MATHEMATICAL ENGINEERING
TECHNICAL REPORTS

DC Algorithm for
Extended Robust Support Vector Machine

Shuhei FUJIWARA, Akiko TAKEDA and Takafumi KANAMORI

METR 2014–38 December 2014

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.keisu.t.u-tokyo.ac.jp/research/techrep/index.html

DC Algorithm for Extended Robust Support Vector Machine

Shuhei Fujiwara[*] Akiko Takeda[*] Takafumi Kanamori[†]

Abstract

Non-convex extensions for Support Vector Machines (SVMs) have been developed for various purposes. For example, robust SVMs attain robustness to outliers using a non-convex loss function, while Extended ν-SVM (Eν-SVM) extends the range of the hyper-parameter introducing a non-convex constraint. We consider Extended Robust Support Vector Machine (ER-SVM) which is a robust variant of Eν-SVM. ER-SVM combines the two non-convex extensions of robust SVMs and Eν-SVM. Because of two non-convex extensions, the existing algorithm which is proposed by Takeda, Fujiwara and Kanamori needs to be divided into two parts depending on whether the hyper-parameter value is in the extended range or not. It also heuristically solves the non-convex problem in the extended range.

In this paper, we propose a new efficient algorithm for ER-SVM. The algorithm deals with two types of non-convex extensions all together never paying more computation cost than that of Eν-SVM and robust SVMs and finds a generalized Karush-Kuhn-Tucker (KKT) point of ER-SVM. Furthermore, we show that ER-SVM includes existing robust SVMs as a special case. Numerical experiments confirm the effectiveness of integrating the two non-convex extensions.

1 Introduction

Support Vector Machine (SVM) is one of the most successful machine learning models, and it has many extensions. The original form of SVM, which is called C-SVM [6], is popular because of its generalization ability and convexity. ν-SVM proposed by [16] is equivalent to C-SVM, and Extended ν-SVM (Eν-SVM) [10] is a non-convex extension of ν-SVM. Eν-SVM introduces a non-convex norm constraint instead of regularization term in the objective function, and the non-convex constraint makes it possible to extend

[*] Department of Mathematical Informatics, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan. {shuhei_fujiwara, takeda}@mist.i.u-tokyo.ac.jp

[†] Department of Computer Science and Mathematical Informatics, Nagoya University, Chikusa-ku, Nagoya-shi, Aichi 464-8603, Japan. kanamori@is.nagoya-u.ac.jp

the range of the hyper-parameter ν. Eν-SVM includes ν-SVM as a special case and Eν-SVM empirically outperforms ν-SVM owing to the extension (see [10]). Furthermore, [21] showed that Eν-SVM minimizes Conditional Value-at-Risk (CVaR) which is a popular convex and coherent risk measure in finance. However, CVaR is sensitive to tail-risks, and the same holds true for Eν-SVM. Unfortunately, it also implies that SVMs might not be sufficiently robust to outliers.

Various non-convex SVMs have been studied with the goal of ensuring robustness to outliers. Indeed, there are many models which are called robust SVM. In this paper, "robust SVMs" means any robust variants of SVMs. Especially, [5, 25] worked on Ramp-Loss SVM which is a popular robust SVM. The idea is to truncate the hinge-loss and bound the value of the loss function by a constant (see Figure 1). Not only hinge-loss, but also any loss functions can be extended by truncating in the same way of ramp-loss. The framework of such truncated loss functions has been studied, for example, in [17, 26]. Xu, Crammer and Schuurmans [25] also proposed *Robust Outlier Detection (ROD)* which is derived from Ramp-Loss SVM. While Ramp-Loss SVM and ROD are robust variants of C-SVM, Extended Robust SVM (ER-SVM), recently proposed in [20], is a robust variant of Eν-SVM.

1.1 Non-convex Optimization and DC Programming

The important issue on non-convex extensions is how to solve the difficult non-convex problems. Difference of Convex functions (DC) programming is a powerful framework for dealing with non-convex problems. It is known that various non-convex problems can be formulated as DC programs. For example, every function whose second partial derivatives are continuous everywhere has DC decomposition (cf. [8]).

DC Algorithm (DCA) introduced in [22] is one of the most efficient algorithm for DC programs. The basic idea behind the algorithms is to linearize the concave part and sequentially solve the convex subproblem. The local and global optimality conditions, convergence properties, and the duality of DC programs were well studied using convex analysis [14]. For general DC program, every limit point of the sequence generated by DCA is a *critical point* which is also called *generalized Karush-Kuhn-Tucker (KKT) point*. It is remarkable that DCA does not require differentiability in order to assure its convergence properties. Furthermore, it is known that DCA converges quite often to a global solution [9, 11].

In the machine learning literature, a similar method which is called *ConCave-Convex Procedure (CCCP)* [27] has been studied. The work [18] proposed constrained CCCP to deal with DC constraints, and [19] studied the global convergence properties of (constrained) CCCP, proving that the sequence generated by CCCP converges to a stationary point under condi-

Table 1: Relation of existing models: the models in right (resp. bottom) cell include the models in left (resp. top) cell as a special case.

		Regularizer	
		Convex	Non-Convex
Loss	Convex	C-SVM [6], ν-SVM [16]	Eν-SVM [10]
	Non-convex	Robust Outlier Detection [25] Ramp-Loss SVM [5, 25]	ER-SVM [20]

tions such as differentiability and strict convexity. However, since our model is not differentiable, we will use DCA for our problem and take advantage of theoretical results of DCA such as convergence properties.

1.2 Contributions

The main contribution of this paper is a new efficient algorithm based on DCA for ER-SVM. We prove that ER-SVM minimizes the difference of two CVaRs which are known to be convex risk measures. This result allows us to apply DCA to ER-SVM and gives an intuitive interpretation for ER-SVM. The previous algorithm proposed by [20] is heuristics and does not have a theoretical guarantee. However, our new algorithm finds a critical point which is also called generalized KKT point. Though ER-SVM enjoys both of non-convex extensions of Eν-SVM and robust SVMs such as ROD and Ramp-Loss SVM, our new algorithm is simple and comparable to those of Eν-SVM and Ramp-Loss SVM. While the existing algorithm for Eν-SVM [10] needs to use two different procedures depending on the value of the hyper-parameter ν, our new algorithm works with any value of the hyper-parameter ν. Besides, our algorithm is similar to Collobert et al.'s algorithm [5] of Ramp-Loss SVM which was shown to be fast.

Furthermore, we clarify the relation of ER-SVM, Ramp-Loss SVM and ROD. We show that ER-SVM includes Ramp-Loss SVM and ROD as a special case in the sense of KKT points. That is, a special case of ER-SVM (whose range of the hyper-parameter ν is limited), Ramp-Loss SVM and ROD share all KKT points. Therefore, as in Table 1, ER-SVM can be regarded not just as a robust variant of Eν-SVM but as a natural extension of Ramp-Loss SVM and ROD.

1.3 Outline of the Paper

This paper is organized as follows: Section 2 is preliminary. In Section 2.1 and 2.2, we introduce existing SVMs and their extensions. Section 2.3 briefly describes definitions and properties of some popular financial risk measures such as CVaR and VaR. Section 3 describes some important properties of ER-SVM. Section 3.1 gives a DC decomposition of ER-SVM using CVaRs which is a key property for our algorithm. Section 3.2 shows the relation of Ramp-Loss SVM, ROD, and ER-SVM. Section 4 describes our new algorithm after a short introduction of DC programming and DCA. The non-convex extension for regression and its algorithm are briefly discussed in Section 5. Finally, numerical result is presented in Section 6.

2 Preliminary

Here, let us address the binary classification of supervised learning. Suppose we have a set of training samples $\{(\boldsymbol{x}_i, y_i)\}_{i \in I}$ where $\boldsymbol{x}_i \in \mathbb{R}^n, y_i \in \{-1, 1\}$ and I is the index set of the training samples. $I_\pm$ is the index set such that $y_i = \pm 1$ and we suppose $|I_\pm| > 0$. SVM learns the decision function $h(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b$ and predicts the label of $\boldsymbol{x}$ as $\hat{y} = \text{sign}(h(\boldsymbol{x}))$. We define

$$r_i(\boldsymbol{w}, b) := -y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b),$$

wherein the absolute value of $r_i(\boldsymbol{w}, b)$ is proportional to the distance from the hyperplane $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0$ to the sample $\boldsymbol{x}_i$. $r_i(\boldsymbol{w}, b)$ becomes negative if the sample $\boldsymbol{x}_i$ is classified correctly and positive otherwise.

Though our algorithm can be extended to nonlinear models using kernel method, we consider linear models for simplicity. Instead, we mention kernel method in Section 3.4.

2.1 Support Vector Machines

2.1.1 Convex SVMs

C-SVM [6] is the most standard form of SVMs, which minimizes the hinge-loss and regularizer:

$$\min_{w,b} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C \sum_{i \in I} [1 + r_i(\boldsymbol{w}, b)]^+,$$

where $[x]^+ := \max\{0, x\}$ and $C > 0$ is a hyper-parameter. ν-SVM [16] is formulated as

$$\begin{aligned} \min_{w,b,\rho} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 - \nu\rho + \frac{1}{|I|} \sum_{i \in I} [\rho + r_i(\boldsymbol{w}, b)]^+ \\ \text{s.t.} \quad & \rho \geq 0, \end{aligned} \tag{1}$$

which is equivalent to C-SVM if ν and C are set appropriately. The hyper-parameter $\nu \in (0, 1]$ has an upper threshold

$$\overline{\nu} := 2 \min \{|I_+|, |I_-|\} / |I|$$

and a lower threshold $\underline{\nu}$. The optimal solution is trivial ($\boldsymbol{w} = \boldsymbol{0}$) if $\nu \leq \underline{\nu}$, and the optimal value is unbounded if $\nu > \overline{\nu}$ (see [4]). Therefore, we define the range of ν for ν-SVM as $(\underline{\nu}, \overline{\nu}]$.

