Simplex-based Proximal Multicategory Support Vector Machine

Fu, Sheng; Chen, Piao; Ye, Zhisheng

DOI
10.1109/TIT.2022.3222266

Publication date
2023

Document Version
Final published version

Published in
IEEE Transactions on Information Theory

Simplex-Based Proximal Multicategory Support Vector Machine

Sheng Fu, Piao Chen, and Zhisheng Ye, *Senior Member, IEEE*

***Abstract*— The multicategory support vector machine (MSVM) has been widely used for multicategory classification. Despite its widespread popularity, regular MSVM cannot provide direct probabilistic results and suffers from excessive computational cost, as it is formulated on the hinge loss function and it solves a sum-to-zero constrained quadratic programming problem. In this study, we propose a general refinement of regular MSVM, termed as the simplex-based proximal MSVM (SPMSVM). Our SPMSVM uses a novel family of squared error loss functions in place of the hinge loss and it removes the explicit sum-to-zero constraint by the simplex structure. Consequently, the SPMSVM only requires solving an unconstrained linear system, leading to closed-form solutions. In addition, the SPMSVM can be cast into a weighted regression problem so that it is scalable for large-scale applications. Moreover, the SPMSVM naturally yields an estimate of the conditional category probability, which is more informative than regular MSVM. Theoretically, the SPMSVM is shown to include many existing MSVMs as its special cases, and its asymptotic and finite-sample statistical properties are well established. Simulations and real examples show that the proposed SPMSVM is a stable, scalable and competitive classifier.**

***Index Terms*— Category probability, fisher consistency, kernel learning, multicategory classification, universal consistency.**

I. Introduction

CLASSIFICATION is an ubiquitous problem in many statistical applications [1], [2]. Given a training dataset with subjects having both covariates and class labels, the learning task is to develop a classification rule to predict the label for a future sample based on its input. Among various classification methods, support vector machine (SVM) and deep neural network (DNN) have become the popular ones during the last decades [3], [4], [5]. The performance of DNN and SVM in classification depends heavily on the quantity and quality of the training data. Generally, if the training data is complex and has a large size, DNN often performs better than SVM in prediction but is also more computationally intensive. On the other hand, SVM is preferred when the data size is moderate or when some fundamental properties of the training data are well understood so that a domain-specific kernel could be specified, e.g., [6], [7], and [8]. Moreover, SVM enjoys several main advantages, such as the regularization parameter to control over-fitting, global solution via convex optimization and substantial theoretical foundations. This study focuses on the development of the SVM-based methods.

The original SVM is a typical binary classifier and it aims to find a hyperplane in the feature space with maximum separation between the two classes [9], [10], in which hinge loss function is used. The classical SVM has been extended by researchers in last decades, such as LSSVM [11], ψ-learning of SVM [12], L2-SVM [13], ramp-SVM [14] and pin-SVM [15]. Although binary SVMs have many successful applications, multicategory problems are commonly seen in practice [16]. A multitude of multicategory extensions to SVMs have been proposed in the literature. One natural idea is to use the binary SVMs sequentially, which are known as the one-versus-rest and one-versus-one approaches [17], [18]. Although these procedures are conceptually straightforward to implement, they do not necessarily yield the optimal prediction in terms of the classification accuracy [19], [20]. Hence, it is more appropriate to consider all the classes simultaneously for multicategory classification. For a k-category classification problem ($k \geq 2$), a simultaneous classifier requires k classification functions in principle. To reduce the redundant parameter space, the sum-to-zero constraint is often imposed on the k functions either explicitly [19], [21], [22] or implicitly [23], [24], [25]. Such a constraint is used to ensure that the degree of freedom of the k classification functions is $k-1$. In this study we refer to multicategory SVMs that use k classification functions as regular MSVMs. Similar terminology of regular MSVMs is also used by [20], [26], [27], [28], and [29]

Regular MSVMs can be cast into constrained optimization problems, but the extra classification function and the intrinsic sum-to-zero constraint make their optimization computationally expensive, especially for large-scale applications. In last decades, the efficient simplex-based classification framework has been proposed to address the sum-to-zero constraint in regular MSVMs, e.g., [27], [28], [30], [31], [32], [33], [34], and [35]. A simplex structure plays a critical role for these simplex-based classifiers, which naturally generalizes the classical binary methods and also removes the sum-to-zero constraint. Consider a regular simplex centred at the

Manuscript received 11 November 2020; revised 24 August 2022; accepted 1 November 2022. Date of publication 15 November 2022; date of current version 17 March 2023. This work was supported in part by the National Science Foundation of China under 72071138 and in part by the Singapore MOE AcRF Tier 2 under Grant R-266-000-143-112. *(Corresponding author: Piao Chen.)*

Sheng Fu is with the Division of Cancer Epidemiology and Genetics, National Cancer Institute, Bethesda, MD 20892 USA (e-mail: fusheng1007@gmail.com).

Piao Chen is with the Delft Institute of Applied Mathematics, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: p.chen-6@tudelft.nl).

Zhisheng Ye is with the Department of Industrial Systems Engineering & Management, National University of Singapore, Singapore 119077 (e-mail: yez@nus.edu.sg).

Communicated by C. Suh, Associate Editor for Machine Learning.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIT.2022.3222266.

Digital Object Identifier 10.1109/TIT.2022.3222266

origin in the $(k-1)$-dimensional Euclidean space, and assume that each vertex vector represents one category. Then only $k-1$ classification functions are needed and the classifier can be efficiently trained without the sum-to-zero constraint. The simplex-based MSVMs and their extensions have been well studied and applied in a variety of applications, e.g., [36], [37], [38], [39], [40], and [41]. As shown by [33], the simplex-based classifiers can enjoy a lower computational cost than regular procedures with the sum-to-zero constraint.

Despite their widespread popularity, both regular and simplex-based MSVMs suffer from an intrinsic disadvantage. Due to the non-differentiable hinge loss, the resulting classifiers do not directly yield the estimate of the category probability [22], [42], [43], which is informative to indicate the quality or confidence of the outcome of classification. In other words, these MSVMs are not able to provide the estimate of the category probability given the explanatory variables, which is undesirable in a variety of practical applications, e.g., [44]. In the literature, several efforts have been made to obtain the probabilistic outputs for MSVMs. There are generally two approaches to overcome the difficulty brought by the non-differentiable hinge loss. The first one is to modify MSVMs by refitting procedure [45], [46], [47] or sequential weighted learning [48], [49]. The former depends heavily on the assumptions of the refitting model while the latter involves extensive training. This category of methods consists of a two-step procedure with indirect probabilistic results from the model fitting. On the other hand, the second category of methods is to replace the hinge loss in MSVMs by some proper loss functions, such as the quadratic (least square) loss [11], [50], [51], [52], [53] and large-margin unified machine loss [22], [33]. Benefiting from the modified loss functions, these classifiers can estimate the class conditional probability explicitly and conveniently. However, the connections among these existing methods are still unclear. There are other multicategory classifiers using more complicated loss functions, e.g., [27], 54], and [55], which cannot yield a convenient or tractable category probability estimation.

Among the various multicategory classifiers, we focus on the ones built on quadratic loss in this study, known as proximal MSVMs, which can be viewed as competitive approximators of their original counterparts based on hinge loss [56]. Due to the connection between hinge loss and quadratic loss, the proximal MSVMs enjoy several merits. First, they can be efficiently trained by solving a linear system [50], [52]. Second, the conditional class probability can then be readily estimated [53], [57], [58]. There exist some seminal works incorporating the quadratic loss and the simplex-based structure, e.g., [30], [32], and [43], that can share the additional advantages of simplex-based approaches. However, the relationships among these regular and simplex-based classifiers are still unclear, and several important theoretical properties have not been thoroughly investigated, such as the category probability estimation, the convergence analysis, the generalization bound and the universal consistency in the reproducing Hilbert kernel space.

The main objective of this study is to propose a unified class of multicategory classifiers by using the quadratic loss and simplex-based framework, which is termed as Simplex-based Proximal MSVM (SPMSVM). In particular, we propose a general family of squared loss functions under the simplex framework, which generalizes many loss functions of existing proximal MSVMs [32], [43], [53], [57], [58]. Specifically, SPMSVM solves a unconstrained optimization problem, which is equivalent to a weighted regression model. Hence, the closed-form solution for SPMSVM can be readily obtained via solving a system of linear equations. Furthermore, we can borrow the advanced solvers from the linear regression literature for scalable implementation of SPMSVM. Unlike MSVMs, SPMSVM is able to estimate the category probability due to the elaborate loss functions. Moreover, the intimate relationship between many existing MSVMs and SPMSVM is theoretically established, and they are indeed special cases of SPMSVM. From this perspective, SPMSVM can be treated as a general integration of many existing classifiers.

Furthermore, the statistical learning theory of SPMSVM is thoroughly investigated. First, we show that under some conditions, the proposed loss functions are Fisher consistent, which is a fundamental requirement for classifiers [59], [60], [61]. We then derive the closed-form expressions for the category probability, which is important in knowing the strength of the prediction. Due to the flexible loss functions, the established probability expressions cover many existing results. In addition, we also establish some other theoretical properties of SPMSVM, including comparison inequalities for the excess misclassification risk, convergence rate, finite sample bound and universal consistency. These properties are important in learning and justifying the performance of classifier asymptotically and in finite samples, but they are rarely discussed in the literature of MSVMs possibly due to the aforementioned technical difficulties. For example, [43] proposed a simplex-based MSVM using a quadratic loss, which is actually a special case of SPMSVM. However, only Fisher consistency and category probability estimation were investigated in their paper. On the other hand, these theoretical gaps are well filled in this study as the statistical properties established for SPMSVM can be naturally extended to many existing MSVMs.

The rest of this article is organized as follows. In Section II, we give a brief review of regular MSVMs and the simplex-based approaches. In Section III, we propose the novel SPMSVM and explore its theoretical properties. We study some statistical properties of the kernel SPMSVM in Section IV. Computationally efficient and scalable algorithms are developed in Section V. Numerical studies are presented in Section VI. Some discussions are provided in Section VII. All theoretical proofs and technical details are provided in the Appendix.

II. Simultaneous Multicategory Classification

In this section, we review the large margin multicategory classification. Suppose we are given a training set $\mathcal{T} = \{(\boldsymbol{x}_i, y_i),\ i = 1, \ldots, n\}$, obtained from an unknown underlying distribution $\mathcal{P}(X, Y)$. Here, $\boldsymbol{x}_i \in \mathcal{X} \subseteq \mathbb{R}^d$ is the input predictor and $y_i \in \mathcal{Y} = \{1, \ldots, k\}$ is the corresponding

TABLE I
EXAMPLES OF LOSS FUNCTIONS IN MULTICATEGORY CLASSIFICATION, WHERE $[u]_+ = \max(u, 0)$

Reference	$V(\boldsymbol{g}(\boldsymbol{x}), y)$
[18]	$[1 - g_y(\boldsymbol{x})]_+$
[50]	$(1 - g_y(\boldsymbol{x}))^2 + \sum_{j \neq y}(1 + g_j(\boldsymbol{x}))^2$
[23]	$\sum_{j \neq y}[1 - (f_y(\boldsymbol{x}) - f_j(x))]_+$
[24]	$[1 - \min_{j \neq y}(f_y(\boldsymbol{x}) - f_j(x))]_+$
[19]	$\sum_{j \neq y}[1 + g_j(\boldsymbol{x})]_+$
[53]	$\sum_{j \neq y}(1 + g_j(\boldsymbol{x}))^2$
[57]	$(1 - g_y(\boldsymbol{x}))^2$
[58]	$\gamma(k - 1 - g_y(\boldsymbol{x}))^2 + (1 - \gamma)\sum_{j \neq y}(1 + g_j(\boldsymbol{x}))^2,\ \gamma \in [0, 1]$
[21]	$\gamma[k - 1 - g_y(\boldsymbol{x})]_+ + (1 - \gamma)\sum_{j \neq y}[1 + g_j(\boldsymbol{x})]_+,\ \gamma \in [0, 1]$
[63]	$\|\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{e}_y\|^2$
[64]	$[1 - g_y(\boldsymbol{x})]_+ + \sum_{j \neq y}[1 + g_j(\boldsymbol{x})]_+$
[65]	$[\alpha - g_y(\boldsymbol{x})]_+ + \sum_{j \neq y}[1 + g_j(\boldsymbol{x})]_+,\ \alpha \in \mathbb{R}$

category. Our target is to find a classifier mapping $\mathcal{X}$ to $\mathcal{Y}$ based on the dataset $\mathcal{T}$, so that we can predict the class membership of a new observation.

In what follows, let $\mathbf{1}$ be a vector of 1's, $\mathbf{0}$ be a vector of 0's, and $\boldsymbol{e}_j$ be the j-th column of an identity matrix $\mathbf{I}$. Their dimensions can be contextually inferred. Denote $\|\cdot\|$ as the Euclidean norm of a vector, and $\mathrm{Tr}(\cdot)$ as the trace of a square matrix. Define $\mathrm{sign}(\cdot)$ as the sign function, and $\mathbb{1}(\cdot)$ as the indicator function taking 1 when the statement is true and 0 otherwise.

A. Regular Classification Framework

We review some multicategory classification techniques from the perspective of category coding scheme. For a k-category classification problem, we encode the j-th category as $\boldsymbol{e}_j \in \mathbb{R}^k$. To consider all classes together, many existing large margin classifiers require the classification function $\boldsymbol{g} = (g_1, \ldots, g_k) : \mathcal{X} \mapsto \mathbb{R}^k$, such as [19], [21], [23], and [24]. The consequent prediction rule is defined as $\mathrm{argmax}_j \langle \boldsymbol{g}(\boldsymbol{x}), \boldsymbol{e}_j \rangle = \mathrm{argmax}_j\, g_j(\boldsymbol{x})$ for any $\boldsymbol{x} \in \mathcal{X}$. That is to say, the component $g_j(\boldsymbol{x})$ represents the score of classifying $\boldsymbol{x}$ as the j-th category.

Observe that the prediction rule is invariant under a shift of each component of classification function. In particular, if we add the same function $h : \mathcal{X} \mapsto \mathbb{R}$ to every g_j, the predicted label does not change as $\mathrm{argmax}_j\, g_j(\boldsymbol{x}) = \mathrm{argmax}_j\{g_j(\boldsymbol{x}) + h(\boldsymbol{x})\}$. To address this undesired obstacle and obtain a unique solution, the sum-to-zero constraint $\sum_{j=1}^k g_j(\cdot) = 0$ is often imposed for multicategory classification; see [2] and [62] for a comprehensive review.

For a given classification function $\boldsymbol{g}$, a point $(\boldsymbol{x}, y)$ is misclassified if and only if $y \neq \mathrm{argmax}_j\, g_j(\boldsymbol{x})$. Let $V(\boldsymbol{g}(\boldsymbol{x}), y)$ be a loss function that measures the loss of using $\boldsymbol{g}(\boldsymbol{x})$ to predict its label y. A sensible loss V should enforce g_y to be the maximum among $g_1, \ldots, g_k$. For example, we list several commonly used loss functions in the MSVMs in Table I. Notice that the hinge loss and least square function play a central role in these loss functions.

Typically, a large margin multicategory classifier follows the "$loss + penalty$" framework,

$$\min_{\boldsymbol{g} \in \mathcal{G}} \ \frac{1}{n}\sum_{i=1}^n V(\boldsymbol{g}(\boldsymbol{x}_i), y_i) + \lambda\Omega(\boldsymbol{g}), \tag{1}$$

where $\mathcal{G} = \{\boldsymbol{g} : \mathcal{X} \mapsto \mathbb{R}^k | \sum_{j=1}^k g_j(\boldsymbol{x}) = 0,\ \forall \boldsymbol{x} \in \mathcal{X}\}$ is the hypothesis class, $\Omega(\cdot)$ is the roughness penalty of $\boldsymbol{g}$ to control the model complexity and prevent overfitting, and $\lambda > 0$ is the tuning parameter to balance the loss and penalty terms.

The sum-to-zero constraint for $\boldsymbol{g}$ is needed to make the model identifiable. To illustrate, consider the binary classification with labels $\{1, 2\}$. We have $g_2 = -g_1$ and the resulting prediction rule is $\frac{3}{2} - \frac{1}{2}\mathrm{sign}(g_1(\boldsymbol{x}))$, where only one single function works. This constraint obviously incurs non-negligible computational efforts in solving (1).

B. Simplex-Based Classification Framework

To avoid the cumbersome sum-to-zero constraint in (1), [27], [30], [31], [32], [33] considered the multicategory classification under a simplex structure, and they showed that the simplex-based methods enjoy more efficient optimization than regular ones. To begin with, define k simplex vertex vectors in $\mathbb{R}^{k-1}$ as follows,

$$\mathbf{w}_j = \begin{cases} \frac{1}{\sqrt{k-1}}\mathbf{1}, & \text{if } j = 1 \\ -\frac{1+\sqrt{k}}{(k-1)^{3/2}}\mathbf{1} + \sqrt{\frac{k}{k-1}}\boldsymbol{e}_{j-1}, & \text{if } 2 \leq j \leq k \end{cases}, \tag{2}$$

where $\mathbf{1}$ and $\boldsymbol{e}_j$'s are vectors in $\mathbb{R}^{k-1}$. Thus, these $\mathbf{w}_j$'s form a k-vertex simplex, denoted as a matrix $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_k) \in \mathbb{R}^{(k-1)\times k}$. One can verify that $\sum_{j=1}^k \mathbf{w}_j = \mathbf{0}$, each $\mathbf{w}_j$ has Euclidean norm 1 and the inner products between any distinct pairs are equal, i.e., $\langle \mathbf{w}_i, \mathbf{w}_j \rangle = -\frac{1}{k-1}$ for any $i \neq j$.

For illustration, we show an example of the simplex-based classification with $k = 3$ in Figure 1. In general, we assign the j-th category to the j-th vertex vector $\mathbf{w}_j$ $(j = 1, \ldots, k)$. A simplex-based classifier requires a $(k-1)$-dimensional vector-valued function $\boldsymbol{f} : \mathcal{X} \mapsto \mathbb{R}^{k-1}$ with the prediction rule $\mathrm{argmax}_j \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j \rangle$, which removes one redundant component compared to regular classifiers. The k inner products are related to the angles between the mapped data and each vertex vector of the simplex $\mathbf{W}$, so the simplex-based classification is also called angle-based classification by [33]. Since $\sum_{j=1}^k \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j \rangle \equiv 0$ naturally holds for any $\boldsymbol{x} \in \mathcal{X}$, the sum-to-zero constraint is implicitly transferred to the k inner products.

With the help of the simplex $\mathbf{W}$, we can simplify the functional space of the classification function, as shown in the following proposition.

Proposition 1: Consider two spaces $\mathcal{F} = \{\boldsymbol{f} | \boldsymbol{f} : \mathcal{X} \mapsto \mathbb{R}^{k-1}\}$ and $\mathcal{G} = \{\boldsymbol{g} : \mathcal{X} \mapsto \mathbb{R}^k | \sum_{j=1}^k g_j(\boldsymbol{x}) = 0,\ \forall \boldsymbol{x} \in \mathcal{X}\}$. Then $\mathcal{G}$ is equivalent to $\mathcal{G}' = \{\mathbf{W}^\top \boldsymbol{f} : \mathcal{X} \mapsto \mathbb{R}^k | \boldsymbol{f} \in \mathcal{F}\}$.

By Proposition 1, the simplex-based multicategory classification solves the following problem

$$\min_{\boldsymbol{f} \in \mathcal{F}} \ \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + \lambda\Omega(\boldsymbol{f}), \tag{3}$$

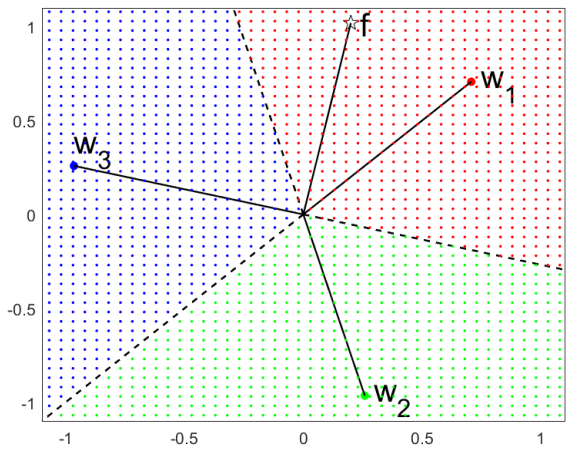

Fig. 1. Illustration for the simplex-based classification with $k = 3$. The dashed lines stand for decision boundaries, which split the space into the red/green/blue regions corresponding to classes 1/2/3. The mapped $\boldsymbol{f}$ is closest to $\mathbf{w}_1$ with the predicted label 1.

TABLE II
EXAMPLES OF LOSS FUNCTIONS IN SIMPLEX-BASED CLASSIFICATION

Reference	$\ell(\boldsymbol{f}(\boldsymbol{x}), y)$
[32]	$[1 - \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y \rangle]_+$
[32]	$\sum_{j \neq y} [1 + \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j \rangle]_+$
[28]	$\gamma[k - 1 - \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y \rangle]_+ + (1-\gamma) \sum_{j \neq y} [1 + \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j \rangle]_+, \ \gamma \in [0, 1]$
[43]	$\|\boldsymbol{f}(\boldsymbol{x}) - \mathbf{w}_y\|^2$

where $\ell(\cdot, \cdot)$ is a proper simplex-based loss function. For the learned classifier $\widehat{\boldsymbol{f}}$ from (3), the corresponding decision boundary classifier can be represented as $\{\boldsymbol{x} \in \mathcal{X} | \ \widehat{\boldsymbol{f}}(\boldsymbol{x}) = t(\mathbf{w}_i + \mathbf{w}_j), \ t \geq 0, \ \forall i \neq j\}$. When $k = 2$, it reduces to the well-known binary decision boundary, $\{\boldsymbol{x} \in \mathcal{X} | \ \widehat{f}(\boldsymbol{x}) = 0\}$.

The simplex coding strategy brings in several distinctive merits, such as symmetry, elimination of extra parameters and constraints, and simplification of computation and model interpretation. Table II shows some loss functions for the simplex-based classifiers, which are closely related to regular loss functions in Table II. Because the loss function in [27] is too complicated and does not have a compact form, it is not included in Table II. Interested readers can refer to the paper for details.

With a clear geometric interpretation in Figure 1 as well as [31], [32], and [33], it is easy to understand the simplex-based classification. In contrast to regular methods, the simplex-based classifier (3) efficiently solves an unconstrained optimization problem involving fewer parameters. Hereafter, we focus on the simplex-based classification framework, and propose a flexible family of loss functions to guarantee some properties for the resulting classifiers.

III. SIMPLEX-BASED PROXIMAL MULTICATEGORY SVM

A desirable multicategory classifier should enjoy sound theoretical properties, be efficient to compute and be able to estimate conditional class probabilities [58]. These requirements are mainly determined by the employed loss functions. Since many MSVMs are built on the non-differentiable hinge loss and its generalizations, they do not attain any information of the conditional class probability [44]. On the other hand, although the simplex-based MSVMs can circumvent the sum-to-zero constraint, the corresponding algorithms are still computationally intensive due to the hinge loss [28]. To overcome these difficulties, we propose the simplex-based proximal MSVM (SPMSVM) in this section.

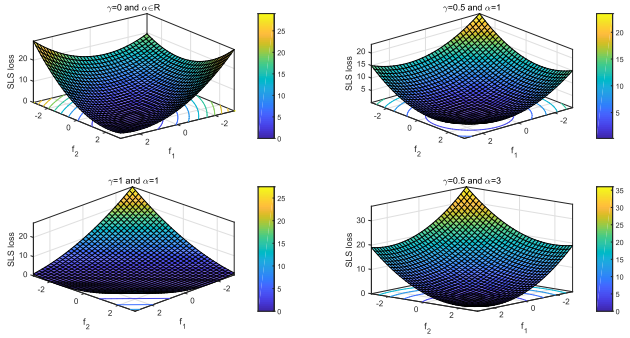

Fig. 2. Plot of the SLS loss functions with different $\{\gamma, \alpha\}$ under $k = 3$ and $y = 1$.

Based on the general formula in (3), we first design a novel family of simplex-based least square (SLS) loss functions as follows,

$$\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}), y) = \gamma(\alpha - \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y \rangle)^2 + (1-\gamma) \sum_{j \neq y} (1 + \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j \rangle)^2, \tag{4}$$

where $\gamma \in [0, 1]$ is a convex combination parameter and α is a scale parameter. A reasonable SLS loss function should enforce large $\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y \rangle$ and small $\{\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j \rangle, \ j \neq y\}$ for correct classification. Figure 2 shows the plot of the SLS loss function (4) under different values of (γ, α). It can be seen that as γ increases, the value of the loss function increases when both f_1 and f_2 are negative, while the loss decreases when only one of f_j's is negative. Moreover, as α increases, the loss function increases when at least one of f_j's is negative, and the loss decreases when both f_1 and f_2 are positive.

