On the Optimization of Margin Distribution

Meng-Zhang Qian[1] , **Zheng Ai**[1] , **Teng Zhang**[2] and **Wei Gao**[1]
[1]National Key Laboratory for Novel Software Technology, Nanjing University, China
[2]School of Computer Science and Technology, Huazhong University of Science and Technology, China
{qianmz, aiz, gaow}@lamda.nju.edu.cn, tengzhang@hust.edu.cn

Abstract

Margin has played an important role on the design and analysis of learning algorithms during the past years, mostly working with the maximization of the minimum margin. Recent years have witnessed the increasing empirical studies on the optimization of margin distribution according to different statistics such as medium margin, average margin, margin variance, etc., whereas there is a relative paucity of theoretical understanding.

In this work, we take one step on this direction by providing a new generalization error bound, which is heavily relevant to margin distribution by incorporating ingredients such as average margin and semi-variance, a new margin statistics for the characterization of margin distribution. Inspired by the theoretical findings, we propose the MsvMAv, an efficient approach to achieve better performance by optimizing margin distribution in terms of its empirical average margin and semi-variance. We finally conduct extensive experiments to show the superiority of the proposed MsvMAv approach.

1 Introduction

Margin has played an important role on the design of learning algorithms from the pioneer work [Vapnik, 1982], which proposed the famous Support Vector Machines (SVMs) by maximizing the minimum margin, i.e. the smallest distance from the instances to the classification boundary. Boser *et al.* [1992] introduced the kernel technique for SVMs to relax the linear separation. Large margin has been one of the most important principles on the design of learning algorithms in the history of machine learning [Cortes and Vapnik, 1995; Schapire *et al.*, 1998; Rosset *et al.*, 2003; Shivaswamy and Jebara, 2010; Ji *et al.*, 2021], even for recent deep learning [Sokolić *et al.*, 2017; Weinstein *et al.*, 2020].

Various margin-based bounds have been presented to study the generalization performance of learning algorithms. Bartlett and Shawe-Taylor [1999] possibly presented the first generalization margin bounds based on VC dimension and fat-shattering dimension. Bartlett and Mendelson [2002] introduced the famous margin bounds based on Rademacher complexity, a data-dependent and finite-sample complexity measure. Kabán and Durrant [2020] took advantage of geometric structure to provide margin bounds for compressive learning. Grønlund *et al.* [2020] presented the near-tight margin generalization bound for SVMs. Margin has also been an ingredient to analyze the generalization performance for other algorithms such as boosting [Schapire *et al.*, 1998; Breiman, 1999; Gao and Zhou, 2013], and deep learning [Bartlett *et al.*, 2017; Wei and Ma, 2020].

Margin distribution has been considered as an important ingredient on the design and analysis of learning algorithms, and the basic idea is to optimize some margin statistics, relevant to the whole margin distribution rather than single margin. Garg and Roth [2003] introduced the model complexity measure to optimize margin distribution. Pelckmans *et al.* [2007] optimized margin distribution via average margin, while Aiolli *et al.* [2008] tried to maximize the minimum margin and average margin. Zhang and Zhou [2014] proposed the large margin distribution machine by considering average margin and margin variance simultaneously, which motivates the design of a series learning algorithms on the optimization of margin distribution [Cheng *et al.*, 2016; Rastogi *et al.*, 2020]. For deep learning, Jiang *et al.* [2019] introduced some margin distribution statistics, such as total variation, median quartile, etc., to analyze the generalization of neural networks. There is a relative paucity of theoretical understanding on how to correlate margin distribution with the generalization of learning algorithms.

This work tries to fill the gap between theoretical and empirical studies on the optimization of margin distribution, and the main contributions can be summarized as follows:

- We present a new generalization error bound, which is heavily relevant to margin distribution by incorporating factors such as average margin and semi-variance. Here, semi-variance is a new statistics, counting the average of squared distances between average margin and the instances' margin, that is smaller than average margin.
- Motivated from our theoretical result, we develop the MsvMAv approach, which tries to achieve better generalization performance by optimizing margin distribution in terms of empirical average margin and semi-variance. We find the closed-form solution in optimization, and improve its efficiency via Sherman-Morrison formula.

- We conduct extensive empirical studies to validate the effectiveness of the MsvMAv approach in comparisons with the state-of-the-art algorithms on large-margin or margin distribution optimization.

The rest of this paper is organized as follows. Section 2 introduces some preliminaries. Section 3 presents theoretical analysis. Section 4 proposes the MsvMAv approach. Section 5 conducts extensive empirical studies, and Section 6 concludes with future work.

2 Preliminaries

Let $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} = \{+1, -1\}$ denote the instance and label space, respectively. Suppose that $\mathcal{D}$ is an underlying (unknown) distribution over the product space $\mathcal{X} \times \mathcal{Y}$. Let

$$S_n = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_n, y_n)\}$$

be a training sample with each element drawn independently and identically (i.i.d.) from distribution $\mathcal{D}$. We use $\Pr_{\mathcal{D}}[\cdot]$ and $E_{\mathcal{D}}[\cdot]$ to refer to the probability and expectation according to distribution $\mathcal{D}$, respectively.

Let $\mathcal{H} = \{h\colon \mathcal{X} \to [-1, +1]\}$ be a function space. We define the classification error (or generalization risk) with respect to function $h \in \mathcal{H}$ and distribution $\mathcal{D}$, as

$$\mathcal{E}(h) = \Pr_{\mathcal{D}}[\operatorname{sgn}[h(\boldsymbol{x})] \neq y] = E_{\mathcal{D}}[\mathbb{I}[yh(\boldsymbol{x}) \leq 0]] \ ,$$

where the sign function sgn$[\cdot]$ returns $+1$, 0 and -1 if the argument is positive, zero and negative, respectively, and the indicator function $\mathbb{I}[\cdot]$ returns 1 when the argument is true, and 0 otherwise.