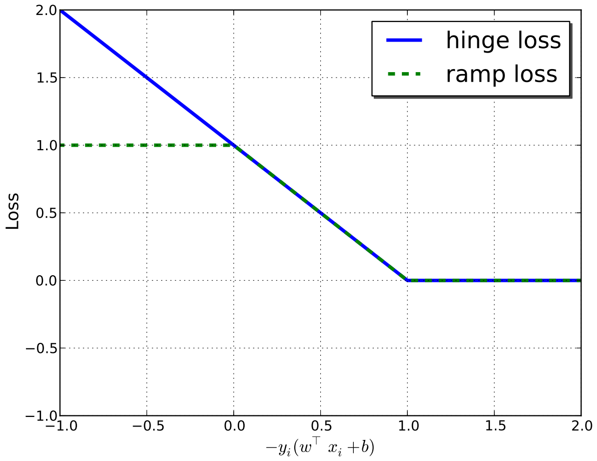


Figure 1: Loss functions

2.1.2 Non-Convex SVMs

Here, we introduce two types of non-convex extensions for SVMs. The first is Extended ν-SVM (Eν-SVM) [10] which is an extended model of ν-SVM. Eν-SVM introducing a non-convex constraint is formulated as

$$\begin{aligned} \min_{\rho,w,b} \quad & -\nu\rho + \frac{1}{|I|}\sum_{i\in I}[\rho + r_i(\boldsymbol{w}, b)]^+ \\ \text{s.t.} \quad & \|\boldsymbol{w}\|^2 = 1. \end{aligned} \tag{2}$$

Eν-SVM has the same set of optimal solutions to ν-SVM if $\nu > \underline{\nu}$, and obtains non-trivial solutions ($\boldsymbol{w} \neq \boldsymbol{0}$) even if $\nu \leq \underline{\nu}$ owing to the constraint $\|\boldsymbol{w}\|^2 = 1$. Therefore, we define the range of ν for Eν-SVM as $(0, \overline{\nu}]$. Eν-SVM removes the lower threshold $\underline{\nu}$ of ν-SVM and extends the admissible range of the hyper-parameter ν. It was empirically shown that Eν-SVM sometimes achieves high accuracy in the extended range of ν. We will mention other concrete advantages of Eν-SVM over ν-SVM in Section 3.3.

The second is Ramp-Loss SVM which is a robust variant of C-SVM. The resulting classifier is robust to outliers at the expense of the convexity of the hinge-loss function. The idea behind ramp-loss is to clip large losses with a hyper-parameter $s \geq 0$.

$$\min_{w,b} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i\in I}\min\{[1 + r_i(\boldsymbol{w}, b)]^+, s\}. \tag{3}$$

$C \in (0, \infty)$ is also a hyper-parameter. Ramp-loss can be described as a difference of hinge-loss functions: therefore ConCave-Convex Procedure (CCCP),

Table 2: If ν is greater than lower threshold (Case C), the non-convex constraint of Eν-SVM and ER-SVM can be relaxed to a convex constraint without changing their optimal solutions. Case C of Eν-SVM is equivalent to ν-SVM.

		Case N	Case C	
ν-SVM	Range of ν	$\nu \le \underline{\nu}$	$\underline{\nu} < \nu \le \overline{\nu}$	$\overline{\nu} < \nu$
	Opt. Val.	0	negative	unbounded
	Opt. Sol.	$\boldsymbol{w} = \boldsymbol{0}$	admissible	–
Eν-SVM	Range of ν	$\nu \le \underline{\nu}$	$\underline{\nu} < \nu \le \overline{\nu}$	$\overline{\nu} < \nu$
	Opt. Val.	non-negative	negative	unbounded
	Opt. Sol.	admissible	admissible	–
	Constraint	$\lVert\boldsymbol{w}\rVert^2 = 1$	$\lVert\boldsymbol{w}\rVert^2 \le 1$	
ER-SVM	Range of ν	$\nu \le \underline{\nu}_\mu$	$\underline{\nu}_\mu < \nu \le \overline{\nu}_\mu$	$\overline{\nu}_\mu < \nu$
	Opt. Val.	non-negative	negative	unbounded
	Opt. Sol.	admissible	admissible	–
	Constraint	$\lVert\boldsymbol{w}\rVert^2 = 1$	$\lVert\boldsymbol{w}\rVert^2 \le 1$	

which is an effective algorithm for DC programming, can be applied to the problem (see [5] for details). On the other hand, [25] gave another representation of Ramp-Loss SVM using η-hinge-loss:

$$\begin{aligned} \min_{w,b,\eta} \quad & \frac{1}{2}\lVert\boldsymbol{w}\rVert^2 + C\sum_{i\in I}\{\eta_i[1 + r_i(\boldsymbol{w}, b)]^+ + s(1-\eta_i)\} \\ \text{s.t.} \quad & 0 \le \eta_i \le 1, \end{aligned} \tag{4}$$

and applied semidefinite programming relaxation to (4). They also proposed *Robust Outlier Detection (ROD)*:

$$\begin{aligned} \min_{w,b,\eta} \quad & \frac{1}{2}\lVert\boldsymbol{w}\rVert^2 + C\sum_{i\in I}\eta_i[1 + r_i(\boldsymbol{w}, b)]^+ \\ \text{s.t.} \quad & 0 \le \eta_i \le 1, \ \sum_{i\in I}(1-\eta_i) = \mu|I|, \end{aligned} \tag{5}$$

which is derived from Ramp-Loss SVM (4). $C \in (0, \infty)$ and $\mu \in [0, 1)$ are hyper-parameters. In the original formulation [25], ROD is defined with an inequality constraint $\sum_{i\in I}(1-\eta_i) \le \mu|I|$ but it is replaced by the equality variant since it does not change the optimal value.

2.2 Extended Robust Support Vector Machine

Recently, [20] proposed Extended Robust SVM (ER-SVM):

$$\begin{aligned} \min_{w,b,\rho,\eta} \quad & -\rho + \frac{1}{(\nu-\mu)|I|}\sum_{i\in I}\eta_i[\rho + r_i(\boldsymbol{w},b)]^+ \\ \text{s.t.} \quad & 0 \le \eta_i \le 1, \sum_{i\in I}(1-\eta_i) = \mu|I|, \|\boldsymbol{w}\|^2 = 1, \end{aligned} \tag{6}$$

where $\nu \in (\mu, 1]$ and $\mu \in [0,1)$ are hyper-parameters.

Note that we relax the 0-1 integer constraints $\eta_i \in \{0,1\}$ of the original formulation in [20] and replace $\sum_{i\in I}(1-\eta_i) \le \mu|I|$ of the original one by the equality variant. The relaxation does not change the problem if $\mu|I| \in \mathbb{N}$. More precisely, if $\mu|I| \in \mathbb{N}$, ER-SVM (6) has an optimal solution such that $\boldsymbol{\eta}^* \in \{0,1\}^{|I|}$. In this case, ER-SVM (6) removes $\mu|I|$ samples and applies Eν-SVM (2) using the rest, as intended in [20]. Hence, in this paper, we use the formulation (6) and call it ER-SVM.

It can be easily seen that for fixed μ, the optimal value of (6) is decreasing with respect to ν. Moreover, it is shown in [20, Lemma 1] that the non-convex constraint $\|\boldsymbol{w}\|^2 = 1$ can be relaxed to $\|\boldsymbol{w}\|^2 \le 1$ without changing the optimal solution as long as the optimal value is negative; just like Eν-SVM, ER-SVM (6) has a threshold (we denote it by $\underline{\nu}_\mu$) of the hyper-parameter ν where the optimal value equals zero and the non-convex constraint $\|\boldsymbol{w}\|^2 = 1$ is essential for ER-SVM with $\nu \le \underline{\nu}_\mu$. The non-convex constraint $\|\boldsymbol{w}\|^2 = 1$ removes the lower threshold of ν and extends the admissible range of ν in the same way of Eν-SVM (see Table 2.2). We will show in Section 3.2 that a special case (case C in Table 2.2) of ER-SVM is equivalent to ROD (5) and Ramp-Loss SVM (4) in the way where a special case (case C) of Eν-SVM is equivalent to ν-SVM. Hence, ER-SVM can be seen as a natural extension of Robust SVMs such as ROD and Ramp-Loss SVM. ER-SVM (6) also has an upper threshold $\overline{\nu}_\mu$ which makes the problem bounded similar to Eν-SVM and ν-SVM.

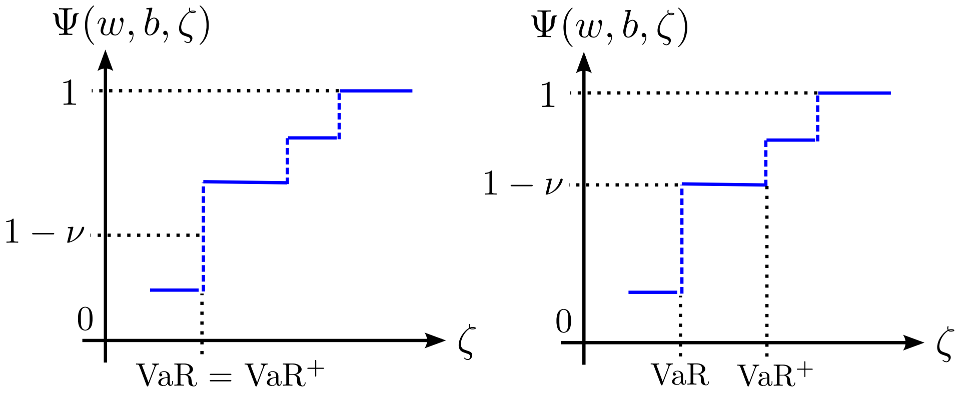
2.3 Financial Risk Measures

We define financial risk measures as in [15]. Let us consider the distribution of $r_i(\boldsymbol{w},b)$:

$$\begin{aligned} \Psi(\boldsymbol{w},b,\zeta) :=& P(r_i(\boldsymbol{w},b) \le \zeta) \\ =& \frac{1}{|I|}|\{i \in I : r_i(\boldsymbol{w},b) \le \zeta\}|. \end{aligned}$$

For $\nu \in (0,1]$, let $\zeta_{1-\nu}(\boldsymbol{w},b)$ be the $100(1-\nu)$-percentile of the distribution, known as the Value-at-Risk (VaR) in finance. More precisely, $(1-\nu)$-VaR is defined as

$$\zeta_{1-\nu}(\boldsymbol{w},b) := \min\{\zeta : \Psi(\boldsymbol{w},b,\zeta) \ge 1-\nu\},$$



(a)

(b)

Figure 2: Difference between VaR and VaR$^+$: VaR$^+$ corresponds to VaR if equation $\Psi(\boldsymbol{w}, b, \zeta) = 1 - \nu$ has no solution (Figure 2(a)).

and $(1-\nu)$-VaR$^+$ which we call $(1-\nu)$-*upper-VaR* is defined as

$$\zeta^+_{1-\nu}(\boldsymbol{w}, b) := \inf\{\zeta : \Psi(\boldsymbol{w}, b, \zeta) > 1 - \nu\}.$$

The difference between VaR and VaR$^+$ is illustrated in Figure 2.

Conditional Value-at-Risk (CVaR) is also a popular risk measure in finance because of its coherency and computational properties. Formally, $(1-\nu)$-CVaR is defined as

$$\varphi_{1-\nu}(\boldsymbol{w}, b) := \text{mean of the } (1-\nu)\text{-tail distribution of } r_i(\boldsymbol{w}, b),$$

where the $(1-\nu)$-tail distribution is defined by

$$\Psi_{1-\nu}(\boldsymbol{w}, b, \zeta) := \begin{cases} 0 & \text{for } \zeta < \zeta_{1-\nu}(\boldsymbol{w}, b) \\ \dfrac{\Psi(\boldsymbol{w}, b, \zeta) - (1-\nu)}{\nu} & \text{for } \zeta \geq \zeta_{1-\nu}(\boldsymbol{w}, b). \end{cases}$$

The computational advantage of CVaR over VaR is shown by the following theorem.

