DC Algorithm for Extended Robust Support Vector Machine

Shuhei Fujiwara
shuhei.fujiwara@gmail.com
TOPGATE Co., Bunkyo-ku, Tokyo, 113-0033, Japan

Akiko Takeda
atakeda@ism.ac.jp
Department of Mathematical Analysis and Statistical Inference, Institute of Statistical Mathematics, Tachikawa, Tokyo 190-8562, Japan; and RIKEN Center for Advanced Intelligence Project, 1-4-1, Nihonbashi, Chuo-ku, Tokyo 103-0027, Japan

Takafumi Kanamori
kanamori@is.nagoya-u.ac.jp
Department of Computer Science and Mathematical Informatics, Nagoya University, Chikusa-ku, Nagoya, Aichi 464-8601, Japan; and RIKEN Center for Advanced Intelligence Project, 1-4-1, Nihonbashi, Chuo-ku, Tokyo 103-0027, Japan

Nonconvex variants of support vector machines (SVMs) have been developed for various purposes. For example, robust SVMs attain robustness to outliers by using a nonconvex loss function, while extended ν-SVM (Eν-SVM) extends the range of the hyperparameter by introducing a nonconvex constraint. Here, we consider an extended robust support vector machine (ER-SVM), a robust variant of Eν-SVM. ER-SVM combines two types of nonconvexity from robust SVMs and Eν-SVM. Because of the two nonconvexities, the existing algorithm we proposed needs to be divided into two parts depending on whether the hyperparameter value is in the extended range or not. The algorithm also heuristically solves the nonconvex problem in the extended range.

In this letter, we propose a new, efficient algorithm for ER-SVM. The algorithm deals with two types of nonconvexity while never entailing more computations than either Eν-SVM or robust SVM, and it finds a critical point of ER-SVM. Furthermore, we show that ER-SVM includes the existing robust SVMs as special cases. Numerical experiments confirm the effectiveness of integrating the two nonconvexities.

1 Introduction

The support vector machine (SVM), one of the most successful machine learning models, has many variants. The original form of SVM, C-SVM (Cortes & Vapnik, 1995), is popular because of its generalization ability.
and convexity. An alternative SVM formulation, $\nu$-SVM (Schölkopf, Smola, Williamson, & Bartlett, 2000), is known to be equivalent to C-SVM, and extended $\nu$-SVM (Ev-SVM) (Perez-Cruz, Weston, Hermann, & Schölkopf, 2003) is a nonconvex extension of $\nu$-SVM. Ev-SVM introduces a nonconvex norm constraint instead of a regularization term in the objective function, and this nonconvex constraint makes it possible to extend the range of the hyperparameter $\nu$. Ev-SVM includes $\nu$-SVM as a special case, and Ev-SVM empirically outperforms $\nu$-SVM owing to the extension (see Perez-Cruz et al., 2003). Furthermore, Takeda and Sugiyama (2008) showed that Ev-SVM minimizes the conditional value-at-risk (CVaR), a popular coherent risk measure in finance. However, CVaR is sensitive to tail risks, and the same holds true for Ev-SVM. Unfortunately, it also implies that SVMs might not be sufficiently robust to outliers.

Various nonconvex SVMs have been studied with the goal of ensuring robustness to outliers. Indeed, there are many such models, which are called robust SVMs. In this letter, “robust SVMs” means any robust variant of SVM that uses a nonconvex loss function.\(^1\) Ramp-loss SVM (Collobert, Sinz, Weston, & Bottou, 2006; Xu, Crammer, & Schuurmans, 2006) is a popular robust SVM. The idea is to truncate the hinge loss and bound the value of the loss function by a constant. Moreover, any loss function, not only the hinge loss, can be truncated in the same way as the ramp loss. The framework of such truncated loss functions is studied, for example, in Shen, Tseng, Zhang, and Wong (2003) and Yu, Yang, Xu, White, and Schuurmans (2010).

Xu et al. (2006) also proposed robust outlier detection (ROD) as a model explicitly identifying outliers. CVaR-$(\alpha_l, \alpha_u)$-SVM (Tsyurmasto, Uryasev, & Gotoh, 2013) has also been proposed as a robust SVM. However, we can prove that ROD and CVaR-$(\alpha_l, \alpha_u)$-SVM are equivalent to ramp-loss SVM with appropriately set parameters. On the other hand, Takeda, Fujiwara, and Kanamori (2014) recently proposed extended robust SVM (ER-SVM) as a robust variant of Ev-SVM, a nonconvex extension of $\nu$-SVM. That is, while the robust SVMs are robust variants of C-SVM or $\nu$-SVM, extended robust SVM (ER-SVM) is a robust variant of Ev-SVM, as indicated in Table 1.

1.1 Nonconvex Optimization and DC Programming. The important issue regarding nonconvex variants is how to solve their difficult nonconvex problems. Difference of convex functions (DC) programming is a powerful framework for dealing with nonconvex problems. It is known that various nonconvex problems can be formulated as DC programs whose objective function is expressed as $u(\cdot) - v(\cdot)$ by using two convex functions $u$ and $v$ (see Horst & Tuy, 1996).

---

\(^1\)Robust SVM in Xu, Caramanis, and Mannor (2009) denotes the model formed by considering the data input to be uncertain and to belong to some fixed uncertainty set such as an ellipsoid. The robust SVMs dealt with in this letter are different from those in Xu et al. (2009).
Table 1: Relationships among Existing Models.

<table>
<thead>
<tr>
<th>Regularizer</th>
<th>Convex</th>
<th>Nonconvex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss</td>
<td>C-SVM, ( \nu )-SVM</td>
<td>( E\nu )-SVM</td>
</tr>
<tr>
<td></td>
<td>Robust outlier detection</td>
<td>ER-SVM</td>
</tr>
<tr>
<td></td>
<td>ramp-loss SVM</td>
<td>CVaR-(( \alpha_L, \alpha_U ))-SVM</td>
</tr>
</tbody>
</table>

Note: The models in the right (resp. bottom) cell include the models in the left (resp. top) cell as special cases.

The DC algorithm (DCA) introduced in Pham Dinh (1988) is one of the most efficient algorithms for DC programs. The basic idea behind it 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 studied using convex analysis (Rockafellar, 1970). For a general DC program, every limit point of the sequence generated by DCA is a critical point, which is also called a generalized Karush-Kuhn-Tucker (KKT) point. It is remarkable that DCA does not require differentiability in order to ensure its convergence. Furthermore, it is known that DCA converges quite often to a global solution (see Le Thi & Pham Dinh, 2005; Pham Dinh & Le Thi, 1997).

A similar method, the concave-convex procedure (CCCP) (Yuille & Rangarajan, 2003), has been studied in the machine learning literature. Indeed, it can be shown that if \( v(\cdot) \) of the objective function is differentiable, then DCA exactly reduces to CCCP. Smola, Vishwanathan, and Hofmann (2005) proposed constrained CCCP to deal with DC constraints, while Sriperumbudur and Lanckriet (2012) studied the global convergence properties of (constrained) CCCP, proving that the sequence generated by CCCP converges to a stationary point under conditions such as differentiability and strict convexity. However, since our model is not differentiable, we will use DCA and take advantage of the theoretical results on DCA such as on its convergence properties.

1.2 Contributions. The main contribution of this letter is a new, efficient algorithm based on DCA for ER-SVM. ER-SVM has two nonconvexities: the nonconvexity in the objective function is due to the truncated loss function and the nonconvexity in the constraint is due to the extension of the parameter range \( \nu \). We express the truncated loss function of ER-SVM as the difference of two CVaRs, which are convex functions, and move the nonconvex term in its constraint to the objective by using an exact penalty. While being equivalent to ER-SVM, the resulting formulation allows us to apply DCA to ER-SVM and gives an intuitive interpretation for ER-SVM.
The previous algorithm proposed by Takeda et al. (2014) is heuristic and does not have a theoretical guarantee. Furthermore, it is not simple because it needs to use two different procedures depending on the value of the hyperparameter \( \nu \). Our algorithm instead works with any value of \( \nu \), and it can find a critical point of ER-SVM. Though ER-SVM enjoys both of the nonconvexities from \( E_\nu \)-SVM and robust SVMs (e.g., ROD and ramp-loss SVM), our new algorithm is simple and comparable to their algorithms. Indeed, our algorithm is similar to the algorithm (Collobert et al., 2006) of ramp-loss SVM, which is known to be fast. Our code for solving ER-SVM by DCA is publicly available at Fujiwara, Takeda, and Kanamori (2016).

Furthermore, we clarify the relation of various robust SVMs, including ER-SVM. Though there are many robust variants of SVMs, their relations to each other have not been discussed, except that Takeda et al. (2014) showed that ER-SVM and ramp-loss SVM share the same global optimal solutions. In this letter, we prove that ROD and CVaR-(\( \alpha_L, \alpha_U \))-SVM are equivalent to ramp-loss SVM and also show that ER-SVM includes ramp-loss SVM and ROD as special cases in the sense of having the same KKT points. More specifically, a special case of ER-SVM (whose range of the hyperparameter \( \nu \) is limited), ramp-loss SVM, and ROD share all KKT points. Therefore, as Table 1 shows, ER-SVM can be regarded not simply as a robust variant of \( E_\nu \)-SVM, but rather as a natural extension of ramp-loss SVM and ROD.

1.3 Outline of the Letter. This letter is organized as follows. Section 2 provides a preliminary description of the basic notions. In sections 2.1 and 2.2, we describe the existing SVMs and their variants. Section 2.3 briefly describes the definitions and properties of popular financial risk measures such as CVaR and VaR. Section 3 describes some important properties of ER-SVM. After showing a DC programming formulation of ER-SVM, which is a minimization of the difference of CVaRs, we discuss the relationship of ramp-loss SVM, ROD, and ER-SVM. Section 4 describes our new algorithm, DCA for ER-SVM. Numerical results are presented in section 5.

2 Preliminary

Here, we briefly review the binary classification of supervised learning. Suppose we have a set of training samples \( \{(x_i, y_i)\}_{i \in I} \), where \( x_i \in \mathbb{R}^n \), \( y_i \in \{-1, 1\} \) and \( I \) is the set of indices of training samples. \( I_+ \) (or \( I_- \)) is the index set for the label +1 (or −1, respectively), and we suppose \( |I_\pm| > 0 \), where \( |\cdot| \) denotes the size of the set. SVM learns the function \( h(x) := \langle w, x \rangle + b \) and predicts the label of \( x \) as \( \hat{y} = \text{sign}(h(x)) \). We define

\[
r_i(w, b) := -y_i(\langle w, x_i \rangle + b),
\]
wherein the absolute value of $r_i(w, b)$ is proportional to the distance from the hyperplane $(w, x) + b = 0$ to the sample $x_i$. $r_i(w, b)$ becomes negative if the sample $x_i$ is classified correctly and is positive otherwise.