Given an example $(\boldsymbol{x}, y)$, the *margin* of $h \in \mathcal{H}$ is defined as $yh(\boldsymbol{x})$, which can be viewed as a measure of the confidence of the classification. We further define the average margin of $h \in \mathcal{H}$ over distribution $\mathcal{D}$ as

$$\theta_h = E_{(\boldsymbol{x},y)\sim\mathcal{D}}[yh(\boldsymbol{x})] \ . \tag{1}$$

We also introduce the empirical Rademacher complexity [Bartlett and Mendelson, 2002] to measure the complexity of function space $\mathcal{H}$ as follows:

$$\widehat{\mathfrak{R}}_{S_n}(\mathcal{H}) = E_{\sigma_1,\sigma_2,\ldots,\sigma_n}\left[\sup_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i h(\boldsymbol{x}_i)\right] ,$$

where each σ_i is a Rademacher variable with $\Pr[\sigma_i = +1] = \Pr[\sigma_i = -1] = 1/2$ for $i \in [n]$.

We finally introduce some notations used in this work. Write $[d] = \{1, 2, \ldots, d\}$ for integer $d > 0$, and $\langle \boldsymbol{w}, \boldsymbol{x}\rangle$ represents the inner product of $\boldsymbol{w}$ and $\boldsymbol{x}$. Let $\boldsymbol{I}_d$ be the identity matrix of size $d \times d$, and denote by $^\top$ the transpose of vectors or matrices. For positive $f(n)$ and $g(n)$, we write $f(n) = O(g(n))$ if $g(n)/f(n) \to c$ for constant $c < +\infty$.

3 Theoretical Analysis

We begin with the squared margin loss as follows:

Definition 1. *For $\theta > 0$, we define the squared margin loss ℓ_θ with respect to function $h \in \mathcal{H}$ as*

$$\ell_\theta\big(h, (\boldsymbol{x}, y)\big) = \left[(1 - yh(\boldsymbol{x})/\theta)_+\right]^2 ,$$

where $(a)_+ = \max(0, a)$.

This is a simple extension from the traditional margin loss [Bartlett and Mendelson, 2002], while we consider the squared loss and unbounded constraint for the negative $yh(\boldsymbol{x})$. The margin parameter θ is generally irrelevant to learned function h and data distribution in most previous theoretical and algorithmic studies.

In this work, we select margin parameter θ as the average margin when $\theta_h > 0$, to correlate generalization performance with margin distribution, that is,

$$\theta = \theta_h = E_{\mathcal{D}}[yh(\boldsymbol{x})] \ ,$$

which is dependent on data distribution and learned function. Given training sample S_n, we try to learn a function h by minimizing the squared margin loss as follows:

$$\min_{h\in\mathcal{H}\colon\, \theta_h>0} \left\{ \frac{1}{n}\sum_{i=1}^{n}\left[\left(1 - \frac{y_i h(\boldsymbol{x}_i)}{\theta_h}\right)_+\right]^2 \right\} . \tag{2}$$

For simplicity, we further introduce the notion of *margin semi-variance* [Markowitz, 1952] as follows:

Definition 2. *Given function $h \in \mathcal{H}$ and training sample S_n, we define the margin semi-variance as*

$$\mathit{SV}(h) = \frac{1}{n}\sum_{i=1}^{n}\left[(\theta_h - y_i h(\boldsymbol{x}_i))_+\right]^2 ,$$

where θ_h denotes the average margin defined by Eqn. (1).

The margin semi-variance essentially counts the average of squared deviation between average margin and the margins $y_i h(\boldsymbol{x}_i)$, which are smaller than average margin. This yields an equivalent expression for Eqn. (2) as follows:

$$\min_{h\in\mathcal{H}:\theta_h\geq\nu>0} \left\{\mathrm{SV}(h)/\theta_h^2\right\} ,$$

that is, optimizing the squared margin loss with parameter $\theta = \theta_h$ is equivalent to minimizing margin semi-variance and maximizing average margin simultaneously.

For most real applications, we could learn some relatively-good functions from sufficient training data. Motivated from the notion of weak learner in boosting [Freund and Schapire, 1996], we formally define the *set of relatively-good functions* for function space $\mathcal{H}$ as follows:

$$\mathcal{H}_\nu = \{h \in \mathcal{H}, \theta_h \geq \nu\} \ \text{ for some small constant } \ \nu > 0 \ .$$

Essentially, a relatively-good function is similar to a weak learner, which achieves slightly better performance than the randomly-guessed classifier.

We now present the main theoretical result as follows:

Theorem 1. *For small constant $\nu \geq 0$, let $\mathcal{H}$ be a function space with relative-good set $\mathcal{H}_\nu$. For any $\delta \in (0, 1)$ and for every $h \in \mathcal{H}_\nu$, the following holds with probability at least $1 - \delta$ over the training sample S_n*

$$\mathcal{E}(h) \leq \frac{\mathit{SV}(h)}{\theta_h^2} + O\left(\frac{\widehat{\mathfrak{R}}_n(\mathcal{H}_\nu)}{\theta_h^2} + \sqrt{\frac{1}{2n}\ln\frac{4n}{\delta}}\right) ,$$

with empirical Rademacher complexity $\widehat{\mathfrak{R}}_n(\mathcal{H}_\nu) \leq \widehat{\mathfrak{R}}_n(\mathcal{H})$.

This theorem presents a new generalization error bound, which is heavily relevant to margin distribution by incorporating factors such as average margin and semi-variance. This could shed some new insights on the design of algorithms on the optimization of margin distribution as shown in Section 4.

The proof follows the empirical Rademacher complexity [Bartlett and Mendelson, 2002], while the challenge lies in the distribution-dependent average margin θ_h. We solve it by constructing a sequence of intervals for average margin θ_h, and the detailed proof is presented in [Qian *et al.*, 2022].

