Safe Feature Elimination for the LASSO and Sparse Supervised Learning Problems

Laurent El Ghaoui ELGHAOUI@EECS.BERKELEY.EDU
Vivian Viallon VIALLON@EECS.BERKELEY.EDU
Tarek Rabbani TRABBANI@BERKELEY.EDU
Department of EECS
University of California
Berkeley, CA 94720-1776, USA

Editor:

Abstract

We describe a fast method to eliminate features (variables) in l_1-penalized least-square regression (or LASSO) problems. The elimination of features leads to a potentially substantial reduction in running time, especially for large values of the penalty parameter. Our method is not heuristic: it only eliminates features that are guaranteed to be absent after solving the LASSO problem. The feature elimination step is easy to parallelize and can test each feature for elimination independently. Moreover, the computational effort of our method is negligible compared to that of solving the LASSO problem - roughly it is the same as single gradient step. Our method extends the scope of existing LASSO algorithms to treat larger data sets, previously out of their reach. We show how our method can be extended to general l_1-penalized convex problems and present preliminary results for the Sparse Support Vector Machine and Logistic Regression problems.

Keywords: Sparse Regression, LASSO, Feature Elimination, SVM, Logistic Regression

1. Introduction

"Sparse" classification or regression problems, which involve an ℓ_1-norm regularization has attracted a lot of interest in the statistics (Tibshirani, 1996), signal processing (Chen et al., 2001), and machine learning communities. The ℓ_1 regularization leads to sparse solutions, which is a desirable property to achieve model selection, or data compression. For instance, consider the problem of ℓ_1-regularized least square regression commonly referred to as the LASSO (Tibshirani, 1996). In this context, we are given a set of m observations $a_i \in \mathbb{R}^n$, $i = 1, \ldots, m$ and a response vector $y \in \mathbb{R}^m$. Denoting by $X = (a_1, \ldots, a_m)^T \in \mathbb{R}^{m \times n}$ the feature matrix of observations, the LASSO problem is given by

$$\mathcal{P}(\lambda) \ : \ \phi(\lambda) := \min_w \frac{1}{2} \|Xw - y\|_2^2 + \lambda \|w\|_1 , \tag{1}$$

where λ is a regularization parameter and $w \in \mathbb{R}^n$ is the optimization variable. For large enough values of λ, any solution $w^\star \in \mathbb{R}^n$ of (1) is typically sparse, i.e. $w^\star$ has few entries that are non-zero, and therefore identifies the features in X (columns of X) that are useful to predict y.

Several efficient algorithms have been developed for the LASSO problem, including Efron et al. (2004); Kim et al. (2007); Park and Hastie (2007); Donoho and Tsaig (2008); Friedman et al. (2007); Becker et al. (2010); Friedman et al. (2010) and references therein. However, the complexity of these algorithms, when it is known, grows fast with the number of variables. While the LASSO problem is particularly appealing in presence of very high-dimensional problems, the available algorithms can be quite slow in such contexts. In some applications, the feature matrix is so big that it can not even be loaded and LASSO solvers cannot be used at all. Hence it is of paramount interest to be able to efficiently eliminate features in a pre-processing step, in order to reduce dimensionality and solve the optimization problem on a reduced matrix.

Assume that a sparse solution exists to (1) and that we were able to identify e zeros of $w^\star$ **a priori** to solving the LASSO problem. Identifying e zeros in $w^\star$ a priori to solving (1) is equivalent to removing e features (columns) from the feature matrix X. If e is large, we can obtain $w^\star$ by solving (1) with a "small" feature matrix X.

In this paper we propose a "safe" feature elimination (SAFE) method that can identify zeros in the solution $w^\star$ a priori to solving the LASSO problem. Once the zeros are identified we can safely remove the corresponding features and then solve the LASSO problem (1) on the reduced feature matrix.

Feature selection methods are often used to accomplish dimensionality reduction, and are of utmost relevance for data sets of massive dimension, see for example Fan and Lv (2010). These methods, when used as a pre-processing step, have been referred to in the literature as *screening* procedures (Fan and Lv, 2010, 2008). They typically rely on univariate models to score features, independently of each other, and are usually computationally fast. Classical procedures are based on correlation coefficients, two-sample t-statistics or chi-square statistics (Fan and Lv, 2010); see also Forman (2003) and the references therein for an overview in the specific case of text classification. Most screening methods might remove features that could otherwise have been selected by the regression or classification algorithm. However, some of them were recently shown to enjoy the so-called "sure screening" property (Fan and Lv, 2008): under some technical conditions, no relevant feature is removed, with probability tending to one.

