Scaling Up Sparse Support Vector Machines by Simultaneous Feature and Sample Reduction

Weizhong Zhang[*1 2] Bin Hong[*1 3] Wei Liu[2] Jieping Ye[3] Deng Cai[1] Xiaofei He[1] Jie Wang[3]

Abstract

Sparse support vector machine (SVM) is a popular classification technique that can simultaneously learn a small set of the most interpretable features and identify the support vectors. It has achieved great successes in many real-world applications. However, for large-scale problems involving a huge number of samples and extremely high-dimensional features, solving sparse SVMs remains challenging. By noting that sparse SVMs induce sparsities in both feature and sample spaces, we propose a novel approach, which is based on accurate estimations of the primal and dual optima of sparse SVMs, to simultaneously identify the features and samples that are guaranteed to be irrelevant to the outputs. Thus, we can remove the identified inactive samples and features from the training phase, leading to substantial savings in both the memory usage and computational cost without sacrificing accuracy. To the best of our knowledge, the proposed method is the *first static* feature and sample reduction method for sparse SVM. Experiments on both synthetic and real datasets (e.g., the kddb dataset with about 20 million samples and 30 million features) demonstrate that our approach significantly outperforms state-of-the-art methods and the speedup gained by our approach can be orders of magnitude.

1. Introduction

Sparse support vector machine (SVM) (Bi et al., 2003; Wang et al., 2006) is a powerful technique that can simultaneously perform classification by margin maximization and variable selection by ℓ_1-norm penalty. The last few years have witnessed many successful applications of sparse SVMs, such as text mining (Joachims, 1998; Yoshikawa et al., 2014), bioinformatics (Narasimhan & Agarwal, 2013) and image processing (Mohr & Obermayer, 2004; Kotsia & Pitas, 2007). Many algorithms (Hastie et al., 2004; Fan et al., 2008; Catanzaro et al., 2008; Hsieh et al., 2008; Shalev-Shwartz et al., 2011) have been proposed to efficiently solve sparse SVM problems. However, the applications of sparse SVMs to large-scale learning problems, which involve a huge number of samples and extremely high-dimensional features, remain challenging.

An emerging technique, called *screening* (El Ghaoui et al., 2012), has been shown to be promising in accelerating large-scale sparse learning. The essential idea of screening is to quickly identify the zero coefficients in the sparse solutions without solving any optimization problems such that the corresponding features or samples—that are called *inactive* features or samples—can be removed from the training phase. Then, we only need to perform optimization on the reduced datasets instead of the full datasets, leading to substantial savings in the computational cost and memory usage. Here, we need to emphasize that screening differs greatly from feature selection methods, although they look similar at the first glance. To be precise, screening is devoted to accelerating the training of many sparse models including Lasso, Sparse SVM, etc., while feature selection is the goal of these models. In the past few years, many screening methods are proposed for a large set of sparse learning techniques, such as Lasso (Tibshirani et al., 2012; Xiang & Ramadge, 2012; Wang et al., 2013), group Lasso (Ndiaye et al., 2016), ℓ_1-regularized logistic regression (Wang et al., 2014), and SVM (Ogawa et al., 2013). Empirical studies indicate that screening methods can lead to orders of magnitude of speedup in computation time.

However, most existing screening methods study either feature screening or sample screening individually (Shibagaki et al., 2016) and their applications have very different scenarios. Specifically, to achieve better performance (say, in terms of speedup), we favor feature screening methods when the number of features p is much larger than the number of samples n, while sample screening methods are

[*] Equal contribution [1] State Key Lab of CAD&CG, Zhejiang University, China [2] Tencent AI Lab, Shenzhen, China [3] University of Michigan, USA. Correspondence to: Jie Wang <jiewangustc@gmail.com>.

Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).

preferable when $n \gg p$. Note that there is another class of sparse learning techniques, like sparse SVMs, which induce sparsities in both feature and sample spaces. All these screening methods are helpless in accelerating the training of these models with large n and p. We also cannot address this problem by simply combining the existing feature and sample screening methods. The reason is that they could mistakenly discard relevant data as they are specifically designed for different sparse models. Recently, Shibagaki et al. (Shibagaki et al., 2016) consider this problem and propose a method to simultaneously identify the inactive features and samples in a *dynamic* manner (Bonnefoy et al., 2014); that is, during the optimization process, they trigger their testing rule when there is a sufficient decrease in the duality gap. Thus, the method in (Shibagaki et al., 2016) can discard more inactive features and samples as the optimization proceeds and one has small-scale problems to solve in the late stage of the optimization. Nevertheless, the overall speedup can be limited as the problems' size can be large in the early stage of the optimization. To be specific, the method in (Shibagaki et al., 2016) depends heavily on the duality gap during the optimization process. The duality gap in the early stage can always be large, which makes the dual and primal estimations inaccurate and finally results in ineffective screening rules. Hence, it is essentially solving a large problem in the early stage.

In this paper, to address the limitations in the dynamic screening method, we propose a novel screening method that can **S**imultaneously identify **I**nactive **F**eatures and **S**amples (SIFS) for sparse SVMs in a *static* manner, that is, we only need to perform SIFS once *before* (instead of during) optimization. Thus, we only need to run the optimization algorithm on small-scale problems. The major technical challenge in developing SIFS is that we need to accurately estimate the primal and dual optima. The more accurate the estimations are, the more effective SIFS is in detecting inactive features and samples. Thus, our major technical contribution is a novel framework, which is based on the strong convexity of the primal and dual problems of sparse SVMs [see problems (P*) and (D*) in Section 2] for deriving accurate estimations of the primal and dual optima (see Section 3). Another appealing feature of SIFS is the so-called *synergy effect* (Shibagaki et al., 2016). Specifically, the proposed SIFS consists of two parts, i.e., **I**nactive **F**eature **S**creening (IFS) and **I**nactive **S**amples **S**creening (ISS). We show that discarding inactive features (samples) identified by IFS (ISS) leads to a more accurate estimation of the primal (dual) optimum, which in turn dramatically enhances the capability of ISS (IFS) in detecting inactive samples (features). Thus, SIFS applies IFS and ISS in an alternating manner until no more inactive features and samples can be identified, leading to much better performance in scaling up large-scale problems than the application of ISS or IFS individually. Moreover, SIFS (see Section 4) is safe in the sense that the detected features and samples are guaranteed to be absent from the sparse representations. To the best of our knowledge, SIFS is the first static screening rule for sparse SVM that is able to simultaneously detect inactive features and samples. Experiments (see Section 5) on both synthetic and real datasets demonstrate that SIFS significantly outperforms the state-of-the-art (Shibagaki et al., 2016) in improving the efficiency of sparse SVMs and the speedup can be orders of magnitude. Detailed proofs of theoretical results in the main text are in the supplementary supplements.

Notations: Let $\|\cdot\|_1$, $\|\cdot\|$, and $\|\cdot\|_\infty$ be the ℓ_1, ℓ_2, and ℓ_∞ norms, respectively. We denote the inner product of vectors $\mathbf{x}$ and $\mathbf{y}$ by $\langle \mathbf{x}, \mathbf{y} \rangle$, and the i-th component of $\mathbf{x}$ by $[\mathbf{x}]_i$. Let $[p] = \{1, 2..., p\}$ for a positive integer p. Given a subset $\mathcal{J} := \{j_1, ..., j_k\}$ of $[p]$, let $|\mathcal{J}|$ be the cardinality of $\mathcal{J}$. For a vector $\mathbf{x}$, let $[\mathbf{x}]_{\mathcal{J}} = ([\mathbf{x}]_{j_1}, ..., [\mathbf{x}]_{j_k})^T$. For a matrix $\mathbf{X}$, let $[\mathbf{X}]_{\mathcal{J}} = (\mathbf{x}_{j_1}, ..., \mathbf{x}_{j_k})$ and ${}_{\mathcal{J}}[\mathbf{X}] = ((\mathbf{x}^{j_1})^T, ..., (\mathbf{x}^{j_k})^T)^T$, where $\mathbf{x}^i$ and $\mathbf{x}_j$ are the i^{th} row and j^{th} column of $\mathbf{X}$, respectively. For a scalar t, we denote $\max\{0, t\}$ by $[t]_+$.