It remains to study the empirical Rademacher complexity in Theorem 1, and we focus on linear and kernel functions. For instance space $\mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^d \colon \|\boldsymbol{x}\| \leq r\}$ and linear function space $\mathcal{H} = \{h(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} \colon \|\boldsymbol{w}\| \leq \Lambda\}$, let $\mathcal{H}_\nu$ denote the set of relatively-good classifiers. We upper bound the empirical Rademacher complexity as

$$\widehat{\mathfrak{R}}_{S_n}(\mathcal{H}_\nu) \leq \widehat{\mathfrak{R}}_{S_n}(\mathcal{H}) \leq r\Lambda/\sqrt{n}$$

from the work of [Shalev-Shwartz and Ben-David, 2014]. For kernel function $\kappa(\cdot,\cdot)$, we have the kernel function space

$$\mathcal{H} = \Big\{h(\boldsymbol{x}) = \sum_{i=1}^n a_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) \colon \sum_{i,j=1}^n a_i a_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq \Lambda^2\Big\}.$$

We could upper bound the empirical Rademacher complexity for kernel functions, from [Bartlett and Mendelson, 2002],

$$\widehat{\mathfrak{R}}_{S_n}(\mathcal{H}_\nu) \leq \widehat{\mathfrak{R}}_{S_n}(\mathcal{H}) \leq \frac{2\Lambda}{n}\Big(\sum_{i=1}^n \kappa(\boldsymbol{x}_i, \boldsymbol{x}_i)\Big)^{1/2} .$$

It is also noteworthy that the average margin θ_h is unknown on the design of new algorithms because of the unknown distribution $\mathcal{D}$, and we resort to the empirical average margin from training sample S_n in practice.

4 The MsvMAv Approach

Motivated from Theorem 1, this section develops the MsvMAv approach on the optimization of margin distribution, and we focus on linear and kernel functions.

4.1 Linear Functions

For linear space $\mathcal{H} = \{h_{\boldsymbol{w}}(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x}\rangle \colon \|\boldsymbol{w}\| = 1\}$ and training sample S_n, we have the empirical average margin

$$\hat{\theta}_{\boldsymbol{w}} = \frac{1}{n}\sum_{i=1}^n y_i \langle \boldsymbol{w}, \boldsymbol{x}_i\rangle .$$

For simplicity, we omit a bias term on the design of algorithm, and we will augment $\boldsymbol{w}$ and instance $\boldsymbol{x}$ with bias term b and 1 in experiments, respectively, as shown in Section 5. Our optimization problem can be formally written as

$$\min_{\|\boldsymbol{w}\|_2=1} \Big\{\frac{\widehat{\mathrm{SV}}(\boldsymbol{w})}{\hat{\theta}_{\boldsymbol{w}}}\Big\} ,$$

where empirical average margin $\hat{\theta}_{\boldsymbol{w}} > 0$ and empirical margin semi-variance $\widehat{\mathrm{SV}}(\boldsymbol{w}) = \sum_{i=1}^n [(\hat{\theta}_{\boldsymbol{w}} - y_i\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle)_+]^2/n$. Obviously, this is a non-convex optimization problem, and we would optimize the empirical margin semi-variance and average margin alternatively.

Initialize the linear function $\boldsymbol{w}_0$ by optimizing empirical average margin, that is,

$$\boldsymbol{w}_0 = \mathop{\arg\max}_{\|\boldsymbol{w}\|_2^2=1} \sum_{i=1}^n \frac{y_i\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle}{n} = \sum_{i=1}^n \frac{y_i \boldsymbol{x}_i}{\|\sum_{i=1}^n y_i \boldsymbol{x}_i\|_2} , \tag{3}$$

where we solve $\boldsymbol{w}_0$ from its dual problem using Lagrangian function, and the details are given by Qian *et al.* [2022].

Optimization of Empirical Margin Semi-variance

In the k-th iteration ($k \geq 1$) with previous classifier $\boldsymbol{w}_{k-1}$, we first calculate the empirical average margin $\hat{\theta}_{\boldsymbol{w}_{k-1}}$ as

$$\hat{\theta}_{\boldsymbol{w}_{k-1}} = \frac{1}{n}\sum_{i=1}^n y_i\langle \boldsymbol{w}_{k-1}, \boldsymbol{x}_i\rangle . \tag{4}$$

We then introduce the minimization of empirical margin semi-variance as follows:

$$\min_{\boldsymbol{w}} \Big\{ \sum_{i=1}^n \frac{[(\hat{\theta}_{\boldsymbol{w}_{k-1}} - y_i\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle)_+]^2}{n} + \beta_k \|\boldsymbol{w} - \boldsymbol{w}_{k-1}\|_2^2 \Big\},$$

where β_k is a proximal regularization parameter. We now introduce the following index set, to present a closed-form solution for the above minimization,

$$\mathcal{A}_k = \big\{i \colon y_i\langle \boldsymbol{w}_{k-1}, \boldsymbol{x}_i\rangle < \hat{\theta}_{\boldsymbol{w}_{k-1}} \quad \text{for} \quad i \in [n]\big\} , \tag{5}$$

i.e., the index set of instance with margins below the empirical average margin $\hat{\theta}_{\boldsymbol{w}_{k-1}}$. We can rewrite the minimization of empirical margin semi-variance as

$$\min_{\boldsymbol{w}} \Big\{ \sum_{i\in\mathcal{A}_k} \frac{(\hat{\theta}_{\boldsymbol{w}_{k-1}} - y_i\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle)^2}{n} + \beta_k \|\boldsymbol{w} - \boldsymbol{w}_{k-1}\|_2^2 \Big\} .$$

Denote by $\boldsymbol{w}'_k$ the minimizer of the above problem, and we obtain the closed-form solution for $\boldsymbol{w}'_k$ as follows

$$\Big(\boldsymbol{I}_d + \sum_{i\in\mathcal{A}_k} \frac{\boldsymbol{x}_i\boldsymbol{x}_i^\top}{n\beta_k}\Big)^{-1}\Big(\frac{\hat{\theta}_{\boldsymbol{w}_{k-1}}}{n\beta_k}\sum_{i\in\mathcal{A}_k} y_i\boldsymbol{x}_i + \boldsymbol{w}_{k-1}\Big). \tag{6}$$

One problem is to calculate the inverse in Eqn. (6), which takes $O(d^3)$ computational costs (d is dimensionality). This remains one challenge to deal with high-dimensional tasks.

