Distributionally Robust Safe Screening

Hiroyuki Hanada*† Satoshi Akahane‡ Tatsuya Aoyama‡ Tomonari Tanaka‡
Yoshito Okura‡ Yu Inatsu§ Noriaki Hashimoto* Taro Murayama¶ Lee Hanju¶
Shinya Kojima¶ Ichiro Takeuchi‡*‖

April 26, 2024

Abstract

In this study, we propose a method *Distributionally Robust Safe Screening (DRSS)*, for identifying unnecessary samples and features within a DR covariate shift setting. This method effectively combines DR learning, a paradigm aimed at enhancing model robustness against variations in data distribution, with safe screening (SS), a sparse optimization technique designed to identify irrelevant samples and features prior to model training. The core concept of the DRSS method involves reformulating the DR covariate-shift problem as a weighted empirical risk minimization problem, where the weights are subject to uncertainty within a predetermined range. By extending the SS technique to accommodate this weight uncertainty, the DRSS method is capable of reliably identifying unnecessary samples and features under any future distribution within a specified range. We provide a theoretical guarantee of the DRSS method and validate its performance through numerical experiments on both synthetic and real-world datasets.

1 Introduction

In this study, we consider the problem of identifying unnecessary samples and features in a class of supervised learning problems within dynamically changing environments. Identifying unnecessary samples/features offers several benefits. It helps in decreasing the storage space required for keeping the training data for updating the machine learning (ML) models in the future. Moreover, in situations demanding real-time adaptation of ML models to quick environmental changes, the use of fewer samples/features enables more efficient learning.

Our basic idea to tackle this problem is to effectively combine *distributionally robust (DR)* learning and *safe screening (SS)*. DR learning is a ML paradigm that focuses on developing models robust to variations in the data distribution, providing performance guarantees across different distributions (see, e.g., [1]). On the other hand, SS refers to sparse optimization techniques that can identify irrelevant samples/features before model training, ensuring computational efficiency by avoiding unnecessary computations on certain samples/features which do not contribute to the final solution [2, 3]. The key technical idea of SS is to identify a bound of the optimal solution before solving the optimization problem. This allows for the identification of unnecessary samples/features, even without knowing the optimal solution.

As a specific scenario of dynamically changing environment, we consider *covariate shift* setting [4, 5] with unknown test distribution. In this setting, the

*RIKEN, Wako, Saitama, Japan
†hiroyuki.hanada@riken.jp
‡Nagoya University, Nagoya, Aichi, Japan
§Nagoya Institute of Technology, Nagoya, Aichi, Japan
¶DENSO CORPORATION, Kariya, Aichi, Japan
‖ichiro.takeuchi@mae.nagoya-u.ac.jp

distribution of input features in the training data may undergo changes in the test phase, yet the actual nature of these changes remains unknown. A ML problem (e.g., regression/classification problem) in covariate shift setting can be formulated as a *weighted empirical risk minimization (weighted ERM)* problem, where weights are assigned based on the density ratio of each sample in the training and test distributions. Namely, by assigning higher weights to training samples that are important in the test distribution, the model can focus on learning from relevant samples and mitigate the impact of distribution differences between the training and the test phases. If the distribution during the test phase is known, the weights can be uniquely fixed. However, if the test distribution is unknown, it is necessary to solve a weighted ERM problem with unknown weights.

Our main contribution is to propose a DRSS method for covariate shift setting with unknown test distribution. The proposed method can identify unnecessary samples/features regardless of how the distribution changes within a certain range in the test phase. To address this problem, we extend the existing SS methods in two stages. The first is to extend the SS for ERM so that it can be applied to weighted ERM. The second is to further extend the SS so that it can be applied to weighted ERM when the weights are unknown. While the first extension is relatively straightforward, the second extension presents a non-trivial technical challenge (Figure 1). To overcome this challenge, we derive a novel bound of the optimal solutions of the weighted ERM problem, which properly accounts for the uncertainty in weights stemming from the uncertainty of the test distribution.

In this study, we consider DRSS for samples in sample-sparse models such as SVM [6], and that for features for feature-sparse models such as Lasso [7]. We denote the DRSS for samples as distributionally robust safe *sample* screening (DRSsS) and that for features as distributionally robust safe *feature* screening (DRSfS), respectively.

Our contributions in this study are summarized as follows. First, by effectively combining DR and SS, we introduce a framework for identifying unnecessary samples/features under dynamically changing uncertain environment. Second, We consider a DR covariate-shift setting where the input distribution of an ERM problem changes within a certain range. In this setting, we propose a novel method called DRSS method that can identify samples/features that are guaranteed not to affect the optimal solution, regardless of how the distribution changes within the specified range. Finally, through numerical experiments, we verify the effectiveness of the proposed DRSS method. Although the DRSS method is developed for convex ERM problems, in order to demonstrate the applicability to deep learning models, we also present results where the DRSS method is applied in a problem setting where the final layer of the model is fine-tuned according to changes in the test distribution.

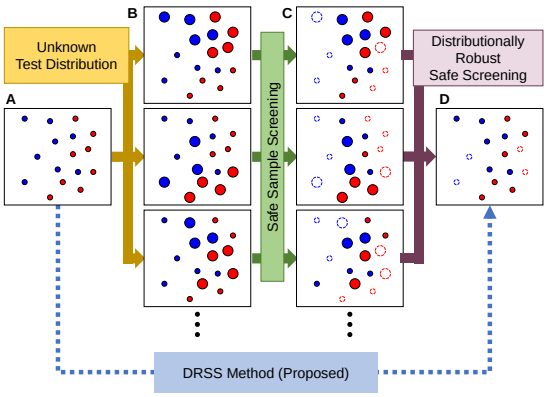

Figure 1: Schematic illustration of the proposed Distributionally Robust Safe Screening (DRSS) method. Panel **A** displays the training samples, each assigned equal weight, as indicated by the uniform size of the points. Panel **B** depicts various unknown test distributions, highlighting how the significance of training samples varies with different realizations of the test distribution. Panel **C** shows the outcomes of safe sample screening (SsS) across multiple realizations of test distributions. Finally, Panel **D** presents the results of the proposed DRSS method, demonstrating its capability to identify redundant samples regardless of the observed test distribution.

1.1 Related Works

The DR setting has been explored in various ML problems, aiming to enhance model robustness against data distribution variations. A DR learning problem is typically formulated as a worst-case optimization problem since the goal of DR learning is to ensure model performance under the worst-case data distribution within a specified range. Hence, a variety of optimization techniques tailored to DR learning have been investigated within both the ML and optimization communities [8, 9, 1]. The proposed DRSS method is one of such DR learning methods, focusing specifically on the problem of sample/feature deletion. The ability to identify irrelevant samples/features is of practical significance. For example, in the context of continual learning (see, e.g., [10]), it is crucial to effectively manage data by selectively retaining and discarding samples/features, especially in anticipation of changes in future data distributions. Incorrect deletion of essential data can lead to *catastrophic forgetting* [11], a phenomenon where a ML model, after being trained on new data, quickly loses information previously learned from older datasets. The proposed DRSS method tackles this challenge by identifying samples/features that, regardless of future data distribution shifts, will not have any influence on all possible newly trained model in the future.

SS refers to optimization techniques in sparse learning that identify and exclude irrelevant samples or features from the learning process. SS can reduce computational cost without changing the final trained model. Initially, SfS was introduced by [2] for the Lasso. Subsequently, SsS was proposed by [3] for the SVM. Among various SS methods developed so far, the most commonly used is based on the duality gap [12, 13]. Our proposed DRSS method also adopts this approach. Over the past decade, SS has seen diverse developments, including methodological improvements and expanded application scopes [14, 15, 16, 17, 18, 19, 20, 21]. Unlike other SS studies that primarily focused on reducing computational costs, this study adopts SS for a different purpose. We employ SS across scenarios where data distribution varies within a defined range, aiming to discard unnecessary samples/features. To our knowledge, no existing studies have utilized SS within the DR learning framework.

Table 1: Notations used in the paper. $\mathbb{R}$: all real numbers, $\mathbb{N}$: all positive integers, $n, m, p \in \mathbb{N}$: integers, $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$: convex function, $M \in \mathbb{R}^{n\times m}$: matrix, $\boldsymbol{v} \in \mathbb{R}^n$: vector.

$m_{ij} \in \mathbb{R}$ (small case of matrix variable)	the element at the i^{th} row and the j^{th} column of M
$v_i \in \mathbb{R}$ (nonbold font of vector variable)	the i^{th} element of $\boldsymbol{v}$
$M_{i:} \in \mathbb{R}^{1\times n}$	the i^{th} row of M
$M_{:j} \in \mathbb{R}^{m\times 1}$	the j^{th} column of M
$[n]$	$\{1, 2, \ldots, n\}$
$\mathbb{R}_{\geq 0}$	all nonnegative real numbers
$\otimes$	elementwise product
$\mathrm{diag}(\boldsymbol{v}) \in \mathbb{R}^{n\times n}$	diagonal matrix; $(\mathrm{diag}(\boldsymbol{v}))_{ii} = v_i$ and $(\mathrm{diag}(\boldsymbol{v}))_{ij} = 0$ $(i \neq j)$
$\boldsymbol{v}\boxtimes M \in \mathbb{R}^{n\times m}$	$\mathrm{diag}(\boldsymbol{v})M$
$\mathbf{0}_n \in \mathbb{R}^n$	$[0, 0, \ldots, 0]^\top$ (vector of size n)
$\mathbf{1}_n \in \mathbb{R}^n$	$[1, 1, \ldots, 1]^\top$ (vector of size n)
$\|\boldsymbol{v}\|_p \in \mathbb{R}_{\geq 0}$	$(\sum_{i=1}^n v_i^p)^{1/p}$ (*p-norm*)
$\partial f(\boldsymbol{v}) \subseteq \mathbb{R}^n$	all $\boldsymbol{g} \in \mathbb{R}^n$ s.t. "for any $\boldsymbol{v}' \in \mathbb{R}^n$, $f(\boldsymbol{v}') - f(\boldsymbol{v}) \geq \boldsymbol{g}^\top(\boldsymbol{v}' - \boldsymbol{v})$" (*subgradient*)
$\mathcal{Z}[f] \subseteq \mathbb{R}^n$	$\{\boldsymbol{v}' \in \mathbb{R}^n \mid \partial f(\boldsymbol{v}') = \{\mathbf{0}_n\}\}$
$f^*(\boldsymbol{v}) \in \mathbb{R} \cup \{+\infty\}$	$\sup_{\boldsymbol{v}'\in\mathbb{R}^n}(\boldsymbol{v}^\top\boldsymbol{v}' - f(\boldsymbol{v}'))$ (*convex conjugate*)
"f is κ-strongly convex" $(\kappa > 0)$	$f(\boldsymbol{v}) - \kappa\|\boldsymbol{v}\|_2^2$ is convex with respect to $\boldsymbol{v}$
"f is μ-smooth" $(\mu > 0)$	$\|f(\boldsymbol{v}) - f(\boldsymbol{v}')\|_2 \leq \mu\|\boldsymbol{v} - \boldsymbol{v}'\|_2$ for any $\boldsymbol{v}, \boldsymbol{v}' \in \mathbb{R}^n$

2 Preliminaries

Notations used in this paper are described in Table 1.

2.1 Weighted Regularized Empirical Risk Minimization (Weighted RERM) for Linear Prediction

We mainly assume the weighted regularized empirical risk minimization (weighted RERM) for linear prediction. This may include kernelized versions, which are discussed in Appendix C. Suppose that we learn the model parameters as linear prediction coefficients, that is, learn $\boldsymbol{\beta}^{*(\boldsymbol{w})} \in \mathbb{R}^d$ such that the outcome for a sample $\boldsymbol{x} \in \mathbb{R}^d$ is predicted as $\boldsymbol{x}^\top \boldsymbol{\beta}^{*(\boldsymbol{w})}$.

Definition 2.1. Given n training samples of d-dimensional input variables, scalar output variables and scalar sample weights, denoted by $X \in \mathbb{R}^{n \times d}$, $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{w} \in \mathbb{R}^n_{\geq 0}$, respectively, the training computation of weighted RERM for linear prediction is formulated as follows:

$$\boldsymbol{\beta}^{*(\boldsymbol{w})} := \operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^d} P_{\boldsymbol{w}}(\boldsymbol{\beta}), \quad \text{where}$$
$$P_{\boldsymbol{w}}(\boldsymbol{\beta}) := \sum_{i=1}^n w_i \ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}) + \rho(\boldsymbol{\beta}). \tag{1}$$

Here, $\ell_y : \mathbb{R} \to \mathbb{R}$ is a convex *loss function*[1], $\rho : \mathbb{R}^d \to \mathbb{R}$ is a convex *regularization function*, and $\check{X} \in \mathbb{R}^{n \times d}$ is a matrix calculated from X and $\boldsymbol{y}$ and determined depending on ℓ. In this paper, unless otherwise noted, we consider binary classifications ($\boldsymbol{y} \in \{-1, +1\}^n$) with $\check{X} := \boldsymbol{y} \boxtimes X$. For regressions ($\boldsymbol{y} \in \mathbb{R}^n$) we usually set $\check{X} := X$.

