(\mathbf{H}) \geq n\lambda, \lambda_{\min}\left(\mathbf{H}^i\right) \geq (n-1)\lambda. \tag{11}$$

According to (8), (10), and (11), we have

$$\begin{aligned}&\left|f_S(\mathbf{x})-f_{S^i}(\mathbf{x})\right|\\&\le\frac{\left\|\mathbf{k}^T\right\|\left(\left\|\mathbf{K}-\mathbf{K}^i\right\|_2+\|\lambda\mathbf{I}\|_2\right)\|\mathbf{y}\|}{n(n-1)\lambda^2}+\left|\mathbf{k}^T\mathbf{A}_i\mathbf{y}\right|\\&=\frac{\left\|\mathbf{k}^T\right\|\left(\left\|\mathbf{K}-\mathbf{K}^i\right\|_2+\lambda\right)\|\mathbf{y}\|}{n(n-1)\lambda^2}+\frac{|K(\mathbf{x}_i,\mathbf{x}_i)y_i|}{\lambda(n-1)}.\end{aligned}$$

Taking into account that $\|\mathbf{y}\|\le\sqrt{n}M$ and $\|\mathbf{k}\|\le\sqrt{n}\kappa$, we have

$$\left|f_S(\mathbf{x})-f_{S^i}(\mathbf{x})\right|\le\frac{\kappa M}{(n-1)\lambda^2}\left\|\mathbf{K}-\mathbf{K}^i\right\|_2+\frac{2\kappa M}{(n-1)\lambda}. \quad (12)$$

By the definition of β-KS (see Definition 2 for detail), we know that

$$\left\|\mathbf{K}-\mathbf{K}^i\right\|_2\le\beta. \quad (13)$$

Plugging (13) to (12), we have

$$\left|f_S(\mathbf{x})-f_{S^i}(\mathbf{x})\right|\le\frac{\kappa M\beta}{(n-1)\lambda^2}+\frac{2\kappa M}{(n-1)\lambda}.$$

■

Proof of Theorem 1: One can see that

$$\begin{aligned}|f_S(\mathbf{x})|&\le\|\mathbf{k}\|\left\|\mathbf{H}^{-1}\right\|_2\|\mathbf{y}\|\le\frac{n\kappa M}{\lambda_{\min}(\mathbf{H})}\le\frac{\kappa M}{\lambda}\\\left|f_{S^i}(\mathbf{x})\right|&\le\|\mathbf{k}_i\|\left\|\mathbf{H}_i^{-1}\right\|_2\|\mathbf{y}\|\le\frac{(n-1)\kappa M}{\lambda_{\min}(\mathbf{H}_i)}\le\frac{\kappa M}{\lambda}. \quad (14)\end{aligned}$$

So, according to Theorem 8 and (14), we have

$$\begin{aligned}&\left|\ell(f_S(\mathbf{x}),y)-\ell\left(f_{S^i}(\mathbf{x}),y\right)\right|\\&=\left|(y-f_S(\mathbf{x}))^2-\left(y-f_{S^i}(\mathbf{x})\right)^2\right|\\&=\left|f_S(\mathbf{x})-f_{S^i}(\mathbf{x})\right|\cdot\left|2y-f_S(\mathbf{x})-f_{S^i}(\mathbf{x})\right|\\&\le\frac{C_1\beta+C_2}{n-1}(2M+2\kappa M/\lambda). \quad (15)\end{aligned}$$

The above equation shows that the KRR (or LSSVM) with β-KS is $[(C_1\beta+C_2)/n-1](2M+2\kappa M/\lambda)$ uniform stability. ■