We now present an efficient method to calculate of the inverse in Eqn. (6). For simplicity, we denote by

$$\boldsymbol{M}_k = \Big(\boldsymbol{I}_d + \sum\nolimits_{i\in\mathcal{A}_k} \boldsymbol{x}_i\boldsymbol{x}_i^\top/n\beta_k\Big)^{-1} \quad \text{for} \quad k = 1, 2, \cdots ,$$

and it is easy to derive the following recursive relation:

$$\boldsymbol{M}_k^{-1} = \boldsymbol{M}_{k-1}^{-1} - \sum_{i\in\mathcal{A}_{k-1}\setminus\mathcal{A}_k} \frac{\boldsymbol{x}_i\boldsymbol{x}_i^\top}{n\beta} + \sum_{i\in\mathcal{A}_k\setminus\mathcal{A}_{k-1}} \frac{\boldsymbol{x}_i\boldsymbol{x}_i^\top}{n\beta} ,$$

with $\boldsymbol{M}_0 = \boldsymbol{I}_d$.

We calculate $\boldsymbol{M}_k$ efficiently from $\boldsymbol{M}_{k-1}$ and Sherman-Morrison formula [Sherman and Morrison, 1950]. In other

Algorithm 1 The MsvMAv Approach

```
Input: Training sample $S_n$, iteration number $T$, and proximal parameters $\alpha_k$ and $\beta_k$
Output: $\boldsymbol{w}$
Initialize $\boldsymbol{M}_0 = \boldsymbol{I}_d$, $A_0 = \emptyset$ and $\boldsymbol{w}_0$ by Eqn. (3)
for $k = 1, 2, \cdots, T$ do
    Compute empirical average margin $\hat{\theta}_{\boldsymbol{w}_{k-1}}$ by Eqn. (4)
    Solve the index set $\mathcal{A}_k$ by Eqn. (5)
    Compute $\boldsymbol{M}_k = \left(\boldsymbol{I}_d + \sum_{i\in\mathcal{A}_k} \boldsymbol{x}_i\boldsymbol{x}_i^\top / n\beta_k\right)^{-1}$ by Eqns. (7) and (8)
    Compute the minimizer $\boldsymbol{w}_k'$ for empirical margin semi-variance by Eqn. (9)
    Solve the empirical average margin maximizer $\boldsymbol{w}_k$ by Eqn. (10), and normalize $\boldsymbol{w}_k = \boldsymbol{w}_k / \|\boldsymbol{w}_k\|_2$
end for
return $\boldsymbol{w} = \boldsymbol{w}_T$
```

words, we initialize $\boldsymbol{M}' = \boldsymbol{M}_{k-1}$, and make the following updates iteratively, based on Sherman-Morrison formula,

$$\boldsymbol{M}' = \boldsymbol{M}' - \frac{\boldsymbol{M}'\boldsymbol{x}_i\boldsymbol{x}_i^\top\boldsymbol{M}'}{\boldsymbol{x}_i^\top\boldsymbol{M}'\boldsymbol{x}_i - n\beta_k} \quad \text{for} \quad i \in \mathcal{A}_{k-1} \setminus \mathcal{A}_k\ , \tag{7}$$

$$\boldsymbol{M}' = \boldsymbol{M}' - \frac{\boldsymbol{M}'\boldsymbol{x}_i\boldsymbol{x}_i^\top\boldsymbol{M}'}{\boldsymbol{x}_i^\top\boldsymbol{M}'\boldsymbol{x}_i + n\beta_k} \quad \text{for} \quad i \in \mathcal{A}_k \setminus \mathcal{A}_{k-1}\ . \tag{8}$$

We then obtain $\boldsymbol{M}_k = \boldsymbol{M}'$, and the minimizer of empirical margin semi-variance is given by

$$\boldsymbol{w}_k' = \boldsymbol{M}'\Big(\frac{\hat{\theta}_{\boldsymbol{w}_{k-1}}}{n\beta_k}\sum_{i\in\mathcal{A}_k} y_i\boldsymbol{x}_i + \boldsymbol{w}_{k-1}\Big). \tag{9}$$

Optimization of Empirical Average Margin

We now study the maximization of empirical average margin, which can be formalized as:

$$\boldsymbol{w}_k = \arg\min_{\boldsymbol{w}} \left\{ -\frac{1}{n}\sum_{i=1}^{n} y_i\langle\boldsymbol{w},\boldsymbol{x}_i\rangle + \alpha_k \left\|\boldsymbol{w} - \boldsymbol{w}_k'\right\|_2^2 \right\},$$

where α_k is a proximal regularization parameter. It is easy to obtain the closed-form solution as follows:

$$\boldsymbol{w}_k = \boldsymbol{w}_k' + \frac{1}{2\alpha_k n}\sum_{i=1}^{n} y_i\boldsymbol{x}_i\ . \tag{10}$$

We obtain $\boldsymbol{w}_k = \boldsymbol{w}_k / \|\boldsymbol{w}_k\|$ in the k-th iteration. Algorithm 1 presents a detailed description of our MsvMAv approach.

4.2 Kernelization

This section focuses on kernel mapping $\boldsymbol{\phi} : \mathcal{X} \to \mathbb{H}$ for Hilbert space $\mathbb{H}$, we consider $h(\boldsymbol{x}) = \langle\boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x})\rangle$ with $\boldsymbol{w} \in \mathbb{H}$ and $\boldsymbol{\phi}(\boldsymbol{x}) \in \mathbb{H}$. The optimization problem is given by

$$\min_{\boldsymbol{w}} \left\{ \frac{\widehat{\mathrm{SV}}(\boldsymbol{w})}{\hat{\theta}_{\boldsymbol{w}}} \right\},$$

where average margin $\hat{\theta}_{\boldsymbol{w}} = \sum_{i=1}^{n} y_i\langle\boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}_i)\rangle / n > 0$, and margin semi-variance

$$\widehat{\mathrm{SV}}(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n}\left[\Big(\hat{\theta}_{\boldsymbol{w}} - y_i\langle\boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}_i)\rangle\Big)_+\right]^2 .$$

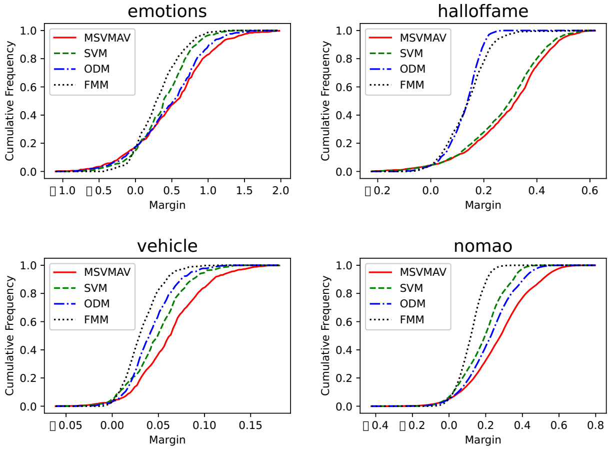

Figure 1: Cumulative frequency versus margin of our MsvMAv and other algorithms such as SVM, ODM and FMM. The more right the curve, the better the margin distribution.