Theorem 1 (Rockafellar and Uryasev [15])**.** *One has*

$$\varphi_{1-\nu}(\boldsymbol{w}, b) = \min_{\zeta} F_{1-\nu}(\boldsymbol{w}, b, \zeta), \tag{7}$$

where

$$F_{1-\nu}(\boldsymbol{w}, b, \zeta) := \zeta + \frac{1}{\nu|I|}\sum_{i \in I}[r_i(\boldsymbol{w}, b) - \zeta]^+.$$

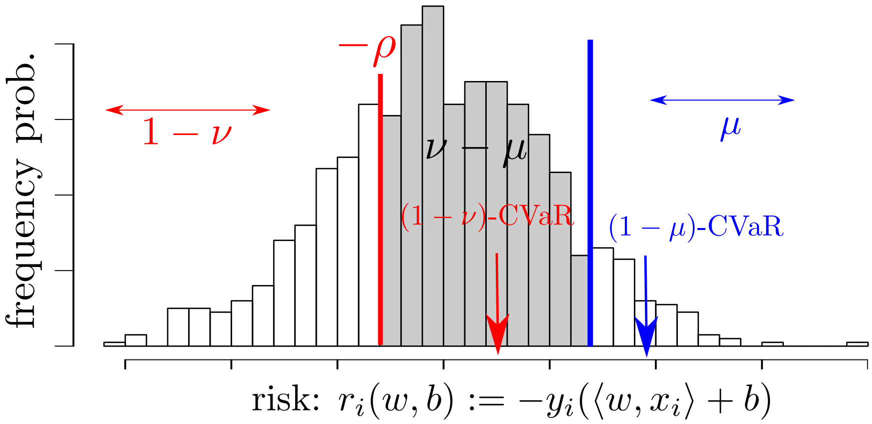


Figure 3: Distribution of $r_i(\boldsymbol{w}, b)$: ER-SVM minimizes the mean of r_i, $i \in I$, in the gray area.

Moreover

$$\zeta_{1-\nu}(\boldsymbol{w}, b) = \textit{lower endpoint of } \operatorname*{argmin}_{\zeta} F_{1-\nu}(\boldsymbol{w}, b, \zeta)$$
$$\zeta^{+}_{1-\nu}(\boldsymbol{w}, b) = \textit{upper endpoint of } \operatorname*{argmin}_{\zeta} F_{1-\nu}(\boldsymbol{w}, b, \zeta)$$

hold.

Using the property that CVaR is a polyhedral convex function (i.e., piecewise linear and convex function), CVaR can be described as a maximum of linear functions as follows:

$$\varphi_{1-\mu}(\boldsymbol{w}, b) = \max_{\eta} \left\{ \frac{1}{\mu|I|} \sum_{i \in I} (1-\eta_i) r_i(\boldsymbol{w}, b) : 0 \le \eta_i \le 1, \; \sum_{i \in I} (1-\eta_i) = \mu|I| \right\}. \tag{8}$$

We will use the above different representations of CVaR (7) and (8) in the proof of Proposition 1.

3 Properties of Extended Robust SVM

3.1 Decomposition using Conditional Value-at-Risks

Here, we will give an intuitive interpretation to ER-SVM (6) using two CVaRs. Eν-SVM has been shown to minimize $(1-\nu)$-CVaR $\varphi_{1-\nu}(\boldsymbol{w}, b)$ in [21]. On the other hand, ER-SVM (6) ignores the fraction μ of the samples and solves Eν-SVM using the rest: that is, ER-SVM (6) minimizes CVaR using the rest of the samples. Hence, ER-SVM can be regarded as the one that minimizes the mean of the distribution of the gray area in Figure 3. The mean of the gray area in Figure 3 can be described using two CVaRs as in [24]:

Proposition 1. *ER-SVM (6) is described as a problem minimizing the difference of two convex functions using the two CVaRs:*

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{\nu-\mu}\{\nu\varphi_{1-\nu}(\boldsymbol{w},b) - \mu\varphi_{1-\mu}(\boldsymbol{w},b)\} \\ \text{s.t.} \quad & \|\boldsymbol{w}\|^2 = 1. \end{aligned} \tag{9}$$

The proof of Proposition 1 is shown in Appendix A.

Since CVaR is convex, this decomposition allows us to apply the existing techniques for DC program to ER-SVM. A similar model which relaxes the non-convex constraint $\|\boldsymbol{w}\|^2 = 1$ of (9) by $\|\boldsymbol{w}\|^2 \leq 1$ was recently proposed in [23]. As in Table 2.2, Tsyurmasto, Uryasev and Gotoh's model [23] is a special case (Case C) of ER-SVM, and their model is essentially equivalent to Ramp-Loss SVM and ROD.

3.2 Relationship with Existing Models

Here, we discuss the relation of ER-SVM, ROD, and Ramp-Loss SVM using the KKT conditions shown in Appendix B. Let us begin with showing the equivalence of Ramp-Loss SVM and ROD. Though the formulation of ROD is derived from Ramp-Loss, their equivalence is not particularly discussed in the original paper [25].

Lemma 1 (Relation between ROD and Ramp-Loss SVM)**.** *Ramp-Loss SVM* (4) *and ROD* (5) *share all KKT points in the following sense.*

1. *Let* $(\boldsymbol{w}^*, b^*, \boldsymbol{\eta}^*)$ *be a KKT point of Ramp-Loss SVM* (4)*. Then it is also a KKT point of ROD* (5) *with* $\mu = \frac{1}{|I|}\sum_{i\in I}(1-\eta_i^*)$*.*
2. *A KKT point of ROD* (5) *having the Lagrange multiplier* τ^* *for* $\sum_{i\in I}(1-\eta_i) = \mu|I|$ *is also a KKT point of Ramp-Loss SVM* (4) *with* $s = \frac{\tau^*}{C}$*.*

The proof of Lemma 1 is shown in Appendix B.1.

Theorem 2 shows the equivalence of ROD and the special case (Case C in Table 2.2) of ER-SVM.

Theorem 2 (Relation between ER-SVM and ROD)**.** *Case C of ER-SVM (in Table 2.2), that is* (26)*, and ROD* (5) *share all KKT points in the following sense.*

1. *Let* $(\boldsymbol{w}^*, b^*)$ *satisfy the KKT conditions of ROD* (5) *and suppose* $\boldsymbol{w}^* \neq \boldsymbol{0}$*.* $\frac{1}{\|\boldsymbol{w}^*\|}(\boldsymbol{w}^*, b^*)$ *satisfies the KKT conditions of Case C of ER-SVM with a corresponding hyper-parameter value.*

2. *Let $(\boldsymbol{w}^*, b^*, \rho^*)$ satisfy the KKT conditions of Case C of ER-SVM. Suppose $\rho^* \neq 0$ and the objective value is non-zero. $\frac{1}{\rho^*}(\boldsymbol{w}^*, b^*)$ satisfies the KKT conditions of ROD with a corresponding hyper-parameter value.*

See Appendix B.2 for the proof of Theorem 2.

From Lemma 1 and Theorem 2, Ramp-Loss SVM and ROD are regarded as a special case (Case C in Table 2.2) of ER-SVM. As we showed in Section 3.1, Tsyurmasto, Uryasev and Gotoh 's model [23] is also equivalent to Case C (in Table 2.2) of ER-SVM. Theorem 2 is similar to the relation between C-SVM and ν-SVM (which is a special case of Eν-SVM). It was shown that the sets of global solutions of C-SVM and ν-SVM correspond to each other when the hyper-parameters are set properly [4, 16]. We used KKT conditions to show the relation of non-convex models.

3.3 Motivation for Non-Convex Regularizer

The motivation of the non-convex constraint in Eν-SVM is somewhat not intuitive while that of the robust extension for loss function is easy to understand. Here, we show the motivation of the extension.

As Table 2.2 shows, the non-convex constraint removes the lower threshold of the hyper-parameter ν. The extension of the admissible range of ν has some important advantages. Empirically, [10] showed examples where Eν-SVM outperforms ν-SVM owing to the extended range of ν. They also pointed that small ν achieves sparse solutions since ν controls the number of support vectors.

Here, we show the case where the admissible range of ν for ν-SVM is empty. Theorem 3 gives an explicit condition where C-SVM and ν-SVM obtain a trivial classifier for any hyper-parameter value of C and ν. The conditions also apply to robust SVMs after removing all outliers with $\eta_i^* = 0$.

Rifkin, Pontil and Verri [13] studied the condition where C-SVM obtains a trivial solution. We directly connect their statements to ν-SVM and strengthen them as in Theorem 3 by adding a geometric interpretation for ν-SVM in the case where the admissible range of ν is empty for ν-SVM.

Theorem 3. *Suppose $0 < |I_-| \leq |I_+|$ without loss of generality. Let us define Reduced Convex Hull (RCH) [7]:*

$$RCH_{\pm}(\nu) = \left\{ \sum_{i \in I_{\pm}} \lambda_i \boldsymbol{x}_i : \sum_{i \in I_{\pm}} \lambda_i = 1,\ 0 \leq \lambda_i \leq \frac{2}{\nu |I|} \text{ for all } i \right\}. \tag{10}$$

C-SVM and ν-SVM lead to the trivial classifier ($\boldsymbol{w} = \boldsymbol{0}$) for any hyper-parameter values $C \in (0, \infty)$ and $\nu \in (\underline{\nu}, \overline{\nu}]$ if and only if a training set $\{\boldsymbol{x}_i, y_i\}_{i \in I}$ satisfies $\sum_{i \in I_-} \frac{1}{|I_-|}\boldsymbol{x}_i \in RCH_+(\overline{\nu})$. When $|I_-| > |I_+|$, the above statement is modified by $\sum_{i \in I_+} \frac{1}{|I_+|}\boldsymbol{x}_i \in RCH_-(\overline{\nu})$.

The proof is shown in Appendix C.

3.4 Kernelization

Learning methods using linear models can be extended to more powerful learning algorithms by using kernel methods. Here, let us briefly introduce the kernel variant of ER-SVM (6). In kernel methods, the input sample $\boldsymbol{x}$ is mapped into $\phi(\boldsymbol{x})$ in a high (even infinite) dimensional inner product space $\mathcal{H}$, and the classifier of the form $h(\boldsymbol{x}) = \langle \boldsymbol{w}, \phi(\boldsymbol{x})\rangle_{\mathcal{H}} + b$ is learned from the training samples, where $\langle \boldsymbol{a}, \boldsymbol{b}\rangle_{\mathcal{H}}$ is the inner product of $\boldsymbol{a}$ and $\boldsymbol{b}$ in $\mathcal{H}$. The kernel function $k(\boldsymbol{x}, \boldsymbol{x}')$ is defined as $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}')\rangle_{\mathcal{H}}$.

