Fast OSCAR and OWL Regression via Safe Screening Rules

Runxue Bao [1] **Bin Gu** [2] **Heng Huang** [1 2]

Abstract

Ordered Weighted L_1 (OWL) regularized regression is a new regression analysis for high-dimensional sparse learning. Proximal gradient methods are used as standard approaches to solve OWL regression. However, it is still a burning issue to solve OWL regression due to considerable computational cost and memory usage when the feature or sample size is large. In this paper, we propose the first safe screening rule for OWL regression by exploring the order of the primal solution with the unknown order structure via an iterative strategy, which overcomes the difficulties of tackling the non-separable regularizer. It effectively avoids the updates of the parameters whose coefficients must be zero during the learning process. More importantly, the proposed screening rule can be easily applied to standard and stochastic proximal gradient methods. Moreover, we prove that the algorithms with our screening rule are guaranteed to have identical results with the original algorithms. Experimental results on a variety of datasets show that our screening rule leads to a significant computational gain without any loss of accuracy, compared to existing competitive algorithms.

1. Introduction

OWL regression (Bogdan et al., 2013; Zeng & Figueiredo, 2014; Bogdan et al., 2015; Figueiredo & Nowak, 2016; Bao et al., 2019) has emerged as a useful procedure for high-dimensional sparse regression recently, which can promote the sparsity and grouping simultaneously. Unlike group Lasso (Yuan & Lin, 2006) and its variants, OWL regression can identify precise grouping structures of strongly correlated covariates automatically during the learning process without any prior information of feature groups. Remarkably, (Bu et al., 2019) concluded that it has two good properties to achieve the minimax estimation from the estimation side without any prior knowledge of coefficients (Su et al., 2016; Bellec et al., 2018) and controls the false discovery rate from the testing side (Bogdan et al., 2015; Brzyski et al., 2019), which do not simultaneously exist in other models such as Lasso (Tibshirani, 1996) and knockoffs (Barber et al., 2015). Owing to its effectiveness, OWL is widely used in various kinds of applications, *e.g.*, gene expression (Bogdan et al., 2015), brain networks (Oswal et al., 2016) and neural networks training (Zhang et al., 2018).

Although proximal gradient methods are used as standard approaches (Bondell & Reich, 2008; Bogdan et al., 2015) to solve OWL regression, it still suffers from high computational cost and memory usage when the feature or sample size is large in practice. The main bottleneck is the computation to update the solution in each iteration depends on all the data points. The screening technique is an easy-to-implement and promising approach for accelerating the training of sparse learning models by eliminating the features whose coefficients must be zero, which can safely avoid these useless computation during the whole training process.

The safe screening rules introduced by (Laurent El Ghaoui, 2012) for generalized l_1 regularized problems eliminate features whose associated coefficients are proved to be zero at the optimum. The screening rule in (Laurent El Ghaoui, 2012) is called static safe rules, which is only performed once, prior to any optimization algorithm. Relaxing the safe rules, heuristic strategies, called strong rules (Tibshirani et al., 2012), reduce the computational cost using an active set strategy at the price of possible mistakes, which requires difficult post-processing to check for features possibly wrongly discarded. Another road to screening method is called sequential safe rules (Wang et al., 2013; Xiang et al., 2016). The sequential screening rule relies on the exact dual optimal solution, which could be very time-consuming and lead to be unsafe in practice. Recently, the introduction of safe dynamic rules (Fercoq et al., 2015) has opened a promising venue by conducting safe screening not only at the beginning of the algorithm, but also during the learning process. Following (Fercoq et al., 2015) for Lasso, many dynamic screening rules relying on the duality gap

[1]Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, USA [2]JD Finance America Corporation. Correspondence to: Heng Huang <heng.huang@pitt.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

Table 1. Representative safe screening algorithms. "Type of screening" represents the algorithm screening samples or features. "Size" represents the number of the hyperparameters in regularization where g is the group number of sparse-group Lasso and d is feature size. "Fixation" represents whether the regularization hyperparameter of each variable is fixed during the learning process.

Problem	Type of screening	Size	Separability	Fixation
Lasso (Liu et al., 2014)	features	1	Separable	Fixed
Lasso (Fercoq et al., 2015)	features	1	Separable	Fixed
Sparse SVM (Shibagaki et al., 2016)	features and samples	3	Separable	Fixed
Sparse-group Lasso (Ndiaye et al., 2016)	features	g+2	Separable	Fixed
Sparse SVM (Zhang et al., 2017)	features and samples	3	Separable	Fixed
Proximal Weighted Lasso (Rakotomamonjy et al., 2019)	features	d	Separable	Fixed
OWL regression (Ours)	features	d	Non-separable	Unfixed

are proposed in (Shibagaki et al., 2016; Ndiaye et al., 2016; Rakotomamonjy et al., 2019; Zhai et al., 2019) for a broad class of sparse learning problems with both good empirical and theoretical results.

This work is concerned with algorithmic acceleration of OWL regression through safe screening rules to safely avoid useless computation whose parameters must be zero during the training process without any influence on the final learned model. We summarized several representative safe screening algorithms in Table 1. It shows that existing safe screening rules have been widely used to accelerate algorithms in sparse learning by screening useless samples or features while all of them are limited to separable penalties and the fixed regularization hyperparameter of each variable, which is essential to derive the screening rules. So far there are still no safe screening rules proposed for OWL regression. This vacuum is because OWL penalty is non-separable, meaning it cannot be written as $\Omega_\lambda(\beta) = \sum_{i=1}^{d} \lambda_i \omega(\beta_i)$. Thus, all the hyperparameters for each variable in OWL penalty are unfixed until we finish the whole learning process while they are fixed in other models at the initial stage. Besides, how to derive an efficient screening rule with the numerous hyperparameters is another key point to be considered. Because of the challenges to derive screening rules for the non-separable OWL penalty with numerous unfixed hyperparameters, speeding up OWL regression by screening rules is still an open and challenging problem.