Our algorithm can be applied to nonlinear models by using a kernel technique. We describe a kernel variant of ER-SVM in section 3.4.

2.1 Support Vector Machines.

2.1.1 Convex SVMs. C-SVM (Cortes & Vapnik, 1995), the most standard form of SVM, minimizes the hinge loss and regularizer;

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

where $[a]^+ := \max\{0, a\}$ and $C > 0$ is a hyperparameter. $\nu$-SVM (Schölkopf et al., 2000) is formulated as

$$
\min_{w, b, \rho} \frac{1}{2} \|w\|^2 - \nu \rho + \frac{1}{|I|} \sum_{i \in I} [\rho + r_i(w, b)]^+
\quad \text{s.t.} \quad \rho \geq 0, \tag{2.1}
$$

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

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

and a lower threshold, $\underline{\nu}$. The optimal solution is trivial ($w = 0$) if $\nu \leq \underline{\nu}$ and the optimal value is unbounded if $\nu > \overline{\nu}$ (see Chang & Lin, 2001b). Therefore, we define the range of $\nu$ for $\nu$-SVM as $(\underline{\nu}, \overline{\nu}]$.

2.1.2 Nonconvex SVMs. Here, we introduce two types of nonconvexity for SVMs. The first is extended $\nu$-SVM (Ev-SVM) (Perez-Cruz et al., 2003), an extended model of $\nu$-SVM. Ev-SVM is formulated as

$$
\min_{w, b, \rho} -\nu \rho + \frac{1}{|I|} \sum_{i \in I} [\rho + r_i(w, b)]^+
\quad \text{s.t.} \quad \|w\|^2 = 1. \tag{2.2}
$$

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

The second is ramp-loss SVM, 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 the ramp loss is to clip large losses with a hyperparameter \( s \geq 0 \). Ramp-loss SVM is formulated as

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

\( C \in (0, \infty) \) is also a hyperparameter. The ramp loss can be described as the difference between hinge-loss functions; therefore, the concave-convex procedure (CCCP), an effective algorithm of DC programming, can be applied to the problem (see Collobert et al., 2006, for details). Xu et al. (2006) gave another representation of ramp-loss SVM by using the \( \eta \)-hinge loss;

\[
\min_{w, b, \eta} \frac{1}{2} \| w \|^2 + C \sum_{i \in I} \eta_i [1 + r_i(w, b)]^+ + s \sum_{i \in I} (1 - \eta_i)
\]

s.t. \( 0 \leq \eta_i \leq 1 \),

\( |I| \) is the cardinality of \( I \).

\[
(2.4)
\]

and applied semidefinite programming (SDP) relaxation to equation 2.4. To explicitly identify outliers, Xu et al. (2006) proposed robust outlier detection (ROD):

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

s.t. \( 0 \leq \eta_i \leq 1 \), \( \sum_{i \in I} (1 - \eta_i) = \mu |I| \).

\( C \in (0, \infty) \) and \( \mu \in [0, 1] \) are hyperparameters. The original formulation in Xu et al. (2006) defines ROD with an inequality constraint \( \sum_{i \in I} (1 - \eta_i) \leq \mu |I| \), but we can replace it by the equality one because the replacement does not change the optimal value.

### 2.2 Extended Robust Support Vector Machine

Recently, Takeda et al. (2014) proposed extended robust SVM (ER-SVM):

\[
\min_{w, b, \rho, \eta} -\rho + \frac{1}{(\nu - \mu)|I|} \sum_{i \in I} \eta_i [\rho + r_i(w, b)]^+
\]

s.t. \( 0 \leq \eta_i \leq 1 \), \( \sum_{i \in I} (1 - \eta_i) = \mu |I| \), \( \| w \|^2 = 1 \),

\( \nu \in (\mu, 1] \) and \( \mu \in [0, 1] \) are hyperparameters.
Table 2: Range of $\nu$ for $\nu$-SVM, $E\nu$-SVM, and ER-SVM.

<table>
<thead>
<tr>
<th></th>
<th>Case N</th>
<th>Case C</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\nu$-SVM</td>
<td>$\nu \leq \nu$</td>
<td>$\nu &lt; \nu \leq \nu$</td>
</tr>
<tr>
<td>Optimal value</td>
<td>0</td>
<td>Negative</td>
</tr>
<tr>
<td>Optimal solution</td>
<td>$w = 0$</td>
<td>Admissible</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Case N</th>
<th>Case C</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E\nu$-SVM</td>
<td>$\nu \leq \nu$</td>
<td>$\nu &lt; \nu \leq \nu$</td>
</tr>
<tr>
<td>Optimal value</td>
<td>Nonnegative</td>
<td>Negative</td>
</tr>
<tr>
<td>Optimal solution</td>
<td>Admissible</td>
<td>Admissible</td>
</tr>
<tr>
<td>Constraint</td>
<td>$|w|^2 = 1$</td>
<td>$|w|^2 \leq 1$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Case N</th>
<th>Case C</th>
</tr>
</thead>
<tbody>
<tr>
<td>ER-SVM</td>
<td>$\nu \leq \nu_\mu$</td>
<td>$\nu_\mu &lt; \nu \leq \nu_\mu$</td>
</tr>
<tr>
<td>Optimal value</td>
<td>Nonnegative</td>
<td>Negative</td>
</tr>
<tr>
<td>Optimal solution</td>
<td>Admissible</td>
<td>Admissible</td>
</tr>
<tr>
<td>Constraint</td>
<td>$|w|^2 = 1$</td>
<td>$|w|^2 \leq 1$</td>
</tr>
</tbody>
</table>

Notes: If $\nu$ is greater than the lower threshold (case C), the nonconvex constraint of $E\nu$-SVM and ER-SVM can be relaxed to a convex constraint without changing the optimal solutions. Case C of $E\nu$-SVM is equivalent to $\nu$-SVM.

Remark 1. ER-SVM, equation 2.6, is obtained by relaxing the 0-1 integer constraints $\eta_i \in \{0, 1\}$ of the original formulation in Takeda et al. (2014) and replacing $\sum_{i \in I} (1 - \eta_i) \leq \mu |I|$ 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, equation 2.6, has an optimal solution such that $\eta^* \in \{0, 1\}^{|I|}$.

In this case, ER-SVM, equation 2.6, removes $\mu |I|$ samples and applies $E\nu$-SVM, equation 2.2, to the rest. Indeed, the heuristic algorithm of Takeda et al. (2014) for $E\nu$-SVM removes samples little by little until the total number of removed samples becomes $\mu |I|$. In this letter, we use formulation 2.6 and call it ER-SVM.

It can be easily seen that for fixed $\mu$, the optimal value of equation 2.6 decreases as $\nu$ increases. Moreover, it is shown in Takeda et al. (2014, lemma 1) that the nonconvex constraint $\|w\|^2 = 1$ can be relaxed to $\|w\|^2 \leq 1$ without changing the optimal solution so long as the optimal value is negative. Just like $E\nu$-SVM, ER-SVM, equation 2.6, has a threshold (we denote it by $\nu_\mu$) of the hyperparameter $\nu$ where the optimal value equals zero and the nonconvex constraint $\|w\|^2 = 1$ is essential for ER-SVM with $\nu \leq \nu_\mu$. The nonconvex constraint $\|w\|^2 = 1$ removes the lower threshold of $\nu$ and extends the admissible range of $\nu$ in the same way as $E\nu$-SVM (see Table 2). We show in section 3.2 that a special case (case C in Table 2) of ER-SVM is equivalent to ROD, equation 2.5, and ramp-loss SVM, equation 2.4, in the way where a special case (case C) of $E\nu$-SVM is equivalent to $\nu$-SVM.
\( \nu \)-SVM. Hence, ER-SVM can be seen as a natural extension of robust SVMs such as ROD and ramp-loss SVM. ER-SVM, equation 2.6, also has an upper threshold \( \bar{\nu} \), which, similar to \( \nu \)-SVM and \( \nu \)-SVM, makes the problem a bounded one.

### 2.3 Financial Risk Measures.

We define the financial risk measures as in Rockafellar and Uryasev (2002). Let us consider the cumulative distribution function for \( r_i(w, b) \):

\[
\Psi(w, b, \xi) := P(r_i(w, b) \leq \xi) = \frac{1}{|I|} |\{i \in I : r_i(w, b) \leq \xi\}|.
\]

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

\[
\zeta_{1-\nu}(w, b) := \min\{\xi : \Psi(w, b, \xi) \geq 1 - \nu\},
\]

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

\[
\zeta_{1-\nu}^+(w, b) := \inf\{\xi : \Psi(w, b, \xi) > 1 - \nu\}.
\]

The difference between VaR and VaR\(^+\) is illustrated in Figure 1.

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

\[
\phi_{1-\nu}(w, b) := \text{mean of the } (1 - \nu)\text{-tail distribution of } r_i(w, b),
\]
where the \((1 - \nu)\)-tail cumulative distribution function is as defined in Rockafellar and Uryasev (2002) by

\[
\Psi_{1-\nu}(w, b, \xi) := \begin{cases} 
0 & \text{for } \xi < \xi_{1-\nu}(w, b) \\
\frac{\Psi(w, b, \xi) - (1 - \nu)}{\nu} & \text{for } \xi \geq \xi_{1-\nu}(w, b).
\end{cases}
\]

The explicit mathematical form for calculating CVaR is given by the following theorem.

**Theorem 1.** (Rockafellar & Uryasev (2002)). One has

\[
\varphi_{1-\nu}(w, b) = \min_{\xi} F_{1-\nu}(w, b, \xi),
\]

where

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

Moreover,

\[
\xi_{1-\nu}(w, b) = \text{lower end point of } \arg\min_{\xi} F_{1-\nu}(w, b, \xi)
\]

\[
\xi_{1-\nu}^+(w, b) = \text{upper end point of } \arg\min_{\xi} F_{1-\nu}(w, b, \xi)
\]

hold.

