Multi-Class Support Vector Machine Based on Minimization of Reciprocal-Geometric-Margin Norms

Yoshifumi Kusunoki (yoshifumi.kusunoki@omu.ac.jp)
Osaka Metropolitan University https://orcid.org/0000-0003-2305-5437
Keiji Tatsumi
Osaka University

Research Article

Keywords: Support vector machine, Multi-class classification, Geometric margin maximization, All-together approach

Posted Date: November 2nd, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3507410/v1

Multi-Class Support Vector Machine Based on Minimization of Reciprocal-Geometric-Margin Norms

Yoshifumi Kusunoki[1*] and Keiji Tatsumi[2†]

[1*]Graduate School of Informatics, Osaka Metropolitan University, Gakuen-cho 1-1, Naka-ku, Sakai, 599-8531, Osaka, Japan.
[2]Graduate School of Engineering, Osaka University, Yamada-oka 2-1, Suita, 565-0871, Osaka, Japan.

*Corresponding author(s). E-mail(s): yoshifumi.kusunoki@omu.ac.jp;
Contributing authors: tatsumi@eei.eng.osaka-u.ac.jp;
†These authors contributed equally to this work.

Abstract

In this paper, we propose a SVM algorithm for multi-class classification. It follows multi-objective multi-class SVM (MMSVM), which maximizes class-pair margins on a multi-class linear classifier. The proposed method, called reciprocal-geometric-margin-norm SVM (RGMNSVM) is derived by applying the ℓ_p-norm scalarization and convex approximation to MMSVM. Additionally, we develop the margin theory for multi-class linear classification, in order to justify maximization of class-pair geometric margins. Experimental results on artificial datasets explain situations where the proposed RGMNSVM successfully works, while conventional multi-class SVMs fail in generalization. Results of classification performance evaluation using benchmark data sets show that RGMNSVM is generally comparable with one of the best multi-class SVMs proposed by Weston and Watkins (1999). However, we observe that the proposed approach to geometric margin maximization actually has better generalization capability for certain real-world data sets.

Keywords: Support vector machine, Multi-class classification, Geometric margin maximization, All-together approach

1 Introduction

Support vector machines (SVMs) (Cortes and Vapnik (1995); Vapnik (1998)) are popular machine learning methods. The original SVM is a learning algorithm for binary linear classifiers. Given binary labeled instances, it learns the hyperplane that separate them with the maximum geometric margin. The theoretical foundations of this criterion were well studied. In these foundations, geometric margin maximization is associated with regularization of statistics. It becomes a motivation to develop regularization-based algorithms.

There are several extensions of SVM for multi-class classification problems, that is, more than two classes are considered (see, e.g. Hsu and Lin (2002); Rifkin and Klautau (2004); Hill and Doucet (2007); Doğan et al (2016)). These extensions are roughly divided into two approaches. One is to learn multiple binary classifiers and combine them to output class label predictions. The other is to optimize parameters of a multi-class linear classifier by a single optimization problem. A popular method of the former approach is one-versus-the-rest (OVR) or one-versus-all (OVA, or one-against-all), in which a c-class problem is reduced to the c binary problems that one class is separated from the others. Another method is one-versus-one (OVO, or one-against-one) or all-versus-all (AVA), in which $c(c-1)/2$ binary classifiers are trained, each of which separates instances in a specific class pair. The combination of binary classifiers of OVRSVM is rather straightforward. On the other hand, that of OVOSVM needs additional heuristics to resolve conflict among outputs of the binary classifiers. Rifkin and Klautau (2004) reported that OVR and OVOSVM are comparable in classification performance.

The study of this paper is closely related to the latter approach, called all-together (AT), all-at-once (AO), or single-machine. Several ATSVMs have been proposed, in which WW (Weston and Watkins (1999)), CS (Crammer and Singer (2002)), and LLW (Lee et al (2004)) are popular. Multinomial logistic regression can be regarded as another variant of ATSVM. Most ATSVMs are regularization-based algorithms using their specific loss functions, which are extensions of margin-based loss functions of binary SVMs. In general, ATSVMs are slower to train than OVR and OVOSVM. On the other hand, Doğan et al (2016) reported that WWSVM is slightly better on average classification performance than OVRSVM, especially for linear classification problems.

Two unified ATSVM frameworks were proposed. Hill and Doucet (2007) proposed geometric framework to understand multi-class classification. The central idea is to introduce a $(c-1)$-dimensional output space which is divided into c class-specific regions by c class-target vectors. Using this framework, we can compare loss functions in the output space and discuss whether they are Fisher consistent. Moreover, Its utility is also demonstrated through a derivation of improved generalization bounds. Doğan et al (2016) proposed a unified view on multi-class SVMs, covering OVRSVM and prominent variants of ATSVM. It categorizes margin-based loss functions mainly in two aspects. The first is the margin concepts, that is whether values of linear functions are

optimized relatively or absolutely. The second is types of aggregation operators for combining margin violations.

Geometric interpretation of the original binary SVM is attractive because its validity can be explained without statistical knowledge. However, the ATSVMs based on regularization mentioned above do not exactly maximize geometric margins, as pointed out by Tatsumi et al (2010, 2011); Tatsumi and Tanino (2014). They formulated a multi-class SVM as a multi-objective optimization problem which simultaneously maximizes all class-pair margins. This model is called multi-objective multi-class SVM (MMSVM). One of the advantages of the multi-objective approach is to provide a diversity of classifiers by Pareto optimal solutions, and systematic selection of them by scalarization techniques.

However, due to the non-convexity of MMSVM, conventional scalarization methods cannot be efficiently applied, with the exception of the ε-constraint method. Tatsumi and Tanino (2014) showed that MMSVM with the ε-constraint method has better generalization ability than ATSVM and OVRSVM. However, that method has difficulty in tuning the best ε parameters. Recently, Kusunoki and Tatsumi (2018) have proposed a convex approximation of MMSVM, called AMMSVM (approximate MMSVM), and a learning algorithm that uses the scalarization that aggregates reciprocal geometric margins by the squared summation. Matsugi et al (2018) have proposed scalarizations of AMMSVM using the reference point method, in which class centroids or OVO solutions are used as a reference point. Furthermore, Kusunoki and Tatsumi (2019) have proposed a scalarization of AMMSVM based on the largest-q norm, which provides a spectrum between the ℓ_∞ and ℓ_1 norms. The numerical experiments of those studies demonstrate that AMMSVM with a scalarization method is at least as good as OVRSVM and WWSVM in generalization capability.

This paper is a revised study of Kusunoki and Tatsumi (2018). First, we review the multi-objective formulation that maximizes all class-pair margins geometrically. We modify this formulation for a soft-margin setting by expanding input vectors. A similar soft-margin formulation is found in Liu et al (2021). Next, we propose a generalization bound for multi-class linear classifiers, which justifies maximization of class-pair geometric margins. The derivation of the bound depends a lot on the margin theory explained in Mohri et al (2018). Finally, we propose a multi-class SVM method by scalarizing MMSVM, in which the ℓ_p-norm is used to aggregate the inverses of the class-pair geometric margins. After the scalarization, we consider a convex approximation of the single-objective optimization problem. We call it reciprocal-geometric-margin-norm SVM (RGMNSVM). It is a generalization of the sum-of-squared-reciprocals scalarization proposed in Kusunoki and Tatsumi (2018). Since RGMNSVM is a kind of extension of WWSVM, we expect its performance to be at least as good as that of WWSVM. The optimization problem of the multi-class SVM falls into a class of conic optimization, which can be solved by existing numerical solvers.

Experiments using two artificial data sets confirm that the proposed RGMNSVM trains large margin classifiers and that results in better generalization performance comparing OVRSVM and a conventional ATSVM (precisely, it is WWSVM with the ℓ_2 hinge loss function). By a performance evaluation using 12 benchmark data sets, in the soft-margin setting, we generally do not observe clear differences between RGMNSVM and ATSVM. However, for some data sets, RGMNSVM is actually better than ATSVM. In the hard-margin setting, we observe that RGMNSVM clearly has better performance than ATSVM for several data sets. The difference of soft- and hard-margin settings is graphically explained using the "iris" data set.

This paper is organized as follows. In Section 2, after introducing multi-class classification problems, we explain class-pair geometric margins in a multi-class linear classifier, which is an extension of the (unique) geometric margin in a binary linear classifier, and then we formulate MMSVM. In Section 3, we develop a margin theory for multi-class linear classification, in order to justify maximization of the class-pair geometric margins. In Section 4, we propose a multi-class SVM, called RGMNSVM, applying norm-based scalarization and convex approximation to MMSVM. We also discuss the approximation ratio of the algorithm. In Section 5, using two artificial data sets, we explain situations where RGMNSVM works successfully. After that, we show results of the performance evaluation of RGMNSVM. Finally, in Section 6, we close this paper with concluding remarks.

2 Multi-objective Multi-class Support Vector Machine

In this paper, we follow studies on multi-objective multi-class SVM (MMSVM) proposed by Tatsumi et al (2010, 2011); Tatsumi and Tanino (2014), which is an application of geometric margin maximization to multi-class linear classification. After introducing the multi-class problems, we recall the margin and SVM for binary problems. Then, we extend these ideas to multi-class problems. Finally, we explain the kernel method. In the rest of the paper, for simplicity we refer to "geometric margin" as "margin."

2.1 Multi-class Linear Classification

The input space $\mathcal{X}$ is a subset of n-dimensional real space $\mathbf{R}^n$, and the set of class labels is defined by $\mathcal{Y} = \{1, 2, \ldots, c\}$. A learning problem is to find a function $\mathcal{C} : \mathcal{X} \to \mathcal{Y}$, called a classifier, using m input vectors with class labels: $(x_1, y_1), \ldots, (x_m, y_m) \in \mathcal{X} \times \mathcal{Y}$. These labeled input vectors are called training instances. The objective of this learning problem is to find classifiers with high classification accuracy not only for training instances, but also for unseen instances.

When $c = 2$, this problem is called binary classification. On the other hand, when $c \geq 3$, it is called multi-class classification. In this paper, we study multi-class classification problems.

Moreover, we study linear classifiers $\mathcal{C}$, which are given in the following form: for every $x \in \mathcal{X}$,

$$\mathcal{C}(x) = \operatorname*{argmax}_{y \in \mathcal{Y}} f(x, y), \tag{1}$$

where

$$f(x, y) = w_y^\top x + b_y. \tag{2}$$

$(w_1, b_1), \ldots, (w_c, b_c) \in \mathbf{R}^{n+1}$ are parameter vectors of this linear classifier $\mathcal{C}$ or the function f. If there is more than one label y whose value $f(x, y)$ is the maximum, we arbitrarily select one among them. The parameters $(w_1, b_1), \ldots, (w_c, b_c)$ are trained using the training instances. In order for the trained classifier to have high accuracy for unseen instances, it is important to address the generalization capability of learning algorithms.