Remark 2.2. We add that, we adopt the formulation $X_{:d} = \mathbf{1}_n$ so that $\beta_d^{*(\boldsymbol{w})}$ (the last element) represents the common coefficient for any sample (called the *intercept*).

Since ℓ and ρ are convex, we can easily confirm that $P_{\boldsymbol{w}}(\boldsymbol{\beta})$ is convex with respect to $\boldsymbol{\beta}$.

Applying *Fenchel's duality theorem* (Appendix A.2), we have the following *dual problem* of (1):

$$\boldsymbol{\alpha}^{*(\boldsymbol{w})} := \operatorname*{argmax}_{\boldsymbol{\alpha} \in \mathbb{R}^n} D_{\boldsymbol{w}}(\boldsymbol{\alpha}), \quad \text{where}$$
$$D_{\boldsymbol{w}}(\boldsymbol{\alpha}) := \tag{2}$$
$$-\sum_{i=1}^n w_i \ell^*_{y_i}(-\gamma_i \alpha_i) - \rho^*(((\boldsymbol{\gamma} \otimes \boldsymbol{w}) \boxtimes \check{X})^\top \boldsymbol{\alpha}),$$

where $\boldsymbol{\gamma}$ is a positive-valued vector. The relationship between the original problem (1) (called the *primal* problem) and the dual problem (2) are described as follows:

$$P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\boldsymbol{w})}) = D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\boldsymbol{w})}), \tag{3}$$
$$\boldsymbol{\beta}^{*(\boldsymbol{w})} \in \partial \rho^*(((\boldsymbol{\gamma} \otimes \boldsymbol{w}) \boxtimes \check{X})^\top \boldsymbol{\alpha}^{*(\boldsymbol{w})}), \tag{4}$$
$$\forall i \in [n] : \quad -\gamma_i \alpha_i^{*(\boldsymbol{w})} \in \partial \ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}). \tag{5}$$

2.2 Sparsity-inducing Loss Functions and Regularization Functions

In weighted RERM, we call that a loss function ℓ induces *sample-sparsity* if elements in $\boldsymbol{\alpha}^{*(\boldsymbol{w})}$ are easy to become zero. Due to (5), this can be achieved by ℓ such that $\{t \in \mathbb{R} \mid 0 \in \partial \ell_y(t)\}$ is not a point but an interval.

Similarly, we call that a regularization function ρ induces *feature-sparsity* if elements in $\boldsymbol{\beta}^{*(\boldsymbol{w})}$ are easy to become zero. Due to (4), this can be achieved by ρ such that $\{\boldsymbol{v} \in \mathbb{R}^d \mid \exists j \in [d-1] :\ 0 \in [\partial \rho^*(\boldsymbol{v})]_j\}$ is not a point but a region.

For example, the *hinge loss* $\ell_y(t) = \max\{0, 1 - t\}$ ($y \in \{-1, +1\}$) is a sample-sparse loss function since $\{t \in \mathbb{R} \mid 0 \in \partial \ell_y(t)\} = [1, +\infty)$. Similarly, the *L1-regularization* $\rho(\boldsymbol{v}) = \lambda \sum_{j=1}^{d-1} |v_j|$ ($\lambda > 0$: hyperparameter) is a feature-sparse regularization function since $\{\boldsymbol{v} \in \mathbb{R}^d \mid \exists j \in [d-1] :\ 0 \in [\partial \rho^*(\boldsymbol{v})]_j\} = \{\boldsymbol{v} \in \mathbb{R}^d \mid \exists j \in [d-1] :\ |v_j| \leq \lambda,\ v_d = 0\}$. See Section 4 for examples of using them.

3 Distributionally Robust Safe Screening

In this section we show DRSS rules for weighted RERM with two steps. First, in Sections 3.1 and

[1] For $\ell_y(t)$, we assume that only t is a variable of the function (y is assumed to be a constant) when we take its subgradient or convex conjugate.

3.2, we show SS rules for weighted RERM but not DR setup. To do this, we extended existing SS rules in [13, 15]. Then we derive DRSS rules in Section 3.3.

3.1 (Non-DR) Safe Sample Screening

We consider identifying training samples that do not affect the training result $\boldsymbol{\beta}^{*(\boldsymbol{w})}$. Due to the relationship (4), if there exists $i \in [n]$ such that $\alpha_i^{*(\boldsymbol{w})} = 0$, then the i^{th} row (sample) in $\check{X}$ does not affect $\boldsymbol{\beta}^{*(\boldsymbol{w})}$. However, since computing $\boldsymbol{\alpha}^{*(\boldsymbol{w})}$ is as costly as $\boldsymbol{\beta}^{*(\boldsymbol{w})}$, it is difficult to use the relationship as it is. To solve the problem, the SsS first considers identifying the possible region $\mathcal{B}^{*(\boldsymbol{w})} \subset \mathbb{R}^d$ such that $\boldsymbol{\beta}^{*(\boldsymbol{w})} \in \mathcal{B}^{*(\boldsymbol{w})}$ is assured. Then, with $\mathcal{B}^{*(\boldsymbol{w})}$ and (5), we can conclude that the i^{th} training sample do not affect the training result $\boldsymbol{\beta}^{*(\boldsymbol{w})}$ if $\bigcup_{\boldsymbol{\beta} \in \mathcal{B}^{*(\boldsymbol{w})}} \partial \ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}) = \{0\}$.

First we show how to compute $\mathcal{B}^{*(\boldsymbol{w})}$. In this paper we adopt the computation methods that is available when the regularization function ρ in $P_{\boldsymbol{w}}$ (and also $P_{\boldsymbol{w}}$ itself) of (1) are strongly convex.

Lemma 3.1. *Suppose that ρ in $P_{\boldsymbol{w}}$ (and also $P_{\boldsymbol{w}}$ itself) of (1) are κ-strongly convex. Then, for any $\hat{\boldsymbol{\beta}} \in \mathbb{R}^d$ and $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^n$, we can assure $\boldsymbol{\beta}^{*(\boldsymbol{w})} \in \mathcal{B}^{*(\boldsymbol{w})}$ by taking*

$$\mathcal{B}^{*(\boldsymbol{w})} := \left\{ \boldsymbol{\beta} \;\middle|\; \|\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}\|_2 \leq r(\boldsymbol{w}, \boldsymbol{\gamma}, \kappa, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\alpha}}) \right\},$$
$$\text{where} \quad r(\boldsymbol{w}, \boldsymbol{\gamma}, \kappa, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\alpha}}) := \sqrt{\frac{2}{\kappa}[P_{\boldsymbol{w}}(\hat{\boldsymbol{\beta}}) - D_{\boldsymbol{w}}(\hat{\boldsymbol{\alpha}})]}.$$

The proof is presented in Appendix A.3. The amount $P_{\boldsymbol{w}}(\hat{\boldsymbol{\beta}}) - D_{\boldsymbol{w}}(\hat{\boldsymbol{\alpha}})$ is known as the *duality gap*, which must be nonnegative due to (3). So we obtain the following *gap safe sample screening rule* from Lemma 3.1:

Lemma 3.2. *Under the same assumptions as Lemma 3.1, $\alpha_i^{*(\boldsymbol{w})} = 0$ is assured (i.e., the i^{th} training sample does not affect the training result $\boldsymbol{\beta}^{*(\boldsymbol{w})}$) if there exists $\hat{\boldsymbol{\beta}} \in \mathbb{R}^d$ and $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^n$ such that*

$$[\check{X}_{i:}\hat{\boldsymbol{\beta}} - \|\check{X}_{i:}\|_2 r(\boldsymbol{w}, \boldsymbol{\gamma}, \kappa, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\alpha}}),$$
$$\check{X}_{i:}\hat{\boldsymbol{\beta}} + \|\check{X}_{i:}\|_2 r(\boldsymbol{w}, \boldsymbol{\gamma}, \kappa, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\alpha}})] \subseteq \mathcal{Z}[\ell_{y_i}].$$

The proof is presented in Appendix A.4.

3.2 (Non-DR) Safe Feature Screening

We consider identifying $j \in [d]$ such that $\beta_j^{*(\boldsymbol{w})} = 0$, that is, identifying that the j^{th} feature is not used in the prediction, even when the sample weights $\boldsymbol{w}$ are changed.

For simplicity, suppose that the regularization function ρ is decomposable, that is, ρ is represented as $\rho(\boldsymbol{\beta}) := \sum_{j=1}^d \sigma_j(\beta_j)$ ($\sigma_1, \sigma_2, \ldots, \sigma_d$: $\mathbb{R} \to \mathbb{R}$). Then, since $\rho^*(\boldsymbol{v}) = \sum_{j=1}^d \sigma_j^*(v_j)$ and therefore $[\partial \rho^*(\boldsymbol{v})]_j = \partial \sigma_j^*(v_j)$, from (4) we have

$$\begin{aligned} \beta_j^{*(\boldsymbol{w})} &\in \partial \sigma_j^*((\boldsymbol{\gamma} \otimes \boldsymbol{w} \otimes \check{X}_{:j})^\top \boldsymbol{\alpha}^{*(\boldsymbol{w})}) \\ &= \partial \sigma_j^*(\check{\check{X}}_{:j}^{(\boldsymbol{\gamma}, \boldsymbol{w})\top} \boldsymbol{\alpha}^{*(\boldsymbol{w})}), \\ \text{where} \quad \check{\check{X}}_{:j}^{(\boldsymbol{\gamma}, \boldsymbol{w})} &:= \boldsymbol{\gamma} \otimes \boldsymbol{w} \otimes \check{X}_{:j}. \end{aligned}$$

If we know $\boldsymbol{\alpha}^{*(\boldsymbol{w})}$, we can identify whether $\beta_j^{*(\boldsymbol{w})} = 0$ holds. However, like SsS (Section 3.1), we would like to check the condition without computing $\boldsymbol{\alpha}^{*(\boldsymbol{w})}$ or $\boldsymbol{\beta}^{*(\boldsymbol{w})}$.

So, like SsS, SfS first considers identifying the possible region $\mathcal{A}^{*(\boldsymbol{w})} \subset \mathbb{R}^n$ such that $\boldsymbol{\alpha}^{*(\boldsymbol{w})} \in \mathcal{A}^{*(\boldsymbol{w})}$ is assured. Then we can conclude that $\beta_j^{*(\boldsymbol{w})} = 0$ is assured if $\bigcup_{\boldsymbol{\alpha} \in \mathcal{A}^{*(\boldsymbol{w})}} \partial \sigma_j^*(\check{\check{X}}_{:j}^{(\boldsymbol{\gamma}, \boldsymbol{w})\top} \boldsymbol{\alpha}) = \{0\}$.

Then we show how to compute $\mathcal{A}^{*(\boldsymbol{w})}$. With Lemma A.3, we can calculate $\mathcal{A}^{*(\boldsymbol{w})}$ as follows, if the loss function ℓ_y in $P_{\boldsymbol{w}}$ of (1) is smooth:

Lemma 3.3. *Suppose that ℓ_y in $P_{\boldsymbol{w}}$ of (1) is μ-smooth. Then, for any $\hat{\boldsymbol{\beta}} \in \mathbb{R}^d$ and $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^n$, we can assure $\boldsymbol{\alpha}^{*(\boldsymbol{w})} \in \mathcal{A}^{*(\boldsymbol{w})}$ by taking*

$$\mathcal{A}^{*(\boldsymbol{w})} := \left\{ \boldsymbol{\alpha} \;\middle|\; \|\boldsymbol{\alpha} - \hat{\boldsymbol{\alpha}}\|_2 \leq \bar{r}(\boldsymbol{w}, \boldsymbol{\gamma}, \mu, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\alpha}}) \right\},$$
$$\text{where} \quad \bar{r}(\boldsymbol{w}, \boldsymbol{\gamma}, \mu, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\alpha}}) := \sqrt{\frac{2\mu}{\min_{i \in [n]} w_i \gamma_i^2}[P_{\boldsymbol{w}}(\hat{\boldsymbol{\beta}}) - D_{\boldsymbol{w}}(\hat{\boldsymbol{\alpha}})]}.$$

The proof is presented in Appendix A.5. Similar to Lemma 3.2, we obtain the *gap safe feature screening rule* from Lemma 3.3:

Lemma 3.4. *Under the same assumptions as Lemma 3.3, $\beta_j^{*(\boldsymbol{w})} = 0$ is assured (i.e., the j^{th} feature does*

not affect prediction results) if there exists $\hat{\boldsymbol{\beta}} \in \mathbb{R}^d$ *and* $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^n$ *such that*

$$[\check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})\top}\hat{\boldsymbol{\alpha}} - \|\check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})}\|_2 \bar{r}(\boldsymbol{w},\boldsymbol{\gamma},\mu,\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\alpha}}),$$
$$\check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})\top}\hat{\boldsymbol{\alpha}} + \|\check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})}\|_2 \bar{r}(\boldsymbol{w},\boldsymbol{\gamma},\mu,\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\alpha}})] \subseteq \mathcal{Z}[\sigma_j^*].$$

The proof is almost same as Lemma 3.2.

3.3 Application to Distributionally Robust Setup

In Sections 3.1 and 3.2 we showed the conditions when samples or features are screened out. In this section we show how to use the conditions for the change of sample weights $\boldsymbol{w}$.