CVaR can also be described as a maximum of linear functions as follows:

\[
\varphi_{1-\mu}(w, b) = \max_{\eta} \left\{ \frac{1}{\mu |I|} \sum_{i \in I} (1 - \eta_i) r_i(w, b) : 0 \leq \eta_i \leq 1, \sum_{i \in I} (1 - \eta_i) = \mu |I| \right\}. \tag{2.8}
\]

Here, we have used the properties of CVaR for a discrete loss distribution (see proposition 8 of Rockafellar & Uryasev, 2002). The description, equation 2.8, ensures the convexity of CVaR, \(\varphi_{1-\mu}(w, b)\), in \((w, b)\). We will use the above different representations of CVaR, equations 2.7 and 2.8, in the proof of proposition 1.
Figure 2: Distribution of $r_i(w,b)$: ER-SVM minimizes the average of $r_i$, $i \in I$, in the gray area. $(1 - \nu)$-CVaR is shown in red and $(1 - \mu)$-CVaR in blue.

3 Properties of Extended Robust SVM

3.1 Decomposition using Conditional Value-at-Risks. Here, we give an intuitive interpretation to ER-SVM, equation 2.6, using two CVaRs. Ev-SVM has been shown to minimize $(1 - \nu)$-CVaR $\varphi_{1-\nu}(w,b)$ (Takeda & Sugiyama, 2008). On the other hand, ER-SVM, equation 2.6, ignores a fraction $\mu$ of the samples and solves Ev-SVM by using the rest: that is, it minimizes CVaR using the rest of the samples. Hence, ER-SVM can be regarded as one that minimizes the average of $r_i(w,b)$ in the gray area of Figure 2. The mean of the gray area in Figure 2 can be described using two CVaRs, as is done in Wozabal (2012):

Proposition 1. ER-SVM, equation 2.6, is described as a problem minimizing the difference of two convex functions by using the two CVaRs:

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

s.t. $\|w\|^2 = 1$.  \hspace{1cm} (3.1)

The proof of proposition 1 is in appendix A.

Since CVaR $\varphi_a(w,b)$ is convex in $(w,b)$ for fixed $a \in [0,1)$, the objective function in equation 2.9 can be described as the difference between two convex functions. A similar model, CVaR-$(\alpha_L, \alpha_U)$-SVM, which relaxes the nonconvex constraint $\|w\|^2 = 1$ of equation 3.1 by $\|w\|^2 \leq 1$, was recently proposed (Tsyurmasto et al., 2013). As in Table 2, such a model is a special case (case C) of ER-SVM, and it is essentially equivalent to ramp-loss SVM or ROD.
3.2 Relationship with Existing Models. Here, we discuss the relationship between ER-SVM, ROD, and ramp-loss SVM by using the KKT conditions shown in appendix B. We begin by showing the equivalence of ramp-loss SVM and ROD. While ROD was proposed (Xu et al., 2006) as a direct approach that explicitly incorporates outlier suppression unlike ramp-loss SVM, the following lemma implies their equivalence.

**Lemma 1.** (relation between ROD and ramp-loss SVM). Ramp-loss SVM, equation 2.4, and ROD, equation 2.5, share all KKT points in the following sense.

1. Let \((w^*, b^*, \eta^*)\) be a KKT point of ramp-loss SVM, equation 2.4. Then it is also a KKT point of ROD, equation 2.5, with \(\mu = \frac{1}{|I|} \sum_{i \in I} (1 - \eta_i^*)\).

2. A KKT point of ROD, equation 2.5, having the Lagrange multiplier \(\tau^*\) for \(\sum_{i \in I} (1 - \eta_i) = \mu |I|\) is also a KKT point of ramp-loss SVM, equation 2.4, with \(s = \frac{\tau^*}{c}\).

The proof of lemma 1 is in section B.1.

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

**Theorem 2.** (relation between ER-SVM and ROD). Case C of ER-SVM (in Table 2), that is, equation B.3, and ROD, equation 2.5, share all KKT points in the following sense:

1. Let \((w^*, b^*)\) satisfy the KKT conditions of ROD, equation 2.5, and suppose \(w^* \neq 0\). \(\frac{1}{\|w^*\|^2} (w^*, b^*)\) satisfies the KKT conditions of case C of ER-SVM with a corresponding hyperparameter value.

2. Let \((w^*, b^*, \rho^*)\) satisfy the KKT conditions of case C of ER-SVM. Suppose \(\rho^* \neq 0\) and the objective value is nonzero. \(\frac{1}{\rho^*} (w^*, b^*)\) satisfies the KKT conditions of ROD with a corresponding hyperparameter value.

See section B.2 for the proof of theorem 2.

From lemma 1 and theorem 2, ramp-loss SVM and ROD can be regarded as a special case (case C in Table 2) of ER-SVM. As we discussed in section 3.1, CVaR-(\(\alpha_L, \alpha_U\))-SVM (Tsyurmasto et al., 2013) is also equivalent to case C (in Table 2) of ER-SVM. Theorem 2 is similar to the relation between C-SVM and \(\nu\)-SVM (which is a special case of E\(\nu\)-SVM). It was shown that the sets of global solutions of C-SVM and \(\nu\)-SVM correspond to each other when the hyperparameters are set properly (see Chang & Lin, 2001b and Schölkopf et al., 2000). We used KKT conditions to show the relationship between the nonconvex models.

3.3 Motivation for Nonconvex Regularizer. The nonconvex constraint \(\|w\|^2 = 1\) helps to remove the lower threshold of the hyperparameter \(\nu\), as Table 2 shows. The extension of the admissible range of \(\nu\) gives us a chance of finding a better classifier that cannot be found by using a convex regularizer as in \(\nu\)-SVM, equation 2.1, and robust SVMs, equation 2.3. Perez-Cruz et al.
(2003) showed examples where Ev-SVM outperforms ν-SVM owing to the extended range of ν.

Besides this empirical evidence, there is theoretical evidence as to why the extended parameter range is needed for ν-SVM. More precisely, theorem 3 gives an explicit condition for data sets \{x_i, y_i\}_{i \in I} where the admissible range of ν for ν-SVM is empty, and therefore, C-SVM and ν-SVM obtain a trivial classifier having \( w = 0 \) for any hyperparameter value of C and ν. The condition also applies to robust SVMs after removing all outliers with \( \eta_i = 0 \). Indeed, Figure 4e shows a numerical example where C-SVM, ν-SVM, and ramp-loss SVM for any hyperparameter value obtain trivial classifiers \( w = 0 \) but ER-SVM obtains a nontrivial one.

Rifkin, Pontil, and Verri (1999) studied the conditions under which C-SVM obtains a trivial solution. We directly connect their statements to ν-SVM and strengthen them in theorem 3 by adding a geometric interpretation for ν-SVM for when the admissible range of ν is empty for ν-SVM.

**Theorem 3.** Suppose \( 0 < |I_-| \leq |I_+| \) without loss of generality and define the reduced convex hull (RCH) (Crisp & Burges, 2000):

\[
\text{RCH}_\pm(v) = \left\{ \sum_{i \in I_\pm} \lambda_i x_i : \sum_{i \in I_\pm} \lambda_i = 1, \ 0 \leq \lambda_i \leq \frac{2}{v|I|} \right\}
\]

\( C \)-SVM and ν-SVM lead to the trivial classifier \( (w = 0) \) for any hyperparameter values \( C \in (0, \infty) \) and \( v \in (\nu, \nu^\ast] \) if and only if the training set \( \{x_i, y_i\}_{i \in I} \) satisfies \( \sum_{i \in I_-} \frac{1}{|I_-|} x_i \in \text{RCH}_+(\nu) \). When \( |I_-| > |I_+| \), the above statement is modified into \( \sum_{i \in I_+} \frac{1}{|I_+|} x_i \in \text{RCH}_-(\nu) \).

The proof is in appendix C.

### 3.4 Kernelization.

Linear learning models can be modified into more powerful nonlinear ones by using a kernel technique. Here, we briefly introduce a kernel variant of ER-SVM, equation 2.6. The input sample \( x \) is mapped into \( \phi(x) \) in a high (even infinite) dimensional inner product space \( \mathcal{H} \), and the classifier of the form \( h(x) = \langle w, \phi(x) \rangle_{\mathcal{H}} + b \) is learned from the training samples, where \( \langle a, b \rangle_{\mathcal{H}} \) is the inner product of \( a \) and \( b \) in \( \mathcal{H} \). The kernel function \( k(x, x') \) is defined as \( k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}} \).

Here, we show how the equality constraint \( \|w\|^2_2 = 1 \) in ER-SVM, equation 2.6, is dealt with in the kernel method. Let \( S \) be the subspace in \( \mathcal{H} \) spanned by \( \phi(x_i), i \in I \) and \( S^\perp \) be the orthogonal subspace of \( S \). Then the weight vector \( w \) can be decomposed into \( w = v + v^\perp \), where \( v \in S \) and \( v^\perp \in S^\perp \). The vector \( v^\perp \) does not affect the value of \( r_i(w, b) = -y_i(\langle w, \phi(x_i) \rangle_{\mathcal{H}} + b) \). The vector \( v \) can be expressed as a linear combination of \( \phi(x_i) \), such as \( v = \sum_{i \in I} \alpha_i \phi(x_i) \). If \( S = \mathcal{H} \) holds, \( w = v \) should also hold, and the constraint \( \langle w, w \rangle_{\mathcal{H}} = 1 \) is equivalent to \( \sum_{i, j \in I} \alpha_i \alpha_j k(x_i, x_j) = 1 \). When \( S \neq \mathcal{H} \) (i.e., the...
dimension of $S^\perp$ is not zero), one can prove that the constraint $\langle w, w \rangle_H = 1$ can be replaced with the convex constraint $\sum_{i,j \in I} \alpha_i \alpha_j k(x_i, x_j) \leq 1$, where one uses the fact that the gram matrix $(k(x_i, x_j))_{i,j}$ is nonnegative definite. Indeed, since the objective function depends on $w$ through the component $v$, the constraint on $w$ can be replaced with its projection onto the subspace $S$. Thus, the above convex constraint is obtained unless $S = H$. In such case, the kernel variant of ER-SVM is

$$\min_{\alpha, b, \rho, \eta} -\rho + \frac{1}{(v - \mu)|I|} \sum_{i \in I} \eta_i [\rho + r_i^k(\alpha, b)]^+$$

s.t. $0 \leq \eta_i \leq 1$, $\sum_{i \in I} (1 - \eta_i) = \mu |I|$, $\sum_{i,j \in I} \alpha_i \alpha_j k(x_i, x_j) \leq 1$, (3.3)