2. Basics and Motivations

In this section, we briefly review some basics of sparse SVMs and then motivate SIFS via the KKT conditions. Specifically, we focus on the ℓ_1-regularized SVM with a smoothed hinged loss that has strong theoretical guarantees (Shalev-Shwartz & Zhang, 2016), which takes the form of

$$\min_{\mathbf{w} \in \mathbb{R}^p} P(\mathbf{w}; \alpha, \beta) = \frac{1}{n} \sum_{i=1}^{n} \ell(1 - \langle \bar{\mathbf{x}}_i, \mathbf{w} \rangle) + \frac{\alpha}{2} \|\mathbf{w}\|^2 + \beta ||\mathbf{w}||_1, \tag{P*}$$

where $\mathbf{w}$ is the parameter vector to be estimated, $\{\mathbf{x}_i, y_i\}_{i=1}^n$ is the training set, $\mathbf{x}_i \in \mathbb{R}^p$, $y_i \in \{-1, +1\}$, $\bar{\mathbf{x}}_i = y_i \mathbf{x}_i$, α and β are positive parameters, and the loss function $\ell(\cdot) : \mathbb{R} \to \mathbb{R}$ is

$$\ell(t) = \begin{cases} 0, & \text{if } t < 0, \\ \frac{t^2}{2\gamma}, & \text{if } 0 \le t \le \gamma, \\ t - \frac{\gamma}{2}, & \text{if } t > \gamma, \end{cases}$$

where $\gamma \in (0, 1)$. We present the Lagrangian dual problem of problem (P*) and the KKT conditions in the following theorem, which plays a fundamentally important role in developing our screening rule.

Theorem 1. *Let $\bar{\mathbf{X}} = (\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, ..., \bar{\mathbf{x}}_n)$ and $\mathcal{S}_\beta(\cdot)$ be the soft-thresholding operator (Hastie et al., 2015), i.e., $[\mathcal{S}_\beta(\mathbf{u})]_i = \text{sign}([\mathbf{u}]_i)(|[\mathbf{u}]_i| - \beta)_+$. Then, for problem (P*), the followings hold:*

(i) : *The dual problem of (P*) is*

$$\min_{\theta\in[0,1]^n} D(\theta;\alpha,\beta) = \frac{1}{2\alpha}\left\|\mathcal{S}_\beta\left(\frac{1}{n}\bar{\mathbf{X}}\theta\right)\right\|^2 + \frac{\gamma}{2n}\|\theta\|^2 - \frac{1}{n}\langle\mathbf{1},\theta\rangle, \tag{D*}$$

where $\mathbf{1}\in\mathbb{R}^n$ is a vector with all components equal to 1.
(ii) : *Denote the optima of (P*) and (D*) by $\mathbf{w}^*(\alpha,\beta)$ and $\theta^*(\alpha,\beta)$, respectively. Then,*

$$\mathbf{w}^*(\alpha,\beta) = \frac{1}{\alpha}\mathcal{S}_\beta\left(\frac{1}{n}\bar{\mathbf{X}}\theta^*(\alpha,\beta)\right), \tag{KKT-1}$$

$$[\theta^*(\alpha,\beta)]_i = \begin{cases} 0, & \text{if } 1-\langle\bar{\mathbf{x}}_i,\mathbf{w}^*(\alpha,\beta)\rangle < 0; \\ 1, & \text{if } 1-\langle\bar{\mathbf{x}}_i,\mathbf{w}^*(\alpha,\beta)\rangle > \gamma; \\ \frac{1}{\gamma}(1-\langle\bar{\mathbf{x}}_i,\mathbf{w}^*(\alpha,\beta)\rangle), & \text{otherwise}. \end{cases} \tag{KKT-2}$$

According to KKT-1 and KKT-2, we define 4 index sets:

$$\begin{aligned}
\mathcal{F} &= \left\{j\in[p] : \frac{1}{n}|[\bar{\mathbf{X}}\theta^*(\alpha,\beta)]_j| \le \beta\right\}, \\
\mathcal{R} &= \{i\in[n] : 1-\langle\mathbf{w}^*(\alpha,\beta),\bar{\mathbf{x}}_i\rangle < 0\}, \\
\mathcal{E} &= \{i\in[n] : 1-\langle\mathbf{w}^*(\alpha,\beta),\bar{\mathbf{x}}_i\rangle \in [0,\gamma]\}, \\
\mathcal{L} &= \{i\in[n] : 1-\langle\mathbf{w}^*(\alpha,\beta),\bar{\mathbf{x}}_i\rangle > \gamma\},
\end{aligned}$$

which imply that

$$\begin{aligned}
&\text{(i): } i\in\mathcal{F} \Rightarrow [\mathbf{w}^*(\alpha,\beta)]_i = 0, \\
&\text{(ii): } \begin{cases} i\in\mathcal{R} \Rightarrow & [\theta^*(\alpha,\beta)]_i = 0, \\ i\in\mathcal{L} \Rightarrow & [\theta^*(\alpha,\beta)]_i = 1. \end{cases}
\end{aligned} \tag{R}$$

Thus, we call the j^{th} feature *inactive* if $j\in\mathcal{F}$. The samples in $\mathcal{E}$ are the so-called support vectors and we call the samples in $\mathcal{R}$ and $\mathcal{L}$ *inactive* samples.

Suppose that we are given subsets of $\mathcal{F}$, $\mathcal{R}$, and $\mathcal{L}$, then by (R), we can see that many coefficients of $\mathbf{w}^*(\alpha,\beta)$ and $\theta^*(\alpha,\beta)$ are known. Thus, we may have much less unknowns to solve and the problem size can be dramatically reduced. We formalize this idea in Lemma 1.

Lemma 1. *Given index sets $\hat{\mathcal{F}}\subseteq\mathcal{F}, \hat{\mathcal{R}}\subseteq\mathcal{R}$, and $\hat{\mathcal{L}}\subseteq\mathcal{L}$, the followings hold*
(i) : $[\mathbf{w}^*(\alpha,\beta)]_{\hat{\mathcal{F}}} = 0$, $[\theta^*(\alpha,\beta)]_{\hat{\mathcal{R}}} = 0$, $[\theta^*(\alpha,\beta)]_{\hat{\mathcal{L}}} = 1$.
(ii) : *Let $\hat{\mathcal{D}} = \hat{\mathcal{R}}\cup\hat{\mathcal{L}}$, $\hat{\mathbf{G}}_1 = {}_{\hat{\mathcal{F}}^c}[\bar{\mathbf{X}}]_{\hat{\mathcal{D}}^c}$, and $\hat{\mathbf{G}}_2 = {}_{\hat{\mathcal{F}}^c}[\bar{\mathbf{X}}]_{\hat{\mathcal{L}}}$, where $\hat{\mathcal{F}}^c = [p]\setminus\hat{\mathcal{F}}$, $\hat{\mathcal{D}}^c = [n]\setminus\hat{\mathcal{D}}$, and $\hat{\mathcal{L}}^c = [n]\setminus\hat{\mathcal{L}}$. Then, $[\theta^*(\alpha,\beta)]_{\hat{\mathcal{D}}^c}$ solves the following scaled dual problem:*

$$\min_{\hat{\theta}\in[0,1]^{|\hat{\mathcal{D}}^c|}} \left\{\frac{1}{2\alpha}\left\|\mathcal{S}_\beta\left(\frac{1}{n}\hat{\mathbf{G}}_1\hat{\theta} + \frac{1}{n}\hat{\mathbf{G}}_2\mathbf{1}\right)\right\|^2 + \frac{\gamma}{2n}\|\hat{\theta}\|^2 - \frac{1}{n}\langle\mathbf{1},\hat{\theta}\rangle\right\}. \tag{scaled-D*}$$