We show how the equality constraint $\|\boldsymbol{w}\|^2 = 1$ in ER-SVM (6) is dealt with in the kernel method. Let $\mathcal{S}$ be the subspace in $\mathcal{H}$ spanned by $\phi(\boldsymbol{x}_i), i \in I$, and $\mathcal{S}^{\perp}$ be the orthogonal subspace of $\mathcal{S}$. Then the weight vector $\boldsymbol{w}$ is decomposed into $\boldsymbol{w} = \boldsymbol{v} + \boldsymbol{v}^{\perp}$, where $\boldsymbol{v} \in \mathcal{S}$ and $\boldsymbol{v}^{\perp} \in \mathcal{S}^{\perp}$. The vector $\boldsymbol{v}^{\perp}$ does not affect the value of $r_i(\boldsymbol{w}, b) = -y_i(\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i)\rangle_{\mathcal{H}} + b)$. The vector $\boldsymbol{v}$ is expressed as the linear combination of $\phi(\boldsymbol{x}_i)$ such as $\boldsymbol{v} = \sum_{i\in I} \alpha_i \phi(\boldsymbol{x}_i)$. If $\mathcal{S} = \mathcal{H}$ holds, $\boldsymbol{w} = \boldsymbol{v}$ should hold and the constraint $\langle \boldsymbol{w}, \boldsymbol{w}\rangle_{\mathcal{H}} = 1$ is equivalent with $\sum_{i,j\in I} \alpha_i\alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1$. On the other hand, when $\mathcal{S} \neq \mathcal{H}$, i.e., the dimension of $\mathcal{S}^{\perp}$ is not zero, one can prove that the constraint $\langle \boldsymbol{w}, \boldsymbol{w}\rangle_{\mathcal{H}} = 1$ is replaced with the convex constraint $\sum_{i,j\in I} \alpha_i\alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq 1$, where the fact that the gram matrix $(k(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j}$ is non-negative definite is used. Indeed, since the objective function depends on $\boldsymbol{w}$ through the component $\boldsymbol{v}$, the constraint on $\boldsymbol{w}$ can be replaced with its projection onto the subspace $\mathcal{S}$. Thus the above convex constraint is obtained unless $\mathcal{S} = \mathcal{H}$. In such case, the kernel variant of ER-SVM is given as

$$\begin{aligned}
\min_{\alpha,b,\rho,\eta} \quad & -\rho + \frac{1}{(\nu-\mu)|I|}\sum_{i\in I} \eta_i[\rho + r_i^k(\alpha, b)]^+ \\
\text{s.t.} \quad & 0 \leq \eta_i \leq 1, \sum_{i\in I}(1-\eta_i) = \mu|I|, \\
& \sum_{i,j\in I} \alpha_i\alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq 1,
\end{aligned} \tag{11}$$

where $r_i^k(\alpha, b) = -y_i(\sum_{j\in I} k(\boldsymbol{x}_i, \boldsymbol{x}_j)\alpha_j + b)$. When $\mathcal{S} = \mathcal{H}$ holds, the inequality constraint of α should be replaced with the equality constraint.

4 Algorithm

Let us begin with a brief introduction of Difference of Convex functions (DC) program and DC Algorithm (DCA). DC program is formulated by using lower semicontinuous proper convex functions u and v as

$$\min_{z}\{f(\boldsymbol{z}) := u(\boldsymbol{z}) - v(\boldsymbol{z}) : \boldsymbol{z} \in \mathbb{R}^n\}. \tag{12}$$

DC Algorithm (DCA) is an efficient algorithm for (12) and theoretically well-studied in e.g., [11]. We shall use simplified DCA, which is the standard form of DCA. Simplified DCA sequentially linearizes the concave part in (12) and solves convex subproblems as follows:

$$\boldsymbol{z}_{k+1} \in \operatorname*{argmin}_{z} \{u(\boldsymbol{z}) - (v(\boldsymbol{z}_k) + \langle \boldsymbol{z} - \boldsymbol{z}_k, \boldsymbol{g}_k \rangle)\}, \tag{13}$$

where $\boldsymbol{g}_k \in \partial v(\boldsymbol{z}_k)$ is a subgradient of v at $\boldsymbol{z}_k$. The sequence $\{\boldsymbol{z}_k\}$ generated by Simplified DCA has the following good convergence properties:

- the objective value is decreasing (i.e., $f(\boldsymbol{z}_{k+1}) \leq f(\boldsymbol{z}_k)$),
- DCA has linear convergence,
- every limit point of the sequence $\{\boldsymbol{z}_k\}$ is a *critical point* of $u - v$, which is also called *generalized KKT point.*

$\boldsymbol{z}^*$ is said to be a critical point of $u - v$ if $\partial u(\boldsymbol{z}^*) \cap \partial v(\boldsymbol{z}^*) \neq \emptyset$. It implies that a critical point $\boldsymbol{z}^*$ has $\boldsymbol{g}_u \in \partial u(\boldsymbol{z}^*)$ and $\boldsymbol{g}_v \in \partial v(\boldsymbol{z}^*)$ such that $\boldsymbol{g}_u - \boldsymbol{g}_v = 0$ which is a necessary condition for local minima. When (12) has a convex constraint $\boldsymbol{z} \in Z$, we can define the critical point by replacing u with $u(\boldsymbol{z}) + \delta(\boldsymbol{z} \mid Z)$, where $\delta(\boldsymbol{z} \mid Z)$ is an indicator function equal to 0 if $\boldsymbol{z} \in Z$ and $+\infty$ otherwise.

4.1 DCA for Extended Robust SVM

As shown in Section 3.1, ER-SVM (6) can be described as a difference of CVaRs (9). Moreover, (9) can be reformulated into a problem of minimizing the DC objective upon a convex constraint using a sufficiently large constant t:

$$\begin{array}{ll} \min\limits_{w,b} & \nu\varphi_{1-\nu}(\boldsymbol{w}, b) - \mu\varphi_{1-\mu}(\boldsymbol{w}, b) - t\|\boldsymbol{w}\|^2 \\ \text{s.t.} & \|\boldsymbol{w}\|^2 \leq 1. \end{array} \tag{14}$$

This reformulation is a special case of the *exact penalty approach* (see [11, 12]). There exists t_o such that (9) and (14) have the same set of optimal solutions for all $t > t_o$. We can estimate an upper bound of t_o in our case by invoking the following lemma.

Lemma 2. *If $t \geq 0$ in* (14) *is sufficiently large such that the optimal value of (14) is negative,* (9) *and (14) have the same set of optimal solutions.*

The key point in the proof of Lemma 2 is that CVaR has a positive homogeneity (i.e., $\varphi(a\boldsymbol{w}, ab) = a\varphi(\boldsymbol{w}, b)$ for all a such that $a \in \mathbb{R}, a > 0$). This is a well-known property of coherent risk measures such as CVaR (e.g., [1]).

Proof of Lemma 2. Let $(\boldsymbol{w}^*, b^*)$ be an optimal solution of (14) and suppose to the contrary that $\|\boldsymbol{w}^*\| < 1$. $(\frac{\boldsymbol{w}^*}{\|\boldsymbol{w}^*\|}, \frac{b^*}{\|\boldsymbol{w}^*\|})$ achieves a smaller objective value than $(\boldsymbol{w}^*, b^*)$ since

$$\begin{aligned}
&\nu\varphi_{1-\nu}(\frac{\boldsymbol{w}^*}{\|\boldsymbol{w}^*\|}, \frac{b^*}{\|\boldsymbol{w}^*\|}) - \mu\varphi_{1-\mu}(\frac{\boldsymbol{w}^*}{\|\boldsymbol{w}^*\|}, \frac{b^*}{\|\boldsymbol{w}^*\|}) - t\frac{\|\boldsymbol{w}^*\|^2}{\|\boldsymbol{w}^*\|^2} \\
&= \frac{1}{\|\boldsymbol{w}^*\|}\{\nu\varphi_{1-\nu}(\boldsymbol{w}^*, b^*) - \mu\varphi_{1-\mu}(\boldsymbol{w}^*, b^*) - t\|\boldsymbol{w}^*\|\} \\
&\leq \frac{1}{\|\boldsymbol{w}^*\|}\underbrace{\{\nu\varphi_{1-\nu}(\boldsymbol{w}^*, b^*) - \mu\varphi_{1-\mu}(\boldsymbol{w}^*, b^*) - t\|\boldsymbol{w}^*\|^2\}}_{\text{negative}} \\
&< \nu\varphi_{1-\nu}(\boldsymbol{w}^*, b^*) - \mu\varphi_{1-\mu}(\boldsymbol{w}^*, b^*) - t\|\boldsymbol{w}^*\|^2.
\end{aligned}$$

However, this contradicts the optimality of $(\boldsymbol{w}^*, b^*)$. Therefore, the optimal solution of (14) satisfies $\|\boldsymbol{w}\| = 1$, which implies that it is also optimal to (9). □

Therefore, (14) is represented as the following DC program:

$$\min_{w,b}\{u(\boldsymbol{w}, b) - v(\boldsymbol{w}, b)\} \tag{15}$$

where

$$\begin{aligned}
u(\boldsymbol{w}, b) &= \delta(\boldsymbol{w} \mid W) + \nu\varphi_{1-\nu}(\boldsymbol{w}, b) \\
v(\boldsymbol{w}, b) &= \mu\varphi_{1-\mu}(\boldsymbol{w}, b) + t\|\boldsymbol{w}\|^2 \\
W &= \{\boldsymbol{w} \mid \|\boldsymbol{w}\|^2 \leq 1\}.
\end{aligned}$$

Here, we apply simplified DCA to the problem (15). At kth iteration of simplified DCA, we solve a subproblem as in (13) linearizing the concave part. Let $(\boldsymbol{w}^k, b^k)$ be the solution obtained in the previous iteration $k-1$ of simplified DCA. The subproblem of simplified DCA for (15) is described as

$$\begin{array}{ll}
\min\limits_{w,b} & \nu\varphi_{1-\nu}(\boldsymbol{w}, b) - \mu\langle \boldsymbol{g}_w^k, \boldsymbol{w}\rangle - \mu g_b^k b - 2t\langle \boldsymbol{w}^k, \boldsymbol{w}\rangle \\
\text{s.t.} & \|\boldsymbol{w}\|^2 \leq 1,
\end{array} \tag{16}$$

where $(\boldsymbol{g}_w^k, g_b^k) \in \partial\varphi_{1-\mu}(\boldsymbol{w}^k, b^k)$. $\partial\varphi_{1-\mu}$ is the subdifferential of $\varphi_{1-\mu}$ and $(\boldsymbol{g}_w^k, g_b^k)$ is a subgradient of $\varphi_{1-\mu}$ at $(\boldsymbol{w}^k, b^k)$. The optimal solution of (16) is denoted by $(\boldsymbol{w}^{k+1}, b^{k+1})$. We will show how to calculate a subgradient $(\boldsymbol{g}_w^k, g_b^k)$ and how to choose a sufficiently large constant t.