Appendix B
Proof of Theorem 2

Note that $f_S(\mathbf{x})$ and $f_{S^i}(\mathbf{x})$ are the hypothesis returned by SVM with $\mathbf{K}$ and $\mathbf{K}^i$, respectively. Thus, according to [32, Proposition 2]

$$\begin{aligned}&\left|f_S(\mathbf{x})-f_{S^i}(\mathbf{x})\right|\\&\le\frac{\kappa^{\frac{3}{4}}}{\sqrt{2}\lambda}\left\|\mathbf{K}^i-\mathbf{K}\right\|_2^{\frac{1}{4}}\left[1+\left[\frac{\left\|\mathbf{K}^i-\mathbf{K}\right\|_2}{4\kappa}\right]^{\frac{1}{4}}\right]\\&\le\frac{\kappa^{\frac{3}{4}}}{\sqrt{2}\lambda}\beta^{\frac{1}{4}}\left[1+\left[\frac{\beta}{4\kappa}\right]^{\frac{1}{4}}\right].\end{aligned}$$

Since the hinge loss ℓ is 1-Lipschitz, we have

$$\left|\ell(f_S(\mathbf{x}),y)-\ell\left(f_{S^i}(\mathbf{x}),y\right)\right|\le\frac{\kappa^{\frac{3}{4}}}{\sqrt{2}\lambda}\beta^{\frac{1}{4}}\left[1+\left[\frac{\beta}{4\kappa}\right]^{\frac{1}{4}}\right]. \quad (16)$$

Thus, the SVM with β-KS is $[(\kappa^{(3/4)})/(\sqrt{2}\lambda)]\beta^{(1/4)}[1+[\beta/4\kappa]^{(1/4)}]$ uniform stability.

Appendix C
Proof of Theorem 3

Note that

$$\begin{aligned}\ell(f_S(\mathbf{x}),y)&=(f_S(\mathbf{x})-y)^2\\&\le2f_S^2(\mathbf{x})+2|y|^2\le\frac{2\kappa^2M^2}{\lambda^2}+2M^2. \quad (17)\end{aligned}$$

Thus, according to [29, Th. 11] with

$$\gamma=\frac{(C_1\beta+C_2)(2M+2\kappa M/\lambda)}{n-1},\ Q=\frac{2\kappa^2M^2}{\lambda^2}+2M^2$$

we have

$$\begin{aligned}R(S)\le R_{\text{emp}}(S)&+\frac{C_3(C_1\beta+C_2)}{n-1}\\&+\left(\frac{2C_3n(C_1\beta+C_2)}{n-1}+Q\right)\sqrt{\frac{\ln1/\delta}{2n}}\end{aligned}$$

where $C_3=4M+(4\kappa M/\lambda)$ and $Q=[(2\kappa^2M^2)/\lambda^2]+2M^2$.

Appendix D
Proof of Theorem 4

Taking into account that

$$f_S(\mathbf{x})=\sum_{j=1}^{n}\alpha_jK\left(\mathbf{x}_j,\mathbf{x}\right),0\le\alpha_i\le\frac{1}{\lambda n}.$$

Therefore,

$$\ell(f_S(\mathbf{x}),y)=\max(0,1-yf_S(\mathbf{x}))\le|1-yf_S(\mathbf{x})|\le1+\lambda\kappa.$$

Using [29, Th. 11] with $\gamma=[(\kappa^{(3/4)})/(\sqrt{2}\lambda)]\beta^{(1/4)}[1+[\beta/4\kappa]^{(1/4)}]$ and $Q=1+\lambda\kappa$, the proof is completed. ■

Appendix E
Proof of Theorem 5

Let $T=\{\mathbf{x}_i\}_{i=1}^m$ and $m=((k-1)n/k)$. The (empirical) loss of the hypothesis f_T on a set Q is defined as

$$\ell_{f_T}(Q)=\frac{1}{|Q|}\sum_{z\in Q}\ell(f_T(\boldsymbol{x}),y).$$

Let $\mathbf{K}_T$ be the kernel matrix respect to dataset T with elements $[\mathbf{K}_T]_{i,j}=K(\mathbf{x}_i,\mathbf{x}_j)$, $\mathbf{x}_i,\ \mathbf{x}_j\in T$. $\mathbf{K}_T^i$ is the $m\times m$ ith removed kernel matrix with

$$\begin{cases}\left[\mathbf{K}_T^i\right]_{jk}=K\left(\mathbf{x}_j,\mathbf{x}_k\right) & \text{if } j\ne i \text{ and } k\ne i\\\left[\mathbf{K}_T^i\right]_{jk}=0 & \text{if } j=i \text{ or } k=i.\end{cases}$$