(iii) : *Suppose that $\theta^*(\alpha,\beta)$ is known. Then,*

$$[\mathbf{w}^*(\alpha,\beta)]_{\hat{\mathcal{F}}^c} = \frac{1}{\alpha}\mathcal{S}_\beta\left(\frac{1}{n}{}_{\hat{\mathcal{F}}^c}[\bar{\mathbf{X}}]\theta^*(\alpha,\beta)\right).$$

Lemma 1 indicates that, if we can identify index sets $\hat{\mathcal{F}}$ and $\hat{\mathcal{D}}$ and the cardinalities of $\hat{\mathcal{F}}^c$ and $\hat{\mathcal{D}}^c$ are much smaller than the feature dimension p and the dataset size n, we only need to solve a problem (scaled-D*) that may be much *smaller* than problem (D*) to exactly recover the optima $\mathbf{w}^*(\alpha,\beta)$ and $\theta^*(\alpha,\beta)$ *without sacrificing any accuracy*.

However, we cannot directly apply the rules in (R) to identify subsets of $\mathcal{F}$, $\mathcal{R}$, and $\mathcal{L}$, as they require the knowledge of $\mathbf{w}^*(\alpha,\beta)$ and $\theta^*(\alpha,\beta)$ that are usually unavailable. Inspired by the idea in (El Ghaoui et al., 2012), we can first estimate regions $\mathcal{W}$ and Θ that contain $\mathbf{w}^*(\alpha,\beta)$ and $\theta^*(\alpha,\beta)$, respectively. Then, by denoting

$$\hat{\mathcal{F}} := \left\{j\in[p] : \max_{\theta\in\Theta}\left\{\left|\frac{1}{n}[\bar{\mathbf{X}}\theta]_j\right|\right\} \le \beta\right\}, \tag{1}$$

$$\hat{\mathcal{R}} := \left\{i\in[n] : \max_{\mathbf{w}\in\mathcal{W}}\{1-\langle\mathbf{w},\bar{\mathbf{x}}_i\rangle\} < 0\right\}, \tag{2}$$

$$\hat{\mathcal{L}} := \left\{i\in[n] : \min_{\mathbf{w}\in\mathcal{W}}\{1-\langle\mathbf{w},\bar{\mathbf{x}}_i\rangle\} > \gamma\right\}, \tag{3}$$

since it is easy to know that $\hat{\mathcal{F}}\subset\mathcal{F}, \hat{\mathcal{R}}\subset\mathcal{R}$ and $\hat{\mathcal{L}}\subset\mathcal{L}$, the rules in (R) can be relaxed as follows:

$$\text{(i): } j\in\hat{\mathcal{F}} \Rightarrow [\mathbf{w}^*(\alpha,\beta)]_j = 0, \tag{R1}$$

$$\text{(ii): } \begin{cases} i\in\hat{\mathcal{R}} \Rightarrow & [\theta^*(\alpha,\beta)]_i = 0, \\ i\in\hat{\mathcal{L}} \Rightarrow & [\theta^*(\alpha,\beta)]_i = 1. \end{cases} \tag{R2}$$

In view of R1 and R2, we sketch the development of SIFS as follows.

Step 1: Derive estimations $\mathcal{W}$ and Θ such that $\mathbf{w}^*(\alpha,\beta)\in\mathcal{W}$ and $\theta^*(\alpha,\beta)\in\Theta$, respectively.

Step 2: Develop SIFS by deriving the relaxed screening rules R1 and R2, i.e., by solving the optimization problems in Eq. (1), Eq. (2) and Eq. (3).

3. Estimate the Primal and Dual Optima

In this section, we first show that the primal and dual optima admit closed form solutions for specific values of α and β (see Section 3.1). Then, in Sections 3.2 and 3.3, we present accurate estimations of the primal and dual optima, respectively.

3.1. Effective Intervals of the Parameters α and β

We first show that, if the value of β is sufficiently large, no matter what α is, the primal solution is 0.

Theorem 2. *Let $\beta_{\max} = \|\frac{1}{n}\bar{\mathbf{X}}\mathbf{1}\|_\infty$. Then, for $\alpha > 0$ and $\beta \geq \beta_{\max}$, we have*

$$\mathbf{w}^*(\alpha, \beta) = \mathbf{0}, \qquad \theta^*(\alpha, \beta) = \mathbf{1}.$$

For any β, the next result shows that, if α is large enough, the primal and dual optima admit closed form solutions.

Theorem 3. *If we denote*

$$\alpha_{\max}(\beta) = \frac{1}{1-\gamma} \max_{i \in [n]} \left\{ \langle \bar{\mathbf{x}}_i, \mathcal{S}_\beta(\frac{1}{n}\bar{\mathbf{X}}\mathbf{1}) \rangle \right\},$$

then for all $\alpha \in [\max\{\alpha_{\max}(\beta), 0\}, \infty) \cap (0, \infty)$, we have

$$\mathbf{w}^*(\alpha, \beta) = \frac{1}{\alpha}\mathcal{S}_\beta\left(\frac{1}{n}\bar{\mathbf{X}}\mathbf{1}\right), \quad \theta^*(\alpha, \beta) = \mathbf{1}. \tag{4}$$

By Theorems 2 and 3, we only need to consider the cases with $\beta \in (0, \beta_{\max}]$ and $\alpha \in (0, \alpha_{\max}(\beta)]$.

3.2. Primal Optimum Estimation

In Section 1, we mention that the proposed SIFS consists of IFS and ISS, and an alternating application of IFS and ISS can improve the estimation of the primal and dual optima, which can in turn make ISS and IFS more effective in identifying inactive samples and features, respectively. Lemma 2 shows that discarding inactive features by IFS leads to a more accurate estimation of the primal optimum.