The proposed SLS loss with parameters $\{\gamma, \alpha, k\}$ contains a broad family of loss functions. For example, when $\gamma = 0$, it is the simplex-based generalization of that in [53]. When $(\gamma, \alpha) = (1, 1)$, it extends the squared loss in [57] by the simplex. When $\alpha = k - 1$, it is the simplex-based counterpart of the composite least square loss in [58]. When $(\gamma, \alpha) = (\frac{1}{2}, 1)$, it refines the sequential binary proximal SVM based on the one-vs-all approach, where there are no regions of ambiguity in the prediction space due to the simplex encoding [27], [59]. When $(\gamma, \alpha) = (\frac{1}{2}, \frac{1}{k-1})$, one can verify that

$$\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}), y) = \frac{k}{2(k-1)} \|\boldsymbol{f}(\boldsymbol{x}) - \mathbf{w}_y\|^2 + \text{Constant}, \tag{5}$$

which is equivalent to the simplex least square loss in [32] and the least distance loss used in [43], by ignoring the factor $\frac{k}{2(k-1)}$ and the irrelevant constant. Consider binary classification with $k = 2$. The simplex becomes $(w_1, w_2) = (1, -1)$, and the SLS loss is arranged as

$$\mathcal{L}(f(\boldsymbol{x}), y) = [(\gamma\alpha + 1 - \gamma) - w_y f(\boldsymbol{x})]^2 + \text{Constant}.$$

If $\gamma\alpha+1-\gamma>0$, we can recover the binary classifiers in [11] and [51]. Hence, the proposed SLS loss can provide a unified insight on a multitude of existing classifiers.

For the training set $\mathcal{T}$, we propose the following SPMSVM classifier

$$\min_{\boldsymbol{f}\in\mathcal{F}} \quad \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i),y_i)+\lambda\Omega(\boldsymbol{f}), \tag{6}$$

which can be viewed as a proximal version of multiclass SVM although the "support vector" property does not clearly hold. More precisely, the support vectors of non-proximal MSVMs are determined by a small part of training set, while the support vectors of the SPMSVM contain all the training data points, which is a major difference noticed by [51], [52], and [53]. Due to the simplex-based classification framework, SPMSVM enjoys computational convenience by solving an unconstrained optimization problem with fewer parameters.

To investigate the statistical properties of the SPMSVM (6), we consider the ideal case when n goes to infinity and $\lambda=0$. Recall that $\mathcal{P}$ is the underlying distribution of (X,Y). Let $\mathcal{P}_X$ be the marginal distribution of X, and $P_j(\boldsymbol{x})=\Pr(Y=j|X=\boldsymbol{x})$ be the conditional class probability for the j-th category and $\boldsymbol{x}\in\mathcal{X}$. Sometimes, we simply write $P_j(\boldsymbol{x})$ as P_j by omitting the dependence on $\boldsymbol{x}$.

For any classifier $\mathcal{C}:\mathcal{X}\to\mathcal{Y}$, we employ the misclassification error to measure its performance, also known as 0–1 risk and defined as

$$\mathcal{R}(\mathcal{C})=\mathbb{E}_{\mathcal{P}}[\mathbb{1}(\mathcal{C}(X)\neq Y)]=\Pr(\mathcal{C}(X)\neq Y). \tag{7}$$

Thus, the 0–1 risk $\mathcal{R}(\mathcal{C})$ is identical to the probability of classification error. The optimal classification rule that minimizes $\mathcal{R}(\mathcal{C})$ is the Bayes classifier

$$\mathcal{C}^*(\boldsymbol{x})=\operatorname*{argmax}_j P_j(\boldsymbol{x}), \quad \forall\boldsymbol{x}\in\mathcal{X}. \tag{8}$$

For a general classification function $\boldsymbol{f}:\mathcal{X}\mapsto\mathbb{R}^{k-1}$, the resulting simplex-based classifier $\mathcal{C}_{\boldsymbol{f}}$ is defined as $\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})=\operatorname{argmax}_j\langle\boldsymbol{f}(\boldsymbol{x}),\mathbf{w}_j\rangle$ for $\boldsymbol{x}\in\mathcal{X}$. Thus, the misclassification risk for $\mathcal{C}_{\boldsymbol{f}}$ is given by

$$\begin{aligned}\mathcal{R}(\mathcal{C}_{\boldsymbol{f}})&=\int_{\mathcal{X}}P(Y\neq\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})|X=\boldsymbol{x})dP(\boldsymbol{x})\\&=1-\int_{\mathcal{X}}P(Y=\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})|X=\boldsymbol{x})dP(\boldsymbol{x}).\end{aligned}$$

Due to the optimality of $\mathcal{C}^*$, we have $\mathcal{R}(\mathcal{C}_{\boldsymbol{f}})\geq\mathcal{R}(\mathcal{C}^*)$ for any $\boldsymbol{f}$.

The SPMSVM uses the SLS loss $\mathcal{L}$ as a surrogate of the 0–1 risk. Similarly, the full $\mathcal{L}$-risk of a classification function $\boldsymbol{f}$ is defined as

$$\begin{aligned}\mathcal{E}(\boldsymbol{f})&=\mathbb{E}_{\mathcal{P}}[\mathcal{L}(\boldsymbol{f}(X),Y)]\\&=\int_{\mathcal{X}\times\mathcal{Y}}\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}),y)dP(\boldsymbol{x},y)\\&=\int_{\mathcal{X}}\sum_{j=1}^{k}P_j(\boldsymbol{x})\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}),j)dP(\boldsymbol{x}).\end{aligned} \tag{9}$$

The conditional $\mathcal{L}$-risk at a fixed $\boldsymbol{x}\in\mathcal{X}$ is given by

$$\mathcal{S}_{\boldsymbol{x}}(\mathbf{u})=\sum_{j=1}^{k}P_j(\boldsymbol{x})\mathcal{L}(\mathbf{u},j), \quad \mathbf{u}\in\mathbb{R}^{k-1}. \tag{10}$$

Clearly, $\mathcal{E}(\boldsymbol{f})$ is a functional of $\boldsymbol{f}$ and $\mathcal{S}_{\boldsymbol{x}}(\mathbf{u})$ is a function of $\mathbf{u}$. As suggested by [32], we consider a hypothesis space of measurable functions

$$L_2(\mathcal{P}_X)=\Big\{\boldsymbol{f}:\mathcal{X}\mapsto\mathbb{R}^{k-1}\Big|\int_{\mathcal{X}}\|\boldsymbol{f}(\boldsymbol{x})\|^2\,dP(\boldsymbol{x})<\infty\Big\}.$$

The following proposition summarizes some properties of the expected $\mathcal{L}$-risk.

Proposition 2: The full $\mathcal{L}$-risk is a convex and continuous functional $\mathcal{E}:L_2(\mathcal{P}_X)\mapsto\mathbb{R}_+$.

With Proposition 2, we can define the population minimizer of $\mathcal{E}(\boldsymbol{f})$ as

$$\boldsymbol{f}^*=\operatorname*{arginf}_{\boldsymbol{f}\in L_2(\mathcal{P}_X)}\mathcal{E}(\boldsymbol{f}). \tag{11}$$

With the above setup, we will conduct theoretical analysis on the SPMSVM. Specifically, we establish some properties about $\boldsymbol{f}^*$, including Fisher consistency, probability estimation, excess risk bounds and convergence rate.

A. Fisher Consistency

Fisher consistency is a fundamental asymptotic property for classifiers. Fisher consistency requires that when the sample size is sufficiently large, the classifier learned from a surrogate loss function approximates the Bayes classification rule, which corresponds to the minimum misclassification rate. Specifically, a general simplex-based loss function $\mathcal{L}$ is Fisher consistent if and only if $\mathcal{C}_{\boldsymbol{f}^*}=\mathcal{C}^*$, where $\boldsymbol{f}^*$ is the population minimizer similar to (11); see [32].

Fisher consistency defined as above depends on the full $\mathcal{L}$-risk. We can define an alternative Fisher consistency in a pointwise manner, i.e., $\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})=\mathcal{C}^*(\boldsymbol{x})$, which is constructed on the minimization of the conditional $\mathcal{L}$-risk at an arbitrary $\boldsymbol{x}\in\mathcal{X}$. Such an adjustment is helpful to verify Fisher consistency in practice; see [28], [33], and [43]. The definition for the general and pointwise manner of Fisher consistency was introduced in [59], wherein this property is also called infinite-sample consistency. To investigate Fisher consistency for the SLS loss (4), we introduce a mild assumption which is widely adopted in the MSVM literature.

Assumption 1: Assume that $P_j(\boldsymbol{x})>0$ for all $\boldsymbol{x}\in\mathcal{X}$ and $j=1,\ldots,k$.

Let $\boldsymbol{f}^*$ be the theoretical minimizer of the full $\mathcal{L}$-risk $\mathcal{E}(\boldsymbol{f})$, as defined in (11). For any fixed $\boldsymbol{x}\in\mathcal{X}$, the following proposition characterizes the minimizer of the conditional $\mathcal{L}$-risk $\mathcal{S}_{\boldsymbol{x}}(\cdot)$ and the explicit expression of $\boldsymbol{f}^*(\boldsymbol{x})$.

Proposition 3: For the SLS loss $\mathcal{L}$ with (γ,α), under Assumption 1, the vector $\boldsymbol{f}^*(\boldsymbol{x})$ minimizes the conditional $\mathcal{L}$-risk at any $\boldsymbol{x}\in\mathcal{X}$, and takes the form

$$\boldsymbol{f}^*(\boldsymbol{x})=\sum_{j=1}^{k}s_j(\boldsymbol{x})\mathbf{w}_j, \tag{12}$$

where $s_j(\boldsymbol{x}) = \frac{1}{k}(\gamma\alpha+1-\gamma)(k-1)a_j\Big(P_j - \frac{\sum_{t=1}^k a_t P_t}{\sum_{t=1}^k a_t}\Big)$ and $a_j = \frac{1}{(2\gamma-1)P_j+1-\gamma}$. Moreover, $\boldsymbol{f}^* \in L_2(\mathcal{P}_X)$.

For the full $\mathcal{L}$-risk minimizer $\boldsymbol{f}^*$, Proposition 3 implies that $\boldsymbol{f}^*(\boldsymbol{x})$ is the minimizer of the conditional $\mathcal{L}$-risk $\mathcal{S}_{\boldsymbol{x}}(\cdot)$ at any $\boldsymbol{x} \in \mathcal{X}$. From the expression (12), we know that $\boldsymbol{f}^*(\boldsymbol{x})$ is a linear combination of the vertex vector $\mathbf{w}_j$'s, where the coefficients are determined by $P_j(\boldsymbol{x})$'s. For example, when $\gamma = \frac{1}{2}$, $\boldsymbol{f}^*(\boldsymbol{x}) = \frac{(k-1)(\alpha+1)}{k}\sum_{j=1}^k P_j(\boldsymbol{x})\mathbf{w}_j$. Moreover, if $P_j(\boldsymbol{x}) \equiv \frac{1}{k}$ for any $j = 1, \ldots, k$ and $\boldsymbol{x} \in \mathcal{X}$, we have the degenerated case $\boldsymbol{f}^*(\boldsymbol{x}) \equiv \mathbf{0}$, and the predicted label can be any of the possible classes.

The following theorem provides a sufficient condition for the SLS loss to achieve Fisher consistency.

Theorem 1: Under Assumption 1, if $\gamma\alpha + 1 - \gamma > 0$, then the SLS loss is Fisher consistent.

Assumption 1 in Proposition 3 and Theorem 1 is adopted to simplify expressions, and it is useful to eliminate some corner cases. For example, if $P_t = 0$ and $\gamma = 1$, we have $\boldsymbol{f}^* = -(k-1)\alpha\mathbf{w}_t$ from Proposition 3. No matter what α is, we have $\langle \boldsymbol{f}^*, \mathbf{w}_j\rangle \equiv \alpha$ for any $j \neq t$, which does not depend on the information of $\{P_j,\ j \neq t\}$. Obviously, it is a case of little interest in practice.

We illustrate Theorem 1 by considering some special examples. If $\gamma = 0$, the SLS loss does not contain α, and the consequent simplex-based classifier enjoys Fisher consistency, which is consistent with the results for regular classifier in [53]. For each $\gamma \in (0, 1]$, we should require $\alpha > 1 - \frac{1}{\gamma}$ to ensure Fisher consistency. When $\gamma = 1$, the SLS loss becomes $(\alpha - \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y\rangle)^2$, which is Fisher consistent for $\alpha > 0$. [57] proved Fisher consistency for regular least square loss $(1 - g_y(\boldsymbol{x}))^2$ under the sum-to-zero constraint, and here we generalize it to the advanced simplex-based classification framework. Moreover, we develop a simplex-based version of the composite least square loss used in [58] by setting $\alpha = k-1$, and Fisher consistency is still reserved. For the SLS loss with $(\gamma, \alpha) = (\frac{1}{2}, \frac{1}{k-1})$, we recover the Fisher consistency in [43]. Compared with existing simplex-based MSVMs, e.g., [28] and [33], our results about Fisher consistency are more general due to the wider range $\gamma \in [0, 1]$.

B. Estimation of Category Probability

In addition to the category membership prediction, estimation of the conditional class probability is also important in understanding the data [44]. Classifiers that can provide an estimation of the class probability are termed as "soft" classifiers by [42], which have drawn extensive attention in the literature; see [22], [33], [43], [53], and [58] and the reference therein. Here, we show that our SPMSVM can naturally provide an estimation of the conditional class probability.

As shown by Proposition 3, Eq. (12) implies that the theoretical minimizer $\boldsymbol{f}^*(\boldsymbol{x})$ is a function of the conditional class probabilities, $P_j(\boldsymbol{x})$'s. A natural idea is to establish the link functions between $P_j(\boldsymbol{x})$'s and $\boldsymbol{f}^*(\boldsymbol{x})$, which represent $P_j(\boldsymbol{x})$ as a function of $\boldsymbol{f}^*(\boldsymbol{x})$. Furthermore, once an estimate of $\boldsymbol{f}^*$ is obtained from the dataset, it can be used to predict class probabilities. The following theorem gives the explicit form of link functions for SPMSVM.

Theorem 2: Let $\boldsymbol{f}^*$ be the minimizer of full-$\mathcal{L}$ risk as defined in (11). For any fixed $\boldsymbol{x} \in \mathcal{X}$, the link functions can be expressed as

$$P_j(\boldsymbol{x}) = \begin{cases} \frac{1}{\alpha+1}\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_j\rangle + \frac{1}{k}, & \text{if } \gamma = \frac{1}{2} \\ \Big(1 + \frac{k(1-\gamma)}{2\gamma-1}\Big)\frac{c_j}{\sum_{i=1}^k c_i} - \frac{1-\gamma}{2\gamma-1}, & \text{if } \gamma \neq \frac{1}{2} \end{cases}, \quad (13)$$

where $c_j = \frac{1}{(2\gamma-1)\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_j\rangle - (\gamma\alpha+1-\gamma)}$ for $j = 1, \ldots, k$.

Theorem 2 is a guideline to estimate the conditional class probability, via the link function (13). We observe that $P_i(\boldsymbol{x}) = P_j(\boldsymbol{x})$ if and only if $\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_i\rangle = \langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_j\rangle$. If the fitted $\widehat{\boldsymbol{f}}$ is a consistent estimator of true $\boldsymbol{f}^*$, based on (13), then the estimated probabilities $\{\widehat{P}_j(\widehat{\boldsymbol{f}}),\ j = 1, \ldots, k\}$ are also consistent. Under some mild assumptions on the data generating process, the consistency and convergence results will be discussed in Section III-D.

The exact form of link functions for SLS loss with general (γ, α), as shown in (13), is more prominent than some existing methods. [58] only studied the link functions for three corner values of γ. If $\gamma = \frac{1}{2}$, in contrast to Theorem 2 in [43], our link functions avoid the calculation of a matrix inverse. In addition, the results in [53] can be recovered when $\gamma = 0$. We should mention in passing that the simplex-based MSVMs involving hinge loss provide no results on the conditional class probability; see [28] and [33].

When $\gamma \neq \frac{1}{2}$, it is difficult to simplify the link function (13) except for $\gamma = 0$ and 1. For a given $\widehat{\boldsymbol{f}}$, the estimated conditional class probabilities at $\gamma \in \{0, \frac{1}{2}, 1\}$ are as follows,

$$\widehat{P}_j(\boldsymbol{x}) = \begin{cases} 1 - (k-1)\frac{1/[1+\langle \widehat{\boldsymbol{f}}(\boldsymbol{x}), \mathbf{w}_j\rangle]}{\sum_{t=1}^k 1/[1+\langle \widehat{\boldsymbol{f}}(\boldsymbol{x}), \mathbf{w}_t\rangle]}, & \text{if } \gamma = 0 \\ \frac{1}{\alpha+1}\langle \widehat{\boldsymbol{f}}(\boldsymbol{x}), \mathbf{w}_j\rangle + \frac{1}{k}, & \text{if } \gamma = \frac{1}{2} \\ \frac{1/[\langle \widehat{\boldsymbol{f}}(\boldsymbol{x}), \mathbf{w}_j\rangle - \alpha]}{\sum_{t=1}^k 1/[\langle \widehat{\boldsymbol{f}}(\boldsymbol{x}), \mathbf{w}_t\rangle - \alpha]}, & \text{if } \gamma = 1 \end{cases}. \quad (14)$$

For $\gamma > 0$ and $\alpha \to +\infty$, each estimated probability approximates $\frac{1}{k}$ for any $\boldsymbol{x} \in \mathcal{X}$. Then, the procedure of probability estimation at this corner case is degenerated.

With the help of Theorem 2, the conditional class probability estimators $\widehat{P}_j(\boldsymbol{x})$'s naturally satisfy the sum-to-one condition, i.e., $\sum_{j=1}^k \widehat{P}_j(\boldsymbol{x}) = 1$. A potential issue is that the individual estimator $\widehat{P}_j(\boldsymbol{x})$ could be outside of $[0, 1]$. To ensure a proper probability estimation, we consider the following rescale procedure on the original $\widehat{P}_j$'s

$$\widehat{P}_j^{\text{scaled}}(\boldsymbol{x}) = \frac{\widehat{P}_j(\boldsymbol{x}) - \min_{i=1,\ldots,k}\widehat{P}_i(\boldsymbol{x})}{\sum_{l=1}^k [\widehat{P}_l(\boldsymbol{x}) - \min_{i=1,\ldots,k}\widehat{P}_i(\boldsymbol{x})]} \in [0, 1].$$

One can check that the scaled probabilities still satisfy the sum-to-one condition. Similar modifications can be found in [43], [58], and [66].

C. Relaxation Error Analysis

For classification tasks, the adopted loss function plays a pivotal role as a relaxation of the misclassification risk. To quantify the error incurred by relaxation, it is of great interest to derive comparison inequalities explicitly relating the excess misclassification risk to the excess expected loss.

In statistical learning theory, we can use these inequalities to obtain rates of convergence or oracle inequalities. The excess risk concept is first discussed in binary classification by [60], and some classical results for particular multicategory classifiers are established by [30], [32], and [59]. To the best of our knowledge, a systematic error analysis for the classifiers involving the least square loss functions is still lacking in the literature. To fill this gap, we conduct a thorough error analysis for the SPMSVM in this section. Since the proposed SLS loss is a rich family of least square functions and the SPMSVM covers many popular MSVMs in a unified framework, these results can be naturally extended to many existing classifiers.

For the misclassification risk $\mathcal{R}(\mathcal{C})$ in (7), $\mathcal{C}^*$ is the Bayes classifier defined by (8). For the full $\mathcal{L}$-risk $\mathcal{E}(\boldsymbol{f})$ in (9), $\boldsymbol{f}^*$ is the population minimizer defined by (11). Denote $\mathcal{R}^* = \mathcal{R}(\mathcal{C}^*)$ and $\mathcal{E}^* = \mathcal{E}(\boldsymbol{f}^*)$. Under the regularity conditions stated in Theorem 1, the SLS loss enjoys Fisher consistency, which implies $\mathcal{C}^* = \mathcal{C}_{\boldsymbol{f}^*}$ and $\mathcal{R}(\mathcal{C}_{\boldsymbol{f}^*}) = \mathcal{R}(\mathcal{C}^*) = \mathcal{R}^*$. Further, the comparison inequality depicts a quantitative relation between the excess misclassification risk and the excess $\mathcal{L}$-risk as

$$\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \leq \psi(\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*) \tag{15}$$

for any function $\boldsymbol{f}$ and a nondecreasing function $\psi : [0, \infty) \mapsto [0, \infty)$. A suitable ψ should satisfy $\psi(0) = 0$. Note that ψ highly depends on the loss function $\mathcal{L}$, and possibly the data distribution. If ψ is known, then the inequality (15) not only implies Fisher consistency, but also allows to bound the excess risk by the excess $\mathcal{L}$-risk. In particular, the bounds on the excess $\mathcal{L}$-risk can yield bounds on the excess misclassification risk.

We first make a technical assumption as follows, which is slightly stronger than Assumption 1.

Assumption 2: Assume that there exists a constant $\delta \in (0, \frac{1}{2})$ such that $\delta \leq P_j(\boldsymbol{x}) \leq 1 - \delta$ for all $\boldsymbol{x} \in \mathcal{X}$ and $j = 1, \ldots, k$.

Next, we state the main results in the following theorem.

Theorem 3: Suppose that Assumption 2 is satisfied and $\gamma\alpha + 1 - \gamma > 0$. The following comparison inequality hold for any $\boldsymbol{f} \in L_2(\mathcal{P}_X)$,

$$\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \leq C_\gamma \sqrt{\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*}, \tag{16}$$

where C_γ is $\frac{2\gamma - 1 + k(1-\gamma)}{\gamma\alpha + 1 - \gamma}\sqrt{\frac{2}{\delta}}$ for $\gamma \in [0, \frac{1}{2})$, $\frac{2}{\alpha+1}$ for $\gamma = \frac{1}{2}$ and $\frac{\{2\gamma - 1 + k(1-\gamma)\}(1-\delta)}{(\gamma\alpha + 1 - \gamma)\sqrt{\delta^3}}$ for $\gamma \in (\frac{1}{2}, 1]$.

Note that Assumption 2 is mild because it only affects the constant C_γ under the setting $\gamma \neq \frac{1}{2}$. Consider a sequence of estimates $\widehat{\boldsymbol{f}}^n$, such that $\widehat{\boldsymbol{f}}^n \to \boldsymbol{f}^*$. The corresponding $\mathcal{L}$-risk sequences satisfy $\mathcal{E}(\widehat{\boldsymbol{f}}^n) \to \mathcal{E}^*$, due to the continuity of $\mathcal{E}(\cdot)$. By Theorem 3, we have $\mathcal{R}(\mathcal{C}_{\widehat{\boldsymbol{f}}^n})$ converges to $\mathcal{R}^*$, i.e., the resulting sequences of classifiers $\mathcal{C}_{\widehat{\boldsymbol{f}}^n}$ enjoy Fisher consistency. In addition, the order of excess risk $\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*$ can be improved for some special distributions. Motivated by [32] and [59], we introduce the multiclass low noise condition in the following assumption, which can be viewed as a generalization of Tsybakov's binary noise condition [67].

Assumption 3: Given $\boldsymbol{x} \in \mathcal{X}$, let $P_{(1)}(\boldsymbol{x})$ and $P_{(2)}(\boldsymbol{x})$ be the first largest and second largest conditional probability, respectively. Assume that there exists $C > 0$, $a \geq 0$ and $h^* > 0$, such that the distribution $\mathcal{P}(X, Y)$ satisfies the following condition:

$$\mathcal{P}_X(\{\boldsymbol{x} \in \mathcal{X} | P_{(1)}(\boldsymbol{x}) - P_{(2)}(\boldsymbol{x}) \leq h\}) \leq Ch^a, \quad \forall h \in (0, h^*]. \tag{17}$$

Intuitively, when it is difficult to distinguish the class with the highest probability from others, i.e., $P_{(1)}(\boldsymbol{x})$ is very close to $P_{(2)}(\boldsymbol{x})$ for some $\boldsymbol{x}$, there may exist extremely large misclassification error. In particular, if $a = 0$, condition (17) reduces to the case without any noise assumption. When $a = \infty$, it becomes the ideal noiseless case.

With Assumption 3, we can improve the results of Theorem 3 as follows.

Theorem 4: Under the same conditions in Theorem 3, if Assumption 3 holds, then we have

$$\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \leq \widetilde{C}^{\frac{a+1}{a+2}} \{\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*\}^{\frac{a+1}{a+2}}, \tag{18}$$

where $\widetilde{C} = 4(a+1)C^{\frac{1}{a+1}} a^{-\frac{a}{a+1}} C_\gamma^2 > 0$.

Remarkably, Theorem 3 is covered as a special case of Theorem 4 with $a = 0$. If $a > 0$, the order of $\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*$ is $\frac{a+1}{a+2} > \frac{1}{2}$, which is better than that of Theorem 3. Further, setting $a = \infty$ leads to $\frac{a+1}{a+2} \to 1$, which refines the results in Theorem 3.