Screening procedures typically ignore the specific classification task to be solved after feature elimination. In this paper, we propose to remove features based on the supervised learning problem considered, that is on both the structure of the loss function and the problem data. While we focus mainly on the LASSO problem here, we provide results for a large class of convex classification or regression problems. The features are eliminated according to a sufficient, in general conservative, condition, which we call SAFE (for SAfe Feature Elimination). With SAFE, we never remove features unless they are *guaranteed* to be absent if one were to solve the full-fledged classification or regression problem.

An interesting fact is that SAFE becomes extremely aggressive at removing features for large values of the penalty parameter λ. The specific application we have in mind involves large data sets of text documents, and sparse matrices based on occurrence, or other score, of words or terms in these documents. We seek extremely sparse optimal coefficient vectors, even if that means operating at values of the penalty parameter that are substantially larger than those dictated by a pure concern for predictive accuracy. The fact that we need to operate at high values of this parameter opens the hope that, at least for the application considered, the number of features eliminated by using our fast test is high enough to allow a dramatic reduction in computing time and memory requirements. Our experimental results indicate that for many of these data sets, we do observe a dramatic reduction in the number of variables, typically by an order of magnitude or more. The method has two main advantages: for medium- to large-sized problem, it enables to reduce the computational time. More importantly, SAFE allows to tackle problems that are too huge to be even loaded in memory, thereby expanding the reach of current algorithms

The paper is organized as follows. In section 2, we derive the SAFE method for the LASSO problem. In section 3, we illustrate the use of SAFE and detail some relevant algorithms. In section 4, we extend the results of SAFE to general convex problems and derive preliminary SAFE results for the Sparse Support Vector Machine and Logistic regression problems. In section 5, we experiment the SAFE for LASSO method on synthetic data and on data derived from text classification sources. Numerical results demonstrate that SAFE provides a substantial reduction in problem size, and, as a result, it enables the LASSO algorithms to run faster and solve huge problems originally out of their reach.

Notation. We use $\mathbf{1}$ and $\mathbf{0}$ to denote a vector of ones and zeros, with size inferred from context, respectively. For a scalar a, a_+ denotes the positive part of a. For a vector a, this operation is component-wise, so that $\mathbf{1}^T a_+$ is the sum of the positive elements in a. We take the convention that a sum over an empty index sets, such as $\sum_{i=1}^k a_i$ with $k \leq 0$, is zero.

2. The SAFE method for the LASSO

The SAFE method crucially relies on duality and optimality conditions. We begin by reviewing the appropriate facts.

2.1 Dual problem and optimality conditions for the LASSO

A dual to the LASSO problem (1) (Kim et al., 2007) can be written as

$$\mathcal{D}(\lambda) \ : \ \phi(\lambda) := \max_{\theta} G(\theta) \ : \ \left|\theta^T x_k\right| \leq \lambda, \, k = 1, \ldots, n, \tag{2}$$

with $x_k \in \mathbb{R}^m$, $k = 1, \ldots, n$, the k-th column of X and $G(\theta) = \frac{1}{2} \|y\|_2^2 - \frac{1}{2} \|\theta + y\|_2^2$. In this context, we call $\mathcal{P}(\lambda)$ the primal problem, w the primal variable, and $w^\star$ a primal optimal point. The dual problem $\mathcal{D}(\lambda)$ is a convex optimization problem with dual variable $\theta \in \mathbb{R}^m$. We call θ dual feasible when it satisfies the constraints in $\mathcal{D}(\lambda)$. Figure 1(a) shows the geometry of the feasibility set in the dual space. The quantity $G(\theta)$ gives a lower bound on the optimal value $\phi(\lambda)$ for any dual feasible point θ, i.e. $G(\theta) \leq \phi(\lambda)$, $\left|\theta^T x_k\right| \leq \lambda$, $k = 1, \ldots, n$. For the LASSO problem (1) strong duality holds and the optimal value of $\mathcal{D}(\lambda)$ achieves $\phi(\lambda)$ at $\theta^\star$ the solution of (2) or the dual optimal point. Furthermore, the following relation holds at optimum: $\theta^\star = Xw^\star - y$.

We consider the dual problem $\mathcal{D}(\lambda)$ because of an important property that helps us derive our SAFE method. Assuming $w^\star$ is sparse, knowledge of $\theta^\star$ allows us to identify the zeros in $w^\star$ by checking the optimality condition (Boyd and Vandenberghe, 2004):

$$\left|\theta^{\star T} x_k\right| < \lambda \Rightarrow (w^\star)_k = 0. \tag{3}$$

Figure 1(b) illustrates the geometric interpretation of the inequality test $\left|\theta^{\star T} x_k\right| < \lambda$ in (3).