2.2 Geometric Margin and Binary SVM

The original support vector machine (SVM) is a solution for binary classification problems. For introduction to our multi-class extension of SVM, we first review the binary SVM. In binary classification, a linear classifier $\mathcal{C}$ is reduced to the following form:

$$\mathcal{C}(x) = \operatorname{sgn} f(x), \tag{3}$$

where $f(x) = w^\top x + b$ and sgn is the sign function. Here, we assume that the labels are $\mathcal{Y} = \{-1, +1\}$. The vector $(w, b) \in \mathbf{R}^{n+1}$ is the parameters of the classifier. Every instance x with $f(x) = 0$ is arbitrarily classified.

SVM selects the linear classifier whose boundary hyperplane has the largest margin. Assume that all training instances are correctly classified by f, that is, $y_i f(x_i) > 0$. Let Z be the set of training instances $(x_1, y_1), \ldots, (x_m, y_m)$. The margin $\mu_Z(w, b)$ of a hyperplane $\{x \mid f(x) = 0\}$ is the distance between the hyperplane and the nearest instance in Z. It is expressed by

$$\mu_Z(w, b) = \frac{\min_{(x,y) \in Z} y(w^\top x + b)}{\|w\|}, \tag{4}$$

where $\|\cdot\|$ is the Euclidean norm. Instead of maximizing the margin, the learning algorithm of SVM is derived from minimizing the squared reciprocal margin.

$$\begin{aligned} &\underset{w,b}{\text{minimize}} && \left(\mu_Z^{-1}(w, b)\right)^2 \\ &\text{subject to} && y_i(w^\top x_i + b) > 0,\ i \in Z. \end{aligned} \tag{5}$$

Here, we identify Z as its index set $\{1, \ldots, m\}$. Note that $\mu_Z^{-1}(w, b) = \|w\|/(\min_{i \in Z} y_i(w^\top x_i + b))$. We introduce a new variable $r = (\mu_Z^{-1}(w, b))^2$ and $s = \min_{i \in Z} y_i(w^\top x_i + b) > 0$. Then, the problem is expressed by

$$\begin{aligned} \underset{w,b,r,s}{\text{minimize}} \quad & r \\ \text{subject to} \quad & \sqrt{r}s \geq \|w\|, \\ & y_i(w^\top x_i + b) \geq s > 0, \ i \in Z. \end{aligned} \tag{6}$$

Since the objective r is invariant with respect to the positive scalar multiplication for (w, b, s), without loss of generality, we can fix $s = 1$. Consequently, the objective function is replaced by $\|w\|^2$. Finally, we obtain the standard SVM formulation.

$$\begin{aligned} \underset{w,b}{\text{minimize}} \quad & \|w\|^2 \\ \text{subject to} \quad & y_i(w^\top x_i + b) \geq 1, \ i \in Z \end{aligned} \tag{7}$$

For $i \in Z$, let α_i be the optimal dual variable with respect to constraint $y_i(w^\top x_i + b) \geq 1$. A training instance x_i is called a support vector if $\alpha_i > 0$. The optimal hyperplane $\{x \mid w^\top x + b = 0\}$ depends only on the set of support vectors.

To derive the learning model (7), we assume $y_i f(x_i) > 0$ for all $i \in Z$. When this assumption is violated, the model has no feasible solution. Additionally, even if it holds, it is better to relax the constraint of (7), and include more training instances close to the hyperplane as support vectors. When these situations are considered, errors are taken into account for the classification of training instances.

$$\underset{w,b}{\text{minimize}} \quad \lambda^2\|w\|^2 + \sum_{i \in Z} (l_i(w, b))^2, \tag{8}$$

where $l_i(w, b) = \max\{0, 1 - y_i(w^\top x_i + b)\}$ is called hinge loss function. This learning model, which tolerates errors, is called soft-margin. On the other hand, the model (7) is called hard-margin. λ is a hyperparameter to balance the reciprocal margin and the sum of errors. We introduce new variables $\xi_i = l_i(w, b)/y_i$ for every $i \in Z$. It is equivalent to

$$\begin{aligned} \underset{w,b,\xi}{\text{minimize}} \quad & \lambda^2\|w\|^2 + \sum_{i \in Z} \xi_i^2 \\ \text{subject to} \quad & y_i(w^\top x_i + b + \xi_i) \geq 1, \ i \in Z, \end{aligned} \tag{9}$$

where $\xi = (\xi_1, \ldots, \xi_m)^\top \in \mathbf{R}^m$ is the vector of additional decision variables.

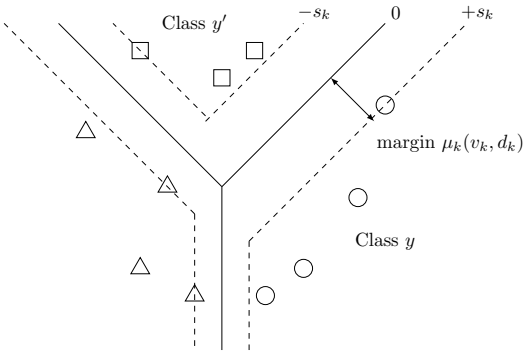


Fig. 1: Linear classifier for 3-class problem. The solid lines are the boundaries with respect to class pairs, i.e., $\{x \in \mathbf{R}^n \mid v_k^\top x + d_k = 0\}$ for each $e_k = (y, y')$, and the two dashed lines parallel with each line indicate the margin region, i.e., $\{x \in \mathbf{R}^n \mid |v_k^\top x + d_k| \leq s_k\}$, where $s_k = \min_{i \in Z_k} y_i^k (v_k^\top x_i^k + d_k)$.

We define $\tilde{w} = (\lambda w_1, \ldots, \lambda w_n, \xi_1, \ldots, \xi_m)^\top \in \mathbf{R}^{n+m}$, where $w_1, \ldots, w_n$ are the n elements of w. Additionally, we define

$$\tilde{x}_i = (x_{i1}/\lambda, \ldots, x_{in}/\lambda, \underbrace{0, \ldots, 0}_{i-1}, 1, 0, \ldots, 0)^\top \in \mathbf{R}^{n+m},$$

where $x_{i1}, \ldots, x_{in}$ are the n elements of x_i. Then, the above optimization problem is equivalent to maximization of the margin function $\mu_{\tilde{Z}}(\tilde{w}, b)$ with respect to data $\tilde{Z} = \{(\tilde{x}_1, y_1), \ldots, (\tilde{x}_m, y_m)\}$. This means the soft-margin SVM (9) can be expressed by the following reciprocal margin minimization:

$$\begin{array}{ll} \underset{\tilde{w}, b}{\text{minimize}} & (\mu_{\tilde{Z}}^{-1}(\tilde{w}, b))^2 \\ \text{subject to} & y_i(\tilde{w}^\top \tilde{x}_i + b) > 0,\ i \in \tilde{Z}. \end{array} \tag{10}$$

Here, $\tilde{Z}$ is identified with its index set $\{1, \ldots, m\}$.

2.3 Geometric Margins and Multi-class SVM

Here, we discuss geometric margins in multi-class linear classification. In Figure 1, we show an example of 3-class linear classification in a 2-dimensional input space. As shown in the figure, a margin can be defined for the boundary of each label pair.

Let $\mathcal{Y}^2 = \{(y, y') \mid y, y' \in \mathcal{Y}, y < y'\}$ be the set of class label pairs. The cardinality of $\mathcal{Y}^2$ is denoted by $\bar{c} = c(c-1)/2$, and we assume that its elements are indexed as follows: $\mathcal{Y}^2 = \{e_1, e_2, \ldots, e_{\bar{c}}\}$.

Let $e_k = (y, y') \in \mathcal{Y}^2$. We express the training set whose labels are y or y' as $x_1^k, \ldots, x_{m_k}^k$, where m_k is the number of them. For each x_i^k, we define a new class label $y_i^k = +1$ if $y_i = y$ and define $y_i^k = -1$ if $y_i = y'$. Let Z_k be the

set of training instances $(x_1^k, y_1^k), \ldots, (x_{m_k}^k, y_{m_k}^k)$. The boundary hyperplane dividing classes y and y' is defined by $\{x \in \mathbf{R}^n \mid f_y(x) - f_{y'}(x) = 0\}$. Note that $f_y(x) - f_{y'}(x) = (w_y - w_{y'})^\top x + (b_y - b_{y'})$. Assume that this hyperplane correctly divides the training instances into Z_k, namely $y^k(f_y(x^k) - f_{y'}(x^k)) > 0$ for all $(x^k, y^k) \in Z_k$. Using these definitions, the margin of the hyperplane for class pair $e_k = (y, y')$ is defined as follows:

$$\mu_{Z_k}(v_k, d_k) = \frac{\min_{(x^k, y^k) \in Z_k} y^k (v_k^\top x^k + d_k)}{\|v_k\|}, \tag{11}$$

where $v_k = w_y - w_{y'}$ and $d_k = b_y - b_{y'}$.

In contrast to the binary case, there are more than two margins in the multi-class linear classifier. Hence, MMSVM defines a learning model by the following multi-objective optimization problem.

$$\begin{array}{ll} \underset{(v,d) \in \mathcal{V}}{\text{minimize}} & \mu_{Z_1}^{-1}(v_1, d_1), \ldots, \mu_{Z_{\bar{c}}}^{-1}(v_{\bar{c}}, d_{\bar{c}^2}) \\ \text{subject to} & y_i^k (v_k^\top x_i^k + d_k) > 0,\ i \in Z_k,\ k \in \mathcal{Y}^{\bar{2}}, \end{array} \tag{HM}$$

where Z_k and $\mathcal{Y}^{\bar{2}}$ are regarded as their index sets $\{1, \ldots, m_k\}$ and $\{1, \ldots, \bar{c}\}$, respectively. Moreover, $(v, d) \in \mathbf{R}^{(n+1)\bar{c}}$ is the vector stacking $v_1, \ldots, v_{\bar{c}}$ and $d_1, \ldots, d_{\bar{c}}$ and $\mathcal{V}$ is the set in which for each element (v, d) there exist $w_1, \ldots, w_c$ and $b_1, \ldots, b_c$ such that

$$v_k = w_y - w_{y'},\ d_k = b_y - b_{y'},\ e_k = (y, y') \in \mathcal{Y}^{\bar{2}}. \tag{12}$$

Next, we introduce the soft-margin model of MMSVM. We derive the soft margin in the same way as (10). For each $k \in \mathcal{Y}^{\bar{2}}$, introducing new variables $\xi_{k1}, \ldots, \xi_{km_k}$, we define $\tilde{v}_k = (\lambda v_{k1}, \ldots, \lambda v_{kn}, \xi_{k1}, \ldots, \xi_{km_k})^\top \in \mathbf{R}^{n+m_k}$. Moreover, we define

$$\tilde{x}_i^k = (x_{i1}^k/\lambda, \ldots, x_{in}^k/\lambda, \underbrace{0, \ldots, 0}_{i-1}, 1, 0, \ldots, 0)^\top \in \mathbf{R}^{n+m_k}.$$

Let $\tilde{Z}_k$ be the set of $(\tilde{x}_1^k, y_1^k), \ldots, (\tilde{x}_{m_k}^k, y_{m_k}^k)$. Then, we define soft-margin MMSVM as minimization of the reciprocal class-pair margins with respect to the class-pair data sets $\tilde{Z}_k$.

$$\begin{array}{ll} \underset{(\tilde{v},d) \in \tilde{\mathcal{V}}}{\text{minimize}} & \mu_{\tilde{Z}_1}^{-1}(\tilde{v}_1, d_1), \ldots, \mu_{\tilde{Z}_{\bar{c}}}^{-1}(\tilde{v}_{\bar{c}}, d_{\bar{c}}), \\ \text{subject to} & y_i^k (\tilde{v}_k^\top \tilde{x}_i^k + d_k) > 0,\ i \in \tilde{Z}_k,\ k \in \mathcal{Y}^{\bar{2}}, \end{array} \tag{SM}$$

where $\tilde{\mathcal{V}}$ is the set defined by the constraint (12), and $\tilde{Z}_k$ is identified with $\{1, \ldots, m_k\}$.