Lemma 2. *Suppose that the reference solution $\mathbf{w}^*(\alpha_0, \beta_0)$ with $\beta_0 \in (0, \beta_{\max}]$ and $\alpha_0 \in (0, \alpha_{\max}(\beta_0)]$ is known. Consider problem (P*) with parameters $\alpha > 0$ and β_0. Let $\hat{\mathcal{F}}$ be the index set of the inactive features identified by the previous* IFS *steps, i.e., $[\mathbf{w}^*(\alpha, \beta_0)]_{\hat{\mathcal{F}}} = \mathbf{0}$. We define*

$$\mathbf{c} = \frac{\alpha_0 + \alpha}{2\alpha}[\mathbf{w}^*(\alpha_0, \beta_0)]_{\hat{\mathcal{F}}^c}, \tag{5}$$

$$\begin{aligned} r^2 = &\frac{(\alpha_0 - \alpha)^2}{4\alpha^2}\|\mathbf{w}^*(\alpha_0, \beta_0)\|^2 \\ &- \frac{(\alpha_0 + \alpha)^2}{4\alpha^2}\|[\mathbf{w}^*(\alpha_0, \beta_0)]_{\hat{\mathcal{F}}}\|^2. \end{aligned} \tag{6}$$

Then, the following holds:

$$[\mathbf{w}^*(\alpha, \beta_0)]_{\hat{\mathcal{F}}^c} \in \mathcal{W} := \{\mathbf{w} : \|\mathbf{w} - \mathbf{c}\| \leq r\}. \tag{7}$$

As $\hat{\mathcal{F}}$ is the index set of identified inactive features, we have $[\mathbf{w}^*(\alpha, \beta_0)]_{\hat{\mathcal{F}}} = \mathbf{0}$. Hence, we only need to find an accurate estimation of $[\mathbf{w}^*(\alpha, \beta_0)]_{\hat{\mathcal{F}}^c}$. Lemma 2 shows that $[\mathbf{w}^*(\alpha, \beta_0)]_{\hat{\mathcal{F}}^c}$ lies in a ball of radius r centered at $\mathbf{c}$. Note that, before we perform IFS, the set $\hat{\mathcal{F}}$ is empty and thus the second term on the right hand side (RHS) of Eq. (6) is 0. If we apply IFS multiple times (alternating with ISS), the set $\hat{\mathcal{F}}$ will be monotonically increasing. Thus, Eq. (6) implies that the radius will be monotonically decreasing, leading to a more accurate primal optimum estimation.

3.3. Dual Optimum Estimation

Similar to Lemma 2, the next result shows that ISS can improve the estimation of the dual optimum.

Lemma 3. *Suppose that the reference solution $\theta^*(\alpha_0, \beta_0)$ with $\beta_0 \in (0, \beta_{\max}]$ and $\alpha_0 \in (0, \alpha_{\max}(\beta_0)]$ is known. Consider problem (D*) with parameters $\alpha > 0$ and β_0. Let $\hat{\mathcal{R}}$ and $\hat{\mathcal{L}}$ be the index sets of inactive samples identified by the previous* ISS *steps, i.e., $[\theta^*(\alpha, \beta_0)]_{\hat{\mathcal{R}}} = \mathbf{0}$, $[\theta^*(\alpha, \beta_0)]_{\hat{\mathcal{L}}} = \mathbf{1}$, and $\hat{\mathcal{D}} = \hat{\mathcal{R}} \cup \hat{\mathcal{L}}$. We define*

$$\mathbf{c} = \frac{\alpha - \alpha_0}{2\gamma\alpha}\mathbf{1} + \frac{\alpha_0 + \alpha}{2\alpha}[\theta^*(\alpha_0, \beta_0)]_{\hat{\mathcal{D}}^c}, \tag{8}$$

$$\begin{aligned} r^2 = &\frac{(\alpha_0 - \alpha)^2}{4\alpha^2}\left\|\theta^*(\alpha_0, \beta_0) - \frac{1}{\gamma}\mathbf{1}\right\|^2 \\ &- \left\|\frac{(2\gamma - 1)\alpha + \alpha_0}{2\gamma\alpha}\mathbf{1} - \frac{\alpha_0 + \alpha}{2\alpha}[\theta^*(\alpha_0, \beta_0)]_{\hat{\mathcal{L}}}\right\|^2 \\ &- \left\|\frac{\alpha - \alpha_0}{2\gamma\alpha}\mathbf{1} + \frac{\alpha_0 + \alpha}{2\alpha}[\theta^*(\alpha_0, \beta_0)]_{\hat{\mathcal{R}}}\right\|^2. \end{aligned} \tag{9}$$

Then, the following holds:

$$[\theta^*(\alpha, \beta_0)]_{\hat{\mathcal{D}}^c} \in \Theta := \|\theta : \theta - \mathbf{c}\| \leq r. \tag{10}$$

Similar to Lemma 2, Lemma 3 also bounds $[\theta^*(\alpha, \beta_0)]_{\hat{\mathcal{D}}^c}$ by a ball. In view of Eq. (9), a similar discussion of Lemma 2—that is, the index sets $\hat{\mathcal{L}}$ and $\hat{\mathcal{R}}$ monotonically increase and thus the last two terms on the RHS of Eq. (9) monotonically increase when we perform ISS multiple times (alternating with IFS)—implies that the ISS steps can reduce the radius and thus improve the dual optimum estimation.

Remark 1. *To estimate $\mathbf{w}^*(\alpha, \beta_0)$ and $\theta^*(\alpha, \beta_0)$ by Lemmas 2 and 3, we have a free reference solution pair $\mathbf{w}^*(\alpha_0, \beta_0)$ and $\theta^*(\alpha_0, \beta_0)$ with $\alpha_0 = \alpha_{\max}(\beta_0)$. From Theorems 2 and 3, we know that in this setting, $\mathbf{w}^*(\alpha_0, \beta_0)$ and $\theta^*(\alpha_0, \beta_0)$ admit closed form solutions.*

4. The Proposed SIFS Screening Rule

We first present the IFS and ISS rules in Sections 4.1 and 4.2, respectively. Then, in Section 4.3, we develop the SIFS screening rule by an alternating application of IFS and ISS.

4.1. Inactive Feature Screening (IFS)

Suppose that $\mathbf{w}^*(\alpha_0, \beta_0)$ and $\theta^*(\alpha_0, \beta_0)$ are known, we derive IFS to identify inactive features for problem (P*) at (α, β_0) by solving the optimization problem in Eq. (1) (see Section E in the supplementary material):

$$s^i(\alpha, \beta_0) = \max_{\theta \in \Theta}\left\{\frac{1}{n}|\langle[\bar{\mathbf{x}}^i]_{\hat{\mathcal{D}}^c}, \theta\rangle + \langle[\bar{\mathbf{x}}^i]_{\hat{\mathcal{L}}}, \mathbf{1}\rangle|\right\}, i \in \hat{\mathcal{F}}^c, \tag{11}$$

where Θ is given by Eq. (10) and $\hat{\mathcal{F}}$ and $\hat{\mathcal{D}} = \hat{\mathcal{R}} \cup \hat{\mathcal{L}}$ are the index sets of inactive features and samples that have been identified in previous screening processes, respectively. The next result shows the closed form solution of problem (11).

Lemma 4. *Consider problem* (11)*. Let* $\mathbf{c}$ *and* r *be given by Eq. (8) and Eq. (9). Then, for all* $i \in \hat{\mathcal{F}}^c$*, we have*

$$s^i(\alpha, \beta_0) = \frac{1}{n}(|\langle [\bar{\mathbf{x}}^i]_{\hat{\mathcal{D}}^c}, \mathbf{c}\rangle + \langle [\bar{\mathbf{x}}^i]_{\hat{\mathcal{L}}}, \boldsymbol{1}\rangle| + \|[\bar{\mathbf{x}}^i]_{\hat{\mathcal{D}}^c}\| r).$$

We are now ready to present the IFS rule.