To address these challenges, in this paper, we propose a safe screening rule for the linear regression with the family of OWL regularizers based on the intermediate duality gap, which is significantly helpful for accelerating the training algorithms. As far as we know, this work is the first attempt in this direction. We effectively overcome the difficulties caused by the non-separable penalty by exploring the order of the primal solution with the unknown order structure via an iterative strategy, which leads to better understanding of the non-separable penalty for future. Specifically, in high-dimensional tasks, as the size of non-zero coefficients is much smaller than the size of features, our screening rule can effectively identify the features whose parameters must be zero in each iteration and then accelerate the original algorithms by skipping the useless updates of these parameters. Theoretically, we not only rigorously prove that our screening rule is safe for the whole training process, but also prove that our screening rule can be safely applied to existing standard iterative optimization algorithms both in the batch and stochastic setting without any loss of accuracy. The empirical performance shows the superiority of our algorithms with significant computational gain to the most popular proximal gradient methods, *e.g.*, APGD (accelerated proximal gradient descent) algorithm and SPGD (stochastic proximal gradient descent with variance reduction) algorithm.

2. Preliminary

2.1. OWL Regularized Regression

We consider the linear regression with the family of OWL norms by solving the minimization problem as follows:

$$\min_{\beta} P_\lambda(\beta) := \frac{1}{2}\|y - X\beta\|_2^2 + \sum_{i=1}^{d} \lambda_i |\beta|_{[i]}, \tag{1}$$

where $X = [x_1, x_2, \ldots, x_d] \in \mathbb{R}^{n\times d}$ is the design matrix, $y \in \mathbb{R}^d$ is the measurement vector, β is the unknown coefficient vector of the model, $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_d]$ is a non-negative regularization parameter vector of d non-increasing weights and $|\beta|_{[1]} \geq |\beta|_{[2]} \geq \ldots \geq |\beta|_{[d]}$ are the ordered coefficients in absolute value. Each feature has a corresponding regularization parameter. OWL penalty (denoted as $\Omega_\lambda(\beta)$ henceforth) penalizes the coefficients according to their magnitude: the larger the magnitude, the larger the penalty. OWL regression has been shown to outperform conventional Lasso in many applications, particularly when β is sparse and d is larger than n (Bogdan et al., 2015). (Zeng & Figueiredo, 2014; Figueiredo & Nowak, 2016) provided theoretical analysis of the sparsity and grouping properties of OWL penalty for sparse linear regression tasks with strongly correlated features.

Note that OWL regression is a general form of a set of sparse learning models. For example, Lasso is a special case of (1) if $\lambda_1 = \lambda_2 = \ldots = \lambda_d$, where $\lambda_i > 0$. L_∞-norm regression is a special case of (1) if $\lambda_1 > 0$ and $\lambda_2 = \ldots = \lambda_d = 0$. OSCAR (Bondell & Reich, 2008) is another special case of (1) if $\lambda_i = \alpha_1 + \alpha_2(d-i)$, where α_1 and α_2 are non-negative parameters.

We get the Fermat's rule of OWL regression by subdifferentials (Kruger, 2003; Mordukhovich et al., 2006) as follows:

$$X^\top(y - X\beta^*) \in \partial\Omega_\lambda(\beta^*), \tag{2}$$

where β^* is the optimum of the primal and $\partial\Omega_\lambda(\beta^*)$ is the subdifferential of $\Omega_\lambda(\beta^*)$.

From (2), we can derive the optimality conditions of OWL regression as follows:

$$-x_i^\top(y - X\beta^*) + \lambda_{r(\beta_i^*)}\text{sign}(\beta_i^*) = 0, \quad \textit{if}\ \beta_i^* \neq 0, \tag{3}$$

$$|x_i^\top(y - X\beta^*)| \leq \lambda_{r(\beta_i^*)}, \quad \textit{if}\ \beta_i^* = 0, \tag{4}$$

where $r(\beta_i^*)$ is the order of $|\beta_i^*|$ in coefficient β *w.r.t.* absolute value.

2.2. Proximal Gradient Methods

Proximal gradient methods are used as standard approaches to solve OSCAR and OWL regression. However, a major drawback is that it has slow convergence. Thus, accelerated proximal gradient methods are proposed to solve the optimization problems with the non-smooth penalty. Inspired by FISTA (Fast Iterative Shrinkage-Thresholding Algorithm) (Beck & Teboulle, 2009), (Zhong & Kwok, 2012) proposed an APGD algorithm to solve OSCAR by efficiently addressing the proximal operator. Further, (Bogdan et al., 2015) proposed an APGD algorithm to solve the general OWL regression with the proximal operator as:

$$\text{prox}(y, \lambda) := \arg\min_{x \in \mathbb{R}^d} \frac{1}{2}\|y - x\|_2^2 + \sum_{i=1}^d \lambda_i |x|_{[i]}. \tag{5}$$

Nevertheless, APGD algorithm still suffers from high computational costs and memory burden when either the size of features or samples is large. Specifically, the computation of each proximal step above takes $O(d \log d)$. The computational cost of APGD algorithm for each iteration is $O(d(n + \log d))$.

Further, as an update of each iteration in APGD algorithm depends on all the samples, each iteration of APGD algorithm can be very expensive in large-scale learning since it requires the computation of full gradients. In large-scale learning, SPGD algorithm is proposed in (Xiao & Zhang, 2014) as an effective alternative, which only requires the gradients of the samples of a mini-batch size each time.

Remark 1. *In practice, OWL regression is typically performed in the high-dimensional setting. Hence, APGD and SPGD algorithms usually suffer from high computational costs and memory burden for large feature size d. Thus, it is important and promising to speed up OWL regression by the screening technique for both APGD and SPGD algorithms.*

3. Screening Rule

In this section, we first provide the dual formulation of OWL regression and then derive the screening test based on the dual formulation. Next, we provide safe screening rules for OWL regression.

3.1. Dual of OWL Regression

In this part, we derive the dual problem of OWL regression and the screening test for OWL regression.