It is worth mentioning the loss function of the soft-margin MMSVM. For each $k \in \mathcal{Y}^2$, we define s_k as follows.

$$s_k = \min_{i \in \tilde{Z}_k} y_i^k (\tilde{v}_k^\top \tilde{x}_i^k + d_k) = \min_{i \in \tilde{Z}_k} y_i^k (v_k^\top x_i^k + d_k + \xi_{ki}) > 0$$

The squared reciprocal class-pair margin is expressed as follows.

$$(\mu_{\tilde{Z}_k}^{-1}(\tilde{v}_k, d_k))^2 = \|\tilde{v}_k / s_k\|^2 = \lambda^2 \|v_k / s_k\|^2 + \sum_{i \in \tilde{Z}_k} (y_i^k \xi_{ki} / s_k)^2$$

Since the margin is minimized and s_k and $\xi_{k1}, \ldots, \xi_{km_k}$ are independent of the other margins, without loss of generality, we have the following equation.

$$y_i^k \xi_{ki} / s_k = \max\{0, 1 - y_i^k (v_k^\top x_i^k + d_k)/s_k\}$$

Hence, the loss function of MMSVM is given by $l_{ki}(v_k/s_k, d_k/s_k) = \max\{0, 1 - y_i^k((v_k/s_k)^\top x_i^k + d_k/s_k)\}$ for each class pair $k \in \mathcal{Y}^2$ and each instance $(x_i^k, y_i^k) \in Z_k$. This is the hinge loss function, as it is used in the soft-margin binary SVM; however, the parameters are divided by s_k. Remember that s_k can be set to 1 in the binary SVM.

Although problem (HM) as well as (SM) is non-convex due to reciprocal margin functions, Tatsumi and Tanino (2014) proposed efficient solution methods using the ϵ-constraint method, which is a popular scalarization method for multi-objective optimization (Ehrgott (2005)). Using this scalarization, it is reduced to a second-order cone programming (SOCP) and is easily dealt with by several numerical solvers. Recently, Kusunoki and Tatsumi (2018, 2019); Matsugi et al (2018) have proposed a convex approximation of (SM), and solved it by conventional scalarization methods. A revised scalarization of the latter approach is shown in Section 4.

2.4 Kernel Method

In the case where the linear classifier (1) is not suitable for the classification problems considered, we apply the kernel trick to MMSVM. The key idea of the kernel trick is to manipulate vectors in a high-dimensional feature space using only a (positive definite) kernel function $\kappa : \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$. For any kernel κ, there is a mapping $\phi : \mathbf{R}^n \to \mathcal{S}_\kappa$ such that $\kappa(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{S}_\kappa}$ for all $x, x' \in \mathbf{R}^n$, where $\langle \cdot, \cdot \rangle_{\mathcal{S}_\kappa}$ is the inner product of the reproducing Hilbert space $\mathcal{S}_\kappa$ associated with κ (Mohri et al (2018)). The mapping can be given by $x \mapsto \phi(x) = \kappa(x, \cdot) \in \mathcal{S}_\kappa$. We reverse the above derivation. That is, we can compute the inner product $\langle \phi(x), \phi(x') \rangle$ in a reproducing kernel Hilbert space $\mathcal{S}$ using only the corresponding kernel function $\kappa(x, x')$, even if $\mathcal{S}$ is infinite dimensional.

Consider the MMSVM problems (HM) and (SM) for the training instances $(\phi(x_1), y_1), \ldots, (\phi(x_m), y_m)$ in the feature space. Let $\bar{\mathcal{S}}$ be the empirical feature space (see, e.g., Abe (2005)), that is, the finite subspace spanned by

$\phi(x_1), \ldots, \phi(x_m)$. Without loss of generality, we can restrict variables $v_1, \ldots, v_{\bar{c}}$ (and also $w_1, \ldots, w_c$) of (HM) to $\bar{\mathcal{S}}$, because the orthogonal component of v_k with respect to the subspace does not affect any of the constraints and only increases the objective functions, namely the representer theorem (see e.g. Mohri et al (2018)) is valid. Furthermore, if the vectors $w_1, \ldots, w_c$ are spanned in $\bar{\mathcal{S}}$, then for every unseen input vector x, $\phi(x)$ can also be restricted to the subspace without changing scores $f(x, y_1), \ldots, f(x, y_c)$. Therefore, we can obtain the same results if all feature vectors are projected to $\bar{\mathcal{S}}$. If the feature mapping is derived from a kernel function, the coordinate system corresponding to $\bar{\mathcal{S}}$ can be obtained by kernel principal component analysis (KPCA).

By KPCA, the feature vector $\phi(x)$ for every $x \in \mathbf{R}^n$ is expressed as a set of finite coordinates, in which each coordinate is obtained by the inner product of the corresponding principal axis and $\phi(x)$. Again, the inner production can be calculated using only the kernel function. Furthermore, we can reduce the computational effort of MMSVM by dropping principal axes corresponding to some of the smallest eigenvalues. In summary, even if MMSVM is performed in a feature space, the corresponding optimization problem can be reduced to the form of (HM).

We remark that we can formulate the kernelized problem without KPCA. For example, consider the problem of (HM). Based on the representer theorem, we have $w_y = \sum_{j \in Z} \alpha_{yj} \phi(x_j)$ for $y \in \mathcal{Y}$. Then, for $e_k = (y, y') \in \mathcal{Y}^2$, we have $v_k = \sum_{j \in Z} (\alpha_{yj} - \alpha_{y'j}) \phi(x_j) = \sum_{j \in Z} \beta_{kj} \phi(x_j)$, where we define $\beta_{kj} = \alpha_{yj} - \alpha_{y'j}$. Then, $\langle \phi(x_i^k), v_k \rangle_{\mathcal{S}_\kappa}$ and $\|v_k\|_{\mathcal{S}_\kappa} = \sqrt{\langle v_k, v_k \rangle_{\mathcal{S}_\kappa}}$ appearing in the kernelized problem of (HM) are expressed as follows.

$$\langle \phi(x_i^k), v_k \rangle_{\mathcal{S}_\kappa} = \sum_{j \in Z} \beta_{kj} \kappa(x_i^k, x_j), \quad \|v_k\|_{\mathcal{S}_\kappa} = \sqrt{\sum_{j \in Z} \sum_{j' \in Z} \beta_{kj} \beta_{kj'} \kappa(x_j, x_{j'})}.$$

Hence, the objective functions and the constraint of the kernelized problem of (HM) can be obtained by a finite number of calculations. Its decision variables are $\alpha_{11}, \alpha_{12}, \ldots, \alpha_{1m}, \ldots, \alpha_{cm}$. This formulation produces the same score functions $f(x, y_1), \ldots, f(x, y_c)$ as those of the above formulation derived from KPCA. It is because we have $\kappa(x, x') = \bar{\phi}(x)^\top \bar{\phi}(x')$ for any input vector x and any training input vector x', where $\bar{\phi}(x)$ is the finite representation of the feature vector of x obtained by KPCA without dropping components.

In the numerical experiments in this paper, we use the radial basis function (RBF) kernel. For input vectors $x, x' \in \mathbf{R}^n$, the value $\kappa_{\mathrm{RBF}}(x, x')$ of the RBF kernel is defined by

$$\kappa_{\mathrm{RBF}}(x, x') = \sigma^2 \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right), \tag{13}$$

where σ is a parameter for scaling input vectors. If σ is large, transformed input vectors are concentrated in similar feature vectors. On the other hand, if

σ is small, the transformed input vectors are nearly orthogonal to each other in the feature space.

3 Margin Theory

In this section, we study the theory of the generalization capability of SVM, which is called margin theory. Our result mostly follows that of Mohri et al (2018), however, extension the multi-class version of the margin theory is different from it.

Recall that $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ is the space of possible instances. An underlying distribution on $\mathcal{Z}$ is denoted by $\mathcal{D}$. For the sake of simplicity, we assume that the input space $\mathcal{X}$ is a finite-dimensional vector space; however, the result of this section can be extended to reproducing kernel Hilbert spaces. Moreover, we omit the bias terms $b_1, \ldots, b_c$ of the linear classifier.

First, we define the generalization error.

Definition 1 (Generalization error) Given a hypothesis $h \in \mathcal{H}$, and an underlying distribution $\mathcal{D}$ on $\mathcal{Z}$, the generalization error or risk of h is defined by

$$R(h) = \mathbb{P}_{(x,y)\sim\mathcal{D}}[h(x) \neq y] = \mathbb{E}_{(x,y)\sim\mathcal{D}}[1_{h(x)\neq y}],$$

where $\mathbb{P}[A]$ denotes the probability of A, $\mathbb{E}[X]$ denotes the expectation of X, and 1_ω is the indicator function of an event ω.

The Rademacher complexity is one of the major tools of the margin theory. It evaluates to what extent a family of classifiers can follow random label assignments.

Definition 2 (Empirical Rademacher complexity) Let $\mathcal{G}$ be a family of functions mapping from $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ to $[a, b]$ and $Z = \{z_1, \ldots, z_m\}$ a fixed sample of size m with elements in $\mathcal{Z}$. Then, the empirical Rademacher complexity of $\mathcal{G}$ with respect to the sample Z is defined as:

$$\widehat{\mathfrak{R}}_Z(\mathcal{G}) = \mathop{\mathbb{E}}_{\sigma}\left[\sup_{g\in\mathcal{G}} \frac{1}{m}\sum_{i=1}^{m} \sigma_i g(z_i)\right],$$

where $\sigma = (\sigma_1, \ldots, \sigma_m)^\top$, with σ_is independent uniform random variables taking values in $\{-1, +1\}$.

Definition 3 (Rademacher complexity) Let $\mathcal{D}$ denote the distribution according to which samples are drawn. For any integer $m \geq 1$, the Rademacher complexity of $\mathcal{G}$ is the expectation of the empirical Rademacher over all samples of size m drawn according to $\mathcal{D}$:

$$\mathfrak{R}_m(\mathcal{G}) = \mathop{\mathbb{E}}_{Z\sim\mathcal{D}^m}[\hat{\mathfrak{R}}_Z(\mathcal{G})].$$

The functions h and g appearing in the above definitions are supposed to be measurable functions. In our situation, they are only a linear function or a finite combination of linear functions.

Using these definitions, we obtain the following theorem about an upper bound of the generalization error.