It is intractable to solve such optimization problem directly because of high or even infinity dimensionality. According to Representer theorem [Schölkopf *et al.*, 2002], we first have $\boldsymbol{w}^* = \sum_{i=1}^{n} a_i\boldsymbol{\phi}(\boldsymbol{x}_i)$, spanned by $\{\boldsymbol{\phi}(\boldsymbol{x}_i), i \in [n]\}$ with coefficients $a_1, \cdots, a_n$. This follows the prediction

$$h(\boldsymbol{x}) = \langle\boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x})\rangle = \sum_{i=1}^{n} a_i\kappa(\boldsymbol{x}, \boldsymbol{x}_i)\ ,$$

where $\kappa(\cdot,\cdot)$ denotes the kernel function. For simplicity, denote by $\boldsymbol{a} = (a_1, a_2, \cdots, a_n)^\top$, and write the gram matrix of instances in S_n as

$$\boldsymbol{K} = (\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_n) = \big(K_{ij}\big)_{n\times n} = \big(\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)\big)_{n\times n},$$

where $\boldsymbol{K}_i$ denotes the k-th column of matrix $\boldsymbol{K}$. Hence, our optimization problem can be further rewritten as

$$\min_{\boldsymbol{a}} \left\{ \frac{1}{n}\sum_{i=1}^{n}\left[\left(1 - \frac{y_i\langle\boldsymbol{K}_i, \boldsymbol{a}\rangle}{\hat{\theta}_{\boldsymbol{a}}}\right)_+\right]^2 \right\} = \min_{\boldsymbol{a}} \left\{ \frac{\widehat{\mathrm{SV}}(\boldsymbol{a})}{\hat{\theta}_{\boldsymbol{a}}} \right\},$$

where the empirical average margin $\hat{\theta}_{\boldsymbol{a}} = \sum_{i=1}^{n} y_i\langle\boldsymbol{K}_i, \boldsymbol{a}\rangle / n$ and semi-variance $\widehat{\mathrm{SV}}(\boldsymbol{a}) = \sum_{i=1}^{n}[(\hat{\theta}_{\boldsymbol{a}} - y_i\langle\boldsymbol{K}_i, \boldsymbol{a}\rangle)_+]^2 / n$.

We first initialize the classifier $\boldsymbol{a}_0$ by maximizing the empirical average margin as follows:

$$\boldsymbol{a}_0 = \arg\max_{\boldsymbol{a}^\top\boldsymbol{K}\boldsymbol{a}=1} \left\{ \frac{1}{n}\sum_{i=1}^{n} y_i\langle\boldsymbol{K}_i, \boldsymbol{a}\rangle \right\}.$$

In the k-th iteration with previous classifier $\boldsymbol{a}_{k-1}$, we minimize the margin semi-variance based on previous average margin $\hat{\theta}_{\boldsymbol{a}_{k-1}}$. We write

$$\left\|\boldsymbol{a} - \boldsymbol{a}_{k-1}\right\|_{\boldsymbol{I}_n + \boldsymbol{K}} = \Big((\boldsymbol{a} - \boldsymbol{a}_{k-1})^\top(\boldsymbol{I}_n + \boldsymbol{K})\big(\boldsymbol{a} - \boldsymbol{a}_{k-1}\big)\Big)^{1/2},$$

and the optimization problem can be given by

$$\min_{\boldsymbol{a}} \left\{ \sum_{i=1}^{n} \frac{[(\hat{\theta}_{\boldsymbol{a}_{k-1}} - y_i\langle\boldsymbol{K}_i, \boldsymbol{a}\rangle)_+]^2}{n} + \beta_k\|\boldsymbol{a} - \boldsymbol{a}_{k-1}\|_{\boldsymbol{I}_n + \boldsymbol{K}}^2 \right\},$$