Subdifferential of CVaR Here, we show how to calculate the subdifferential of CVaR (8). The following technique is described in [24]. The subdifferential of CVaR at $(\boldsymbol{w}^k, b^k)$ is

$$\partial_w \varphi_{1-\mu}(\boldsymbol{w}^k, b^k) = \mathrm{co}\left\{ -\frac{1}{\mu|I|}\sum_{i\in I}(1-\eta_i^k)y_i\boldsymbol{x}_i : \boldsymbol{\eta}^k \in H(\boldsymbol{w}^k, b^k)\right\},$$

$$\partial_b \varphi_{1-\mu}(\boldsymbol{w}^k, b^k) = \mathrm{co}\left\{ -\frac{1}{\mu|I|}\sum_{i\in I}(1-\eta_i^k)y_i : \boldsymbol{\eta}^k \in H(\boldsymbol{w}^k, b^k)\right\},$$

where $\mathrm{co}\, X$ is the convex hull of the set X and

$$H(\boldsymbol{w}^k, b^k) = \underset{\eta}{\mathrm{argmax}}\left\{\sum_{i\in I}(1-\eta_i)r_i(\boldsymbol{w}^k, b^k) : \sum_{i\in I}(1-\eta_i) = \mu|I|, 0 \le \eta_i \le 1\right\}. \tag{17}$$

We can easily find an optimal solution $\boldsymbol{\eta}^k \in H(\boldsymbol{w}^k, b^k)$ by assigning 0 to η_i^k in descending order of $r_i(\boldsymbol{w}^k, b^k)$ for all i.

Efficient Update of t The update of the large constant t in each iteration makes our algorithm more efficient. We propose to use, in the kth iteration, t_k such that

$$t^k > \nu\varphi_{1-\nu}(\boldsymbol{w}^k, b^k) - \mu\varphi_{1-\mu}(\boldsymbol{w}^k, b^k). \tag{18}$$

The condition (18) ensures the optimal value of (14) being negative, since the solution $(\boldsymbol{w}^k, b^k)$ in the previous iteration has achieved a negative objective value. With such t^k, Lemma 2 holds.

Explicit Form of Subproblem We are ready to describe the subproblem (16) explicitly. Using the above results and substituting (7) for $\varphi_{1-\nu}(\boldsymbol{w}, b)$, (16) results in

$$\begin{aligned} \min_{w,b,\rho,\xi} \quad & -\nu\rho + \frac{1}{|I|}\sum_{i\in I}\xi_i - \frac{1}{|I|}\sum_{i\in I}(1-\eta_i^k)r_i(\boldsymbol{w}, b) - 2t^k\langle \boldsymbol{w}^k, \boldsymbol{w}\rangle \\ \text{s.t.} \quad & \xi_i \ge \rho + r_i(\boldsymbol{w}, b), \xi_i \ge 0, \|\boldsymbol{w}\|^2 \le 1. \end{aligned} \tag{19}$$

Hence, we summarize our algorithm as Algorithm 1.

5 Regression

Some parts of our analysis and algorithm can be applied to regression. Regression models seek to estimate a linear function $f(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x}\rangle + b$ based on data $(\boldsymbol{x}_i, y_i) \in \mathbb{R}^n \times \mathbb{R}$, $i \in I$. ν-Support Vector Regression (ν-SVR) [16] is formulated as

$$\min_{w,b,\epsilon} \frac{1}{2C}\|\boldsymbol{w}\|^2 + \nu\epsilon + \frac{1}{|I|}\sum_{i\in I}[|\langle \boldsymbol{w}, \boldsymbol{x}_i\rangle + b - y_i| - \epsilon]_+, \tag{20}$$

Algorithm 1 DCA for Difference of CVaRs

```
Input: $\mu \in [0,1)$, $\nu \in (\mu, \overline{\nu}_\mu]$, $(\boldsymbol{w}^0, b^0)$ such that $\|\boldsymbol{w}^0\|^2 = 1$ and a small
    value $\epsilon_1, \epsilon_2 > 0$.
$k = 0$.
repeat
    Select $\boldsymbol{\eta}^k \in H(\boldsymbol{w}^k, b^k)$ arbitrarily.
    Update $t^k$ as
    $t^k \leftarrow \max\left\{0, \frac{\nu\varphi_{1-\nu}(\boldsymbol{w}^k, b^k) - \mu\varphi_{1-\mu}(\boldsymbol{w}^k, b^k)}{1-\epsilon_2}\right\}$.

    $(\boldsymbol{w}^{k+1}, b^{k+1})$ = a solution of subproblem (19).
    $k \leftarrow k+1$.
until $f(\boldsymbol{w}^k, b^k) - f(\boldsymbol{w}^{k+1}, b^{k+1}) < \epsilon_1$ where $f(\boldsymbol{w}, b) = \nu\varphi_{1-\nu}(\boldsymbol{w}, b) -$
    $\mu\varphi_{1-\mu}(\boldsymbol{w}, b)$.
```

where $C \geq 0$ and $\nu \in [0,1)$ are hyper-parameters. Following the case of classification, we formulate robust ν-SVR as

$$\begin{array}{ll} \min\limits_{w,b,\epsilon,\eta} & \frac{1}{2C}\|\boldsymbol{w}\|^2 + (\nu - \mu)\epsilon + \frac{1}{|I|}\sum_{i=1}^{m} \eta_i[|\boldsymbol{w}^\top \boldsymbol{x}_i + b - y_i| - \epsilon]_+ \\ \text{s.t.} & 0 \leq \eta_i \leq 1, \ \sum\limits_{i \in I}(1 - \eta_i) = \mu|I|, \end{array} \tag{21}$$

where $C \geq 0$, $\nu \in (\mu, 1]$ and $\mu \in [0,1)$ are hyper-parameters. Let us consider the distribution of

$$r_i^{reg}(\boldsymbol{w}, b) := |\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle - y_i|.$$

$\varphi_{1-\nu}^{reg}(\boldsymbol{w}, b)$ (resp. $\varphi_{1-\mu}^{reg}(\boldsymbol{w}, b)$) denotes $(1-\nu)$-CVaR (resp. $(1-\mu)$-CVaR) of the distribution. Robust ν-SVR (21) can be decomposed by the two CVaRs as

$$\min_{w,b} \ \frac{1}{2C}\|\boldsymbol{w}\|^2 + \nu\varphi_{1-\nu}^{reg}(\boldsymbol{w}, b) - \mu\varphi_{1-\mu}^{reg}(\boldsymbol{w}, b). \tag{22}$$

Since (22) is decomposed to the convex part and concave part, we can use simplified DCA for robust ν-SVR.

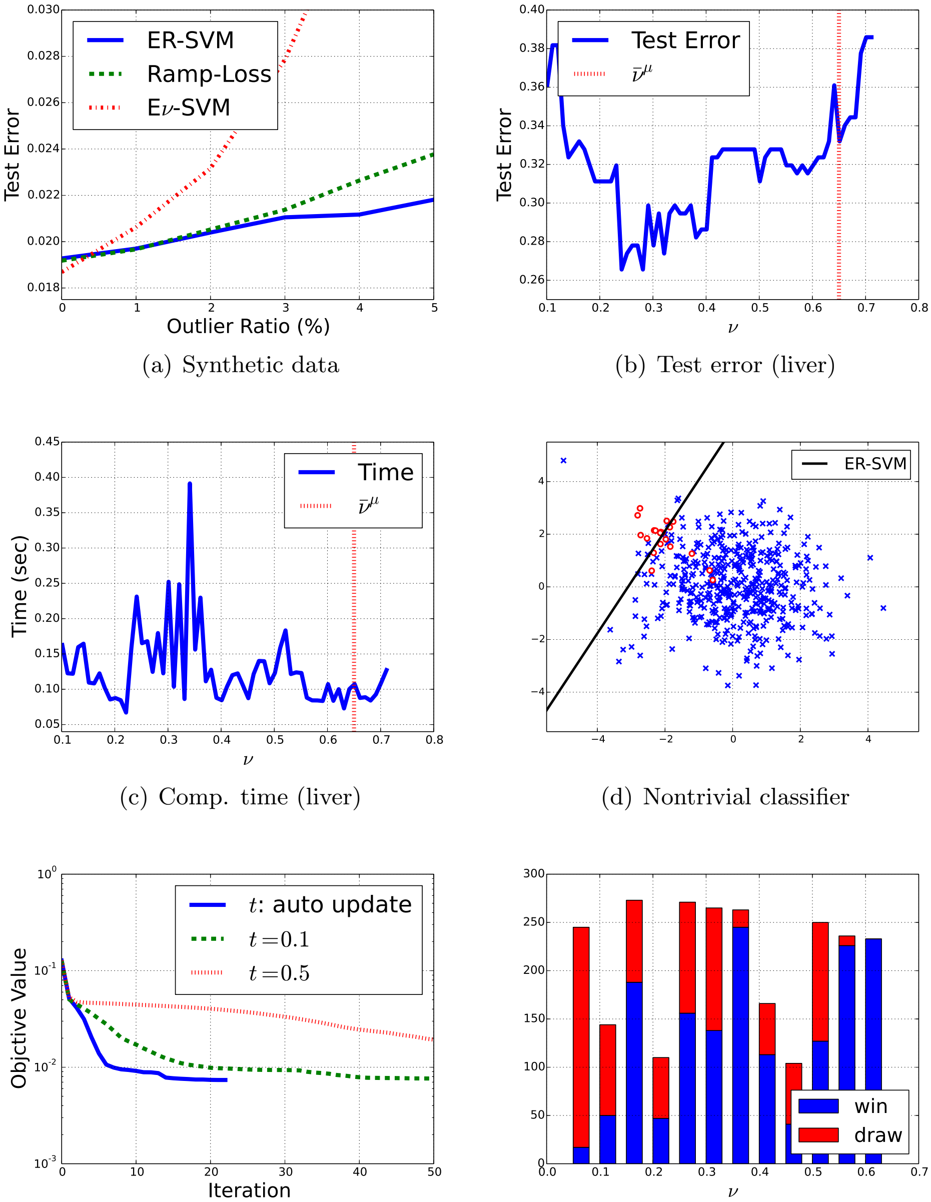
6 Numerical Results

We compared ER-SVM with ramp-loss SVM by CCCP [5] and Eν-SVM [10]. The hyper-parameter s of the ramp-loss function was fixed to 1, and μ in ER-SVM was fixed to 0.05.