Theorem 1 (Mohri et al (2018)) *Let $\mathcal{G}$ be a family of functions mapping from $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ to $[0, a]$. Then, for any $\delta > 0$, with probability at least $1 - \delta$ over the draw of an i.i.d sample Z of size m, the following holds for all $g \in \mathcal{G}$:*

$$\mathop{\mathbb{E}}_{z\sim\mathcal{D}}[g(z)] \leq \frac{1}{m}\sum_{i=1}^{m} g(z_i) + 2\mathfrak{R}_m(\mathcal{G}) + a\sqrt{\frac{\log 1/\delta}{2m}}.$$

For any $s > 0$, we define a function $\Phi_s : \mathbf{R} \to \mathbf{R}$ as follows. This transforms a margin to a value bounded on $[0, 1]$, called s-margin loss.

$$\Phi_s(x) = \min\{1, \max\{0, 1 - x/s\}\} = \begin{cases} 1 & \text{if } x \leq 0 \\ 1 - x/s & \text{if } 0 \leq x \leq s \\ 0 & \text{if } s \leq x. \end{cases} \tag{14}$$

The next lemma evaluates the Rademacher complexity of margin losses. We note that Φ_s is a $1/s$-Lipschitz function.

Lemma 2 (Talagrand's lemma) *Let Φ be an l-Lipschitz function from $\mathbf{R}$ to $\mathbf{R}$ and $\sigma_1, \ldots \sigma_m$ be Rademacher random variables. Then, for any set $\mathcal{H}$ of real-valued functions, the following inequality holds:*

$$\mathop{\mathbb{E}}_{\sigma}\left[\sup_{h\in\mathcal{H}}\sum_{i=1}^{m}\sigma_i(\Phi\circ h)(x_i)\right] \leq l\mathop{\mathbb{E}}_{\sigma}\left[\sup_{h\in\mathcal{H}}\sum_{i=1}^{m}\sigma_i h(x_i)\right].$$

Now, we are ready to show our margin theory for multi-class problems. Recall that the linear classifier of the multi-class classification is given by $\mathcal{C}(x) = \mathrm{argmax}_{y\in\mathcal{Y}}\, w_y^\top x$ with the parameters $w_1, \ldots, w_c$. We define a family $\mathcal{H}$ of real-valued functions associated with this classifier.

$$\mathcal{H} = \{(x, e) \mapsto \hat{v}_e^\top x \mid \|\hat{v}_e\| \leq 1, \hat{v}_e = (w_y - w_{y'})/u_e, u_e > 0, e = (y, y') \in \mathcal{Y}^{\bar{2}}\}. \tag{15}$$

For each label $y \in \mathcal{Y}$ and each label-pair $e = (y', y'') \in \mathcal{Y}^{\bar{2}}$ such that $y' = y$ or $y'' = y$, we define

$$y^e = \begin{cases} +1 & y' = y, \\ -1 & y'' = y. \end{cases}$$

For each instance $(x, y) \in \mathcal{X} \times \mathcal{Y}$, the classifier $\mathcal{C}$ fails to classify it if and only if the corresponding function $h \in \mathcal{H}$ satisfies

$$\min_{y\in e\in\mathcal{Y}^{\bar{2}}}\{y^e h(x, e)\} \leq 0,$$

where $y \in e \in \mathcal{Y}^{\bar{2}}$ means $e \in \mathcal{Y}^{\bar{2}}$ such that $e = (y, y')$ or $e = (y', y)$. Therefore, the generalization error $R(h)$ for $h \in \mathcal{H}$ is defined by

$$R(h) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[1_{\min_{y\in e\in\mathcal{Y}^{\bar{2}}}\{y^e h(x,e)\}\le 0}].$$

Using the function of (14), we define a margin loss function $\psi_{h,s}$ with respect to $h \in \mathcal{H}$ and a positive vector $s = (s_e)_{e\in\mathcal{Y}^{\bar{2}}}$.

$$\psi_{h,s}(x,y) = \sum_{y\in e\in\mathcal{Y}^{\bar{2}}} \Phi_{s_e}(y^e h(x,e)).$$

Then, $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\psi_{h,s}(x,y)]$ is an upper bound of $R(h)$.

$$\begin{aligned} R(h) &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[1_{\min_{y\in e\in\mathcal{Y}^{\bar{2}}}\{y^e h(x,e)\}\le 0}] \\ &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{y\in e\in\mathcal{Y}^{\bar{2}}}\{1_{y^e h(x,e)\le 0}\}\right] \\ &\le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\max_{y\in e\in\mathcal{Y}^{\bar{2}}}\Phi_{s_e}(y^e h(x,e))\right] \\ &\le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\sum_{y\in e\in\mathcal{Y}^{\bar{2}}}\Phi_{s_e}(y^e h(x,e))\right] = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\psi_{h,s}(x,y)] \end{aligned}$$

Moreover, given a sample $Z = \{(x_i, y_i)\}$ of size m, we define the empirical margin loss $\widehat{R}_{Z,s}(h)$.

$$\widehat{R}_{Z,s}(h) = \frac{1}{m}\sum_{i=1}^{m}\psi_{h,s}(x_i,y_i).$$

We prove the following theorem.

Theorem 3 *Let $\mathcal{H}$ be the family of functions defined by (15). Assume that $Z \subseteq \{x \mid \|x\| \le r\}$. Fix $s = (s_e)_{e\in\mathcal{Y}^{\bar{2}}} > 0$. Then, for any δ, with probability at least $1-\delta$ over the choice of a sample Z of size m, the following holds for all $h \in \mathcal{H}$:*

$$R(h) \le \widehat{R}_{Z,s}(h) + 2\sqrt{\frac{r^2}{m}}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_e} + c\sqrt{\frac{\log 1/\delta}{2m}}.$$

Proof Let $\Pi_s(\mathcal{H}) = \{(x,y) \mapsto \psi_{h,s}(x,y) \mid h \in \mathcal{H}\}$. By the above discussion and Theorem 1, we have the following inequality. For any $h \in \mathcal{H}$,

$$R(h) \le \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\psi_{h,s}(x,y)] \le \frac{1}{m}\sum_{i=1}^{m}\psi_{h,s}(x_i,y_i) + 2\mathfrak{R}_m(\Pi_s(\mathcal{H})) + c\sqrt{\frac{\log 1/\delta}{2m}}.$$

We focus on $\mathfrak{R}_m(\Pi_s(\mathcal{H}))$. It is the expectation of $\widehat{\mathfrak{R}}_Z(\Pi_s(\mathcal{H}))$, which is expressed as follows.

$$\begin{aligned}\widehat{\mathfrak{R}}_Z(\Pi_s(\mathcal{H})) &= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{\psi_{h,s}\in\Pi_s(\mathcal{H})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\psi_{h,s}(x_i,y_i)\right]\\ &= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{h\in\mathcal{H}}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\sum_{y_i\in e\in\mathcal{Y}^{\bar{2}}}\Phi_{s_e}(y_i^e h(x_i,e)))\right]\\ &= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{h\in\mathcal{H}}\frac{1}{m}\sum_{e\in\mathcal{Y}^{\bar{2}}}\sum_{y_i\in e}\sigma_i\Phi_{s_e}(y_i^e h(x_i,e))\right].\end{aligned}$$

Let $\mathcal{F}=\{x\mapsto v^\top x \mid \|v\|\le 1\}$. Taking the supremum of $h(\cdot,e)$ for each $e\in\mathcal{Y}^{\bar{2}}$, we have an upper bound of $\widehat{\mathfrak{R}}_Z(\Pi_s(\mathcal{H}))$. Then, we apply Talagrand's lemma to it.

$$\begin{aligned}\widehat{\mathfrak{R}}_Z(\Pi_s(\mathcal{H})) &\le \frac{1}{m}\sum_{e\in\mathcal{Y}^{\bar{2}}}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{h(\cdot,e)\in\mathcal{F}}\sum_{y_i\in e}\sigma_i\Phi_{s_e}(y^e h(x_i,e))\right]\\ &\le \frac{1}{m}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_e}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{h(\cdot,e)\in\mathcal{F}}\sum_{y_i\in e}\sigma_i y_i^e h(x_i,e)\right]\\ &= \frac{1}{m}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_e}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{h(\cdot,e)\in\mathcal{F}}\sum_{y_i\in e}\sigma_i h(x_i,e)\right]\\ &= \frac{1}{m}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{|Z_e|}{s_e}\mathfrak{R}_{Z_e}(\mathcal{F}),\end{aligned}$$

where $Z_e=\{z=(x,y)\in Z \mid y\in e\}$ and $\mathfrak{R}_{Z_e}(\mathcal{F})$ is the empirical Rademacher complexity of $\mathcal{F}$. As shown in (Mohri et al (2018)), it is evaluated as $\widehat{\mathfrak{R}}_{Z_e}(\mathcal{F})\le\sqrt{r^2/|Z_e|}$. Hence, noting $|Z_e|\le m$, we have

$$\widehat{\mathfrak{R}}_Z(\Pi_s(\mathcal{H}))\le\frac{1}{m}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_e}\sqrt{r^2|Z_e|}\le\sqrt{\frac{r^2}{m}}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_e},$$

and $\mathfrak{R}_m(\Pi_s(\mathcal{H}))=\mathbb{E}_{Z\sim\mathcal{D}^m}[\widehat{\mathfrak{R}}_Z(\Pi_s(\mathcal{H}))]\le\sqrt{r^2/m}\sum_{e\in\mathcal{Y}^{\bar{2}}}1/s_e$. □

Finally, we extend Theorem 3 to the case where the inequality of the theorem holds for all $s\in(0,a]^{\bar{c}}$, providing some fixed bound $a>0$.