Dataset	MsvMAv	SVM	SVR	LSSVM	ODM	MAMC	FMM
advertise	.9837±.0015	.9823±.0021●	.9825±.0024●	.9819±.0019●	.9701±.0021●	.9132±.0007●	.9799±.0025●
australian	.8314±.0079	.8167±.0077●	.8171±.0048●	.8220±.0036●	.8104±.0065●	.7700±.0201●	.8295±.0100
bibtex	.7417±.0040	.7378±.0069●	.7469±.0060○	.7478±.0046○	.7469±.0053○	.6689±.0123●	.7371±.0062●
biodeg	.8712±.0087	.8741±.0068	.8490±.0107●	.8741±.0068	.8681±.0075	.6872±.0000●	.8613±.0089●
breastw	.9730±.0038	.9725±.0041	.9701±.0051●	.9684±.0051●	.8428±.0099●	.4088±.0000●	.9720±.0063
diabetes	.7530±.0088	.7561±.0104	.7478±.0077●	.7491±.0078●	.6110±.0146●	.5974±.0000●	.7496±.0082
emotions	.8216±.0074	.7983±.0130●	.7675±.0181●	.8036±.0127●	.8084±.0111●	.6975±.0000●	.7697±.0218●
german	.7518±.0097	.7457±.0095●	.7525±.0107	.7530±.0106	.7502±.0095	.6800±.0000●	.7525±.0102
halloffame	.9644±.0025	.9617±.0047●	.9593±.0041●	.9617±.0047●	.9620±.0028●	.9280±.0000●	.9598±.0036●
hill-valley	.7771±.0631	.5972±.0195●	.6999±.0572●	.5972±.0195●	.8898±.0079○	.5000±.0000●	.8526±.0296○
kc1	.8713±.0025	.8711±.0039	.8642±.0026●	.8711±.0039	.8690±.0035●	.8649±.0000●	.8701±.0028●
parkinsons	.9222±.0277	.8923±.0388●	.9342±.0315	.9444±.0338○	.8863±.0342●	.7949±.0000●	.8932±.0266●
pbcseq	.6626±.0074	.6439±.0147●	.6595±.0104	.6555±.0104●	.6562±.0125●	.6700±.0187○	.6549±.0117●
sleepdata	.6925±.0056	.6743±.0051●	.6833±.0064●	.6712±.0124●	.5691±.0156●	.5659±.0000●	.6833±.0047●
students	.8957±.0062	.8913±.0088●	.8870±.0064●	.8867±.0069●	.8893±.0046●	.5133±.0129●	.8898±.0040●
titanic	.7658±.0038	.7636±.0000●	.7636±.0000●	.7636±.0000●	.7509±.0211●	.6386±.0000●	.7636±.0000●
tokyo1	.9351±.0040	.9307±.0031●	.9337±.0027●	.9281±.0054●	.9248±.0053●	.7523±.0534●	.9363±.0037
vehicle	.9708±.0057	.9422±.0473●	.9746±.0031○	.9748±.0034○	.9720±.0061	.7041±.0000●	.9718±.0083
vertebra	.8194±.0197	.7978±.0149●	.7731±.0144●	.7753±.0131●	.7957±.0105●	.7581±.0000●	.7763±.0100●
wdbc	.9778±.0067	.9787±.0084	.9725±.0030●	.9696±.0044●	.9237±.0094●	.5398±.1287●	.9655±.0064●
a9a	.8433±.0009	.8417±.0007●	.8403±.0007●	.8358±.0006●	.8430±.0012	.7577±.0000●	.8390±.0012●
acoustic	.7494±.0011	.7321±.0037●	.7394±.0004●	N/A●	.7406±.0023●	.7206±.0055●	.7402±.0005●
bank	.9057±.0004	.8960±.0008●	.9015±.0004●	N/A●	.9021±.0006●	.8854±.0000●	.9021±.0005●
eurgbp	.5332±.0027	.5042±.0061●	.5317±.0028●	N/A●	.5111±.0084●	.4985±.0000●	.5288±.0028●
jm1	.8125±.0015	.8132±.0012	.8119±.0016	.8123±.0015	.8065±.0000●	.8065±.0000●	.8076±.0010●
magic	.7998±.0005	.7976±.0012●	.7928±.0008●	.7912±.0009●	.7943±.0007●	.6514±.0000●	.7987±.0014●
nomao	.9453±.0005	.9421±.0061●	.9439±.0005●	N/A●	.9452±.0004	.7062±.0000●	.9442±.0008●
phishing	.9388±.0011	.9405±.0017○	.9371±.0007●	.9343±.0008●	.9386±.0014	.5532±.0008●	.9318±.0009●
pol	.9054±.0013	.8746±.0398●	.9002±.0019●	.9041±.0018●	.6788±.0037●	.6740±.0000●	.8866±.0016●
run-walk	.7260±.0007	.7169±.0000●	.7077±.0004●	N/A●	.7104±.0061●	.5431±.0757●	.7261±.0054
Win/Tie/Loss		23/6/1	24/4/2	23/4/3	22/6/2	29/0/1	22/7/1

Table 1: Comparisons of the test accuracies (mean±std.) on 30 datasets. ●/○ indicates that our MsvMAv approach is significantly better/worse than the corresponding algorithms (pairwise t-tests at 95% significance level). 'N/A' indicates that LSSVM does not return results on the data set within 12 hours.

where β_k is a proximal regularization parameter. We introduce the index set $\mathcal{A}_k = \{i \colon y_i \langle \boldsymbol{K}_i, \boldsymbol{a} \rangle < \hat{\theta}_{\boldsymbol{a}_{k-1}} \text{ for } i \in [n]\}$, and obtain the empirical margin semi-variance minimizer

$$\boldsymbol{a}'_k = \boldsymbol{M}_k \Big((\boldsymbol{K} + \boldsymbol{I}_n) \boldsymbol{a}_{k-1} + \hat{\theta}_{\boldsymbol{a}'_k} \sum_{i \in \mathcal{A}_k} \frac{y_i \boldsymbol{K}_i}{n\beta} \Big)$$

where we use the Sherman-Morrison formula to calculate

$$\boldsymbol{M}_k = \left(\sum_{i \in \mathcal{A}_k} \frac{\boldsymbol{K}_i \boldsymbol{K}_i^\top}{n\beta} + \boldsymbol{K} + \boldsymbol{I}_n \right)^{-1} .$$

We finally maximize the empirical average margin based on the following optimization problem:

$$\min_{\boldsymbol{a_k}} \Big\{ -\frac{1}{n} \sum_{i=1}^{n} y_i \langle \boldsymbol{K}_i, \boldsymbol{a}_k \rangle + \alpha_k (\boldsymbol{a}_k - \boldsymbol{a}'_k)^\top \boldsymbol{K} (\boldsymbol{a}_k - \boldsymbol{a}'_k) \Big\},$$

where α_k is a proximal regularization parameter, and it is easy to get the closed-form solution as follows:

$$\boldsymbol{a}_k = \boldsymbol{a}'_k + [y_1, y_2, \cdots, y_n]^\top / 2\alpha_k n \, .$$

We get the final $\boldsymbol{a}_k = \boldsymbol{a}_k / \|\boldsymbol{a}_k\|_{\boldsymbol{K}}$ in the k-th iteration.

5 Empirical Study

In this section, we present extensive empirical studies to verify the effectiveness of our proposed MsvMAv approach. We consider 30 datasets, including 20 regular and 10 large-scale datasets. The number of instances varies from 208 to 88588 while the feature dimensionality ranges from 2 to 1836, covering a wide range of properties. The statistics for all datasets can be found in [Qian *et al.*, 2022].

We compare our proposed MsvMAv approach with state-of-the-art algorithms on large-margin and margin distribution optimization: 1) SVM [Boser *et al.*, 1992], 2) SVR [Drucker *et al.*, 1997] with binary targets, 3) LSSVM [Suykens *et al.*, 2002], 4) MAMC [Pelckmans *et al.*, 2007], 5) ODM [Zhang and Zhou, 2019], 6) FMM [Ji *et al.*, 2021]. The details of compared algorithms can be found in [Qian *et al.*, 2022].