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

4 Algorithm

4.1 Simplified DCA. We begin with a brief introduction to the difference of convex functions (DC) program and the DC algorithm (DCA). The DC program is formulated using lower semicontinuous proper convex functions $u$ and $v$:

$$\min_z \{ f(z) := u(z) - v(z) : z \in \mathbb{R}^n \}.$$ (4.1)

DCA is an efficient algorithm for equation 4.1, and it is theoretically studied in Pham Dinh and Le Thi (1997). We shall use simplified DCA, which is the standard form of DCA. Simplified DCA sequentially linearizes the concave part in equation 4.1 and solves convex subproblems as follows:

$$z_{k+1} \in \arg\min_z \{ u(z) - (v(z_k) + \langle z - z_k, g_k \rangle) \},$$ (4.2)

where $g_k \in \partial v(z_k)$ is a subgradient of $v$ at $z_k$. The sequence $\{ z_k \}$ generated by simplified DCA has the following good convergence properties:

- The objective value is decreasing (i.e., $f(z_{k+1}) \leq f(z_k)$).
- DCA has global convergence.
- Every limit point of the sequence $\{ z_k \}$ is a critical point of $u - v$, which is also called a generalized KKT point.
DC Algorithm for Extended Robust Support Vector Machine

A critical point \( z^* \) of \( u - v \) is said to be critical if \( \partial u(z^*) \cap \partial v(z^*) \neq \emptyset \). This implies that a critical point \( z^* \) has \( g_u \in \partial u(z^*) \) and \( g_v \in \partial v(z^*) \) such that \( g_u - g_v = 0 \), a necessary condition for local minima. When equation 4.1 has a convex constraint \( z \in Z \), we can define the critical point by replacing \( u \) with \( u(z) + 1_Z(z) \), where \( 1_Z(z) \) is an indicator function equal to 0 if \( z \in Z \) and \( +\infty \) otherwise.

4.2 DCA for Extended Robust SVM. As shown in section 3.1, ER-SVM, equation 2.6 can be described as equation 3.1, whose objective function is a difference of CVaR functions. DCA is still not applicable to equation 3.1 because it has a nonconvex constraint \( \|w\|^2 = 1 \). We further reformulate equation 3.1 into a problem with a convex constraint using a sufficiently large constant \( t \):

\[
\min_{w,b} \nu \varphi_{1-\nu}(w, b) - \mu \varphi_{1-\mu}(w, b) - t \|w\|^2
\]

s.t. \( \|w\|^2 \leq 1 \).

This reformulation is a special case of the exact penalty approach (see Pham Dinh & Le Thi, 1997; Pham Dinh, Le Thi, & Muu, 1999). There exists \( t_0 \) such that equations 3.1 and 4.3 have the same set of optimal solutions for all \( t \geq t_0 \). We can estimate an upper bound of \( t_0 \) in our case by invoking the following lemma.

**Lemma 2.** If \( t \geq 0 \) in equation 4.3 is sufficiently large such that the optimal value of equation 4.3 is negative, equations 3.1 and 4.3 have the same set of optimal solutions.

The key point in the proof of lemma 2 is that CVaR has positive homogeneity (i.e., \( \varphi(aw, ab) = a \varphi(w, b) \) 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., Artzner, Delbaen, Eber, & Heath, 1999). 

**Proof of Lemma 2.** Let \((w^*, b^*)\) be an optimal solution of equation 4.3 and suppose to the contrary that \( \|w^*\| < 1 \). \((\frac{w^*}{\|w^*\|}, \frac{b^*}{\|w^*\|})\) achieves a smaller objective value than \((w^*, b^*)\) since

\[
\nu \varphi_{1-\nu} \left( \frac{w^*}{\|w^*\|}, \frac{b^*}{\|w^*\|} \right) - \mu \varphi_{1-\mu} \left( \frac{w^*}{\|w^*\|}, \frac{b^*}{\|w^*\|} \right) - t \|w^*\|^2
\]

\[
= \frac{1}{\|w^*\|} \left\{ \nu \varphi_{1-\nu}(w^*, b^*) - \mu \varphi_{1-\mu}(w^*, b^*) - t \|w^*\| \right\}
\]

\[
\leq \frac{1}{\|w^*\|} \left\{ \nu \varphi_{1-\nu}(w^*, b^*) - \mu \varphi_{1-\mu}(w^*, b^*) - t \|w^*\|^2 \right\}
\]

\[
< \nu \varphi_{1-\nu}(w^*, b^*) - \mu \varphi_{1-\mu}(w^*, b^*) - t \|w^*\|^2.
\]
However, this contradicts the optimality of \((w^*, b^*)\). Therefore, the optimal solution of equation 4.3 satisfies \(\|w\| = 1\), which implies that it is also optimal to equation 3.1.

The following DC program represents equation 4.3:

\[
\min_{w, b} \{ u(w, b) - v(w, b) \} \tag{4.4}
\]

where

\[
u(w, b) := 1_W(w) + \nu \varphi_{1-\nu}(w, b),
\]

\[
v(w, b) := \mu \varphi_{1-\mu}(w, b) + t \|w\|^2,
\]

\[W := \{ w \mid \|w\|^2 \leq 1 \}.
\]

Here, we apply simplified DCA to problem 4.4. Let \((w^k, b^k)\) be the solution obtained in the \((k-1)\)th iteration of simplified DCA. At the \(k\)th iteration of simplified DCA, we solve a subproblem, equation 4.2, constructed by linearizing the concave part \(-v(w, b)\) in equation 4.5 at \((w^k, b^k)\). The subproblem, equation 4.2, of the \(k\)th iteration is thus described as

\[
\min_{w, b} \nu \varphi_{1-\nu}(w, b) - \mu \langle g^k_w, w \rangle - \mu g^k_b b - 2t \langle w^k, w \rangle
\]

s.t. \(\|w\|^2 \leq 1\),

\[\tag{4.5}
\]

where \((g^k_w, g^k_b) \in \partial \varphi_{1-\mu}(w^k, b^k)\). \(\partial \varphi_{1-\mu}\) is the subdifferential of \(\varphi_{1-\mu}\) and \((g^k_w, g^k_b)\) is a subgradient of \(\varphi_{1-\mu}\) at \((w^k, b^k)\). An optimal solution of equation 4.5 is denoted by \((w^{k+1}, b^{k+1})\). Below, we show how to calculate a subgradient \((g^k_w, g^k_b)\) and how to choose a sufficiently large constant \(t\).

### 4.2.1 Subdifferential of CVaR

Here, we show how to calculate the subdifferential of CVaR, equation 2.8, by using the technique described in Wozabal (2012). The subdifferential of CVaR at \((w^k, b^k)\) is

\[
\partial_w \varphi_{1-\mu}(w^k, b^k) = \text{co} \left\{ -\frac{1}{\mu |I|} \sum_{i \in I} (1 - \eta^k_i) y_i x_i : \eta^k \in H(w^k, b^k) \right\},
\]

\[
\partial_b \varphi_{1-\mu}(w^k, b^k) = \text{co} \left\{ -\frac{1}{\mu |I|} \sum_{i \in I} (1 - \eta^k_i) y_i : \eta^k \in H(w^k, b^k) \right\},
\]

where \text{co} \(X\) is the convex hull of the set \(X\) and
Algorithm 1: DCA for Difference of CVaRs.

Input: $\mu \in [0, 1)$, $\nu \in (, \bar{\nu}]$, $(\mathbf{w}^0, b^0)$ such that $\|\mathbf{w}^0\|^2 = 1$ and a small value $\epsilon_1, \epsilon_2 > 0$.

1: $k = 0$.
2: repeat
3: Select $\eta^k \in H(\mathbf{w}^k, b^k)$ arbitrarily.
4: Update $t^k$ as
   
   $t^k \leftarrow \max \left\{ 0, \frac{\nu \varphi_{1-\nu}(\mathbf{w}^k, b^k) - \mu \varphi_{1-\mu}(\mathbf{w}^k, b^k)}{1 - \epsilon_2} \right\}$.
5: 
6: $(\mathbf{w}^{k+1}, b^{k+1})$ = a solution of subproblem 4.8.
7: $k \leftarrow k + 1$.
8: until $f(\mathbf{w}^k, b^k) - f(\mathbf{w}^{k+1}, b^{k+1}) < \epsilon_1$ where $f(\mathbf{w}, b) = \nu \varphi_{1-\nu}(\mathbf{w}, b) - \mu \varphi_{1-\mu}(\mathbf{w}, b)$.

$$H(\mathbf{w}^k, b^k) = \arg \max_{\eta^k} \left\{ \sum_{i \in I} (1 - \eta_i) r_i(\mathbf{w}^k, b^k) : \sum_{i \in I} (1 - \eta_i) = \mu |I|, 0 \leq \eta_i \leq 1 \right\}. \quad (4.6)$$

Let $\tilde{I} \subseteq I$ be the index set, where $r_i(\mathbf{w}^k, b^k), i \in \tilde{I},$ are the $\mu |I|$ largest values among $r_i(\mathbf{w}^k, b^k), i \in I$. We can easily find an optimal solution $\eta^k \in H(\mathbf{w}^k, b^k)$ by assigning 0 to $\eta_i^k, i \in \tilde{I}$, and 1 to others.

4.2.2 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 $k$th iteration, $t^k$ such that

$$t^k > \nu \varphi_{1-\nu}(\mathbf{w}^k, b^k) - \mu \varphi_{1-\mu}(\mathbf{w}^k, b^k). \quad (4.7)$$

Condition 4.7 ensures that the optimal value of equation 4.6 is negative, since the solution $(\mathbf{w}^k, b^k)$ in the previous iteration has a negative objective value. With such $t^k$, lemma 2 holds.

4.2.3 Explicit Form of Subproblem. Now we are ready to describe the subproblem 4.5 explicitly. By using the above results and substituting equation 2.7 for $\varphi_{1-\nu}(\mathbf{w}, b)$, equation 4.5 results in

$$\min_{\mathbf{w}, b, \rho, \xi} -\nu \rho + \frac{1}{|I|} \sum_{i \in I} \xi_i - \frac{1}{|I|} \sum_{i \in I} (1 - \eta_i)(\mathbf{w}, b) - 2t^k(\mathbf{w}^k, \mathbf{w}) \quad \text{s.t.} \quad \xi_i \geq \rho + r_i(\mathbf{w}, b), \xi_i \geq 0, \|\mathbf{w}\|^2 \leq 1. \quad (4.8)$$

The steps of the algorithm are listed in algorithm 1. The algorithm also has nice convergence properties, such as linear convergence, as shown in section 4.1.
4.2.4 CVaR-Function Decomposition. The decomposition of two CVaR functions enables us to interpret the algorithm and the resulting solution. The algorithm repeatedly solves subproblems by linearizing a concave part $-v(w, b)$ that corresponds to underestimating the sum of losses of outliers. The approximation becomes more accurate as the algorithm iterates.