We consider the primal objective (1) of OWL regression, which is convex, non-smooth and non-separable. Following the derivation of l_1 regularized regression in appendix E of (Johnson & Guestrin, 2015), let $a_i = X_{i,:}^\top$ and $f_i(z_i) = \frac{1}{2}(y_i - z_i)^2$, we can derive the dual of OWL regression as follows:

$$\begin{aligned}
&\min_\beta \frac{1}{2}\|y - X\beta\|_2^2 + \sum_{i=1}^d \lambda_i |\beta|_{[i]} && \text{(6a)}\\
=\ &\min_\beta \frac{1}{2}\sum_{i=1}^n (y_i - a_i^\top\beta)^2 + \sum_{i=1}^d \lambda_i |\beta|_{[i]}\\
=\ &\min_\beta \sum_{i=1}^n f_i(a_i^\top\beta) + \sum_{i=1}^d \lambda_i |\beta|_{[i]}\\
=\ &\min_\beta \sum_{i=1}^n f_i^{**}(a_i^\top\beta) + \sum_{i=1}^d \lambda_i |\beta|_{[i]}\\
=\ &\min_\beta \sum_{i=1}^n \max_{\theta_i}[(a_i^\top\beta)\theta_i - f_i^*(\theta_i)] + \sum_{i=1}^d \lambda_i |\beta|_{[i]}\\
=\ &\min_\beta \max_\theta -\sum_{i=1}^n f_i^*(\theta_i) + \beta^\top X^\top\theta + \sum_{i=1}^d \lambda_i |\beta|_{[i]}\\
=\ &\max_\theta -\sum_{i=1}^n f_i^*(\theta_i) + \min_\beta \beta^\top X^\top\theta + \sum_{i=1}^d \lambda_i |\beta|_{[i]} && \text{(6b)}\\
=\ &\max_{\theta:|X^\top\theta| \preceq \lambda_{r(\beta)}} \sum_{i=1}^n -f_i^*(\theta_i) && \text{(6c)}\\
=\ &\max_{\theta:|X^\top\theta| \preceq \lambda_{r(\beta)}} -\frac{1}{2}\|\theta\|_2^2 - \theta^\top y, && \text{(6d)}
\end{aligned}$$

where θ is the solution of the dual and $\preceq$ means the conditions are satisfied element-wisely.

Note that f_i^* is the convex conjugate of function f_i as:

$$f_i^*(\theta_i) = \max_{z_i} \theta_i z_i - f_i(z_i). \tag{7}$$

The penultimate step to derive the dual uses the optimality condition of the following problem:

$$\min_{\beta} \beta^\top X^\top \theta + \sum_{i=1}^{d} \lambda_i |\beta|_{[i]}. \tag{8}$$

Suppose the order of β^* is known, the optimality conditions of (8) are as follows:

$$x_i^\top \theta^* + \lambda_{r(\beta_i^*)} \text{sign}(\beta_i^*) = 0, \quad \textit{if } \beta_i^* \neq 0, \tag{9}$$

$$|x_i^\top \theta^*| \leq \lambda_{r(\beta_i^*)}, \quad \textit{if } \beta_i^* = 0, \tag{10}$$

where θ^* is the optimum of the dual, which can be transformed as the constraints in (6c). Hence, we get the dual formulation of OWL regression as above.

Suppose the optimum primal and dual solutions are known, we can derive the screening condition for each variable from the optimality condition (9) and (10) as:

$$|x_i^\top \theta^*| < \lambda_{r(\beta_i^*)} \Rightarrow \beta_i^* = 0. \tag{11}$$

to identify the variables whose coefficient must be zero. Then, in the latter training process, we can train the model with less parameters and features while keeping the same output. However, the optimum in the left and right term of the screening condition in (11) are both unknown during the training process.

Hence, the aim of our screening rule is to screen as many variables whose coefficients should be zero as possible by constructing a small and safe region for the left term of the screening condition in (11) with the unknown dual optimum and exploring the unknown order structure of the primal optimum for the right term of the screening condition in (11).

3.2. Upper Bound for the Left Term

In this part, we derive a tight upper bound for $|x_i^\top \theta^*|$ in (11) by utilizing the intermediate duality gap at each iteration during the training process.

By the triangle inequality, we can derive the following bound as:

$$|x_i^\top \theta^*| \leq |x_i^\top \theta| + \|x_i\| \|\theta^* - \theta\|. \tag{12}$$

Note that the dual formulation $D(\theta)$ derived in (6d) is as follows:

$$\max_{\theta} D(\theta) := -\frac{1}{2}\|\theta\|_2^2 - \theta^\top y, \quad s.t. \quad |X^\top \theta| \preceq \lambda_{r(\beta)} \tag{13}$$

and thus the dual $D(\theta)$ is a strongly concave function. We have the following Property 1.

Property 1. *Dual $D(\theta)$ is strongly concave* w.r.t. θ. *Hence, we have*

$$D(\theta) \leq D(\theta^*) - \nabla D(\theta^*)^\top (\theta^* - \theta) - \frac{1}{2}\|\theta - \theta^*\|_2^2. \tag{14}$$

Considering Property 1, we can further bound the distance between the intermediate solution and the optimum of the dual in Corollary 1 based on the first-order optimality condition of constrained optimization.

Corollary 1. *Suppose θ and θ^* are any feasible solution and the optimum of the dual respectively, we have:*

$$\|\theta - \theta^*\| \leq \sqrt{2G(\beta, \theta)}, \tag{15}$$

where $G(\beta, \theta) = P(\beta) - D(\theta)$ is the intermediate duality gap.

Proof. By the first-order optimality condition for strongly concave dual $D(\theta)$, we have:

$$\nabla D(\theta^*)^\top (\theta^* - \theta) \geq 0. \tag{16}$$

Hence, based on (14), we have:

$$\frac{1}{2}\|\theta - \theta^*\|_2^2 \leq D(\theta^*) - D(\theta). \tag{17}$$

By strong duality that $P(\beta) \geq D(\theta^*)$, we have

$$\frac{1}{2}\|\theta - \theta^*\|_2^2 \leq P(\beta) - D(\theta), \tag{18}$$

which completes the proof. □

Hence, we can substitute $\|\theta - \theta^*\|$ in (12) by Corollary 1 based on the intermediate duality gap and then derive the screening test with the upper bound for the left term as follows:

$$|x_i^\top \theta| + \|x_i\| \sqrt{2G(\beta, \theta)} < \lambda_{r(\beta_i^*)}. \tag{19}$$

The intermediate duality gap can be computed by β and θ. β and θ can be easily obtained in the original proximal gradient algorithms.

