Multi-Class Support Vector Machine with Maximizing Minimum Margin

Feiping Nie[1,2*], **Zhezheng Hao**[1,2], **Rong Wang**[2]

[1]School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China
[2] School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China

Abstract

Support Vector Machine (SVM) stands out as a prominent machine learning technique widely applied in practical pattern recognition tasks. It achieves binary classification by maximizing the "margin", which represents the minimum distance between instances and the decision boundary. Although many efforts have been dedicated to expanding SVM to multi-class case through strategies such as one versus one and one versus the rest, satisfactory solutions remain to be developed. In this paper, we propose a novel method for multi-class SVM that incorporates pairwise class loss considerations and maximizes the minimum margin. Adhering to this concept, we derive a formulation through a new multi-objective optimization strategy. Furthermore, the correlations between the proposed method and multiple forms of multi-class SVM are analyzed. Empirical evaluations demonstrate the effectiveness and superiority of our proposed method over existing multi-classification methods. Complete version is available at https://arxiv.org/pdf/2312.06578.pdf. Code is available at https://github.com/zz-haooo/M3SVM.

Introduction

Support vector machine (SVM), a fundamental machine learning technique, initially emerged as a binary linear classifier (Boser, Guyon, and Vapnik 1992). Rooted in the theory of VC-dimension, SVM achieves structural risk minimization by maximizing the margin between two class. Its mathematical rigor and notable performance in practical applications have garnered significant attention. SVM-based classification methods have found extensive application in diverse machine learning tasks, including image classification (Wei and Hoai 2016), text classification (Nie et al. 2014), etc. Diverse SVM variants have emerged over time, such as Twin SVM (Khemchandani, Chandra et al. 2007), Optimal Margin Distribution Machine (Zhang and Zhou 2019),Decision Tree SVM (Nie, Zhu, and Li 2020), etc. Furthermore, SVM has been extended to encompass various scenarios and guides advanced methodologies and models (Tarzanagh et al. 2023; Xu and Schuurmans 2005; Amer, Goldstein, and Abdennadher 2013).

Despite its outstanding performance in binary classification tasks, multi-class SVM progresses sluggishly. Existing multi-class SVM can be summarized into two main categories, the first of which are One versus the Rest (OvR) (Vapnik 1999) and One versus One (OvO) techniques (Hsu and Lin 2002). For a c-class classification, OvR utilizes c binary SVMs to separate each class from the rest, while OvO utilizes $\frac{c(c-1)}{2}$ binary SVMs to separate each pair of classes. A critical shortcoming of OvR arises from the unbalancedness of each subproblem. Besides, treating the rest of the classes as a single class may lead to potential inseparability (Pisner and Schnyer 2020). The drawback of OvO lies in the high time overhead of testing. OvO needs $\mathcal{O}(c^2d)$ for each test sample, which is time-consuming when c is large. Since OvR and OvO involve multiple independent binary subproblems, they fail to achieve a complete partition of the feature space and results in certain regions with constant tied votes.

The second major method is multi-class SVM with unified formulation (Weston and Watkins 1998; Crammer and Singer 2001; Guermeur 2002). While these multi-class methods achieve sound performance on well-structured data, they may be less effective for datasets with ambiguous class boundaries. Moreover, many of these multi-class SVM methods deviate from the fundamental principle of margin, thus impacting generalization capabilities.

To address the aforementioned issues, we propose a novel method called Multi-class **S**upport **V**ector **M**achines with **M**aximizing **M**inimum **M**argin (**M^{3}SVM**) to overcome the limitations of existing methods. The main contributions of this paper are summarized as follows:

- In this paper, we propose a concise yet effective multi-class SVM method that computes classification loss between each class pair. More importantly, we derive a novel regularizer that enlarges the lower bound of the margin by introducing a parameter p.
- The proposed method bases on a strategy for multi-objective optimization and offers a lucid geometric interpretation. We theoretically analyze the association of M^{3}SVM with the previous methods. It also functions as a plug-and-play improvement over the softmax in neural networks. Besides, the proposed regularizer can be interpreted in the context of minimizing structural risk.
- Through exhaustive experiments on realistic datasets, our proposed method demonstrates a marked enhancement in classification performance.

*Corresponding author.

The proofs of all involved lemmas and theorems together with supplementary experiments are relegated to Appendix in the complete version.

Notations: The vectors and matrices are denoted by bold lowercase and bold uppercase letters, respectively. Set $\{1, 2, \cdots, n\}$ is abbreviated by $[n]$ for simplicity.

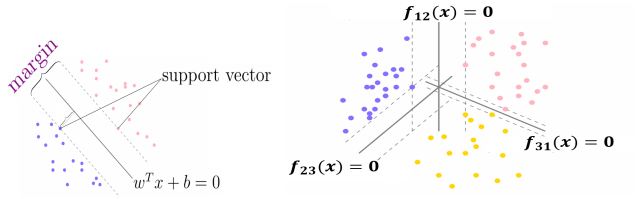
Related Work

Binary SVM

SVM was first proposed under the form of hard margin, whose basic formulation can be written as the following optimization problem (Boser, Guyon, and Vapnik 1992):

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2, \quad \text{s.t. } y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geqslant 1, i \in [n], \quad (1)$$

where $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ is the train set. SVM aims to seek a separating hyperplane by maximizing the margin between two classes such that $\mathbf{w}^T\mathbf{x}_i + b - 1 \geqslant 0$ with $y_i = 1$ and $\mathbf{w}^T\mathbf{x}_i + b + 1 \leqslant 0$ with $y_i = -1$. The principle of SVM is demonstrated in Figure 1a. It maximizes the "margin" in the figure through searching the appropriate support vectors. The separating hyperplane $\mathbf{w}^T\mathbf{x}+b = 0$ obtained by solving the quadratic programming problem above can be employed to classify the upcoming data without label.



(a) Binary SVM. (b) Multi-class SVM.

Figure 1: From binary SVM to multi-class SVM.