2.2 Basic idea

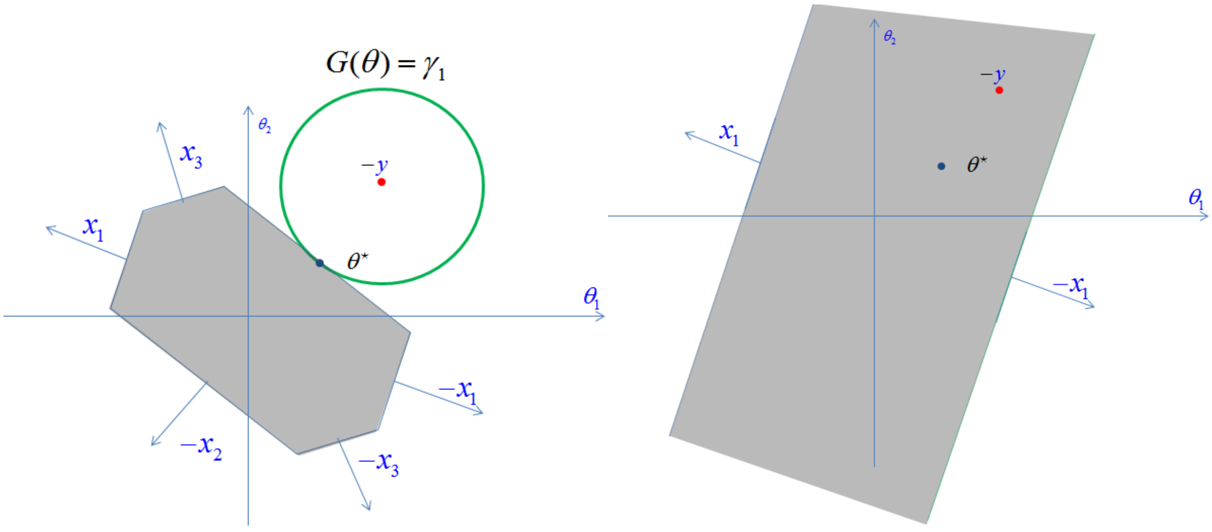
The basic idea behind SAFE is to use the optimality condition (3) with $\theta^\star$ in the inequality test replaced by a set Θ that contains the dual optimal point, i.e. $\left|\theta^T x_k\right| < \lambda$, $\forall \theta \in \Theta$ and $\theta^\star \in \Theta$. If the inequality test holds for the whole set Θ, then the k-th entry of $w^\star$ is zero, $(w^\star)_k = 0$.

In the following sections, we show how to construct the set Θ using optimality conditions of the dual problem, and derive the corresponding SAFE test.

In our derivation, we assume that we have knowledge of a solution $w_0^\star$ of $\mathcal{P}(\lambda_0)$ for some λ_0, and we seek to apply SAFE for $\mathcal{P}(\lambda)$ with $\lambda \leq \lambda_0$. By default, we can choose λ_0 to be large enough for $w_0^\star$ to be identically zero. To find such a λ_0, we substitute $w_0^\star = 0$ in (1) to obtain $\phi(\lambda_0) = \frac{1}{2} \|y\|_2^2$. By strong duality, $\mathcal{D}(\lambda_0)$ achieves a value of $\phi(\lambda_0) = \frac{1}{2} \|y\|_2^2$ at the unique solution $\theta_0^\star = -y$. The point $\theta_0^\star$ is a dual feasible point and satisfies the constraints $\lambda_0 \geq \left|(-y)^T x_k\right|$, $k = 1, \ldots, n$. Note that λ_0 is not uniquely defined but we choose the smallest value above which $w_0^\star = 0$, that is $\lambda_0 = \max_{1 \leq j \leq n} |y^T x_j| = \|X^T y\|_\infty$.

2.3 Constructing Θ

We start by finding a set Θ that contains the dual optimal point $\theta^\star$ of $\mathcal{D}(\lambda)$. We express Θ as the intersection of two sets Θ_1 and Θ_2, where each set corresponds to different optimality conditions.