Definition 3.5 (weight-changing safe screening (WCSS))**.** Given $X \in \mathbb{R}^{n\times d}$, $\boldsymbol{y} \in \mathbb{R}^n$, $\tilde{\boldsymbol{w}} \in \mathbb{R}^n_{\geq 0}$ and $\boldsymbol{w} \in \mathbb{R}^n_{\geq 0}$, suppose that $\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$ in Definition 2.1 (and also $\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}$) are already computed, but $\boldsymbol{\beta}^{*(\boldsymbol{w})}$ not. Then *WCSsS (resp. WCSfS) from* $\tilde{\boldsymbol{w}}$ *to* $\boldsymbol{w}$ is defined as finding $i \in [n]$ satisfying Lemma 3.2 (resp. $j \in [d-1]$ satisfying Lemma 3.4).

Definition 3.6 (Distributionally robust safe screening (DRSS))**.** Given $X \in \mathbb{R}^{n\times d}$, $\boldsymbol{y} \in \mathbb{R}^n$, $\tilde{\boldsymbol{w}} \in \mathbb{R}^n_{\geq 0}$ and $\mathcal{W} \subset \mathbb{R}^n_{\geq 0}$, suppose that $\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$ in Definition 2.1 (and also $\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}$) are already computed. Then the *DRSsS (resp. DRSfS) for* $\mathcal{W}$ is defined as finding $i \in [n]$ satisfying Lemma 3.2 (resp. $j \in [d-1]$ satisfying Lemma 3.4) for any $\boldsymbol{w} \in \mathcal{W}$.

For Definition 3.5, we have only to apply SS rules in Lemma 3.2 or 3.4 by setting $\hat{\boldsymbol{\beta}} \leftarrow \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$ and $\hat{\boldsymbol{\alpha}} \leftarrow \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}$. On the other hand, for Definition 3.6, we need to maximize or minimize the interval in Lemma 3.2 or 3.4 in $\boldsymbol{w} \in \mathcal{W}$.

Theorem 3.7. *The DRSsS rule for* $\mathcal{W}$ *is calculated as:*

$$[\check{X}_{i:}\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})} - \|\check{X}_{i:}\|_2 R, \check{X}_{i:}\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})} + \|\check{X}_{i:}\|_2 R] \subseteq \mathcal{Z}[\ell_{y_i}],$$

where $R := \max_{\boldsymbol{w}\in\mathcal{W}} r(\boldsymbol{w},\boldsymbol{\gamma},\kappa,\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})},\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})$.

Similarly, the DRSfS rule for $\mathcal{W}$ *is calculated as:*

$$[\underline{L} - N\overline{R}, \overline{L} + N\overline{R}] \subseteq \mathcal{Z}[\sigma_j^*], \quad \textit{where}$$
$$\underline{L} := \min_{\boldsymbol{w}\in\mathcal{W}} \check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})\top}\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})} = \min_{\boldsymbol{w}\in\mathcal{W}} (\boldsymbol{\gamma}\otimes\check{X}_{:j}\otimes\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})^\top \boldsymbol{w},$$
$$\overline{L} := \max_{\boldsymbol{w}\in\mathcal{W}} \check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})\top}\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})} = \max_{\boldsymbol{w}\in\mathcal{W}} (\boldsymbol{\gamma}\otimes\check{X}_{:j}\otimes\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})^\top \boldsymbol{w},$$
$$N := \max_{\boldsymbol{w}\in\mathcal{W}} \|\check{\check{X}}_{:j}^{(\boldsymbol{\gamma},\boldsymbol{w})}\|_2 = \sqrt{\max_{\boldsymbol{w}\in\mathcal{W}} \|\boldsymbol{w}\otimes\boldsymbol{\gamma}\otimes\check{X}_{:j}\|_2^2},$$
$$\overline{R} := \max_{\boldsymbol{w}\in\mathcal{W}} \bar{r}(\boldsymbol{w},\boldsymbol{\gamma},\mu,\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})},\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}).$$

Thus, solving the maximizations and/or minimizations in Theorem 3.7 provides DRSsS and DRSfS rules. However, how to solve it largely depends on the choice of ℓ, ρ and $\mathcal{W}$. In Section 4 we show specific calculations of Theorem 3.7 for some typical setups.

4 DRSS for Typical ML Setups

In this section we show DRSS rules derived in Section 3.3 for two typical ML setups: DRSsS for L1-loss L2-regularized SVM (Section 4.1) and DRSfS for L2-loss L1-regularized SVM (Section 4.2) under $\mathcal{W} := \{\boldsymbol{w} \mid \|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2 \leq S\}$.

In the processes, we need to solve constrained maximizations of convex functions. Although maximizations of convex functions are not easy in general (minimizations are easy), we show that the maximizations need in the processes can be algorithmically solved in Section 4.3.

4.1 DRSsS for L1-loss L2-regularized SVM

L1-loss L2-regularized SVM is a sample-sparse model for binary classification ($\boldsymbol{y} \in \{-1,+1\}^n$) that satisfies the preconditions to apply SsS (Lemma 3.1). Detailed calculations are presented in Appendix B.1.

For L1-loss L2-regularized SVM, we set ρ and ℓ as:

$$\rho(\boldsymbol{\beta}) := \frac{\lambda}{2}\|\boldsymbol{\beta}\|_2^2 \quad (\lambda > 0 : \text{ hyperparameter}),$$
$$\ell_y(t) := \max\{0, 1-t\} \quad (\text{where } y \in \{-1,+1\}).$$

Then ρ is λ-strongly convex. Setting $\boldsymbol{\gamma} = \mathbf{1}_n$, the dual objective function is described as

$$D_{\boldsymbol{w}}(\boldsymbol{\alpha}) = \begin{cases} \sum_{i=1}^n w_i\alpha_i - \frac{1}{2\lambda}\boldsymbol{\alpha}^\top(\boldsymbol{w}\boxtimes\check{X})(\boldsymbol{w}\boxtimes\check{X})^\top\boldsymbol{\alpha}, \\ \qquad\qquad (\forall i \in [n] : 0 \leq \alpha_i \leq 1) \\ -\infty. \qquad\qquad (\text{otherwise}) \end{cases} \tag{6}$$

Here, in the viewpoint of minimization, we may consider this problem as a maximization with the constraint "$\forall i \in [n] : 0 \leq \alpha_i \leq 1$".

Optimality conditions (4) and (5) are described as:

$$\boldsymbol{\beta}^{*(\boldsymbol{w})} = \frac{1}{\lambda}(\boldsymbol{w}\boxtimes\check{X})^\top\boldsymbol{\alpha}^{*(\boldsymbol{w})}, \tag{7}$$

$$\forall i \in [n] : \quad \alpha_i^{*(\boldsymbol{w})} \in \begin{cases} \{1\}, & (\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})} \leq 1) \\ [0,1], & (\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})} = 1) \\ \{0\}. & (\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})} \geq 1) \end{cases} \tag{8}$$

Noticing that $\mathcal{Z}[\ell_{y_i}] = (1, +\infty)$, by Theorem 3.7, the DRSsS rule for $\mathcal{W}$ is calculated as:

$$\check{X}_{i:}\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})} - \|\check{X}_{i:}\|_2 \max_{\boldsymbol{w}\in\mathcal{W}} r(\boldsymbol{w}, \boldsymbol{\gamma}, \kappa, \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}, \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}) > 1, \tag{9}$$

where

$$r(\boldsymbol{w}, \boldsymbol{\gamma}, \kappa, \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}, \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}) := \sqrt{\frac{2}{\kappa}[P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})]},$$

$$P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}) := \sum_{i=1}^n w_i[\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - \alpha_i^{*(\tilde{\boldsymbol{w}})}] + \lambda\|\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}\|_2^2 + \frac{1}{2\lambda}\boldsymbol{w}^\top(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}\boxtimes\check{X})(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}\boxtimes\check{X})^\top\boldsymbol{w}.$$

Here, we can find that $P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})$, which we need to maximize in reality, is the sum of linear function and convex quadratic function with respect to $\boldsymbol{w} \in \mathcal{W}$. (Since $(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}\boxtimes\check{X})(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}\boxtimes\check{X})^\top$ is positive semidefinite, we know that it is convex quadratic). Although constrained maximization of a convex function is difficult in general, for this case we can algorithmically maximize it (Section 4.3).

4.2 DRSfS for L2-loss L1-regularized SVM

L2-loss L1-regularized SVM is a feature-sparse model for binary classification ($\boldsymbol{y} \in \{-1, +1\}^n$) that satisfies the preconditions to apply SfS (Lemma 3.3). Detailed calculations are presented in Appendix B.2.

For L2-loss L1-regularized SVM, we set σ_j (and consequently ρ) and ℓ as:

$$\forall j \in [d-1] : \ \sigma_j(\beta_j) := \lambda|\beta_j| \quad (\lambda > 0 : \text{ hyperparameter}),$$
$$\sigma_d(\beta_d) := 0,$$
$$\ell_y(t) := (\max\{0, 1-t\})^2 \quad (\text{where } y \in \{-1, +1\}).$$

Notice that $\sigma_d(\beta_d)$ is not defined as $\lambda|\beta_d|$ but 0: we rarely regularize the intercept with L1-regularization.

Setting $\boldsymbol{\gamma} = \lambda\mathbf{1}_n$, the dual objective function is described as

$$D_{\boldsymbol{w}}(\boldsymbol{\alpha}) = \begin{cases} -\lambda\sum_{i=1}^n w_i\frac{\lambda\alpha_i^2 - 4\alpha_i}{4}, & ((11)\text{–}(13) \text{ are met}) \\ -\infty, & (\text{otherwise}) \end{cases} \tag{10}$$

$$\text{where} \quad \alpha_i \geq 0, \tag{11}$$

$$\forall j \in [d-1] : \ |(\boldsymbol{w}\otimes\check{X}_{:j})^\top\boldsymbol{\alpha}| \leq 1, \tag{12}$$

$$(\boldsymbol{w}\otimes\check{X}_{:d})^\top\boldsymbol{\alpha} = (\boldsymbol{w}\otimes\boldsymbol{y})^\top\boldsymbol{\alpha} = 0. \tag{13}$$

Optimality conditions (4) and (5) are described as

$$\forall j \in [d-1] : \ |(\boldsymbol{w}\otimes\check{X}_{:j})^\top\boldsymbol{\alpha}^{*(\boldsymbol{w})}| < 1 \Rightarrow \beta_j^{*(\boldsymbol{w})} = 0, \tag{14}$$

$$\forall i \in [n] : \quad \alpha_i^{*(\boldsymbol{w})} = \frac{2}{\lambda}\max\{0, 1 - \check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}\}. \tag{15}$$

Noticing that $\mathcal{Z}[\sigma_j^*] = (-\lambda, \lambda)$, by Theorem 3.7, the DRSfS rule for $\mathcal{W}$ is calculated as:

$$\underline{L} - N\overline{R} > -\lambda, \quad \overline{L} + N\overline{R} < \lambda,$$

where

$$\begin{aligned}
\underline{L} &:= \lambda \min_{\boldsymbol{w}\in\mathcal{W}} (\check{X}_{:j} \otimes \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})^\top \boldsymbol{w}, \\
\overline{L} &:= \lambda \max_{\boldsymbol{w}\in\mathcal{W}} (\check{X}_{:j} \otimes \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})^\top \boldsymbol{w}, \\
N &:= \lambda \sqrt{\max_{\boldsymbol{w}\in\mathcal{W}} \|\boldsymbol{w} \otimes \check{X}_{:j}\|_2^2} \\
&= \lambda \sqrt{\max_{\boldsymbol{w}\in\mathcal{W}} \{\boldsymbol{w}^\top \mathrm{diag}(\check{X}_{:j} \otimes \check{X}_{:j}) \boldsymbol{w}\}}, \\
\overline{R} &:= \max_{\boldsymbol{w}\in\mathcal{W}} \bar{r}(\boldsymbol{w}, \boldsymbol{\gamma}, \mu, \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}, \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}), \\
&\bar{r}(\boldsymbol{w}, \boldsymbol{\gamma}, \mu, \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}, \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}) \\
&:= \sqrt{\frac{2\mu}{\min_{i\in[n]} w_i \gamma_i^2} [P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})]},
\end{aligned}$$

$$\begin{aligned}
&P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}) \\
&= \sum_{i=1}^{n} w_i \left[\ell_{y_i}(\check{X}_{i:} \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) + \lambda \frac{\lambda (\alpha^{*(\tilde{\boldsymbol{w}})})_i^2 - 4\alpha_i^{*(\tilde{\boldsymbol{w}})}}{4} \right] \\
&\quad + \rho(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}).
\end{aligned}$$

Here, the expressions in $\underline{L}$ and $\overline{L}$ are linear with respect to $\boldsymbol{w}$, and the expression in N inside the square root is convex and quadratic with respect to $\boldsymbol{w}$. Also, $\overline{R}$ is decomposed to two maximizations $\frac{2\mu}{\min_{i\in[n]} w_i \gamma_i^2}$ and $P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\boldsymbol{w})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\boldsymbol{w})})$, where the former is easily computed while the latter is linear with respect to $\boldsymbol{w}$. So, similar to L1-loss L2-regularized SVM, we can obtain the maximization result by maximizing or minimizing the linear terms by Lemma A.4 in Appendix A, and maximizing the convex quadratic function by the method of Section 4.3.