Theorem 4 *Fix $r>0$ and assume $Z\subseteq\{x \mid \|x\|\le r\}$. Let $\mathcal{H}$ be the family of functions defined by (15). Fix $a>0$. Then, for any δ, with probability at least $1-\delta$ over the choice of a sample Z of size m, the following holds for all $s\in(0,a]^{\bar{c}}$ and $h\in\mathcal{H}$:*

$$R(h)\le\widehat{R}_{Z,s}(h)+4\sqrt{\frac{r^2}{m}}\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_e}+c\sqrt{\sum_{e\in\mathcal{Y}^{\bar{2}}}\frac{\log\log_2(2a/s_e)}{m}}+c\sqrt{\frac{\log(2^{\bar{c}}/\delta)}{2m}}.$$

Proof Let $\boldsymbol{k} = (k_1, ..., k_{\bar{c}})$, where $k_1, \ldots, k_{\bar{c}}$ are nonnegative integers. Consider two sets $\{s_{\boldsymbol{k}}\}_{\boldsymbol{k}\geq \mathbf{1}}$ and $\{\epsilon_{\boldsymbol{k}}\}_{\boldsymbol{k}\geq \mathbf{1}}$ with $s_{\boldsymbol{k}} \in (0, a]^{\bar{c}}$ and $\epsilon_{\boldsymbol{k}} \in (0, 1]$, where $\boldsymbol{k} \geq \mathbf{1}$ means $k_1 \geq 1, \ldots, k_{\bar{c}} \geq 1$. By Theorem 3, for any fixed $\boldsymbol{k} \geq \mathbf{1}$,

$$\mathbb{P}\left[\sup_{h\in\mathcal{H}}\left(R(h) - \widehat{R}_{Z,s_{\boldsymbol{k}}}(h)\right) > 2\sqrt{\frac{r^2}{m}}\sum_{l=1}^{\bar{c}}\frac{1}{(s_{\boldsymbol{k}})_l} + \epsilon_{\boldsymbol{k}}\right] \leq \exp(-2m\epsilon_{\boldsymbol{k}}^2/c^2).$$

Choose $\epsilon_{\boldsymbol{k}} = \epsilon + c\sqrt{\sum_{l=1}^{\bar{c}}\frac{\log k_l}{m}}$, where $\epsilon > 0$. Then, by the union bound, the following holds:

$$\begin{aligned}
&\mathbb{P}\left[\sup_{h\in\mathcal{H},\boldsymbol{k}\geq\mathbf{1}}\left(R(h) - \widehat{R}_{Z,s_{\boldsymbol{k}}}(h) - 2\sqrt{\frac{r^2}{m}}\sum_{l=1}^{\bar{c}}\frac{1}{(s_{\boldsymbol{k}})_l} - \epsilon_{\boldsymbol{k}}\right) > 0\right] \\
&\leq \sum_{k_1\geq 1}\cdots\sum_{k_{\bar{c}}\geq 1}\exp\left(-2m\epsilon_{\boldsymbol{k}}^2/c^2\right) \\
&\leq \exp(-2m\epsilon^2/c^2)\sum_{k_1\geq 1}\cdots\sum_{k_{\bar{c}}\geq 1}\prod_{l=1}^{\bar{c}}\exp(-2\log k_l) \\
&= \exp(-2m\epsilon^2/c^2)\prod_{l=1}^{\bar{c}}\sum_{k_l\geq 1}\exp(-2\log k_l) \\
&= \exp(-2m\epsilon^2/c^2)\prod_{l=1}^{\bar{c}}\sum_{k_l\geq 1}1/k_l^2 \leq \exp(-2m\epsilon^2/c^2)\left(\frac{\pi^2}{6}\right)^{\bar{c}} \leq 2^{\bar{c}}\exp(-2m\epsilon^2/c^2)
\end{aligned}$$

We define each $s_{\boldsymbol{k}}$ as follows: $(s_{\boldsymbol{k}})_l = a/2^{k_l}$ for $l = 1, \ldots, \bar{c}$. For any $s \in (0, a]^{\bar{c}}$, there exists $\boldsymbol{k} \geq \mathbf{1}$ such that $s_{\boldsymbol{k}} \leq s \leq 2s_{\boldsymbol{k}}$. Since $s_{\boldsymbol{k}} \leq s$, we have for any $h \in \mathcal{H}$, $\widehat{R}_{Z,s_{\boldsymbol{k}}}(h) \leq \widehat{R}_{Z,s}(h)$, and since $s \leq 2s_{\boldsymbol{k}}$, we have $\sum_{l=1}^{\bar{c}} 1/(s_{\boldsymbol{k}})_l \leq 2\sum_{l=1}^{\bar{c}} 1/s_l$. Finally, since $s \leq 2s_{\boldsymbol{k}}$, we have $\sum_{l=1}^{\bar{c}}\log k_l = \sum_{l=1}^{\bar{c}}\log\log_2(a/(s_{\boldsymbol{k}})_l) \leq \sum_{l=1}^{\bar{c}}\log\log_2(2a/s_l)$. Thus, we have

$$\begin{aligned}
&\sup_{h\in\mathcal{H},\boldsymbol{k}\geq\mathbf{1}}\left(R(h) - \widehat{R}_{Z,s_{\boldsymbol{k}}}(h) - 2\sqrt{\frac{r^2}{m}}\sum_{l=1}^{\bar{c}}\frac{1}{(s_{\boldsymbol{k}})_l} - c\sqrt{\sum_{l=1}^{\bar{c}}\frac{\log k_l}{m}} - \epsilon\right) \\
&\geq \sup_{h\in\mathcal{H},s\in(0,a]^{\bar{c}}}\left(R(h) - \widehat{R}_{Z,s}(h) - 4\sqrt{\frac{r^2}{m}}\sum_{l=1}^{\bar{c}}\frac{1}{s_l} - c\sqrt{\sum_{l=1}^{\bar{c}}\frac{\log\log_2(2a/s_l)}{m}} - \epsilon\right).
\end{aligned}$$

Therefore, the following inequality holds:

$$\begin{aligned}
\mathbb{P}\Bigg[\sup_{h\in\mathcal{H},s\in(0,a]^{\bar{c}}}\Bigg(&R(h) - \widehat{R}_{Z,s}(h) - 4\sqrt{\frac{r^2}{m}}\sum_{l=1}^{\bar{c}}\frac{1}{s_l} \\
&- c\sqrt{\sum_{l=1}^{\bar{c}}\frac{\log\log_2(2a/s_l)}{m}} - \epsilon\Bigg) > 0\Bigg] \leq 2^{\bar{c}}\exp(-2m\epsilon^2/c^2).
\end{aligned}$$

□

4 Scalarization by Reciprocal-Geometric-Margin Norms and Approximate Solutions

Theorem 4 shows that the generalization error of a classifier $h \in \mathcal{H}$ is bounded by the sum of the empirical margin loss $\widehat{R}_{Z,s}(h)$ and $\sum_{k\in\mathsf{y}^{\bar{z}}} 1/s_k$. Moreover,

we have the following inequality about $\widehat{R}_{Z,s}(h)$.

$$\widehat{R}_{Z,s}(h) = \frac{1}{m}\sum_{i=1}^{m}\psi_{h,s}(x_i,y_i) = \frac{1}{m}\sum_{k\in\mathcal{Y}^2}\sum_{i\in Z_k}\Phi_{s_k}(y_i^k h(x_i^k,k)) \leq \frac{1}{m}\sum_{k\in\mathcal{Y}^2}\sum_{i\in Z_k}\max\{0,1-y_i^k(v_k^\top x_i^k)/(u_k s_k)\}.$$

We replace $u_k s_k$ with s_k, and without loss of generality set $u_k = \|v_k\|$. Then, we can see that the generalization error is bounded above by the sum of class-pair hinge losses $\sum_{i\in Z_k}\max\{0,1-y_i^k(v_k^\top x_i^k)/s_k\}$ and the sum of class-pair reciprocal geometric margins $\|v_k\|/s_k$. It justifies the formulation of soft-margin MMSVM that minimizes class-pair reciprocal geometric margins: $\mu_{\tilde{Z}_k}^{-1}(\tilde{v}_k,d_k)$, $k=1,\ldots,\bar{c}$.

Now, we discuss methods to solve MMSVM. A popular approach to solve multi-objective optimization problems is scalarization, in which the objective functions are reduced to a single objective function. In previous work, Tatsumi and Tanino (2014) applied the ϵ-constraint scalarization method to MMSVM. That method has the advantage that the derived single-objective optimization problem becomes convex. However, there is difficulty in selecting the parameters.

In this paper, we propose the minimization of a norm of the reciprocal geometric margin vector. The reason why we adopt this scalarization is that it becomes a similar formulation to WWSVM. Hence, we expect that its generalization performance is at least as good as WWSVM, which is one of the best multi-class SVMs (Doğan et al (2016)). Additionally, because of the similarity, we also expect that its computational effort is not much greater than that of WWSVM. In this paper, we only show the unweighted version of the norm-based scalarization and compute one (approximate) Pareto solution. On the other hand, by introducing component-wise weights for the reciprocal geometric margin vector, we can obtain different Pareto solutions.

However, as with MMSVM, this scalarized optimization problem is not convex. Hence, we approximate the scalarization to a convex optimization problem. We show that the optimum of the modified problem is an approximate solution of the original scalarized problem. In the following subsection, we only consider the hard-margin model; however, it can be easily extended to soft-margin by the technique explained in Section 2.3.

The vector of reciprocal margins is denoted by

$$\mu^{-1}(v,d) = (\mu_1^{-1}(v_1,d_1),\ldots,\mu_{\bar{c}}^{-1}(v_{\bar{c}},d_{\bar{c}}))^\top,$$

where $\mu_k^{-1}(v_k,d_k)$ means $\mu_{Z_k}^{-1}(v_k,d_k)$. Let $p>1$, and $\|\cdot\|_p$ denotes the ℓ_p-norm, i.e., $\|r\|_p = (\sum_{k=1}^{l}|r_k|^p)^{1/p}$ for $r\in\mathbf{R}^l$. The optimization problem that

minimizes the ℓ_p-norm of reciprocal margins is defined as follows.

$$\begin{aligned} \underset{(v,d)\in\mathcal{V}}{\text{minimize}} \quad & \|\mu^{-1}(v,d)\|_p \\ \text{subject to} \quad & y_i^k(v_k^\top x_i^k + d_k) > 0,\ i \in Z_k,\ k \in \mathcal{Y}^{\bar{2}}, \end{aligned} \tag{NpS}$$

This kind of scalarization is called the achievement function method (Ehrgott (2005)). Since $\mu_k^{-1}(v,d)$ is nonnegative, this scalarization works well. The norm has the following property: for two nonnegative reals $r, r' \in \mathbf{R}_+^l$, $r \leq r'$ and $r \neq r'$ imply $\|r\|_p < \|r'\|_p$. Therefore, by Theorem 4.29 in (Ehrgott (2005)), every optimum (v^*, d^*) of (NpS) is an efficient solution of the original multi-objective problem (HM). This means that there is no other feasible solution (v,d) such that $\mu^{-1}(v,d) \leq \mu^{-1}(v^*,d^*)$ and $\mu^{-1}(v,d) \neq \mu^{-1}(v^*,d^*)$.

Let $r_k = (\mu^{-1}(v_k,d_k))^p$ and raise the objective function to the p-th power. Furthermore, let $s_k = \min_{i\in Z_k} y_i^k(v_k^\top x_i^k + d_k)$ and without loss of generality we add $s_k \geq 1$. Then (NpS) is expressed as follows.

$$\begin{aligned} \underset{(v,d)\in\mathcal{V},r,s}{\text{minimize}} \quad & \sum_{k\in\mathcal{Y}^{\bar{2}}} r_k \\ \text{subject to} \quad & r_k^{1/p} s_k \geq \|v_k\|,\ k \in \mathcal{Y}^{\bar{2}}, \\ & y_i^k(v_k^\top x_i^k + d_k) \geq s_k \geq 1,\ i \in Z_k,\ k \in \mathcal{Y}^{\bar{2}}, \end{aligned} \tag{16}$$

The first constraint is not convex, however if s_k is replaced with $s_k^{1-1/p}$ it becomes convex. Therefore, we propose the following modified problem.

$$\begin{aligned} \underset{(v,d)\in\mathcal{V},r,s}{\text{minimize}} \quad & \sum_{k\in\mathcal{Y}^{\bar{2}}} r_k \\ \text{subject to} \quad & r_k^{1/p} s_k^{1-1/p} \geq \|v_k\|,\ k \in \mathcal{Y}^{\bar{2}} \\ & y_i^k(v_k^\top x_i^k + d_k) \geq s_k \geq 1,\ i \in Z_k,\ k \in \mathcal{Y}^{\bar{2}}, \end{aligned} \tag{ANpS}$$

The first constraint forms convex cones called radial power cone (Kapelevich et al (2022)). This form of optimization problem can be solved by existing numerical solvers (e.g. MOSEK ApS (2022)). We call the learning model (ANpS) reciprocal-geometric-margin-norm SVM (RGMNSVM).