(a)

(b)

Figure 1: Geometry of the dual problem $\mathcal{D}(\lambda)$. (a) Feasibility set of the dual problem. The grey shaded polytope shows the feasibility set of $\mathcal{D}(\lambda)$. The feasibility set is the intersection of n slabs in the dual space corresponding to the n features x_k, $k = 1, \ldots, n$. The level set $G(\theta) = \gamma_1$, where $\gamma_1 = G(\theta^\star)$, corresponds to the optimal value of the dual function and is tangent to the feasibility set at the dual optimal point $\theta^\star$. (b) Geometry of the inequality test in (3). The grey shaded region is the slab corresponding to feature x_k, i.e. $\left\{\theta \mid \left|\theta^T x_k\right| \leq \lambda\right\}$. The test $\left|\theta^{\star T} x_k\right| < \lambda$ is a strict inequality when the point $\theta^\star$ is in the interior of the slab defined by the feature x_k. Thus if the dual optimal point is inside a slab defined by feature x_k, by optimality condition (3) the k-th entry of the primal optimal solution $w^\star$ is zero, i.e. $(w^\star)_k = 0$.

We construct Θ_1 using the optimality condition of $\mathcal{D}(\lambda)$: $\theta^\star$ is a dual optimal point if $G(\theta^\star) \geq G(\theta)$ for all dual feasible points θ. Let θ_s be a dual feasible point to $\mathcal{D}(\lambda)$, and $\gamma := G(\theta_s)$. Obviously $G(\theta^\star) \geq \gamma$ and the set $\Theta_1 := \{\theta \mid G(\theta) \geq \gamma\}$ contains $\theta^\star$, i.e. $\theta^\star \in \Theta_1$.

One way to obtain a lower bound γ is by dual scaling. We set θ_s to be a scaled feasible dual point in terms of $\theta_0^\star$, $\theta_s := s\theta_0^\star$ with $s \in \mathbb{R}$ constrained so that θ_s is a dual feasible point for $\mathcal{D}(\lambda)$, that is, $\|X^T\theta_s\|_\infty \leq \lambda$ or $|s| \leq \lambda/\lambda_0$. We then set γ according to the convex optimization problem:

$$\gamma = \max_s \left\{ G(s\theta_0^\star) \ : \ |s| \leq \frac{\lambda}{\lambda_0} \right\} = \max_s \left\{ \beta_0 s - \frac{1}{2} s^2 \alpha_0 \ : \ |s| \leq \frac{\lambda}{\lambda_0} \right\},$$

with $\alpha_0 := \theta_0^{\star T}\theta_0^\star > 0$, $\beta_0 := |y^T\theta_0^\star|$. We obtain

$$\gamma = \frac{\beta_0^2}{2\alpha_0} \left(1 - \left(1 - \frac{\alpha_0}{\beta_0} \frac{\lambda}{\lambda_0} \right)_+^2 \right). \tag{4}$$

We construct Θ_2 by applying a first order optimality condition on $\mathcal{D}(\lambda_0)$: $\theta_0^\star$ is a dual optimal point if $g^T(\theta_0 - \theta_0^\star) \leq 0$ for every dual point θ_0 that is feasible for $\mathcal{D}(\lambda_0)$, where $g := \nabla G(\theta_0^\star) = \theta_0^\star + y$. For $\lambda \leq \lambda_0$, any dual point θ feasible for $\mathcal{D}(\lambda)$ is also dual feasible for $\mathcal{D}(\lambda_0)$ ($|\theta^T x_k| \leq \lambda \leq \lambda_0 \ \ k = 1, \ldots, n$). Since $\theta^\star$ is dual feasible for $\mathcal{D}(\lambda_0)$, we conclude $\theta^\star \in \Theta_2 := \left\{\theta \mid g^T(\theta - \theta_0^\star) \leq 0\right\}$.

Figure 2(a) shows the geometry of Θ_1, Θ_2 and Θ in the dual space; Figure 2(b) shows the geometric interpretation of the inequality test when it is applied to the set Θ.

2.4 SAFE-LASSO theorem

Our criterion to identify the k-th zero in $w^\star$ and thus remove the k-th feature (column) from the feature matrix X in problem $\mathcal{P}(\lambda)$ becomes

$$\lambda > \left|\theta^T x_k\right| = \max(\theta^T x_k, -\theta^T x_k) \ : \ \theta \in \Theta. \tag{5}$$

An equivalent formulation of condition (5) is

$$\lambda > \max(P(\gamma, x_k), P(\gamma, -x_k)),$$

where $P(\gamma, x_k)$ is the optimal value of a convex optimization problem with constraints $\theta \in \Theta_1$ and $\theta \in \Theta_2$:

$$P(\gamma, x_k) := \max_\theta x_k^T\theta \ : \ G(\theta) \geq \gamma, \ g^T(\theta - \theta_0^\star) \geq 0. \tag{6}$$

It turns out that the above problem is simple enough to admit a closed-form solution (see Appendix A). The resulting test can be summarized as follows.