There is another advantage to the decomposition. If the given parameter value $\nu$ is in case C, the term $-t\|w\|^2$ disappears from the objective function in equation 4.3 and the proposed algorithm acts like a polyhedral DC algorithm, which has the nice property of finite convergence (see Pham Dinh & Le Thi, 1997). This property is due to the polyhedral convexity of CVaR functions.

5 Numerical Results

We compared our approach, DCA for ER-SVM, with a heuristic algorithm (Takeda et al., 2014) for ER-SVM, CCCP (Collobert et al., 2006) for ramp-loss SVM, $\nu$-SVM algorithm (Perez-Cruz et al., 2003), and C-SVM from the LIBSVM software package (Chang & Lin, 2001a). The hyperparameter $s$ of ramp-loss SVM, equation 2.3, was set to 1, and $\mu$ in ER-SVM, equation 2.6, was set to 0.05; all models thus had one hyperparameter to be tuned. A comparison of ROD, equation 2.5 using SDP relaxation (Xu et al., 2006) is omitted because ROD is equivalent to ramp-loss SVM (see lemma 1).

We implemented all methods in Python (version 2.7.6) and used IBM ILOG CPLEX (version 12.6) to solve the optimization problems involved in these four methods: DCA and the heuristic algorithm for ER-SVM, CCCP for ramp-loss SVM, and $\nu$-SVM algorithm. We solved the primal formulation, equation 4.8, while running DCA to deal with data sets with a large sample size and small features. In the CCCP algorithm for ramp-loss SVM, dual formulations of convex subproblems were sequentially solved (see algorithm 2 of Collobert et al., 2006). Almost all of the numerical experiments were done on a PC (CPU: Intel Core i5-3437U (1.90 GHz) ×4, Memory: 8GB, OS: Ubuntu 14.04); a LINUX server (CPU: Intel Xeon E5-2680 (2.70GHz ×8 core) ×2, Memory: 64 GB, OS: Red Hat Enterprise Linux Server) was used for the large-scale data sets.

5.1 Accuracy for Synthetic Data Sets. We used synthetic data generated by following the procedure in Xu et al. (2006). 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 data set is $h(x) = x_1 - x_2 = 0$. We added outliers only to the training set with the label $-1$ by drawing samples uniformly from a half-ring with center 0, inner radius $R = 75$, and outer radius $R + 1$ in the space $x$ of $h(x) > 0$. The training set contained 50 samples from each class (100 in total), including outliers. The ratio of outliers in the training set was set to from 0 to 10%. The test set had 1000 samples from each class.
Figure 3: (a) The average test accuracy for the synthetic data set. (b, c) Computation times for the synthetic data set and cod-rna.

(2000 in total). We repeated the experiments 100 times, drawing training and test sets every repetition. We found the best parameter setting from nine candidates, \( \nu = 0.1, 0.2, \ldots, 0.9 \) and \( C = 5^{-4}, 5^{-3}, \ldots, 5^4 \).

Figure 3a plots the test accuracy of each model versus the outlier ratio. Each error bar spans from the 25th to 75th percentiles. ER-SVM by DCA achieved high average accuracy with small standard deviations especially when the outlier ratio was large. DCA and the heuristic algorithm (Takeda et al., 2014) solved the same problem for ER-SVM, but the heuristic one had poor performance; it seemed to get stuck in poor local solutions.

5.2 Accuracy on Real Data Sets. We used the data sets of the UCI repository (Blake & Merz, 1998) and LIBSVM (Chang and Lin, 2001a; see Table 3). We scaled the original data sets such that all attributes had zero mean and unit variance. For most data sets, we generated outliers \( \hat{x} \) uniformly from a ring with center 0 and radius \( R \). The radius \( R \) of generating outliers was set so that the outliers would have an impact on the test accuracy. When the feature size \( n \) of the data set was large, we generated outliers in different...
Table 3: List of UCI and LIBSVM Data Sets.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Used Classes</th>
<th>Number of Feature n</th>
<th>Number of Samples</th>
<th>Outlier Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>svmguide1</td>
<td></td>
<td>4</td>
<td>3089</td>
<td>(0.12, 0.71) R = 50</td>
</tr>
<tr>
<td>cod-rna</td>
<td></td>
<td>8</td>
<td>53,581</td>
<td>(−0.67) R = 30</td>
</tr>
<tr>
<td>diabetes</td>
<td></td>
<td>8</td>
<td>768</td>
<td>(0.52, 0.70) R = 20</td>
</tr>
<tr>
<td>vehicle</td>
<td>Class 1 versus rest</td>
<td>18</td>
<td>846</td>
<td>(0.42, 0.50) R = 200</td>
</tr>
<tr>
<td>satimage</td>
<td>Class 6 versus rest</td>
<td>36</td>
<td>4435</td>
<td>(0.21, 0.47) R = 75</td>
</tr>
<tr>
<td>splice</td>
<td></td>
<td>60</td>
<td>1000</td>
<td>(0.37, 0.97) R = 200</td>
</tr>
<tr>
<td>mushrooms</td>
<td></td>
<td>112</td>
<td>8124</td>
<td>(0.00, 0.96) label-flip</td>
</tr>
<tr>
<td>adult</td>
<td></td>
<td>123</td>
<td>1605</td>
<td>(0.32, 0.49) R = 300</td>
</tr>
<tr>
<td>dna</td>
<td>Class 2 versus class 3</td>
<td>180</td>
<td>3955</td>
<td>(0.00, 0.63) class 1</td>
</tr>
<tr>
<td>MNIST</td>
<td>Class 1 versus class 7</td>
<td>768</td>
<td>15,170</td>
<td>(0.00, 0.96) class 9</td>
</tr>
<tr>
<td>internet ad</td>
<td></td>
<td>1554</td>
<td>3279</td>
<td>(0.02, 0.28) label-flip</td>
</tr>
</tbody>
</table>

Notes: For most data sets, outliers are generated on the sphere with center 0 and radius R. The radius was set so that the outliers would have an impact on the test accuracy. When the feature size n of the data set is large, outliers are generated in different ways: the label flip for binary class data sets or the samples with the third class for multiclass data sets. \( \nu \) of cod-rna could not be calculated because the algorithm ran out of memory.

The best parameter was chosen from nine candidates, \( \nu \in (0, \bar{\nu}] \) with equal intervals and \( C = 5^{-4}, 5^{-3}, \ldots, 5^{4} \). In this experiment, we split the data set into a training set, validation set, and test set in the ratio of 4:3:3. The hyperparameters were determined on the training set and validation set, both of which were contaminated. The performance was evaluated on the clean test set.

Table 4 shows the results of 30 trials for most data sets (only 3 or 10 trials for large-scale data sets because of long computation times). “-” in Table 4 indicates the out-of-memory error. “N” in Table 4 means that ER-SVM (or \( E_{\nu}\)-SVM) achieved the best accuracy with \( \nu \leq \nu_{\mu} \) (or \( \nu \leq \nu \)) at least once, and “C” means that the best accuracy was achieved with \( \nu_{\mu} < \nu \) (or \( \nu < \nu \)) in all trials.