4.3 Maximizing Linear and Convex Quadratic Functions in Hyperball Constraint

To derive DRSS rules of Sections 4.1 and 4.2, we need to compute the following forms of optimization problems:

$$\max_{\boldsymbol{w}\in\mathcal{W}} \boldsymbol{w}^\top A \boldsymbol{w} + 2\boldsymbol{b}^\top \boldsymbol{w}, \tag{16}$$

$$\begin{aligned}
\text{where} \quad & \mathcal{W} := \{\boldsymbol{w} \in \mathbb{R}^n \mid \|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2 \le S\}, \\
& \tilde{\boldsymbol{w}} \in \mathbb{R}^n, \quad \boldsymbol{b} \in \mathbb{R}^n, \\
& A \in \mathbb{R}^{n\times n} : \text{ symmetric, positive semidefinite, nonzero.}
\end{aligned}$$

Lemma 4.1. *The maximization* (16) *is achieved by the following procedure. First, we define $Q \in \mathbb{R}^{n\times n}$ and $\Phi := \mathrm{diag}(\phi_1, \phi_2, \ldots, \phi_n)$ as the eigendecomposition of A such that $A = Q^\top \Phi Q$, Q is orthogonal ($QQ^\top = Q^\top Q = I$). Also, let $\boldsymbol{\xi} := -\Phi Q \tilde{\boldsymbol{w}} - Q\boldsymbol{b} \in \mathbb{R}^n$, and*

$$\mathcal{T}(\nu) = \sum_{i=1}^{n} \left(\frac{\xi_i}{\nu - \phi_i} \right)^2. \tag{17}$$

Then, the maximization (16) *is equal to the largest value among them:*

- *For each ν such that $\mathcal{T}(\nu) = S^2$ (see Lemma 4.2), the value $\nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top Q^\top (\Phi - \nu I)^{-1} \boldsymbol{\xi} + \boldsymbol{b}^\top \tilde{\boldsymbol{w}}$, and*
- *For each $\nu \in \{\phi_1, \phi_2, \ldots, \phi_n\}$ (duplication removed) such that "$\forall i \in [n] : \ \phi_i = \nu \Rightarrow \xi_i = 0$", the value*

$$\begin{aligned}
&\max_{\boldsymbol{\tau}\in\mathbb{R}^n} [\nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top Q^\top \boldsymbol{\tau} + \boldsymbol{b}^\top \tilde{\boldsymbol{w}}], \\
&\textit{subject to} \quad \forall i \in \mathcal{F}_\nu : \quad \tau_i = \frac{\xi_i}{\phi_i - \nu}, \\
&\qquad\qquad \sum_{i\in\mathcal{U}_\nu} \tau_i^2 = S^2 - \sum_{i\in\mathcal{F}_\nu} \tau_i^2, \\
&\textit{where} \quad \mathcal{U}_\nu := \{i \mid i \in [n], \ \phi_i = \nu\}, \quad \mathcal{F}_\nu := [n] \setminus \mathcal{U}_\nu.
\end{aligned}$$

  (Note that the maximization is easily computed by Lemma A.4.)

The proof is presented in Appendix A.6.

Lemma 4.2. *Under the same definitions as Lemma 4.1, The equation $\mathcal{T}(\nu) = S^2$ can be solved by the following procedure: Let $\boldsymbol{e} := [e_1, e_2, \ldots, e_N]$ ($N \le n$, $k \ne k' \Rightarrow e_k \ne e_{k'}$) be a sequence of indices such that*

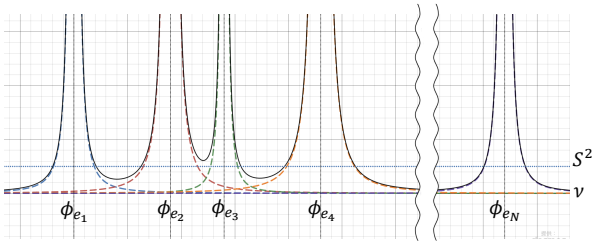

Figure 2: An example of the expression $\mathcal{T}(\nu)$ (black solid line) in Lemmas 4.1 and 4.2. Colored dash lines denote terms in the summation $(\xi_{e_k}/(\nu - \phi_{e_k}))^2$. We can see that, given an interval $(\phi_{e_k}, \phi_{e_{k+1}})$ $(k \in [N-1])$, the function is convex.

1. $e_k \in [n]$ *for any* $k \in [N]$,
2. $i \in [n]$ *is included in* $\boldsymbol{e}$ *if and only if* $\xi_i \neq 0$, *and*
3. $\phi_{e_1} \leq \phi_{e_2} \leq \cdots \leq \phi_{e_N}$.

Note that, if $\phi_{e_k} < \phi_{e_{k+1}}$ $(k \in [N-1])$, *then* $\mathcal{T}(\nu)$ *is a convex function in the interval* $(\phi_{e_k}, \phi_{e_{k+1}})$ *with* $\lim_{\nu \to \phi_{e_k}+0} = \lim_{\nu \to \phi_{e_{k+1}}-0} = +\infty$. *Then, unless* $N = 0$, *each of the following intervals contains just one solution of* $\mathcal{T}(\nu) = S^2$*:*

- *Intervals* $(-\infty, \phi_{e_1})$ *and* $(\phi_{e_N}, +\infty)$.
- *Let* $\nu^{\#(k)} := \mathrm{argmin}_{\phi_{e_k} < \nu < \phi_{e_{k+1}}} \mathcal{T}(\nu)$. *For each* $k \in [N-1]$ *such that* $\phi_{e_k} < \phi_{e_{k+1}}$,
  - *intervals* $(\phi_{e_k}, \nu^{\#(k)})$ *and* $(\nu^{\#(k)}, \phi_{e_{k+1}})$ *if* $\mathcal{T}(\nu^{\#(k)}) < S^2$,
  - *interval* $[\nu^{\#(k)}, \nu^{\#(k)}]$ *(i.e., point) if* $\mathcal{T}(\nu^{\#(k)}) = S^2$.

It follows that $\mathcal{T}(\nu) = S^2$ *has at most* $2n$ *solutions.*

By Lemma 4.2, in order to compute the solution of $\mathcal{T}(\nu) = S^2$, we have only to compute $\nu^{\#(k)}$ by Newton method or the like, and to compute the solution for each interval by Newton method or the like. We show an example of $\mathcal{T}(\nu)$ in Figure 2, and the proof in Appendix A.7.

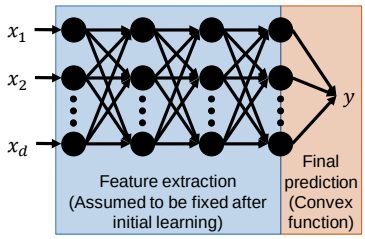

Figure 3: Concept of how to apply SS for deep learning. SS is applied to the last layer for the final prediction.

5 Application to Deep Learning

So far, our discussion of SS rules has primarily focused on ML models with linear predictions and convex loss and regularization functions. However, there may be scenarios where we would like to employ more complex ML models, such as deep learning (DL).

For DL models, deriving SS rules for the entire model can be challenging due to the complexity of bounding the change in model parameters against changes in sample weights. However, we can simplify the process by focusing on the fact that each layer of DL is often represented as a convex function. Therefore, we propose applying SS rules specifically to the last layer of DL models.

In this formulation, the layers preceding the last one are considered as a fixed feature extraction process, even when the sample weights change (see Figure 3). We believe that this approach is valid when the change in sample weights is not significant. We plan to experimentally evaluate the effectiveness of this formulation in Section 6.3.

6 Numerical Experiment

6.1 Experimental Settings

We evaluate the performances of DRSsS and DRSfS across different values of acceptable weight changes S and hyperparameters for regularization strength λ. Performance is measured using *safe screening rates*, representing the ratio of screened samples or features to all samples or features. We consider three setups:

DRSsS with L1-loss L2-regularized SVM (Section 4.1), DRSfS with L2-loss L1-regularized SVM (Section 4.2), and DRSsS with deep learning (Section 5) where the last layer incorporates DRSsS with L1-loss L2-regularized SVM.

In these experiments, we set initialize the sample weights before change ($\tilde{\boldsymbol{w}}$) as $\tilde{\boldsymbol{w}} = \mathbf{1}_n$. Then, we set S in DRSS for $\mathcal{W} := \{\boldsymbol{w} \mid \|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2 \leq S\}$ (Section 4) as follows:

- First we assume the weight change that the weights for positive samples ($\{i \mid y_i = +1\}$) from 1 to a, while retaining the weights for negative samples ($\{i \mid y_i = -1\}$) as 1.
- Then, we defined S as the size of weight change above; specifically, we set $S = \sqrt{n^+}|a - 1|$ (n^+: number of positive samples in the training dataset).

We vary a within the range $0.9 \leq a \leq 1.1$, assuming a maximum change of up to 10% per sample weight.

6.2 Relationship between the Weight Changes and Safe Screening Rate

First, we present safe screening rates for two SVM setups. The datasets used in these experiments are detailed in Table 2. In this experiment, we adapt the regularization hyperparameter λ based on the characteristics of the data. These details are described in Appendix D.1.

As an example, for the "sonar" dataset, we show the DRSsS result in Figure 4 and the DRSfS result in Figure 5. Results for other datasets are presented in Appendix D.2.

These plots allow us to assess the tolerance for changes in sample weights. For instance, with $a = 0.98$ (weight of each positive sample is reduced by two percent, or equivalent weight change in L2-norm), the sample screening rate is 0.31 for L1-loss L2-regularized SVM with $\lambda = 6.58\mathrm{e}+1$, and the feature screening rate is 0.29 for L2-loss L1-regularized SVM with $\lambda = 3.47\mathrm{e}+1$. This implies that, even if the weights are changed in such ranges, a number of samples or features are still identified as redundant in the sense of prediction.

Table 2: Datasets for DRSsS/DRSfS experiments. All are binary classification datasets from LIBSVM dataset [22]. The mark † denotes datasets with one feature removed due to computational constraints. See Appendix D.1 for details.

Task	Name	n	n^+	d
DRSsS	australian	690	307	15
	breast-cancer	683	239	11
	heart	270	120	14
	ionosphere	351	225	35
	sonar	208	97	61
	splice (train)	1000	517	61
	svmguide1 (train)	3089	2000	5
DRSsS	madelon (train)	2000	1000	† 500
	sonar	208	97	† 60
	splice (train)	1000	517	61

6.3 Safe Sample Screening for Deep Learning Model

We applied DRSsS to DL models (Section 5), assuming that all layers are fixed except for the last layer.

We utilized a neural network architecture comprising the following components: firstly, *ResNet50* [23] with an output of 2,048 features, followed by a fully connected layer to reduce the features to 10, and finally, L1-loss L2-regularized SVM (Section 4.1) accompanied by the intercept feature (Remark 2.2).

For the experiment, we employed the CIFAR-10 dataset [24], a well-known benchmark dataset for image classification tasks. We configured the network to classify images into two classes: "airplane" and "automobile". Given that there are 5,000 images for each class, we split the dataset into training:validation:testing=6:2:2, resulting in a total of 6,000 images in the training dataset.

The resulting safe sample screening rates are illustrated in Figure 6. We observed similar outcomes to those obtained with ordinary SVMs in Section 6.2.

This experiment validates the feasibility of applying DRSsS to DL models, demonstrating consistent results with traditional SVM setups.

7 Conclusion

In this paper, we discussed DR-SS, considering the possible changes in sample weights to represent DR setup. We developed a method for calculating SS that can handle changes in sample weights by introducing nontrivial computational techniques, such as constrained maximization of certain convex functions (Section 4.3). Additionally, to address the constraint of SS, which typically applies to ML by minimizing convex functions, we provided an application to DL by applying SS to the last layer of DL model. While this approach is an approximation, it holds certain validity.

For the future work, we aim to explore different environmental changes. In this paper, we focused on weight constraint by L2-norm $\|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2 \leq S$ (Section 4) due to computational considerations. However, when interpreting changes in weights, the constraint of L1-norm $\|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_1 \leq S$ may be more appropriate, as it reflects changes in weights by altering the number of samples. Furthermore, in the context of DR-SS for DL, we are interested in loosening the constraint of fixing the network except for the last layer. Investigating this aspect could provide valuable insights into the flexibility of DR-SS methodologies in DL applications.

Software and Data

The code and the data to reproduce the experiments are available as the attached file.

Potential Broader Impact

This paper contributes to machine learning in dynamically changing environments, a scenario increasingly prevalent in real-world data analyses. We believe that, in such situations, ensuring prediction performance against environmental changes and minimizing storage requirements for expanding datasets will be beneficial. The method does not present significant ethical concerns or foreseeable societal consequences because this work is theoretical and, as of now, has no direct applications that might impact society or ethical considerations.

Acknowledgements

This work was partially supported by MEXT KAKENHI (20H00601), JST CREST (JPMJCR21D3 including AIP challenge program, JPMJCR22N2), JST Moonshot R&D (JPMJMS2033-05), JST AIP Acceleration Research (JPMJCR21U2), NEDO (JPNP18002, JPNP20006) and RIKEN Center for Advanced Intelligence Project.

References

[1] Ruidi Chen and Ioannis Ch. Paschalidis. Distributionally robust learning. arXiv Preprint, 2021.