3.3. Iterative Strategy for the Screening Rule

The screening condition (11) only works when the order of the primal optimum is known in advance, which is unknown until we finish the training process in practice. To make the screening condition applicable, we design an efficient and effective iterative strategy to explore the order of the primal optimum with the unknown order structure.

We can do screening test first as:

$$|x_i^\top \theta| + \|x_i\|\sqrt{2G(\beta,\theta)} < \lambda_d \Rightarrow \beta_i^* = 0. \tag{20}$$

According to the screening test above and for the following similarly, we can partition the variables into an safe active set $\mathcal{A}$ and an safe inactive set $\mathcal{A}'$ where the active set is the set of the variables that cannot be removed yet by our screening rule and the inactive set is the complementary set of the active set.

Suppose active set $\mathcal{A}$ has m active features at iteration k, we can assign an arbitrary permutation of $d-m$ smallest parameters $\lambda_{m+1}, \lambda_{m+2}, \ldots, \lambda_d$ to these screened coefficients without any influence to the final learned model. Thus, the order of these variables whose coefficients must be zero is known to be $d-m$ minimal absolute values of all.

Then, with m active features, by doing screening test as:

$$|x_i^\top \theta| + \|x_i\|\sqrt{2G(\beta,\theta)} < \lambda_m \Rightarrow \beta_i^* = 0, \tag{21}$$

we can find new active set $\mathcal{A}$ with m' active features where $m' \leq m$ and further derive the order of the $m-m'$ screened variables by assigning the parameters similarly as above.

At each iteration, we repeat the screening test to explore the order of primal optimum until the active set keeps unchanged. The procedure of our iterative screening rule is summarized in Algorithm 1.

The following Property 2 show our screening rule is safe to screen the variables whose coefficients should be zero with the unknown dual optimum and the unknown order structure of the primal optimum.

Property 2. *The iterative screening rule we proposed is guaranteed to be safe for Algorithm 1 and the whole training process of OWL regression.*

Proof. First, we prove our screening rule is safe for Algorithm 1. At the first iteration of Algorithm 1, active set $\mathcal{A}$ has total d active features. We do screening test (20). Since $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_d]$ is a non-increasing vector, we have $\lambda_d \leq \lambda_{r(\beta_i^*)}$. Hence, the screening test above can make sure $|x_i^\top \theta| + \|x_i\|\sqrt{2G(\beta,\theta)} < \lambda_{r(\beta_i^*)}$. Thus, our screening test is safe at the first iteration.

Suppose our screening test is safe for the first k iterations and active set $\mathcal{A}$ has m active features at iteration k, the parameters of the $d-m$ screened variables whose coefficients should be zero at the optimum are assigned as a permutation of $[\lambda_{m+1}, \lambda_{m+2}, \ldots, \lambda_d]$. Then, the new regularization parameter vector for the variables that has not been screened is a permutation of $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_m]$.

Thus, we can do the screening test for the left active variables as (21) to make sure $|x_i^\top \theta| + \|x_i\|\sqrt{2G(\beta,\theta)} < \lambda_{r(\beta_i^*)}$, which shows the screening test is safe at iteration $k+1$. Thus, our screening rule is proved to be safe for Algorithm 1.

For the latter sub-problem with less parameters and features to be solved in the iterative optimization algorithm, the way to do the screening test is similar to the original problem. Thus, following the proof above, we can easily prove that our screening rule is safe for the latter sub-problem and further for the whole training process of OWL regression, which completes the proof for Property 2.

□

Algorithm 1 Safe Screening Rule for OWL Regression with Iterative Strategy

Input: $\mathcal{A}, \beta_k, \theta_k, G(\beta_k, \theta_k)$.
1: **while** $\mathcal{A}$ still changes **do**
2: Do the screening test based on (21).
3: Update $\mathcal{A}$.
4: **end while**
Output: New active set $\mathcal{A}$.

4. Screening Rule in the Proximal Gradient Algorithms

In this section, we apply the screening rule to the APGD and SPGD algorithm in the batch and stochastic setting respectively for OWL regression.

4.1. Proposed Algorithms

In the batch setting, we compute the dual solution and duality gap first. Then, we compute the active set by Algorithm 1 and update the solution as the original APGD algorithm with the obtained active variables. If active set $\mathcal{A}$ is updated in the current iteration, we also update the step size. As the iteration increases, the solution is closer to the optimum and thus the duality gap also becomes smaller. Correspondingly, more inactive variables are screened by our screening rule. We present the procedures of our algorithm for the batch setting in Algorithm 2.

Similarly, in the stochastic setting, we compute the dual solution and duality gap in the main loop first. After that, we derive the active set by Algorithm 1 and update the solution as the original SPGD algorithm with the obtained active variables. Let $F(\beta) := \frac{1}{2}\|y - X\beta\|_2^2$, we present the procedures of our algorithm for the stochastic setting in Algorithm 3.

Interestingly, the duality gap, which is the main time-consuming step of our screening rule in Algorithm 1, has been computed by the original APGD and SPGD algorithms. Moreover, suppose the size of the active set for iteration k

Algorithm 2 Accelerated Proximal Gradient Descent Algorithm with Safe Screening Rules

Input: $\beta^0, b^1 = \beta^0, t_1 = 1$.
for $k = 1, 2, \ldots$ **do**
  Compute dual θ and duality gap.
  Update $\mathcal{A}$ based on Algorithm 1.
  if $\mathcal{A}$ changes **then**
    $t_k = t_1$.
  end if
  $\beta^k = \text{prox}_{t_k,\lambda}(b^k - t_k X^\top (Xb^k - y))$.
  $t_{k+1} = \frac{1}{2}(1 + \sqrt{1 + 4t_k^2})$.
  $b^{k+1} = \beta^k + \frac{t_k - 1}{t_{k+1}}(\beta^k - \beta^{k-1})$.
end for
Output: Coefficient β.