Theorem (SAFE-LASSO) *Consider the LASSO problem $\mathcal{P}(\lambda)$ in (1). Let $\lambda_0 \geq \lambda$ be a value for which an optimal solution $w_0^\star \in \mathbb{R}^n$ is known. Denote by x_k the k-th feature (column) of the matrix X. Define*

$$\mathcal{E} = \{k \mid \lambda > \max(P(\gamma, x_k), P(\gamma, -x_k)\}, \tag{7}$$

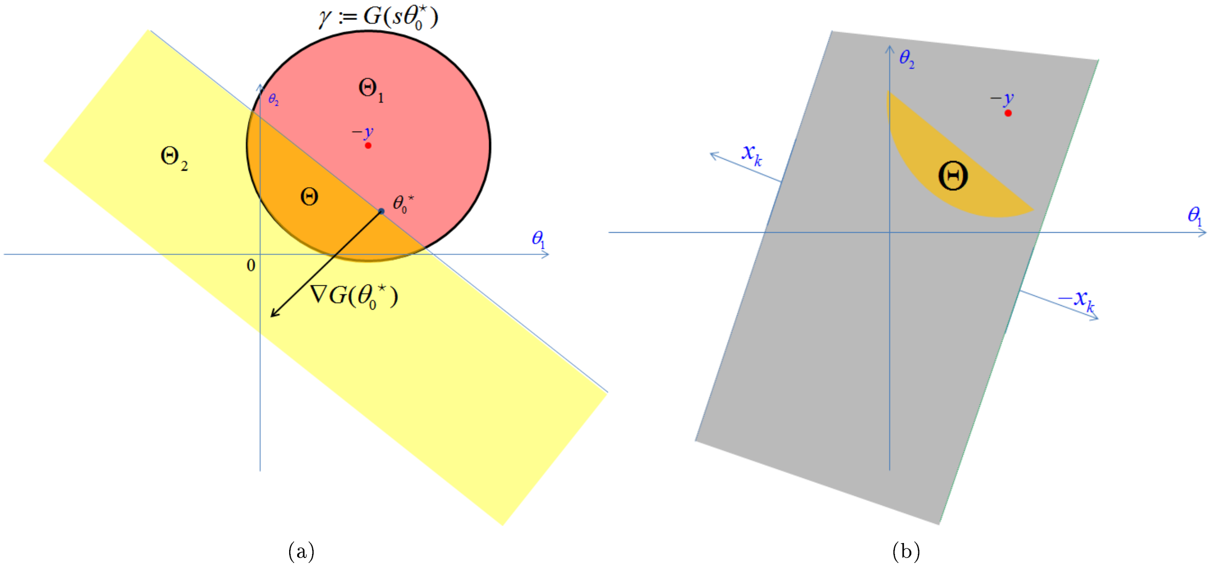
where

$$P(\gamma, x_k) = \begin{cases} \theta_0^{\star T} x_k + \Psi_k \tilde{D}(\gamma) & \|g\|_2^2 \|x_k\|_2 \geq D(\gamma) x_k^T g, \\ -y^T x_k + \|x_k\|_2 D(\gamma) & \|g\|_2^2 \|x_k\|_2 \leq D(\gamma) x_k^T g, \end{cases} \tag{8}$$

with

$$\theta_0^\star = Xw_0^\star - y, \ \ g := \theta_0^\star + y, \ \ \alpha_0 := \theta_0^{\star T}\theta_0^\star, \ \ \beta_0 := |y^T\theta_0^\star|, \ \ \gamma := \tfrac{\beta_0^2}{2\alpha_0}\left(1 - \left(1 - \tfrac{\alpha_0}{\beta_0}\tfrac{\lambda}{\lambda_0}\right)_+^2\right),$$

$$D(\gamma) = \left(\|y\|_2^2 - 2\gamma\right)^{1/2}, \ \ \tilde{D}(\gamma) = \left(D(\gamma)^2 - \|g\|_2^2\right)^{1/2}, \ \ \Psi_k := \left(\|x_k\|_2^2 - \tfrac{(x_k^T g)^2}{\|g\|_2^2}\right)^{1/2}.$$



(a)

(b)