For each dataset, we scale all features into the interval $[0, 1]$, and augment each instance $\boldsymbol{x}$ with constant 1 for the bias of linear model. The empirical average margin $\theta_{\boldsymbol{w}}$ may be smaller than zero in experiments, when the proximal regularization parameter β_k is set too small. In such case, we take the opposite model $-\boldsymbol{w}$ so as to keep the positiveness of empirical average margin.

For our MsvMAv approach, parameters α_k and β_k are set

Dataset	MsvMAv	SVM	SVR	LSSVM	ODM	MAMC	FMM
advertise	.9837±.0014	.9820±.0023●	.9835±.0019	.9848±.0016○	.9838±.0016	.9547±.0026●	.9810±.0028●
australian	.8580±.0078	.8225±.0087●	.8210±.0080●	.8357±.0111●	.8329±.0088●	.8302±.0122●	.8237±.0060●
bibtex	.7554±.0036	.7506±.0050●	.7508±.0053●	.7505±.0056●	.7570±.0045	.6714±.0397●	.7509±.0050●
biodeg	.8834±.0081	.8687±.0115●	.8580±.0089●	.8712±.0093●	.8845±.0095	.8559±.0098●	.8915±.0096○
breastw	.9783±.0013	.9710±.0013●	.9703±.0050●	.9713±.0018●	.9710±.0013●	.9774±.0022●	.9710±.0013●
diabetes	.7576±.0074	.7574±.0094	.7502±.0096●	.7411±.0112●	.7504±.0082●	.6779±.0197●	.7385±.0105●
emotions	.7986±.0122	.7899±.0156●	.7756±.0164●	.8115±.0124○	.8101±.0147○	.7952±.0120	.8078±.0126○
german	.7465±.0099	.7443±.0096	.7198±.0174●	.7353±.0143●	.7285±.0156●	.7198±.0154●	.7277±.0127●
halloffame	.9677±.0025	.9625±.0020●	.9578±.0053●	.9523±.0060●	.9617±.0025●	.9510±.0028●	.9590±.0034●
hill-valley	.6826±.0184	.6668±.0177●	.6482±.0135●	.7116±.0678○	.5950±.0360●	.5402±.0320●	.7886±.0299○
kc1	.8739±.0042	.8701±.0057●	.8689±.0045●	.8703±.0044●	.8716±.0055	.8764±.0027○	.8706±.0038●
parkinsons	.9573±.0223	.9282±.0203●	.9393±.0193●	.9393±.0224●	.9385±.0157●	.9214±.0174●	.9402±.0167●
pbcseq	.7350±.0143	.7214±.0152●	.7317±.0151	.7312±.0165	.7269±.0221●	.7076±.0149●	.7238±.0184●
sleepdata	.7407±.0129	.7192±.0105●	.7211±.0061●	.7037±.0083●	.7050±.0083●	.7055±.0056●	.7182±.0094●
students	.8977±.0119	.8920±.0079●	.8665±.0098●	.8898±.0072●	.8805±.0142●	.6543±.0185●	.8993±.0062
titanic	.7825±.0048	.7823±.0052	.7823±.0052	.7823±.0052	.7767±.0075●	.7823±.0052	.7823±.0052
tokyo1	.9406±.0037	.9241±.0050●	.9257±.0060●	.9337±.0054●	.9253±.0053●	.9229±.0039●	.9248±.0050●
vehicle	.9793±.0083	.9795±.0090	.9856±.0073○	.9899±.0070○	.9722±.0088●	.9625±.0105●	.9805±.0093
vertebra	.8280±.0240	.7898±.0098●	.7957±.0183●	.8108±.0176●	.8000±.0122●	.7769±.0126●	.7962±.0153●
wdbc	.9819±.0081	.9810±.0056	.9772±.0066●	.9842±.0035	.9795±.0052	.9526±.0089●	.9526±.0089●
Win/Tie/Loss		15/5/0	16/3/1	13/3/4	14/5/1	17/2/1	14/3/3

Table 2: Comparisons of the test accuracies (mean±std.) on 20 datasets. We use Gaussian kernel for all algorithms. ●/○ indicates that our MsvMAv approach is significantly better/worse than the corresponding algorithms (pairwise t-tests at 95% significance level).

to be constant and selected by 5-fold cross validation from $\{2^{-10}, 2^{-8}, \cdots, 2^{10}\}$, and the width of Gaussian kernel is chosen from $\{2^{-10}/d, 2^{-8}/d, \cdots, 2^{10}/d\}$. We select the maximum iteration number $T = 100$ as a stopping criteria for MsvMAv . For SVM, SVR, LSSVM and ODM, we set regularization parameter $C \in \{2^{-10}, 2^{-8}, \cdots, 2^{10}\}$ by 5-fold cross validation again, and the others are set according to their respective references, also shown in [Qian *et al.*, 2022].

We first compare the margin distributions of our proposed MsvMAv approach with other algorithms. Figure 1 illustrates the cumulative margin distributions of different algorithms on four datasets, and similar trends can be observed on other datasets. As can be seen, the curves of our MsvMAv approach generally lie on the rightmost side, which shows the margin distributions of MsvMAv are generally better than that of SVM, ODM and FMM.

We further analyze the generalization performance of our proposed MsvMAv approach with other compared algorithms. All algorithms are evaluated by 30 times of random partitions of datasets with 80% and 20% of data for training and testing, respectively. The test accuracies are obtained by averaging over 30 times. Tables 1 and 2 show the empirical results of our MsvMAv and other algorithms with linear and Gaussian kernel functions, respectively.

From Tables 1 and 2, our proposed MsvMAv approach takes significantly better performance than other algorithms for linear and kernel functions, since win/tie/loss counts show that our approach wins for most datasets, and rarely losses. One intuitive explanation is that our MsvMAv approach achieves better margin distribution by maximizing the empirical average margin and minimizing empirical margin semi-variance, as shown in Figure 1. SVM, SVR and FMM maximize the minimum margin, which ignores the margin distribution. LSSVM and MAMC essentially maximize average margin only, which fails to learn from other margin statistics. ODM takes the average margin and margin variance into consideration, but the process of margin variance minimization could constrain some large margins.