Theorem 4. *Consider problem* (P*)*. We suppose that* $\mathbf{w}^*(\alpha_0, \beta_0)$ *and* $\theta^*(\alpha_0, \beta_0)$ *are known. Then,*

(1): *The feature screening rule* IFS *takes the form of*

$$s^i(\alpha, \beta_0) \le \beta_0 \Rightarrow [\mathbf{w}^*(\alpha, \beta_0)]_i = 0, \forall i \in \hat{\mathcal{F}}^c \quad \text{(IFS)}$$

(2): *We update the index set* $\hat{\mathcal{F}}$ *by*

$$\hat{\mathcal{F}} \leftarrow \hat{\mathcal{F}} \cup \{i : s^i \le \beta_0, i \in \hat{\mathcal{F}}^c\}. \tag{12}$$

Recall that (Lemma 3), previous sample screening results give us a more tighter dual estimation, i.e., a smaller feasible region Θ for problem (11), which results in a smaller $s^i(\alpha, \beta_0)$. It finally leads us to a more powerful feature screening rule IFS. This is the so called synergy effect.

4.2. Inactive Sample Screening (ISS)

Similar to IFS, we derive ISS to identify inactive samples by solving the optimization problems in Eq. (2) and Eq. (3) (see Section G in the supplementary material for details):

$$u_i(\alpha, \beta_0) = \max_{\mathbf{w} \in \mathcal{W}} \{1 - \langle [\bar{\mathbf{x}}_i]_{\hat{\mathcal{F}}^c}, \mathbf{w}\rangle\}, i \in \hat{\mathcal{D}}^c, \tag{13}$$

$$l_i(\alpha, \beta_0) = \min_{\mathbf{w} \in \mathcal{W}} \{1 - \langle [\bar{\mathbf{x}}_i]_{\hat{\mathcal{F}}^c}, \mathbf{w}\rangle\}, i \in \hat{\mathcal{D}}^c, \tag{14}$$

where $\mathcal{W}$ is given by Eq. (7) and $\hat{\mathcal{F}}$ and $\hat{\mathcal{D}} = \hat{\mathcal{R}} \cup \hat{\mathcal{L}}$ are the index sets of inactive features and samples that have been identified in previous screening processes. We show that problems (13) and (14) admit closed form solutions.

Lemma 5. *Consider problems* (13) *and* (14)*. Let* $\mathbf{c}$ *and* r *be given by Eq. (5) and Eq. (6). Then,*

$$u_i(\alpha, \beta_0) = 1 - \langle [\bar{\mathbf{x}}_i]_{\hat{\mathcal{F}}^c}, \mathbf{c}\rangle + \|[\bar{\mathbf{x}}_i]_{\hat{\mathcal{F}}^c}\| r, i \in \hat{\mathcal{D}}^c,$$
$$l_i(\alpha, \beta_0) = 1 - \langle [\bar{\mathbf{x}}_i]_{\hat{\mathcal{F}}^c}, \mathbf{c}\rangle - \|[\bar{\mathbf{x}}_i]_{\hat{\mathcal{F}}^c}\| r, i \in \hat{\mathcal{D}}^c.$$

We are now ready to present the ISS rule.

Theorem 5. *Consider problem* (D*)*. We suppose that* $\mathbf{w}^*(\alpha_0, \beta_0)$ *and* $\theta^*(\alpha_0, \beta_0)$ *are known. Then,*

(1): *The sample screening rule* ISS *takes the form of*

$$\begin{matrix} u_i(\alpha, \beta_0) < 0 \Rightarrow [\theta^*(\alpha, \beta_0)]_i = 0, \\ l_i(\alpha, \beta_0) > \gamma \Rightarrow [\theta^*(\alpha, \beta_0)]_i = 1, \end{matrix} \forall i \in \hat{\mathcal{D}}^c \quad \text{(ISS)}$$

(2): *We update the the index sets* $\hat{\mathcal{R}}$ *and* $\hat{\mathcal{L}}$ *by*

$$\hat{\mathcal{R}} \leftarrow \hat{\mathcal{R}} \cup \{i : u_i(\alpha, \beta_0) < 0, i \in \hat{\mathcal{D}}^c\}, \tag{15}$$

$$\hat{\mathcal{L}} \leftarrow \hat{\mathcal{L}} \cup \{i : l_i(\alpha, \beta_0) > \gamma, i \in \hat{\mathcal{D}}^c\}. \tag{16}$$

The synergy effect also exists here. Recall that (Lemma 2), previous feature screening results lead a smaller feasible region $\mathcal{W}$ for the problems (13) and (14), which results in smaller $u_i(\alpha, \beta_0)$ and bigger $l_i(\alpha, \beta_0)$. It finally leads us to a more accurate sample screening rule ISS.

4.3. The Proposed SIFS Rule by An Alternating Application of IFS and ISS

In real applications, the optimal parameter values of α and β are usually unknown. To determine appropriate parameter values, common approaches, like cross validation and stability selection, need to solve the model over a grid of parameter values $\{(\alpha_{i,j}, \beta_j) : i \in [M], j \in [N]\}$ with $\beta_{\max} > \beta_1 > ... > \beta_N > 0$ and $\alpha_{\max}(\beta_j) > \alpha_{1,j} > ... > \alpha_{M,j} > 0$. This can be very time-consuming. Inspired by Strong Rule (Tibshirani et al., 2012) and SAFE (El Ghaoui et al., 2012), we develop a sequential version of SIFS in Algorithm 1. Specifically, given the primal and dual optima $\mathbf{w}^*(\alpha_{i-1,j}, \beta_j)$ and $\theta^*(\alpha_{i-1,j}, \beta_j)$ at $(\alpha_{i-1,j}, \beta_j)$, we apply SIFS to identify the inactive features and samples for problem (P*) at $(\alpha_{i,j}, \beta_j)$. Then, we perform optimization on the reduced dataset and solve the primal and dual optima

Algorithm 1 SIFS

```
Input: $\beta_{\max} > \beta_1 > ... > \beta_N > 0$ and $\alpha_{\max}(\beta_j) = \alpha_{0,j} > \alpha_{1,j} > ... > \alpha_{M,j} > 0$.
for $j = 1$ to $N$ do
  Compute the first reference solution $\mathbf{w}^*(\alpha_{0,j}, \beta_j)$ and $\theta^*(\alpha_{0,j}, \beta_j)$ using the close-form formula (4).
  for $i = 1$ to $M$ do
    Initialization: $\hat{\mathcal{F}} = \hat{\mathcal{R}} = \hat{\mathcal{L}} = \emptyset$
    repeat
      Run sample screening using rule ISS based on $\mathbf{w}^*(\alpha_{i-1,j}, \beta_j)$.
      Update $\hat{\mathcal{R}}$ and $\hat{\mathcal{L}}$ by Eq. (15) and Eq. (16), respectively.
      Run feature screening using rule IFS based on $\theta^*(\alpha_{i-1,j}, \beta_j)$.
      Update $\hat{\mathcal{F}}$ by Eq. (12).
    until No new inactive features or samples are identified
    Compute $\mathbf{w}^*(\alpha_{i,j}, \beta_j)$ and $\theta^*(\alpha_{i,j}, \beta_j)$ by solving the scaled problem.
  end for
end for
Output: $\mathbf{w}^*(\alpha_{i,j}, \beta_j)$ and $\theta^*(\alpha_{i,j}, \beta_j), i \in [M], j \in [N]$.
```

at $(\alpha_{i,j}, \beta_j)$. We repeat this process until we solve problem (P*) at all pairs of parameter values.

Note that we insert $\alpha_{0,j}$ into every sequence $\{\alpha_{i,j} : i \in [M]\}$ (see line 1 in Algorithm 1) to obtain a closed-form solution as the first reference solution. In this way, we can avoid solving problem at $(\alpha_{1,j}, \beta_j), j \in [N]$ directly (without screening), which is time consuming. At last, we would like to point out that the values $\{(\alpha_{i,j}, \beta_j) : i \in [M], j \in [N]\}$ in SIFS can be specified by users arbitrarily.