Algorithm 3 Stochastic Proximal Gradient Descent Algorithm with Safe Screening Rules

Input: β^0, l.
for $k = 1, 2, \ldots$ **do**
  Compute dual θ and duality gap.
  Update $\mathcal{A}$ based on Algorithm 1.
  $\beta = \beta^{k-1}$.
  $\tilde{v} = \nabla F(\beta)$.
  $\tilde{\beta}^0 = \beta$.
  for $t = 1, 2, \ldots, T$ **do**
    Pick mini-batch $I_t \subseteq X$ of size l.
    $v_t = (\nabla F_{I_t}(\tilde{\beta}^{t-1}) - \nabla F_{I_t}(\beta))/l + \tilde{v}$.
    $\tilde{\beta}^t = \text{prox}_{\eta,\lambda}(\tilde{\beta}^{t-1} - \eta v_t)$.
  end for
  $\beta^k = \tilde{\beta}^T$
end for
Output: Coefficient β.

is d_k, the computation complexity of the screening rule for each iteration is only $O(d_k)$, which is even cheaper than the complexity of the original stopping criterion evaluation $O(d)$ and thus can be skipped for the analysis with the complexity $O(d_k(n + \log d_k))$ or $O(d_k(n + Tl + T \log d_k))$ for each iteration in the batch and stochastic setting respectively.

More importantly, for iteration k with d_k active variables, our Algorithm 2 only requires $O(d_k(n + \log d_k))$, which is much smaller than the complexity $O(d(n + \log d))$ required by the original APGD algorithm. Similarly, our Algorithm 3 only requires $O(d_k(n + Tl + T \log d_k))$ for main loop k where T is number of the inner loop and l is the size of mini-batch, which is much smaller than $O(d(n + Tl + T \log d))$ required by the original SPGD algorithm. Hence, in high-dimensional sparse learning, the computation costs of both APGD and SPGD algorithms are effectively reduced by our screening rule.

4.2. Theoretical Analysis

In this part, we give the properties of convergence and screening ability when our screening rule is applied to standard iterative optimization algorithms.

In terms of the convergence, our algorithms have the following Property 3.

Property 3. *Suppose iterative algorithm Ψ to solve OWL regression converges to the optimum, algorithm Ψ with our screening rule to solve OWL regression also converges to the optimum.*

Proof. We denote the sub-problem at iteration k as P_k. First, we know Ψ converges to the optimum for P_1. Then, suppose algorithm Ψ with the screening rule converges to the optimum for P_k. Considering iteration $k + 1$, P_{k+1} is a sub-problem of P_k. Thus, the convergence of P_{k+1} can be guaranteed as P_k, which completes the proof. □

Property 3 shows the convergence of standard iterative optimization algorithms with our screening rules can be guaranteed by the original algorithms. Thus, our screening rule can be combined with existing iterative optimization algorithms, *e.g.*, APGD, SPGD and *et al.*.

In terms of the screening ability, our algorithms have the following Property 4 and 5.

Property 4. *θ converges to θ^* of the dual if β converges to β^* of the primal.*

Proof. Considering the maximization part of (6b) as follows:

$$\max_{\theta} -\frac{1}{2}\|\theta\|_2^2 - \theta^\top (y - X\beta), \tag{22}$$

we can get the primal-dual link equation as:

$$\theta^* = X\beta^* - y. \tag{23}$$

Thus, as β converges to β^* of the primal, θ converges to θ^* of the dual. □

Property 4 shows the convergence of the dual can be guaranteed by the convergence of the primal, which means the intermediate duality gap becomes smaller as the iteration increases and thus our screening rule is promising to screen more inactive variables.

Further, we give Property 5 to show the excellent screening ability of our screening rule.

Property 5. *Based on the optimality conditions, we have that final active set $\mathcal{A}^*$ satisfies that $\min_{i \in A^*} |x_i^\top \theta^*| = \lambda_{|\mathcal{A}^*|}$ where $|\mathcal{A}^*|$ is the size of $\mathcal{A}^*$. Then, as algorithm Ψ converges, there exists an iteration number $K_0 \in \mathbb{N}$ s.t.*

$\forall k \geq K_0$, any variable $j \notin \mathcal{A}^*$ is screened by our screening rule.

Proof. As Ψ converges, owing to the strong duality, the intermediate duality gap converges towards zero. Thus, for any given ϵ, there exists K_0 such that $\forall k \geq K_0$, we have

$$\|\theta^k - \theta^*\|_2 \leq \epsilon, \tag{24}$$

and

$$\sqrt{2G(\beta^k, \theta^k)} \leq \epsilon. \tag{25}$$

For any $j \notin \mathcal{A}^*$, we have

$$\begin{aligned} & |x_j^\top \theta^k| + \|x_j\| \sqrt{2G(\beta^k, \theta^k)} \qquad (26) \\ \leq \; & |x_j^\top (\theta^k - \theta^*)| + |x_j^\top \theta^*| + \|x_j\| \sqrt{2G(\beta^k, \theta^k)} \\ \leq \; & 2\|x_j\|\epsilon + |x_j^\top \theta^*| \end{aligned}$$

The first inequality is obtained by the triangle inequality and the second inequality is obtained by (24) and (25). Thus, if we choose

$$\epsilon < \frac{\lambda_{|\mathcal{A}^*|} - |x_j^\top \theta^*|}{2\|x_j\|} \tag{27}$$

where $\lambda_{|\mathcal{A}^*|} - |x_j^\top \theta^*| > 0$ is easily obtained since $j \notin \mathcal{A}^*$, we have $|x_j^\top \theta^k| + \|x_j\| \sqrt{2G(\beta^k, \theta^k)} < \lambda_{|\mathcal{A}^*|}$, which is the screening rule we proposed. That is to say, variable j is screened out by our screening rule at this iteration, which completes the proof. □

Property 5 shows all the inactive variables $j \notin \mathcal{A}^*$ are correctly detected and effectively screened by our screening rule in a finite number of iterations.