D. Convergence Rate Analysis

The recent work by [22] and [33] studied the convergence of the excess risk for a family of multicategory classifiers inspired from large margin unified loss [42]. In this section, we are interested in the convergence rate analysis for the proposed SPMSVM. Specifically, we establish the bounds on the excess $\mathcal{L}$-risk $\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*$. By the comparison inequalities in Section III-C, one can bound the excess misclassification risk, $\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*$, in terms of the excess $\mathcal{L}$-risk. Consequently, these can lead to the bounds on the excess misclassification risk.

Let $\widehat{\boldsymbol{f}}^n$ be the learned SPMSVM classification function from the dataset $\mathcal{T}$ and a functional space H, i.e., $\widehat{\boldsymbol{f}}^n = \operatorname{argmin}_{\boldsymbol{f} \in H} \frac{1}{n}\sum_{i=1}^n \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i)$, which is also called the empirical risk minimizer. For the full $\mathcal{L}$-risk $\mathcal{E}(\boldsymbol{f})$, the population minimizer within the space H is $\boldsymbol{f}^H = \operatorname{arginf}_{\boldsymbol{f} \in H} \mathcal{E}(\boldsymbol{f})$. In what follows, we reveal the relationship between the convergence rate of $\widehat{\boldsymbol{f}}^n$ to $\boldsymbol{f}^H$ and that of the excess $\mathcal{L}$-risk, as well as the size of the functional space H.

Recall the conditional $\mathcal{L}$-risk $\mathcal{S}_{\boldsymbol{x}}(\mathbf{u})$ defined in (10). By Proposition 3, we denote the minimizer of $\mathcal{S}_{\boldsymbol{x}}(\mathbf{u})$ as $\mathbf{u}^* = \operatorname{argmin}_{\mathbf{u} \in \mathbb{R}^{k-1}} \mathcal{S}_{\boldsymbol{x}}(\mathbf{u})$, where $\mathbf{u}^*$ depends on the feature $\boldsymbol{x}$. Meanwhile, for the population minimizer $\boldsymbol{f}^* \in L_2(\mathcal{P}_X)$ defined by (11), we have $\boldsymbol{f}^*(\boldsymbol{x}) = \mathbf{u}^*$ pointwisely for $\boldsymbol{x}$. Let $\Delta\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) = \mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) - \mathcal{S}_{\boldsymbol{x}}(\mathbf{u}^*)$ and $\Delta\mathcal{E}(\boldsymbol{f}) = \mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*$. Furthermore, the excess $\mathcal{L}$-risk satisfies $\Delta\mathcal{E}(\boldsymbol{f}) = \mathbb{E}_{\mathcal{P}_X}[\Delta\mathcal{S}_{\boldsymbol{x}}(\boldsymbol{f}(\boldsymbol{x}))]$.

The following theorem states the property of the excess conditional $\mathcal{L}$-risk at a fixed $\boldsymbol{x}$.

Theorem 5: Under Assumption 1, we have

$$\Delta\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) = (\mathbf{u} - \mathbf{u}^*)^\top \left(\sum_{j=1}^k [\gamma P_j + (1-\gamma)(1-P_j)] \mathbf{w}_j \mathbf{w}_j^\top \right) (\mathbf{u} - \mathbf{u}^*).$$

Theorem 5 can be used to establish the connection of the convergence rate of the classification function and that of the excess $\mathcal{L}$-risk. Since $\boldsymbol{f}^H$ minimizes the $\mathcal{L}$-risk within the space H, we can decompose the excess $\mathcal{L}$-risk as

$$\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n) = \left[\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n) - \Delta\mathcal{E}(\boldsymbol{f}^H)\right] + \Delta\mathcal{E}(\boldsymbol{f}^H),$$

where the first term depends on the data fitting and is called the *$\mathcal{L}$-estimation error*, and the second term is deterministic and is called the *$\mathcal{L}$-approximation error*. When H is sufficiently rich, so that

$$\boldsymbol{f}^H = \operatorname*{arginf}_{\boldsymbol{f}\in H} \mathcal{E}(\boldsymbol{f}) = \operatorname*{arginf}_{\boldsymbol{f}\in L_2(\mathcal{P}_X)} \mathcal{E}(\boldsymbol{f}) = \boldsymbol{f}^*, \tag{19}$$

the estimator $\widehat{\boldsymbol{f}}^n$ will converge to $\boldsymbol{f}^*$ as the sample size n grows, under some regularity conditions. One typical example is that H is dense in $L_2(\mathcal{P}_X)$. In this case, the $\mathcal{L}$-approximation error is zero, and the excess $\mathcal{L}$-risk is essentially the $\mathcal{L}$-estimation error. Thus, we can explore how the convergence rate of $\widehat{\boldsymbol{f}}^n$ affects the convergence rate of the excess $\mathcal{L}$-risk.

We first introduce some notations and assumptions, which are similar to those in [22]. Note that γ and α are parameters of the SLS loss $\mathcal{L}$ (4). Let $\mu(\cdot)$ be regular Lebesgue measure. For a fixed pair (γ, α), associated with the distribution $\mathcal{P}(X, Y)$, we can naturally define $k-1$ probability measures on the real line: $\tau_j(B) = P(f_j^* \in B),\ j = 1, \ldots, k-1$, where B is any Borel measurable set.

Assumption 4: For any $\gamma \in [0, 1]$ and $\alpha > 0$, $\tau_j \ll \mu,\ j = 1, \ldots, k-1$. Namely, every measure τ_j is absolutely continuous with respect to the Lebesgue measure μ.

Assumption 5: For any $\gamma \in [0, 1]$ and $\alpha > 0$, $n^q\{\widehat{\boldsymbol{f}}^n(X, Y) - \boldsymbol{f}^*(X, Y)\} \to \boldsymbol{T}(\gamma, \alpha, X, Y)$ in distribution, where $\boldsymbol{T}(\gamma, \alpha, X, Y) = (T_1, \ldots, T_{k-1})$ is a multivariate random variable, whose distribution depends on (γ, α), and varies among different (X, Y); $q > 0$ is a constant. Moreover, suppose that for fixed (γ, α), $\int_{X,Y} |\sup_{1\le j\le k-1} T_j|^2 dP(X, Y) < \infty$.

Assumption 4 is valid if there is no probability mass point in the distribution $\mathcal{P}$, and Assumption 5 is essential to prevent the distribution of $\boldsymbol{T}$ from diverging with large probability when (X, Y) varies. Next we state the main result in the following theorem.

Theorem 6: Consider the SPMSVM model with the underlying distribution $\mathcal{P}$. Suppose Assumptions 1, 4 and 5 are satisfied, and (19) holds. Then for any fixed γ and α,

$$\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n) = O(n^{-2q}).$$

Under Assumption 4, the $\mathcal{L}$-risk $\mathcal{E}(\boldsymbol{f})$ has bounded second order derivative for a fixed (γ, α) almost surely. For proper (γ, α), if $q = \frac{1}{2}$ for regular finite dimensional problems, then $\widehat{\boldsymbol{f}}^n$ is $\sqrt{n}-$consistent. From Theorem 6, we can claim that the excess $\mathcal{L}$-risk is n-consistent under these conditions.

When the $\mathcal{L}$-approximation error is nonzero, i.e.,

$$\inf_{\boldsymbol{f}\in H} \mathcal{E}(\boldsymbol{f}) > \inf_{\boldsymbol{f}\in L_2(\mathcal{P}_X)} \mathcal{E}(\boldsymbol{f}), \tag{20}$$

then the excess $\mathcal{L}$-risk does not converge to 0, and Theorem 6 becomes inapplicable. In such cases, we are interested in the convergence rate of the $\mathcal{L}$-estimation error, $\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n) - \Delta\mathcal{E}(\boldsymbol{f}^H) = \mathcal{E}(\widehat{\boldsymbol{f}}^n) - \mathcal{E}(\boldsymbol{f}^H)$. First, we should modify the previous Assumption 5 as follows.

Assumption 6: For any $\gamma \in [0, 1]$ and $\alpha > 0$, $n^q\{\widehat{\boldsymbol{f}}^n(X, Y) - \boldsymbol{f}^H(X, Y)\} \to \boldsymbol{T}(\gamma, \alpha, X, Y)$ in distribution, where $\boldsymbol{T}(\gamma, \alpha, X, Y) = (T_1, \ldots, T_{k-1})$ is a multivariate random variable, whose distribution depends on (γ, α), and varies among different (X, Y); $q > 0$ is a constant. Moreover, suppose that for fixed (γ, α), $\int_{X,Y} |\sup_{1\le j\le k-1} T_j| dP(X, Y) < \infty$ and $\int_{X,Y} |\sup_{1\le j\le k-1} f_j^*| + |\sup_{1\le j\le k-1} f_j^H| dP(X, Y) < \infty$.

Theorem 7: Consider the SPMSVM model with the underlying distribution $\mathcal{P}$. Suppose Assumptions 1, 4 and 6 are satisfied, and (20) holds. Then for any fixed γ and α,

$$\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n) - \Delta\mathcal{E}(\boldsymbol{f}^H) = O(n^{-q}).$$

Based on Theorem 7, if $\boldsymbol{f}^* \notin H$, the convergence rate of the excess $\mathcal{L}$-risk $\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n)$ is the same as $\widehat{\boldsymbol{f}}^n$. Hence, the excess $\mathcal{L}$-risk is also $\sqrt{n}-$consistent under some mild conditions when $q = \frac{1}{2}$.

E. Comparison With Existing Methods

In this section, we compare the theoretical results of SPMSVM with some related multicategory classifiers involving quadratic loss functions, as shown in Table III. As seen, the Fisher consistency and category probability estimation have been extensively studied, while our method serves as a unified framework to connect them. In addition, we extend the results of the excess risk to a more flexible family of SLS loss. On the other hand, the convergence rate is a novel theoretical contribution to the literature. In this study, we establish the relationship between the convergence rate of the classification function $\widehat{\boldsymbol{f}}^n$ and that of the excess SLS risk, as well as the size of the functional space. The results can be extended to other MSVMs in the table.

IV. Kernel Learning for SPMSVM

Notice that in Section III, we consider the ideal setting for the SPMSVM with infinite samples and without the penalty, and we put no restrictions on the form of the classification functions. In this section, we concentrate on the L_2 regularized SPMSVM in a reproducing kernel Hilbert space (RKHS) and reveal the connections between regular multicategory classifier and its simplex-based version. The closed-form solution of the kernel SPMSVM is derived for efficient computation. In particular, the finite sample generalization bound and universal consistency are also established, which are novel contributions to the MSVM literature.

First, we need to introduce some conventional notations. Let $K(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ be a positive definite kernel, and H_K be a structured RKHS generated by K. Specifically, H_K is defined as the linear span of the set of functions $\{K_{\boldsymbol{x}} = K(\boldsymbol{x}, \cdot) : \boldsymbol{x} \in \mathcal{X}\}$ with the inner product $\langle K_{\boldsymbol{x}}, K_{\boldsymbol{y}}\rangle_{H_K}$ given by $\langle K_{\boldsymbol{x}}, K_{\boldsymbol{y}}\rangle_{H_K} = K(\boldsymbol{x}, \boldsymbol{y})$. RKHS also has the following

TABLE III
COMPARISON ON THE THEORETICAL RESULTS OF SPMSVM WITH SOME EXISTING METHODS

Method	Connection	Fisher consistency	Probability estimation	Excess risk	Convergence rate
[59]	$\gamma=\frac{1}{2},\ \alpha=1$	✓	—	✓	—
[53]	$\gamma=0,\ \alpha\in\mathbb{R}$	✓	✓	—	—
[57]	$\gamma=1,\ \alpha=1$	✓	✓	—	—
[58]	$\gamma=1,\ \alpha=k-1$	✓	✓	—	—
[32]	$\gamma=\frac{1}{2},\ \alpha=\frac{1}{k-1}$	✓	—	✓	—
[43]	$\gamma=\frac{1}{2},\ \alpha=\frac{1}{k-1}$	✓	✓	—	—
SPMSVM	$\gamma\alpha+1-\gamma>0$	✓(Sec. III-A)	✓(Sec. III-B)	✓(Sec. III-C)	✓(Sec. III-D)

reproducing property

$$\langle K_{\boldsymbol{x}}, h\rangle_{H_K} = h(\boldsymbol{x}),\quad \forall h\in H_K,\ \boldsymbol{x}\in\mathcal{X}.$$

Denote $\|h\|_{H_K}$ as the square norm of the function $h\in H_K$. We refer to [3] for more details about RKHS.

A. Connections With Regular Methods

Consider the multicategory classifier (1), i.e., $\min_{\boldsymbol{g}\in\mathcal{G}}\ \frac{1}{n}\sum_{i=1}^{n}V(\boldsymbol{g}(\boldsymbol{x}_i),y_i)+\lambda\Omega(\boldsymbol{g})$. Assume that $g_j(\cdot)=p_j+q_j(\cdot)$ with $p_j\in\mathbb{R}$ and $q_j\in H_K$ for $j=1,\ldots,k$. When the linear kernel $K(\boldsymbol{x}_1,\boldsymbol{x}_2)=\boldsymbol{x}_1^\top\boldsymbol{x}_2$ is used, the standard linear learning for (1) is recovered. The kernelized multicategory classifier with the L_2 penalty solves

$$\begin{aligned}\min_{\boldsymbol{g}\in\prod_{j=1}^{k}(\{1\}+H_K)}\quad & \frac{1}{n}\sum_{i=1}^{n}V(\boldsymbol{g}(\boldsymbol{x}_i),y_i)+\lambda\sum_{j=1}^{k}\|q_j\|_{H_K}^2,\\ \text{subject to}\quad & \sum_{j=1}^{k}g_j(\boldsymbol{x})=0,\quad\forall\boldsymbol{x}\in\mathcal{X}.\end{aligned}\tag{21}$$

In [19] and [21], the sum-to-zero constraints at all possible values of $\boldsymbol{x}$ can be equivalently relaxed to the n observations $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n\}$. According to Proposition 1, there exists a function $\boldsymbol{f}\in\prod_{j=1}^{k-1}(\{1\}+H_K)$ such that $\boldsymbol{g}=\mathbf{W}^\top\boldsymbol{f}$. For notational simplicity, we write $\boldsymbol{f}(\cdot)=\boldsymbol{c}+\boldsymbol{h}(\cdot)$ with $\boldsymbol{c}\in\mathbb{R}^{k-1}$ and $\boldsymbol{h}=(h_1,\ldots,h_{k-1})\in H_K^{k-1}$. As a result, $g_j=\mathbf{w}_j^\top\boldsymbol{f}$, $p_j=\mathbf{w}_j^\top\boldsymbol{c}$ and $q_j=\mathbf{w}_j^\top\boldsymbol{h}$ for $j=1,\ldots,k$. Given that $\sum_{j=1}^{k}g_j(\boldsymbol{x})=\sum_{j=1}^{k}\mathbf{w}_j^\top\boldsymbol{f}(\boldsymbol{x})\equiv 0$, the cumbersome constraints in (21) can be naturally removed. Consequently, we can replace $\boldsymbol{g}$ by $\mathbf{W}^\top\boldsymbol{f}$ to simplify (21), and the main result is summarized in the following theorem.

Theorem 8: With the simplex $\mathbf{W}$, the regularized multicategory classifier (21) can be reduced to

$$\min_{\boldsymbol{f}\in\prod_{j=1}^{k-1}(\{1\}+H_K)}\ \frac{1}{n}\sum_{i=1}^{n}V(\mathbf{W}^\top\boldsymbol{f}(\boldsymbol{x}_i),y_i)+\frac{k\lambda}{k-1}\sum_{j=1}^{k-1}\|h_j\|_{H_K}^2.\tag{22}$$

Theorem 8 indicates that if one legitimately defines the simplex-based loss function as $\ell(\boldsymbol{f}(\boldsymbol{x}),y)=V(\mathbf{W}^\top\boldsymbol{f}(\boldsymbol{x}),y)$, then the tuning parameter λ changes with a scale factor $\frac{k}{k-1}$. In other words, the simplex-based classifier (22) is equivalent to regular classifier (21). Nevertheless, (22) only uses $k-1$ classification functions and it solves an unconstrained optimization problem, and hence its computation and statistical analysis are much less challenging. As an application of Theorem 8, we can reveal the equivalence between the reinforced MSVM [21] and the simplex-based MSVM [28]. In the literature, [68] showed the equivalence between the two methods by [19] and [32] under a general unregularized and cost-sensitive scenario, while [69] studied such equivalence under a linear setting, which are special cases of our results. Interestingly, guaranteed by Theorem 8, we can show that the classifier in [23] is equivalent to a special example of GenSVM [27], because the GenSVM loss function with $p=1$ and $\kappa\to-1$ is the simplex-based version of [23]. Although the simplex-based classifiers are equivalent to their regular versions under the sum-to-zero constraint, they can be more efficiently solved due to the benefits from the simplex structure [28], [33].

In what follows, we focus on the SPMSVM with the SLS loss $\mathcal{L}$ in (4). Following the RKHS-based statistical learning, we assume that the classification function $\boldsymbol{f}:\mathcal{X}\mapsto\mathbb{R}^{k-1}$ takes the form

$$f_j(\boldsymbol{x})=c_j+h_j(\boldsymbol{x}),\quad j=1,\ldots,k-1,$$

where $c_j\in\mathbb{R}$ and $h_j\in H_K$. Then, the kernel SPMSVM seeks $\boldsymbol{f}\in\prod_{j=1}^{k-1}(\{1\}+H_K)$ to minimize the L_2 penalized empirical SLS loss function as follows,

$$\min_{\boldsymbol{f}\in\prod_{j=1}^{k-1}(\{1\}+H_K)}\ \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i),y_i)+\lambda\sum_{j=1}^{k-1}\|h_j\|_{H_K}^2.\tag{23}$$

Due to the flexibility of the loss $\mathcal{L}$, we extend existing MSVMs in [32], [43], [53], and [58] to a general kernel setting.

Although the sum-to-zero constraint has been removed, the optimization of (23) is over an infinite dimensional hypothesis space, and it is still a challenging task. The following representer theorem is helpful to convert problem (23) into a finite-dimensional optimization problem, which is much easier to solve.

Theorem 9: Let $\boldsymbol{f}^*$ be the solution of problem (23). Then, there exists some coefficients $b_{ij}\in\mathbb{R}$ and $c_j\in\mathbb{R}$ $(i=1,\ldots,n;\ j=1,\ldots,k-1)$, such that $\boldsymbol{f}^*(\boldsymbol{x})$ can be represented as

$$f_j^*(\boldsymbol{x})=\sum_{i=1}^{n}b_{ij}K(\boldsymbol{x}_i,\boldsymbol{x})+c_j,\quad j=1,\ldots,k-1.$$

For more discussions on the representer theorem, we refer readers to [70], [71], [72], [73], [74], and [75].

B. Finite Sample Generalization Bound

In reality, all statistical models are estimated from a set of finite samples. In this section, we study the finite sample data-dependent generalization bound on the full $\mathcal{L}$-risk, i.e., $\mathcal{E}(\boldsymbol{f}) = \mathbb{E}_{\mathcal{P}}[\mathcal{L}(\boldsymbol{f}(X), Y)]$. Following the conventions, since $f_j(\cdot) = c_j + h_j(\cdot)$, we can absorb the intercept c_j into h_j for notational convenience. Let $\Omega(\boldsymbol{f}) = \sum_{j=1}^{k-1} \|f_j\|_{H_K}^2$ and $\mathcal{H} = H_K^{k-1}$. Furthermore, the kernel SPMSVM is formulated as

$$\min_{\boldsymbol{f}\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + \lambda\Omega(\boldsymbol{f}). \tag{24}$$

By duality with convex $\mathcal{L}$ and regularization $\Omega(\cdot)$, (24) is equivalent to the optimization problem

$$\min_{\boldsymbol{f}\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i), \quad \text{subject to} \quad \Omega(\boldsymbol{f}) \le \Lambda. \tag{25}$$

Note that there is a one-to-one correspondence for (λ, Λ). Then, we consider the hypothesis space

$$\mathcal{F}_\Lambda = \{\boldsymbol{f} : \mathcal{X} \mapsto \mathbb{R}^{k-1} | f_j \in H_K,\ j = 1, \ldots, k-1,\ \Omega(\boldsymbol{f}) \le \Lambda\}. \tag{26}$$

Assumption 7 is needed to establish the generalization bound of the full $\mathcal{L}$-risk, and the main results about generalization bound are presented in Theorem 10.

Assumption 7: There exists a constant $C_X > 0$ such that $\sup_{\boldsymbol{x}\in\mathcal{X}} \sqrt{K(\boldsymbol{x}, \boldsymbol{x})} = C_X < \infty$.

Theorem 10: Suppose that Assumption 7 is satisfied. Denote $\mu = \frac{2k}{k-1} C_X \sqrt{\Lambda} + 2|\gamma\alpha + 1 - \gamma|$ and $M = \gamma\alpha^2 + (1-\gamma)(k-1) + \mu C_X \sqrt{\Lambda}$, where γ and α are parameters for the SLS loss $\mathcal{L}$. Then for any $0 < \theta < 1$, with probability at least $1 - \theta$, the following holds for all $\boldsymbol{f} \in \mathcal{F}_\Lambda$:

$$\mathcal{E}(\boldsymbol{f}) \le \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + 2\mu C_X \sqrt{\frac{2(k-1)\Lambda}{n}} + M\sqrt{\frac{\log(1/\theta)}{2n}}.$$

Under Assumption 7, one can verify that $\mathcal{F}_\Lambda \subseteq L_2(\mathcal{P}_X)$, so $\mathcal{E}^* = \inf_{\boldsymbol{f}\in L_2(\mathcal{P}_X)} \mathcal{E}(\boldsymbol{f}) \le \inf_{\boldsymbol{f}\in\mathcal{F}_\Lambda} \mathcal{E}(\boldsymbol{f})$. Assume that $\widehat{\boldsymbol{f}}^n \in \mathcal{F}_\Lambda$ is the solution to (25). Applying Theorem 10 to $\widehat{\boldsymbol{f}}^n$, we obtain a upper bound for the minimum full $\mathcal{L}$-risk with high probability

$$\mathcal{E}^* \le \mathcal{E}(\widehat{\boldsymbol{f}}^n) \le \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\widehat{\boldsymbol{f}}^n(\boldsymbol{x}_i), y_i) + 2\mu C_X \sqrt{\frac{2(k-1)\Lambda}{n}} + M\sqrt{\frac{\log(1/\theta)}{2n}}.$$

The above bound is computable from the dataset $\mathcal{T}$ and the chosen model.

It is noteworthy that the technical analysis using Rademacher complexity plays a critical role to derive the generalization bound in Theorem 10. Under the regularization framework (24) of the kernel SPMSVM, we will employ the similar technique to investigate the universal consistency of the resulting estimator, which will be introduced in the next section.

C. Universal Consistency

Note that the kernel SPMSVM is a non-parametric function estimation problem in a product RKHS. In this section, we elucidate the behavior of the $\mathcal{L}$-approximation error and establish the universal consistency of the regularized estimator under a universal kernel. The concept of universal consistency is a fundamental asymptotical property for a machine learning method, which requires that when the sample size grows to infinity, the method eventually approaches the Bayes rule without any specifications of the distribution of the data.

The definition of a universal kernel is adapted from [76] and [77]. Assume that $\mathcal{X} \subseteq \mathbb{R}^d$ is a compact input space of X and $\mathfrak{C}(\mathcal{X})$ is the space of all continuous functions $g : \mathcal{X} \mapsto \mathbb{R}$. We say the kernel $K(\cdot, \cdot)$ defined on $\mathcal{X}$ is universal if the RKHS H_K generated by K is dense in $\mathfrak{C}(\mathcal{X})$, i.e., for any $\epsilon > 0$ and any $g \in \mathfrak{C}(\mathcal{X})$, there is an $f \in H_K$ such that $\|f - g\|_\infty = \sup_{x\in X} |f(x) - g(x)| < \epsilon$.

If the RKHS is rich enough, then the $\mathcal{L}$-approximation error can be arbitrarily small. For a general kernel K, let $\boldsymbol{f}^{\mathcal{H}} = \inf_{\boldsymbol{f}\in\mathcal{H}} \mathcal{E}(\boldsymbol{f})$ with $\mathcal{H} = H_K^{k-1}$. Recall that $\boldsymbol{f}^* = \inf_{\boldsymbol{f}\in L_2(\mathcal{P}_X)} \mathcal{E}(\boldsymbol{f})$ and $\mathcal{E}^* = \mathcal{E}(\boldsymbol{f}^*)$. The $\mathcal{L}$-approximation error is given by $\Delta\mathcal{E}(\boldsymbol{f}^{\mathcal{H}}) = \mathcal{E}(\boldsymbol{f}^{\mathcal{H}}) - \mathcal{E}^*$.

We first show that, if K is a universal kernel, then the $\mathcal{L}$-approximation error $\Delta\mathcal{E}(\boldsymbol{f}^{\mathcal{H}}) = 0$.

Theorem 11: Suppose that Assumption 1 holds. Let $\mathcal{X}$ be a compact space and H_K be induced by a universal kernel K. Then we have $\Delta\mathcal{E}(\boldsymbol{f}^{\mathcal{H}}) = 0$.