This section omits partial empirical results due to the page limit, including the empirical curves of margin distributions, as well as running time comparisons for our MsvMAv and other compared algorithms. Relevant results can be found in our full work [Qian *et al.*, 2022].

6 Conclusion

Large margin has been one of the most important principles on the design of algorithms in machine learning, and recent empirical studies show new insights on the optimization of margin distribution yet without theoretical supports. This work takes one step on this direction by providing a new generalization error bound, which is heavily relevant to margin distribution by incorporating factors such as average margin and semi-variance. Based on the theoretical results, we develop the MsvMAv approach for margin distribution optimization, and extensive experiments verify its superiority. An interesting future work is to exploit more effective statistics to characterize the whole margin distribution.

Acknowledgements

The authors want to thank the reviewers for helpful comments and suggestions. This work is partially supported by the NSFC (61921006, 61876078, 62006088). Wei Gao is the corresponding author.

References

[Aiolli *et al.*, 2008] F. Aiolli, G. Da San Martino, and A. Sperduti. A kernel method for the optimization of the margin distribution. In *ICANN*, pages 305–314, 2008.

[Bartlett and Mendelson, 2002] P. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *JMLR*, 3:463–482, 2002.

[Bartlett and Shawe-Taylor, 1999] P. Bartlett and J. Shawe-Taylor. Generalization performance of support vector machines and other pattern classifiers. *Advances in Kernel Methods: Support Vector Learning*, pages 43–54, 1999.

[Bartlett *et al.*, 2017] P. Bartlett, D. J. Foster, and M. Telgarsky. Spectrally-normalized margin bounds for neural networks. In *NIPS 30*, pages 6240–6249. 2017.

[Boser *et al.*, 1992] B. E. Boser, I. M. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *COLT*, pages 144–152, 1992.

[Breiman, 1999] L. Breiman. Prediction games and arcing algorithms. *Neural Comput.*, 11(7):1493–1517, 1999.

[Cheng *et al.*, 2016] F.-Y. Cheng, J. Zhang, and C.-H. Wen. Cost-sensitive large margin distribution machine for classification of imbalanced data. *Pattern Recognit. Lett.*, 80:107–112, 2016.

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

[Drucker *et al.*, 1997] H. Drucker, J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In *NIPS 9*, pages 155–161. 1997.

[Freund and Schapire, 1996] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *ICML*, pages 148–156, 1996.

[Gao and Zhou, 2013] W. Gao and Z.-H. Zhou. On the doubt about margin explanation of boosting. *Artif. Intell.*, 203:1–18, 2013.

[Garg and Roth, 2003] A. Garg and D. Roth. Margin distribution and learning. In *ICML*, pages 210–217, 2003.

[Grønlund *et al.*, 2020] A. Grønlund, L. Kamma, and K. G. Larsen. Near-tight margin-based generalization bounds for support vector machines. In *ICML*, pages 3779–3788, 2020.

[Ji *et al.*, 2021] Z.-W. Ji, N. Srebro, and M. Telgarsky. Fast margin maximization via dual acceleration. In *ICML*, pages 4860–4869, 2021.

[Jiang *et al.*, 2019] Y. Jiang, D. Krishnan, H. Mobahi, and S. Bengio. Predicting the generalization gap in deep networks with margin distributions. In *ICLR*, 2019.

[Kabán and Durrant, 2020] Ata Kabán and Robert J Durrant. Structure from randomness in halfspace learning with the zero-one loss. *JAIR*, 69:733–764, 2020.

[Markowitz, 1952] Harry Markowitz. Portfolio selection. *Journal of Finance*, 7(1):77–91, 1952.

[Pelckmans *et al.*, 2007] K. Pelckmans, J. A. K. Suykens, and B. De Moor. A risk minimization principle for a class of parzen estimators. In *NIPS 20*, pages 1137–1144. 2007.

[Qian *et al.*, 2022] M.-Z. Qian, Z. Ai, T. Zhang, and W. Gao. On the optimization of margin distribution. *CoRR*, abs/2204.14118, 2022.

[Rastogi *et al.*, 2020] R. Rastogi, P. Anand, and S. Chandra. Large-margin distribution machine-based regression. *Neural Comput. Appl.*, 32(8):3633–3648, 2020.

[Rosset *et al.*, 2003] S. Rosset, J. Zhu, and T. Hastie. Margin maximizing loss functions. In *NIPS 16*, pages 1237–1244. 2003.

[Schapire *et al.*, 1998] R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *Ann. Stat.*, 26(5):1651–1686, 1998.

[Schölkopf *et al.*, 2002] B. Schölkopf, A. J. Smola, and F. Bach. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, 2002.

[Shalev-Shwartz and Ben-David, 2014] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, Cambridge, 2014.

[Sherman and Morrison, 1950] J. Sherman and W. J. Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *Ann. Math. stat.*, 21(1):124–127, 1950.

[Shivaswamy and Jebara, 2010] P. K. Shivaswamy and T. Jebara. Maximum relative margin and data-dependent regularization. *JMLR*, 11(2), 2010.

[Sokolić *et al.*, 2017] J. Sokolić, R. Giryes, G. Sapiro, and M. R. Rodrigues. Robust large margin deep neural networks. *IEEE Trans. Signal Process.*, 65(16):4265–4280, 2017.

[Suykens *et al.*, 2002] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[Vapnik, 1982] V. Vapnik. *Estimation of Dependences based on Empirical Data*. Springer-Verlag, New York, 1982.

[Wei and Ma, 2020] C. Wei and T.-Y. Ma. Improved sample complexities for deep neural networks and robust classification via an all-layer margin. In *ICLR*, 2020.

[Weinstein *et al.*, 2020] B. Weinstein, S. Fine, and Y. Hel-Or. Margin-based regularization and selective sampling in deep neural networks. *CoRR*, abs/2009.06011, 2020.

[Zhang and Zhou, 2014] T. Zhang and Z.-H. Zhou. Large margin distribution machine. In *KDD*, pages 313–322, 2014.

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