5. Experiments

In this section, we first give the experimental setup and then present our experimental results with discussions.

5.1. Experimental Setup

5.1.1. Design of Experiments

We conduct experiments on six real-world benchmark datasets not only to verify the effectiveness of our algorithm on reducing running time, but also to show the effectiveness and safety on screening inactive variables.

To validate the effectiveness of our algorithms on reducing running time, we evaluate the running time of our algorithms and other competitive algorithms to solve OWL regression under different settings. To confirm the effectiveness and safety of our algorithms on screening inactive variables, we evaluate the screening rate at each iteration of our algorithm and the prediction errors of different algorithms. The compared algorithms are summarized as follows:

Table 2. The real-world datasets used in the experiments.

Dataset	Sample size	Attributes
Duke Breast Cancer	44	7129
Colon Cancer	62	2000
Cardiac Left	3360	1600
Cardiac Right	3360	1600
IndoorLoc Longitude	21048	529
Slice Localization	53500	386

- APGD: Accelerated proximal gradient descent algorithm (Bogdan et al., 2015).
- APGD + Screening: Accelerated proximal gradient descent algorithm with the safe screening rule.
- SPGD: Stochastic proximal gradient descent algorithm with variance reduction we adopt in (Xiao & Zhang, 2014).
- SPGD + Screening: Stochastic proximal gradient descent algorithm with variance reduction and the safe screening rule.

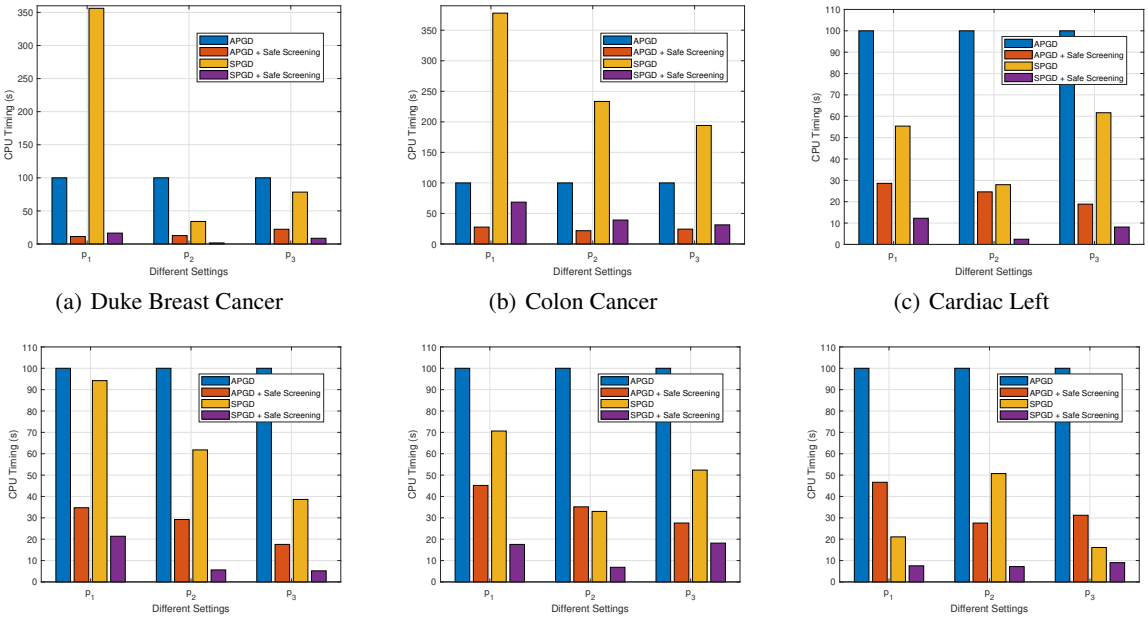
5.1.2. Implementation Details

Our experiments were performed on a 4-core Intel i7-6820 machine. We implement all the algorithms in MATLAB and compare the average running CPU time of different algorithms at the same platform for 5 trials. For the comparison convenience, the CPU time of each algorithm is shown as the percentage of APGD under each setting. Following the setting in (Bogdan et al., 2015), tolerance error ϵ of duality gap and dual infeasibility in our experiments are set as 10^{-6}. At the very early stage, the solution is far from the optimum and thus the screening rule can only screen a small portion of variables. We run our algorithms with a warm start. Please note all the experimental setup in Algorithm 2 and 3 follows the original APGD and SPGD algorithms with the same hyperparameters of the size of mini-batch, the number of inner loop and step size η, which range from 5 to 100, 5 to 80 and 10^{-6} to 10^{-3} respectively for different datasets, are selected by grid search.

We use the popular OSCAR setting (also called OWL regression with linear decay), which is widely used in (Oswal et al., 2016; Zhong & Kwok, 2012; Zhang et al., 2018), as

Table 3. Prediction errors of different algorithms.

DATASET	APGD	APGD + SCREENING	SPGD	SPGD + SCREENING
DUKE BREAST CANCER	0.6523	**0.6523**	0.6523	**0.6523**
COLON CANCER	0.9453	**0.9453**	0.9453	**0.9453**
CARDIAC LEFT	0.9453	**0.9453**	0.9453	**0.9453**
CARDIAC RIGHT	0.5276	**0.5276**	0.5276	**0.5276**
INDOORLOC LONGITUDE	0.5531	**0.5531**	0.5531	**0.5531**
SLICE LOCALIZATION	0.6162	**0.6162**	0.6162	**0.6162**



(a) Duke Breast Cancer (b) Colon Cancer (c) Cardiac Left

(d) Cardiac Right (e) IndoorLoc Longitude (f) Slice Localization

Figure 1. Average running time of different algorithms without and with safe screening rules under different settings.

follows:

$$\lambda_i = \alpha_1 + \alpha_2(d - i), \tag{28}$$

where $\alpha_1 = p_i \|X^\top y\|_\infty$ and $\alpha_2 = \alpha_1/d$. For a fair comparison, the factor p_i is used to control the sparsity. In our experiments, we set $p_i = i * e^{-\tau}$, $i = 1, 2, 3$, $\tau = 2$ for Duke Breast Cancer, IndoorLoc Longitude and Slice Localization datasets and $\tau = 3$ for Colon Cancer, Cardiac Left and Cardiac Right datasets.