Theorem 11 also implies a fact that $\mathcal{H}$ is dense in $L_2(\mathcal{P}_X)$. Consider the regularized estimator

$$\widehat{\boldsymbol{f}}^n = \operatorname*{argmin}_{\boldsymbol{f}\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + \lambda\Omega(\boldsymbol{f}). \tag{27}$$

The following theorem demonstrates that the $\mathcal{L}$-risk at $\widehat{\boldsymbol{f}}^n$ is consistent under some conditions.

Theorem 12: Suppose conditions in Theorem 11 and Assumption 7 are satisfied. Let $\lambda = \lambda_n \to 0$, such that $n\lambda_n^2 \to \infty$ as $n \to \infty$. For the SLS loss $\mathcal{L}$ with $\gamma \in [0, 1]$ and finite α, the estimator $\widehat{\boldsymbol{f}}^n$ is defined by (27). Then, we have $\mathcal{E}(\widehat{\boldsymbol{f}}^n) - \mathcal{E}^* \xrightarrow{a.s.} 0,\ \forall\mathcal{P}(X, Y)$.

Assumption 7 holds for the Gaussian kernel with $C_X \le 1$. Hence, Theorem 12 indicates that the SPMSVM involving the Gaussian kernel is consistent. Moreover, we can extend the results of Theorem 12 to establish the consistency of the misclassification risk in the following corollary.

Corollary 1: Under the same conditions as in Theorems 3 and 12, we have

$$\mathcal{R}(\mathcal{C}_{\widehat{\boldsymbol{f}}^n}) - \mathcal{R}^* \xrightarrow{a.s.} 0, \quad \forall\mathcal{P}(X, Y).$$

V. Algorithm Implementation

A. General Closed-Form Solution

Define the coefficient matrix $\mathbf{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k-1}) \in \mathbb{R}^{n\times(k-1)}$, where $\boldsymbol{b}_j = (b_{1j}, \ldots, b_{nj})^\top \in \mathbb{R}^n$ is the j-th column vector. Denote $\boldsymbol{c} = (c_1, \ldots, c_{k-1})^\top \in \mathbb{R}^{k-1}$ as a vector of intercepts. Let $\mathbf{K}$ be the $n \times n$ Gram matrix with the (i, j)-entry $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. By Theorem 9, we have $f_j = h_j + c_j$ and $h_j = \sum_{i=1}^n b_{ij} K(\boldsymbol{x}_i, \cdot)$. Furthermore, $\sum_{j=1}^{k-1} \|h_j\|_{H_K}^2 = \sum_{j=1}^{k-1} \boldsymbol{b}_j^\top \mathbf{K} \boldsymbol{b}_j = \mathrm{Tr}(\mathbf{B}^\top \mathbf{K} \mathbf{B})$, using the reproducing property of the RKHS. In particular, denote $\mathbf{K}_i$ as the i-th column vector of $\mathbf{K}$, and then $\boldsymbol{f}(\boldsymbol{x}_i) = \mathbf{B}^\top \mathbf{K}_i + \boldsymbol{c}$. Hence, we can reformulate problem (23) as

$$\min_{\mathbf{B},\mathbf{c}} \ \frac{1}{n} \sum_{i=1}^n \mathcal{L}(\mathbf{B}^\top \mathbf{K}_i + \boldsymbol{c}, y_i) + \lambda \mathrm{Tr}(\mathbf{B}^\top \mathbf{K} \mathbf{B}). \tag{28}$$

Notice that [28] and [58] used a technical trick by imposing penalization on the intercept c_j's, and showed that the incurred difference by penalizing the intercepts is often negligible in many situations. Using the same treatment, we add the extra term $\lambda \boldsymbol{c}^\top \boldsymbol{c}$ into the objective function of (28), and derive its closed-from solution. Let $\widetilde{\mathbf{K}}_i = (1, \mathbf{K}_i^\top)^\top \in \mathbb{R}^{n+1}$ for $i = 1, \ldots, n$. To simplify the mathematical expression, we define two matrices as follows,

$$\widetilde{\mathbf{B}} = \begin{bmatrix} \boldsymbol{c}^\top \\ \mathbf{B} \end{bmatrix} \in \mathbb{R}^{(n+1)\times(k-1)}$$

and

$$\mathbf{G} = \begin{bmatrix} 1 & \mathbf{0}_n^\top \\ \mathbf{0}_n & \mathbf{K} \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}.$$

Then the inner product between $\boldsymbol{f}(\boldsymbol{x}_i)$ and $\mathbf{w}_j$ is $\langle \boldsymbol{f}(\boldsymbol{x}_i), \mathbf{w}_j \rangle = \mathbf{K}_i^\top \mathbf{B} \mathbf{w}_j + \boldsymbol{c}^\top \mathbf{w}_j = \widetilde{\mathbf{K}}_i^\top \widetilde{\mathbf{B}} \mathbf{w}_j$, and the modified regularization is $\mathrm{Tr}(\mathbf{B}^\top \mathbf{K} \mathbf{B}) + \boldsymbol{c}^\top \boldsymbol{c} = \mathrm{Tr}(\widetilde{\mathbf{B}}^\top \mathbf{G} \widetilde{\mathbf{B}})$. Define $\tau_{ij} = \gamma \mathbb{1}(y_i = j) + (1-\gamma)\mathbb{1}(y_i \neq j)$ and $z_{ij} = \alpha \mathbb{1}(y_i = j) - \mathbb{1}(y_i \neq j)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, k$. As a result, the objective function of problem (28) can be modified as

$$\begin{aligned} &\frac{1}{n} \sum_{i=1}^n \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i) + \lambda \mathrm{Tr}(\mathbf{B}^\top \mathbf{K} \mathbf{B}) + \lambda \boldsymbol{c}^\top \boldsymbol{c} \\ =&\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^k \tau_{ij} (z_{ij} - \widetilde{\mathbf{K}}_i^\top \widetilde{\mathbf{B}} \mathbf{w}_j)^2 + \lambda \mathrm{Tr}(\widetilde{\mathbf{B}}^\top \mathbf{G} \widetilde{\mathbf{B}}). \end{aligned} \tag{29}$$

Setting its derivatives with respect to $\widetilde{\mathbf{B}}$ to 0, we obtain the following KKT condition,

$$\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^k 2\tau_{ij} (\widetilde{\mathbf{K}}_i^\top \widetilde{\mathbf{B}} \mathbf{w}_j - z_{ij}) \widetilde{\mathbf{K}}_i \mathbf{w}_j^\top + 2\lambda \mathbf{G} \widetilde{\mathbf{B}} = \mathbf{0}. \tag{30}$$

Denote $\vec{\mathbf{A}} = (a_{11}, \ldots, a_{m1}, a_{12}, \ldots, a_{m2}, \ldots, a_{1n}, \ldots, a_{mn})^\top \in \mathbb{R}^{mn}$ as the vectorization of $\mathbf{A} \in \mathbb{R}^{m\times n}$. Arranging the KKT condition, we obtain an explicit solution of $\widetilde{\mathbf{B}}$,

$$\vec{\widetilde{\mathbf{B}}} = b \cdot \Big(\frac{1}{n} \sum_{i=1}^n \mathbf{\Lambda}_i \otimes (\widetilde{\mathbf{K}}_i \widetilde{\mathbf{K}}_i^\top) + \lambda \mathbf{I}_{k-1} \otimes \mathbf{G} \Big)^{-1} \Big(\frac{1}{n} \sum_{i=1}^n \mathbf{w}_{y_i} \otimes \widetilde{\mathbf{K}}_i \Big), \tag{31}$$

where $b = \gamma\alpha + 1 - \gamma$, $\mathbf{\Lambda}_i = \frac{k(1-\gamma)}{k-1} \mathbf{I}_{k-1} + (2\gamma - 1) \mathbf{w}_{y_i} \mathbf{w}_{y_i}^\top$ and $\otimes$ is the standard Kronecker product for two matrixes and. The closed-form solution (31) involves some fundamental matrix operations. Details on deriving (31) are deferred to the Appendix. In particular, if $\gamma = \frac{1}{2}$, the equation (31) can be simplified as

$$\vec{\widehat{\mathbf{B}}} = \frac{\alpha + 1}{2} \cdot \Big(\frac{k}{2n(k-1)} \sum_{i=1}^n \widetilde{\mathbf{K}}_i \widetilde{\mathbf{K}}_i^\top + \lambda \mathbf{G} \Big)^{-1} \Big(\frac{1}{n} \sum_{i=1}^n \widetilde{\mathbf{K}}_i \mathbf{w}_{y_i}^\top \Big).$$

The closed-form solution (31) is helpful to investigate the effect of α. Consider two pairs (γ, α_1) and (γ, α_2). According to (31), the resulting SPMSVMs based on the same training set and the same tuning parameter λ have the same coefficient matrices except the term b. Let $b_i = \gamma\alpha_i + 1 - \gamma$ for $i = 1, 2$. The learned decision functions for SPMSVMs are linear dependent, i.e., $b_2 \widehat{\boldsymbol{f}}_1 = b_1 \widehat{\boldsymbol{f}}_2$. If b_1 and b_2 have the same sign, then these two SPMSVMs have the same predicted labels. Furthermore, based on Theorem 2, one can verify that the estimated probabilities are exactly the same. Inspired by this observation, we could select α to make the term $b = \gamma\alpha + 1 - \gamma$ be either negative or positive, which simplifies the procedure of tuning an optimal α in practice. The effects of α will be investigated numerically in Section VI-A.

The proposed SPMSVM and many existing MSVMs can be cast into convex quadratic programming (QP) problems. Nevertheless, the proposed SPMSVM simply solves a unconstrained problem while regular MSVMs suffer from the linear constraints. Note that the convex QP can be tackled in polynomial time with either the ellipsoid or interior point method [78], [79]. Therefore, the cost of solving SPMSVM for a fixed λ is at worst $O(n^3)$ via the formula (31), similar to that of [32]. For large n, the unconstrained QP can be solved efficiently by linear conjugate gradient algorithm with optimal convergence rate [80]. Compared to regular MSVMs, another advantage of SPMSVM is that it requires fewer parameters due to the simplex-based structure. More specifically, the kernel SPMSVM involves $(n+1)(k-1)$ parameters, while regular MSVMs have $(n+1)k$ parameters. Thus, $n+1$ parameters are reduced, and so is the required memory.

B. Scalable Algorithm for Linear SPMSVM

Note that when $n/d/k$ is too large, directly using the closed-form solution (31) may become less sufficient, due to practical constraints such as exhaustive matrix operations and large amount of storage [81]. Therefore, it is desirable to design a more scalable algorithm for SPMSVM to handle massive datasets. For large-scale SVM, state-of-the-art solvers include LibSVM [82] and LibLinear [83] using block dual coordinate descent scheme [84], [85], cutting plane method [86] and Pegasos [87] based on stochastic gradient descent procedures. Unlike the direct solver (31), the above-mentioned methods can solve the optimization problem iteratively, which is tractable for modern large datasets. Among them, the approaches using coordinate decent are more prominent with many successful applications for large datasets [2], [84], [88], since the coordinate subproblem is univariate and generally has a closed-form solution. Here,

we concentrate on the implementation of large-scale linear SPMSVM.

Assume that $f_j(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{b}_j + c_j$ for $j = 1, \ldots, k-1$. Let $\mathbf{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k-1}) \in \mathbb{R}^{d\times(k-1)}$ with $\boldsymbol{b}_j = (b_{1j}, \ldots, b_{dj})^\top \in \mathbb{R}^d$ being the j-th column vector, and $\boldsymbol{c} = (c_1, \ldots, c_{k-1})^\top \in \mathbb{R}^{k-1}$. Similar to the arguments in (29), we obtain the following problem

$$\min_{\mathbf{B},\boldsymbol{c}} \ \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k}\tau_{ij}(z_{ij} - \boldsymbol{x}_i^\top \mathbf{B}\mathbf{w}_j - \boldsymbol{c}^\top \mathbf{w}_j)^2 + \lambda(\mathrm{Tr}(\mathbf{B}^\top\mathbf{B}) + \boldsymbol{c}^\top\boldsymbol{c}). \tag{32}$$

We are now ready to convert (32) to a weighted linear regression model. More precisely, consider a set of observations $\{(\widetilde{\boldsymbol{x}}_t, \widetilde{z}_t, \omega_t),\ t = 1, \ldots, n\times k = N\}$, with $\widetilde{\boldsymbol{x}}_t = \mathbf{w}_j \otimes (1, \boldsymbol{x}_i) \in \mathbb{R}^{(k-1)(d+1)}$, $\widetilde{z}_t = z_{ij}$ and $\omega_t = k\tau_{ij}$ for $t = (i-1)k + j$. Define $\boldsymbol{\beta} = (c_1, \boldsymbol{b}_1^\top, \ldots, c_{k-1}, \boldsymbol{b}_{k-1}^\top)^\top$. It can be verified that (32) is equivalent to

$$\min_{\boldsymbol{\beta}} \ \frac{1}{N}\sum_{t=1}^{N}\omega_t(\widetilde{z}_t - \widetilde{\boldsymbol{x}}_t^\top\boldsymbol{\beta})^2 + \lambda\boldsymbol{\beta}^\top\boldsymbol{\beta}, \tag{33}$$

where ω_t is the weight for the t-th observation. In contrast to the original setting, problem (33) has more observations and larger dimensions of covariates, but the number of parameters remains the same. There are many well-developed solvers for the linear regression problem (33), such as stochastic algorithms [89], [90] and many numerical methods in the monograph [91]. In this study, we apply the R `glmnet` package which employs the coordinate descent strategy. More details on the method can be found in [92].

VI. Numerical Studies

In this section, we use extensive simulations and several real datasets to assess the performance of the proposed SPMSVM method. For comparison purposes, we consider several competitive MSVMs, including OVOSVM (one-versus-one extension of binary SVM), WWMSVM [23], CSMSVM [24], GenSVM [27] and RAMSVM [28]. Specifically, we select the best GenSVM in terms of predication accuracy with hyperparameters $(p, \tau) \in \{1, 1.5, 2\} \times \{-0.9, 0.5, 5\}$, and the best RAMSVM with $\gamma \in \{0, 0.1, \ldots, 1\}$, which are suggested in [27] and [28], respectively. Moreover, the REC method [43] can be viewed a special case of SPMSVM, whose results are also exhibited for comparison. All the methods are implemented in the R environment [93], and some well-known packages including `e1071`, `kernlab`, `gensvm` and `ramsvm` are used for existing MSVMSs. As mentioned earlier, the classifiers in [19], [21], [53], and [58] are special cases of these considered methods, so we do not highlight them individually due to the space limit.

Consider a testing set $\{(\boldsymbol{x}_i, y_i),\ i = 1, \ldots, n^*\}$. If the predicted labels by a classifier are $\{\widehat{y}_i,\ i = 1, \ldots, n^*\}$, then the corresponding prediction error is defined as $\frac{1}{n^*}\sum_{i=1}^{n^*}\mathbb{1}(y_i \neq \widehat{y}_i)$, i.e., the proportion of misclassified testing points. In particular, only the SPMSVM (and hence the special case REC) can directly provide probability estimation. We utilize the mean absolute error (MAE) to measure the errors of probability estimation, $\frac{1}{n^*k}\sum_{i=1}^{n^*}\sum_{j=1}^{k}|\widehat{P}_j(\boldsymbol{x}_i) - P_j(\boldsymbol{x}_i)|$, where $P_j(\boldsymbol{x}_i)$ and $\widehat{P}_j(\boldsymbol{x}_i)$ are the true and estimated probabilities, respectively. The MAE has been widely used in the MSVM literature to measure the quality of a classifier [22], [33], [43], [58]. In addition, we also consider the Brier score [94] as a complement, which does not require the true probability. Specifically, the Brier score is defined as $\frac{1}{n^*}\sum_{i=1}^{n^*}\sum_{j=1}^{k}[\widehat{P}_j(\boldsymbol{x}_i) - \mathbb{1}(y_i = j)]^2$. For both MAE and Brier score, a lower value indicates a better performance with 0 being the best possible value.

For each method, we tune the best classifier on a grid of 30 different values of λ: $\{2^{-15}, 2^{-14}, \ldots, 2^{14}\}$. Note that the SLS loss contains two parameters, the convex combination parameter γ and the scale parameter α. We will consider different combinations of (γ, α) to examine their effects on SPMSVM.

A. Simulations

In this section, we conduct three simulations, ranging from small size to large size. For each simulation, we generate a training set to learn a model, a separate tuning set to select the tuning parameter λ, and a testing set to evaluate the performance of the tuned model. Let n and n^* be the size of the training/tuning set and the testing set, respectively. In particular, we fix $n^* = 10000$. The detailed data generating processes are stated as follows.

Example 1: We consider a classification problem with 6 categories and d covariates. For $j = 1, \cdots, 6$, let $\Pr(Y = j) = \frac{1}{6}$ and the first 2 covariates of $[X|Y = j]$ be distributed as $\mathcal{N}(\boldsymbol{\mu}_j, \sigma^2\boldsymbol{I}_2)$ with centers

$$\boldsymbol{\mu}_j = \begin{bmatrix}2\\0\end{bmatrix}, \ \begin{bmatrix}1\\\sqrt{3}\end{bmatrix}, \ \begin{bmatrix}-1\\\sqrt{3}\end{bmatrix}, \ \begin{bmatrix}-2\\0\end{bmatrix}, \ \begin{bmatrix}-1\\-\sqrt{3}\end{bmatrix}, \ \begin{bmatrix}1\\-\sqrt{3}\end{bmatrix},$$

and $\sigma \in \{0.5, 0.7\}$. The last $d-2$ covariates are noise and assumed i.i.d. $\mathcal{N}(0, 0.5)$.

Example 2: In this example, we generate a 4-category classification dataset with 5 covariates. Assume that $\Pr(Y = j) = \frac{1}{4}$ for each category. Specifically, the first two covariates of $[X|Y = j]$ follows a mixture Gaussian distribution

$$\frac{1}{2}\mathcal{N}((\cos(j\pi/4), \sin(j\pi/4))^\top, \sigma^2\boldsymbol{I}_2) + \frac{1}{2}\mathcal{N}((\cos(j\pi/4 + \pi), \sin(j\pi/4 + \pi))^\top, \sigma^2\boldsymbol{I}_2),$$

where $\sigma \in \{0.3, 0.4\}$. The remaining 3 covariates are i.i.d. $\mathcal{N}(0, 0.3)$.

Note that the parameter σ in both examples controls the scale of strength of true signal. As σ grows, there are more overlaps for all classes, and the classification performance may naturally get worse. Due to the settings of these simulations, we apply linear learning for Example 1 and kernel learning with Gaussian radial basis function (RBF) kernel for Example 2. Specifically, we follow the guideline provided in [28] to choose the bandwidth parameter for the Gaussian kernel, that is, we chose it as the median of all the pairwise Euclidean distances of training inputs.

First of all, we investigate the effects of (γ, α) on SPMSVM based on Example 1 with $d = 10$ and $\sigma = 0.5$. We use 100 data points for training, 100 data points for tuning and

TABLE IV
CLASSIFICATION ERRORS OF SPMSVM BASED ON EXAMPLE 1 UNDER DIFFERENT VALUES OF (γ, α)

α	-10,-9	-8,-7,...,-4	-3	-2	-1	0	1,2,...,10
γ=0	0.2539	0.2539	0.2539	0.2539	0.2539	0.2539	0.2539
γ=0.1	0.9998	0.2546	0.2546	0.2546	0.2546	0.2546	0.2546
γ=0.2	0.9999	0.9999	0.2544	0.2544	0.2544	0.2544	0.2544
γ=0.3	0.9999	0.9999	0.9999	0.2541	0.2541	0.2541	0.2541
γ=0.4	0.9999	0.9999	0.9999	0.9999	0.2552	0.2552	0.2552
γ=0.5	0.9999	0.9999	0.9999	0.9999	0.8331	0.2522	0.2522
γ=0.6	0.9999	0.9999	0.9999	0.9999	0.9999	0.2449	0.2449
γ=0.7	0.9999	0.9999	0.9999	0.9999	0.9999	0.2269	0.2269
γ=0.8	0.9999	0.9999	0.9999	0.9999	0.9999	0.1992	0.1992
γ=0.9	0.9999	0.9999	0.9999	0.9999	0.9999	0.1578	0.1578
γ=1	0.9972	0.9972	0.9972	0.9972	0.9972	0.8331	0.1029

another 10000 data points for testing. We consider $\gamma \in \{0, 0.1, \ldots, 1\}$ and $\alpha \in \{-10, -9, \cdots, 0, 1, 2 \ldots, 10\}$. Based on 100 replications, the estimated classification errors are shown in Table IV. Generally, for a fixed γ, the prediction error is non-increasing as α increases. For a fixed $\alpha < 0$, there is a threshold γ_0 such that SPMSVM performs inadequately at $\gamma \geq \gamma_0$. This is consistent with our theoretical analysis on the effects of (γ, α) in Section V-A. More specifically, Fisher consistency is a fundamental requirement for the successful implementation of classifiers. In terms of the proposed SPMSVM, the Fisher consistency may not hold when $\gamma\alpha + 1 - \gamma \leq 0$ and hence unsatisfactory performance can be expected. In what follows, we consider $\gamma \in \{0, 0.1, \ldots, 1\}$ and $\alpha \in \{-1, 0, 1\}$ for tuning of SPMSVM.

Next, we conduct comparison studies based on Example 1 under a variety of settings of (n, d, σ). The average computational time of training and the label prediction errors of testing based on 100 replications are reported in Table V. As seen, SPMSVM achieves the most accurate label prediction in all the considered settings. As the dimension d or the noise parameter σ increases, the prediction accuracy of all methods deteriorates, which is consistent with our simulation designs. We also observe that the prediction error of SPMSVM decreases and approaches the Bayes error as the sample size n grows, which indicates that SLS loss is Fisher consistent. In terms of training time, SPMSVM and REC are fairly comparable and they outperform the other MSVMs. On the other hand, Table VI exhibits the comparison results for Example 2 under the Gaussian kernel. SPMSVM is again the best classifier in terms of prediction accuracy, and it is computationally efficient.

Because only REC and SPMSVM can directly estimate the category probability and OVOSVM can heuristically provide some probabilistic outputs [46], [95], we present the estimated MAE and Brier score by these three MSVMs based on Examples 1 and 2 in Table VII. As seen, OVOSVM outperforms SPMSVM in several settings for linear learning of Example 1, while SPMSVM is uniformly better for nonlinear learning of Example 2, which implies that the heuristic method for probability estimation in OVOSVM is less efficient for datasets that are not linearly separable. It is important to recall that SPMSVM outperforms OVOSVM by a large margin in terms of the classification performance even for the linear cases, as previously shown in Table V. Between SPMVM and REC, it is observed that SPMSVM outperforms REC for both criterion in all the simulation settings. This is reasonable as REC is a special case of SPMSVM with $\gamma = 0.5$, and the results actually indicate that the optimal γ for all the simulated data is not 0.5. Moreover, the MAE is uniformly smaller than the corresponding Brier score as the exact distribution of data generating process is known and the true probability of category is available.

Our last simulation is intended to investigate the scalability of the proposed SPMSVM. The datasets are generated with the true signal linearly depending on a few covariates. The simulation settings are given as follows.

Example 3: We consider 10 categories and 10 covariates. Assume that $\Pr(Y = j) = 0.1$, and the first two covariates of $[X|Y = j]$ follows $\mathcal{N}((\cos(j\pi/5), \sin(j\pi/5))^\top, 0.04\boldsymbol{I}_2)$, where the 10 mean vectors are equally distributed on the unit circle. The other 8 covariates are i.i.d. $\mathcal{N}(0, 0.01)$. Let the sizes of training, tuning and testing sets be 100000, 100000 and 10000, respectively.

For each simulated data, we apply linear learning for all compared methods. In particular, we implement the linear SPMSVM using the `glmnet` package, as discussed in Section V-B. Similar to Example 1 and 2, we conduct 100 replications for different simulated dataset, and record the means of classification error, computational time for training a model, and probability estimation measures for those related methods. Because CSMSVM, GenSVM, RAMSVM and WWMSVM suffer from the memory or convergence issues, we only summarize the results for the other methods in Table VIII. As seen, SPMSVM enjoys the near-optimal performance comparing with the Bayes classifier, as well as more accurate estimated probability. It is evident that the running time of SPMSVM is considerably shorter than that of OVOSVM, indicating that the regression-based implementation for SPMSVM could scale very well to large data sizes.