The prediction performance of ER-SVM by DCA was better than that of the other methods and very stable as the outlier ratio increased. C-SVM often performs well when data sets contain no or few outliers, but it tended to be beaten by ER-SVM (DCA) as the outlier ratio increased. Our DCA algorithm could find a better solution, and this led to ER-SVM by DCA having better prediction performance than the heuristic method. We can see from the table that the nonconvex cases (N) of ER-SVM and \( E_{\nu}\)-SVM performed better than their convex cases on the intractable data sets (e.g., diabetes, vehicle, splice), where all linear SVMs performed poorly.
Table 4: Average Test Accuracy and F1 Score \((2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}})\) over 30 Trials for Most Data Sets of the Proposed Method, ER-SVM (DCA), Compared with the Others.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>OR</th>
<th>ave.acc</th>
<th>F1</th>
<th>ave.acc</th>
<th>F1</th>
<th>ave.acc</th>
<th>F1</th>
<th>ave.acc</th>
<th>F1</th>
<th>ave.acc</th>
<th>F1</th>
<th>ave.acc</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>svmguide1</td>
<td>0%</td>
<td>0.954</td>
<td>0.965</td>
<td>C</td>
<td>0.956</td>
<td>0.966</td>
<td>N</td>
<td>0.959</td>
<td>0.968</td>
<td>0.954</td>
<td>0.964</td>
<td>0.954</td>
<td>0.964</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.944</td>
<td>0.956</td>
<td>C</td>
<td>0.939</td>
<td>0.952</td>
<td>N</td>
<td>0.946</td>
<td>0.958</td>
<td>0.921</td>
<td>0.939</td>
<td>0.879</td>
<td>0.911</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.948</td>
<td>0.959</td>
<td>C</td>
<td>0.935</td>
<td>0.950</td>
<td>N</td>
<td>0.946</td>
<td>0.958</td>
<td>0.909</td>
<td>0.931</td>
<td>0.758</td>
<td>0.846</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.937</td>
<td>0.952</td>
<td>N</td>
<td>0.920</td>
<td>0.939</td>
<td>N</td>
<td>0.893</td>
<td>0.925</td>
<td>0.887</td>
<td>0.916</td>
<td>0.658</td>
<td>0.790</td>
</tr>
<tr>
<td>cod-rna</td>
<td>0%</td>
<td>0.939</td>
<td>0.912</td>
<td>C</td>
<td>0.939</td>
<td>0.911</td>
<td>N</td>
<td>–</td>
<td>–</td>
<td>0.938</td>
<td>0.909</td>
<td>0.939</td>
<td>0.910</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.937</td>
<td>0.908</td>
<td>C</td>
<td>0.937</td>
<td>0.908</td>
<td>N</td>
<td>–</td>
<td>–</td>
<td>0.935</td>
<td>0.903</td>
<td>0.914</td>
<td>0.863</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.938</td>
<td>0.911</td>
<td>C</td>
<td>0.938</td>
<td>0.910</td>
<td>N</td>
<td>–</td>
<td>–</td>
<td>0.936</td>
<td>0.905</td>
<td>0.773</td>
<td>0.522</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.936</td>
<td>0.904</td>
<td>C</td>
<td>0.936</td>
<td>0.906</td>
<td>N</td>
<td>–</td>
<td>–</td>
<td>0.932</td>
<td>0.897</td>
<td>0.667</td>
<td>0.500</td>
</tr>
<tr>
<td>diabetes</td>
<td>0%</td>
<td>0.757</td>
<td>0.829</td>
<td>N</td>
<td>0.755</td>
<td>0.827</td>
<td>N</td>
<td>0.760</td>
<td>0.828</td>
<td>0.762</td>
<td>0.828</td>
<td>0.769</td>
<td>0.836</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.756</td>
<td>0.823</td>
<td>N</td>
<td>0.742</td>
<td>0.819</td>
<td>N</td>
<td>0.752</td>
<td>0.820</td>
<td>0.754</td>
<td>0.821</td>
<td>0.749</td>
<td>0.824</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.757</td>
<td>0.827</td>
<td>N</td>
<td>0.749</td>
<td>0.822</td>
<td>N</td>
<td>0.753</td>
<td>0.822</td>
<td>0.746</td>
<td>0.816</td>
<td>0.747</td>
<td>0.823</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.750</td>
<td>0.822</td>
<td>N</td>
<td>0.728</td>
<td>0.811</td>
<td>N</td>
<td>0.742</td>
<td>0.818</td>
<td>0.732</td>
<td>0.811</td>
<td>0.728</td>
<td>0.812</td>
</tr>
<tr>
<td>vehicle</td>
<td>0%</td>
<td>0.789</td>
<td>0.535</td>
<td>N</td>
<td>0.781</td>
<td>0.524</td>
<td>N</td>
<td>0.787</td>
<td>0.534</td>
<td>0.792</td>
<td>0.530</td>
<td>0.791</td>
<td>0.532</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.773</td>
<td>0.517</td>
<td>N</td>
<td>0.762</td>
<td>0.480</td>
<td>N</td>
<td>0.777</td>
<td>0.492</td>
<td>0.767</td>
<td>0.488</td>
<td>0.780</td>
<td>0.465</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.779</td>
<td>0.498</td>
<td>N</td>
<td>0.759</td>
<td>0.434</td>
<td>N</td>
<td>0.761</td>
<td>0.403</td>
<td>0.751</td>
<td>0.420</td>
<td>0.756</td>
<td>0.423</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.768</td>
<td>0.461</td>
<td>N</td>
<td>0.749</td>
<td>0.326</td>
<td>N</td>
<td>0.757</td>
<td>0.129</td>
<td>0.714</td>
<td>0.304</td>
<td>0.745</td>
<td>0.386</td>
</tr>
<tr>
<td>satimage</td>
<td>0%</td>
<td>0.908</td>
<td>0.792</td>
<td>C</td>
<td>0.905</td>
<td>0.788</td>
<td>N</td>
<td>0.908</td>
<td>0.793</td>
<td>0.909</td>
<td>0.798</td>
<td>0.911</td>
<td>0.801</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.903</td>
<td>0.786</td>
<td>N</td>
<td>0.893</td>
<td>0.764</td>
<td>N</td>
<td>0.899</td>
<td>0.777</td>
<td>0.896</td>
<td>0.773</td>
<td>0.898</td>
<td>0.778</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.899</td>
<td>0.775</td>
<td>N</td>
<td>0.889</td>
<td>0.752</td>
<td>N</td>
<td>0.898</td>
<td>0.775</td>
<td>0.892</td>
<td>0.764</td>
<td>0.891</td>
<td>0.762</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.893</td>
<td>0.766</td>
<td>N</td>
<td>0.888</td>
<td>0.753</td>
<td>N</td>
<td>0.897</td>
<td>0.773</td>
<td>0.890</td>
<td>0.753</td>
<td>0.874</td>
<td>0.738</td>
</tr>
<tr>
<td>splice</td>
<td>0%</td>
<td>0.784</td>
<td>0.784</td>
<td>C</td>
<td>0.772</td>
<td>0.778</td>
<td>N</td>
<td>0.783</td>
<td>0.781</td>
<td>0.794</td>
<td>0.790</td>
<td>0.783</td>
<td>0.786</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.771</td>
<td>0.775</td>
<td>N</td>
<td>0.756</td>
<td>0.760</td>
<td>N</td>
<td>0.772</td>
<td>0.776</td>
<td>0.774</td>
<td>0.777</td>
<td>0.777</td>
<td>0.779</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.769</td>
<td>0.765</td>
<td>C</td>
<td>0.745</td>
<td>0.747</td>
<td>N</td>
<td>0.756</td>
<td>0.756</td>
<td>0.761</td>
<td>0.761</td>
<td>0.755</td>
<td>0.765</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.756</td>
<td>0.753</td>
<td>N</td>
<td>0.727</td>
<td>0.730</td>
<td>N</td>
<td>0.725</td>
<td>0.725</td>
<td>0.733</td>
<td>0.737</td>
<td>0.737</td>
<td>0.743</td>
</tr>
</tbody>
</table>
Table 4: Continued.

<table>
<thead>
<tr>
<th>Data Set</th>
<th>OR</th>
<th>ER-SVM (DCA)</th>
<th>ER-SVM (heuristics)</th>
<th>Ramp-Loss SVM</th>
<th>Ev-SVM</th>
<th>C-SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>ave.acc</td>
<td>F1</td>
<td>ave.acc</td>
<td>F1</td>
<td>ave.acc</td>
</tr>
<tr>
<td>mushrooms</td>
<td>0%</td>
<td>0.982</td>
<td>0.981</td>
<td>C</td>
<td>0.983</td>
<td>0.982</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.998</td>
<td>0.998</td>
<td>C</td>
<td>0.941</td>
<td>0.936</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.998</td>
<td>0.998</td>
<td>C</td>
<td>0.917</td>
<td>0.904</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>0.998</td>
<td>0.997</td>
<td>C</td>
<td>0.903</td>
<td>0.887</td>
</tr>
<tr>
<td>adult</td>
<td>0%</td>
<td>0.820</td>
<td>0.588</td>
<td>C</td>
<td>0.808</td>
<td>0.438</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.821</td>
<td>0.592</td>
<td>C</td>
<td>0.797</td>
<td>0.381</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.820</td>
<td>0.597</td>
<td>C</td>
<td>0.790</td>
<td>0.435</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>0.817</td>
<td>0.580</td>
<td>C</td>
<td>0.776</td>
<td>0.345</td>
</tr>
<tr>
<td>dna</td>
<td>0%</td>
<td>0.971</td>
<td>0.955</td>
<td>C</td>
<td>0.971</td>
<td>0.955</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.968</td>
<td>0.944</td>
<td>C</td>
<td>0.961</td>
<td>0.938</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.957</td>
<td>0.933</td>
<td>C</td>
<td>0.954</td>
<td>0.931</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>0.950</td>
<td>0.924</td>
<td>C</td>
<td>0.945</td>
<td>0.919</td>
</tr>
<tr>
<td>MNIST</td>
<td>0%</td>
<td>0.988</td>
<td>0.989</td>
<td>C</td>
<td>0.987</td>
<td>0.988</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.986</td>
<td>0.986</td>
<td>C</td>
<td>0.981</td>
<td>0.982</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.989</td>
<td>0.989</td>
<td>C</td>
<td>0.976</td>
<td>0.976</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td>0.982</td>
<td>0.982</td>
<td>C</td>
<td>0.980</td>
<td>0.980</td>
</tr>
<tr>
<td>internet ad</td>
<td>0%</td>
<td>0.944</td>
<td>0.758</td>
<td>C</td>
<td>0.921</td>
<td>0.619</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.955</td>
<td>0.820</td>
<td>C</td>
<td>0.938</td>
<td>0.720</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.960</td>
<td>0.837</td>
<td>C</td>
<td>0.948</td>
<td>0.788</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.953</td>
<td>0.798</td>
<td>C</td>
<td>0.953</td>
<td>0.800</td>
</tr>
</tbody>
</table>

Notes: The ratio of outliers (OR) varies from 0%. C indicates an out-of-memory error. N implies that the objective value of ER-SVM (or Ev-SVM) was positive (that is, case N in table 2 occurred and the nonconvex constraint $\|w\|^2 = 1$ worked effectively) at least once in the trials. Only 3 or 10 trials were used for large-scale data sets because of the long computation times. The best scores among these methods are in bold.
5.3 Comparison of CPU Times. Here, we show the trend of computation time of our method. Figure 3b plots the average of the CPU times in the range from the 25th to 75th percentile versus the outlier ratio of each method on the synthetic data set of Figure 3a. While the computation times of some algorithms such as C-SVM (with $C = 25$) and ramp-loss SVM slightly increased as the outlier ratio increased, the time of ER-SVM (DCA) was not significantly affected by increasing the outlier ratio.

Figure 3c shows the CPU time of each method with respect to the sample size of the cod-rna data set. Subsets with the sample size indicated on the horizontal axis were randomly chosen 10 times. The error bars show the range from the 25th to 75th percentile.

Both figures imply that our method, ER-SVM by DCA, performed comparably to the other methods despite the fact that it has two kinds of nonconvexities. Indeed, our method was faster than other robust SVM approaches: heuristics for ER-SVM and ramp-loss SVM. We omitted ramp-loss SVM from Figure 3c because out-of-memory errors occurred for 10,000 training samples when we tried to solve dual-formulated subproblems of ramp-loss SVM by using CPLEX (state-of-the-art commercial optimization software package). We can also see that the computation time of our method was in a smaller range than that of LIBSVM for C-SVM for varied parameter values.

5.4 Detailed Observations on DCA.

5.4.1 Comparison of DCA and Heuristics (Figure 4a). Table 4 reveals the advantage of DCA (ER-SVM (DCA)) over the heuristic method (ER-SVM (heuristics)) in terms of prediction accuracy. Here, we assess the quality of the solutions found by these algorithms by comparing their optimal values for the liver data set ($n = 6$ and $m = 345$).

The initial solutions $(w^0, b^0)$ were selected from a uniform random distribution on the unit sphere, and the experiments were repeated 300 times. We set the hyperparameter $\mu$ for DCA corresponding to the one automatically selected in the heuristic algorithm. We counted how many times our algorithm (DCA) achieved smaller objective values than those of the heuristics. We counted win, lose, and draw cases in 300 trials (win or lose is when the gap between the objective values is more than 3% considering the numerical error). Figure 4a shows that our algorithm (DCA) tended to achieve wins or draws, especially for larger $\nu$. This result supports the claim of Le Thi and Pham Dinh (2005) and Pham Dinh and Le Thi (1997) that DCA tends to converge to a global solution.