We evaluate the quality of the optimum of (ANpS) relative to the original problem (NpS). Let $(v^\star, d^\star)$ be one of the optima of the original problem and define $s_k^\star = \min_{i\in Z_k} y_i^k(v_k^{\star\top} x_i^k + d_k^\star)$ for each $k \in \mathcal{Y}^{\bar{2}}$. Let (v^*, d^*, r^*, s^*) be one of the optima of (ANpS). The lower bound of s_k of (ANpS) can be an arbitrary positive value without changing $\mu^{-1}(v^*, s^*)$. Hence, without loss of generality, we assume $s_k^* \geq \min_{k'\in\mathcal{Y}^{\bar{2}}} s_{k'}^\star$ for all $k \in \mathcal{Y}^{\bar{2}}$. We have $s_k^* = \min_{i\in Z_k} y_i^k(v_k^{*\top} x_i^k + d_k^*)$ for each $k \in \mathcal{Y}^{\bar{2}}$. If not, r_k^* can be decreased. Since

$(v^\star, s^\star)$ is the optimum of (NpS), we have the following inequality.

$$\sum_{k\in\mathcal{Y}^{\bar{2}}}\left(\frac{\|v_k^*\|}{s_k^*}\right)^p \geq \sum_{k\in\mathcal{Y}^{\bar{2}}}\left(\frac{\|v_k^\star\|}{s_k^\star}\right)^p.$$

On the other hand, since (v^*, s^*) is the optimum of (ANpS), we have

$$\sum_{k\in\mathcal{Y}^{\bar{2}}}\left(\frac{\|v_k^\star\|}{s_k^\star}\right)^p \geq \sum_{k\in\mathcal{Y}^{\bar{2}}}\frac{1}{s_k^\star}\frac{\|v_k^\star\|^p}{(s_k^\star)^{p-1}} \geq \frac{1}{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}\sum_{k\in\mathcal{Y}^{\bar{2}}}\frac{\|v_k^*\|^p}{(s_k^*)^{p-1}}.$$

Furthermore,

$$\begin{aligned}\frac{1}{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}\sum_{k\in\mathcal{Y}^{\bar{2}}}\frac{\|v_k^*\|^p}{(s_k^*)^{p-1}} &= \frac{1}{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}\sum_{k\in\mathcal{Y}^{\bar{2}}} s_k^*\left(\frac{\|v_k^*\|}{s_k^*}\right)^p \\ &\geq \frac{\min_{k\in\mathcal{Y}^{\bar{2}}} s_k^*}{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}\sum_{k\in\mathcal{Y}^{\bar{2}}}\left(\frac{\|v_k^*\|}{s_k^*}\right)^p.\end{aligned}$$

Finally, by the assumption and the fact that $s_k^* \geq \min_{k'\in\mathcal{Y}^{\bar{2}}} s_{k'}^\star$, we have

$$\frac{\min_{k\in\mathcal{Y}^{\bar{2}}} s_k^*}{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}\sum_{k\in\mathcal{Y}^{\bar{c}}}\left(\frac{\|v_k^*\|}{s_k^*}\right)^p \geq \frac{\min_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}\sum_{k\in\mathcal{Y}^{\bar{2}}}\left(\frac{\|v_k^*\|}{s_k^*}\right)^p$$

Consequently, we have the following theorem.

Theorem 5 *Let* OPT *be the optimal value of* (NpS)*. Let* Ω *and* $\widetilde{\Omega}$ *be the sets of optimum solutions of* (NpS) *and* (ANpS)*, respectively. For each* $(v^\star, d^\star) \in \Omega$*, we define* $s_k^\star = \min_{i\in Z_k} y_i^k(v_k^{\star\top} x_i^k + d_k^\star)$*,* $k \in \mathcal{Y}^{\bar{2}}$*. We define* θ *as follows:*

$$\theta = \inf_{(v^\star,d^\star)\in\Omega}\frac{\max_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}{\min_{k\in\mathcal{Y}^{\bar{2}}} s_k^\star}.$$

Then, for every $(v^*, d^*, s^*, r^*) \in \widetilde{\Omega}$*, we have*

$$\|\mu^{-1}(v^*, d^*)\|_p \leq \theta^{1/p}\mathrm{OPT}.$$

This theorem says that we can obtain an approximate solution for (NpS) by solving (ANpS) with the approximation ratio $\theta^{1/p}$. The value θ evaluates the heterogeneity of the functional margins $s_1^\star, \ldots, s_{\bar{c}}^\star$. The ratio becomes smaller as the functional margins are more homogeneous and p is larger. Especially, as p approaches infinity, it approaches 1.

Finally, we remark that if $p = 2$ and $s_k = 1$ for all $k \in \mathcal{Y}^{\bar{2}}$ then (ANpS) is reduced to WWSVM.

Table 1: Margins of classifiers for data set D1

Class pair	$(1,2)$	$(1,3)$	$(2,3)$
OVRSVM	1.33	0.70	1.44
ATSVM	1.48	0.74	1.56
RGMNSVM ($p = 1.2$)	3.83	0.74	2.94
RGMNSVM ($p = 2$)	6.34	0.74	5.87
RGMNSVM ($p = 4$)	7.66	0.74	7.24

5 Numerical Experiments

5.1 Examination Using Artificial Data Sets

In this subsection, we examine characteristics of the proposed SVM comparing conventional multi-class SVMs using two artificial data sets. Conventional SVMs considered here are OVRSVM (one-versus-the-rest SVM) and ATSVM (all-together SVM). Note that ATSVM appearing in this section means WWSVM (Weston and Watkins (1999)) with the ℓ_2 hinge loss function. We examine RGMNSVM with three values $p \in \{1.2, 2, 4\}$ and fixed $\lambda = 0.01$.

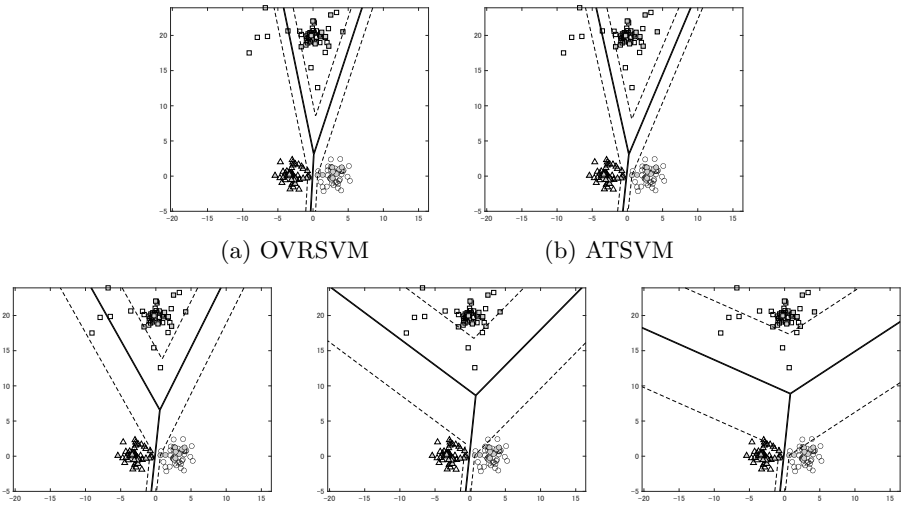
Both of the data sets have 3 classes generated by probability distributions. Figs. 2 and 3 show these data sets. Triangles, squares, and circles, shown in the figures, indicate instances of classes 1, 2, and 3, respectively.

The first data set in Fig. 2, called D1, consists of two normal distributions and one generalized t-distribution. The instances of classes 1 and 3 are generated by transforming sample points x of the 2-dimensional standard normal distribution as $x + \mu$, where $\mu = (-3, 0)$ for class 1 and $\mu = (+3, 0)$ for class 3. Those of class 2 are generated by transforming sample points x of the t-distribution as $x + (0, 20)$, where the number of degrees of freedom is 2. For each class, 50 instances are generated, 10 of which are used for training. The gray-filled markers in the figure are the training instances.

We consider what classifier (w_1, w_2 and w_3) ATSVM obtains for this data set. Classes 1 and 3 are closer to each other than class 2. Hence, the norm of $v_{(1,3)} = w_1 - w_3$ should be larger than the others. Considering the symmetry of classes 1 and 3 around the vertical line, w_1 and w_3 are in opposite directions from each other. Moreover, the norms of $v_{(1,2)} = w_1 - w_2$ and $v_{(2,3)} = w_2 - w_3$ are minimized, so w_2 should be small. As a result, the directions of $v_{(1,2)}$ and $v_{(2,3)}$ become roughly horizontal and some outliers of class 2 are misclassified. We have a similar result for OVRSVM. On the other hand, RGMNSVM aims to increase the margins of class-pairs $(1, 2)$ and $(2, 3)$. Hence, the corresponding boundary lines lie in roughly 30 to 60 degrees counter-clockwise and clockwise, respectively, from the vertical axis.

Table 1 shows the class-pair margins of the obtained classifiers. The margins $(1, 2)$ and $(2, 3)$ of RGMNSVM are larger than those of OVRSVM and ATSVM, and we obtain the largest margins by RGMNSVM with $p = 4$.

The second data set in Fig. 3, called D2, consists of one normal distribution and two mixed distributions. The horizontal coordinates of instances of classes 1 and 3 are generated by transforming sample points x_1 of the standard normal



(a) OVRSVM (b) ATSVM

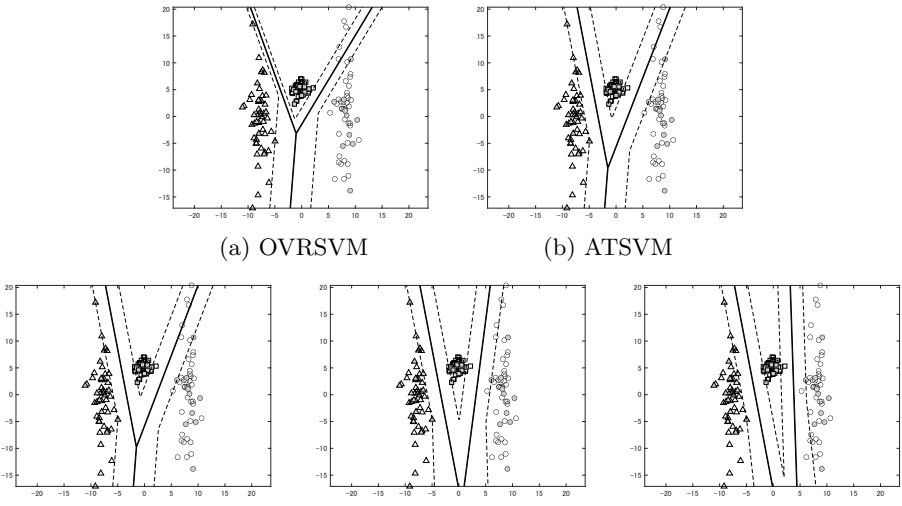
(c) RGMNSVM ($p = 1.2$) (d) RGMNSVM ($p = 2$) (e) RGMNSVM ($p = 4$)

Fig. 2: Boundaries of classifiers for data set D1

distribution as $x_1 - 8$ and $x_1 + 8$, respectively. The vertical coordinates of instances of classes 1 and 3 are generated by transforming sample points x_2 of the t-distribution as $5x_2$, where the number of degrees of freedom is 2. Instances of class 2 are generated by transforming sample points x of the 2-dimensional standard normal distribution as $x + (0, 5)$. For each class, 50 instances are generated, 10 of which are used for training. The gray-filled markers in the figure are the training instances.