[2] Laurent El Ghaoui, Vivian Viallon, and Tarek Rabbani. Safe feature elimination for the lasso and sparse supervised learning problems. *Pacific Journal of Optimization*, 8(4):667–698, 2012.

[3] Kohei Ogawa, Yoshiki Suzuki, and Ichiro Takeuchi. Safe screening of non-support vectors in pathwise svm computation. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1382–1390, 2013.

[4] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

[5] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(35):985–1005, 2007.

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

[7] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Sta-*

tistical Society Series B: Statistical Methodology, 58(1):267–288, 1996.

[8] Joel Goh and Melvyn Sim. Distributionally robust optimization and its tractable approximations. *Operations Research*, 58(4-1):902–917, 2010.

[9] Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612, 2010.

[10] Liyuan Wang, Xingxing Zhang, Kuo Yang, Longhui Yu, Chongxuan Li, Lanqing HONG, Shifeng Zhang, Zhenguo Li, Yi Zhong, and Jun Zhu. Memory replay with data compression for continual learning. In *International Conference on Learning Representations*, 2022.

[11] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

[12] Olivier Fercoq, Alexandre Gramfort, and Joseph Salmon. Mind the duality gap: safer rules for the lasso. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 333–342, 2015.

[13] Eugene Ndiaye, Olivier Fercoq, Alexandre Gramfort, and Joseph Salmon. Gap safe screening rules for sparse multi-task and multi-class models. In *Advances in Neural Information Processing Systems*, pages 811–819, 2015.

[14] Shota Okumura, Yoshiki Suzuki, and Ichiro Takeuchi. Quick sensitivity analysis for incremental data modification and its application to leave-one-out cv in linear classification problems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 885–894, 2015.

[15] Atsushi Shibagaki, Masayuki Karasuyama, Kohei Hatano, and Ichiro Takeuchi. Simultaneous safe screening of features and samples in doubly sparse modeling. In *International Conference on Machine Learning*, pages 1577–1586, 2016.

[16] Kazuya Nakagawa, Shinya Suzumura, Masayuki Karasuyama, Koji Tsuda, and Ichiro Takeuchi. Safe pattern pruning: An efficient approach for predictive pattern mining. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1785–1794. ACM, 2016.

[17] Shaogang Ren, Shuai Huang, Jieping Ye, and Xiaoning Qian. Safe feature screening for generalized lasso. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2992–3006, 2018.

[18] Jiang Zhao, Yitian Xu, and Hamido Fujita. An improved non-parallel universum support vector machine and its safe sample screening rule. *Knowledge-Based Systems*, 170:79–88, 2019.

[19] Zhou Zhai, Bin Gu, Xiang Li, and Heng Huang. Safe sample screening for robust support vector machine. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6981–6988, 2020.

[20] Hongmei Wang and Yitian Xu. A safe double screening strategy for elastic net support vector machine. *Information Sciences*, 582:382–397, 2022.

[21] Takumi Yoshida, Hiroyuki Hanada, Kazuya Nakagawa, Kouichi Taji, Koji Tsuda, and Ichiro Takeuchi. Efficient model selection for predictive pattern mining model by safe pattern pruning. *Patterns*, 4(12):100890, 2023.

[22] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011. Datasets are provided in authors' website: `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[24] Alex Krizhevsky. The cifar-10 dataset, 2009.

[25] Ralph Tyrell Rockafellar. *Convex analysis*. Princeton university press, 1970.

[26] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods*. Springer, 1993.

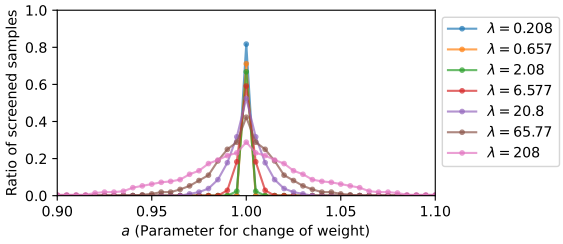

Figure 4: Ratio of screened samples by DRSsS for dataset "sonar".

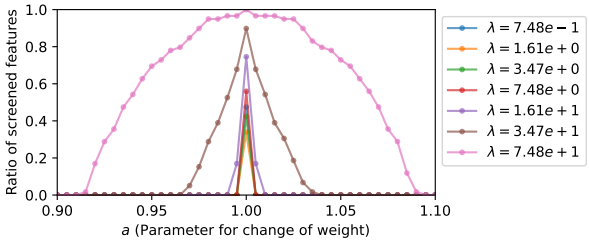

Figure 5: Ratio of screened features by DRSfS for dataset "sonar".

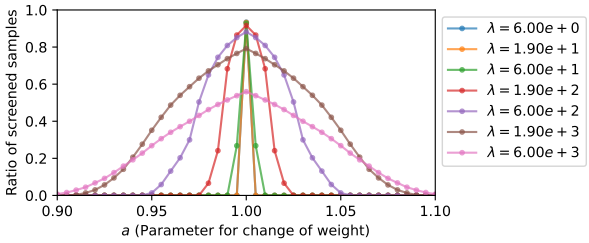

Figure 6: Ratio of screened samples by DRSsS for dataset with CIFAR-10 dataset and DL model ResNet50.

A Proofs

A.1 General Lemmas

Lemma A.1. *For a convex function $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$, f^{**} is equivalent to f if f is convex, proper (i.e., $\exists \boldsymbol{v} \in \mathbb{R}^d :\ f(\boldsymbol{v}) < +\infty$) and lower-semicontinuous.*

Proof. See Section 12 of [25] for example. □

Lemma A.1 is known as *Fenchel-Moreau theorem*. Especially, Lemma A.1 holds if f is convex and $\forall \boldsymbol{v} \in \mathbb{R}^d :\ f(\boldsymbol{v}) < +\infty$.

Lemma A.2. *For a convex function $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$,*

- *f^* is $(1/\nu)$-strongly convex if f is proper and ν-smooth.*
- *f^* is $(1/\kappa)$-smooth if f is proper, lower-semicontinuous and κ-strongly convex.*

Proof. See Section X.4.2 of [26] for example. □

Lemma A.3. *Suppose that $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ is a κ-strongly convex function, and let $\boldsymbol{v}^* = \operatorname{argmin}_{\boldsymbol{v} \in \mathbb{R}^d} f(\boldsymbol{v})$ be the minimizer of f. Then, for any $\boldsymbol{v} \in \mathbb{R}^d$, we have*

$$\|\boldsymbol{v} - \boldsymbol{v}^*\|_2 \leq \sqrt{\frac{2}{\kappa}[f(\boldsymbol{v}) - f(\boldsymbol{v}^*)]}.$$

Proof. See [13] for example. □

Lemma A.4. *For any vector $\boldsymbol{a}, \boldsymbol{c} \in \mathbb{R}^n$ and $S > 0$,*

$$\min_{\boldsymbol{v} \in \mathbb{R}^n :\ \|\boldsymbol{v} - \boldsymbol{c}\|_2 \leq S} \boldsymbol{a}^\top \boldsymbol{v} = \boldsymbol{a}^\top \boldsymbol{c} - S\|\boldsymbol{a}\|_2, \qquad \max_{\boldsymbol{v} \in \mathbb{R}^n :\ \|\boldsymbol{v} - \boldsymbol{c}\|_2 \leq S} \boldsymbol{a}^\top \boldsymbol{v} = \boldsymbol{a}^\top \boldsymbol{c} + S\|\boldsymbol{a}\|_2.$$

Proof. By Cauchy-Schwarz inequality,

$$-\|\boldsymbol{a}\|_2 \|\boldsymbol{v} - \boldsymbol{c}\|_2 \leq \boldsymbol{a}^\top (\boldsymbol{v} - \boldsymbol{c}) \leq \|\boldsymbol{a}\|_2 \|\boldsymbol{v} - \boldsymbol{c}\|_2.$$

Noticing that the first inequality becomes equality if $\exists \omega > 0 :\ \boldsymbol{a} = -\omega(\boldsymbol{v} - \boldsymbol{c})$, while the second inequality becomes equality if $\exists \omega' > 0 :\ \boldsymbol{a} = \omega'(\boldsymbol{v} - \boldsymbol{c})$. Moreover, since $\|\boldsymbol{v} - \boldsymbol{c}\|_2 \leq S$,

$$-S\|\boldsymbol{a}\|_2 \leq \boldsymbol{a}^\top (\boldsymbol{v} - \boldsymbol{c}) \leq S\|\boldsymbol{a}\|_2$$

also holds, with the equality holds if $\|\boldsymbol{v} - \boldsymbol{c}\|_2 = S$.

On the other hand, if we take $\boldsymbol{v}$ that satisfies both of the equality conditions of Cauchy-Schwarz inequality above, that is,

- (for the first inequality being equality) $\boldsymbol{v} = \boldsymbol{c} - (S/\|\boldsymbol{a}\|_2)\boldsymbol{a}$,
- (for the second inequality being equality) $\boldsymbol{v} = \boldsymbol{c} + (S/\|\boldsymbol{a}\|_2)\boldsymbol{a}$,

then the inequalities become equalities. This proves that $-S\|\boldsymbol{a}\|_2$ and $S\|\boldsymbol{a}\|_2$ are surely the minimum and maximum of $\boldsymbol{a}^\top (\boldsymbol{v} - \boldsymbol{c})$, respectively. □

A.2 Derivation of Dual Problem by Fenchel's Duality Theorem

As the formulation of Fenchel's duality theorem, we follow the one in Section 31 of [25].

Lemma A.5 (A special case of Fenchel's duality theorem: $f, g < +\infty$)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$ be convex functions, and $A \in \mathbb{R}^{n \times d}$ be a matrix. Moreover, we define*

$$\boldsymbol{v}^* := \min_{\boldsymbol{v} \in \mathbb{R}^d} [f(A\boldsymbol{v}) + g(\boldsymbol{v})], \tag{18}$$

$$\boldsymbol{u}^* := \max_{\boldsymbol{u} \in \mathbb{R}^n} [-f^*(-\boldsymbol{u}) - g^*(A^\top \boldsymbol{u})]. \tag{19}$$

Then Fenchel's duality theorem assures that

$$\begin{aligned} &f(A\boldsymbol{v}^*) + g(\boldsymbol{v}^*) = -f^*(-\boldsymbol{u}^*) - g^*(A^\top \boldsymbol{u}^*), \\ &-\boldsymbol{u}^* \in \partial f(A\boldsymbol{v}^*), \\ &\boldsymbol{v}^* \in \partial g^*(A^\top \boldsymbol{u}^*). \end{aligned}$$

Sketch of the proof. Introducing a dummy variable $\boldsymbol{\psi} \in \mathbb{R}^n$ and a Lagrange multiplier $\boldsymbol{u} \in \mathbb{R}^n$, we have

$$\min_{\boldsymbol{v} \in \mathbb{R}^d} [f(A\boldsymbol{v}) + g(\boldsymbol{v})] = \max_{\boldsymbol{u} \in \mathbb{R}^n} \min_{\boldsymbol{v} \in \mathbb{R}^d,\ \boldsymbol{\psi} \in \mathbb{R}^n} [f(\boldsymbol{\psi}) + g(\boldsymbol{v}) - \boldsymbol{u}^\top (A\boldsymbol{v} - \boldsymbol{\psi})] \tag{20}$$

$$= -\min_{\boldsymbol{u} \in \mathbb{R}^n} \max_{\boldsymbol{v} \in \mathbb{R}^d,\ \boldsymbol{\psi} \in \mathbb{R}^n} [-f(\boldsymbol{\psi}) - g(\boldsymbol{v}) + \boldsymbol{u}^\top (A\boldsymbol{v} - \boldsymbol{\psi})] = -\min_{\boldsymbol{u} \in \mathbb{R}^n} \max_{\boldsymbol{v} \in \mathbb{R}^d,\ \boldsymbol{\psi} \in \mathbb{R}^n} [\{(-\boldsymbol{u})^\top \boldsymbol{\psi} - f(\boldsymbol{\psi})\} + \{(A^\top \boldsymbol{u})^\top \boldsymbol{v} - g(\boldsymbol{v})\}]$$

$$= -\min_{\boldsymbol{u} \in \mathbb{R}^n} [f^*(-\boldsymbol{u}) + g^*(A^\top \boldsymbol{u})] = \max_{\boldsymbol{u} \in \mathbb{R}^n} [-f^*(-\boldsymbol{u}) - g^*(A^\top \boldsymbol{u})]. \tag{21}$$

Moreover, by the optimality condition of a problem with a Lagrange multiplier (20), the optima of it, denoted by $\boldsymbol{v}^*$, $\boldsymbol{\psi}^*$ and $\boldsymbol{u}^*$, must satisfy

$$A\boldsymbol{v}^* = \boldsymbol{\psi}^*, \quad A^\top \boldsymbol{u}^* \in \partial g(\boldsymbol{v}^*), \quad -\boldsymbol{u}^* \in \partial f(\boldsymbol{\psi}^*) = \partial f(A\boldsymbol{v}^*).$$