6.1 Synthetic Datasets

We used synthetic data generated by following the procedure in [25]. We generated two-dimensional samples with labels $+1$ and -1 from two normal distributions with different mean vectors and the same covariance matrix. The optimal hyperplane for the noiseless dataset is $h(\boldsymbol{x}) = x_1 - x_2 = 0$ with $\boldsymbol{w} = \frac{1}{\sqrt{2}}(1, -1)$ and $b = 0$. We added outliers only to the training set with the label -1 by drawing samples uniformly from a half-ring with center $\mathbf{0}$, inner-radius $R = 75$ and outer-radius $R + 1$ in the space $\boldsymbol{x}$ of $h(\boldsymbol{x}) > 0$. The training set contains 50 samples from each class (i.e., 100 in total) including outliers. The ratio of outliers in the training set was set to a value from 0 to 5%. The test set has 1000 samples from each class (i.e., 2000 in total). We repeated the experiments 100 times, drawing training and test sets every repetition. We found the best parameter setting from 9 candidates, $\nu = 0.1, 0.2, \ldots, 0.9$ and $C = 10^{-4}, 10^{-3}, \ldots, 10^{4}$. Figure 4(a) shows the outlier ratio and the test error of each models. ER-SVM ($\mu = 0.05$) achieved good accuracy especially when the outlier ratio was large.

6.2 Real Datasets



(a) Synthetic data

(b) Test error (liver)

(c) Comp. time (liver)

(d) Nontrivial classifier

(e) Convergence

(f) Objective value

Figure 4: (a) shows the average errors for synthetic dataset. (b) is an example where ER-SVM achieved the minimum test error with $\nu \leq \underline{\nu}_{\mu}$ in the extended parameter range. (c) shows the computational time of Algorithm 1. (d) shows the example where ER-SVM obtains a non-trivial classifier, though C-SVM, ν-SVM and ramp-loss SVM obtain trivial classifiers $\boldsymbol{w} = \mathbf{0}$. (e) implies that our update rule of t_k as in (18) achieves much faster convergence. (f) shows how many times ER-SVM achieves smaller objective values than the heuristic algorithm in 300 trials.

Table 3: Average error of real datasets in 10 trials

Dataset	Dim	# train	# test	$(\underline{\nu}, \overline{\nu})$	Outlier Ratio	ER-SVM		Ramp	Eν-SVM	
Liver	6	345	10-cross	$(0.72, 0.84)$	0%	**0.270**	N	0.284	0.290	N
$(R = 5)$					1%	**0.284**	N	0.287	0.310	C
					3%	**0.278**	N	0.417	0.490	C
Diabetes	8	768	10-cross	$(0.52, 0.70)$	0%	0.227	N	**0.219**	0.224	N
$(R = 10)$					3%	**0.219**	N	0.232	0.238	C
					5%	**0.236**	N	0.289	0.288	N
Heart	13	270	10-cross	$(0.33, 0.89)$	0%	**0.156**	C	0.159	**0.156**	C
$(R = 50)$					3%	0.174	C	**0.170**	0.185	C
					5%	**0.167**	C	0.200	0.219	C
Splice	60	1000	10-cross	$(0.37, 0.97)$	0%	**0.186**	N	0.199	0.188	C
$(R = 100)$					5%	**0.188**	N	0.254	0.295	C
Adult	123	1605	30956	$(0.32, 0.49)$	0%	0.159	N	0.159	**0.158**	C
$(R = 150)$					5%	**0.157**	C	0.166	0.161	C
Vehicle (class 1 vs rest)	18	846	10-cross	$(0.42, 0.50)$	0%	**0.201**	N	0.204	**0.201**	N
$(R = 100)$					5%	**0.217**	N	0.242	0.371	C
Satimage (class 6 vs rest)	36	4435	2000	$(0.21, 0.47)$	0%	**0.102**	N	0.105	0.106	C
$(R = 150)$					3%	**0.100**	N	0.143	0.184	C

We used the datasets of the UCI repository [2] and LIBSVM [3]. We scaled all attributes of the original dataset from -1.0 to 1.0. We generated outliers $\hat{\boldsymbol{x}}$ uniformly from a ring with center $\mathbf{0}$ and radius R and assigned the wrong label $\hat{y}$ to $\hat{\boldsymbol{x}}$ by using the optimal classifiers of Eν-SVM. The radius R of generating outliers was set properly so that the outliers would have an impact on the test errors. The best parameter was chosen from 9 candidates, $\nu \in (0, \overline{\nu}]$ with equal intervals and $C = 5^{-4}, 5^{-3}, \ldots, 5^{4}$. These parameters are decided using 10-fold cross validation and the error is the average of 10 trials. Table 3 shows the results for real datasets. ER-SVM often achieved smaller test errors than ramp-loss SVM and Eν-SVM, and the prediction performance of ER-SVM were very stable to increasing the outlier ratio. 'N' in Table 3 implies that ER-SVM (or Eν-SVM) achieved the best accuracy with $\nu \leq \underline{\nu}_\mu$ (or $\nu \leq \underline{\nu}$) and 'C' implies that the best accuracy was achieved with $\underline{\nu}_\mu < \nu$ (or $\underline{\nu} < \nu$).

6.3 Effectiveness of the Extension

Let us show an example where the extension of parameter range works. We used 30% of the liver dataset for training and 70% for the test. We tried the hyper-parameter ν from 0.1 to the $\overline{\nu}_\mu$ with the interval 0.01 and μ is fixed to 0.03. The computational time is the average of 100 trials. Figure 4(b) and (c) show the test error and the computational time. The vertical line is an estimated value of $\underline{\nu}_\mu$. The extension for parameter range corresponds to the left side of the vertical line where the non-convex constraint $\|\boldsymbol{w}\|^2 = 1$ worked. ER-SVM with $\nu < \underline{\nu}_\mu$ can find classifiers which ROD or ramp-loss SVM can not find. In Figure 4(b), ER-SVM achieved the minimum test error with $\nu \leq \underline{\nu}_\mu$. In Figure 4(c) it seems that the computational time does not change so much though the non-convex constraint works in the left side of the vertical line. Note that the computational time becomes large around $\nu = 0.3$. The optimal margin variable ρ is zero around $\nu = 0.3$. It might make the problem difficult and numerically unstable.

Figure 4(d) shows an example of the effectiveness of the non-convex constraint $\|\boldsymbol{w}\|^2 = 1$. When the number of the samples in each class is imbalanced or the samples in two classes are largely overlapped as in Figure 4(d), C-SVM, ν-SVM, and ramp-loss SVM obtain trivial classifiers ($\boldsymbol{w} = \mathbf{0}$) while ER-SVM obtains a non-trivial classifier. That is, this figure implies the effectiveness of the non-convex constraint $\|\boldsymbol{w}\|^2 = 1$.

6.4 Efficient Update of t_k

We show the effectiveness of the auto update rule of the constant t_k as in (18). We used the liver dataset. Figure 4(e) implies that our auto update rule achieves much faster convergence than the fixed constant $t = 0.1$ or 0.5.

6.5 Comparison of DCA and Heuristics

We compared the performance of the heuristic algorithm [20] for ER-SVM and our new algorithm based on DCA. We ran the two algorithm on liver-disorder data set and computed the difference of the objective values. Initial values are selected from uniform random distribution on unit sphere and the experiments were repeated 300 times. Since the hyper-parameter μ is automatically selected in the heuristic algorithm, we set corresponding μ for DCA. To compare the quality of the solutions obtained by each algorithm, we counted how many times our algorithm (DCA) achieved smaller objective values than the heuristics. We counted 'win', 'lose', 'draw' cases in 300 trials. However it is difficult to judge whether a small difference in objective values is caused by numerical error or the difference of local solutions. Hence, we call 'win' (or 'lose') for the case where DCA achieves a smaller (or larger) objective value than the heuristic algorithm by more than 3% difference. Figure 4(f) shows the result. Our algorithm (DCA) tends to achieve 'win' or 'draw' in many cases. Some papers (e.g., [9, 11]) state that DCA tends to converge to a global solution. This result may support the discussion.

7 Conclusions

We gave theoretical analysis to ER-SVM: we proved that ER-SVM is a natural extension of ROD and discuss the condition under which such the extension works. Furthermore, we proposed a new efficient algorithm which has theoretically good properties. Numerical experiments showed that our algorithm worked efficiently.

We might be able to speed up the proposed algorithm by solving the dual of subproblems (19). The problem has the similar structure with the dual SVM and Sequential Minimal Optimization (SMO) might be applicable to the dual of subproblems.

References

[1] P. Artzner, F. Delbaen, J. Eber, and D. Heath. Coherent measures of risk. *Mathematical Finance*, 9:203–228, 1999.

[2] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998.

[3] C. C. Chang and C. J. Lin. Libsvm : A library for support vector machines. Technical report, Department of Computer Science, National Taiwan University, 2001.

[4] C.-C. Chang and C.-J. Lin. Training ν-support vector classifiers: Theory and algorithms. *Neural Computation*, 13(9):2119–2147, 2001.

[5] R. Collobert, F. Sinz, J. Weston, and L. Bottou. Trading convexity for scalability. In *International Conference on Machine Learning*, pages 129–136, 2006.

[6] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[7] D.J. Crisp and C.J.C. Burges. A geometric interpretation of ν-svm classifiers. In *Neural Information Processing Systems 12*, pages 244–250, 2000.

[8] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches.* Springer-Verlag, Berlin, 3rd edition, 1996.

[9] H. A. Le Thi and T. Pham Dinh. The dc (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems. *Annals of Operations Research*, 133(1-4):23–46, 2005.

[10] F. Perez-Cruz, J. Weston, D. J. L. Hermann, and B. Schölkopf. Extension of the ν-SVM range for classification. In *Advances in Learning Theory: Methods, Models and Applications 190*, pages 179–196, Amsterdam, 2003. IOS Press.

[11] T. Pham Dinh and H. A. Le Thi. Convex analysis approach to d.c. programming: Theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997.

[12] T. Phanm Dinh, H. A. Le Thi, and Le D. Muu. Exact penalty in d.c. programming. *Vietnam Journal of Mathematics*, 27(2):169–178, 1999.

[13] Ryan M. Rifkin, Massimiliano Pontil, and Alessandro Verri. A note on support vector machine degeneracy. In *Algorithmic Learning Theory*, pages 252–263, 1999.

[14] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, 1970.

[15] R. T. Rockafellar and S. Uryasev. Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance*, 26(7):1443–1472, 2002.

[16] B. Schölkopf, A. Smola, R. Williamson, and P. Bartlett. New support vector algorithms. *Neural Computation*, 12(5):1207–1245, 2000.

[17] Xiaotong Shen, George C Tseng, Xuegong Zhang, and Wing Hung Wong. On ψ-learning. *Journal of the American Statistical Association*, 98(463):724–734, 2003.

[18] A. J. Smola, S. V. N. Vishwanathan, and T. Hofmann. Kernel methods for missing variables. In *In Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, pages 325–332, 2005.

[19] Bharath K. Sriperumbudur and Gert R. G. Lanckriet. A proof of convergence of the concave-convex procedure using zangwill's theory. *Neural Computation*, 24(6):1391–1407, 2012.