Cortes and Vapnik (Cortes and Vapnik 1995) proposed SVM with soft margin by introducing slack variables,

$$\begin{aligned} &\min_{\mathbf{w},b} \sum_{i=1}^n \xi_i + \lambda\|\mathbf{w}\|_2^2, \\ &\text{s.t. } y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geqslant 1 - \xi_i, i \in [n]. \end{aligned} \quad (2)$$

Each slack variable corresponds to one sample, depicting the degree of unsatisfied constraints. By substituting the slack variables, Eq. (2) can be formulated as the form of "loss + regularization". Thus, the general form of soft margin SVM with hinge loss is as follows:

$$\min_{\mathbf{w},b} \sum_{i=1}^n [1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)]_+ + \lambda\|\mathbf{w}\|_2^2. \quad (3)$$

where $[\cdot]_+ = max\{0, \cdot\}$. λ is a trade-off parameter to weigh the two objectives. Subsequent studies have refined the vanilla binary SVM in multiple ways (Crisp and Burges 1999; Mangasarian and Musicant 2001; Grandvalet et al. 2008; Ladicky and Torr 2011; Zhou et al. 2012; Nie et al. 2014; Zhang et al. 2017). Nevertheless, the exploration of multi-class SVM remains incomplete. A few representative methods are outlined below.

Multi-class SVM with Unified Formulation

When there is c classes, the decision function is of the following form

$$y = arg\max_{k\in[c]} \mathbf{w}_k^T\mathbf{x} + b_k. \quad (4)$$

In this way, each class corresponds to one projection vector $\mathbf{w}$. In accordance with such decision function, a series of multi-class SVM models have been proposed. Weston and Watkins (Weston and Watkins 1998) integrated multi-class SVM into a unified framework rather than solving multiple subproblems separately, which is formulated as

$$\begin{aligned} &\min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \sum_{i=1}^n \sum_{k\neq y_i} \xi_{ki} + \lambda \sum_{k=1}^c \|\mathbf{w}_k\|_2^2 \\ &s.t. \begin{cases} \mathbf{w}_{y_i}^T\mathbf{x}_i + b_{y_i} \geq \mathbf{w}_k^T\mathbf{x}_i + b_k + 1 - \xi_{ki}, \\ \xi_{ki} \geqslant 0, k \in [c], k \neq y_i, i \in [n]. \end{cases} \end{aligned} \quad (5)$$

where $\mathbf{W} \in \mathbb{R}^{d\times c}$ and $\mathbf{b} \in \mathbb{R}^c$ are applied for test data through Eq. (4). Crammer and Singer (Crammer and Singer 2001) proposed a new-look loss function from the perspective of decision function, which is formulated as follows.

$$\begin{aligned} &\min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \sum_{i=1}^n \xi_i + \lambda \sum_{k=1}^c \|\mathbf{w}_k\|_2^2 \\ &s.t. \mathbf{w}_{y_i}^T\mathbf{x}_i + \delta_{y_i,k} - \mathbf{w}_k^T\mathbf{x}_i \geqslant 1 - \xi_i, \xi_i > 0, i \in [n]. \end{aligned} \quad (6)$$

where $\delta_{y_i,k}$ equals to 1 if $k = y_i$ and 0 otherwise.

Denoting $[1 - (\mathbf{w}_{y_i}^T\mathbf{x}_i + b_{y_i} - \mathbf{w}_k^T\mathbf{x}_i - b_k)]_+$ as Δ_{ik}, the optimization objective in Eq. (5) can be converted to

$$\min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \sum_{i=1}^n \sum_{k\neq y_i} \Delta_{ik} + \lambda \sum_{k=1}^c \|\mathbf{w}_k\|_2^2. \quad (7)$$

By adding bias term $\mathbf{b}$, Eq. (6) can be rewritten as

$$\min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \sum_{i=1}^n \max_{k\neq y_i} \Delta_{ik} + \lambda \sum_{k=1}^c \|\mathbf{w}_k\|_2^2. \quad (8)$$

This makes it clear that per-sample loss in Eq. (5) is the sum of loss of misclassification while that in Eq. (6) is the maximum loss.

The two methods above represent unified multi-class models, however, the explicit interpretation of the margin concept is lacking. Bredensteiner and Bennett (Bredensteiner and Bennett 1999) constructed the M-SVM upon the definition of piecewise-linear separability, which introduced explicit margin for multi-class SVM. Guermeur (Guermeur 2002) explained multi-class SVM based on uniform strong law of large numbers and proved the equivalence with (Bredensteiner and Bennett 1999). The aforementioned methods are representative, and subsequent work mainly revolves around their improvement (Vural and Dy 2004; Lauer et al. 2011; G van den Burg and Groenen 2016). The explanation from the margin perspective was collated by Xu et al. (Xu et al. 2017). Nie el al. proposed capped ℓ_p-norm multi-class SVM to deal with light and heavy outliers (Nie, Wang, and Huang 2017). Doan et al. (Doan, Glasmachers, and

Igel 2016) composed and experimented with the aforementioned multi-class SVM methods. Lapin et al. (Lapin, Hein, and Schiele 2015) further proposed a new multi-class SVM model based on a tight convex upper bound of the top-k error.

Method

Problem Setup

Rethinking the decision hyperplane from a simple perspective, with each class associated with parameters $(\mathbf{w}, b)$, a desirable linear multi-classifier is supposed to meet

$$\mathbf{w}_{y_i}^T \mathbf{x}_i + b_{y_i} > \mathbf{w}_k^T \mathbf{x}_i + b_k, k \neq y_i, \forall i \in [n]. \tag{9}$$

Accordingly, a c-classification problem can be implemented by solving c vectors. From this, the distinction between samples belonging to class j and class k is established through the decision function:

$$f_{kl}(\mathbf{x}) = (\mathbf{w}_k - \mathbf{w}_l)^T \mathbf{x} + b_k - b_l, k < l. \tag{10}$$

Thereby, $f_{kl}(\mathbf{x}) = 0$ is the corresponding separating hyperplane, $f_{kl}(\mathbf{x}) > 0$ for class k and $f_{kl}(\mathbf{x}) < 0$ for class l. The illustration of our multi-class SVM is shown in Figure 1b.

To achieve OvO strategy with a unified model, our multi-class SVM seeks separating hyperplanes by maximizing the margin between two classes such that $(\mathbf{w}_k - \mathbf{w}_l)^T \mathbf{x}_i + b_k - b_l \geqslant 1$ with $y_i = k$ and $(\mathbf{w}_k - \mathbf{w}_l)^T \mathbf{x}_i + b_k - b_l \leqslant -1$ with $y_i = l$. Similar to binary case in Figure 1a, the margin of separating between class k and class l is

$$Margin\,(C_k, C_l) = 2d_{kl} = \frac{2}{\|\mathbf{w}_k - \mathbf{w}_l\|_2}. \tag{11}$$

For multi-class SVM, the ideal scheme would be to maximize the margin between class pair. However, due to the mutual limitation between hyperplanes, multi-class SVM cannot do what OvO method does. Since c vectors are expected to represent $\frac{c(c-1)}{2}$ hyperplanes, there exists mutual restriction between hyperplanes. Therefore, concurrently maximizing the margin between each class pair is impractical.