Figure 2: (a) Sets containing $\theta^\star$ in the dual space. The set $\Theta_1 := \{\theta \mid G(\theta) \geq \gamma\}$ shown in red corresponds to a ball in the dual space with center $-y$. The set $\Theta_2 := \left\{\theta \mid g^T(\theta - \theta_0^\star) \leq 0\right\}$ with $g := \nabla G(\theta_0^\star)$ shown in yellow corresponds to a half space with supporting hyperplane passing through $\theta_0^\star$ and normal to $\nabla G(\theta_0^\star)$. The set $\Theta = \Theta_1 \cap \Theta_2$ shown in orange contains the dual optimal point $\theta^\star$. (b) Geometry of the inequality test $\left|\theta^T x_k\right| < \lambda,\ \forall \theta \in \Theta$. The grey shaded region is the slab corresponding to feature x_k, i.e. $\left\{\theta \mid \theta^T x_k \leq \lambda\right\}$. The test $\left|\theta^T x_k\right| < \lambda,\ \forall \theta \in \Theta$ is a strict inequality when the entire set Θ (shown in orange) is inside the slab defined by the feature x_k. In such case, the dual optimal point $\theta^\star \in \Theta$ is also inside the slab and by (3) we conclude $(w^\star)_k = 0$.

Then, for every index $e \in \mathcal{E}$, the e-th entry of $w^\star$ is zero, i.e. $(w^\star)_e = 0$, and feature x_e can be safely eliminated from X a priori to solving the LASSO problem (1). ■

When we don't have access to a solution $w_0^\star$ of $\mathcal{P}(\lambda_0)$, we can set $w_0^\star = 0$ and $\lambda_0 = \lambda_{\max} := \|X^T y\|_\infty$. In this case, the inequality test $\lambda > \max(P(\gamma, x_k), P(\gamma, -x_k)$ in the SAFE-LASSO theorem takes the form $\lambda > \rho_k \lambda_{\max}$, with

$$\rho_k = \frac{\|y\|_2 \|x_k\|_2 + |y^T x_k|}{\|y\|_2 \|x_k\|_2 + \lambda_{\max}}.$$

In the case of scaled data sets, for which $\|y\|_2 = 1$ and $\|x_k\|_2 = 1$ for every k, ρ_k has a convenient geometrical interpretation:

$$\rho_k = \frac{1 + |\cos\alpha_k|}{1 + \max\limits_{1 \le j \le n} |\cos\alpha_j|},$$

where α_k is the angle between the k-th feature and the response vector y. Our test then consists in eliminating features based on how closely they are aligned with the response, *relative* to the most closely aligned feature. For scaled data sets, our test is very similar to standard correlation-based feature selection (Fan and Lv, 2008); in fact, for scaled data sets, the ranking of features it produces is then exactly the same. The big difference here is that our test is not heuristic, as it only eliminates features that are *guaranteed* to be absent when solving the full-fledged sparse supervised learning problem.

2.5 SAFE for LASSO with intercept problem

The SAFE-LASSO theorem can be applied to the LASSO with intercept problem

$$\mathcal{P}_{\text{int}}(\lambda) \ : \ \phi(\lambda) := \min_{w,\nu} \frac{1}{2} \|Xw + \nu - y\|_2^2 + \lambda \|w\|_1 ,$$

with $\nu \in \mathbb{R}^m$ the intercept term, by using a simple transformation. Taking the derivative of the objective function of $\mathcal{P}_{\text{int}}(\lambda)$ w.r.t ν and setting it to zero, we obtain $\nu = \bar{y} - \bar{X}^T w$ with $\bar{y} = (1/m)\mathbf{1}^T y$, $\bar{X} = (1/m)X\mathbf{1}$ and $\mathbf{1} \in \mathbb{R}^m$ the vector of ones . Using the expression of ν, $\mathcal{P}_{\text{int}}(\lambda)$ can be expressed as

$$\mathcal{P}_{\text{int}}(\lambda) \ : \ \phi(\lambda) := \min_{w} \frac{1}{2} \|X_{\text{cent}} w - y_{\text{cent}}\|_2^2 + \lambda \|w\|_1 ,$$

with $X_{\text{cent}} := X - \bar{X}\mathbf{1}^T$ and $y_{\text{cent}} = y - \bar{y}\mathbf{1}$. Thus the SAFE-LASSO theorem can be applied to $\mathcal{P}_{\text{int}}$ and eliminate features (columns) from X_{cent} .