Definition 3 (Loss Stability [43]): The loss stability of a learning algorithm f trained on m examples and with respect to a loss ℓ is defined as

$$\text{ls}_{m,\ell}(f)=\mathbb{E}_{T:|T|=m,z',z}\left[\left(\ell(f_T(\mathbf{x}),y)-\ell'\left(f_{T^{z'}}(\mathbf{x}),y\right)\right)^2\right]$$

where $\ell'(f_T(\mathbf{x}),y)=\ell(f_T(\mathbf{x}),y)-\mathbb{E}_z[\ell(f_T(\mathbf{x}),y)]$, $T^{z'}$ denote the set of examples obtained by replacing an example chosen uniformly at random from T by z'. A learning algorithm f is γ-loss stable if $\text{ls}_{m,\ell}(f)\le\gamma$.

Proof of Theorem 5: From Theorem 7, we know that

$$\left\|\mathbf{K}_T - \mathbf{K}_T^i\right\|_2 = \frac{\mathbf{K}_{ii} + \sqrt{\mathbf{K}_{ii}^2 + 4\sum_{j=1,j\neq i}^{m} \mathbf{K}_{ji}^2}}{2}$$

$$\left\|\mathbf{K} - \mathbf{K}^i\right\|_2 = \frac{\mathbf{K}_{ii} + \sqrt{\mathbf{K}_{ii}^2 + 4\sum_{j=1,j\neq i}^{n} \mathbf{K}_{ji}^2}}{2}.$$

Taking into account that T is a subset of S

$$\left\|\mathbf{K}_T - \mathbf{K}_T^i\right\|_2 \leq \left\|\mathbf{K} - \mathbf{K}^i\right\|_2 \leq \beta.$$

From (15), one can see that

$$\left|\ell(f_T(\mathbf{x}), y) - \ell\big(f_{T^i}(\mathbf{x}), y\big)\right| \leq \frac{C_1\beta + C_2}{m-1}(2M + 2\kappa M/\lambda).$$

According to [43, Lemma 2], we know that

$$\mathrm{ls}_{m,\ell}(f) \leq \mathbb{E}_{T:|T|=m,z',z}\left[\left(\ell\big(f_T(\mathbf{x}, y) - \ell\big(f_{T^{z'}}(\mathbf{x}), y\big)\big)^2\right)\right].$$

Thus,

$$\mathrm{ls}_{m,\ell}(f) \leq \left(\frac{C_1\beta + C_2}{m-1}\left(2M + \frac{2\kappa M}{\lambda}\right)\right)^2 = \gamma. \tag{18}$$

From [43, Lemma 5], it is easy to verify that

$$\begin{aligned}
&\operatorname*{var}_S\Big(\ell_{f_{S\setminus S_1}}(S_1)\Big) \\
&= \operatorname*{cov}_S\Big(\ell_{f_{S\setminus S_1}}(S_1), \ell_{f_{S\setminus S_1}}(S_1)\Big) \\
&= \mathbb{E}_{S\setminus S_1, z_1', z_2}\left[\left(\ell'_{f_{S\setminus S_1}}(z_2) - \ell'_{f_{(S\setminus S_1)^{z_1'}}}(z_2)\right)^2\right] \\
&= \mathrm{ls}_{m,\ell}(f) \\
&\leq \left(\frac{C_1\beta + C_2}{m-1}\left(2M + \frac{2\kappa M}{\lambda}\right)\right)^2 \\
&= \gamma\big[\text{According to (18)}\big].
\end{aligned} \tag{19}$$

Substituting (18) and (19) into [43, Th. 1]

$$\operatorname*{var}_S\big(\ell_{f_{\mathrm{kcv}}}(S)\big) \leq \frac{1}{k}\gamma + \left(1 - \frac{1}{k}\right)\gamma = \gamma.$$