B. Real Data Analysis

In this subsection, we demonstrate the SPMSVM via seven real datasets available from open data sources. A summary of

TABLE V

RESULTS OF LABEL PREDICTION ERROR AND TRAINING TIME FOR EXAMPLE 1. THE ESTIMATED BAYES ERROR FOR $\sigma = 0.5/0.7$ IS 0.0432/0.1534, RESPECTIVELY. THE BOLD NUMBERS INDICATE THE BEST RESULTS FOR THAT SETTING. THE NUMBERS IN PARENTHESES REPRESENT THE ESTIMATED STANDARD ERRORS

		σ= 0.5		σ= 0.7	
(n, d)	Methods	Error(%)	Time(s)	Error(%)	Time(s)
(100,10)	OVOSVM	12.14(0.0155)	0.0151(0.0046)	25.65(0.0191)	0.0161(0.0061)
	WWMSVM	11.81(0.0214)	0.0220(0.0066)	25.33(0.0252)	0.0248(0.0118)
	CSMSVM	13.06(0.0214)	0.0174(0.0077)	27.43(0.0228)	0.0325(0.0431)
	GenSVM	26.57(0.2049)	2.4393(2.0896)	37.77(0.1685)	0.4639(0.3671)
	RAMSVM	11.71(0.0236)	0.0805(0.0235)	23.64(0.0290)	0.0646(0.0224)
	REC	25.22(0.0794)	**0.0003**(0.0000)	32.41(0.0612)	**0.0003**(0.0000)
	SPMSVM	**10.29**(0.0195)	**0.0003**(0.0000)	**22.04**(0.0244)	**0.0003**(0.0000)
(100,50)	OVOSVM	23.47(0.0189)	0.0235(0.0081)	35.38(0.0178)	0.0236(0.0081)
	WWMSVM	31.38(0.0475)	0.0257(0.0091)	37.36(0.0400)	0.0254(0.0086)
	CSMSVM	38.89(0.0265)	0.0630(0.2644)	45.57(0.0212)	0.0485(0.1942)
	GenSVM	43.18(0.1328)	9.1188(6.9305)	50.27(0.1215)	8.2830(6.3901)
	RAMSVM	24.02(0.0276)	0.0484(0.0182)	32.83(0.0280)	0.0469(0.0152)
	REC	31.04(0.0573)	**0.0051**(0.0008)	36.72(0.0476)	**0.0053**(0.0010)
	SPMSVM	**22.58**(0.0259)	0.0064(0.0005)	**31.42**(0.0241)	0.0065(0.0008)
(500,10)	OVOSVM	6.54(0.0052)	0.0288(0.0042)	18.09(0.0065)	0.0365(0.0014)
	WWMSVM	5.74(0.0046)	0.0488(0.0163)	17.36(0.0072)	0.7079(5.5898)
	CSMSVM	5.94(0.0053)	0.1109(0.1717)	17.76(2.7063)	0.0057(9.7990)
	GenSVM	26.29(0.2759)	5.8145(5.1555)	32.94(0.2222)	1.5795(1.3247)
	RAMSVM	5.85(0.0051)	1.7899(0.4645)	17.01(0.0073)	1.2323(0.4350)
	REC	9.88(0.0339)	**0.0025**(0.0001)	20.22(0.0293)	**0.0026**(0.0001)
	SPMSVM	**5.62**(0.0043)	0.0026(0.0001)	**16.81**(0.0069)	**0.0026**(0.0001)
(500,50)	OVOSVM	13.34(0.0089)	0.0853(0.0213)	28.34(0.0108)	0.0984(0.0248)
	WWMSVM	11.10(0.0125)	0.0967(0.0293)	22.57(0.0218)	0.0907(0.0258)
	CSMSVM	12.72(0.0105)	0.1975(0.0745)	26.88(0.0133)	0.2469(0.1500)
	GenSVM	29.56(0.2385)	12.2821(9.1841)	39.63(0.1788)	10.7368(8.0889)
	RAMSVM	9.11(0.0076)	1.4632(0.3179)	20.07(0.0099)	1.2053(0.2997)
	REC	12.74(0.0310)	**0.020**(0.0024)	22.36(0.0257)	**0.0219**(0.0033)
	SPMSVM	**8.55**(0.0069)	0.0229(0.0022)	**19.65**(0.0085)	0.00237(0.0028)

TABLE VI

RESULTS OF LABEL PREDICTION ERROR AND TRAINING TIME FOR EXAMPLE 2. THE ESTIMATED BAYES ERROR FOR $\sigma = 0.3/0.4$ IS 0.2017/0.3386, RESPECTIVELY. THE BOLD NUMBERS INDICATE THE BEST RESULTS FOR THAT SETTING. THE NUMBERS IN PARENTHESES REPRESENT THE ESTIMATED STANDARD ERRORS

		σ = 0.3		σ = 0.4	
n	method	Error(%)	Time(s)	Error(%)	Time(s)
100	OVOSVM	42.75(0.0382)	0.0153(0.0003)	51.70(0.0296)	0.0113(0.0210)
	WWMSVM	35.16(0.0401)	0.0276(0.0043)	47.96(0.0361)	0.0201(0.0309)
	CSMSVM	38.72(0.0460)	0.0219(0.0070)	50.70(0.0326)	0.0146(0.0165)
	GenSVM	38.18(0.0307)	2.4571(5.7214)	50.53(0.0303)	1.2786(2.8758)
	RAMSVM	35.38(0.0263)	0.2829(0.0475)	47.22(0.0216)	0.1335(0.0311)
	REC	38.13(0.0243)	**0.0089**(0.0005)	48.65(0.0225)	**0.0056**(0.0103)
	SPMSVM	**35.00**(0.0251)	**0.0089**(0.0010)	**46.56**(0.0205)	**0.0056**(0.0102)
200	OVOSVM	31.73(0.0242)	**0.0221**(0.0035)	44.18(0.0201)	**0.0141**(0.0006)
	WWMSVM	29.11(0.0275)	0.0423(0.0140)	43.37(0.0337)	0.0313(0.0286)
	CSMSVM	33.45(0.0397)	0.0515(0.0394)	46.71(0.0287)	0.0306(0.0336)
	GenSVM	32.02(0.0202)	0.1378(0.0953)	45.77(0.0210)	0.9135(2.5911)
	RAMSVM	29.47(0.0176)	1.1826(0.1379)	42.70(0.0171)	0.5449(0.0955)
	REC	32.17(0.0204)	0.0768(0.0047)	44.01(0.0187)	0.0272(0.0032)
	SPMSVM	**29.05**(0.0200)	0.0779(0.0048)	**41.93**(0.0169)	0.0278(0.0033)

these datasets is shown in Table IX, where $n_{\min}/n_{\max}$ is the size of the minority/majority categories, respectively. The predictors of every dataset are standardized to have sample mean zero and standard deviation one. In our analysis, we randomly split each dataset into three subsets with equal size for training, tuning and testing. Based on our preliminary investigation, we find that linear classifiers are sufficient, and hence only carry out linear learning for all compared methods. Since the

TABLE VII

RESULTS OF PROBABILITY ESTIMATION ERRORS ON ALL SIMULATED EXAMPLES FOR OVOSVM, REC AND SPMSVM. THE BOLD NUMBERS INDICATE THE BEST RESULTS FOR THAT SETTING

Simulation		MAE			Brier Score		
		OVOSVM	REC	SPMSVM	OVOSVM	REC	SPMSVM
Example 1 (n, d, σ)	(100,10,0.5)	**0.0212**	0.0361	0.0247	**0.0458**	0.0994	0.0612
	(100,50,0.5)	0.0380	0.0363	**0.0277**	0.1069	0.1011	**0.0728**
	(500,10,0.5)	**0.0060**	0.0355	0.0235	**0.0174**	0.0986	0.0580
	(500,50,0.5)	**0.0154**	0.0356	0.0251	**0.0423**	0.0988	0.0614
	(100,10,0.7)	0.0231	0.0311	**0.0228**	**0.0733**	0.1017	0.0777
	(100,50,0.7)	0.0354	0.0315	**0.0255**	0.1147	0.1037	**0.0866**
	(500,10,0.7)	**0.0081**	0.0301	0.0210	**0.0443**	0.1005	0.0738
	(500,50,0.7)	**0.0191**	0.0302	0.0233	**0.0717**	0.1009	0.0786
Example 2 (n, σ)	(100,0.3)	0.0554	0.0478	**0.0394**	0.1431	0.1337	**0.1233**
	(100,0.4)	0.0500	0.0427	**0.0415**	0.1660	0.1607	**0.1587**
	(200,0.3)	0.0392	0.0422	**0.0305**	0.1172	0.1216	**0.1039**
	(200,0.4)	0.0383	0.0354	**0.0312**	0.1487	0.1464	**0.1423**

TABLE VIII

RESULTS OF LABEL PREDICTION ERROR, PROBABILITY ESTIMATION ERRORS AND TRAINING TIME FOR EXAMPLE 3. THE ESTIMATED BAYES ERROR IS 0.1221. THE NUMBERS IN PARENTHESES REPRESENT THE ESTIMATED STANDARD ERRORS

	Error (%)	MAE	Brier Score	Time (s)
OVOSVM	12.35(0.0031)	0.0200(0.0003)	0.1512(0.0006)	607.1286(155.4224)
REC	13.21(0.0045)	0.0145(9.1E-05)	0.0753(6.8E-06)	**8.5230**(1.7724)
SPMSVM	**12.22**(0.0034)	**0.0115**(0.0001)	**0.0647**(0.0004)	8.7630(1.5660)

TABLE IX

SUMMARY OF THE REAL DATASETS

	Name (source)	n	d	k	$n_{\min}$	$n_{\max}$
Dataset1	wine (UCI, [96])	178	13	3	48	71
Dataset2	seeds (UCI, [96])	210	7	3	70	70
Dataset3	svmguide2 (LibSVM, [82])	391	20	3	53	221
Dataset4	red wine quality (UCI, [96])	1599	11	6	10	681
Dataset5	abalone (UCI, [96])	4177	8	28	1	689
Dataset6	waveform (version1) (UCI, [96])	5000	21	3	1647	1696
Dataset7	waveform (version2) (UCI, [96])	5000	40	3	1653	1692

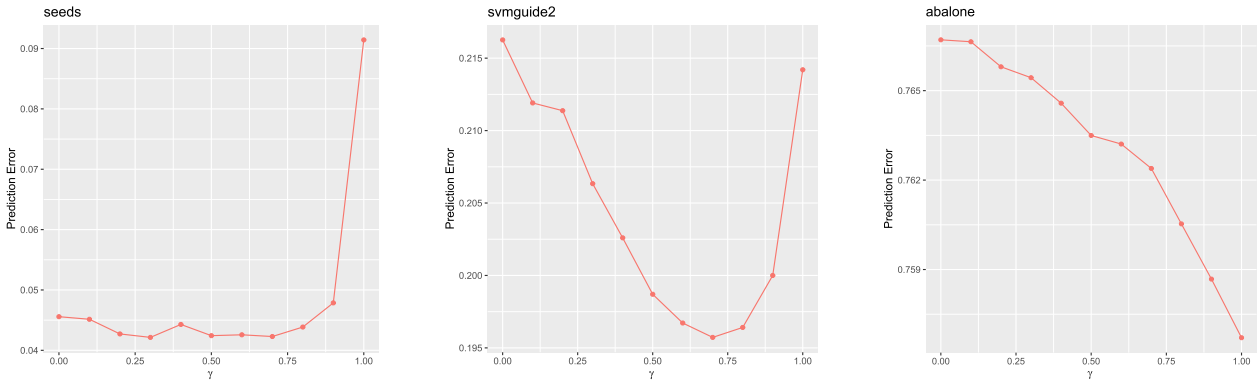

Fig. 3. Plots of the average prediction errors on the serval real datasets for SPMSVM.

true probabilities for all datasets remain unknown, we only record the averages and standard deviations of prediction error and computational time for training based on 100 replications. The other settings are analogous to those in Section VI-A.

The results of prediction error and training time on the real datasets are reported in Table X. It is evident that the SPMSVM achieves the most accurate prediction for almost all the datasets and the difference with RAMSVM on Dataset 4 is actually negligible. On the other hand, the simplex-based MSVMs, i.e., REC and SPMSVM, are more computationally efficient than the other MSVMs (note CSMSVM and RAMSVM do not converge within 48 hours for Datasets 5). Because REC is a special case of SPMSVM, it generally requires less computational time. It is important to note that prediction errors are generally large for Datasets 4 and 5 due to the imbalanced categories. Based on our investigation, the DNN-based methods also fail to provide a satisfactory solution for these two datasets and the prediction errors are respectively

TABLE X

RESULTS OF LABEL PREDICTION ERROR AND TRAINING TIME FOR THE REAL DATASETS. THE BOLD NUMBERS INDICATE THE BEST RESULTS FOR THAT SETTING. THE NUMBERS IN PARENTHESES REPRESENT THE ESTIMATED STANDARD ERRORS

		OVOSVM	WWMSVM	CSMSVM	GenSVM	RAMSVM	REC	SPMSVM
Dataset1	Error(%)	4.00	3.55	3.28	17.90	3.15	2.92	**2.63**
		(0.0242)	(0.0243)	(0.0253)	(0.2189)	(0.0237)	(0.0235)	(0.0231)
	Time(s)	0.0103	0.0124	0.0282	0.3752	0.0104	**0.0002**	**0.0002**
		(0.0002)	(0.0024)	(0.1422)	(0.3476)	(0.0048)	(0.0000)	(0.0000)
Dataset2	Error(%)	8.07	7.91	7.56	11.53	5.63	4.24	**4.21**
		(0.0308)	(0.0357)	(0.0339)	(0.1590)	(0.0279)	(0.0205)	(0.0209)
	Time(s)	0.0095	0.0097	0.0105	0.7419	0.0647	**0.0001**	**0.0001**
		(0.0009)	(0.0020)	(0.0069)	(0.3956)	(0.0239)	(0.0000)	(0.0000)
Dataset3	Error(%)	43.08	20.52	23.17	23.24	23.81	19.87	**19.57**
		(0.0340)	(0.0449)	(0.0744)	(0.0750)	(0.0448)	(0.0319)	(0.0327)
	Time(s)	0.0148	0.0138	2.9282	0.0200	0.2591	**0.0002**	**0.0002**
		(0.0005)	(0.0050)	(7.6993)	(0.0029)	(0.0051)	(0.0000)	(0.0000)
Dataset4	Error(%)	42.05	42.36	44.05	44.62	**41.26**	42.63	41.27
		(0.0194)	(0.0199)	(0.0218)	(0.0604)	(0.0184)	(0.0191)	(0.0188)
	Time(s)	0.0795	0.6653	4.0061	0.9274	17.7286	**0.0003**	**0.0003**
		(0.0058)	(3.7303)	(10.4527)	(0.5920)	(9.8677)	(0.0002)	(0.0000)
Dataset5	Error(%)	75.87	80.13	—	77.49	—	76.35	**75.67**
		(0.0114)	(0.0475)		(0.0390)		(0.0124)	(0.0120)
	Time(s)	0.4113	0.6582	—	5.7848	—	**0.0044**	0.0061
		(0.0228	(0.7175)		(1.8324)		(0.0005)	(0.0010)
Dataset6	Error(%)	13.88	13.43	13.49	19.82	13.10	14.29	**13.08**
		(0.0075)	(0.0071)	(0.0058)	(0.1452)	(0.0069)	(0.0002)	(0.0002)
	Time(s)	0.3115	2.2850	0.1434	0.3258	3.2705	**0.0068**	0.0090
		(0.0159)	(6.3520)	(0.0352)	(0.1206)	(1.8656)	(0.0000)	(0.0000)
Dataset7	Error(%)	14.98	13.97	14.01	25.95	13.87	14.79	**13.68**
		(0.0082)	(0.0069)	(0.0069)	(0.1865)	(0.0073)	(0.0004)	(0.0004)
	Time(s)	0.3015	1.2326	1.1259	5.7450	2.7100	**0.0069**	0.0075
		(0.0184)	(4.9568)	(3.2714)	(3.5786)	(0.6563)	(0.0000)	(0.0001)

0.6791 and 0.9199 if the standard multilayer perceptron is employed. Therefore, more advanced methods are needed to improve the performance in presence of imbalanced categories. For example, the weighted learning and cost-sensitive learning could be promising tools for imbalanced classification [97], [98], [99], [100], and see some survey papers, [101], [102], [103], for more advanced methods.

Recall that the classifier in [53] corresponds to SPMSVM with $\gamma = 0$ and the classifier in [43] (REC) is a special case of SPMSVM with $\gamma = 0.5$. To further illustrate the gained advantage of SPMSVM over these two existing methods, Figure 3 provides a visualization of the effect of γ in SPMSVM, whose pattern changes for three different datasets (Datasets 2, 3 and 5). According to Figure 3, these two methods yield the suboptimal performances for the three datasets, while the SPMSVM achieves more accurate prediction due to the flexible γ.

Overall, the effectiveness of the SPMSVM has been verified by the simulated and real datasets. The SPMSVM can provide the label prediction and probability estimation simultaneously, and enjoy the convenience of computation.

VII. CONCLUSION

This study provides a systematic analysis on simplex-based proximal MSVM. Two distinct features of the proposed SPMSVM are a flexible family of squared loss functions and a simplex-based framework. Compared to regular MSVMs, the general closed-form solution for SPMSVM was derived, which is established by solving a unconstrained linear system. Furthermore, the linear SPMSVM was converted to a weighted regression problem and hence is highly scalable by using the well-developed regression solvers. In addition to the label prediction, estimation of the category probability was also achieved by using the SPMSVM. Theoretically, the SPMSVM was shown to cover many existing MSVMs and own many statistical properties, some of which are rarely discussed in the literature of MSVMs. Numerical results demonstrated that the SPMSVM outperforms most existing MSVMs in terms of the computational speed and the prediction accuracy, and it can be a competitive and promising multicategory classifier in a variety of application domains.

One important future direction is to scale the SPMSVM to extremely large datasets (e.g., millions of observations). In our study, we established two possible implementations for SPMSVM: one is to directly use the closed-form solution (31) and the other is to apply the advanced solver for the regression-based formulation (33). Our suggestion is that when there are at most 10k observations, the solver using the closed-form solution is more suitable, while the regression-based implementation is preferable if the dataset contains over 10k observations. On the other hand, when the datasets are beyond the capacity of a single/center machine and there are communication bandwidth constraints, more advanced techniques have to be invoked. One possible way is to use the parallel computation where the data are stored at different machines in a distributed manner. In such cases, distributed statistical learning algorithms have to be developed. For more technical details, we refer readers to [104], [105], and [106] and a recent survey paper by [107].

APPENDIX A
TECHNICAL PROOFS

Lemma 1 implies that $\mathbf{W}$ has the full rank $k-1$.

Lemma 1: For the matrix $\mathbf{W}$, $\mathbf{W}\mathbf{W}^\top = \frac{k}{k-1}\mathbf{I}_{k-1}$.

Proof of Lemma 1: By the definition of $\mathbf{W}$, we know that

$$\mathbf{W}^\top\mathbf{W} = \frac{k}{k-1}\mathbf{I}_k - \frac{1}{k-1}\mathbf{1}_k\mathbf{1}_k^\top.$$

By left multiplying $\mathbf{W}$, we have $\mathbf{W}\mathbf{W}^\top\mathbf{W} = \frac{k}{k-1}\mathbf{W}$, which implies $(\mathbf{W}\mathbf{W}^\top - \frac{k}{k-1}\mathbf{I}_{k-1})\mathbf{W} = \mathbf{0}$. Since $\mathbf{W}$ has rank $k-1$, we can find $k-1$ independent columns as an invertible submatrix $\widetilde{\mathbf{W}}$. Hence, we know $(\mathbf{W}\mathbf{W}^\top - \frac{k}{k-1}\mathbf{I}_{k-1})\widetilde{\mathbf{W}} = \mathbf{0}$, which implies $\mathbf{W}\mathbf{W}^\top = \frac{k}{k-1}\mathbf{I}_{k-1}$. □

Proof of Proposition 1: For a fixed $\boldsymbol{x} \in \mathcal{X}$, let $\boldsymbol{g} = (g_1(\boldsymbol{x}), \ldots, g_k(\boldsymbol{x}))^\top$ and $\boldsymbol{f} = (f_1(\boldsymbol{x}), \ldots, f_{k-1}(\boldsymbol{x}))^\top$. Denote $S_1 = \{\boldsymbol{g} \in \mathbb{R}^k | \sum_{j=1}^k g_j = \mathbf{0}\}$ and $S_2 = \{\mathbf{W}^\top\boldsymbol{f} \in \mathbb{R}^k | \boldsymbol{f} \in \mathbb{R}^{k-1}\}$. We prove $S_1 = S_2$ in two directions.

- First, for any $\boldsymbol{g} \in S_1$, we can find $\boldsymbol{f} = (1-\frac{1}{k})\mathbf{W}\boldsymbol{g} \in \mathbb{R}^{k-1}$ such that $\mathbf{W}^\top\boldsymbol{f} = (1-\frac{1}{k})\mathbf{W}^\top\mathbf{W}\boldsymbol{g} = (\mathbf{I}_k - \frac{1}{k}\mathbf{1}_k\mathbf{1}_k^\top)\boldsymbol{g} = \boldsymbol{g} \in S_2$. Hence, $S_1 \subseteq S_2$ holds.
- Reversely, for any $\boldsymbol{g}' \in S_2$, then there exists a $\boldsymbol{f} \in \mathbb{R}^{k-1}$ such that $\boldsymbol{g}' = \mathbf{W}^\top\boldsymbol{f}$. Furthermore, we have $\sum_{j=1}^k g_j' = \mathbf{1}_k^\top\boldsymbol{g}' = \mathbf{1}_k^\top\mathbf{W}^\top\boldsymbol{f} = (\mathbf{W}\mathbf{1}_k)^\top\boldsymbol{a} = 0$, which implies that $\boldsymbol{g}' \in S_1$. So $S_2 \subseteq S_1$.

Thus, $S_1 = S_2$. By the arbitrariness of $\boldsymbol{x}$, we have $\mathcal{G} = \mathcal{G}'$. □

Proof of Proposition 2: We first show that $\mathcal{L}(\mathbf{u}, y) = \gamma(\alpha - \langle\mathbf{u}, \mathbf{w}_y\rangle)^2 + (1-\gamma)\sum_{j\neq y}(1 + \langle\mathbf{u}, \mathbf{w}_j\rangle)^2$ is 2-Nemitski loss function with $p = 2$.

$$\begin{aligned}
&\mathcal{L}(\mathbf{u}, y)\\
=&\gamma(\alpha - \langle\mathbf{u}, \mathbf{w}_y\rangle)^2 + (1-\gamma)\sum_{j\neq y}(1 + \langle\mathbf{u}, \mathbf{w}_j\rangle)^2\\
=&\gamma\alpha^2 + (1-\gamma)(k-1) - 2(\gamma\alpha + 1 - \gamma)\langle\mathbf{u}, \mathbf{w}_y\rangle\\
&+ \mathbf{u}^\top\Big(\frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\mathbf{w}_y\mathbf{w}_y^\top\Big)\mathbf{u}\\
\leq&\gamma\alpha^2 + (1-\gamma)(k-1) + |\gamma\alpha + 1 - \gamma|(1 + \|\mathbf{u}\|^2) + \frac{k}{k-1}\|\mathbf{u}\|^2\\
=&\gamma\alpha^2 + (1-\gamma)(k-1) + |\gamma\alpha + 1 - \gamma|\\
&+ \Big(|\gamma\alpha + 1 - \gamma| + \frac{k}{k-1}\Big)\|\mathbf{u}\|^2.
\end{aligned}$$

The matrix $\frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\mathbf{w}_y\mathbf{w}_y^\top$ has eigenvalues $\frac{k(1-\gamma)}{k-1}$ and $\frac{k(1-\gamma)}{k-1} + 2\gamma - 1$, smaller than $\frac{k}{k-1}$ for any $\gamma \in [0, 1]$.