(a) Large margin summation. (b) Small margin summation.

Figure 2: Larger margin summation may not lead to better classification performance.

Then, would optimizing the sum of all margins be a good choice? Figures 2a and 2b give an intuitive no. The black dashed lines represent the margins between class pairs. Obviously, Figure 2a has a larger margin summation, but there is hard-to-split class pair. Figure 2b is the preferred classifier although with a smaller margin summation. This implies that multi-class SVM is a challenging multi-objective optimization. In the following subsection we propose a strategy to solve this challenge.

A Strategy for Multi-objective Optimization

Multi-objective optimization refers to achieving multiple conflicting objectives in specific scenarios, where optimizing one objective comes at the expense of others. Suppose we want to maximize the following objectives: $g_1(\mathbf{z}), g_2(\mathbf{z}), \cdots, g_m(\mathbf{z})$. A fundamental proposal for multi-objective optimization is to optimize the worst case, i.e.,

$$\max_{\mathbf{z}} \min_{i \in [m]} g_i(\mathbf{z}). \tag{12}$$

Obviously, the minimization function is hard to address. We propose the following two alternatives to approximate the minimization function[1]:

$$\begin{aligned} &(a)\ \min_{i \in [m]} g_i(\mathbf{z}) = -p \log \left(\sum_{i=1}^{m} e^{-\frac{g_i(\mathbf{z})}{p}} \right), p \to 0. \\ &(b)\ \min_{i \in [m]} g_i(\mathbf{z}) = \left[\sum_{i=1}^{m} g_i^{-p}(\mathbf{z}) \right]^{-\frac{1}{p}}, p \to \infty. \end{aligned} \tag{13}$$

In this way, the max-min problem (12) is converted to a straightforward maximization problem. We introduce p into the model as a tunable hyperparameter so that the strategy is not only an approximation for problem (12), but also takes all the objectives into account.

When the optimization problem becomes

$$\min_{\mathbf{z}} \max_{i \in [m]} g_i(\mathbf{z}), \tag{14}$$

our strategy can be applied correspondingly

$$\begin{aligned} &(a)\ \max_{i \in [m]} g_i(\mathbf{z}) = p \log \left(\sum_{i=1}^{m} e^{\frac{g_i(\mathbf{z})}{p}} \right), p \to 0. \\ &(b)\ \max_{i \in [m]} g_i(\mathbf{z}) = \left[\sum_{i=1}^{m} g_i^{p}(\mathbf{z}) \right]^{\frac{1}{p}}, p \to \infty. \end{aligned} \tag{15}$$

A New Formulation for Multi-class SVM

We address multi-class SVM by maximizing the minimum margin, thereby ensuring the margin between each class pair is not excessively small. This is consistent with problem (12). Then our multi-class SVM optimization objective can be formulated as follows:

$$\begin{aligned} &\max_{\mathbf{W} \in \mathbb{R}^{d \times c}, \mathbf{b} \in \mathbb{R}^c} \min_{k,l \in [c], k<l} \frac{1}{\|\mathbf{w}_k - \mathbf{w}_l\|_2}, \\ &s.t. \begin{cases} f_{kl}(\mathbf{x}_i) \geqslant 1, & y_i = k, \\ f_{kl}(\mathbf{x}_i) \leqslant -1, & y_i = l, \end{cases} \quad i \in [n]. \end{aligned} \tag{16}$$

We next convert problem (16) into a tractable form through (b) in Eq. (13).

Theorem 1. *The problem (16) is equivalent to*

$$\begin{aligned} &\min_{\mathbf{W} \in \mathbb{R}^{d \times c}, \mathbf{b} \in \mathbb{R}^c} \sum_{k=1}^{c-1} \sum_{l=k+1}^{c} \|\mathbf{w}_k - \mathbf{w}_l\|_2^p, \\ &s.t. \begin{cases} f_{kl}(\mathbf{x}_i) \geqslant 1, & y_i = k, \\ f_{kl}(\mathbf{x}_i) \leqslant -1, & y_i = l, \end{cases} \quad i \in [n]. \end{aligned} \tag{17}$$

with the given parameter $p \to \infty$.

[1] $g(\mathbf{z})$ needs to be nonnegative in (b), while it does not in (a).

Optimizing the minimum margin is not sufficient to focus on the all classes. An appropriate p is supposed to globally enlarge the margin between each class pair while enhancing the lower bound of the margins. Hyperparameter p is set in $[1, 8]$ in our experiments.

Similar to binary SVM, slack variable between each class pair can be introduced:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{b}} \sum_{k<l} \sum_{y_i\in\{k,l\}} \xi_{ikl} + \lambda \sum_{k<l} \|\mathbf{w}_k - \mathbf{w}_l\|_2^p, \\ &s.t. \begin{cases} f_{kl}(\mathbf{x}_i) \geqslant 1 - \xi_{ikl}, & y_i = k, \\ f_{kl}(\mathbf{x}_i) \leqslant -1 + \xi_{ikl}, & y_i = l, \end{cases} \quad i \in [n]. \end{aligned} \tag{18}$$

By substituting the slack variables, Eq. (2) can be transformed into an unconstrained optimization problem:

$$\min_{\mathbf{W},\mathbf{b}} \sum_{k<l} \sum_{y_i\in\{k,l\}} [1 - y_{ikl} f_{kl}(\mathbf{x}_i)]_+ + \lambda \sum_{k<l} \|\mathbf{w}_k - \mathbf{w}_l\|_2^p \tag{19}$$

where $y_{ikl} = 1$ if $y_i = k$ and $y_{ikl} = -1$ if $y_i = l$.