[20] A. Takeda, S. Fujiwara, and T. Kanamori. Extended robust support vector machine based on financial risk minimization. *Neural Computation*, 26(11):2541–2569, 2014.

[21] A. Takeda and M. Sugiyama. ν-support vector machine as conditional value-at-risk minimization. In *International Conference on Machine Learning*, pages 1056–1063, 2008.

[22] Pham Dinh Tao. Duality in dc (difference of convex functions) optimization. Subgradient methods. In *Trends in Mathematical Optimization*, pages 277–293. 1988.

[23] P. Tsyurmasto, S. Uryasev, and J. Gotoh. Support vector classification with positive homogeneous risk functionals. Technical report, 2013.

[24] D. Wozabal. Value-at-risk optimization using the difference of convex algorithm. *OR Spectrum*, 34:861–883, 2012.

[25] L. Xu, K. Crammer, and D. Schuurmans. Robust support vector machine training via convex outlier ablation. In *AAAI*, pages 536–542. AAAI Press, 2006.

[26] Y. Yu, M. Yang, L. Xu, M. White, and D. Schuurmans. Relaxed clipping: A global training method for robust regression and classification. In *Neural Information Processing Systems*, pages 2532–2540. MIT Press, 2010.

[27] A. L. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15(4):915–936, 2003.

A Proof of Proposition 1

The objective function of ER-SVM (6) under the constraints can be equivalently rewritten as

$$\begin{aligned}&\zeta + \frac{1}{(\nu-\mu)|I|}\sum_{i\in I}\eta_i[r_i(\boldsymbol{w},b)-\zeta]^+\\
=&\frac{1}{\nu-\mu}\left\{(\nu-\mu)\zeta + \frac{1}{|I|}\sum_{i\in I}\eta_i[r_i(\boldsymbol{w},b)-\zeta]^+\right\}\\
=&\frac{1}{\nu-\mu}\left\{(\nu-\mu)\zeta + \frac{1}{|I|}\sum_{i\in I}[r_i(\boldsymbol{w},b)-\zeta]^+ - \frac{1}{|I|}\sum_{i\in I}(1-\eta_i)[r_i(\boldsymbol{w},b)-\zeta]^+\right\}\\
=&\frac{1}{\nu-\mu}\left\{\nu\zeta + \frac{1}{|I|}\sum_{i\in I}[r_i(\boldsymbol{w},b)-\zeta]^+ - \frac{1}{|I|}\sum_{i\in I}(1-\eta_i)\{[r_i(\boldsymbol{w},b)-\zeta]^+ + \zeta\}\right\}.\end{aligned}$$

The last equality is obtained by using a constraint, $\sum_{i\in I}(1-\eta_i)=\mu|I|$, of ER-SVM (6).

Now we show that the term in the last equation

$$\frac{1}{|I|}\sum_{i\in I}(1-\eta_i)\{[r_i(\boldsymbol{w},b)-\zeta]^+ + \zeta\}$$

can be written as $\frac{1}{|I|}\sum_{i\in I}(1-\eta_i)r_i(\boldsymbol{w},b)$ by showing that $r_i(\boldsymbol{w}^*,b^*)-\zeta^*\geq 0$ holds for any i whenever $1-\eta_i^*>0$ at the optimal solution $(\boldsymbol{w}^*,b^*,\zeta^*,\boldsymbol{\eta}^*)$ of (6). Here we assume that $r_i(\boldsymbol{w}^*,b^*)$, $\forall i$, are sorted into descending order; $r_1(\boldsymbol{w}^*,b^*)\geq r_2(\boldsymbol{w}^*,b^*)\geq\ldots$. Then $\boldsymbol{\eta}^*$ should be

$$\eta_1^*=\ldots=\eta^*_{\lfloor\mu|I|\rfloor}=0, \eta^*_{\lfloor\mu|I|\rfloor+1}>0, \eta^*_{\lfloor\mu|I|\rfloor+2}=\ldots 1.$$

Note that ζ^* must be an optimal solution of the problem:

$$\begin{aligned}\min_{\zeta}\quad & \zeta+\frac{1}{(\nu-\mu)|I|}\sum_{i=\lfloor\mu|I|\rfloor+1}^{|I|}\eta_i^*[r_i(\boldsymbol{w}^*,b^*)-\zeta]^+\\ \text{s.t.}\quad & \|\boldsymbol{w}\|^2=1.\end{aligned}$$

The problem is regarded as minimizing $(1-\alpha)$-CVaR, where $\alpha:=\frac{\nu-\mu}{1-\mu}$ (>0), for the truncated distribution with $\lfloor\mu|I|\rfloor$ samples removed. Theorem 1 ensures that

$$r_{\lfloor\mu|I|\rfloor+1}(\boldsymbol{w}^*,b^*)\geq\zeta^+_{1-\alpha}(\boldsymbol{w}^*,b^*)\geq\zeta^*,$$

which implies that $r_i(\boldsymbol{w}^*,b^*)-\zeta^*\geq 0$ holds for the indices having $\eta_i<1$.

Therefore, we can rewrite the objective function of ER-SVM (6) by

$$\frac{1}{\nu-\mu}\left\{\nu\zeta+\frac{1}{|I|}\sum_{i\in I}[r_i(\boldsymbol{w},b)-\zeta]^+-\frac{1}{|I|}\sum_{i\in I}(1-\eta_i)r_i(\boldsymbol{w},b)\right\}. \tag{23}$$

By using (7) for the first two terms of (23) and using (8) for the last term, we further rewrite (23) as

$$\frac{1}{\nu-\mu}\{\nu\varphi_{1-\nu}(\boldsymbol{w},b)-\mu\varphi_{1-\mu}(\boldsymbol{w},b)\},$$

which is the objective function of (9). This implies that ER-SVM (6) is described as the form of (9).

B KKT Conditions for ER-SVM, ROD, and Ramp-Loss SVM

To define the KKT conditions, we show differentiable formulations of Case C of ER-SVM (6), ROD (5), and Ramp-Loss SVM (4).
Continuous Ramp-Loss SVM:

$$\begin{aligned} \min_{w,b,\eta,\xi} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i\in I}\{\eta_i\xi_i + s(1-\eta_i)\} \\ \text{s.t.} \quad & 0 \le \eta_i \le 1, \\ & \xi_i \ge 1 + r_i(\boldsymbol{w},b),\ \xi_i \ge 0 \end{aligned} \tag{24}$$

Continuous ROD:

$$\begin{aligned} \min_{w,b,\eta,\xi} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i\in I}\eta_i\xi_i \\ \text{s.t.} \quad & 0 \le \eta_i \le 1,\ \sum_{i\in I}(1-\eta_i) = \mu|I|, \\ & \xi_i \ge 1 + r_i(\boldsymbol{w},b),\ \xi_i \ge 0 \end{aligned} \tag{25}$$

Continuous ER-SVM (limited to Case C in Table 2.2):

$$\begin{aligned} \min_{w,b,\rho,\eta,\xi} \quad & -\rho + \frac{1}{(\nu-\mu)|I|}\sum_{i\in I}\eta_i\xi_i \\ \text{s.t.} \quad & 0 \le \eta_i \le 1, \sum_{i\in I}(1-\eta_i) = \mu|I| \\ & \xi_i \ge \rho + r_i(\boldsymbol{w},b), \xi_i \ge 0, \|\boldsymbol{w}\|^2 \le 1 \end{aligned} \tag{26}$$

The KKT conditions of the above problems are defined as follows.

KKT Conditions of (24) **using** $(\boldsymbol{w}, b, \boldsymbol{\eta}, \boldsymbol{\xi}; \boldsymbol{\lambda}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$**:**

$$\sum_{i \in I} \lambda_i y_i = 0, \tag{27a}$$

$$\gamma_i \xi_i = 0, \tag{27b}$$

$$\alpha_i(\eta_i - 1) = 0, \tag{27c}$$

$$\beta_i \eta_i = 0, \tag{27d}$$

$$\lambda_i, \alpha_i, \beta_i, \gamma_i, \geq 0, \tag{27e}$$

$$0 \leq \eta_i \leq 1, \tag{27f}$$

$$\xi_i \geq 0, \tag{27g}$$

$$\lambda_i\{1 - \xi_i + r_i(\boldsymbol{w}, b)\} = 0, \tag{28a}$$

$$-r_i(\boldsymbol{w}, b) \geq 1 - \xi_i, \tag{28b}$$

$$\boldsymbol{w} = \sum_{i \in I} \lambda_i y_i \boldsymbol{x}_i, \tag{28c}$$

$$C\eta_i - \lambda_i - \gamma_i = 0, \tag{28d}$$

and

$$C\xi_i - Cs + \alpha_i - \beta_i = 0 \tag{29}$$

KKT Conditions of (25) **using** $(\boldsymbol{w}, b, \boldsymbol{\eta}, \boldsymbol{\xi}; \boldsymbol{\lambda}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \tau)$**:**
(27), (28),

$$C\xi_i - \tau + \alpha_i - \beta_i = 0, \tag{30}$$

and

$$\sum_{i \in I}(1 - \eta_i) = \mu|I| \tag{31}$$

KKT Conditions of (26) **using** $(\boldsymbol{w}, b, \rho, \boldsymbol{\eta}, \boldsymbol{\xi}; \boldsymbol{\lambda}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \tau, \delta)$**:**

(27), (31), and

$$\lambda_i\{\rho - \xi_i + r_i(\boldsymbol{w}, b)\} = 0, \tag{32a}$$

$$-r_i(\boldsymbol{w}, b) \geq \rho - \xi_i, \tag{32b}$$

$$2\delta\boldsymbol{w} = \sum_{i \in I} \lambda_i y_i \boldsymbol{x}_i, \tag{32c}$$

$$\frac{\eta_i}{|I|} - \lambda_i - \gamma_i = 0, \tag{32d}$$

$$\frac{1}{|I|}\xi_i - \tau + \alpha_i - \beta_i = 0, \tag{32e}$$

$$\sum_{i \in I} \lambda_i = \nu - \mu, \tag{32f}$$

$$\delta(\|\boldsymbol{w}\|^2 - 1) = 0, \tag{32g}$$

$$\delta \geq 0 \tag{32h}$$

B.1 Proof of Lemma 1

The difference between the KKT conditions of Ramp-Loss SVM (24) and ROD (25) is only (29), (30), and (31).

Note that a KKT point $(\boldsymbol{w}^*, b^*, \boldsymbol{\eta}^*)$ of Ramp-Loss SVM (24) satisfies the KKT conditions of ROD (25) with $\mu = \frac{1}{|I|}\sum_{i \in I}(1 - \eta_i^*)$. On the other hand, a KKT point of (25) whose Lagrange multiplier for $\sum_{i \in I}(1 - \eta_i) = \mu|I|$ is τ^* satisfies the KKT conditions of Ramp-Loss SVM (24) with $s = \frac{\tau^*}{C}$.