Note that the Hessian matrix is $\nabla^2\mathcal{L}(\mathbf{u}, y) = \frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\mathbf{w}_y\mathbf{w}_y^\top \succeq \mathbf{0}$. Therefore, $\mathcal{L}(\mathbf{u}, y)$ is convex and continuous. Theorem 1 in the appendix of [32] ensures that $\mathcal{E} : L_2(\mathcal{P}_X) \mapsto \mathbb{R}_+$ is a well defined, convex and continuous functional. □

Proof of Proposition 3: Let $\boldsymbol{f}^* : \mathcal{X} \mapsto \mathbb{R}^{k-1}$ be the minimizer of $\mathcal{E}(\boldsymbol{f})$, then $\boldsymbol{f}^*$ satisfies $\nabla\mathcal{E}(\boldsymbol{f}^*) = \mathbf{0}$. Theorem 1 in the appendix of [32] also guarantees that $\boldsymbol{f}^*(\boldsymbol{x})$ minimizes the conditional $\mathcal{L}$-risk $\mathcal{S}_{\boldsymbol{x}}(\mathbf{u})$ for any $\boldsymbol{x} \in \mathcal{X}$. Denote $\mathbf{A}(\boldsymbol{x}) = \frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\sum_{j=1}^k P_j(\boldsymbol{x})\mathbf{w}_j\mathbf{w}_j^\top$ and $b = \gamma\alpha + 1 - \gamma$, we can write

$$\begin{aligned}
\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) &= \sum_{j=1}^k P_j\mathcal{L}(\mathbf{u}, j)\\
&= \mathbf{u}^\top\mathbf{A}(\boldsymbol{x})\mathbf{u} - 2b\sum_{j=1}^k\langle\mathbf{u}, P_j\mathbf{w}_j\rangle + \gamma\alpha^2 + (1-\gamma)(k-1).
\end{aligned}$$

Let $a_j = \frac{1}{(2\gamma-1)P_j + 1 - \gamma}$ for $j = 1, \ldots, k$. Setting the first derivatives to be zero gives

$$\Big(\sum_{j=1}^k \frac{2}{a_j}\mathbf{w}_j\mathbf{w}_j^\top\Big)\mathbf{u}^* - 2b\sum_{j=1}^k P_j\mathbf{w}_j = \mathbf{0}. \tag{34}$$

Since the matrix $\sum_{j=1}^k \frac{2}{a_j}\mathbf{w}_j\mathbf{w}_j^\top$ is positive definite, (34) has an unique solution $\boldsymbol{f}^*(\boldsymbol{x}) = \mathbf{u}^*$. Denote $s_j = \langle\mathbf{u}^*, \mathbf{w}_j\rangle$ for $j = 1, \ldots, k$. We can rewrite (34) as

$$\sum_{j=1}^k (s_j/a_j - bP_j)\mathbf{w}_j = \mathbf{0}.$$

By the property of $\mathbf{W}$, for $j = 1, \ldots, k$, we have

$$s_j/a_j - bP_j = C \Rightarrow s_j = a_j(bP_j + C).$$

Thanks to the condition $\sum_{j=1}^k s_j = 0$, we have

$$0 = \sum_{j=1}^k a_j(bP_j + C) = C\sum_{j=1}^k a_j + b\sum_{j=1}^k a_jP_j.$$

Thus, we get $C = -\frac{b\sum_{j=1}^k a_jP_j}{\sum_{j=1}^k a_j}$ and

$$s_j = ba_j\Big(P_j - \frac{\sum_{t=1}^k a_tP_t}{\sum_{t=1}^k a_t}\Big),\ j = 1, \ldots, k.$$

Solving the k linear equations $\langle\mathbf{u}^*, \mathbf{w}_j\rangle = s_j$, we have

$$\mathbf{u}^* = (\mathbf{W}\mathbf{W}^\top)^{-1}\sum_{j=1}^k s_j\mathbf{w}_j = \frac{k-1}{k}\sum_{j=1}^k s_j\mathbf{w}_j,$$

where the last equation holds by Lemma 1. When $\boldsymbol{x}$ varies in $\mathcal{X}$, we get the desired closed-form of $\boldsymbol{f}^*(\boldsymbol{x}) = \mathbf{u}^*$ at a fixed $\boldsymbol{x} \in \mathcal{X}$. Specifically, we have

$$\langle\mathbf{u}^*, \mathbf{w}_j\rangle = \begin{cases}(\alpha+1)(P_j - \frac{1}{k}), & \text{if } \gamma = 0.5;\\ \frac{b}{2\gamma-1}\Big(1 - \frac{ka_j}{\sum_{t=1}^k a_t}\Big), & \text{if } \gamma \neq 0.5.\end{cases} \tag{35}$$

Last, we show that $\boldsymbol{f}^* \in L_2(\mathcal{X}, P_X)$. Note that

$$\begin{aligned}
\|\boldsymbol{f}^*(\boldsymbol{x})\|^2 &= \Big(\frac{k-1}{k}\Big)^2\Big(\sum_{j=1}^k s_j\mathbf{w}_j\Big)^\top\Big(\sum_{j=1}^k s_j\mathbf{w}_j\Big)\\
&= \frac{k-1}{k}\sum_{t=1}^k s_t^2\\
&\leq \begin{cases}(\frac{k-1}{k})^2(\alpha+1)^2, & \text{if } \gamma = 0.5;\\ \frac{(k-1)^2b^2}{(2\gamma-1)^2}, & \text{if } \gamma \neq 0.5.\end{cases}
\end{aligned}$$

So $\boldsymbol{f}^*$ is almost surely bounded and belongs to $L_2(\mathcal{P}_X)$. □

Proof of Theorem 1: By Proposition 3 and $\gamma\alpha + 1 - \gamma > 0$, following (35), we can verify that $P_i > P_j \Longleftrightarrow \langle\boldsymbol{f}^*, \mathbf{w}_i\rangle >$

$\langle \boldsymbol{f}^*, \mathbf{w}_j\rangle$. Hence, we have $\operatorname{argmax}_j P_j = \operatorname{argmax}_j \langle \boldsymbol{f}^*, \mathbf{w}_j\rangle$, and Fisher consistency easily follows. □

Proof of Theorem 2: When $\gamma = \frac{1}{2}$, by Proposition 3, we know $\boldsymbol{f}^* = \frac{(k-1)(\alpha+1)}{k}\sum_{j=1}^k P_j \mathbf{w}_j$ and $s_j^* = (\alpha+1)P_j - \frac{\alpha+1}{k}$. Hence, we have $P_j = \frac{1}{\alpha+1}s_j^* + \frac{1}{k}$. Recall the condition (34), we know

$$\sum_{j=1}^k [(2\gamma-1)s_j^* P_j - bP_j + (1-\gamma)s_j^*]\mathbf{w}_j = \mathbf{0},$$

and $(2\gamma-1)s_j^* P_j - bP_j + (1-\gamma)s_j^* = C$ for any j. Let $d_j = (1-\gamma)s_j^*$ and $c_j = \frac{1}{(2\gamma-1)s_j^* - b}$. We get

$$\frac{1}{c_j}P_j + d_j = C \Rightarrow P_j = c_j(C - d_j).$$

Since $\sum_{j=1}^k P_j = 1$, we have $C = \frac{1+\sum_{j=1}^k c_j d_j}{\sum_{j=1}^k c_j}$ and

$$P_j = \frac{1+\sum_{t=1}^k c_t d_t}{\sum_{t=1}^k c_t} c_j - c_j d_j.$$

Plugging d_j's into P_j's and using some tedious algebra, the desired results are established. □

Proof of Theorem 3: Define $\mathbf{A} = \frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\sum_{j=1}^k P_j(\boldsymbol{x})\mathbf{w}_j\mathbf{w}_j^\top$, where the dependence on $\boldsymbol{x}$ is omitted. Assume $\lambda_{\min}(\mathbf{A})$ is the smallest eigenvalue of $\mathbf{A}$. Denote $\mathcal{X}_{\boldsymbol{f}} = \{\boldsymbol{x} \in \mathcal{X} | \mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x}) \neq \mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})\}$ and $b = \gamma\alpha + 1 - \gamma > 0$. By definition, the excess $\mathcal{L}$-risk is

$$\begin{aligned}
&\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^* \\
=&\int_{\mathcal{X}} \sum_{j=1}^k P_j(\boldsymbol{x})\{\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}), j) - \mathcal{L}(\boldsymbol{f}^*(\boldsymbol{x}), j)\}dP(\boldsymbol{x}) \\
=&\int_{\mathcal{X}} \Big\{ \sum_{j=1}^k 2b\langle \boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}), P_j(\boldsymbol{x})\mathbf{w}_j\rangle \\
&+ \boldsymbol{f}^\top(\boldsymbol{x})\mathbf{A}\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}^{*\top}(\boldsymbol{x})\mathbf{A}\boldsymbol{f}^*(\boldsymbol{x})\Big\}dP(\boldsymbol{x}) \\
=&\int_{\mathcal{X}} \Big\{ \boldsymbol{f}^\top(\boldsymbol{x})\mathbf{A}\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{f}^{*\top}(\boldsymbol{x})\mathbf{A}\boldsymbol{f}^*(\boldsymbol{x}) \\
&- 2\boldsymbol{f}^{*\top}(\boldsymbol{x})\mathbf{A}\boldsymbol{f}(\boldsymbol{x})\Big\}dP(\boldsymbol{x}) \\
=&\int_{\mathcal{X}} (\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}^*(\boldsymbol{x}))^\top \mathbf{A}(\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}^*(\boldsymbol{x}))dP(\boldsymbol{x}) \\
\geq&\lambda_{\min}(\mathbf{A}) \int_{\mathcal{X}} \|\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}^*(\boldsymbol{x})\|^2 dP(\boldsymbol{x}). \qquad (36)
\end{aligned}$$

(i) Case $\gamma = \frac{1}{2}$. Since $\mathbf{A} = \frac{k}{2(k-1)}\mathbf{I}_{k-1}$ and $\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^* = \frac{k}{2(k-1)}\int_{\mathcal{X}} \|\boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})\|^2 dP(\boldsymbol{x})$, the excess misclassification risk is

$$\begin{aligned}
&\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \\
=&\int_{\mathcal{X}_{\boldsymbol{f}}} \{P_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}(\boldsymbol{x}) - P_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}(\boldsymbol{x})\}dP(\boldsymbol{x}) \\
=&\int_{\mathcal{X}_{\boldsymbol{f}}} \frac{1}{\alpha+1}\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle dP(\boldsymbol{x}) \\
=&\frac{1}{\alpha+1}\int_{\mathcal{X}_{\boldsymbol{f}}} \{\langle \boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle \\
&+ \langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle\}dP(\boldsymbol{x}) \\
\leq&\frac{1}{\alpha+1}\int_{\mathcal{X}_{\boldsymbol{f}}} \langle \boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle dP(\boldsymbol{x}).
\end{aligned}$$

Moreover, using Jensen and Cauchy-Schwarz inequalities, we can write

$$\begin{aligned}
&\{\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*\}^2 \\
\leq&\frac{1}{(\alpha+1)^2}\int_{\mathcal{X}_{\boldsymbol{f}}} \{\langle \boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle\}^2 dP(\boldsymbol{x}) \\
\leq&\frac{2k}{(k-1)(\alpha+1)^2}\int_{\mathcal{X}_{\boldsymbol{f}}} \|\boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})\|^2 dP(\boldsymbol{x}) \\
=&\frac{4}{(\alpha+1)^2}\{\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*\}.
\end{aligned}$$

Taking square roots, the result for $\gamma = \frac{1}{2}$ in (16) follows.

In what follows, we consider the case $\gamma \neq \frac{1}{2}$. By Proposition 3, we know $\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_j\rangle = \frac{b}{2\gamma-1}\Big(1 - \frac{ka_j}{\sum_{t=1}^k a_t}\Big)$, where $a_j = \frac{1}{(2\gamma-1)P_j + 1 - \gamma}$ for $j = 1, \ldots, k$. By Theorem 2, we have $P_j(\boldsymbol{x}) = \Big(1 + \frac{k(1-\gamma)}{2\gamma-1}\Big)\frac{c_j}{\sum_{i=1}^k c_i} - \frac{1-\gamma}{2\gamma-1}$ with $c_j = \frac{1}{(2\gamma-1)\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_j\rangle - b}$ for $j = 1, \ldots, k$. One can verify that each $c_j = -\frac{kba_j}{\sum_{t=1}^k a_t} < 0$. The corresponding excess misclassification risk is

$$\begin{aligned}
&\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \\
=&\int_{\mathcal{X}_{\boldsymbol{f}}} \{P_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}(\boldsymbol{x}) - P_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}(\boldsymbol{x})\}dP(\boldsymbol{x}) \\
=&\int_{\mathcal{X}_{\boldsymbol{f}}} \Big(1 + \frac{k(1-\gamma)}{2\gamma-1}\Big)\frac{c_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}}{\sum_{t=1}^k c_t}\Big\{\frac{c_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}}{c_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}} - 1\Big\}dP(\boldsymbol{x}) \\
\leq&[2\gamma - 1 + k(1-\gamma)]\int_{\mathcal{X}_{\boldsymbol{f}}} \frac{\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle}{b - (2\gamma-1)\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}\rangle} dP(\boldsymbol{x}). \qquad (37)
\end{aligned}$$

Note that $\mathbf{A} = \frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\sum_{j=1}^k P_j(\boldsymbol{x})\mathbf{w}_j\mathbf{w}_j^\top = \sum_{j=1}^k \{\gamma P_j(\boldsymbol{x}) + (1-\gamma)(1-P_j(\boldsymbol{x}))\}\mathbf{w}_j\mathbf{w}_j^\top$, we know

$$\lambda_{\min}(\mathbf{A}) \geq \min_j \frac{k}{k-1}\{\gamma P_j(\boldsymbol{x}) + (1-\gamma)(1-P_j(\boldsymbol{x}))\} \geq \frac{k\delta}{k-1}. \qquad (38)$$

(ii) Case $\gamma \in [0, \frac{1}{2})$. Since $\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}\rangle \geq 0$ and $\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle \leq 0$, according to (37), the excess misclassification risk is

$$\begin{aligned}
&\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \\
\leq&[2\gamma - 1 + k(1-\gamma)]\int_{\mathcal{X}_{\boldsymbol{f}}} \frac{\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle}{b + (1-2\gamma)\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}\rangle} dP(\boldsymbol{x}) \\
\leq&\frac{2\gamma - 1 + k(1-\gamma)}{b}\int_{\mathcal{X}_{\boldsymbol{f}}} \{\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle\}dP(\boldsymbol{x}) \\
\leq&\frac{2\gamma - 1 + k(1-\gamma)}{b}\int_{\mathcal{X}_{\boldsymbol{f}}} \{\langle \boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}\rangle\}dP(\boldsymbol{x}).
\end{aligned}$$

Combining (36) and (38), we have

$$\begin{aligned}
&\{\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*\}^2 \\
\leq&\Big(\frac{2\gamma - 1 + k(1-\gamma)}{b}\Big)^2 \frac{2k}{k-1}\int_{\mathcal{X}_{\boldsymbol{f}}} \|\boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})\|^2 dP(\boldsymbol{x}) \\
\leq&\frac{2[2\gamma - 1 + k(1-\gamma)]^2}{\delta b^2}\{\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*\}.
\end{aligned}$$

(iii) Case $\gamma \in (\frac{1}{2}, 1]$. Under Assumption 2, we have $\frac{1}{1-\delta} \leq a_j = \frac{1}{\gamma P_j(\boldsymbol{x})+(1-\gamma)(1-P_j(\boldsymbol{x}))} \leq \frac{1}{\delta}$. By (35) and (37), the corresponding excess misclassification risk is

$$\begin{aligned}
&\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \\
\leq&[2\gamma - 1 + k(1-\gamma)] \int_{\mathcal{X}_{\boldsymbol{f}}} \frac{\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*(\boldsymbol{x})}} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})} \rangle}{b - (2\gamma-1)\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*(\boldsymbol{x})}} \rangle} dP(\boldsymbol{x}) \\
=&\frac{2\gamma - 1 + k(1-\gamma)}{b} \int_{\mathcal{X}_{\boldsymbol{f}}} \frac{\sum_{t=1}^k a_t}{k a_{\mathcal{C}_{\boldsymbol{f}^*(\boldsymbol{x})}}} \{\langle \boldsymbol{f}^*(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*(\boldsymbol{x})}} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})} \rangle\} dP(\boldsymbol{x}) \\
\leq&\frac{[2\gamma-1+k(1-\gamma)](1-\delta)}{b\delta} \int_{\mathcal{X}_{\boldsymbol{f}}} \{\langle \boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}^*(\boldsymbol{x})}} - \mathbf{w}_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})} \rangle\} dP(\boldsymbol{x}).
\end{aligned}$$

Combining (36) and (38), we have

$$\begin{aligned}
&\{\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*\}^2 \\
\leq&\Big(\frac{[2\gamma-1+k(1-\gamma)](1-\delta)}{b\delta}\Big)^2 \frac{2k}{k-1} \int_{\mathcal{X}_{\boldsymbol{f}}} \|\boldsymbol{f}^*(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})\|^2 dP(\boldsymbol{x}) \\
\leq&\frac{2[\{2\gamma - 1 + k(1-\gamma)\}(1-\delta)]^2}{b^2\delta^3} \{\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*\}.
\end{aligned}$$

Summarizing the results in Case (i), (ii) and (iii), the proof is complete. □

To prove Theorem 4, we first show the following lemma.

Lemma 2: The generalised Tsybakov condition is equivalent to that for all $\boldsymbol{f} \in L_2(\mathcal{P}_X)$:

$$P(\mathcal{X}_{\boldsymbol{f}}) \leq C_a \{\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*\}^{\frac{a}{a+1}},$$

where $C_a = (a+1)C^{\frac{1}{a+1}} a^{-\frac{a}{a+1}} > 0$ is a constant depending on a.

Proof of Lemma 2: Since SLS loss is Fisher consistent, we have $\mathcal{C}_{\boldsymbol{f}^*}$ is Bayes optimal, which implies that the index (1) in $P_{(1)}(\boldsymbol{x})$ is $\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})$. Denote $m_{\boldsymbol{f}}(\boldsymbol{x}) \triangleq P_{\mathcal{C}_{\boldsymbol{f}^*}(\boldsymbol{x})}(\boldsymbol{x}) - P_{\mathcal{C}_{\boldsymbol{f}}(\boldsymbol{x})}(\boldsymbol{x})$. Therefore, $m_{\boldsymbol{f}}(\boldsymbol{x}) \geq P_{(1)}(\boldsymbol{x}) - P_{(2)}(\boldsymbol{x})$, and further

$$\begin{aligned}
&\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \\
=&\int_{\mathcal{X}_{\boldsymbol{f}}} m_{\boldsymbol{f}}(\boldsymbol{x}) dP(\boldsymbol{x}) \\
\geq&\int_{\mathcal{X}_{\boldsymbol{f}}} m_{\boldsymbol{f}}(\boldsymbol{x}) \mathbb{1}(m_{\boldsymbol{f}}(\boldsymbol{x}) \geq t) dP(\boldsymbol{x}) \\
\geq&t\Big(\int_{\mathcal{X}} \mathbb{1}(m_{\boldsymbol{f}}(\boldsymbol{x}) \geq t) dP(\boldsymbol{x}) - \int_{\mathcal{X}\backslash\mathcal{X}_{\boldsymbol{f}}} \mathbb{1}(m_{\boldsymbol{f}}(\boldsymbol{x}) \geq t) dP(\boldsymbol{x})\Big) \\
\geq&t\Big(1 - \int_{\mathcal{X}} \mathbb{1}(m_{\boldsymbol{f}}(\boldsymbol{x}) < t) dP(\boldsymbol{x}) - \int_{\mathcal{X}\backslash\mathcal{X}_{\boldsymbol{f}}} 1 dP(\boldsymbol{x})\Big) \\
\geq&t(1 - Ct^a - P(\mathcal{X}\backslash\mathcal{X}_{\boldsymbol{f}})) = t(P(\mathcal{X}_{\boldsymbol{f}}) - Ct^a).
\end{aligned}$$

Now taking the minimum of the above bound with respect to t, we get $t^* = \Big(\frac{P(\mathcal{X}_{\boldsymbol{f}})}{C(a+1)}\Big)^{\frac{1}{a}}$. Finally plugging t^* in the bound, we get $\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* \geq \frac{a}{C^{\frac{1}{a}}(a+1)^{\frac{a+1}{a}}} \{P(\mathcal{X}_{\boldsymbol{f}})\}^{\frac{a+1}{a}}$. This shows that

$$P(\mathcal{X}_{\boldsymbol{f}}) \leq (\alpha+1)C^{\frac{1}{a+1}} a^{-\frac{a}{a+1}} \{\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*\}^{\frac{a}{a+1}}.$$

Define $C_a = (a+1)C^{\frac{1}{a+1}} a^{-\frac{a}{a+1}}$, then the desired result is established. □

Proof of Theorem 4: If $t \leq m_{\boldsymbol{f}}(\boldsymbol{x})$, then $t m_{\boldsymbol{f}}(\boldsymbol{x}) \leq m_{\boldsymbol{f}}^2(\boldsymbol{x})$ and therefore $m_{\boldsymbol{f}}(\boldsymbol{x}) \leq \frac{m_{\boldsymbol{f}}^2(\boldsymbol{x})}{t}$. Note that

$$\begin{aligned}
\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^* =& \int_{\mathcal{X}_{\boldsymbol{f}}} m_{\boldsymbol{f}}(\boldsymbol{x}) dP(\boldsymbol{x}) \\
=& \int_{\mathcal{X}_{\boldsymbol{f}}} m_{\boldsymbol{f}}(\boldsymbol{x}) \mathbb{1}(m_{\boldsymbol{f}}(\boldsymbol{x}) \leq t) dP(\boldsymbol{x}) \\
&+ \int_{\mathcal{X}_{\boldsymbol{f}}} m_{\boldsymbol{f}}(\boldsymbol{x}) \mathbb{1}(m_{\boldsymbol{f}}(\boldsymbol{x}) > t) dP(\boldsymbol{x}) \\
\leq& tP(\mathcal{X}_{\boldsymbol{f}}) + \frac{1}{t} \int_{\mathcal{X}_{\boldsymbol{f}}} m_{\boldsymbol{f}}^2(\boldsymbol{x}) dP(\boldsymbol{x}) \\
\leq& tC_a \{\mathcal{R}(\mathcal{C}_{\boldsymbol{f}}) - \mathcal{R}^*\}^{\frac{a}{a+1}} + \frac{C_\gamma^2}{t} \{\mathcal{E}(\boldsymbol{f}) - \mathcal{E}^*\}.
\end{aligned}$$

In the last inequality we used Theorem 3 and Lemma 2. Minimizing the right hand side of the above inequality over t, we get the result (18). □

Proof of Theorem 5: Let $V_1(u) = (\alpha - u)^2$ and $V_2(u) = (1-u)^2$. Note that

$$\begin{aligned}
\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) =& \sum_{j=1}^k \mathcal{L}(\mathbf{u}, j) P_j \\
=& \sum_{j=1}^k P_j \Big[\gamma V_1(\langle \mathbf{u}, \mathbf{w}_j \rangle) + (1-\gamma) \sum_{i \neq j} V_2(-\langle \mathbf{u}, \mathbf{w}_i \rangle)\Big] \\
=& \sum_{j=1}^k [\gamma P_j V_1(\langle \mathbf{u}, \mathbf{w}_j \rangle) + (1-\gamma)(1-P_j) V_2(-\langle \mathbf{u}, \mathbf{w}_j \rangle)].
\end{aligned}$$

By definition, $\Delta\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) = \mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) - \mathcal{S}_{\boldsymbol{x}}(\mathbf{u}^*)$. We can rewrite the RHS of the display as

$$\begin{aligned}
&\sum_{j=1}^k [\gamma P_j V_1(\langle \mathbf{u}, \mathbf{w}_j \rangle) + (1-\gamma)(1-P_j) V_2(-\langle \mathbf{u}, \mathbf{w}_j \rangle)] \\
-&\sum_{j=1}^k [\gamma P_j V_1(\langle \mathbf{u}^*, \mathbf{w}_j \rangle) + (1-\gamma)(1-P_j) V_2(-\langle \mathbf{u}^*, \mathbf{w}_j \rangle)].
\end{aligned}$$

With adding and subtracting, rearrange to obtain that the above display is equivalent to

$$\begin{aligned}
&(\mathbf{u} - \mathbf{u}^*)^\top \Big(\sum_{j=1}^k [\gamma P_j + (1-\gamma)(1-P_j)] \mathbf{w}_j \mathbf{w}_j^\top\Big) (\mathbf{u} - \mathbf{u}^*) \\
+&\sum_{j=1}^k [\gamma P_j V_1'(\langle \mathbf{u}^*, \mathbf{w}_j \rangle) - (1-\gamma)(1-P_j) V_2'(-\langle \mathbf{u}^*, \mathbf{w}_j \rangle)] \langle \mathbf{u} - \mathbf{u}^*, \mathbf{w}_j \rangle
\end{aligned}$$

In combination with the desired inequality, observe that the above is essentially RHS of Theorem 5 plus $\langle \mathbf{u}^* - \mathbf{u}, \sum_{j=1}^k [\gamma P_j V_1'(\langle \mathbf{u}^*, \mathbf{w}_j \rangle) - (1-\gamma)(1-P_j) V_2'(-\langle \mathbf{u}^*, \mathbf{w}_j \rangle)] \mathbf{w}_j \rangle$, so it suffices to show the latter, denoted by U, equals 0.