Assume $\mathbf{W}$ and $\mathbf{b}$ are the optimal solution of problem (19). It is obvious that, for an arbitrary $\boldsymbol{\sigma} \in \mathbb{R}^d$, suppose $\tilde{\mathbf{W}} = \mathbf{W} + \boldsymbol{\sigma}\mathbf{1}^T$, there is $\mathbf{w}_j - \mathbf{w}_k = \tilde{\mathbf{w}}_j - \tilde{\mathbf{w}}_k$. For an arbitrary $\eta \in \mathbb{R}$, suppose $\tilde{\mathbf{b}} = \mathbf{b} + \eta\mathbf{1}$, there is $\mathbf{b}_j - \mathbf{b}_k = \tilde{\mathbf{b}}_j - \tilde{\mathbf{b}}_k$. Therefore $\tilde{\mathbf{W}}$ and $\tilde{\mathbf{b}}$ is also the optimal solution. It is expected that the model has a unique optimal solution for specific data that is not contingent upon the initialization. Without loss of generality, we impose a mean-zero constraint on $\mathbf{W}$ and $\mathbf{b}$ that $\mathbf{W}\mathbf{1} = \mathbf{0}$,$\mathbf{b}^T\mathbf{1} = 0$. Here the constraint is converted into a solvable form by the following theorem.

Theorem 2. *Assume function $f(\mathbf{Z})$ has the property of column translation invariance, i.e., $\forall \boldsymbol{\sigma} \in \mathbb{R}^n$, there is $f(\mathbf{Z}) = f(\mathbf{Z} + \boldsymbol{\sigma}\mathbf{1}^T)$. With given $\varepsilon \to 0$, the following two optimization problems have the same optimal solution*

$$\min_{\mathbf{Z}\in\mathbb{R}^{n\times m}} f(\mathbf{Z}), \quad s.t. \sum_{j=1}^{m} \mathbf{z}_j = \mathbf{0}, \tag{20}$$

$$\min_{\mathbf{Z}\in\mathbb{R}^{n\times m}} f(\mathbf{Z}) + \varepsilon \|\mathbf{Z}\|_F^2. \tag{21}$$

Considering the above theorem, the mean-zero constraint can be transformed into an additional penalty term. The complete model can be written as

$$\begin{aligned} \min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} &\sum_{k<l} \sum_{y_i\in\{k,l\}} [1 - y_{ikl} f_{kl}(\mathbf{x}_i)]_+ + \\ &\lambda \sum_{k<l} \|\mathbf{w}_k - \mathbf{w}_l\|_2^p + \varepsilon \left(\|\mathbf{W}\|_F^2 + \|\mathbf{b}\|_2^2\right). \end{aligned} \tag{22}$$

λ is a trade-off parameter to balance error tolerance and enlargment of inter-class margin. For a small λ, instances that fall within the margin receive a high penalty, whereas for a larger λ, the penalty decreases. The effect of ε to the model is negligible, just for the purpose of unique solution.

Since the computation of inter-class loss is expensive in practice, we simplify it by the following theorem.

Theorem 3. *The following equation holds,*

$$\sum_{k<l} \sum_{y_i\in\{k,l\}} [1 - y_{ikl} f_{kl}(\mathbf{x}_i)]_+ = \sum_{i=1}^{n} \sum_{k\neq y_i} [1 - f_{y_i k}(\mathbf{x}_i)]_+. \tag{23}$$

According to Theorem 3, problem (22) is equivalent to

$$\begin{aligned} \min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} &\sum_{i=1}^{n} \sum_{k\neq y_i} [1 - f_{y_i k}(\mathbf{x}_i)]_+ + \\ &\lambda \sum_{k<l} \|\mathbf{w}_k - \mathbf{w}_l\|_2^p + \varepsilon \left(\|\mathbf{W}\|_F^2 + \|\mathbf{b}\|_2^2\right). \end{aligned} \tag{24}$$

In this way, we transform our loss into the summation of the individual sample losses during optimization.

Smoothness and Convexity

Our method is intended to be applicable to gradient optimization, so a smooth loss function is needed. We employ the following straightforward function to approximate $[x]_+$:

$$g(x) = \frac{x + \sqrt{x^2 + \delta^2}}{2}, \quad (\delta > 0). \tag{25}$$

Lemma 1. *$g(x)$ satisfies $0 \leqslant g(x) - [x]_+ \leqslant \frac{\delta}{2}$. When $\delta \to 0$, $g(x) \to [x]_+$.*

The closeness between two functions exclusively depends on the proximity factor δ. By replacing the hinge loss, our overall model can be formulated as

$$\begin{aligned} \min_{\mathbf{W}\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} &\sum_{i=1}^{n} \sum_{k\neq y_i} \frac{\gamma_{ik} + \sqrt{\gamma_{ik}^2 + \delta^2}}{2} + \\ &\lambda \sum_{k<l} \|\mathbf{w}_k - \mathbf{w}_l\|_2^p + \varepsilon \left(\|\mathbf{W}\|_F^2 + \|\mathbf{b}\|_2^2\right). \end{aligned} \tag{26}$$

where $\gamma_{ik} = 1 - f_{y_i k}(\mathbf{x}_i)$. The decision function between class k and class l is $f_{kl}(\mathbf{x}_i) = (\mathbf{w}_k - \mathbf{w}_l)^T \mathbf{x}_i + b_k - b_l$. It is worth noting that the alterations of ε and δ wield negligible influence on the model. The only parameters that affect the model performance are actually p and λ.

Theorem 4. *Problem (26) is strictly convex.*

Therefore, we adopt Adam (Kingma and Ba 2014) optimization strategy to update $\mathbf{W}$ and $\mathbf{b}$ in problem (26), which enables our method to be applied to the deep model. For the sample $(\mathbf{x}_i, y_i)$ in the train set, there are two cases when we take derivative of the objective function Eq. (26) with respective to $\mathbf{W}$ by column. If $k = y_i$, the derivative with respect to $\mathbf{w}_k$ at iteration t is

$$\begin{aligned} \nabla_k^{(t)} = &- \sum_{l\neq k} \frac{\gamma_{il} + \sqrt{\gamma_{il}^2 + \delta^2}}{2\sqrt{\gamma_{il}^2 + \delta^2}} \mathbf{x}_i + 2\varepsilon \mathbf{w}_k + \\ &\sum_{l\neq k} \lambda p \|\mathbf{w}_k - \mathbf{w}_l\|_2^{p-2} (\mathbf{w}_k - \mathbf{w}_l). \end{aligned} \tag{27}$$

If $k \neq y_i$, the derivative goes to:

$$\begin{aligned} \nabla_k^{(t)} = &\frac{\gamma_{ik} + \sqrt{\gamma_{ik}^2 + \delta^2}}{2\sqrt{\gamma_{ik}^2 + \delta^2}} \mathbf{x}_i + 2\varepsilon \mathbf{w}_k + \\ &\sum_{l\neq k} \lambda p \|\mathbf{w}_k - \mathbf{w}_l\|_2^{p-2} (\mathbf{w}_k - \mathbf{w}_l). \end{aligned} \tag{28}$$

Note that the optimization is performed for $\mathbf{W}$ as a whole, independent of the order in which the columns are updated. The convexity of the objective function ensures its convergence to the global optimum through the employment of Adam. After the training process, test sample $\mathbf{x}$ can be classified through $y = arg\max_k \mathbf{w}_k^T\mathbf{x} + b_k$.