5.4.2 Efficient Update of $t_k$. We use the results for the liver data set to show the effectiveness of the auto update rule of a constant $t_k$ as in equation 4.7. Figure 4b implies that the convergence when using our auto update rule is much faster than when using a fixed constant $t = 0.1$ or 0.5.
Figure 4: (a) The number of times ER-SVM achieved smaller objective values than those of the heuristic algorithm in 300 trials. (b) Faster convergence by our update rule of $t_k$ in equation 4.7 than a fixed constant. (c) Computation time of algorithm 1 for the liver data set. (d) Test accuracy of ER-SVM with the polynomial kernel for liver. (e) An example where ER-SVM obtained a nontrivial classifier, but C-SVM, $\nu$-SVM, and ramp-loss SVM with any hyperparameter value obtained trivial classifiers $w = 0$. (f) The performance of DCA for ER-SVM and its restricted case C (which is equivalent to ramp-loss SVM).

5.4.3 Computation Time on Case N. Here, we investigate the change in computation time with respect to the parameter $\nu$. Figure 4c shows the computation time averaged over 100 trials. The vertical line is the estimated
value of $\nu_\mu$. The extension of the parameter range corresponds to $\nu < \nu_\mu$, where the nonconvex constraint $\|w\|^2 = 1$ worked. ER-SVM with $\nu < \nu_\mu$ found classifiers that ROD and ramp-loss SVM did not. The figure shows that the computation time did not change much except for $\nu = 0.35$. The optimal margin variable $\rho$ was zero around $\nu = 0.35$, which might make the problem numerically unstable.

5.4.4 Kernelized ER-SVM. Figure 4d shows the test accuracy of ER-SVM with a linear kernel or a polynomial kernel defined as $k(x_i, x_j) = (\langle x_i, x_j \rangle + c)^d$. The hyperparameters of the polynomial kernel were $c = 1$ and $d = 2$. We used 50% of the data set as the training set and the rest as the test set. The training set was contaminated by using a half-ring as shown in section 5.2 or Xu et al. (2006). The test accuracy was evaluated by the mean of 100 trials for each $\nu$. The markers in Figure 4d imply that the objective value was positive (i.e., case N in Table 2 occurred and the nonconvex constraint $\|w\|^2 = 1$ worked effectively) at least once in 100 trials. This result implies that the extension is effective for not only a linear kernel but also a polynomial kernel, especially when the data set is contaminated. Furthermore, the polynomial kernel was more accurate than the linear one regardless of the outlier ratio, which indicates the effectiveness of kernelization of ER-SVM.

5.5 Effectiveness of the Extension. Figure 4e verifies theorem 3. When the number of the samples in each class is unbalanced or the samples in two classes overlap substantially as in Figure 4e, C-SVM, $\nu$-SVM, and ramp-loss SVM obtain trivial classifiers ($w = 0$), while ER-SVM obtains a nontrivial classifier. That is, this figure implies the effectiveness of the nonconvex constraint $\|w\|^2 = 1$.

Figure 4f shows how the test accuracy of each method is affected by the degree of intersection of the samples in two classes. We used synthetic data sets that had the same covariance and number of samples as in Figure 4e, but only the mean of the distribution of positive samples was changed. The horizontal axis is the distance between the mean of the distributions of positive and negative samples. The hyperparameters of ER-SVM were fixed to $(\nu, \mu) = (0.05, 0.01)$. The solid line is our method (DCA for ER-SVM), and the dashed line is its case C (which is equivalent to ramp-loss SVM), where $t$ in equation 4.8 is set to zero. Figure 4f implies that case N occurred frequently when the distance was small and the nonextended model had poor test accuracy in such cases because of having $w = 0$ as an optimal solution.

6 Conclusion

We theoretically analyzed ER-SVM, proving that ER-SVM is a natural extension of robust SVMs and discussed the conditions under which such an
extension works. Furthermore, we proposed a new, efficient algorithm that has theoretically good properties. Numerical experiments showed that our algorithm worked efficiently.

In this letter, we focused on binary classification problems. Along the same line, we can formulate extended robust variants of learning algorithms for other statistical problems such as regression problems. The proposed simplified DCA with a similar CVaR-function decomposition is applicable to extended robust learning algorithms. In particular, Wu and Liu (2007) proposed a multiclass extension of binary SVM using the ramp loss. We may be able to formulate such a multiclass extension for ER-SVM and apply the proposed DCA to the resulting problem.

We now have to handle large-scale data sets arising from real-world problems. When the given data set has a large feature size, solving the dual of subproblems 4.8 may speed up our algorithm, DCA, more than by solving the primal subproblems. An alternative practical approach to solving large-scale problems is to use a memory-efficient method such as the SMO algorithm (Platt, 1998) for solving subproblems, although its worst-case computation time can be quite large. When the given data set has a large sample size, a stochastic variant of our method might be necessary. We would like to improve our method so that it works better on large-scale data sets.

Our method, ER-SVM by DCA, tends to perform better than the other methods especially on data sets with small feature sizes. For high-dimensional data sets, we recommend combining our method with feature selection.

Appendix A: Proof of Proposition 1

The objective function of ER-SVM, equation 2.6, under the constraints can be equivalently rewritten as

\[
\zeta + \frac{1}{(v - \mu)|I|} \sum_{i \in I} \eta_i [r_i(w, b) - \zeta]^+ = \frac{1}{v - \mu} \left\{ (v - \mu) \zeta + \frac{1}{|I|} \sum_{i \in I} \eta_i [r_i(w, b) - \zeta]^+ \right\} \\
= \frac{1}{v - \mu} \left\{ (v - \mu) \zeta + \frac{1}{|I|} \sum_{i \in I} [r_i(w, b) - \zeta]^+ - \frac{1}{|I|} \sum_{i \in I} (1 - \eta_i) [r_i(w, b) - \zeta]^+ \right\}
\]
The last equality is obtained by using a constraint, \( \sum_{i \in I} (1 - \eta_i) = \mu |I| \), of ER-SVM, equation 2.6.

Now we show that the term in the last equation,

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

can be written as \( \frac{1}{|I|} \sum_{i \in I} (1 - \eta_i) r_i(w, b) \) by showing that \( r_i(w^*, b^*) - \zeta^* \geq 0 \) holds for any \( i \) whenever \( 1 - \eta_i^* > 0 \) at the optimal solution \( (w^*, b^*, \zeta^*, \eta^*) \) of equation 2.6. Here, we assume that \( r_i(w^*, b^*), \forall i \), are sorted in descending order; \( r_1(w^*, b^*) \geq r_2(w^*, b^*) \geq \ldots \). Then \( \eta^* \) should be

\[
\eta_1^* = \ldots = \eta_{\lfloor \mu |I| \rfloor}^* = 0, \quad \eta_{\lfloor \mu |I| \rfloor + 1}^* > 0, \quad \eta_{\lfloor \mu |I| \rfloor + 2}^* = \ldots 1.
\]

Note that \( \zeta^* \) must be an optimal solution of the problem:

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

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

\[
r_{\lfloor \mu |I| \rfloor + 1}(w^*, b^*) \geq \zeta_{1-\alpha}^+(w^*, b^*) \geq \zeta^*,
\]

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

Therefore, we can rewrite the objective function of ER-SVM, equation 2.6, as

\[
\frac{1}{\nu - \mu} \left\{ v\zeta + \frac{1}{|I|} \sum_{i \in I} [r_i(w, b) - \zeta]^+ - \frac{1}{|I|} \sum_{i \in I} (1 - \eta_i) r_i(w, b) \right\}. \quad (A.1)
\]
By using equation 2.7 for the first two terms of equation A.1 and using equation 2.8 for the last term, we can further rewrite equation A.1 as

\[ \frac{1}{\nu - \mu} \{ \nu \varphi_{1-\nu}(w, b) - \mu \varphi_{1-\mu}(w, b) \}, \]

which is the objective function of equation 3.1. This implies that ER-SVM, equation 2.6, can be described in the form of equation 3.1.

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

We show differentiable formulations of case C of ER-SVM, equation 2.6, ROD, equation 2.5, and ramp-loss SVM, equation 2.4, to write the KKT conditions for each model.

**Continuous ramp-loss SVM**

\[
\begin{aligned}
\min_{w, b, \eta, \xi} & \quad \frac{1}{2} \|w\|^2 + C \sum_{i \in I} \{ \eta_i \xi_i + s(1 - \eta_i) \} \\
\text{s.t.} & \quad \eta_i \leq 1, \\
& \quad \xi_i \geq 1 + r_i(w, b), \quad \xi_i \geq 0
\end{aligned}
\]

(B.1)

**Continuous ROD**

\[
\begin{aligned}
\min_{w, b, \eta, \xi} & \quad \frac{1}{2} \|w\|^2 + C \sum_{i \in I} \eta_i \xi_i \\
\text{s.t.} & \quad \eta_i \leq 1, \quad \sum_{i \in I} (1 - \eta_i) = \mu |I|, \\
& \quad \xi_i \geq 1 + r_i(w, b), \quad \xi_i \geq 0
\end{aligned}
\]

(B.2)

**Continuous ER-SVM (limited to Case C in Table 2)**

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

(B.3)

The KKT conditions of the above problems are defined as follows.
KKT conditions of equation B.1 using \((w, b, \eta, \xi; \lambda, \alpha, \beta, \gamma)\):

\[
\sum_{i \in I} \lambda_i y_i = 0, \tag{B.4a}
\]
\[
\gamma_i \xi_i = 0, \tag{B.4b}
\]
\[
\alpha_i (\eta_i - 1) = 0, \tag{B.4c}
\]
\[
\beta_i \eta_i = 0, \tag{B.4d}
\]
\[
\lambda_i, \alpha_i, \beta_i, \gamma_i \geq 0, \tag{B.4e}
\]
\[
0 \leq \eta_i \leq 1, \xi_i \geq 0, \tag{B.4f}
\]

\[
\lambda_i [1 - \xi_i + r_i(w, b)] = 0, \tag{B.5a}
\]
\[
-r_i(w, b) \geq 1 - \xi_i, \tag{B.5b}
\]
\[
w = \sum_{i \in I} \lambda_i y_i x_i, \tag{B.5c}
\]
\[
C \eta_i - \lambda_i - \gamma_i = 0, \tag{B.5d}
\]

\[
C \xi_i - Cs + \alpha_i - \beta_i = 0 \tag{B.6}
\]

KKT conditions of equation B.2 using \((w, b, \eta, \xi; \lambda, \alpha, \beta, \gamma, \tau)\):
equations B.4 and B.5

\[
C \xi_i - \tau + \alpha_i - \beta_i = 0, \tag{B.7}
\]
\[
\sum_{i \in I} (1 - \eta_i) = \mu |I| \tag{B.8}
\]

KKT conditions of equation B.3 using \((w, b, \rho, \eta, \xi; \lambda, \alpha, \beta, \gamma, \tau, \delta)\):
equations B.4 and B.8

\[
\lambda_i [\rho - \xi_i + r_i(w, b)] = 0, \tag{B.9a}
\]
\[
-r_i(w, b) \geq \rho - \xi_i, \tag{B.9b}
\]
\[
2 \delta w = \sum_{i \in I} \lambda_i y_i x_i, \tag{B.9c}
\]
\[
\frac{\eta_i}{|I|} - \lambda_i - \gamma_i = 0, \tag{B.9d}
\]
\[
\frac{1}{|I|} \xi_i - \tau + \alpha_i - \beta_i = 0, \tag{B.9e}
\]
\[ \sum_{i \in I} \lambda_i = v - \mu, \quad \text{(B.9f)} \]
\[ \delta(\|w\|^2 - 1) = 0, \delta \geq 0 \quad \text{(B.9g)} \]

**B.1 Proof of Lemma 1.** The difference between the KKT conditions of ramp-loss SVM, equation B.1 and ROD, equation B.2, is only equations B.6 to B.8.