SIFS applies ISS and IFS in an alternating manner to reinforce their capability in identifying inactive samples and features. In Algorithm 1, we apply ISS first. Of course, we can also apply IFS first. The theorem below demonstrates that the orders have no impact on the performance of SIFS.

Theorem 6. *Given the optimal solutions $\mathbf{w}^*(\alpha_{i-1,j}, \beta_j)$ and $\theta^*(\alpha_{i-1,j}, \beta_j)$ at $(\alpha_{i-1,j}, \beta_j)$ as the reference solution pair at $(\alpha_{i,j}, \beta_j)$ for SIFS, we assume SIFS with ISS first stops after applying IFS and ISS for p times and denote the identified inactive features and samples as $\hat{\mathcal{F}}_p^A, \hat{\mathcal{R}}_p^A$ and $\hat{\mathcal{L}}_p^A$. Similarly, when we apply IFS first, the results are denoted as $\hat{\mathcal{F}}_q^B, \hat{\mathcal{R}}_q^B$ and $\hat{\mathcal{L}}_q^B$. Then, the followings hold:*
(1) $\hat{\mathcal{F}}_p^A = \hat{\mathcal{F}}_q^B, \hat{\mathcal{R}}_p^A = \hat{\mathcal{R}}_q^B$ *and* $\hat{\mathcal{L}}_p^A = \hat{\mathcal{L}}_q^B$.
(2) *With different orders of applying ISS and IFS, the difference of the times of ISS and IFS we need to apply in SIFS can never be larger than 1, that is, $|p - q| \leq 1$.*

Remark 2. *From Remark 1, we can see that our SIFS can also be applied to solve a single problem, due to the existence of the free reference solution pair.*

5. Experiments

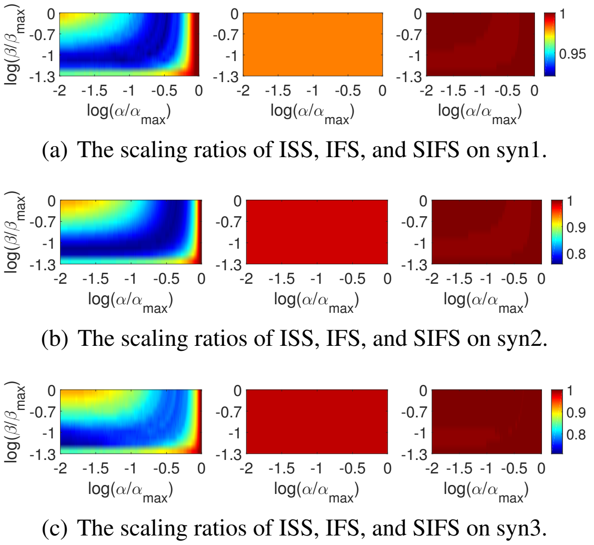
We evaluate SIFS on both synthetic and real datasets in terms of three measurements. The first one is the *scaling ratio*: $1 - \frac{(n-\tilde{n})(p-\tilde{p})}{np}$, where $\tilde{n}$, $\tilde{p}$, n, and p are the numbers of inactive samples and features identified by SIFS, sample size, and feature dimension of the datasets. The second measure is *rejection ratios* of each triggering of ISS and IFS in SIFS: $\frac{\tilde{n}_i}{n_0}$ and $\frac{\tilde{p}_i}{p_0}$, where $\tilde{n}^i$ and $\tilde{p}^i$ are the numbers of inactive samples and features identified in i-th triggering of ISS and IFS in SIFS. n_0 and p_0 are the numbers of inactive samples and features in the solution. The third measure is *speedup*, i.e., the ratio of the running time of the solver without screening to that with screening.

Recall that, we can integrate SIFS with any solvers for problem (P*). In this experiment, we use Accelerated Proximal Stochastic Dual Coordinate Ascent (Accelerated-Prox-SDCA) (Shalev-Shwartz & Zhang, 2016), as it is one of the state-of-the-arts. As we mentioned in the introduction section that screening differs greatly from features selection methods, it is not appropriate to make comparisons with feature selection methods. To this end, we only choose the state-of-art screening method for Sparse SVMs in (Shibagaki et al., 2016) as a baseline in the experiments.

For each dataset, we solve problem (P*) at a grid of turning parameter values. Specifically, we first compute $\beta_{\max}$ by Theorem 2 and then select 10 values of β that are equally spaced on the logarithmic scale of $\beta/\beta_{\max}$ from 1 to 0.05. Then, for each value of β, we first compute $\alpha_{\max}(\beta)$ by Theorem 3 and then select 100 values of α that are equally spaced on the logarithmic scale of $\alpha/\alpha_{\max}(\beta)$ from 1 to 0.01. Thus, for each dataset, we solve problem (P*) at 1000 pairs of parameter values in total. We write the code in C++ along with Eigen library for some numerical computations. We perform all the computations on a single core of Intel(R) Core(TM) i7-5930K 3.50GHz, 128GB MEM.

5.1. Simulation Studies

We evaluate SIFS on 3 synthetic datasets named syn1, syn2 and syn3 with sample and feature size $(n, p) \in \{(10000, 1000), (10000, 10000), (1000, 10000)\}$. We present each data point as $\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2]$ with $\mathbf{x}_1 \in \mathbb{R}^{0.02p}$ and $\mathbf{x}_2 \in \mathbb{R}^{0.98p}$. We use Gaussian distributions $\mathcal{G}_1 = N(\mathbf{u}, 0.75\mathbf{I}), \mathcal{G}_2 = N(-\mathbf{u}, 0.75\mathbf{I})$ and $\mathcal{G}_3 = N(0, 1)$ to generate the data points, where $\mathbf{u} = 1.5\mathbf{1}$ and $\mathbf{I} \in \mathbb{R}^{0.02p \times 0.02p}$ is the identity matrix. To be precise, $\mathbf{x}_1$ for positive and negative points are sampled from $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively. For each entry in $\mathbf{x}_2$, it has chance $\eta = 0.02$ to be sampled from $\mathcal{G}_3$ and chance $1 - \eta$ to be 0.



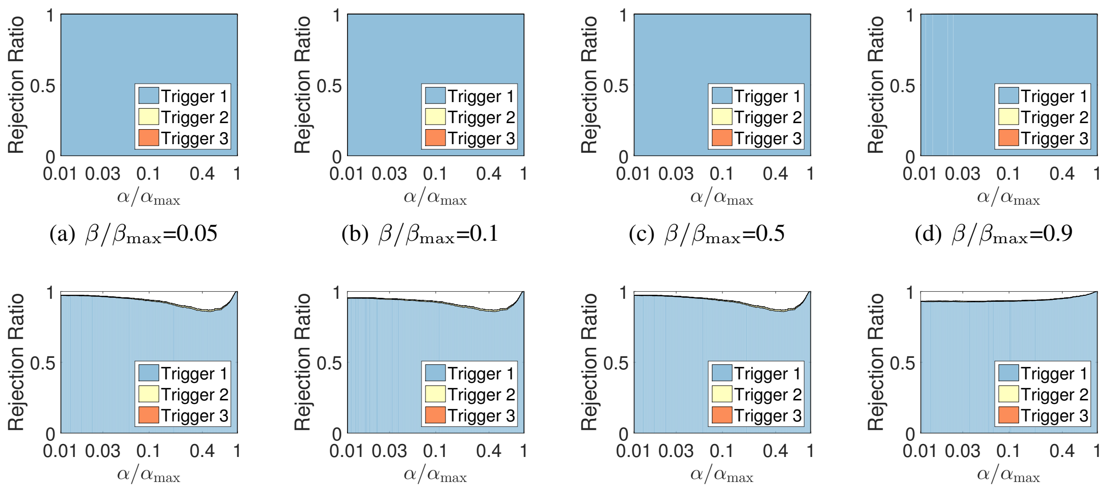
(a) The scaling ratios of ISS, IFS, and SIFS on syn1.