In this data set, we expect that the boundary lines between $(1, 2)$ and $(2, 3)$ are vertically oriented. By the symmetry of classes 1 and 3 around the vertical axis, w_1 and w_3 are ideally in opposite directions to each other. On the other hand, since class 2 lies between classes 1 and 3, the norm of $v_{(1,3)} = w_1 - w_3$ is not required to be large. To minimize the norm of $v_{(1,3)}$, w_1 and w_3 tend to be in the same direction. As a result, the boundary lines between $(1, 2)$ and $(2, 3)$ rotate counterclockwise and clockwise, respectively, and some of the outliers in the upper region are misclassified. We have a similar result for OVRSVM. On the contrary, RGMNSVM tends to maximize the margin between classes 1 and 3. Therefore, those boundary lines are roughly vertical and outliers are correctly classified, especially those of $p = 2$ and $p = 4$.

Table 2 shows the class-pair margins of the obtained classifiers. Values of the margin between classes 1 and 3 obtained by RGMNSVM are larger than those of OVRSVM and ATSVM. On the other hand, values of the margins between classes 2 and 3 obtained by RGMNSVM with $p = 2$ and $p = 4$ are smaller than that of ATSVM.



(a) OVRSVM (b) ATSVM

(c) RGMNSVM ($p = 1.2$) (d) RGMNSVM ($p = 2$) (e) RGMNSVM ($p = 4$)

Fig. 3: Boundaries of classifiers for data set D2

Table 2: Margins of classifiers for data set D2

Class pair	$(1, 2)$	$(1, 3)$	$(2, 3)$
OVRSVM	0.68	3.81	1.60
ATSVM	2.43	3.81	2.63
RGMNSVM ($p = 1.2$)	2.43	3.84	2.63
RGMNSVM ($p = 2$)	2.43	5.00	2.49
RGMNSVM ($p = 4$)	2.42	5.73	2.31

5.2 Performance Evaluation

To examine performance of RGMNSVM, we conducted numerical experiments using 12 benchmark data sets obtained from UCI Machine Learning Repository (Dua and Graff (2017)). We compare the generalization capacity of RGMNSVM (ANpS) with ATSVM. The reason we selected ATSVM (WWSVM) for the comparison is that it gives robust performance in all disciplines (benchmark problems), as shown in Doğan et al (2016). RGMNSVM is examined when the hyperparameter is $p = 2$ or $p = 4$. In order to obtain non-linear classification, we performed SVMs in a feature space corresponding to the RBF kernel (13).

5.2.1 Benchmark Data

The benchmark data sets are summarized in Table 3. Each row is the summary of one data set. The first column indicates the names of the data sets, and the second indicates their descriptions in the repository. The third column shows preprocessing methods applied to data sets. “one-hot” means that discrete

Table 3: Summary of data sets

Data	Description	Preprocessing	m	n	c
acc	Speaker Accent Recognition		329	12	6
bal	Balance Scale		625	4	3
car	Car Evaluation	one-hot	1728	13 (7)	4
der	Dermatology		366	33	6
for	Forest type mapping		523	27	4
iri	Iris		150	4	3
opt	Opt. Rec. of Handwritten Digits		5620	64	10
pag	Page Blocks Classification	scaling	5473	10	5
seg	Image Segment	scaling	2310	19	7
veh	Vehicle Silhouettes	scaling	846	18	4
win	Wine	scaling	178	13	3
vow	Connectionist Bench		990	10	11

input values are transformed to one-hot vectors. "scaling" means that each input coordinate is standardized, that is, make the mean 0 and the standard deviation 1. The forth to sixth columns are the numbers of instances, inputs, and labels, respectively.

5.2.2 Classification Error

To estimate the generalization capability of SVMs, we measure the classification errors of the classifiers obtained over a number of trials. For each data set, 80% of the instances are randomly selected for training, where the class proportion is preserved. Using the classifier obtained by RGMNSVM or ATSVM, we classify the test (remaining) instances. The classification error is the percentage of misclassified test instances. There are two hyperparameters σ and λ for the RBF kernel and SVMs, respectively. They are selected by two times of 5-fold cross-validations, namely, for each set of hyperparameters, we calculate the classification errors for the training instances by two times of cross-validations, and select the set minimizing the sum of two errors. The number of trials is 80.

5.2.3 Features

The feature space in which SVMs is applied is a l-dimensional subspace of the feature space spanned by the training instances. We explain how to obtain coordinate vectors of instances (including test ones) in the feature space. For each data set, we first compute the kernel matrix of the training instances (instances of 4 folds in the case of 5-fold cross-validation) and make it centralized. Then, we obtain the eigenvalues and the eigenvectors of the kernel matrix. By the technique of KPCA, all instances are projected onto the space spanned by the l eigenvectors corresponding to the l largest eigenvalues. In the experiments, we set $l = 200$ to reduce the execution time of the experiments. If the number of training instances m' is less than 200, we set $l = m'$.

5.2.4 Optimization

To solve the optimization problem, we used software package MOSEK (MOSEK ApS (2022)). In order to clarify what is soft-margin RGMNSVM, we revisit the optimization problem of RGMNSVM as follows, where it is actually solved in the experiments.

$$\begin{aligned} \underset{w,b,\xi,r,s}{\text{minimize}} \quad & \sum_{k\in\mathcal{Y}^2} r_k \\ \text{subject to} \quad & r_k^{1/p} s_k^{1-1/p} \geq u_k,\ s_k \geq 1,\ k \in \mathcal{Y}^{\bar{2}} \\ & u_k \geq \|(\sqrt{\lambda}(w_{e_{k1}} - w_{e_{k2}}), \xi_k/\sqrt{\lambda})\|,\ k \in \mathcal{Y}^{\bar{2}} \\ & y_i^k((w_{e_{k1}} - w_{e_{k2}})^\top x_i^k + (b_{e_{k1}} - b_{e_{k2}}) + \xi_{ki}) \geq s_k,\ i \in Z_k,\ k \in \mathcal{Y}^{\bar{2}}. \end{aligned} \tag{17}$$

w and b are the vectors stacking $w_1, \ldots, w_c$ and $b_1, \ldots, b_c$, respectively. ξ_k is the vector consisting of $(\xi_{ki})_{i\in Z_k}$, and ξ is the vector stacking $\xi_1, \ldots, \xi_{\bar{c}}$. Moreover, $(\sqrt{\lambda}(w_{e_{k1}} - w_{e_{k2}}), \xi_k/\sqrt{\lambda})$ is the vector stacking $\sqrt{\lambda}(w_{e_{k1}} - w_{e_{k2}})$ and $\xi_k/\sqrt{\lambda}$. For a pair of class labels e_k, e_{k1} and e_{k2} are the class labels included in e_k, that is, $e_k = (e_{k1}, e_{k2})$. We remark that every x_i^k is a feature vector obtained by KPCA. The second constraint is the same as $u_k \geq \|(\lambda(w_{e_{k1}} - w_{e_{k2}}), \xi_k)\|$, however, we use the above form in order to increase computational stability.

When $p = 2$ and s_k for all $k \in \mathcal{Y}^{\bar{2}}$ is fixed to 1, this optimization problem is reduced to that of ATSVM. Classifiers of ATSVM are obtained by solving this reduced problem by MOSEK.

5.2.5 Results and Discussion

Table 4 shows the classification errors of soft-margin ATSVM (AT) and RGMNSVM (RGMN) with $p = 2$ and $p = 4$. The second and third columns show the integer intervals of the hyperparameters σ and λ, respectively. Each interval for $\log_2(\sigma)$ and $\log_{10}(\lambda)$ includes 5 and 2 values, respectively. Therefore, we select the best of 10 pairs of hyperparameters by cross-validation. For each data set, the intervals are adjusted by a preliminary experiment to achieve the smallest classification error by ATSVM. The numbers before $\pm$ in the fourth to sixth columns are the averages of the classification errors for 80 trials, and those after $\pm$ are the corresponding standard deviations. The numbers in bold indicate the best (smallest) results among three SVMs. The marks * and $\star$ indicate the rejection of the hypothesis that the error distributions obtained by ATSVM and RGMNSVM have the same mean by the two-tailed paired t test, where the significance level is 1%. When RGMNSVM is better than ATSVM, we put *, and otherwise, we put $\star$.

From the result of Table 4, we observe that the classification performances of RGMNSVM and ATSVM are generally comparable. However, for the *bal* data set, RGMNSVM clearly outperforms ATSVM. Furthermore, RGMNSVM with $p = 4$ has statistically better results for *opt* and *pag*, and RGMNSVM

Table 4: Classification errors (%) of soft-margin SVMs.

Data	$\log_2(\sigma)$	$\log_{10}(\lambda)$	AT	RGMN ($p=2$)	RGMN ($p=4$)
acc	$[2,6]$	$[-1,0]$	16.36 ± 4.10	**16.06** ± 4.30	16.53 ± 4.37
bal	$[2,6]$	$[-2,-1]$	0.95 ± 1.09	* 0.44 ± 0.87	* **0.43** ± 0.83
car	$[0,4]$	$[-2,-1]$	0.49 ± 0.49	* **0.39** ± 0.44	0.43 ± 0.47
der	$[1,5]$	$[0,1]$	**2.60** ± 1.55	**2.60** ± 1.54	⋆ 2.84 ± 1.66
for	$[5,9]$	$[0,1]$	9.38 ± 2.68	9.39 ± 2.74	**9.36** ± 2.69
iri	$[2,6]$	$[-1,0]$	3.04 ± 2.70	2.88 ± 2.73	**2.75** ± 2.78
opt	$[4,8]$	$[0,1]$	1.21 ± 0.31	1.21 ± 0.31	* **1.14** ± 0.30
pag	$[3,7]$	$[-2,-1]$	3.29 ± 0.48	3.25 ± 0.45	* **3.20** ± 0.44
seg	$[2,6]$	$[-2,-1]$	**3.62** ± 0.87	3.70 ± 0.79	3.65 ± 0.83
veh	$[2,6]$	$[-2,-1]$	**15.51** ± 2.57	15.62 ± 2.41	15.53 ± 2.31
win	$[2,6]$	$[0,1]$	2.05 ± 2.18	**2.01** ± 2.15	2.08 ± 2.17
vow	$[-1,3]$	$[-2,-1]$	1.62 ± 0.87	1.61 ± 0.90	**1.53** ± 0.88

Table 5: Classification errors (%) of hard-margin SVMs. $\log_2(\sigma) \in [0,9]$.