2.6 SAFE for elastic net

The elastic net problem

$$\mathcal{P}_{\text{elastic}}(\lambda) \ : \ \phi(\lambda) := \min_{w} \frac{1}{2} \|Xw - y\|_2^2 + \lambda \|w\|_1 + \frac{1}{2}\epsilon \|w\|_2^2 ,$$

can be expressed in the form of $\mathcal{P}(\lambda)$ by replacing X and y of (1) with $X_{\text{elastic}} = \left(X^T, \sqrt{\epsilon} I\right)^T$ and $y_{\text{elastic}} = \left(y^T, \mathbf{0}^T\right)^T$. This transformation allows us to apply the SAFE-LASSO theorem on $\mathcal{P}_{\text{elastic}}(\lambda)$ and eliminate features from X_{elastic}.

3. Using SAFE

In this section we illustrate the use of SAFE and detail the relevant algorithms.

3.1 SAFE for reducing memory limit problems

SAFE can extend the reach of LASSO solvers to larger size problems than what they could originally handle. In this section, we are interested in solving for $w^\star_d$ the solution of $\mathcal{P}(\lambda_d)$ under a memory constraint of loading only M features. We can compute $w^\star_d$ by solving a sequence of problems, where each problem has a number of features less than our memory limit M. We start by finding an appropriate λ where our SAFE method can eliminate at least $n - M$ features, we then solve a reduced size problem with $L_F \leq M$ features, where $L_F = |\mathcal{E}^c|$ is the number of features left after SAFE and $\mathcal{E}^c = \{1, \ldots, n\} \backslash \mathcal{E}$ is the complement of the set $\mathcal{E}$ in the SAFE-LASSO theorem. We proceed to the next stage as outlined in algorithm 1.

Algorithm 1 SAFE for reducing memory limit problems

```
given a feature matrix $X \in \mathbb{R}^{m \times n}$, response $y \in \mathbb{R}^m$, penalty parameter $\lambda_d$ , memory limit $M$ and LASSO solver: LASSO, i.e. $w^\star = \texttt{LASSO}(X, y, \lambda)$.
initialize $\lambda_0 = \|X^T y\|_\infty$, $w^\star_0 = \mathbf{0} \in \mathbb{R}^n$,
repeat
    1. Use SAFE to search for a $\lambda$ with $LF \leq M$ . Obtain $\lambda$ and $\mathcal{E}$. % $L_F$ is the number of features left after SAFE and $\mathcal{E}$ is the set defined in the SAFE-LASSO theorem.
    2. if $\lambda < \lambda_d$ then $\lambda = \lambda_d$, apply SAFE to obtain $\mathcal{E}$ end if.
    3. Compute the solution $w^\star$. $w^\star(\mathcal{E}^c) = \texttt{LASSO}(X(\mathcal{E}^c, :), y, \lambda)$, $w^*(\mathcal{E}) = 0$; % $w^\star(\mathcal{E}^c)$ and $X(\mathcal{E}^c, :)$ are the elements and columns of $w^\star$ and $X$ defined by the set $\mathcal{E}^c$, respectively. $\mathcal{E}^c = \{1, \ldots, n\} \backslash \mathcal{E}$ is the complement of the set $\mathcal{E}$ .
    4. $\lambda_0 := \lambda$, $w^\star_0 = w^*$.
until $\lambda_0 = \lambda_d$
```

We use a bisection method to find an appropriate value of λ for which SAFE leaves $L_F \in [M - \epsilon_F, \, M]$ features, where ϵ_F is a number of feature tolerance. The bisection method on λ is outlined in algorithm 2.

Algorithm 2 Bisection method on λ.

```
given a feature matrix $X \in \mathbb{R}^{m \times n}$, response $y \in \mathbb{R}^m$, penalty parameter $\lambda_0$ with LASSO solution $w^\star_0$, tolerance $\epsilon_F > 0$ and memory limit $M$.
initialize $l = 0$, and $u = \lambda_0$.
repeat
    1. Set $\lambda := (l + u)/2$.
    2. Use the SAFE-LASSO theorem to obtain $\mathcal{E}$.
    3. Set $L_F = |\mathcal{E}^c|$.
    4. if $L_F > M$ then set $l := \lambda$ else set $u := \lambda$ end if
until $M - L_F \leq \epsilon_F$ and $L_F \leq M$.
```

3.2 SAFE for LASSO run-time reduction

In some applications like Gawalt et al. (2010), it is of interest to solve a sequence of problems $\mathcal{P}(\lambda_1)$, $\ldots \mathcal{P}(\lambda_s)$ for decreasing values of the penalty parameters, i.e. $\lambda_1 \geq \ldots \geq \lambda_s$. The computational complexities of LASSO solvers depend on the number of features and using SAFE might result in run-time improvements. For each problem in the sequence, we can use SAFE to reduce the number of features a priori to using our LASSO solver as shown in algorithm 3.