Since $\mathbf{u}^*$ minimizes $\mathcal{S}_x(\mathbf{u})$ for fixed $\boldsymbol{x}$, then we know that

$$\sum_{j=1}^k [\gamma P_j V_1'(\langle \mathbf{u}, \mathbf{w}_j \rangle) - (1-\gamma)(1-P_j) V_2'(-\langle \mathbf{u}, \mathbf{w}_j \rangle)] \mathbf{w}_j = \mathbf{0}.$$

Thus $U = 0$, and the desired result follows. □

Proof of Theorem 6: By Theorem 5, we have

$$\Delta\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}) = (\mathbf{u}-\mathbf{u}^*)^\top \Big(\sum_{j=1}^{k}[\gamma P_j + (1-\gamma)(1-P_j)]\mathbf{w}_j\mathbf{w}_j^\top\Big)(\mathbf{u}-\mathbf{u}^*) \le \frac{k}{k-1}(\mathbf{u}-\mathbf{u}^*)^\top(\mathbf{u}-\mathbf{u}^*).$$

Here, we use the fact that the largest eigenvalue of $\sum_{j=1}^{k}[\gamma P_j + (1-\gamma)(1-P_j)]\mathbf{w}_j\mathbf{w}_j^\top$ is smaller than $\frac{k}{k-1}\max_j\{\gamma P_j + (1-\gamma)(1-P_j)\} \le \frac{k}{k-1}$. Because $\tau_j \ll \mu,\ j = 1,\dots,k-1$ in Assumption 4, we can multiply both sides by n^{2q}, and take expectation to obtain

$$n^{2q}\Delta\mathcal{E}(\widehat{\boldsymbol{f}}^n) \le k\int_{X,Y}\Big|\sup_{1\le j\le k-1} T_j\Big|^2 dP(X,Y).$$

Because of Assumption 5, the RHS is bounded, and the desired result follows. □

Proof of Theorem 7: We find that

$$\begin{aligned}
&\Delta\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}^1) - \Delta\mathcal{S}_{\boldsymbol{x}}(\mathbf{u}^2)\\
=&\sum_{j=1}^{k}[\gamma P_j + (1-\gamma)(1-P_j)]\{\langle\mathbf{u}^1-\mathbf{u}^*,\mathbf{w}_j\rangle^2 - \langle\mathbf{u}^2-\mathbf{u}^*,\mathbf{w}_j\rangle^2\}\\
=&\sum_{j=1}^{k}[\gamma P_j + (1-\gamma)(1-P_j)]\langle\mathbf{u}^1-\mathbf{u}^2,\mathbf{w}_j\rangle\langle\mathbf{u}^1+\mathbf{u}^2-2\mathbf{u}^*,\mathbf{w}_j\rangle\\
\le&\|\mathbf{u}^1-\mathbf{u}^2\|_\infty\|\mathbf{u}^1+\mathbf{u}^2-2\mathbf{u}^*\|_\infty\cdot\sum_{j=1}^{k}[\gamma P_j + (1-\gamma)(1-P_j)]\,\|\mathbf{w}_j\|_1^2\\
\le&(k-1)^2\|\mathbf{u}^1-\mathbf{u}^2\|_\infty\|\mathbf{u}^1+\mathbf{u}^2-2\mathbf{u}^*\|_\infty.
\end{aligned}$$

The last inequality uses the fact that $\|\mathbf{w}_j\|_1 \le \sqrt{k-1},\ \forall j$. In particular, let $\mathbf{u}^1 = \widehat{\boldsymbol{f}}^n(\boldsymbol{x})$, $\mathbf{u}^2 = \boldsymbol{f}^H(\boldsymbol{x})$ and $\mathbf{u}^* = \boldsymbol{f}^*(\boldsymbol{x})$. By Assumption 6, $\widehat{f}_j - f_j^H$ is n^q consistent, and $|\widehat{f}_j + f_j^H - 2f_j^*| \le |\widehat{f}_j - f_j^H| + 2(|f_j^H| + |f_j^*|) \to 2(|f_j^H| + |f_j^*|)$. The rest of the proof is analogous to that of Theorem 5. □

Proof of Theorem 8: Plugging $\boldsymbol{g} = \mathbf{W}^\top\boldsymbol{f}$ into problem (21), then we have

$$\min_{\boldsymbol{f}\in\prod_{j=1}^{k-1}(\{1\}+H_K)}\ \frac{1}{n}\sum_{i=1}^{n}V(\mathbf{W}^\top\boldsymbol{f}(\boldsymbol{x}_i),y_i) + \lambda\sum_{j=1}^{k}\|\mathbf{w}_j^\top\boldsymbol{h}\|_{H_K}^2.$$

Define a matrix $\mathbf{H} \in \mathbb{R}^{(k-1)\times(k-1)}$ with the elements $\langle h_i, h_j\rangle_{H_K}$. We can rewrite the regularization as

$$\sum_{j=1}^{k}\|\mathbf{w}_j^\top\boldsymbol{h}\|_{H_K}^2 = \sum_{j=1}^{k}\mathbf{w}_j^\top\mathbf{H}\mathbf{w}_j = \mathrm{Tr}\Big(\sum_{j=1}^{k}\mathbf{w}_j^\top\mathbf{H}\mathbf{w}_j\Big) = \mathrm{Tr}\Big(\mathbf{H}\sum_{j=1}^{k}\mathbf{w}_j\mathbf{w}_j^\top\Big).$$

By Lemma 1, we know $\mathbf{W}\mathbf{W}^\top = \sum_{j=1}^{k}\mathbf{w}_j\mathbf{w}_j^\top = \frac{k}{k-1}\mathbf{I}_{k-1}$. Thus, the regularization is equivalent to $\frac{k}{k-1}\mathrm{Tr}(\mathbf{H}) = \sum_{j=1}^{k-1}\|h_j\|_{H_K}^2$. The formulation (22) is established. □

Proof of Theorem 9: Consider $f_j(\boldsymbol{x}) = c_j + h_j(\boldsymbol{x})$ with $h_j \in H_K$. Decompose $h_j(\cdot) = \sum_{i=1}^{n}b_{ij}K(\boldsymbol{x}_i,\cdot) + \rho_j(\cdot)$ for $j = 1,\dots,k-1$, where b_{ij}'s are some constants and $\rho_j(\cdot)$ is the element in the RKHS orthogonal to the span of $\{K(\boldsymbol{x}_i,\cdot),\ i = 1,\dots,n\}$. By the definition of the reproducing kernel $K(\cdot,\cdot)$, $\langle h_j, K(\boldsymbol{x}_i,\cdot)\rangle_{H_K} = h_j(\boldsymbol{x}_i)$ for $i = 1,\dots,n$. Then

$$\begin{aligned}
f_j(\boldsymbol{x}_i) &= c_j + h_j(\boldsymbol{x}_i) = c_j + \langle h_j, K(\boldsymbol{x}_i,\cdot)\rangle_{H_K}\\
&= c_j + \langle\sum_{s=1}^{n}b_{sj}K(\boldsymbol{x}_s,\cdot) + \rho_j(\cdot), K(\boldsymbol{x}_i,\cdot)\rangle_{H_K}\\
&= c_j + \sum_{s=1}^{n}b_{sj}K(\boldsymbol{x}_s,\boldsymbol{x}_i).
\end{aligned}$$

Thus the data fit functional in (23) does not depend on $\rho_j(\cdot)$ at all for $j = 1,\dots,k-1$. On the other hand, we have $\|h_j\|_{H_K}^2 = \sum_{s=1}^{n}\sum_{t=1}^{n}b_{sj}b_{tj}K(\boldsymbol{x}_s,\boldsymbol{x}_t) + \|\rho_j\|_{H_K}^2$ for $j = 1,\dots,k-1$. Let $h_j^*(\cdot) = \sum_{i=1}^{n}b_{ij}K(\boldsymbol{x}_i,\cdot)$ and $f_j^*(\boldsymbol{x}) = h_j^*(\cdot) + c_j$ for $j = 1,\dots,k-1$. Then $f_j^*(\boldsymbol{x}_i) = f_j(\boldsymbol{x}_i)$ and

$$\begin{aligned}
\sum_{j=1}^{k-1}\|h_j^*\|_{H_K}^2 &= \sum_{j=1}^{k-1}\sum_{s=1}^{n}\sum_{t=1}^{n}b_{sj}b_{tj}K(\boldsymbol{x}_s,\boldsymbol{x}_t)\\
&\le \sum_{j=1}^{k-1}\sum_{s=1}^{n}\sum_{t=1}^{n}b_{sj}b_{tj}K(\boldsymbol{x}_s,\boldsymbol{x}_t) + \sum_{j=1}^{k-1}\|\rho_j\|_{H_K}^2\\
&= \sum_{j=1}^{k-1}\|h_j\|_{H_K}^2.
\end{aligned}$$

Hence, the solution $\boldsymbol{f}^*(\boldsymbol{x})$ to minimize (23) can be expressed as $f_j^*(\boldsymbol{x}) = \sum_{i=1}^{n}b_{ij}K(\boldsymbol{x}_i,\boldsymbol{x}) + c_j$ for $j = 1,\dots,k-1$, where b_{ij}'s and c_j's are some constants. □

Proof of Theorem 10: To start with our proofs, we require some lemmas.

Lemma 3 ([108, Rademacher complexity]): Let $\mathcal{X}$ be any set, $\mathcal{F}$ a class of functions $f : \mathcal{X} \mapsto [0, M]$ and let X and $S = \{X_1,\dots,X_n\}$ be i.i.d. random variables with values in $\mathcal{X}$. Then for any $\theta > 0$, with probability at least $1-\theta$, the following holds for all $f \in \mathcal{F}$:

$$\mathbb{E}[f(X)] \le \frac{1}{n}\sum_{i=1}^{n}f(X_i) + \frac{2}{n}\mathbb{E}_S\Big\{\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{f\in\mathcal{F}}\sum_{i=1}^{n}\sigma_i f(X_i)\Big]\Big\} + M\sqrt{\frac{\log(1/\theta)}{2n}}.$$

Here $\boldsymbol{\sigma} = (\sigma_1,\dots,\sigma_n)$ are independent Rademacher variables, uniformly distributed on $\{-1,1\}$.

Lemma 4 ([109, Corollary 1]): Given samples $\{x_1,\dots,x_n\} \in X^n$. Let $H \subseteq \mathbb{R}^m$ be a Hilbert space and $\mathcal{F}$ be a class of functions $\boldsymbol{f} : X \mapsto H$ and let $h_i : H \mapsto \mathbb{R}$ be L-Lipschitz continuous with respect to the L_2-norm. Then

$$\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}}\sum_{i=1}^{n}\sigma_i h_i(\boldsymbol{f}(x_i))\Big] \le \sqrt{2}L\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}}\sum_{i=1}^{n}\sum_{j=1}^{m}\sigma_{ij}f_j(x_i)\Big],$$

where σ_{ij} is an independent doubly indexed Rademacher sequence and $f_j(x_i)$ is the j-th component of $\boldsymbol{f}(\boldsymbol{x}_i)$.

Lemma 5: Fix $t \in \{1,\dots,k\}$, assume that $\|\mathbf{u}\| \le C$, then $\mathcal{L}(\mathbf{u},t)$ is μ-Lipschitz continuous in $\mathbf{u}$ with respect to the L_2-norm, i.e., for any $\mathbf{u}_1,\mathbf{u}_2 \in \mathbb{R}^{k-1}$,

$$|\mathcal{L}(\mathbf{u}_1,t) - \mathcal{L}(\mathbf{u}_2,t)| \le \mu\,\|\mathbf{u}_1-\mathbf{u}_2\|.$$

with $\mu = \frac{2k}{k-1}C + 2|\gamma\alpha + 1 - \gamma|$. Furthermore, we have $\mathcal{L}(\mathbf{u}, t) \leq \gamma\alpha^2 + (1-\gamma)(k-1) + \mu C$.

Proof of Lemma 5: Recall the SLS loss $\mathcal{L}$ in (4), we note that

$$\begin{aligned}\mathcal{L}(\mathbf{u},t) =&\gamma(\alpha - \langle \mathbf{u}, \mathbf{w}_t\rangle)^2 + (1-\gamma)\sum_{j\neq t}(1+\langle \mathbf{u}, \mathbf{w}_j\rangle)^2\\ \nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u},t) =&2\gamma(\langle \mathbf{u}, \mathbf{w}_t\rangle - \alpha)\mathbf{w}_t + 2(1-\gamma)\sum_{j\neq t}(1+\langle \mathbf{u}, \mathbf{w}_j\rangle)\mathbf{w}_j\\ =&2\mathbf{A}\mathbf{u} + 2(\gamma\alpha + 1 - \gamma)\mathbf{w}_t,\end{aligned}$$

where $\mathbf{A} = \frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma - 1)\mathbf{w}_t\mathbf{w}_t^\top$. From the mean value theorem, there exists $\widetilde{\mathbf{u}}$ lies between $\mathbf{u}_1$ and $\mathbf{u}_2$ such that $\mathcal{L}(\mathbf{u}_1, t) - \mathcal{L}(\mathbf{u}_2, t) = \{\nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u}, t)\}|_{\mathbf{u}=\widetilde{\mathbf{u}}}^\top(\mathbf{u}_1 - \mathbf{u}_2)$. Combining with Cauchy-Schwartz inequality, we have

$$\begin{aligned}&|\mathcal{L}(\mathbf{u}_1,t) - \mathcal{L}(\mathbf{u}_2,t)|\\ \leq&\|\{\nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u},t)\}|_{\mathbf{u}=\widetilde{\mathbf{u}}}\|\,\|\mathbf{u}_1 - \mathbf{u}_2\|\\ =&2\,\|\mathbf{A}\widetilde{\mathbf{u}} + (\gamma\alpha + 1 - \gamma)\mathbf{w}_t\|\,\|\mathbf{u}_1 - \mathbf{u}_2\|\\ \leq&2\{\|\mathbf{A}\widetilde{\mathbf{u}}\| + |\gamma\alpha + 1 - \gamma|\,\|\mathbf{w}_t\|\}\,\|\mathbf{u}_1 - \mathbf{u}_2\|.\end{aligned}$$

The fact that $\lambda_{\max}(\mathbf{A}) \leq \frac{k}{k-1}$ implies $\|\mathbf{A}\mathbf{u}\| \leq \frac{k}{k-1}C$ for any $\|\mathbf{u}\| \leq C$. By simple algebra, the Lipschitz continuity of $\mathcal{L}$ is established. In particular, we have $\mathcal{L}(\mathbf{u}, t) - \mathcal{L}(\mathbf{0}, t) \leq \mu\,\|\mathbf{u}\| \leq \mu C$, thus $\mathcal{L}(\mathbf{u}, t)$ is bounded. □

Lemma 6: Suppose that Assumption 7 is met. Then $\|\boldsymbol{f}(\boldsymbol{x})\| \leq C_X\sqrt{\Lambda}$ for any $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{f} \in \mathcal{F}_\Lambda$.

Proof of Lemma 6: Under Assumption 7, using the properties of RKHS, we know

$$f_j(\boldsymbol{x}) = \langle f_j(\cdot), K(\boldsymbol{x}, \cdot)\rangle \leq \|f_j\|_{H_K}\|K(\boldsymbol{x}, \cdot)\|_{H_K} \leq C_X\|f_j\|_{H_K}.$$

Using Cauchy-Schwartz inequality, $\|\boldsymbol{f}(\boldsymbol{x})\| = \sqrt{\sum_{j=1}^{k-1} f_j^2(\boldsymbol{x})} \leq C_X\sqrt{\sum_{j=1}^{k-1}\|f_j\|_{H_K}^2} \leq C_X\sqrt{\Lambda}$. □

After the above preparations, we begin the main proof. Define a hypothesis space related to $\mathcal{F}_\Lambda$,

$$\mathcal{G} := \{g : (\mathcal{X}, \mathcal{Y}) \mapsto \mathbb{R} | g(\boldsymbol{x}, y) = \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}), y),\ \boldsymbol{f} \in \mathcal{F}_\Lambda\}.$$

By Lemmas 5 and 6, we know $\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}), y) \in [0, M]$. Using Lemma 3 for $\mathcal{G}$, we have

$$\begin{aligned}\mathbb{E}_{\mathcal{P}}[g(X,Y)] \leq&\frac{1}{n}\sum_{i=1}^n g(\boldsymbol{x}_i, y_i) + \frac{2}{n}\mathbb{E}_S\Big\{\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{g\in\mathcal{G}}\sum_{i=1}^n \sigma_i g(\boldsymbol{x}_i, y_i)\Big]\Big\}\\ &+ M\sqrt{\frac{\log(1/\theta)}{2n}}.\end{aligned} \tag{39}$$

By Lemmas 4 and Lemma 5, we know

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{g\in\mathcal{G}}\sum_{i=1}^n \sigma_i g(\boldsymbol{x}_i, y_i)\Big] =& \mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}_\Lambda}\sum_i \sigma_i\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i)\Big]\\ \leq& \sqrt{2}\mu\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}_\Lambda}\sum_{i=1}^n\sum_{j=1}^{k-1}\sigma_{ij}f_j(\boldsymbol{x}_i)\Big],\end{aligned} \tag{40}$$

where σ_{ij}'s are independent Rademacher variables. Moreover, we can simplify the RHS term of (40) as follows,

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}_\Lambda}\sum_{i=1}^n\sum_{j=1}^{k-1}\sigma_{ij}f_j(\boldsymbol{x}_i)\Big]\\ =&\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}_\Lambda}\sum_{i=1}^n\sum_{j=1}^{k-1}\sigma_{ij}\langle f_j(\cdot), K(\boldsymbol{x}_i, \cdot)\rangle_{H_K}\Big]\\ =&\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{\boldsymbol{f}\in\mathcal{F}_\Lambda}\sum_{j=1}^{k-1}\langle f_j(\cdot), \sum_{i=1}^n\sigma_{ij}K(\boldsymbol{x}_i, \cdot)\rangle_{H_K}\Big]\\ \leq&\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\boldsymbol{f}\in\mathcal{F}_\Lambda}\sqrt{\sum_{j=1}^{k-1}\|f_j\|_{H_K}^2}\sqrt{\sum_{j=1}^{k-1}\sum_{s=1}^n\sum_{t=1}^n\sigma_{is}\sigma_{it}K(\boldsymbol{x}_s, \boldsymbol{x}_t)}\right]\\ \leq&\sqrt{\Lambda}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sqrt{\sum_{j=1}^{k-1}\sum_{s=1}^n\sum_{t=1}^n\sigma_{is}\sigma_{it}K(\boldsymbol{x}_s, \boldsymbol{x}_t)}\right].\end{aligned}$$

Using Jensen's inequality and Assumption 7, we have

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{\sigma}}\left[\sqrt{\sum_{j=1}^{k-1}\sum_{s=1}^n\sum_{t=1}^n\sigma_{is}\sigma_{it}K(\boldsymbol{x}_s, \boldsymbol{x}_t)}\right]\\ \leq&\sqrt{\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sum_{j=1}^{k-1}\sum_{s=1}^n\sum_{t=1}^n\sigma_{is}\sigma_{it}K(\boldsymbol{x}_s, \boldsymbol{x}_t)\Big]}\\ =&\sqrt{\sum_{j=1}^{k-1}\sum_{i=1}^n K(\boldsymbol{x}_i, \boldsymbol{x}_i)} \leq C_X\sqrt{n(k-1)}.\end{aligned}$$

Plugging these results in (40), we have

$$\mathbb{E}_{\boldsymbol{\sigma}}\Big[\sup_{g\in\mathcal{G}}\sum_{i=1}^n\sigma_i g(\boldsymbol{x}_i, y_i)\Big] \leq \mu C_X\sqrt{2n(k-1)\Lambda}.$$

Combining (39), by the definition of $\mathcal{G}$, we establish the desired result in Theorem 10. □

Proof of Theorem 11: Note that Assumption 1 is used to guarantee the existence of $\boldsymbol{f}^*$. We need to show that, for any $\epsilon > 0$, there exists $\boldsymbol{f}^\epsilon \in \mathcal{H}$ such that

$$\Delta\mathcal{E}(\boldsymbol{f}^\epsilon) = \mathcal{E}(\boldsymbol{f}^\epsilon) - \mathcal{E}^* < \epsilon. \tag{41}$$

Using the fact in Theorem 5, we know

$$\begin{aligned}\Delta\mathcal{E}(\boldsymbol{f}^\epsilon) =& \int_{\mathcal{X}}\Delta\mathcal{S}_{\boldsymbol{x}}(\boldsymbol{f}^\epsilon(\boldsymbol{x}))dP(\boldsymbol{x})\\ \leq& \frac{k}{k-1}\int_{\mathcal{X}}\|\boldsymbol{f}^\epsilon(\boldsymbol{x}) - \boldsymbol{f}^*(\boldsymbol{x})\|^2 dP(\boldsymbol{x})\\ \leq& k\int_{\mathcal{X}}\sup_{1\leq j\leq k-1}\|f_j^\epsilon(\boldsymbol{x}) - f_j^*(\boldsymbol{x})\|_\infty^2 dP(\boldsymbol{x})\\ \leq& k\sup_{1\leq j\leq k-1}\|f_j^\epsilon - f_j^*\|_\infty^2.\end{aligned} \tag{42}$$

Since $\boldsymbol{f}^*$ is measurable in a compact input space $\mathcal{X}$, by Lusin's theorem, there exists a continuous vector-valued function $\widetilde{\boldsymbol{f}} = (\widetilde{f}_1, \ldots, \widetilde{f}_{k-1})$ such that

$$\widetilde{f}_j \in \mathfrak{C}(\mathcal{X}) \text{ and } \|\widetilde{f}_j - f_j^*\|_\infty < \frac{\epsilon}{2k}, \qquad j = 1, \ldots, k-1. \tag{43}$$

Note that $\widetilde{\boldsymbol{f}}$ is also continuous. The definition of the universal kernel implies the existence of a function $\boldsymbol{f}^\epsilon \in \mathcal{H}$ such that

$$\|f_j^\epsilon - \widetilde{f}_j\|_\infty < \frac{\epsilon}{2k}, \qquad j = 1, \ldots, k-1. \tag{44}$$

By combining expressions (42)–(44), we obtain inequality (41). □

Proof of Theorem 12: Define $\widehat{\mathcal{E}}_n(\boldsymbol{f}) = \frac{1}{n}\sum_{i=1}^n \mathcal{L}(\boldsymbol{f}(\boldsymbol{x}_i), y_i)$ and $\mathcal{L}_0 = \mathcal{L}(\mathbf{0}, y) = \gamma\alpha^2 + (1-\gamma)(k-1) < \infty$ for notational convenience. The objective function in (27) can be simplified as

$$J(\boldsymbol{f}) = \widehat{\mathcal{E}}_n(\boldsymbol{f}) + \lambda_n \Omega(\boldsymbol{f}).$$

Observe that $J(\widehat{\boldsymbol{f}}^n) \le J(\mathbf{0}) \le \mathcal{L}_0$. Therefore $\lambda_n\Omega(\widehat{\boldsymbol{f}}^n) \le \mathcal{L}_0 - \widehat{\mathcal{E}}_n(\widehat{\boldsymbol{f}}^n) \le \mathcal{L}_0$ and hence $\Omega(\widehat{\boldsymbol{f}}^n) \le \frac{\mathcal{L}_0}{\lambda_n}$. Set $\Lambda_n = \frac{\mathcal{L}_0}{\lambda_n}$, and we have $\widehat{\boldsymbol{f}}^n \in \mathcal{F}_{\Lambda_n}$, as defined in (26). Let $\epsilon > 0$. By the Borel-Cantelli Lemma it suffices to show

$$\sup_{n\ge 0} P(\mathcal{E}(\widehat{\boldsymbol{f}}^n) - \mathcal{E}^* \ge \epsilon) < \infty.$$

By Theorem 11, we can fix a $\boldsymbol{f}^\epsilon \in \mathcal{H}$ such that $\mathcal{E}(\boldsymbol{f}^\epsilon) < \mathcal{E}^* + \frac{\epsilon}{2}$. Note that $\boldsymbol{f}^\epsilon \in \mathcal{F}_{\Lambda_n}$ for n sufficiently large. From Theorem 10, for such large n and with probability at least $1 - \frac{\theta}{2}$ w.r.t. the training data,

$$\begin{aligned}
\mathcal{E}(\widehat{\boldsymbol{f}}^n) \le& \widehat{\mathcal{E}}_n(\widehat{\boldsymbol{f}}^n) + 2C_X\mu_n\sqrt{\frac{2(k-1)\Lambda_n}{n}} + M_n\sqrt{\frac{\log(2/\theta)}{n}} \\
\le& \widehat{\mathcal{E}}_n(\boldsymbol{f}^\epsilon) + \lambda_n\Omega(\boldsymbol{f}^\epsilon) - \lambda_n\Omega(\widehat{\boldsymbol{f}}^n) + 2C_X\mu_n\sqrt{\frac{2(k-1)\Lambda_n}{n}} \\
&+ M_n\sqrt{\frac{\log(2/\theta)}{n}} \\
\le& \widehat{\mathcal{E}}_n(\boldsymbol{f}^\epsilon) + \lambda_n\Omega(\boldsymbol{f}^\epsilon) + 2C_X\mu_n\sqrt{\frac{2(k-1)\Lambda_n}{n}} \\
&+ M_n\sqrt{\frac{\log(2/\theta)}{n}},
\end{aligned} \tag{45}$$

where $\mu_n = \frac{2k}{k-1}C_X\sqrt{\Lambda_n} + 2|\gamma\alpha + 1 - \gamma|$ and $M_n = \gamma\alpha^2 + (1-\gamma)(k-1) + C_X\mu_n\sqrt{\Lambda_n}$. Using Theorem 10 and the standarad symmetrization technique [110], the following holds with probability at least $1 - \frac{\theta}{2}$,

$$\widehat{\mathcal{E}}_n(\boldsymbol{f}^\epsilon) \le \mathcal{E}(\boldsymbol{f}^\epsilon) + 2C_X\mu_n\sqrt{\frac{2(k-1)\Lambda_n}{n}} + M_n\sqrt{\frac{\log(2/\theta)}{n}}. \tag{46}$$