Connection to ℓ_2-regularizer

One might consider replacing the regularizer with the form of ℓ_2-norm sum: $\sum_{k=1}^{c}\|\mathbf{w}_k\|_2^2$, like the regularization of most classifiers (Zhang et al. 2003; Raman et al. 2019). The relationship between the two is summarized as follows.

Lemma 2. *The following equation holds,*

$$\sum_{k=1}^{c-1}\sum_{l=k+1}^{c}\|\mathbf{w}_k-\mathbf{w}_l\|_2^2 = c\sum_{k=1}^{c}\|\mathbf{w}_k - \frac{1}{c}\sum_{l=1}^{c}\mathbf{w}_l\|_2^2. \quad (29)$$

According to Theorem 2, there is $\sum_{l=1}^{c}\mathbf{w}_l = \mathbf{0}$ when $\mathbf{W}$ is taken to be optimal in problem (26). So we draw the conclusion that with $p = 2$, problem (26) and the problem replacing the regular term with ℓ_2 regularizer have the same optimal solution.

Theorem 5. *With $p = 2$, the optimization problem (24) and the following problem has the same optimal solution*

$$\min_{\mathbf{W},\mathbf{b}}\sum_{i=1}^{n}\sum_{k\neq y_i}[1-f_{y_ik}(\mathbf{x}_i)]_+ + \lambda c\sum_{k=1}^{c}\|\mathbf{w}_k\|_2^2 + \varepsilon\|\mathbf{b}\|_2^2. \quad (30)$$

According to Theorem 5, Eq. (5) can be regarded as a special case of our method when $p = 2$. The only difference is that Eq. (5) has no constraint on $\mathbf{b}$, which leads to non-unique solutions. However, the naive ℓ_2 regularizer does not have a clear geometric meaning. As will be demonstrated in the experiments, $p = 2$ is not optimal in a wide range of cases.

Structural Risk Minimization

We elucidate the explanation of the prposed M^{3}SVM from structural risk minimization (SRM) inductive principle. Following the fundamental assumptions in statistical learning theory, there is a canonical but unknown joint distribution on $\mathcal{X}\times\mathcal{C}$. The goal of learning is to select a function $f:\mathbf{x}\to\mathbb{R}^c$ (or in terms of probability $f:\mathbf{x}\to[0,1]^c$), among from a specific design functions space $\mathcal{F}$, such that its error on the joint distribution is minimized. The discriminant function for the classification problem is typically in the form of $g(\mathbf{x}) = \max_j f_j(\mathbf{x})$. The risk of the classification task can be written as $\mathcal{R}(f) = \int\mathbb{I}(g(\mathbf{x})\neq y)dP(\mathbf{x},y)$. Since $P(\mathbf{x},y)$ is unknown, one shallow solution is to minimize empirical risk $\mathcal{R}_e(f) = \frac{1}{n}\sum_i^n\mathbb{I}(g(\mathbf{x})\neq y)$ on certain samples. SRM inductive principle is a more recognized technique, which is based on the theory that for any $f\in\mathcal{F}$ with a probability of at least $1-\rho$, the risk meets $\mathcal{R}\leqslant\mathcal{R}_e+\Omega(\mathcal{F},\rho,n)$, where Ω is called guaranteed risk and can be expressed in the form of VC dimension (Vapnik and Chervonenkis 2015), Rademacher complexity (Bartlett and Mendelson 2002), etc. The learnable basis of the binary SVM (Schiilkop, Burgest, and Vapnik 1995) is to reduce the risk of VC dimensional form by minimizing $\|\mathbf{w}\|_2$. For multi-class SVM model, f can be set as a multi-valued function $f:\mathbf{x}\to\mathbf{w}_k\mathbf{x}+b_k, k\in[c]$. Note that different f in $\mathcal{F}$ are only differ in the parameters $\mathbf{W}$ and $\mathbf{b}$ in this case.

The theory of generalized risk derived from Uniform Strong Law of Large Numbers (Guermeur 2002) implement the SRM inductive principle by delineating a compromise between training performance and complexity. For multi-class SVM model, minimizing its guaranteed risk can be approximately equated to minimizing a norm of the linear operator $\|T(f)\|_\omega$, where functional $T:\mathcal{F}\to\mathbf{M}_{2n\times c(c-1)/2}$ mapping a function to a real matrix. The norm is chosen in accordance with the choice of the pseudo-metric on $\mathcal{F}$, for instance, $\forall(f,\bar{f})\in\mathcal{F}^2$,

$$\begin{aligned}\omega_{l_\infty,l_1}(f,\bar{f}) &= \max_x\sum_{k<l}\left|f_k(x)-\bar{f}_k(x)\right|,\\ \omega_{l_\infty,l_\infty}(f,\bar{f}) &= \max_x\max_{k<l}\left|f_k(x)-\bar{f}_k(x)\right|,\end{aligned} \quad (31)$$

which correspond to matrix norm $\|\mathbf{M}\|_{l_\infty,l_1}$ and $\|\mathbf{M}\|_{l_\infty,l_\infty}$ respectively. Define $T(f) = [t^{(1)},\cdots,t^{(2n)}]^T$, where $t^{(i)}(f) = [(\mathbf{w}_1-\mathbf{w}_2)^T\mathbf{x}_i,\cdots,(\mathbf{w}_k-\mathbf{w}_l)^T\mathbf{x}_i,\cdots,(\mathbf{w}_{c-1}-\mathbf{w}_c)^T\mathbf{x}_i]^T\in\mathbf{M}_{1\times c(c-1)/2}$. The matrix norm of $T(f)$ provides a tight upper bound for the guaranteed risk Ω. In M^{3}SVM, the infimum of margin bears a close relationship with the crude upper bound of the norm of $T(f)$. It can be put down to the following theorem.