■

APPENDIX F
PROOF OF THEOREM 6

From (16), one can see that

$$\left|\ell(f_S(\mathbf{x}), y) - \ell\big(f_{S^i}(\mathbf{x}), y\big)\right| \leq \frac{\kappa^{\frac{3}{4}}}{\sqrt{2\lambda}}\beta^{\frac{1}{4}}\left[1 + \left[\frac{\beta}{4\kappa}\right]^{\frac{1}{4}}\right].$$

Unlike the proofs of (18) and (19), we have

$$\mathrm{ls}_{m,\ell}(f_T) \leq \left(\frac{\kappa^{\frac{3}{4}}}{\sqrt{2\lambda}}\beta^{\frac{1}{4}}\left[1 + \left[\frac{\beta}{4\kappa}\right]^{\frac{1}{4}}\right]\right)^2 = \gamma$$

and

$$\operatorname*{var}_S\Big(\ell_{f_{S\setminus S_1}}(S_1)\Big) \leq \left(\frac{\kappa^{\frac{3}{4}}}{\sqrt{2\lambda}}\beta^{\frac{1}{4}}\left[1 + \left[\frac{\beta}{4\kappa}\right]^{\frac{1}{4}}\right]\right)^2 = \gamma.$$

Substituting the above two equations into [43, Th. 1], we complete the proof. ■

REFERENCES

[1] V. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag, 2000.

[2] D. Anguita, A. Ghio, L. Oneto, and S. Ridella, "The impact of unlabeled patterns in Rademacher complexity theory for kernel classifiers," in *Proc. Adv. Neural Inf. Process. Syst. 24 (NIPS)*, Granada, Spain, 2011, pp. 585–593.

[3] X. Liu, L. Wang, J. Yin, E. Zhu, and J. Zhang, "An efficient approach to integrating radius information into multiple kernel learning," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 557–569, Apr. 2013.

[4] X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1083–1095, Jun. 2014.

[5] C. Saunders, A. Gammerman, and V. Vovk, "Ridge regression learning algorithm in dual variables," in *Proc. 15th Int. Conf. Mach. Learn. (ICML)*, 1998, pp. 515–521.

[6] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, no. 3, pp. 293–300, 1999.

[7] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 131–159, 2002.

[8] L. Ding and S. Liao, "Model selection with the covering number of the ball of RKHS," in *Proc. 23rd ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, Shanghai, China, 2014, pp. 1159–1168.

[9] L. Ding and S. Liao, "An approximate approach to automatic kernel selection," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 554–565, Mar. 2017.

[10] P. L. Bartlett and S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results," *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, Nov. 2002.

[11] P. L. Bartlett, S. Boucheron, and G. Lugosi, "Model selection and error estimation," *Mach. Learn.*, vol. 48, nos. 1–3, pp. 85–113, 2002.

[12] Y. Liu, S. Jiang, and S. Liao, "Eigenvalues perturbation of integral operator for kernel selection," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, San Francisco, CA, USA, 2013, pp. 2189–2198.

[13] O. Chapelle and V. Vapnik, "Model selection for support vector machines," in *Proc. Adv. Neural Inf. Process. Syst. 12 (NIPS)*, 1999, pp. 230–236.

[14] U. V. Luxburg, O. Bousquet, and B. Schölkopf, "A compression approach to support vector model selection," *J. Mach. Learn. Res.*, vol. 5, pp. 293–323, Jan. 2004.

[15] G. C. Cawley and N. L. C. Talbot, "Preventing over-fitting during model selection via Bayesian regularisation of the hyper-parameters," *J. Mach. Learn. Res.*, vol. 8, pp. 841–861, Apr. 2007.

[16] M. Debruyne, M. Hubert, and J. A. Suykens, "Model selection in kernel based regression using the influence function," *J. Mach. Learn. Res.*, vol. 9, pp. 2377–2400, Oct. 2008.

[17] C. Cortes, M. Kloft, and M. Mohri, "Learning kernels using local Rademacher complexity," in *Proc. Adv. Neural Inf. Process. Syst. 25 (NIPS)*, 2013, pp. 2760–2768.

[18] Y. Liu and S. Liao, "Eigenvalues ratio for kernel selection of kernel methods," in *Proc. 29th AAAI Conf. Artif. Intell. (AAAI)*, 2015, pp. 2814–2820.