Combining (45) and (46), we have

$$\mathcal{E}(\widehat{\boldsymbol{f}}^n) \le \mathcal{E}(\boldsymbol{f}^\epsilon) + \lambda_n\Omega(\boldsymbol{f}^\epsilon) + 4C_X\mu_n\sqrt{\frac{2(k-1)\Lambda_n}{n}} + 2M_n\sqrt{\frac{\log(2/\theta)}{n}}.$$

Take $\theta = n^{-2}$, and let N be such that $n \ge N$ implies that both $\boldsymbol{f}^\epsilon \in \mathcal{F}_{\Lambda_n}$ and

$$\lambda_n\Omega(\boldsymbol{f}^\epsilon) + 4C_X\mu_n\sqrt{\frac{2(k-1)\Lambda_n}{n}} + 2M_n\sqrt{\frac{\log(2/\theta)}{n}} \le \frac{\epsilon}{2}.$$

Then for $n \ge N$, with probability $1 - \frac{1}{n^2}$,

$$\mathcal{E}(\widehat{\boldsymbol{f}}^n) \le \mathcal{E}(\boldsymbol{f}^\epsilon) + \frac{\epsilon}{2} \le \mathcal{E}^* + \epsilon.$$

Therefore,

$$\sup_{n\ge 0} P(\mathcal{E}(\widehat{\boldsymbol{f}}^n) - \mathcal{E}^* \ge \epsilon) \le N - 1 + \sum_{n\ge N}\frac{1}{n^2} < \infty. \qquad \square$$

APPENDIX B
DETAILS ON (5)

By using the fact $\sum_{j=1}^k \mathbf{w}_j\mathbf{w}_j^\top = \mathbf{W}\mathbf{W}^\top = \frac{k}{k-1}\mathbf{I}_{k-1}$ and $\sum_{j=1}^k \mathbf{w}_j = \mathbf{0}$, we have

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{f}(\boldsymbol{x}), y) \\
=&\frac{1}{2}(\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y\rangle)^2 - \frac{1}{(k-1)}\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y\rangle \\
&+ \frac{1}{2}\sum_{j\ne y}(\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j\rangle)^2 + \sum_{j\ne y}\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j\rangle + \text{Constant} \\
=&\frac{1}{2}\sum_{j=1}^k(\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_j\rangle)^2 - \frac{k}{(k-1)}\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y\rangle + \text{Constant} \\
=&\frac{1}{2}\boldsymbol{f}(\boldsymbol{x})^\top\Big(\sum_{j=1}^k \mathbf{w}_j\mathbf{w}_j^\top\Big)\boldsymbol{f}(\boldsymbol{x}) - \frac{k}{(k-1)}\langle \boldsymbol{f}(\boldsymbol{x}), \mathbf{w}_y\rangle + \text{Constant} \\
=&\frac{k}{2(k-1)}\left\|\boldsymbol{f}(\boldsymbol{x}) - \mathbf{w}_y\right\|^2 + \text{Constant}.
\end{aligned}$$

APPENDIX C
DERIVATION OF (31)

From the equation (30), we know

$$\begin{aligned}
\vec{\vec{\mathbf{B}}} =& \Big(\frac{1}{n}\sum_{i=1}^n\sum_{j=1}^k \tau_{ij}(\mathbf{w}_j\mathbf{w}_j^\top) \otimes (\widetilde{\mathbf{K}}_i\widetilde{\mathbf{K}}_i^\top) + \lambda\mathbf{I}_{k-1}\otimes\mathbf{G}\Big)^{-1} \\
&\times \Big(\frac{1}{n}\sum_{i=1}^n\sum_{j=1}^k \tau_{ij}z_{ij}\mathbf{w}_j \otimes \widetilde{\mathbf{K}}_i\Big).
\end{aligned} \tag{47}$$

By Lemma 1, we know $\frac{k}{k-1}\mathbf{I}_{k-1} = \mathbf{W}\mathbf{W}^\top = \sum_{j=1}^k \mathbf{w}_j\mathbf{w}_j^\top$. Note that

$$\begin{aligned}
\sum_{j=1}^k \tau_{ij}\mathbf{w}_j\mathbf{w}_j^\top &= \gamma\mathbf{w}_{y_i}\mathbf{w}_{y_i}^\top + (1-\gamma)\sum_{j\ne y_i}\mathbf{w}_j\mathbf{w}_j^\top \\
&= \gamma\mathbf{w}_{y_i}\mathbf{w}_{y_i}^\top + (1-\gamma)\Big(\frac{k}{k-1}\mathbf{I}_{k-1} - \mathbf{w}_{y_i}\mathbf{w}_{y_i}^\top\Big) \\
&= \frac{k(1-\gamma)}{k-1}\mathbf{I}_{k-1} + (2\gamma-1)\mathbf{w}_{y_i}\mathbf{w}_{y_i}^\top \triangleq \mathbf{\Lambda}_i,
\end{aligned}$$

and

$$\begin{aligned}
\sum_{j=1}^k \tau_{ij}z_{ij}\mathbf{w}_j &= \gamma\alpha\mathbf{w}_{y_i} - (1-\gamma)\sum_{j\ne y_i}\mathbf{w}_j \\
&= \gamma\alpha\mathbf{w}_{y_i} - (1-\gamma)(-\mathbf{w}_{y_i}) \\
&= (\gamma\alpha + 1 - \gamma)\mathbf{w}_{y_i} = b\mathbf{w}_{y_i}.
\end{aligned}$$

Then we can rewrite (47) as a compact form (31).

ACKNOWLEDGMENT

The authors would like to thank the associate editor and two anonymous referees for the helpful comments which have led to a significant improvement of the paper.

REFERENCES

[1] J. Fan, L. Wu, X. Ma, H. Zhou, and F. Zhang, "Hybrid support vector machines with heuristic algorithms for prediction of daily diffuse solar radiation in air-polluted regions," *Renew. Energy*, vol. 145, pp. 2034–2045, Jan. 2020.

[2] W. Mo and Y. Liu, "Supervised learning," in *Wiley StatsRef: Statistics Reference Online*. Atlanta, GA, USA: American Cancer Society, 2021, pp. 1–20.

[3] B. Schölkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.

[4] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[6] E. Byvatov, U. Fechner, J. Sadowski, and G. Schneider, "Comparison of support vector machine and artificial neural network systems for drug/nondrug classification," *J. Chem. Inf. Comput. Sci.*, vol. 43, no. 6, pp. 1882–1889, Nov. 2003.

[7] R. P. R. Priya and P. Aruna, "SVM and neural network based diagnosis of diabetic retinopathy," *Int. J. Comput. Appl.*, vol. 41, no. 1, pp. 6–12, Mar. 2012.

[8] P. Liu, K.-K. R. Choo, L. Wang, and F. Huang, "SVM or deep learning? A comparative study on remote sensing image classification," *Soft Comput.*, vol. 21, no. 23, pp. 7053–7065, 2017.

[9] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annu. Workshop Comput. Learn. Theory (COLT)*, 1992, pp. 144–152.

[10] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

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

[12] X. Shen, G. C. Tseng, X. Zhang, and W. H. Wong, "On ψ-learning," *J. Amer. Stat. Assoc.*, vol. 98, no. 463, pp. 724–734, 2003.

[13] K.-W. Chang, C.-J. Hsieh, and C.-J. Lin, "Coordinate descent method for large-scale L2-loss linear support vector machines," *J. Mach. Learn. Res.*, vol. 9, pp. 1369–1398, Jun. 2008.

[14] X. Huang, L. Shi, and J. A. K. Suykens, "Ramp loss linear programming support vector machine," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2185–2211, 2014.

[15] X. Huang, L. Shi, and J. A. K. Suykens, "Support vector machine classifier with pinball loss," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 984–997, May 2014.

[16] F. Abramovich and M. Pensky, "Classification with many classes: Challenges and pluses," *J. Multivariate Anal.*, vol. 174, Nov. 2019, Art. no. 104536.

[17] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *J. Artif. Intell. Res.*, vol. 2, no. 1, pp. 263–286, 1995.

[18] V. Vapnik, *Statistical Learning Theory*. Hoboken, NJ, USA: Wiley, 1998.

[19] Y. Lee, Y. Lin, and G. Wahba, "Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data," *J. Amer. Statist. Assoc.*, vol. 99, no. 465, pp. 67–81, 2004.

[20] Y. Liu, "Fisher consistency of multicategory support vector machines," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2007, pp. 291–298.

[21] Y. Liu and M. Yuan, "Reinforced multicategory support vector machines," *J. Comput. Graph. Statist.*, vol. 20, no. 4, pp. 901–919, 2011.

[22] C. Zhang and Y. Liu, "Multicategory large-margin unified machines," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 1349–1386, 2013.

[23] J. Weston and C. Watkins, "Support vector machines for multi-class pattern recognition," in *Proc. Eur. Symp. Artif. Neural Netw.*, 1999, pp. 219–224.

[24] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, Mar. 2001.

[25] Y. Liu and X. Shen, "Multicategory ψ-learning," *J. Amer. Stat. Assoc.*, vol. 101, no. 474, pp. 500–509, 2006.

[26] P. J. F. Groenen, G. Nalbantov, and J. C. Bioch, "SVM-Maj: A majorization approach to linear support vector machines with different Hinge errors," *Adv. Data Anal. Classification*, vol. 2, no. 1, pp. 17–43, Apr. 2008.

[27] G. J. Van Den Burg and P. J. Groenen, "GenSVM: A generalized multiclass support vector machine," *J. Mach. Learn. Res.*, vol. 17, no. 224, pp. 1–42, 2016.

[28] C. Zhang, Y. Liu, J. Wang, and H. Zhu, "Reinforced angle-based multicategory support vector machines," *J. Comput. Graph. Statist.*, vol. 25, no. 3, pp. 806–825, Jul. 2016.

[29] G. Van Den Burg, "Algorithms for multiclass classification and regularized regression," Ph.D. dissertation, Dept. Erasmus Res. Inst. Manage., Erasmus Univ. Rotterdam, Rotterdam, The Netherlands, 2018.

[30] S. I. Hill and A. Doucet, "A framework for kernel-based multi-category classification," *J. Artif. Intell. Res.*, vol. 30, pp. 525–564, Dec. 2007.

[31] K. Lange and T. Tong Wu, "An MM algorithm for multicategory vertex discriminant analysis," *J. Comput. Graph. Statist.*, vol. 17, no. 3, pp. 527–544, Sep. 2008.

[32] Y. Mroueh, T. Poggio, L. Rosasco, and J.-J. Slotine, "Multiclass learning with simplex coding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2789–2797.

[33] C. Zhang and Y. Liu, "Multicategory angle-based large-margin classification," *Biometrika*, vol. 101, no. 3, pp. 625–640, Sep. 2014.

[34] S. Fu, S. Zhang, and Y. Liu, "Adaptively weighted large-margin angle-based classifiers," *J. Multivariate Anal.*, vol. 166, pp. 282–299, Jul. 2018.

[35] C. Zhang, M. Pham, S. Fu, and Y. Liu, "Robust multicategory support vector machines using difference convex algorithm," *Math. Program.*, vol. 169, no. 1, pp. 277–305, May 2018.

[36] S. Fu, Q. He, S. Zhang, and Y. Liu, "Robust outcome weighted learning for optimal individualized treatment rules," *J. Biopharmaceutical Statist.*, vol. 29, no. 4, pp. 606–624, Jul. 2019.

[37] C. Qian, Q. Tran-Dinh, S. Fu, C. Zou, and Y. Liu, "Robust multicategory support matrix machines," *Math. Program.*, vol. 176, nos. 1–2, pp. 429–463, 2019.

[38] C. Zhang, J. Chen, H. Fu, X. He, Y.-Q. Zhao, and Y. Liu, "Multicategory outcome weighted margin-based learning for estimating individualized treatment rules," *Statistica Sinica*, vol. 30, pp. 1857–1879, May 2020.

[39] Y. Yang, Y. Guo, and X. Chang, "Angle-based cost-sensitive multicategory classification," *Comput. Statist. Data Anal.*, vol. 156, Apr. 2021, Art. no. 107107.

[40] Y. Fan, X. Lu, Y. Liu, and J. Zhao, "Angle-based hierarchical classification using exact label embedding," *J. Amer. Stat. Assoc.*, vol. 117, no. 538, pp. 704–717, Apr. 2022.

[41] S. Fu, P. Chen, Y. Liu, and Z. Ye, "Simplex-based multinomial logistic regression with diverging number of categories and covariates," *Statistica Sinica*, 2022, doi: 10.5705/ss.202021.0082.

[42] Y. Liu, H. H. Zhang, and Y. Wu, "Hard or soft classification? Large-margin unified machines," *J. Amer. Stat. Assoc.*, vol. 106, no. 493, pp. 166–177, Mar. 2011.

[43] C. Zhang et al., "REC: Fast sparse regression-based multicategory classification," *Statist. Its Interface*, vol. 10, no. 2, pp. 175–185, 2017.

[44] J. D. Malley, J. Kruppa, A. Dasgupta, K. G. Malley, and A. Ziegler, "Probability machines: Consistent probability estimation using nonparametric learning machines," *Methods Inf. Med.*, vol. 51, no. 1, pp. 74–81, 2012.

[45] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Adv. Large Margin Classifiers*, vol. 10, no. 3 pp. 61–74, Mar. 1999.

[46] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multiclass classification by pairwise coupling," *J. Mach. Learn. Res.*, vol. 5, pp. 975–1005, Aug. 2004.

[47] H.-T. Lin, C.-J. Lin, and R. C. Weng, "A note on Platt's probabilistic outputs for support vector machines," *Mach. Learn.*, vol. 68, no. 3, pp. 267–276, 2007.

[48] J. Wang, X. Shen, and Y. Liu, "Probability estimation for large-margin classifiers," *Biometrika*, vol. 95, no. 1, pp. 149–167, Jan. 2008.

[49] Y. Wu, H. H. Zhang, and Y. Liu, "Robust model-free multiclass probability estimation," *J. Amer. Stat. Assoc.*, vol. 105, no. 489, pp. 424–436, 2010.

[50] J. A. K. Suykens and J. Vandewalle, "Multiclass least squares support vector machines," in *Proc. Int. Joint Conf. Neural Networks. (IJCNN)*, 1999, pp. 900–903.

[51] G. Fung and O. L. Mangasarian, "Proximal support vector machine classifiers," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2001, pp. 77–86.

[52] G. M. Fung and O. L. Mangasarian, "Multicategory proximal support vector machine classifiers," *Mach. Learn.*, vol. 59, nos. 1–2, pp. 77–97, 2005.

[53] Y. Tang and H. H. Zhang, "Multiclass proximal support vector machines," *J. Comput. Graph. Statist.*, vol. 15, no. 2, pp. 339–355, Jun. 2006.

[54] Y. Guermeur and E. Monfrini, "A quadratic loss multi-class SVM for which a radius–margin bound applies," *Informatica*, vol. 22, no. 1, pp. 73–96, Jan. 2011.

[55] Y. Guermeur, "A generic model of multi-class support vector machine," *Int. J. Intell. Inf. Database Syst.*, vol. 6, no. 6, pp. 555–577, 2012.

[56] D. K. Agarwal, "Shrinkage estimator generalizations of proximal support vector machines," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2002, pp. 173–182.

[57] H. Zou, J. Zhu, and T. Hastie, "New multicategory boosting algorithms based on multicategory Fisher-consistent losses," *Ann. Appl. Statist.*, vol. 2, no. 4, pp. 1290–1306, 2008.

[58] S. Y. Park, Y. Liu, D. Liu, and P. Scholl, "Multicategory composite least squares classifiers," *Stat. Anal. Data Mining*, vol. 3, no. 4, pp. 272–286, 2010.

[59] T. Zhang, "Statistical analysis of some multi-category large margin classification methods," *J. Mach. Learn. Res.*, vol. 5, pp. 1225–1251, Oct. 2004.

[60] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, "Convexity, classification, and risk bounds," *J. Amer. Statist. Assoc.*, vol. 101, no. 473, pp. 138–156, Mar. 2006.

[61] A. Tewari and P. L. Bartlett, "On the consistency of multiclass classification methods," *J. Mach. Learn. Res.*, vol. 8, pp. 1007–1025, May 2007.

[62] Ü. Doğan, T. Glasmachers, and C. Igel, "A unified view on multi-class support vector classification," *J. Mach. Learn. Res.*, vol. 17, no. 45, pp. 1–32, Jan. 2016.

[63] Z. Noumir, P. Honeine, and C. Richard, "Multi-class least squares classification at binary-classification complexity," in *Proc. IEEE Stat. Signal Process. Workshop (SSP)*, Jun. 2011, pp. 277–280.

[64] R. Huerta, S. Vembu, J. M. Amigó, T. Nowotny, and C. Elkan, "Inhibition in multiclass classification," *Neural Comput.*, vol. 24, no. 9, pp. 2473–2507, Sep. 2012.

[65] I. Rodriguez-Lujan and R. Huerta, "A Fisher consistent multiclass loss function with variable margin on positive examples," *Electron. J. Statist.*, vol. 9, no. 2, pp. 2255–2292, Jan. 2015.

[66] C. Zhang, Y. Liu, and Z. Wu, "On the effect and remedies of shrinkage on classification probability estimation," *Amer. Statistician*, vol. 67, no. 3, pp. 134–142, Aug. 2013.

[67] A. B. Tsybakov, "Optimal aggregation of classifiers in statistical learning," *Ann. Statist.*, vol. 32, no. 1, pp. 135–166, 2004.

[68] B. A. Pires, C. Szepesvari, and M. Ghavamzadeh, "Cost-sensitive multiclass classification risk bounds," in *Proc. 30th Int. Conf. Mach. Learn.*, vol. 28, no. 3, 2013, pp. 1391–1399.

[69] G. Pouliot, "Equivalence of multicategory SVM and simplex cone SVM: Fast computations and statistical theory," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4133–4140.

[70] G. S. Kimeldorf and G. Wahba, "A correspondence between Bayesian estimation on stochastic processes and smoothing by splines," *Ann. Math. Statist.*, vol. 41, no. 2, pp. 495–502, Apr. 1970.

[71] G. Kimeldorf and G. Wahba, "Some results on Tchebycheffian spline functions," *J. Math. Anal. Appl.*, vol. 33, pp. 82–95, Sep. 1971.

[72] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1990, doi: 10.1137/1.9781611970128.

[73] G. Wahba, "Support vector machines, reproducing kernel Hilbert spaces and the randomized GACV," *Adv. Kernel Methods-Support Vector Learn.*, vol. 6, pp. 69–87, Feb. 1999.

[74] A. Argyriou and F. Dinuzzo, "A unifying view of representer theorems," in *Proc. 31st Int. Conf. Mach. Learn.*, vol. 32, no. 2. Bejing, China, 2014, pp. 748–756.

[75] G. Wahba and Y. Wang, "Representer theorem," in *Wiley StatsRef: Statistics Reference Online*, 2019, pp. 1–11.

[76] I. Steinwart, "Consistency of support vector machines and other regularized kernel classifiers," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 128–142, Jan. 2005.

[77] I. Steinwart, D. Hush, and C. Scovel, "An explicit description of the reproducing kernel Hilbert spaces of Gaussian RBF kernels," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4635–4643, Oct. 2006.

[78] M. K. Kozlov, S. P. Tarasov, and L. G. Khachiyan, "Polynomial solvability of convex quadratic programming," in *Doklady Akademii Nauk*, no. 5. Moscow, Russia: Russian Academy of Sciences, 1979, pp. 1049–1051.

[79] S. A. Vavasis, *Complexity Theory: Quadratic Programming*. Boston, MA, USA: Springer, 2001, pp. 304–307.

[80] S. Bubeck, "Convex optimization: Algorithms and complexity," *Found. Trends Mach. Learn.*, vol. 8, nos. 3–4, pp. 231–357, 2014.

[81] A. Frank, D. Fabregat-Traver, and P. Bientinesi, "Large-scale linear regression: Development of high-performance routines," *Appl. Math. Comput.*, vol. 275, pp. 411–421, Feb. 2016.

[82] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[83] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[84] C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. S. Keerthi, and S. Sundararajan, "A dual coordinate descent method for large-scale linear SVM," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 408–415.

[85] S. S. Keerthi, S. Sundararajan, K.-W. Chang, C.-J. Hsieh, and C.-J. Lin, "A sequential dual method for large scale multi-class linear SVMs," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2008, pp. 408–416.

[86] T. Joachims, "Training linear SVMs in linear time," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2006, pp. 217–226.

[87] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for SVM," *Math. Program.*, vol. 127, no. 1, pp. 3–30, Mar. 2011.

[88] Y. Wang and C. Scott, "An exact solver for the Weston–Watkins SVM subproblem," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10894–10904.

[89] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," in *Proc. 21st Int. Conf. Mach. Learn. (ICML)*, 2004, p. 116.

[90] S. Shalev-Shwartz and T. Zhang, "Stochastic dual coordinate ascent methods for regularized loss minimization," *J. Mach. Learn. Res.*, vol. 14, pp. 567–599, Feb. 2013.

[91] Å. Björck, *Numerical Methods for Least Squares Problems*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1996.

[92] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *J. Statist. Softw.*, vol. 33, no. 1, pp. 1–22, 2010.

[93] R Core Team. (2021). *R: A Language and Environment for Statistical Computing*. Vienna, Austria. [Online]. Available: https://www.R-project.org/

[94] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly Weather Rev.*, vol. 78, no. 1, pp. 1–3, 1950.

[95] A. Karatzoglou, D. Meyer, and K. Hornik, "Support vector machines in R," *J. Stat. Softw.*, vol. 15, no. 9, pp. 1–28, 2006.

[96] D. Dua and C. Graff. (2019). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[97] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognit.*, vol. 40, no. 12, pp. 3358–3378, 2007.

[98] X. Qiao and Y. Liu, "Adaptive weighted learning for unbalanced multicategory classification," *Biometrics*, vol. 65, no. 1, pp. 159–168, Mar. 2009.

[99] Z.-H. Zhou and X.-Y. Liu, "On multi-class cost-sensitive learning," *Comput. Intell.*, vol. 26, no. 3, pp. 232–257, 2010.

[100] A. Iranmehr, H. Masnadi-Shirazi, and N. Vasconcelos, "Cost-sensitive support vector machines," *Neurocomputing*, vol. 343, pp. 50–64, May 2019.

[101] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009.

[102] G. Haixiang et al., "Learning from class-imbalanced data: Review of methods and applications," *Exp. Syst. Appl.*, vol. 73, pp. 220–239, May 2017.

[103] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, pp. 1–54, 2019.

[104] M. I. Jordan, J. D. Lee, and Y. Yang, "Communication-efficient distributed statistical inference," *J. Amer. Stat. Assoc.*, vol. 114, no. 526, pp. 668–681, Feb. 2019.

[105] P. Wang, S. Mou, J. Lian, and W. Ren, "Solving a system of linear equations: From centralized to distributed algorithms," *Annu. Rev. Control*, vol. 47, pp. 306–322, 2019.

[106] J. Fan, Y. Guo, and K. Wang, "Communication-efficient accurate statistical estimation," *J. Amer. Stat. Assoc.*, pp. 1–11, Sep. 2021, doi: 10.1080/01621459.2021.1969238.

[107] Y. Gao, W. Liu, H. Wang, X. Wang, Y. Yan, and R. Zhang, "A review of distributed statistical inference," *Stat. Theory Rel. Fields*, vol. 6, no. 2, pp. 89–99, 2022.

[108] P. L. Bartlett and S. Mendelson, "Rademacher and Gaussian complexities: Risk bounds and structural results," *J. Mach. Learn. Res.*, vol. 3, pp. 463–482, Mar. 2003.

[109] A. Maurer, "A vector-contraction inequality for Rademacher complexities," in *Proc. Int. Conf. Algorithmic Learn. Theory*, 2016, pp. 3–17.

[110] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. Cambridge, MA, USA: MIT Press, 2018.

Sheng Fu received the Ph.D. degree in statistics from the University of Chinese Academy of Sciences in 2018. He is currently a Post-Doctoral Fellow with the Division of Cancer Epidemiology and Genetics, National Cancer Institute, USA. His research interests include statistical machine learning and statistical modeling for genetic and genomic studies.

Piao Chen received the bachelor's degree in industrial engineering from Shanghai Jiao Tong University in 2013 and the Ph.D. degree in industrial and systems engineering management from the National University of Singapore in 2017. He is currently an Assistant Professor with the Delft Institute of Applied Mathematics, Delft University of Technology. His research interests include industrial big data analytics, reliability engineering, and statistical learning.

Zhisheng Ye (Senior Member, IEEE) received the joint B.E. degree in material science and engineering and economics from Tsinghua University, Beijing, China, in 2008, and the Ph.D. degree in industrial and systems engineering from the National University of Singapore, Singapore, in 2012.

He is currently an Associate Professor at the Department of Industrial Systems Engineering and Management, National University of Singapore. His research interests include data-driven operations management, mathematical statistics and industrial statistics, and reliability engineering and complex systems modeling. He is an Associate Editor of IEEE TRANSACTIONS ON RELIABILITY and *IISE Transactions*.