Theorem 6. *Let $\mathcal{F}$ be the multivariate linear model from $\mathcal{X}$ into $\mathbb{R}^c$. $\mathcal{F}$ are endowed with the Euclidean norm. If $\mathcal{X}$ is included in a ball of radius $\Lambda_\mathcal{X}$ about the origin, $\forall f\in\mathcal{F}$(parametrized by $\mathbf{W}$ and $\mathbf{b}$) the following bound holds:*

$$\|T(f)\|_{l_\infty,l_p}\leqslant\Lambda_\mathcal{X}\left(\sum_{k<l}\|\mathbf{w}_k-\mathbf{w}_l\|_2^p\right)^{\frac{1}{p}}. \quad (32)$$

Our regularizer optimizes an upper bound on the guaranteed risk derived from covering numbers (Guermeur 2002). Combining the methods described in the previous section, when $p\to\infty$, minimizing $\sum_{k=1}^{c-1}\sum_{l=k+1}^{c}\|\mathbf{w}_k-\mathbf{w}_l\|_2^p$ is equivalent to maximizing $\inf_{k<l}Margin\,(C_k,C_l)$, which reduces the upper bound of guaranteed risk. Moreover, this method is essentially adapted to multiple metrics of $\mathcal{F}$, whereas the previous methods correspond to a special case when $p = 2$. We draw the conclusion that our method is interpretable in terms of SRM and it is altogether possible to improve the generalization performance (i.e. reduce the guaranteed risk Ω) by maximizing the minimum margin.

Extension to Softmax Loss

Our proposed method exhibits versatility, extending its applicability to other linear classifiers, such as logistic regression (LR). By altering the misclassification loss, our method acts as a regularized softmax loss and can be applied to the last layer of the neural network. As it can learn embeddings with large inter-class margins, the proposed loss guides the learning of network parameters through backpropagation. Geometric interpretation and discussions are relegated to Appendix A.3.

Methods	OvR	OvO	Crammer	M-SVM	Top-k	Multi-LR	SMLR	M^{3}SVM
Cornell	0.812 ± 0.065	0.845 ± 0.028	0.792 ± 0.015	0.755 ± 0.031	0.826 ± 0.016	0.783 ± 0.026	0.803 ± 0.009	$\mathbf{0.865 \pm 0.013}$
ISOLET	0.866 ± 0.046	0.942 ± 0.004	0.922 ± 0.042	0.910 ± 0.004	0.904 ± 0.013	0.940 ± 0.004	0.926 ± 0.008	$\mathbf{0.945 \pm 0.002}$
HHAR	0.845 ± 0.059	0.966 ± 0.014	0.931 ± 0.039	0.953 ± 0.008	0.970 ± 0.007	0.948 ± 0.010	0.952 ± 0.012	$\mathbf{0.981 \pm 0.004}$
USPS	0.887 ± 0.042	0.898 ± 0.005	0.769 ± 0.047	0.910 ± 0.018	0.825 ± 0.009	0.932 ± 0.002	0.937 ± 0.004	$\mathbf{0.956 \pm 0.011}$
ORL	0.919 ± 0.021	$\mathbf{0.975 \pm 0.000}$	0.879 ± 0.018	0.790 ± 0.034	0.879 ± 0.028	0.925 ± 0.000	0.925 ± 0.000	$\mathbf{0.975 \pm 0.000}$
Dermatology	0.939 ± 0.009	0.971 ± 0.003	0.933 ± 0.015	0.868 ± 0.031	0.891 ± 0.047	0.965 ± 0.007	0.965 ± 0.010	$\mathbf{0.988 \pm 0.001}$
Vehicle	0.794 ± 0.016	0.756 ± 0.024	0.757 ± 0.021	0.762 ± 0.019	0.778 ± 0.007	0.780 ± 0.010	0.771 ± 0.020	$\mathbf{0.800 \pm 0.011}$
Glass	0.656 ± 0.075	0.685 ± 0.008	0.594 ± 0.045	0.629 ± 0.044	0.674 ± 0.025	0.664 ± 0.018	0.679 ± 0.015	$\mathbf{0.744 \pm 0.007}$

Table 1: Average performance (w.r.t. Accuracy) on test set over 10 runs by different methods.

Experiments

In this section, we empirically evaluate the effectiveness of our method on multi-class classification task and analyze the experimental results.

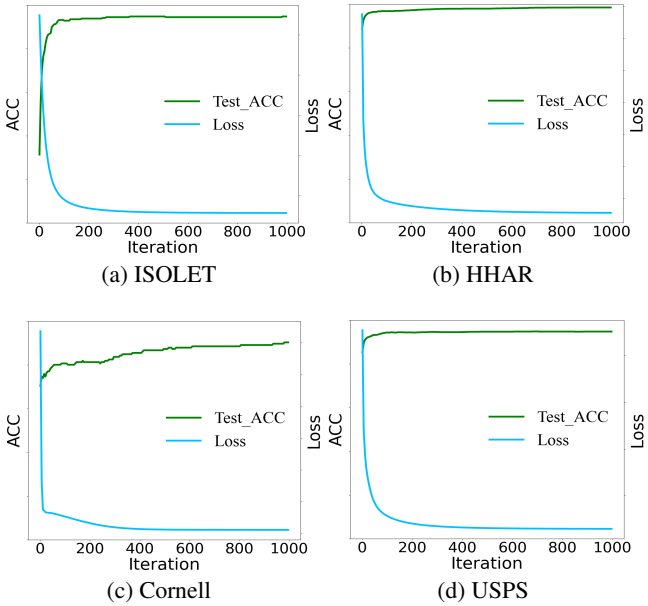
Experiment Settings

The datasets chosen for evaluation include Cornell, ISOLET, HHAR, USPS, ORL, Dermatology, Vehicle and Glass, which represent diverse data types (including image, speech, document, etc). They can all be found at [2]. The details of the datasets are described in Appendix A.4. Our method is compared with six linear multi-classification methods, including OvR(Vapnik 1999), OvO (Hsu and Lin 2002), Crammer (Crammer and Singer 2001), M-SVM (Bredensteiner and Bennett 1999), Top-k (Lapin, Hein, and Schiele 2015), Multi-LR (Böhning 1992) and Sparse Multinomial Logistic Regression (Krishnapuram et al. 2005) (SMLR). For the sake of fairness and generalizability, all methods are directly trained in the original feature space rather than in well-selected kernel spaces. For the hyperparameters involved in M^{3}SVM, λ is set to ten equidistant values within the interval $[1 \times 10^{-4}, 1 \times 10^{-1}]$, while p is set on a grid of $[1, 2, \cdots, 8]$. For all comparative methods, we adhere to the authors' default parameter settings and, where necessary, similarly conduct parameter grid searches to achieve fair comparisons as far as possible. Since the selected standard datasets do not suffer from class imbalance, it is convictive to employ test accuracy (ACC) as the sole evaluation criterion of the methods. We evaluate the performance of each method and report the average results of 10 runs.