On the other hand, introducing a dummy variable $\boldsymbol{\phi} \in \mathbb{R}^d$ and a Lagrange multiplier $\boldsymbol{v} \in \mathbb{R}^d$ for (21), we have

$$\max_{\boldsymbol{u} \in \mathbb{R}^n} [-f^*(-\boldsymbol{u}) - g^*(A^\top \boldsymbol{u})] = \min_{\boldsymbol{v} \in \mathbb{R}^d} \max_{\boldsymbol{u} \in \mathbb{R}^n, \boldsymbol{\phi} \in \mathbb{R}^d} [-f^*(-\boldsymbol{u}) - g^*(\boldsymbol{\phi}) - \boldsymbol{v}^\top (A^\top \boldsymbol{u} - \boldsymbol{\phi})] \tag{22}$$

$$= \min_{\boldsymbol{v} \in \mathbb{R}^d} \max_{\boldsymbol{u} \in \mathbb{R}^n, \boldsymbol{\phi} \in \mathbb{R}^d} [\{(A\boldsymbol{v})^\top (-\boldsymbol{u}) - f^*(-\boldsymbol{u})\} + \{\boldsymbol{v}^\top \boldsymbol{\phi} - g^*(\boldsymbol{\phi})\}]$$

$$= \min_{\boldsymbol{v} \in \mathbb{R}^d} [f^{**}(A\boldsymbol{v}) + g^{**}(\boldsymbol{v})] = \min_{\boldsymbol{v} \in \mathbb{R}^d} [f(A\boldsymbol{v}) + g(\boldsymbol{v})]. \quad (\because \text{ Lemma A.1})$$

Likely above, by the optimality condition of a problem with a Lagrange multiplier (22), the optima of it, denoted by $\boldsymbol{u}^*$, $\boldsymbol{\phi}^*$ and $\boldsymbol{v}^*$, must satisfy

$$A^\top \boldsymbol{u}^* = \boldsymbol{\phi}^*, \quad \boldsymbol{v}^* \in \partial g^*(\boldsymbol{\phi}^*) = \partial g^*(A^\top \boldsymbol{u}^*), \quad A\boldsymbol{v}^* \in \partial f(-\boldsymbol{u}^*).$$

□

Lemma A.6 (Dual problem of weighted regularized empirical risk minimization (weighted RERM))**.** *For the minimization problem*

$$\boldsymbol{\beta}^{*(\boldsymbol{w})} := \operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^d} P_{\boldsymbol{w}}(\boldsymbol{\beta}), \quad \textit{where} \quad P_{\boldsymbol{w}}(\boldsymbol{\beta}) := \sum_{i=1}^n w_i \ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}) + \rho(\boldsymbol{\beta}), \tag{(1) restated}$$

we define the dual problem as the one obtained by applying Fenchel's duality theorem (Lemma A.5), which is defined as

$$\boldsymbol{\alpha}^{*(\boldsymbol{w})} := \operatorname*{argmax}_{\boldsymbol{\alpha}\in\mathbb{R}^n} D_{\boldsymbol{w}}(\boldsymbol{\alpha}), \quad \textit{where} \quad D_{\boldsymbol{w}}(\boldsymbol{\alpha}) := -\sum_{i=1}^{n} w_i \ell^*_{y_i}(-\gamma_i\alpha_i) + \rho^*(((\boldsymbol{\gamma}\otimes\boldsymbol{w})\boxtimes\check{X})^\top\boldsymbol{\alpha}). \quad \text{((2) restated)}$$

Moreover, $\boldsymbol{\beta}^{*(\boldsymbol{w})}$ and $\boldsymbol{\alpha}^{*(\boldsymbol{w})}$ must satisfy

$$P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\boldsymbol{w})}) = D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\boldsymbol{w})}), \quad \text{((3) restated)}$$

$$\boldsymbol{\beta}^{*(\boldsymbol{w})} \in \partial\rho^*(((\boldsymbol{\gamma}\otimes\boldsymbol{w})\boxtimes\check{X})^\top\boldsymbol{\alpha}^{*(\boldsymbol{w})}), \quad \text{((4) restated)}$$

$$\forall i \in [n]: \quad -\gamma_i\alpha_i^{*(\boldsymbol{w})} \in \partial\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}). \quad \text{((5) restated)}$$

Proof. To apply Fenchel's duality theorem, we have only to set f, g and A in Lemma A.5 as

$$f(\boldsymbol{u}) := \sum_{i=1}^{n} w_i\ell_{y_i}(u_i), \quad g(\boldsymbol{\beta}) := \rho(\boldsymbol{\beta}), \quad A := \check{X}.$$

Here, noticing that

$$f^*(\boldsymbol{u}) = \sup_{\boldsymbol{u}'\in\mathbb{R}^n}[\boldsymbol{u}^\top\boldsymbol{u}' - \sum_{i=1}^{n} w_i\ell_{y_i}(u_i')] = \sup_{\boldsymbol{u}'\in\mathbb{R}^n}\sum_{i=1}^{n}[u_iu_i' - w_i\ell_{y_i}(u_i')]$$
$$= \sup_{\boldsymbol{u}'\in\mathbb{R}^n}\sum_{i=1}^{n} w_i\left[\frac{u_i}{w_i}u_i' - \ell_{y_i}(u_i')\right] = \sum_{i=1}^{n} w_i\ell^*_{y_i}\left(\frac{u_i}{w_i}\right),$$

from (19) we have

$$-f^*(-\boldsymbol{u}) - g^*(A^\top\boldsymbol{u}) = -\sum_{i=1}^{n} w_i\ell^*_{y_i}\left(-\frac{u_i}{w_i}\right) - \rho^*(\check{X}^\top\boldsymbol{u}).$$

Replacing $u_i \leftarrow \gamma_i w_i\alpha_i$, that is, $\boldsymbol{u} \leftarrow (\boldsymbol{\gamma}\otimes\boldsymbol{w}\otimes\boldsymbol{\alpha})$, we have the dual problem (2).

The relationships between the primal and the dual problem are described as follows:

$$-\boldsymbol{u}^* \in \partial f(A\boldsymbol{v}^*) \;\Rightarrow\; -\boldsymbol{\gamma}\otimes\boldsymbol{w}\otimes\boldsymbol{\alpha}^{*(\boldsymbol{w})} \in \partial f(\check{X}\boldsymbol{\beta}^{*(\boldsymbol{w})}) \;\Rightarrow\; -\gamma_i w_i\alpha_i^{*(\boldsymbol{w})} \in w_i\partial\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})})$$
$$\Rightarrow -\gamma_i\alpha_i^{*(\boldsymbol{w})} \in \partial\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}),$$
$$\boldsymbol{v}^* \in \partial g^*(A^\top\boldsymbol{u}^*) \;\Rightarrow\; \boldsymbol{\beta}^{*(\boldsymbol{w})} \in \partial g^*(\check{X}^\top\boldsymbol{\gamma}\otimes\boldsymbol{w}\otimes\boldsymbol{\alpha}^{*(\boldsymbol{w})}) = \partial g^*(((\boldsymbol{\gamma}\otimes\boldsymbol{w})\boxtimes\check{X})^\top\boldsymbol{\alpha}^{*(\boldsymbol{w})}).$$

□

A.3 Proof of Lemma 3.1

Proof. [13]

$$\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{*(\boldsymbol{w})}\|_2 \le \sqrt{\frac{2}{\lambda}[P_{\boldsymbol{w}}(\hat{\boldsymbol{\beta}}) - P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\boldsymbol{w})})]} \qquad (\because \text{setting } f \leftarrow P_{\boldsymbol{w}} \text{ in Lemma A.3})$$
$$= \sqrt{\frac{2}{\lambda}[P_{\boldsymbol{w}}(\hat{\boldsymbol{\beta}}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\boldsymbol{w})})]} \qquad (\because (3))$$
$$\le \sqrt{\frac{2}{\lambda}[P_{\boldsymbol{w}}(\hat{\boldsymbol{\beta}}) - D_{\boldsymbol{w}}(\hat{\boldsymbol{\alpha}})]}. \qquad (\because \boldsymbol{\alpha}^{*(\boldsymbol{w})} \text{ is a maximizer of } D_{\boldsymbol{w}})$$

□

A.4 Proof of Lemma 3.2

Proof. Due to (5), if $\partial\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}) = \{0\}$ is assured, then $\alpha_i^{*(\boldsymbol{w})} = 0$ is assured. Since we do not know $\boldsymbol{\beta}^{*(\boldsymbol{w})}$ but know $\mathcal{B}^{*(\boldsymbol{w})}$ (Lemma 3.1), we can assure $\alpha_i^{*(\boldsymbol{w})} = 0$ if $\bigcup_{\boldsymbol{\beta}\in\mathcal{B}^{*(\boldsymbol{w})}} \partial\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}) = \{0\}$ is assured. Noticing that $\partial\ell_{y_i}$ is monotonically increasing[2], we have

$$\bigcup_{\boldsymbol{\beta}\in\mathcal{B}^{*(\boldsymbol{w})}} \partial\ell_{y_i}(\check{X}_{i:}\boldsymbol{\beta}) = \{0\} \quad\Leftrightarrow\quad \bigcup_{\boldsymbol{\beta}\in\mathcal{B}^{*(\boldsymbol{w})}} \check{X}_{i:}\boldsymbol{\beta} \subseteq \mathcal{Z}[\ell_{y_i}] \quad\Leftrightarrow\quad [\min_{\boldsymbol{\beta}\in\mathcal{B}^{*(\boldsymbol{w})}} \check{X}_{i:}\boldsymbol{\beta}, \max_{\boldsymbol{\beta}\in\mathcal{B}^{*(\boldsymbol{w})}} \check{X}_{i:}\boldsymbol{\beta}] \subseteq \mathcal{Z}[\ell_{y_i}]$$

$$\Leftrightarrow \quad \left[\check{X}_{i:}\hat{\boldsymbol{\beta}} - \|\check{X}_{i:}\|_2 r(\boldsymbol{w},\boldsymbol{\gamma},\kappa,\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\alpha}}),\ \check{X}_{i:}\hat{\boldsymbol{\beta}} + \|\check{X}_{i:}\|_2 r(\boldsymbol{w},\boldsymbol{\gamma},\kappa,\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\alpha}})\right] \subseteq \mathcal{Z}[\ell_{y_i}]. \qquad (\because \text{Lemma A.4})$$

□

A.5 Proof of Lemma 3.3

Proof. The proof is almost the same as that for Lemma 3.1 (see Appendix A.3), but we additionally need to show that $-D_{\boldsymbol{w}}$ is $((\min_{i\in[n]} w_i\gamma_i^2)/\mu)$-strongly convex (in this case $D_{\boldsymbol{w}}$ is called *strongly concave*).

As discussed in Lemma A.2, $-\ell_{y_i}^*(t)$ is $(1/\mu)$-strongly convex, that is, $-\ell_{y_i}^*(t) - (1/2\mu)t^2$ is convex. Thus,

- $-\ell_{y_i}^*(-\gamma_i\alpha_i) - (1/2\mu)(\gamma_i\alpha_i)^2$ is convex with respect to α_i,
- $-w_i\ell_{y_i}^*(-\gamma_i\alpha_i) - (w_i\gamma_i^2/2\mu)\alpha_i^2$ is convex with respect to α_i,
- $-\sum_{i=1}^n w_i\ell_{y_i}^*(-\gamma_i\alpha_i) - \sum_{i=1}^n (w_i\gamma_i^2/2\mu)\alpha_i^2$ is convex with respect to $\boldsymbol{\alpha}$.

So, $-\sum_{i=1}^n w_i\ell_{y_i}^*(-\gamma_i\alpha_i)$ is convex with respect to $\boldsymbol{\alpha}$ even subtracted by $\sum_{k=1}^n [\min_{i\in[n]}(w_i\gamma_i^2/2\mu)]\alpha_k^2 = (1/2)[\min_{i\in[n]}(w_i\gamma_i^2/\mu)]\|\boldsymbol{\alpha}\|_2^2$. □

A.6 Proof of Lemma 4.1

Lemma A.7. *For the optimization problem*

$$\begin{aligned}
&\max_{\boldsymbol{w}\in\mathcal{W}} \boldsymbol{w}^\top A\boldsymbol{w} + 2\boldsymbol{b}^\top\boldsymbol{w}, && \text{((16) restated)}\\
&\textit{subject to} \quad \mathcal{W} := \{\boldsymbol{w}\in\mathbb{R}^n \mid \|\boldsymbol{w}-\tilde{\boldsymbol{w}}\|_2 \le S\},\\
&\textit{where} \quad \tilde{\boldsymbol{w}}\in\mathbb{R}^n, \quad \boldsymbol{b}\in\mathbb{R}^n,\\
&\qquad\qquad A\in\mathbb{R}^{n\times n}:\ \textit{symmetric, positive semidefinite, nonzero,}
\end{aligned}$$

its stationary points are obtained as the solution of the following equations with respect to $\boldsymbol{w}$ and $\nu\in\mathbb{R}$:

$$A\boldsymbol{w} + \boldsymbol{b} - \nu(\boldsymbol{w}-\tilde{\boldsymbol{w}}) = \boldsymbol{0}, \qquad (23)$$

$$\|\boldsymbol{w}-\tilde{\boldsymbol{w}}\|_2 = S. \qquad (24)$$

Also, when both (23) *and* (24) *are satisfied, the function to be maximized is calculated as*

$$\boldsymbol{w}^\top A\boldsymbol{w} + 2\boldsymbol{b}^\top\boldsymbol{w} = \nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top(\boldsymbol{w}-\tilde{\boldsymbol{w}}) + \tilde{\boldsymbol{w}}^\top\boldsymbol{b}. \qquad (25)$$

[2]Since $\partial\ell_{y_i}$ is a multi-valued function, the monotonicity must be defined accordingly: we call a multi-valued function $F:\mathbb{R}\to 2^{\mathbb{R}}$ is monotonically increasing if, for any $t<t'$, F must satisfy "$\forall s\in F(t)$, $\forall s'\in F(t')$: $s\le s'$".