[19] Y. Liu, S. Liao, H. Lin, Y. Yue, and W. Wang, "Infinite kernel learning: Generalization bounds and algorithms," in *Proc. 31st AAAI Conf. Artif. Intell. (AAAI)*, 2017, pp. 2280–2286.

[20] J. Li, Y. Liu, R. Yin, H. Zhang, L. Ding, and W. Wang, "Multi-class learning: From theory to algorithm," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, 2018, pp. 1593–1602.

[21] J. Li, Y. Liu, H. Lin, Y. Yue, and W. Wang, "Efficient kernel selection via spectral analysis," in *Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2017, pp. 2124–2130.

[22] Y. Liu, S. Liao, S. Jiang, L. Ding, H. Lin, and W. Wang, "Fast cross-validation for kernel-based algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published. doi: 10.1109/TPAMI.2019.2892371.

[23] Y. Liu, H. Lin, L. Ding, W. Wan, and S. Liao, "Fast cross-validation," in *Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2018, pp. 2497–2503.

[24] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. 14th Int. Conf. Artif. Intell. (IJCAI)*, Montreal, QC, Canada, 1995, pp. 1137–1143.

[25] A. Y. Ng, "Preventing 'overfitting' of cross-validation data," in *Proc. 14th Int. Conf. Mach. Learn. (ICML)*, 1997, pp. 245–253.

[26] G. C. Cawley and N. L. C. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *J. Mach. Learn. Res.*, vol. 11, pp. 2079–2107, Mar. 2010.