Results

The experimental results of M^{3}SVM and seven comparative methods are reported in Table 1, where each result represents the average test accuracy and standard deviation of 10 runs. The best result on each dataset is marked in bold. In comparison to other widely used multi-classification algorithms, M^{3}SVM achieves the best classification performance on all selected datasets, which can be attributed to the flexible factor p that adapt to diverse data structures. It is noteworthy that the OvO method generally exhibits great performance, owing to its independent separation of any two classes. Unfortunately, the complexity of classification for new arrival data considerably limits its application. In addition to this, the tuning parameter λ for each subproblem in OvO are not straightforward, which accounts for the suboptimal experimental results.

Beyond the convexity, the sound convergence property is experimentally verified. The variation of the objective function values and the accuracy (ACC) on test set over the number of iterations is depicted in Figure 3 on six datasets. Throughout the entire training process, accuracy consistently improves as the loss function decreases. It can be found that M^{3}SVM converges rapidly, with convergence observed within 500 iterations across all datasets.



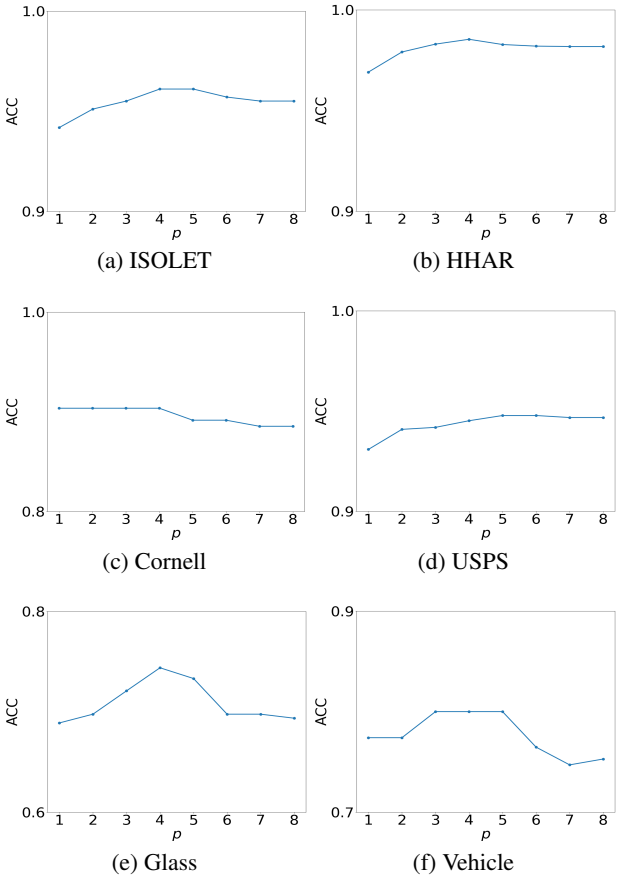
(a) ISOLET (b) HHAR

(c) Cornell (d) USPS

Figure 3: Convergence of the objective function value.

[2] https://archive.ics.uci.edu/ml/datasets.php

The test accuracy under various values of p on six datasets is presented in Figure 4. A noteworthy observation is that as the value of p increases, the model's generalization performance (test accuracy) initially peaks and subsequently diminishes (note that it is totally possible that the peak is not within $[1, 8]$). This phenomenon aligns with our motivation, where p acts as a balancing factor between the global margins and the lower bound of the margins. The increase of p can be interpreted as a prioritization of enhancing the lower bound of the margin. When p is small, enlarging the margin between each class pair contributes to the reduction of the objective function. When p is large, the objective function primarily emphasizes boosting the lower bound of margins. In such case, the value of the rest margins are pulled down due to the interlocking separating hyperplanes. Through extensive experiments, we found model performance is generally better when p is around 4, which can serve as a reasonable prior. Furthermore, it's advisable to avoid excessively large values of p, as they may result in poor convergence.



(a) ISOLET (b) HHAR

(c) Cornell (d) USPS

(e) Glass (f) Vehicle

Figure 4: The effect of parameter p on experimental results.

We study the sensitivity of the trade-off parameter λ. Figure 5 illustrates the variation of the test accuracy on the eight datasets at $p = 4$ as a function of λ, within the range of $[1 \times 10^{-4}, 1]$. One can infer that λ ensures the generalization performance over a broad range. Empirically, the margin term is typically several orders of magnitude larger than the loss term. Therefore, a judicious choice for λ lies in the vicinity of 10^{-3}. Assigning an excessively large value for λ may lead the model to disregard the classification loss. While the primary focus of this paper is on traditional methods, our method can be seamlessly integrated into the realm of deep learning. We assess the enhancement of our method on softmax loss through visual classification tasks. Illustrating representative outcomes, the displayed training and test accuracy curves in Figure. 6 confirm the effectiveness of our method in mitigating overfitting. Comprehensive descriptions, settings and results are presented in Appendix A.5.

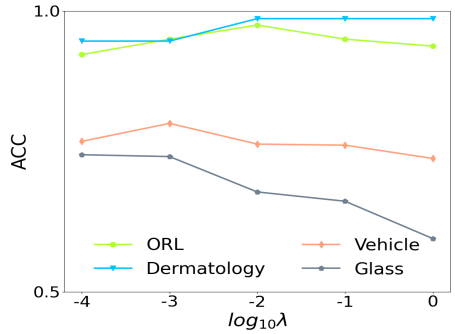
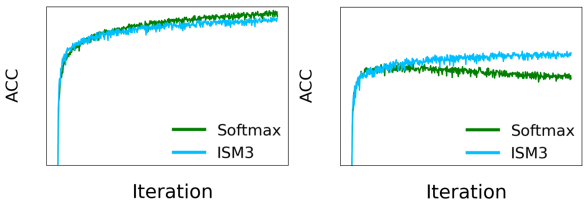


Figure 5: Study of λ.