B.2 Proof of Theorem 2

We will show the first statement. Let $(\boldsymbol{w}^*, b^*, \boldsymbol{\eta}^*, \boldsymbol{\xi}^*)$ be a KKT point of (25) with hyper-parameter C and Lagrange multipliers $(\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \tau^*)$. Suppose $\boldsymbol{w}^* \neq \boldsymbol{0}$. Then $(\frac{\boldsymbol{w}^*}{\|\boldsymbol{w}^*\|}, \frac{b^*}{\|\boldsymbol{w}^*\|}, \boldsymbol{\eta}^*, \frac{\boldsymbol{\xi}^*}{\|\boldsymbol{w}^*\|}, \rho^* = \frac{1}{\|\boldsymbol{w}^*\|})$ is a KKT point of (26) with Lagrange multipliers $\frac{1}{C|I|}(\boldsymbol{\lambda}^*, \frac{\boldsymbol{\alpha}^*}{\|\boldsymbol{w}^*\|}, \frac{\boldsymbol{\beta}^*}{\|\boldsymbol{w}^*\|}, \boldsymbol{\gamma}^*, \frac{\tau^*}{\|\boldsymbol{w}^*\|})$ and $\delta = \frac{\|\boldsymbol{w}^*\|}{2C|I|}$.

We will show the second statement. Let $(\boldsymbol{w}^*, b^*, \boldsymbol{\eta}^*, \boldsymbol{\xi}^*, \rho^*)$ be a KKT point of (26) with Lagrange multipliers $(\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*, \boldsymbol{\gamma}^*, \tau^*, \delta^*)$. Suppose $\rho^* \neq 0$. If $\delta^* \neq 0$, $(\frac{\boldsymbol{w}^*}{\rho^*}, \frac{b^*}{\rho^*}, \boldsymbol{\eta}^*, \frac{\boldsymbol{\xi}^*}{\rho^*})$ satisfies the KKT conditions of (25) with Lagrange multipliers $\frac{1}{2\delta^*}(\frac{\boldsymbol{\lambda}^*}{\rho^*}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*, \frac{\boldsymbol{\gamma}^*}{\rho^*}, \tau^*)$ and hyper-parameter $C = \frac{1}{\delta^*\rho^*|I|}$. Now, we will prove $\delta^* \neq 0$ using the assumption of non-zero optimal value. Consider the following problem which fixes $\boldsymbol{\eta}$ of (26) to $\boldsymbol{\eta}^*$:

$$\begin{array}{ll} \min\limits_{w,b,\rho,\xi} & -\rho + \dfrac{1}{(\nu - \mu)|I|}\displaystyle\sum_{i \in I} \eta_i^* \xi_i \\ \text{s.t.} & \|\boldsymbol{w}\|^2 \leq 1, -r_i(\boldsymbol{w}, b) \geq \rho - \xi_i, \xi_i \geq 0. \end{array} \tag{33}$$

The KKT conditions of (33) are as follows.

$$\begin{aligned} 2\delta \boldsymbol{w} = \sum_{i\in I} \lambda_i y_i \boldsymbol{x}_i, \sum_{i\in I} \lambda_i y_i = 0, \frac{\eta_i}{|I|} - \lambda_i - \gamma_i = 0,\\ \sum_{i\in I} \lambda_i = \nu - \mu,\\ \lambda_i\{\rho - \xi_i + r_i(\boldsymbol{w}, b)\}, \gamma_i \xi_i = 0, \delta(\|\boldsymbol{w}\|^2 - 1) = 0,\\ \delta, \lambda_i, \gamma_i, \geq 0,\\ -r_i(\boldsymbol{w}, b) \geq \rho - \xi_i, \xi_i \geq 0. \end{aligned}$$

$(\boldsymbol{w}^*, b^*, \boldsymbol{\xi}^*, \rho^*)$ is also a KKT point of (33) with the same Lagrange multipliers $(\boldsymbol{\lambda}^*, \boldsymbol{\gamma}^*, \delta^*)$ as (26). Moreover, $(\boldsymbol{w}^*, b^*, \boldsymbol{\xi}^*, \rho^*)$ is not only a KKT point but also an optimal solution of (33) because (33) is a convex problem. Since the objective function of the dual problem of (33) is $-\delta(\|\boldsymbol{w}\|^2+1)$, $\delta^* = 0$ if and only if the objective value is zero. Then, we can see that $\delta^* \neq 0$ under the assumption of non-zero objective value.

C Proof of Theorem 3

Consider the dual problems of ν-SVM and C-SVM:

$$\begin{aligned} \min_{\lambda} \quad & \frac{1}{2}\left\|\sum_{i\in I} \lambda_i y_i \boldsymbol{x}_i\right\|^2 \\ \text{s.t.} \quad & 0 \leq \lambda_i \leq \frac{1}{|I|}, \sum_{i\in I} y_i \lambda_i = 0, \sum_{i\in I} \lambda_i = \nu, \end{aligned} \qquad (D_\nu)$$

$$\begin{aligned} \min_{\lambda} \quad & \frac{1}{2}\left\|\sum_{i\in I} \lambda_i y_i \boldsymbol{x}_i\right\|^2 - \sum_{i\in I} \lambda_i \\ \text{s.t.} \quad & 0 \leq \lambda_i \leq \frac{1}{|I|}, \sum_{i\in I} y_i \lambda_i = 0. \end{aligned} \qquad (D_C')$$

Let us describe the optimal $\boldsymbol{\lambda}$ of (D_ν) and (D_C') as $\boldsymbol{\lambda}^{(\nu)}$ and $\boldsymbol{\lambda}^{(C)}$ respectively. Note that the optimal $\boldsymbol{w}$ of (D_ν) and (D_C') are represented as $\sum_{i\in I} \lambda_i^{(\nu)} y_i \boldsymbol{x}_i$ and $\sum_{i\in I} \lambda_i^{(C)} y_i \boldsymbol{x}_i$, respectively, with the use of the KKT conditions of ν-SVM and C-SVM. Then $\boldsymbol{w} = \boldsymbol{0}$ if and only if $\sum_{i\in I} \lambda_i y_i \boldsymbol{x}_i = \boldsymbol{0}$ for the optimal solutions of (D_ν) and (D_C').

When $0 < |I_-| \leq |I_+|$, $\overline{\nu} = 2|I_-|/|I|$. Then $RCH_-(\overline{\nu}) = \sum_{i\in I_-} \frac{1}{|I_-|}\boldsymbol{x}_i$ holds. Here, we will show the following statements for a training set are equivalent.

(c1) a training set $\{\boldsymbol{x}_i, y_i\}_{i\in I}$ satisfies $RCH_-(\overline{\nu}) \in RCH_+(\overline{\nu})$,

(c2) (D_ν) has an optimal solution such that $\sum_{i\in I} \lambda_i y_i \boldsymbol{x}_i = \boldsymbol{0}$ for all $\nu \in (\underline{\nu}, \overline{\nu}]$, (c3) (D'_C) has an optimal solution such that $\sum_{i \in I} \lambda_i y_i \boldsymbol{x}_i = \mathbf{0}$ for all $C \in (0, \infty)$

(c2) and (c3) imply that ν-SVM and C-SVM obtain a trivial solution such that $\boldsymbol{w} = \sum_{i \in I} \lambda_i y_i \boldsymbol{x}_i = \mathbf{0}$ for any hyper-parameter value.

The equivalence of (c1) and (c2) is shown by the geometric interpretation of ν-SVM. From the result of [7], (D_ν) is described as

$$\min_{x_+ \in RCH_+(\nu),\ x_- \in RCH_-(\nu)} \quad \|\boldsymbol{x}_+ - \boldsymbol{x}_-\|^2. \tag{34}$$

By appropriate scaling: $\tilde{\lambda}_i = 2\lambda_i/\nu$, (D_ν) and (34) has the same set of optimal solutions. We denote the optimal solution of (34) as $\boldsymbol{x}^*_+$ and $\boldsymbol{x}^*_-$. $\boldsymbol{x}^*_\pm$ is represented, using the optimal $\tilde{\lambda}^*$, as $\boldsymbol{x}^*_\pm = \sum_{i \in I_\pm} \tilde{\lambda}^*_i \boldsymbol{x}_i$. Then $\boldsymbol{x}^*_+ = \boldsymbol{x}^*_-$ if and only if $\sum_{i \in I} \tilde{\lambda}^*_i y_i \boldsymbol{x}_i = \mathbf{0}$.

(c1) $\Rightarrow$ (c2): If a training set $\{\boldsymbol{x}_i, y_i\}_{i \in I}$ satisfies $RCH_-(\overline{\nu}) \in RCH_+(\overline{\nu})$, then $RCH_+(\nu) \cap RCH_-(\nu) \neq \emptyset$ for all $\nu \in (\underline{\nu}, \overline{\nu}]$. Therefore, (c2) holds.

(c2) $\Rightarrow$ (c1): If (D_ν) has an optimal solution $\boldsymbol{\lambda}^{(\nu)}$ such that $\sum_{i \in I} \lambda^{(\nu)}_i y_i \boldsymbol{x}_i = \mathbf{0}$ for all $\nu \in (\underline{\nu}, \overline{\nu}]$, then $\boldsymbol{x}^*_+ - \boldsymbol{x}^*_- = \mathbf{0}$ for all $\nu \in (\underline{\nu}, \overline{\nu}]$. That is, $RCH_+(\nu) \cap RCH_-(\nu) \neq \emptyset$ for all $\nu \in (\underline{\nu}, \overline{\nu}]$. Therefore, (c1) holds.

The equivalence of (c2) and (c3) is shown using the result of [4].

(c2) $\Rightarrow$ (c3): We show it using contraposition. Suppose the optimal solution $\boldsymbol{\lambda}^{(C)}$ of (D'_C) satisfy $\sum_{i \in I} \lambda^{(C)}_i y_i \boldsymbol{x}_i \neq \mathbf{0}$ with some hyper parameter $C \in (0, \infty)$. Then, $0 < \sum_{i \in I} \lambda^{(C)}_i$ since the optimal value of (D_ν) is zero or negative. From [4, Theorem 3], $\boldsymbol{\lambda}^{(C)}$ is also an optimal solution of (D_ν) with $\nu = \sum_{i \in I} \lambda^{(C)}_i$.

(c3) $\Rightarrow$ (c2): We show it using contraposition. Suppose the optimal solution $\boldsymbol{\lambda}^{(\nu)}$ of (D_ν) satisfy $\sum_{i \in I} \lambda^{(\nu)}_i y_i \boldsymbol{x}_i \neq \mathbf{0}$ with some hyper parameter $\nu \in (0, 1]$. Then the optimal value of (D_ν) is positive. From [4, Theorem 4], (D_ν)'s optimal solution set is the same as that of at least one (D'_C) if the optimal value of (D_ν) is positive.