Proof. First, $\boldsymbol{w}^\top A\boldsymbol{w} + 2\boldsymbol{b}^\top \boldsymbol{w}$ is convex and not constant. Then we can show that (16) is optimized in $\{\boldsymbol{w} \in \mathbb{R}^n \mid \|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2 = S\}$, that is, at the surface of the hyperball $\mathcal{W}$ (Theorem 32.1 of [25]). This proves (24). Moreover, with the fact, we write the Lagrangian function with Lagrange multiplier $\nu \in \mathbb{R}$ as:

$$L(\boldsymbol{w}, \nu) := \boldsymbol{w}^\top A\boldsymbol{w} + 2\boldsymbol{b}^\top \boldsymbol{w} - \nu(\|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2^2 - S^2).$$

Then, due to the property of Lagrange multiplier, the stationary points of (16) are obtained as

$$\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{w}} &= 2A\boldsymbol{w} + 2\boldsymbol{b} - 2\nu(\boldsymbol{w} - \tilde{\boldsymbol{w}}) = 0, \\
\frac{\partial L}{\partial \nu} &= \|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2^2 - S^2 = 0,
\end{aligned}$$

where the former derives (23).

Finally we show (25). If both (23) and (24) are satisfied,

$$\begin{aligned}
\boldsymbol{w}^\top A\boldsymbol{w} + 2\boldsymbol{b}^\top \boldsymbol{w} &= \boldsymbol{w}^\top(\nu(\boldsymbol{w} - \tilde{\boldsymbol{w}}) - \boldsymbol{b}) + 2\boldsymbol{b}^\top \boldsymbol{w} && (\because (23)) \\
&= \nu \boldsymbol{w}^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top \boldsymbol{w} \\
&= \nu(\boldsymbol{w} - \tilde{\boldsymbol{w}})^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \nu\tilde{\boldsymbol{w}}^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top \tilde{\boldsymbol{w}} \\
&= \nu S^2 + \nu\tilde{\boldsymbol{w}}^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top \tilde{\boldsymbol{w}} && (\because (24)) \\
&= \nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top(\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top \tilde{\boldsymbol{w}} && ((25) \text{ restated})
\end{aligned}$$

□

Proof of Lemma 4.1. The condition (23) is calculated as

$$\begin{aligned}
A\boldsymbol{w} + \boldsymbol{b} &= \nu(\boldsymbol{w} - \tilde{\boldsymbol{w}}), \\
(A - \nu I)(\boldsymbol{w} - \tilde{\boldsymbol{w}}) &= -A\tilde{\boldsymbol{w}} - \boldsymbol{b}.
\end{aligned}$$

Here, let us apply eigendecomposition of A, denoted by $A = Q^\top \Phi Q$, where $Q \in \mathbb{R}^{n\times n}$ is orthogonal ($QQ^\top = Q^\top Q = I$) and $\Phi := \mathrm{diag}(\phi_1, \phi_2, \ldots, \phi_n)$ is a diagonal matrix consisting of eigenvalues of A. Such a decomposition is assured to exist since A is assumed to be symmetric and positive semidefinite. Then,

$$\begin{aligned}
&(Q^\top \Phi Q - \nu I)(\boldsymbol{w} - \tilde{\boldsymbol{w}}) = -Q^\top \Phi Q\tilde{\boldsymbol{w}} - \boldsymbol{b}, \\
&Q^\top(\Phi - \nu I)Q(\boldsymbol{w} - \tilde{\boldsymbol{w}}) = -Q^\top \Phi Q\tilde{\boldsymbol{w}} - \boldsymbol{b}, \\
&(\Phi - \nu I)\boldsymbol{\tau} = \boldsymbol{\xi}, \quad (\text{where} \quad \boldsymbol{\tau} := Q(\boldsymbol{w} - \tilde{\boldsymbol{w}}), \quad \boldsymbol{\xi} := -\Phi Q\tilde{\boldsymbol{w}} - Q\boldsymbol{b} \in \mathbb{R}^n,) && (26) \\
&\forall i \in [n]: \quad (\phi_i - \nu)\tau_i = \xi_i. && (27)
\end{aligned}$$

Note that we have to be also aware of the constraint

$$S = \|\boldsymbol{\tau}\|_2 = \sqrt{\boldsymbol{\tau}^\top \boldsymbol{\tau}} = \sqrt{(\boldsymbol{w} - \tilde{\boldsymbol{w}})^\top Q^\top Q(\boldsymbol{w} - \tilde{\boldsymbol{w}})} = \|\boldsymbol{w} - \tilde{\boldsymbol{w}}\|_2. \tag{28}$$

Here, we consider these two cases.

1. First, consider the case when $(\Phi - \nu I)$ is nonsingular, that is, when ν is different from any of $\phi_1, \phi_2, \ldots, \phi_n$. Then, from (28) we have

$$S^2 = \|\boldsymbol{\tau}\|_2 = \sum_{i=1}^{n} \tau_i^2 = \sum_{i=1}^{n} \left(\frac{\xi_i}{\nu - \phi_i} \right)^2 \quad \big(=: \mathcal{T}(\nu)\big). \tag{29}$$

   So, values of (16) for all stationary points with respect to $\boldsymbol{w}$ and ν (on condition that $(\Phi - \nu I)$ is nonsingular) can be obtained by computing (25) for each ν satisfying (29), that is,

   - for such ν computing $\boldsymbol{\tau}$ by (27), and
   - computing (25) as $\nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top (\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top \tilde{\boldsymbol{w}} = \nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top Q^\top \boldsymbol{\tau} + \boldsymbol{b}^\top \tilde{\boldsymbol{w}}$.

2. Secondly, consider the case when $(\Phi - \nu I)$ is nonsingular, that is, when ν is equal to one of $\phi_1, \phi_2, \ldots, \phi_n$. First, given ν, let $\mathcal{U}_\nu := \{i \mid i \in [n],\ \phi_i = \nu\}$ be the indices of $\{\phi_i\}_i$ equal to ν (this may include more than one indices), and $\mathcal{F}_\nu := [n] \setminus \mathcal{U}_\nu$. Note that, by assumption, $\mathcal{U}_\nu$ is not empty. Then, all stationary points of (16) with respect to $\boldsymbol{w}$ and ν (on condition that $(\Phi - \nu I)$ is singular) can be found by computing the followings for each $\nu \in \{\phi_1, \phi_2, \ldots, \phi_n\}$ (duplication excluded):

   - If $\xi_i \neq 0$ for at least one $i \in \mathcal{U}_\nu$, the equation (27) cannot hold.
   - If $\xi_i = 0$ for all $i \in \mathcal{N}_\nu$, the equation (27) may hold. So we calculate $\boldsymbol{\tau}$ that maximizes (16) as follows:
     - Fix $\tau_i = \xi_i / (\phi_i - \nu)$ for $i \in \mathcal{F}_\nu$.
     - Set the constraint $\sum_{i \in \mathcal{U}_\nu} \tau_i^2 = S^2 - \sum_{i \in \mathcal{F}_\nu} \tau_i^2$ (due to (28)).
     - Maximize (16) with respect to $\{\tau_i\}_{i \in \mathcal{U}_\nu}$ under the constraints above. Here, by (25) we have only to calculate

$$\begin{aligned} &\max_{\boldsymbol{\tau} \in \mathbb{R}^n} [\nu S^2 + (\nu\tilde{\boldsymbol{w}} + \boldsymbol{b})^\top (\boldsymbol{w} - \tilde{\boldsymbol{w}}) + \boldsymbol{b}^\top \tilde{\boldsymbol{w}}], \\ &\text{subject to} \quad \forall i \in \mathcal{F}_\nu : \quad \tau_i = \frac{\xi_i}{\phi_i - \nu}, \\ &\qquad\qquad \sum_{i \in \mathcal{U}_\nu} \tau_i^2 = S^2 - \sum_{i \in \mathcal{F}_\nu} \tau_i^2, \end{aligned} \tag{30}$$

       which is easily computed by Lemma A.4. The value of the maximization result is equal to that of (16) on condition that ν is specified above.

   So, collecting these result and taking the largest one, the maximization (on condition that $(\Phi - \nu I)$ is singular) is completed.

Taking the maximum of the two cases, we have the maximization result of (16). □

A.7 Proof of Lemma 4.2

Proof. We show the statements in the lemma that, if $\phi_{e_k} < \phi_{e_{k+1}}$ $(k \in [N-1])$, then $\mathcal{T}(\nu)$ is a convex function in the interval $(\phi_{e_k}, \phi_{e_{k+1}})$ with $\lim_{\nu \to \phi_{e_k} + 0} = \lim_{\nu \to \phi_{e_{k+1}} - 0} = +\infty$. Then the conclusion immediately follows.

The latter statement clearly holds. The former statement is proved by directly computing the derivative.

$$\frac{d}{d\nu}\mathcal{T}(\nu) = \frac{d}{d\nu}\sum_{i=1}^{n}\left(\frac{\xi_i}{\nu-\phi_i}\right)^2 = -2\sum_{i=1}^{n}\frac{\xi_i^2}{(\nu-\phi_i)^3}.$$

It is an increasing function with respect to ν, as long as ν does not match any of $\{\phi_i\}_{i=1}^n$ such that $\xi_i \neq 0$. So it is convex in the interval $\phi_{e_k} < \nu < \phi_{e_{k+1}}$. □

B Detailed Calculations

In this appendix we describe detailed calculations omitted in the main paper.

B.1 Calculations for L1-loss L2-regularized SVM (Section 4.1)

For this setup, we can calculate as

$$\rho^*(\boldsymbol{\beta}) := \frac{1}{2\lambda}\|\boldsymbol{\beta}\|_2^2, \quad \ell_y^*(t) := \begin{cases} t, & (-1 \le t \le 0) \\ +\infty, & (\text{otherwise}) \end{cases} \quad \partial\rho^*(\boldsymbol{\beta}) := \left\{\frac{1}{\lambda}\boldsymbol{\beta}\right\}, \quad \partial\ell_y(t) := \begin{cases} \{-1\}, & (t<1) \\ [-1,0], & (t=1) \\ \{0\}. & (t>1) \end{cases}$$

Then we have the dual problem in the main paper (6).

B.2 Calculations for L2-loss L1-regularized SVM (Section 4.2)

For this setup, we can calculate as

$$\rho^*(\boldsymbol{\beta}) := \begin{cases} 0, & (\beta_d = 0,\ \forall j \in [d-1]:\ |\beta_j| \le \lambda) \\ +\infty, & (\text{otherwise}) \end{cases} \quad \ell_y^*(t) := \begin{cases} \frac{t^2+4t}{4}, & (t \le 0) \\ +\infty, & (\text{otherwise}) \end{cases}$$

$$\forall j \in [d-1]:\ [\partial\rho^*(\boldsymbol{\beta})]_j := \begin{cases} -\infty, & (\beta_j < -\lambda) \\ [-\infty, 0], & (\beta_j = -\lambda) \\ 0, & (|\beta_j| < \lambda) \\ [0, +\infty], & (\beta_j = \lambda) \\ +\infty, & (\beta_j > \lambda) \end{cases} \quad [\partial\rho^*(\boldsymbol{\beta})]_d := \begin{cases} -\infty, & (\beta_d < 0) \\ [-\infty, +\infty], & (\beta_d = 0) \\ +\infty, & (\beta_d > 0) \end{cases}$$

$$\partial\ell_y(t) := -2\max\{0, 1-t\}.$$

Then, setting $\gamma_i = \lambda$ for all $i \in [n]$, the dual objective function is described as

$$D_{\boldsymbol{w}}(\boldsymbol{\alpha}) = \begin{cases} -\sum_{i=1}^n w_i \frac{\lambda^2\alpha_i^2 - 4\lambda\alpha_i}{4}, & (\text{if (32) are satisfied}) \\ +\infty, & (\text{otherwise}) \end{cases} \tag{31}$$

where

$$\lambda\alpha_i \ge 0 \Leftrightarrow \alpha_i \ge 0, \tag{32a}$$

$$\forall j \in [d-1]:\ |((\lambda\mathbf{1}_n \otimes \boldsymbol{w}) \otimes \check{X}_{:j})^\top\boldsymbol{\alpha}| \le \lambda \Leftrightarrow |(\boldsymbol{w} \otimes \check{X}_{:j})^\top\boldsymbol{\alpha}| \le 1, \tag{32b}$$

$$((\lambda\mathbf{1}_n \otimes \boldsymbol{w}) \otimes \check{X}_{:d})^\top\boldsymbol{\alpha} = 0 \Leftrightarrow (\boldsymbol{w} \otimes \check{X}_{:d})^\top\boldsymbol{\alpha} = 0. \tag{32c}$$

Optimality conditions (4) and (5) are described as

$$\forall j \in [d-1]: \ |(\lambda \mathbf{1}_n \otimes \boldsymbol{w} \otimes \check{X}_{:j})^\top \boldsymbol{\alpha}^{*(\boldsymbol{w})}| < \lambda \Leftrightarrow |(\boldsymbol{w} \otimes \check{X}_{:j})^\top \boldsymbol{\alpha}^{*(\boldsymbol{w})}| < 1 \Rightarrow \beta_j^{*(\boldsymbol{w})} = 0, \tag{33}$$

$$\forall i \in [n]: \quad \lambda \alpha_i^{*(\boldsymbol{w})} = 2\max\{0, 1 - \check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}\}. \tag{34}$$

C Application of Safe Sample Screening to Kernelized Features

The kernel method in ML means computation methods when the input variable vector of a sample $\boldsymbol{x} \in \mathbb{R}^d$ cannot be specifically obtained (this includes the case when d is infinite), but for the input variable vectors for any two samples $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^d$ its inner product $\boldsymbol{x}^\top \boldsymbol{x}'$ can be obtained. In such a case, we cannot discuss SfS since we cannot obtain each feature specifically, however, we can discuss SsS.