Note that a KKT point \((w^*, b^*, \eta^*)\) of ramp-loss SVM, equation B.1, satisfies the KKT conditions of ROD, equation B.2, with \(\mu = \frac{1}{|I|} \sum_{i \in I} (1 - \eta_i^*)\). On the other hand, a KKT point of equation B.2 whose Lagrange multiplier for \(\sum_{i \in I} (1 - \eta_i) = \mu |I|\) is \(\tau^*\) satisfies the KKT conditions of ramp-loss SVM, equation B.1, with \(s = \frac{\tau^*}{\epsilon}\).

**B.2 Proof of Theorem 2.** Let us prove the first statement. Let \((w^*, b^*, \eta^*, \xi^*)\) be a KKT point of equation B.2 with hyperparameter \(C\) and Lagrange multipliers \((\lambda^*, \alpha^*, \beta^*, \gamma^*, \tau^*, \delta^*)\). Suppose \(w^* \neq 0\). Then \((w^*/\|w^*\|, b^*/\|w^*\|, \eta^*, \xi^*/\|w^*\|, \rho^* = \frac{1}{\|w^*\|})\) is a KKT point of equation B.3 with Lagrange multipliers \(\frac{1}{\|w^*\|}(\lambda^*, \alpha^*, \beta^*, \gamma^*, \eta^*, \xi^*)\) and \(\delta = \frac{|w^*|}{2\|w^*\|}\).

Now let us prove the second statement. Let \((w^*, b^*, \eta^*, \xi^*, \rho^*)\) be a KKT point of equation B.3 with Lagrange multipliers \((\lambda^*, \alpha^*, \beta^*, \gamma^*, \tau^*, \delta^*)\). Suppose \(\rho^* \neq 0\). If \(\delta^* \neq 0\), \((w^*/\|w^*\|, b^*/\|w^*\|, \eta^*, \xi^*/\|w^*\|, \rho^* \delta^*/\|w^*\|)\) satisfies the KKT conditions of equation B.2 with Lagrange multipliers \(\frac{1}{\|w^*\|}(\lambda^*, \alpha^*, \beta^*, \gamma^*, \xi^*/\|w^*\|, \tau^*)\) and hyperparameter \(\tilde{\epsilon} = \frac{1}{\|w^*\|}\). Now, we will prove \(\delta^* \neq 0\) using the assumption of a nonzero optimal value. Consider the following problem, which fixes \(\eta\) of equation B.3 to \(\eta^*\):

\[
\begin{aligned}
\min_{w, b, \rho, \xi} & \quad -\rho + \frac{1}{(v - \mu)|I|} \sum_{i \in I} \eta_i^* \xi_i \\
\text{s.t.} & \quad \|w\|^2 \leq 1, -r_i(w, b) \geq \rho - \xi_i, \xi_i \geq 0.
\end{aligned}
\quad \text{(B.10)}
\]

The KKT conditions of equation B.10 are as follows:

\[ 2\delta w = \sum_{i \in I} \lambda_i y_i x_i, \sum_{i \in I} \lambda_i y_i = 0, \frac{\eta_i}{|I|} - \lambda_i - \gamma_i = 0, \]
\[ \sum_{i \in I} \lambda_i = v - \mu, \]
\[ \lambda_i (\rho - \xi_i + r_i(w, b)) = 0, \gamma_i \xi_i = 0, \delta(\|w\|^2 - 1) = 0, \]
\[ \delta, \lambda_i, \gamma_i \geq 0, \]
\[ -r_i(w, b) \geq \rho - \xi_i, \xi_i \geq 0. \]
$(w^*, b^*, \xi^*, \rho^*)$ is also a KKT point of equation B.10 with the same Lagrange multipliers $(\lambda^*, \nu^*, \delta^*)$ as equation B.3. Moreover, $(w^*, b^*, \xi^*, \rho^*)$ is not only a KKT point but also an optimal solution of equation B.10 because B.10 is a convex problem. Since the objective function of the dual problem of equation B.10 is $-\delta(\|w\|^2 + 1)$, $\delta^* = 0$ if and only if the objective value is zero. Then we can see that $\delta^* \neq 0$ under the assumption of a nonzero objective value.

Appendix C: Proof of Theorem 3

Consider the dual problems of $\nu$-SVM and $C$-SVM:

\[
\begin{align*}
\min_{\lambda} & \quad \frac{1}{2} \left\| \sum_{i \in I} \lambda_i y_i x_i \right\|^2 \\
\text{s.t.} & \quad 0 \leq \lambda_i \leq \frac{1}{|I|}, \quad \sum_{i \in I} y_i \lambda_i = 0, \quad \sum_{i \in I} \lambda_i = \nu, \quad (D_\nu) \\
\min_{\lambda} & \quad \frac{1}{2} \left\| \sum_{i \in I} \lambda_i y_i x_i \right\|^2 - \sum_{i \in I} \lambda_i \\
\text{s.t.} & \quad 0 \leq \lambda_i \leq \frac{1}{|I|}, \quad \sum_{i \in I} y_i \lambda_i = 0. \quad (D'_C)
\end{align*}
\]

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

When $0 < |I_-| \leq |I_+|$, $\overline{\nu} = 2|I_-|/|I|$. Then $RCH_-(\overline{\nu}) = \{\sum_{i \in I_-} \frac{1}{|I|} x_i\}$ holds.

Here, we show the following statements for a training set are equivalent.

1. The training set $\{(x_i, y_i)\}_{i \in I}$ satisfies $RCH_-(\overline{\nu}) \subseteq RCH_+(\overline{\nu})$.
2. $(D_\nu)$ has an optimal solution such that $\sum_{i \in I} \lambda_i y_i x_i = 0$ for all $\nu \in (\overline{\nu}, -\overline{\nu}]$.
3. $(D'_C)$ has an optimal solution such that $\sum_{i \in I} \lambda_i y_i x_i = 0$ for all $C \in (0, \infty)$.

Statements c2 and c3 imply that $\nu$-SVM and $C$-SVM obtain a trivial solution such that $w = \sum_{i \in I} \lambda_i y_i x_i = 0$ for any hyperparameter value.

The equivalence of statements c1 and c2 can be seen by making a geometric interpretation of $\nu$-SVM. From the result of Crisp and Burges (2000), $(D_\nu)$ can be described as
\[
\min_{x_+ \in \text{RCH}_+(\nu), \ x_- \in \text{RCH}_-(\nu)} \| x_+ - x_- \|^2. \quad (C.1)
\]

By using an appropriate scaling: \( \tilde{\lambda}_i = 2\lambda_i / \nu \) (\( D_\nu \)) and equation C.1 has the same set of optimal solutions. We denote the optimal solution of equation C.1 as \((x^*_+, x^*_-)\). \( x^*_\pm \) is represented using the optimal \( \tilde{\lambda}^*_\pm \), as
\[
x^*_\pm = \sum_{i \in I_\pm} \tilde{\lambda}^*_i x_i.
\]
Then \( x^*_+ = x^*_- \) if and only if
\[
\sum_{i \in I} \tilde{\lambda}^*_i y_i x_i = 0.
\]

**c1 \( \Rightarrow \) c2:** If the training set \( \{x_i, y_i\}_{i \in I} \) satisfies \( \text{RCH}^-_-(\nu) \subseteq \text{RCH}^+_+(\nu) \), then \( \text{RCH}^+_+(\nu) \cap \text{RCH}^-_-(\nu) \neq \emptyset \) for all \( \nu \in (\nu, \nu] \). Therefore, c2 holds.

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

The equivalence of statements c2 and c3 is shown by using the result from Chang and Lin (2001b):

**c2 \( \Rightarrow \) c3:** We show this by taking the contraposition. Suppose the optimal solution \( \lambda^{(C)} \) of \( (D^{(C)}_C) \) satisfies \( \sum_{i \in I} \lambda^{(C)}_i y_i x_i \neq 0 \) for some hyper-parameter \( C \in (0, \infty) \). Then \( 0 < \sum_{i \in I} \lambda^{(C)}_i \), since the optimal value of \( (D_\nu) \) is zero or negative. From Chang and Lin (2001b, theorem 3), \( \lambda^{(C)} \) is also an optimal solution of \( (D_\nu) \) with \( \nu = \sum_{i \in I} \lambda^{(C)}_i \).

**c3 \( \Rightarrow \) c2:** We show this as well by using contraposition. Suppose the optimal solution \( \lambda^{(\nu)} \) of \( (D_\nu) \) satisfies \( \sum_{i \in I} \lambda^{(\nu)}_i y_i x_i \neq 0 \) for some hyper-parameter \( \nu \in (0, 1] \). Then, the optimal value of \( (D_\nu) \) is positive. From Chang and Lin (2001b, theorem 4), \( (D_\nu) \)'s optimal solution set is the same as that of at least one \( (D^{(C)}_C) \) if the optimal value of \( (D_\nu) \) is positive.

---

**References**


Received October 4, 2016; accepted January 15, 2017.