(b) The scaling ratios of ISS, IFS, and SIFS on syn2.

(c) The scaling ratios of ISS, IFS, and SIFS on syn3.

Figure 1. Scaling ratios of ISS, IFS and SIFS (from left to right).

Fig. 1 shows the scaling ratios by ISS, IFS, and SIFS on the synthetic datasets at 1000 parameter values. We can see that IFS is more effective in scaling problem size than ISS, with scaling ratios roughly 98% against 70 − 90%. Moreover, SIFS, which is an alternating application of IFS and ISS, significantly outperforms ISS and IFS, with scal-



(a) $\beta/\beta_{\max}$=0.05 (b) $\beta/\beta_{\max}$=0.1 (c) $\beta/\beta_{\max}$=0.5 (d) $\beta/\beta_{\max}$=0.9

(e) $\beta/\beta_{\max}$=0.05 (f) $\beta/\beta_{\max}$=0.1 (g) $\beta/\beta_{\max}$=0.5 (h) $\beta/\beta_{\max}$=0.9

Figure 2. Rejection ratios of SIFS on syn 2 (first row: Feature Screening, second row: Sample Screening).

Table 1. Running time (in seconds) for solving problem (P*) at 1000 pairs of parameter values on three synthetic datasets.

Data	Solver	ISS+Solver			IFS+Solver			SIFS+Solver		
		ISS	Solver	Speedup	IFS	Solver	Speedup	SIFS	Solver	Speedup
syn1	499.1	4.9	27.8	15.3	2.3	42.6	11.1	8.6	6.0	**34.2**
syn2	8749.9	24.9	1496.6	5.8	23.0	288.1	28.1	92.6	70.3	**53.7**
syn3	1279.7	2.0	257.1	4.9	2.2	33.4	36.0	7.2	9.5	**76.8**

ing ratios roughly 99.9%. This high scaling ratios imply that SIFS can lead to a significant speedup.

Due to the space limitation, we only report the rejection ratios of SIFS on syn2. Other results can be found in the supplementary material. Fig. 2 shows that SIFS can identify most of the inactive features and samples. However, few features and samples are identified in the second and later triggerings of ISS and IFS. The reason may be that the task here is so simple that one triggering is enough.

Table 1 reports the running time of solver without and with IFS, ISS and SIFS for solving problem (P*) at 1000 pairs of parameter values. We can see that SIFS leads to significant speedups, that is, up to 76.8 times. Taking syn2 for example, without SIFS, the solver takes more than two hours to solve problem (P*) at 1000 pairs of parameter values. However, combined with SIFS, the solver only needs less than three minutes for solving the same set of problems. From the theoretical analysis in (Shalev-Shwartz & Zhang, 2016) for Accelerated-Prox-SDCA, we can see that its computational complexity rises proportionately to the sample size n and the feature dimension p. From this theoretical result, we can see that the results in Figure 1 are roughly consistent with the speedups we achieved shown in Table 1.

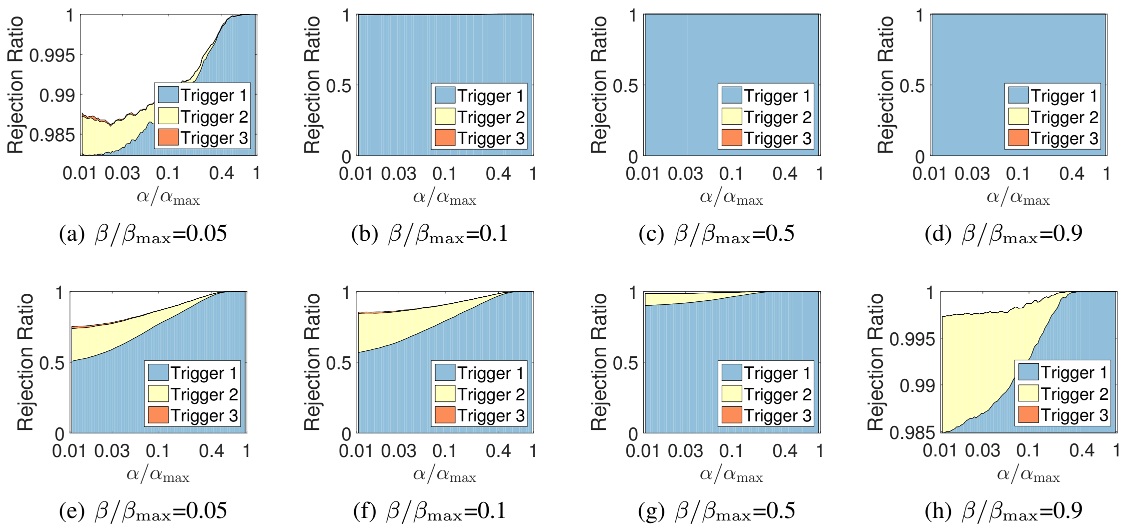
5.2. Experiments on Real Datasets

In this experiment, we evaluate the performance of SIFS on 5 large-scale real datasets: real-sim, rcv1-train, rcv1-test, url, and kddb, which are all collected from the project page of LibSVM (Chang & Lin, 2011). See Table 2 for a brief summary. We note that, the kddb dataset has about 20 million samples with 30 million features.

Table 2. Statistics of the real datasets.

Dataset	Feature size: p	Sample size:n
real-sim	20,958	72,309
rcv1-train	47,236	20,242
rcv1-test	47,236	677, 399
url	3,231,961	2,396,130
kddb	29,890,095	19,264,097

Recall that, SIFS detects the inactive features and samples in a static manner, i.e., we perform SIFS only once before the optimization and thus the size of the problem we need to perform optimization on is fixed. However, the method in (Shibagaki et al., 2016) detects inactive features and samples in a dynamic manner (Bonnefoy et al., 2014), i.e., they perform their method along with the optimization and thus the size of the problem would keep decreasing during the iterative process. Thus, comparing SIFS with the method in (Shibagaki et al., 2016) in terms of rejection ratios is inapplicable. We compare the performance of SIFS with the method in (Shibagaki et al., 2016) in terms of speedup. Specifically, we compare the speedup gained by SIFS and the method in (Shibagaki et al., 2016) for solving problem (P*) at 1000 pairs of parameter values. The code of the method in (Shibagaki et al., 2016) is obtained from (`https://github.com/husk214/s3fs`).



(a) $\beta/\beta_{\max}$=0.05 (b) $\beta/\beta_{\max}$=0.1 (c) $\beta/\beta_{\max}$=0.5 (d) $\beta/\beta_{\max}$=0.9

(e) $\beta/\beta_{\max}$=0.05 (f) $\beta/\beta_{\max}$=0.1 (g) $\beta/\beta_{\max}$=0.5 (h) $\beta/\beta_{\max}$=0.9

Figure 3. Rejection ratios of SIFS on the real-sim dataset (first row: Feature Screening, second row: Sample Screening).

Table 3. Running time (in seconds) for solving problem (P*) at 1000 pairs of parameter values on five real datasets.