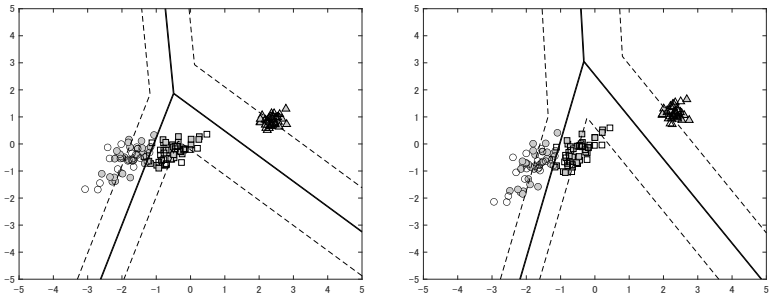
Data	AT	RGMN ($p=2$)	RGMN ($p=4$)
acc	21.08 ± 4.98	* 19.03 ± 4.38	* **18.83** ± 4.63
bal	1.07 ± 1.15	* 0.53 ± 0.89	* **0.40** ± 0.77
car	0.51 ± 0.48	**0.43** ± 0.49	0.45 ± 0.48
der	3.99 ± 1.72	**3.77** ± 1.77	3.90 ± 1.79
for	**16.86** ± 3.00	⋆ 17.75 ± 2.95	⋆ 18.20 ± 3.11
iri	7.04 ± 3.87	* 5.96 ± 3.75	* **5.17** ± 3.37
opt	1.27 ± 0.35	* 1.21 ± 0.32	* **1.19** ± 0.33
pag	—	—	—
seg	4.59 ± 0.99	* 3.98 ± 0.98	* **3.82** ± 0.92
veh	20.02 ± 2.99	* 18.79 ± 2.59	* **18.62** ± 2.42
win	**2.26** ± 2.06	2.40 ± 2.05	⋆ 2.78 ± 2.28
vow	1.58 ± 0.81	1.59 ± 0.90	**1.48** ± 0.83

with $p = 2$ does for *car*. We can say that for some specific data sets, RGMNSVM actually obtains better classifiers than ATSVM.

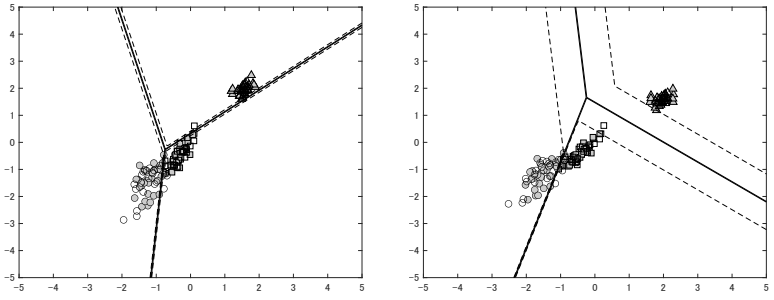
Table 5 shows the classification errors of the hard-margin ATSVM and RGMNSVM with $p = 2$ and $p = 4$. This table can be read in the same way as Table 4. The optimization problem of the hard-margin RGMNSVM is a modification of (17) in which $\xi = 0$ and $\lambda = 1$. The hard-margin ATSVM is obtained by the same manner. The kernel parameter σ is automatically selected by the cross-validation from range $\log_2(\sigma) \in [0,9]$.

From the hard-margin result, we observe a clear difference between RGMNSVM and ATSVM for several data sets, and RGMNSVM with either value of p outperforms ATSVM for *acc*, *bal*, *iri*, *opt*, *seg*, and *veh*.

To explain why the better results of hard-margin RGMNSVM vanish in the case of soft-margin, we plot the classification boundaries in the *iri* data set. Fig. 4 shows them in four different cases: ATSVM with $\lambda = 1$ or 0.01 and RGMNSVM with $p = 2$ and $\lambda = 1$ or 0.01. Both training and test instances are projected on the 2-dimensional space spanned by $w_1 - w_2$ and $w_1 - w_3$, where the gray-filled shapes are the training instances, and the white ones are the test instances. In the soft-margin case ($\lambda = 1$), the boundaries of ATSVM



(a) ATSVM ($\lambda = 1$) (b) RGMNSVM ($p = 2$, $\lambda = 1$)



(c) ATSVM ($\lambda = 0.01$) (d) RGMNSVM ($p = 2$, $\lambda = 0.01$)

Fig. 4: Boundaries of classifiers for *iri* (RBF with $\sigma = 256$) comparing different values of $\lambda \in \{1, 0.01\}$.

and RGMNSVM are similar. While, when the hard-margin case ($\lambda = 0.01$), the boundary of RGMNSVM is similar to the soft-margin case. On the other hand, all of the class-pair margins of ATSVM become too small, and the white-box instance around the center is misclassified. This phenomenon is similar to that of the artificial data D1.

6 Concluding Remarks

In this paper, we have proposed a multi-class SVM algorithm based on MMSVM (multi-objective multi-class SVM), called RGMNSVM (reciprocal-geometric-margin-norm SVM). It is derived by applying ℓ_p-norm-based scalarization and convex approximation to MMSVM. Additionally, we have developed the margin theory to justify the maximization of class-pair geometric margins. The experiments using artificial data sets describe situations in which the proposed method (RGMNSVM) works effectively, while the conventional SVMs (OVRSVM and ATSVM) fail in generalization. In performance evaluation, we have compared RGMNSVM and ATSVM on 12 benchmark data sets. The classification performances of RGMNSVM and ATSVM are generally comparable, especially in the soft-margin case. However, as one of

our main contributions, we have found that the approach of geometric margin maximization actually improves the generalization capability for certain real-world data sets.

We have several matters that remain unsolved. First, we did not discuss optimization algorithms and computation times for the proposed methods, because we used a general-purpose conic optimization solver. Optimization algorithms in which dual problems are solved by coordinate descent are popular in the literature of SVM (Lee and Lin (2013)). Therefore, in future work, we will develop such an effective algorithm for RGMNSVM. Relating to optimization algorithms, we did not consider dual problems. A study in Gotoh and Uryasev (2017) shows geometric interpretations of SVMs from the dual view point. Investigation of the dual problem of RGMNSVM is another task of future work. Furthermore, there is room for considering other norm-based scalarization. For example, the authors have proposed the scalarization based on the largest-k norm (Kusunoki and Tatsumi (2019)). A performance examination of that scalarization is also included in future work.

Acknowledgments. This work was supported by JSPS KAKENHI Grant Number JP21K12062.

Declarations

This work was supported by JSPS KAKENHI Grant Number JP21K12062. The authors have no relevant financial or non-financial interests to disclose.

All authors contributed to the study conception, design, material preparation, data collection, and analysis. The first draft of the manuscript was written by Yoshifumi Kusunoki, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

References

Abe S (2005) Support Vector Machines for Pattern Classification (Advances in Pattern Recognition). Springer-Verlag, Berlin, Heidelberg

Cortes C, Vapnik VN (1995) Support-vector networks. Machine Learning 20(3):273–297. https://doi.org/10.1023/A:1022627411411

Crammer K, Singer Y (2002) On the algorithmic implementation of multiclass kernel-based vector machines. J Mach Learn Res 2:265–292

Doğan Ü, Glasmachers T, Igel C (2016) A unified view on multi-class support vector classification. Journal of Machine Learning Research 17(45):1–32

Dua D, Graff C (2017) UCI machine learning repository. URL http://archive.ics.uci.edu/ml

Ehrgott M (2005) Multicriteria Optimization. Springer-Verlag, Berlin, Heidelberg

Gotoh Jy, Uryasev S (2017) Support vector machines based on convex risk functions and general norms. Annals of Operations Research 249(1):301–328. https://doi.org/10.1007/s10479-016-2326-x

Hill SI, Doucet A (2007) A framework for kernel-based multi-category classification. J Artif Int Res 30(1):525–564

Hsu CW, Lin CJ (2002) A comparison of methods for multiclass support vector machines. IEEE Transactions on Neural Networks 13(2):415–425. https://doi.org/10.1109/72.991427

Kapelevich L, Andersen ED, Vielma JP (2022) Computing conjugate barrier information for nonsymmetric cones. Journal of Optimization Theory and Applications https://doi.org/10.1007/s10957-022-02076-1

Kusunoki Y, Tatsumi K (2018) A multi-class support vector machine based on geometric margin maximization. In: Huynh VN, Inuiguchi M, Tran DH, et al (eds) Integrated Uncertainty in Knowledge Modelling and Decision Making. Springer International Publishing, Cham, pp 101–113

Kusunoki Y, Tatsumi K (2019) Scalarization for approximate multiobjective multiclass support vector machine using the large-k norm. Proceedings of the 2019 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (EUSFLAT 2019)

Lee CP, Lin CJ (2013) A Study on L2-Loss (Squared Hinge-Loss) Multiclass SVM. Neural Computation 25(5):1302–1323. https://doi.org/10.1162/NECO_a_00434

Lee Y, Lin Y, Wahba G (2004) Multicategory support vector machines. Journal of the American Statistical Association 99(465):67–81. https://doi.org/10.1198/016214504000000098

Liu L, Martín-Barragán B, Prieto FJ (2021) A projection multi-objective svm method for multi-class classification. Computers & Industrial Engineering 158:107,425. https://doi.org/10.1016/j.cie.2021.107425

Matsugi Y, Sugimoto T, Qi Y, et al (2018) Approximate multiobjective multiclass svm by using the reference point method. In: 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp 3535–3540

Mohri M, Rostamizadeh A, Talwalkar A (2018) Foundations of Machine Learning, 2nd edn. The MIT Press

MOSEK ApS (2022) MOSEK Optimization Toolbox for MATLAB 10.0.33. URL https://docs.mosek.com/10.0/toolbox/index.html

Rifkin R, Klautau A (2004) In defense of one-vs-all classification. J Mach Learn Res 5:101–141

Tatsumi K, Tanino T (2014) Support vector machines maximizing geometric margins for multi-class classification. TOP 22(3):815–840. https://doi.org/10.1007/s11750-014-0338-8

Tatsumi K, Hayashida K, Kawachi R, et al (2010) Multiobjective multiclass support vector machines maximizing geometric margins. Pacific Journal of Optimization 6:115–140

Tatsumi K, Akao M, Kawachi R, et al (2011) Performance evaluation of multiobjective multiclass support vector machines maximizing geometric margins. Numerical Algebra, Control and Optimization 1(1):151–169. https://doi.org/10.3934/naco.2011.1.151

Vapnik VN (1998) Statistical Learning Theory. A Wiley-Interscience Publication, New York

Weston J, Watkins C (1999) Support vector machines for multi-class pattern recognition. In: ESANN, pp 219–224