To evaluate the screening rate of our algorithms, the screening rate is defined as the percentage of the inactive variables we screened to the total inactive ones. We set the sparsity as p_1 here and for the following part.

To compare the prediction error of different algorithms, we randomly divide the dataset into the training and testing set in proportion to $4:1$ and use root mean squared error (RMSE) as the performance criterion of the linear regression tasks.

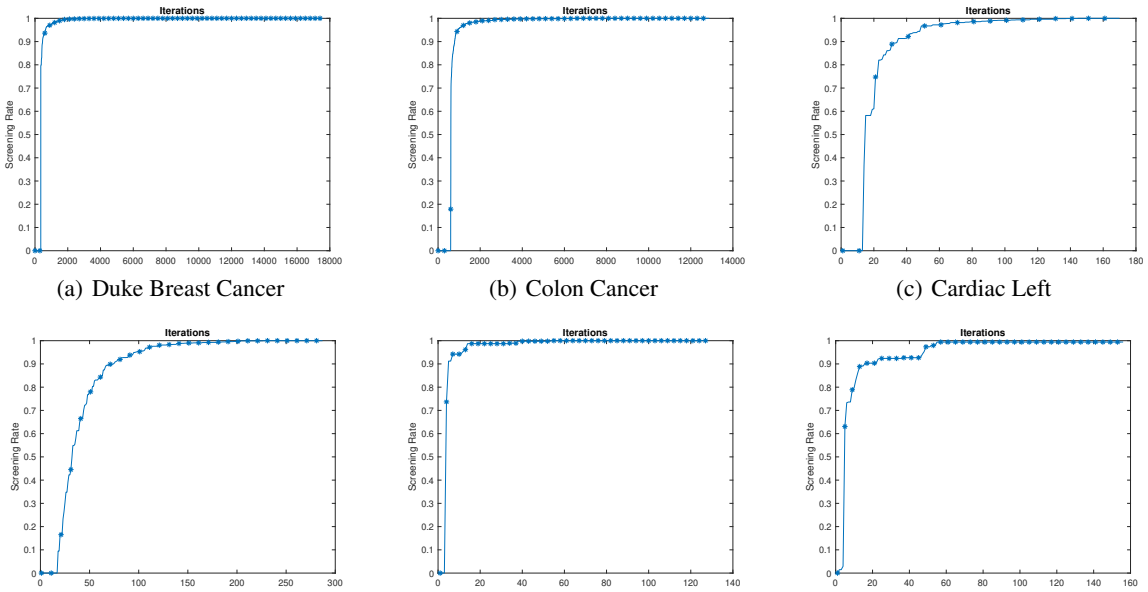
5.1.3. DATASETS

Table 2 summarizes six benchmark datasets used in our experiments. Duke Breast Cancer and Colon Cancer datasets are from the LIBSVM repository, which is available at https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/. IndoorLoc Longitude and Slice Localization datasets are from the UCI benchmark repository (Dua & Graff, 2017), which is available at https://archive.ics.uci.edu/ml/datasets.php. Cardiac Left and Cardiac Right datasets are collected from 3360 MRI images by hospitals (Gu et al., 2014).

5.2. Experimental Results and Discussions

5.2.1. RUNNING TIME

Figures 1(a)-(f) provide the results of the average running time of four algorithms on the six datasets for the OWL regularized regression tasks in different situations. The results confirm that the methods with our screening rule are always much faster than the original ones both in the batch and



(a) Duke Breast Cancer (b) Colon Cancer (c) Cardiac Left

(d) Cardiac Right (e) IndoorLoc Longitude (f) Slice Localization

Figure 2. The screening rate of different datasets in the stochastic setting.

stochastic settings. This is because our screening rule could screen a large portion of inactive variables during the training process. Thus, the algorithms with our screening rule reduce much computational cost of the original algorithms.

When $n \ll d$, the results show, with our safe screening rule, APGD algorithm achieves the computational gain to the original algorithm by a factor of 4x to 8x and SPGD algorithm achieves the computational gain to the original one by 5x to 22x. For large-scale learning where $n \approx d$ or $n \gg d$, the results show SPGD algorithm with our safe screening rule always achieve the largest computational gain, which can accelerate the original APGD algorithm by 4x to 40x. This is because the stochastic methods can reduce computational burden in large-scale learning. Interestingly, with our screening rule, stochastic methods could achieve significant computational gain even when $n \approx d$. This is because the problem degenerates into a sub-problem that $n \gg d$ by screening inactive variables during the training process. Also note we benefit from the screening rule more with larger and sparser datasets.

5.2.2. Screening Rate

Figures 2(a)-(f) present the results of the screening rate of our algorithms on six datasets in the stochastic setting to show the screening ability and characteristics of our screening rule. The results support the conclusion that our algorithm can successfully screen most of the inactive variables at the very early stage, reach the final active set and screen almost all the inactive variables in a finite number of iterations and thus is an effective method to screen inactive variables of OWL regression. This is because the upper bound of our screening test is very tight and the iterative strategy is effective to explore the order structure of primal solution to screen more inactive variables during the training process.

5.2.3. Prediction Error

Table 3 provides the results of prediction errors of four algorithms on six datasets for OWL regularized regression to confirm the safety of our screening rule. According to the experimental results, the prediction errors of our algorithms are identical with the original algorithms. The reason is that our screening rule is guaranteed to be safe and thus our algorithms with our screening rule are guaranteed to yield the exactly same solution as the original ones.