We show that the SsS rules for L1-loss L2-regularized SVM (Section 4.1) can be applied even if the features are kernelized.

First, if features are kernelized, we cannot obtain either X or $\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$ specifically. However, since we can obtain $\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})}$, with (7) we have

$$\forall \boldsymbol{x} \in \mathbb{R}^d: \ \boldsymbol{x}^\top \boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})} = \frac{1}{\lambda} \boldsymbol{x}^\top (\boldsymbol{w} \boxtimes \check{X})^\top \boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})} = \frac{1}{\lambda} \sum_{i=1}^n w_i \alpha_i^{*(\tilde{\boldsymbol{w}})} (\boldsymbol{x}^\top \check{X}_{i:}). \tag{35}$$

This means that we can calculate the inner product of $\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$ and any vector.

Then, in order to calculate the quantity (9) to conduct SsS, we have only to calculate

- $\check{X}_{i:}\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$ can be calculated by (35),
- $\|\check{X}_{i:}\|_2 = \sqrt{\check{X}_{i:}^\top \check{X}_{i:}}$ is obtained as the kernel value, and
- $P_{\boldsymbol{w}}(\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}) - D_{\boldsymbol{w}}(\boldsymbol{\alpha}^{*(\tilde{\boldsymbol{w}})})$ can be calculated by (35) and kernel values since two variables whose values cannot be specifically obtained ($\tilde{X}$ and $\boldsymbol{\beta}^{*(\tilde{\boldsymbol{w}})}$) appears only as inner products.

So, all values needed to derive SsS rules (9) can be computed even if features are kernelized.

D Details of Experiments

D.1 Detailed Experimental Setup

The criteria of selecting datasets (Table 2) and detailed setups are as follows:

- All of the datasets are downloaded from LIBSVM dataset [22]. We used scaled datasets for ones used in DRSfS or only scaled datasets are provided ("ionosphere", "sonar" and "splice"). We used training datasets only if test datasets are provided separately ("splice", "svmguide1" and "madelon").
- For DRSsS, we selected datasets from LIBSVM dataset containing 100 to 10,000 samples, 100 or fewer features, and the area under the curve (AUC) of the receiver operating characteristic (ROC) is 0.9 or higher for the regularization strengths (λ) we examined so that they tend to facilitate more effective sample screening.

- For DRSfS, we selected datasets from LIBSVM dataset containing 50 to 1,000 features, 10,000 or fewer samples, and containing no categorical features. Also, due to computational constraints, we excluded features that have at least one zero (marked "†" in Table 2). As a result, one feature from "madelon" and one from "sonar" have been excluded.
- In the table, the column "d" denotes the number of features including the intercept feature (Remark 2.2).

The choice of regularization hyperparameter λ, based on the characteristics of the data, is as follows:

- For DRSsS, we set λ as n, $n \times 10^{-0.5}$, $n \times 10^{-1.0}$, ..., $n \times 10^{-3.0}$. (For DRSsS with DL, we set 1000 instead of n.) This is because the effect of λ gets weaker for larger n.
- For DRSfS, we determine λ based on $\lambda_{\max}$, defined as the smallest λ for which $\beta_j^{*(\boldsymbol{w})} = 0$ for any $j \in [d-1]$ explained below. We then set λ as $\lambda_{\max}$, $\lambda_{\max} \times 10^{-1/3}$, $\lambda_{\max} \times 10^{-2/3}$, ..., $\lambda_{\max} \times 10^{-2}$.

Finally, we show the calculation of $\lambda_{\max}$ for L2-loss L1-regularized SVM. By (14), we would like to find λ so that $|(\boldsymbol{w} \otimes \check{X}_{:j})^\top \boldsymbol{\alpha}^{*(\boldsymbol{w})}| < 1$ for all $j \in [d-1]$. In order to judge this, we need $\boldsymbol{\alpha}^{*(\boldsymbol{w})}$, which is calculated as follows:

- Solve the primal problem (1) for L2-loss L1-regularized SVM by fixing $\beta_j^{*(\boldsymbol{w})} = 0$ for any $j \in [d-1]$, that is,

$$\begin{aligned}
\beta_d^{*(\boldsymbol{w})} &= \operatorname*{argmin}_{\beta_d} \sum_{i=1}^n w_i \ell_{y_i}(\check{x}_{id}\beta_d) = \operatorname*{argmin}_{\beta_d} \sum_{i=1}^n w_i (\max\{0, 1 - y_i\beta_d\})^2 \\
&= \operatorname*{argmin}_{\beta_d} \sum_{i\in[n],\ y_i=+1} w_i (\max\{0, 1 - \beta_d\})^2 + \sum_{i\in[n],\ y_i=-1} w_i (\max\{0, 1 + \beta_d\})^2 \\
&= \frac{\sum_{i\in[n],\ y_i=+1} w_i - \sum_{i\in[n],\ y_i=-1} w_i}{\sum_{i=1}^n w_i}.
\end{aligned}$$

- With $\beta_d^{*(\boldsymbol{w})}$ computed above and $\beta_j^{*(\boldsymbol{w})} = 0$ for any $j \in [d-1]$, calculate $\boldsymbol{\alpha}^{\$} = \lambda\boldsymbol{\alpha}^{*(\boldsymbol{w})} = [2\max\{0, 1 - \check{X}_{i:}\boldsymbol{\beta}^{*(\boldsymbol{w})}\}]_{i=1}^n$ by (15).
- If $|(\boldsymbol{w} \otimes \check{X}_{:j})^\top \boldsymbol{\alpha}^{\$})| < \lambda$ for all $j \in [d-1]$, then $\beta_j^{*(\boldsymbol{w})} = 0$ for any $j \in [d-1]$. So, we set $\lambda_{\max} = \max_{j\in[d-1]} |(\boldsymbol{w} \otimes \check{X}_{:j})^\top \boldsymbol{\alpha}^{\$})|$.

D.2 All Experimental Results of Section 6.2

For the experiment of Section 6.2, ratios of screened samples by DRSsS setup is presented in Figure 7, while ratios of screened features by DRSfS setup in Figure 8.

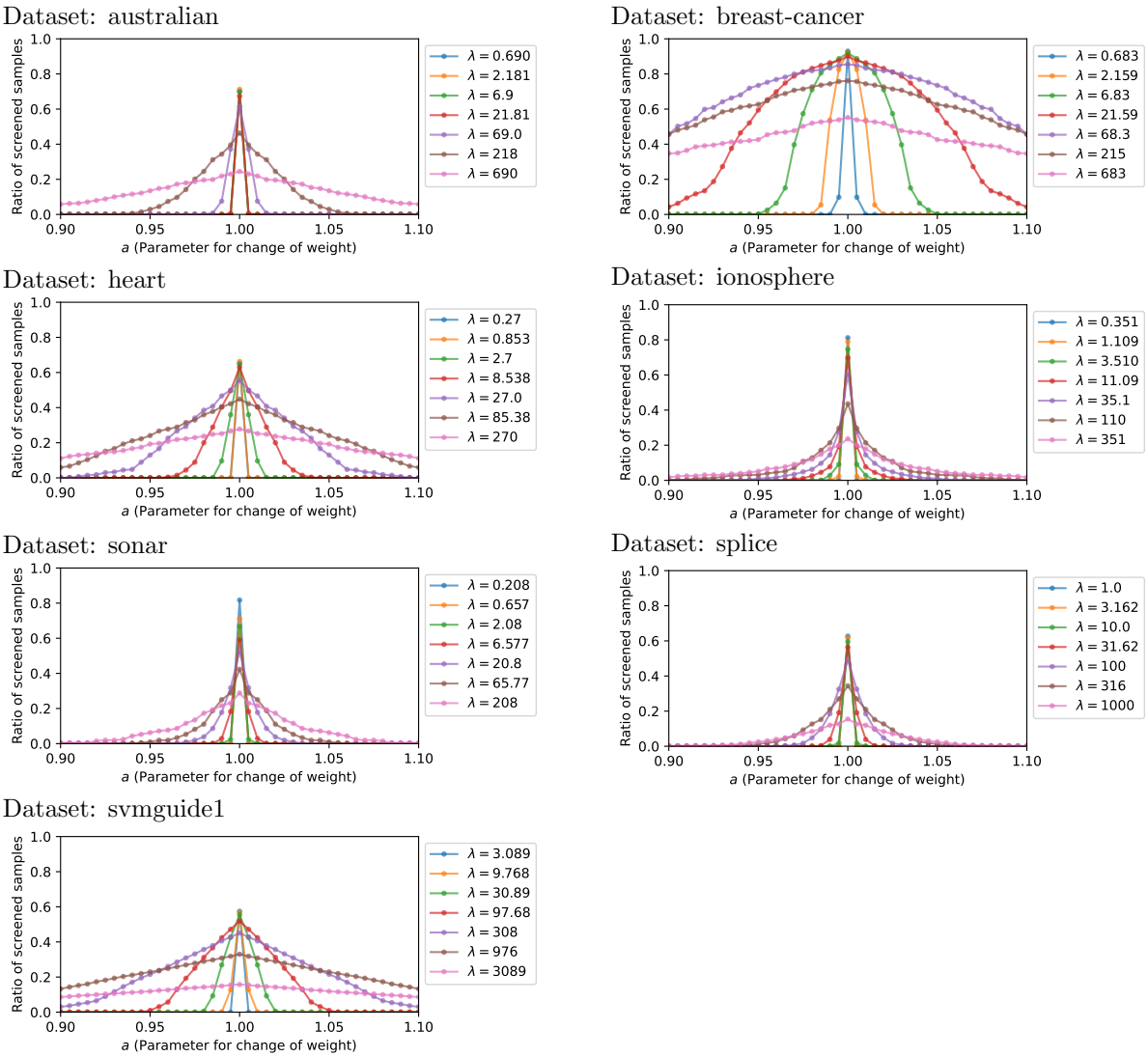

Figure 7: Ratios of screened samples by DRSsS.

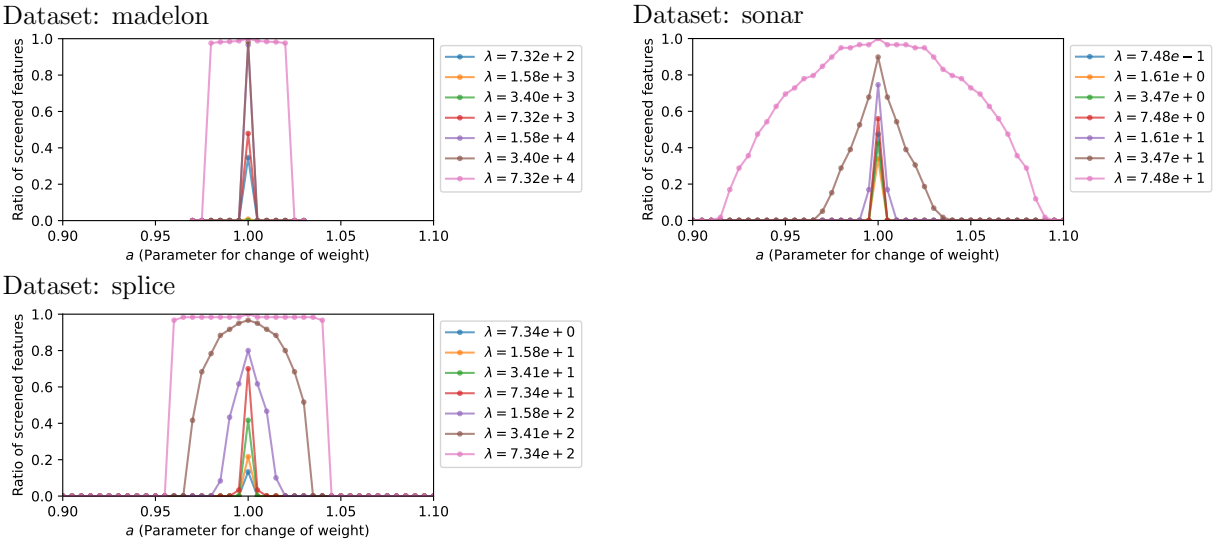

Figure 8: Ratios of screened features by DRSfS.