[27] L. Devroye and T. Wagner, "Distribution-free performance bounds for potential function rules," *IEEE Trans. Inf. Theory*, vol. 25, no. 5, pp. 601–604, Sep. 1979.
[28] M. Kearns and D. Ron, "Algorithmic stability and sanity-check bounds for leave-one-out cross-validation," *Neural Comput.*, vol. 11, no. 6, pp. 1427–1453, Aug. 1999.
[29] O. Bousquet and A. Elisseeff, "Stability and generalization," *J. Mach. Learn. Res.*, vol. 2, pp. 499–526, Mar. 2002.
[30] T. Poggio, R. Rifkin, S. Mukherjee, and P. Niyogi, "General conditions for predictivity in learning theory," *Nature*, vol. 428, no. 6981, pp. 419–422, 2004.
[31] C. Cortes, M. Mohri, D. Pechyony, and A. Rastogi, "Stability of transductive regression algorithms," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, Helsinki, Finland, 2008, pp. 176–183.
[32] C. Cortes, M. Mohri, and A. Talwalkar, "On the impact of kernel approximation on learning accuracy," in *Proc. 13th Int. Conf. Artif. Intell. Stat. (AISTATS)*, 2010, pp. 113–120.
[33] C. Cortes, M. Mohri, and A. Talwalkar, "Algorithms for learning kernels based on centered alignment," *J. Mach. Learn. Res.*, vol. 13, pp. 795–828, Mar. 2012.
[34] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan, "Learnability, stability and uniform convergence," *J. Mach. Learn. Res.*, vol. 11, pp. 2635–2670, Mar. 2010.
[35] W. Gao and Z. Zhou, "Uniform convergence, stability and learnability for ranking problems," in *Proc. 23rd Int. Joint Conf. Artif. Intell. (IJCAI)*, Beijing, China, 2013, pp. 1337–1343.
[36] C. H. Nguyen and T. B. Ho, "Kernel matrix evaluation," in *Proc. 20th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2007, pp. 987–992.
[37] W. H. Rogers and T. J. Wagner, "A finite sample distribution-free performance bound for local discrimination rules," *Ann. Stat.*, vol. 6, no. 3, pp. 506–514, 1978.
[38] Y. Liu and S. Liao, "Preventing over-fitting of cross-validation with kernel stability," in *Proc. Eur. Conf. Mach. Learn. Principles Pract. Knowl. Disc. Databases (ECML)*, 2014, pp. 290–305.
[39] Y. Bengio and Y. Grandvalet, "No unbiased estimator of the variance of *k*-fold cross-validation," *J. Mach. Learn. Res.*, vol. 5, pp. 1089–1105, Jan. 2004.
[40] K. Geras and C. Sutton, "Multiple-source cross-validation," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*, 2013, pp. 1292–1300.
[41] A. Blum, A. Kalai, and J. Langford, "Beating the hold-out: Bounds for *k*-fold and progressive cross-validation," in *Proc. 12th Annu. Conf. Comput. Learn. Theory (COLT)*, Santa Cruz, CA, USA, 1999, pp. 203–208.
[42] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. Washington, DC, USA: Winston, 1977.
[43] R. Kumar, D. Lokshtanov, S. Vassilvitskii, and A. Vattani, "Near-optimal bounds for cross-validation via loss stability," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*, Atlanta, GA, USA, 2013, pp. 27–35.
[44] Y. Liu and S. Liao, "Kernel selection with spectral perturbation stability of kernel matrix," *Sci. China Inf. Sci.*, vol. 57, no. 11, pp. 1–10, 2014.
[45] T. Evgeniou, M. Pontil, and T. Poggio, "Regularization networks and support vector machines," *Adv. Comput. Math.*, vol. 13, no. 1, pp. 1–50, 2000.
[46] A. Smola, A. Gretton, L. Song, and B. Schölkopf, "A Hilbert space embedding for distributions," in *Proc. 18th Int. Conf. Algorithmic Learn. Theory (COLT)*, 2007, pp. 13–31.
[47] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, pp. 723–773, Mar. 2012.
[48] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[49] Y. Liu, S. Jiang, and S. Liao, "Efficient approximation of cross-validation for kernel methods using Bouligand influence function," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, Beijing, China, 2014, pp. 324–332.
[50] A. Christmann and A. V. Messem, "Bouligand derivatives and robustness of support vector machines for regression," *J. Mach. Learn. Res.*, vol. 9, pp. 915–936, Jun. 2008.
[51] P. Drineas and M. W. Mahoney, "On the Nyström method for approximating a gram matrix for improved kernel-based learning," *J. Mach. Learn. Res.*, vol. 6, pp. 2153–2175, Dec. 2005.
[52] S. Wang and Z. Zhang, "Improving CUR matrix decomposition and the Nyström approximation via adaptive sampling," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 2729–2769, 2013.
[53] G. C. Cawley, "Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Vancouver, BC, Canada, 2006, pp. 1661–1668.
[54] T. Evgeniou, M. Pontil, and A. Elisseeff, "Leave one out error, stability, and generalization of voting combinations of classifiers," *Mach. Learn.*, vol. 55, no. 1, pp. 71–97, 2004.

Yong Liu was born in 1986. He received the Ph.D. degree in computer science from Tianjin University, Tianjin, China, in 2016.

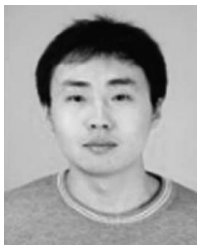
He is currently an Associate Researcher with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His current research interests include large-scale kernel methods, large-scale model selection, AutoML, and machine learning.

Shizhong Liao received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 1997.

He is currently a Professor with the College of Intelligence and Computing, Tianjin University, Tianjin, China. His current research interests include artificial intelligence and theoretical computer science.

Hua Zhang received the Ph.D. degree in computer science from the School of Computer Science and Technology, Tianjin University, Tianjin, China, in 2015.

He is an Associate Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His current research interests include computer vision, multimedia, and machine learning.



Wenqi Ren received the Ph.D. degree from Tianjin University, Tianjin, China, in 2017.

He is an Assistant Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. He was supported by China Scholarship Council as a joint-training Ph.D. student with the Electrical Engineering and Computer Science Department, University of California at Merced, Merced, CA, USA, from 2015 to 2016. His current research interests include image processing and related high-level vision problems.

Weiping Wang received the Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2008.

He is currently a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, National Engineering Research Center for Information Security, Beijing, China, and the National Engineering Laboratory for Information Security Technology, Beijing. His current research interests include database and storage systems.