6. Conclusion

In this paper, we propose the first safe screening rule for OWL regression by effectively tackling the non-separable penalty, which allows to avoid the useless computation of the parameters whose coefficients must be zero. Moreover, the proposed screening rule can be easily applied to existing iterative optimization algorithms. Theoretically, we prove that the algorithms with our screening rule is able to guarantee identical results with the original algorithms. Extensive experiments on six benchmark datasets verify that the screening rule leads to significant computation gain without any loss of accuracy by screening inactive variables.

Acknowledgements

This work was partially supported by U.S. NSF IIS 1836945, IIS 1836938, IIS 1845666, IIS 1852606, IIS 1838627, IIS 1837956.

References

Bao, R., Gu, B., and Huang, H. Efficient approximate solution path algorithm for order weight l_1-norm with accuracy guarantee. In *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 958–963. IEEE, 2019.

Barber, R. F., Candès, E. J., et al. Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43 (5):2055–2085, 2015.

Beck, A. and Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Bellec, P. C., Lecué, G., Tsybakov, A. B., et al. Slope meets lasso: improved oracle bounds and optimality. *The Annals of Statistics*, 46(6B):3603–3642, 2018.

Bogdan, M., Berg, E. v. d., Su, W., and Candes, E. Statistical estimation and testing via the sorted l1 norm. *arXiv preprint arXiv:1310.1969*, 2013.

Bogdan, M., Van Den Berg, E., Sabatti, C., Su, W., and Candès, E. J. Slope—adaptive variable selection via convex optimization. *The annals of applied statistics*, 9 (3):667–698, 2015.

Bondell, H. D. and Reich, B. J. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. *Biometrics*, 64(1):115–123, 2008.

Brzyski, D., Gossmann, A., Su, W., and Bogdan, M. Group slope–adaptive selection of groups of predictors. *Journal of the American Statistical Association*, 114(525):419–433, 2019.

Bu, Z., Klusowski, J., Rush, C., and Su, W. Algorithmic analysis and statistical estimation of slope via approximate message passing. In *Advances in Neural Information Processing Systems*, pp. 9361–9371, 2019.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Fercoq, O., Gramfort, A., and Salmon, J. Mind the duality gap: safer rules for the lasso. In *International Conference on Machine Learning*, pp. 333–342, 2015.

Figueiredo, M. and Nowak, R. Ordered weighted l1 regularized regression with strongly correlated covariates: Theoretical aspects. In *Artificial Intelligence and Statistics*, pp. 930–938, 2016.

Gu, B., Sheng, V. S., Tay, K. Y., Romano, W., and Li, S. Incremental support vector learning for ordinal regression. *IEEE Transactions on Neural networks and learning systems*, 26(7):1403–1416, 2014.

Johnson, T. and Guestrin, C. Blitz: A principled meta-algorithm for scaling sparse optimization. In *International Conference on Machine Learning*, pp. 1171–1179, 2015.

Kruger, A. Y. On fréchet subdifferentials. *Journal of Mathematical Sciences*, 116(3):3325–3358, 2003.

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

Liu, J., Zhao, Z., Wang, J., and Ye, J. Safe screening with variational inequalities and its application to lasso. In *International Conference on Machine Learning*, pp. 289–297, 2014.

Mordukhovich, B. S., Nam, N. M., and Yen, N. Fréchet subdifferential calculus and optimality conditions in nondifferentiable programming. *Optimization*, 55(5-6):685–708, 2006.

Ndiaye, E., Fercoq, O., Gramfort, A., and Salmon, J. Gap safe screening rules for sparse-group lasso. In *Advances in Neural Information Processing Systems*, pp. 388–396, 2016.

Oswal, U., Cox, C., Lambon-Ralph, M., Rogers, T., and Nowak, R. Representational similarity learning with application to brain networks. In *International Conference on Machine Learning*, pp. 1041–1049, 2016.

Rakotomamonjy, A., Gasso, G., and Salmon, J. Screening rules for lasso with non-convex sparse regularizers. In *International Conference on Machine Learning*, pp. 5341–5350, 2019.

Shibagaki, A., Karasuyama, M., Hatano, K., and Takeuchi, I. Simultaneous safe screening of features and samples in doubly sparse modeling. In *International Conference on Machine Learning*, pp. 1577–1586, 2016.

Su, W., Candes, E., et al. Slope is adaptive to unknown sparsity and asymptotically minimax. *The Annals of Statistics*, 44(3):1038–1068, 2016.

Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Tibshirani, R., Bien, J., Friedman, J., Hastie, T., Simon, N., Taylor, J., and Tibshirani, R. 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, J., Zhou, J., Wonka, P., and Ye, J. Lasso screening rules via dual polytope projection. In *Advances in neural information processing systems*, pp. 1070–1078, 2013.

Xiang, Z. J., Wang, Y., and Ramadge, P. J. Screening tests for lasso problems. *IEEE transactions on pattern analysis and machine intelligence*, 39(5):1008–1027, 2016.

Xiao, L. and Zhang, T. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

Yuan, M. and Lin, Y. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68 (1):49–67, 2006.

Zeng, X. and Figueiredo, M. A. Decreasing weighted sorted l_1 regularization. *IEEE Signal Processing Letters*, 21(10): 1240–1244, 2014.

Zhai, Z., Gu, B., Li, X., and Huang, H. Safe sample screening for robust support vector machine. *arXiv preprint arXiv:1912.11217*, 2019.

Zhang, D., Wang, H., Figueiredo, M., and Balzano, L. Learning to share: Simultaneous parameter tying and sparsification in deep learning. In *International Conference on Learning Representations*, 2018.

Zhang, W., Hong, B., Liu, W., Ye, J., Cai, D., He, X., and Wang, J. Scaling up sparse support vector machines by simultaneous feature and sample reduction. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 4016–4025. JMLR. org, 2017.

Zhong, L. W. and Kwok, J. T. Efficient sparse modeling with automatic feature grouping. *IEEE transactions on neural networks and learning systems*, 23(9):1436–1447, 2012.