(a) Train (b) Test

Figure 6: Accuracy curves with iterations on SVHN.

Conclusion

In this paper, we propose a concise but effective multi-class SVM model that enlarge the margin lower bound for all class pairs. We reveal the drawbacks of the related methods, while providing the motivations and detailed derivations of our method. Theoretical analysis confirms that the existing methods can be viewed as non-optimal special cases of our method. We show the proposed method can broadly improve the generalization performance from the SRM perspective. Our method can be integrated into neural networks, not only enhancing inter-class discrimination but also effectively mitigating overfitting. Both traditional and deep empirical evaluations validate the superiority of our method.

Acknowledgments

This work was supported by the National Science Foundation of China under Grant 62276212 and 62176212.

References

Amer, M.; Goldstein, M.; and Abdennadher, S. 2013. Enhancing one-class support vector machines for unsupervised anomaly detection. In *KDD*, 8–15.

Bartlett, P. L.; and Mendelson, S. 2002. Rademacher and Gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3: 463–482.

Böhning, D. 1992. Multinomial logistic regression algorithm. *Annal. Inst. Stat. Math.*, 44(1): 197–200.

Boser, B. E.; Guyon, I. M.; and Vapnik, V. N. 1992. A training algorithm for optimal margin classifiers. In *COLT*, 144–152.

Bredensteiner, E. J.; and Bennett, K. P. 1999. Multicategory classification by support vector machines. In *Comput Optim Appl*, 53–79.

Cortes, C.; and Vapnik, V. 1995. Support-vector networks. *Mach Learn*, 20(3): 273–297.

Crammer, K.; and Singer, Y. 2001. On the algorithmic implementation of multiclass kernel-based vector machines. *J. Mach. Learn. Res.*, 2: 265–292.

Crisp, D.; and Burges, C. J. 1999. A geometric interpretation of v-SVM classifiers. In *NeurIPS*, volume 12.

Doan, U.; Glasmachers, T.; and Igel, C. 2016. A unified view on multi-class support vector classification. *J. Mach. Learn. Res.*

G van den Burg, G.; and Groenen, P. 2016. GenSVM: A generalized multiclass support vector machine. *J. Mach. Learn. Res.*, 17: 1–42.

Grandvalet, Y.; Rakotomamonjy, A.; Keshet, J.; and Canu, S. 2008. Support vector machines with a reject option. In *NeurIPS*, volume 21.

Guermeur, Y. 2002. Combining discriminant models with new multi-class SVMs. *Pattern Analysis & Applications*, 5(2): 168–179.

Hsu, C.-W.; and Lin, C.-J. 2002. A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Networks Learn. Syst.*, 13(2): 415–425.

Khemchandani, R.; Chandra, S.; et al. 2007. Twin support vector machines for pattern classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(5): 905–910.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krishnapuram, B.; Carin, L.; Figueiredo, M. A.; and Hartemink, A. J. 2005. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(6): 957–968.

Ladicky, L.; and Torr, P. 2011. Locally linear support vector machines. In *ICML*, 985–992.

Lapin, M.; Hein, M.; and Schiele, B. 2015. Top-k multiclass SVM. In *NeurIPS*, volume 28.

Lauer, F.; et al. 2011. MSVMpack: a multi-class support vector machine package. *J. Mach. Learn. Res.*, 12: 2269–2272.

Mangasarian, O. L.; and Musicant, D. R. 2001. Lagrangian support vector machines. *J. Mach. Learn. Res.*, 1: 161–177.

Nie, F.; Huang, Y.; Wang, X.; and Huang, H. 2014. New primal SVM solver with linear computational cost for big data classifications. In *ICML*, volume 32, II–505.

Nie, F.; Wang, X.; and Huang, H. 2017. Multiclass capped lp-Norm SVM for robust classifications. In *AAAI*.

Nie, F.; Zhu, W.; and Li, X. 2020. Decision Tree SVM: An extension of linear SVM for non-linear classification. *Neurocomputing*, 401: 153–159.

Pisner, D. A.; and Schnyer, D. M. 2020. Support vector machine. In *Mach Learn*, 101–121. Elsevier.

Raman, P.; Srinivasan, S.; Matsushima, S.; Zhang, X.; Yun, H.; and Vishwanathan, S. 2019. Scaling multinomial logistic regression via hybrid parallelism. In *KDD*, 1460–1470.

Schiilkop, P.; Burgest, C.; and Vapnik, V. 1995. Extracting support data for a given task. In *KDD*, 252–257.

Tarzanagh, D. A.; Li, Y.; Thrampoulidis, C.; and Oymak, S. 2023. Transformers as support vector machines. *arXiv preprint arXiv:2308.16898*.

Vapnik, V. N. 1999. An overview of statistical learning theory. *IEEE Trans. Neural Networks Learn. Syst.*, 10(5): 988–999.

Vapnik, V. N.; and Chervonenkis, A. Y. 2015. On the uniform convergence of relative frequencies of events to their probabilities. In *Meas. of Complex.*, 11–30.

Vural, V.; and Dy, J. G. 2004. A hierarchical method for multi-class support vector machines. In *ICML*, 105.

Wei, Z.; and Hoai, M. 2016. Region ranking SVM for image classification. In *CVPR*, 2987–2996.

Weston, J.; and Watkins, C. 1998. Multi-class support vector machines. Technical report.

Xu, J.; Liu, X.; Huo, Z.; Deng, C.; Nie, F.; and Huang, H. 2017. Multi-class support vector machine via maximizing multi-class margins. In *IJCAI*.

Xu, L.; and Schuurmans, D. 2005. Unsupervised and semi-supervised multi-class support vector machines. In *AAAI*.

Zhang, J.; Jin, R.; Yang, Y.; and Hauptmann, A. G. 2003. Modified logistic regression: an approximation to SVM and its applications in large-scale text categorization. In *ICML*, 888–895.

Zhang, T.; and Zhou, Z.-H. 2019. Optimal margin distribution machine. *IEEE Trans. Knowl. Data Eng.*, 32(6): 1143–1156.

Zhang, W.; Hong, B.; Liu, W.; Ye, J.; Cai, D.; He, X.; and Wang, J. 2017. Scaling up sparse support vector machines by simultaneous feature and sample reduction. In *ICML*, 4016–4025.

Zhou, Y.; Kantarcioglu, M.; Thuraisingham, B.; and Xi, B. 2012. Adversarial support vector machine learning. In *KDD*, 1059–1067.