Data Set	Solver	Method in (Shibagaki et al., 2016)+Solver			SIFS+Solver		
		Screen	Solver	Speedup	Screen	Solver	**Speedup**
real-sim	3.93E+04	24.10	4.94E+03	7.91	60.01	140.25	**195.00**
rcv1-train	2.98E+04	10.00	3.73E+03	7.90	27.11	80.11	**277.10**
rcv1-test	1.10E+06	398.00	1.35E+05	8.10	1.17E+03	2.55E+03	**295.11**
url	—	3.18E+04	8.60E+05	—	7.66E+03	2.91E+04	—
kddb	—	4.31E+04	1.16E+06	—	1.10E+04	3.6E+04	—

Fig. 3 shows the rejection ratios of SIFS on the real-sim dataset (other results are in the supplementary material). In Fig. 3, we can see that some inactive features and samples are identified in the 2nd and 3rd triggering of ISS and IFS, which verifies the necessity of the alternating application of ISS and IFS. SIFS is efficient since it always stops in 3 times of triggering. In addition, most of ($> 98\%$) the inactive features can be identified in the 1st triggering of IFS while identifying inactive samples needs to apply ISS two or more times. It may result from two reasons: 1) We run ISS first, which reinforces the capability of IFS due to the synergy effect (see Sections 4.1 and 4.2), see Section L.1 in the supplementary material for further verification; 2) Feature screening here may be easier than sample screening.

Table 3 reports the running time of solver without and with the method in (Shibagaki et al., 2016) and SIFS for solving problem (P*) at 1000 pairs of parameter values on real datasets. The speedup gained by SIFS is up to 300 times on real-sim, rcv1-train and rcv1-test. Moreover, SIFS significantly outperforms the method in (Shibagaki et al., 2016) in terms of speedup—by about 30 to 40 times faster on the aforementioned three datasets. For datasets url and kddb, we do not report the results of the solver as the sizes of the datasets are huge and the computational cost is prohibitive. Instead, we can see that the solver with SIFS is about 25 times faster than the solver with the method in (Shibagaki et al., 2016) on both datasets url and kddb. Take the dataset kddb as an example. The solver with SIFS takes about 13 hours to solve problem (P*) for all 1000 pairs of parameter values, while the solver with the method in (Shibagaki et al., 2016) needs 11 days to finish the same task.

6. Conclusion

In this paper, we develop a novel data reduction method SIFS to simultaneously identify inactive features and samples for sparse SVM. Our major contribution is a novel framework for an accurate estimation of the primal and d-ual optima based on strong convexity. To the best of our knowledge, the proposed SIFS is the first static screening method that is able to simultaneously identify inactive features and samples for sparse SVMs. An appealing feature of SIFS is that all detected features and samples are guaranteed to be irrelevant to the outputs. Thus, the model learned on the reduced data is identical to the one learned on the full data. Experiments on both synthetic and real datasets demonstrate that SIFS can dramatically reduce the problem size and the resulting speedup can be orders of magnitude. We plan to generalize SIFS to more complicated models, e.g., SVM with a structured sparsity-inducing penalty.

Acknowledgements

This work was supported by the National Basic Research Program of China (973 Program) under Grant 2013CB336500, National Natural Science Foundation of China under Grant 61233011 and National Youth Top-notch Talent Support Program.

References

Bi, Jinbo, Bennett, Kristin, Embrechts, Mark, Breneman, Curt, and Song, Minghu. Dimensionality reduction via sparse support vector machines. *The Journal of Machine Learning Research*, 3:1229–1243, 2003.

Bonnefoy, Antoine, Emiya, Valentin, Ralaivola, Liva, and Gribonval, Rémi. A dynamic screening principle for the lasso. In *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*, pp. 6–10. IEEE, 2014.

Catanzaro, Bryan, Sundaram, Narayanan, and Keutzer, Kurt. Fast support vector machine training and classification on graphics processors. In *Proceedings of the 25th international conference on Machine learning*, pp. 104–111. ACM, 2008.

Chang, Chih-Chung and Lin, Chih-Jen. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

El Ghaoui, Laurent, Viallon, Vivian, and Rabbani, Tarek. Safe feature elimination in sparse supervised learning. *Pacific Journal of Optimization*, 8:667–698, 2012.

Fan, Rong-En, Chang, Kai-Wei, Hsieh, Cho-Jui, Wang, Xiang-Rui, and Lin, Chih-Jen. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.

Hastie, Trevor, Rosset, Saharon, Tibshirani, Robert, and Zhu, Ji. The entire regularization path for the support vector machine. *The Journal of Machine Learning Research*, 5:1391–1415, 2004.

Hastie, Trevor, Tibshirani, Robert, and Wainwright, Martin. *Statistical learning with sparsity: the lasso and generalizations*. CRC Press, 2015.

Hsieh, Cho-Jui, Chang, Kai-Wei, Lin, Chih-Jen, Keerthi, S Sathiya, and Sundararajan, Sellamanickam. A dual coordinate descent method for large-scale linear svm. In *Proceedings of the 25th international conference on Machine learning*, pp. 408–415. ACM, 2008.

Joachims, Thorsten. *Text categorization with support vector machines: Learning with many relevant features*. Springer, 1998.

Kotsia, Irene and Pitas, Ioannis. Facial expression recognition in image sequences using geometric deformation features and support vector machines. *Image Processing, IEEE Transactions on*, 16(1):172–187, 2007.

Mohr, Johannes and Obermayer, Klaus. A topographic support vector machine: Classification using local label configurations. In *Advances in Neural Information Processing Systems*, pp. 929–936, 2004.

Narasimhan, Harikrishna and Agarwal, Shivani. Svm pauc tight: a new support vector method for optimizing partial auc based on a tight convex upper bound. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 167–175. ACM, 2013.

Ndiaye, Eugene, Fercoq, Olivier, Gramfort, Alexandre, and Salmon, Joseph. Gap safe screening rules for sparse-group lasso. In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, pp. 388–396. Curran Associates, Inc., 2016.

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

Shalev-Shwartz, Shai and Zhang, Tong. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, 155(1-2): 105–145, 2016.

Shalev-Shwartz, Shai, Singer, Yoram, Srebro, Nathan, and Cotter, Andrew. Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical programming*, 127(1):3–30, 2011.

Shibagaki, Atsushi, Karasuyama, Masayuki, Hatano, Kohei, and Takeuchi, Ichiro. Simultaneous safe screening of features and samples in doubly sparse modeling. In *Proceedings of The 33rd International Conference on Machine Learning*, 2016.

Tibshirani, Robert, Bien, Jacob, Friedman, Jerome, Hastie, Trevor, Simon, Noah, Taylor, Jonathan, and Tibshirani, Ryan J. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):245–266, 2012.

Wang, Jie, Zhou, Jiayu, Wonka, Peter, and Ye, Jieping. Lasso screening rules via dual polytope projection. In *Advances in Neural Information Processing Systems*, pp. 1070–1078, 2013.

Wang, Jie, Zhou, Jiayu, Liu, Jun, Wonka, Peter, and Ye, Jieping. A safe screening rule for sparse logistic regression. In *Advances in Neural Information Processing Systems*, pp. 1053–1061, 2014.

Wang, Li, Zhu, Ji, and Zou, Hui. The doubly regularized support vector machine. *Statistica Sinica*, pp. 589–615, 2006.

Xiang, Zhen James and Ramadge, Peter J. Fast lasso screening tests based on correlations. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 2137–2140. IEEE, 2012.

Yoshikawa, Yuya, Iwata, Tomoharu, and Sawada, Hiroshi. Latent support measure machines for bag-of-words data classification. In *Advances in Neural Information Processing Systems*, pp. 1961–1969, 2014.