Algorithm 3 Recursive SAFE for the Lasso

```
given a feature matrix $X \in \mathbb{R}^{m\times n}$, response $y \in \mathbb{R}^m$, a sequence of penalty parameters $\lambda_s \leq \ldots \leq \lambda_1 \leq \|X^T y\|_\infty$, and LASSO solver: LASSO.
initialize $\lambda_0 = \|X^T y\|_\infty$, $w_0^\star = \mathbf{0} \in \mathbb{R}^n$.
for $i = 1$ until $i = s$ do
    1. Set $\lambda_0 = \lambda_{i-1}$, and $\lambda = \lambda_i$.
    2. Use the SAFE-LASSO theorem to obtain $\mathcal{E}$.
    3. Compute the solution $w^\star$. $w^\star(\mathcal{E}^c) = \texttt{LASSO}(X(\mathcal{E}^c,:), y, \lambda)$, $w^*(\mathcal{E}) = 0$. % $w^\star(\mathcal{E}^c)$ and $X(\mathcal{E}^c,:)$ are the elements and columns of $w^\star$ and $X$ defined by the set $\mathcal{E}^c$, respectively. $\mathcal{E}^c = \{1,\ldots,n\} \backslash \mathcal{E}$ is the complement of the set $\mathcal{E}$ .
    4. Set $w_0^\star = w^*$.
end for
```

4. SAFE applied to general ℓ_1-regularized convex problems

The SAFE-LASSO result presented in section 2.4 for the LASSO problem (1) can be adapted to a more general class of l_1- regularized convex problems. We consider the family of problems

$$\mathcal{P}(\lambda) \ : \ \phi(\lambda) := \min_{w,\,\nu} \sum_{i=1}^m f(a_i^T w + b_i v + c_i) + \lambda \left\|w\right\|_1, \tag{9}$$

where f is a closed convex function, and non-negative everywhere, $a_i \in \mathbb{R}^n$, $i = 1, \ldots, m$, $b, c \in \mathbb{R}^m$ are given. The LASSO problem (1) is a special case of (9) with $f(\zeta) = (1/2)\zeta^2$, $a_i \in \mathbb{R}^n$, $i = 1, \ldots, m$ the observations, $c = -y$ is the (negative) response vector, and $b = 0$. Hereafter, we refer to the LASSO problem as $\mathcal{P}_{\text{LASSO}}(\lambda)$ and to the general class of l_1-regularized problems as $\mathcal{P}(\lambda)$. In this section, we outline the steps necessary to derive a SAFE method for the general problem $\mathcal{P}(\lambda)$. We show some preliminary results for deriving SAFE methods when $f(\zeta)$ is the hing loss function, $f_{\text{hi}}(\zeta) = (1-\zeta)_+$, and the logistic loss function $f_{\log}(\xi) = \log(1 + e^{-\xi})$.

4.1 Dual Problem

The first step is to devise the dual of problem (9), which is

$$\mathcal{D}(\lambda) \ : \quad \phi(\lambda) = \max_\theta \ G(\theta) \ : \ \theta^T b = 0, \ \ |\theta^T x_k| \leq \lambda, \ \ k = 1, \ldots, n, \tag{10}$$

where

$$G(\theta) := c^T\theta - \sum_{i=1}^m f^*(\theta_i) \tag{11}$$

with $f^*(\vartheta) = \max_\xi \ \xi\vartheta - f(\xi)$ the conjugate of the loss function $f(\zeta)$, and x_k the k-th column or feature of the feature matrix $X = (a_1, \ldots, a_m)^T \in \mathbb{R}^{m\times n}$. $G(\theta)$ is the dual function, which is, by construction, concave. We assume that strong duality holds and primal and dual optimal points are attained. Due to the optimality conditions for the problem (see Boyd and Vandenberghe (2004)), constraints for which $|\theta^T x_k| < \lambda$ at optimum correspond to a zero element in the primal variable: $(w^\star)_k = 0$, i.e.

$$\left|\theta^{\star T} x_k\right| < \lambda \Rightarrow (w^\star)_k = 0. \tag{12}$$

4.2 Optimality set Θ

For simplicity, we consider only the set $\Theta := \{\theta \mid G(\theta) \geq \gamma\}$ which contains $\theta^\star$ the dual optimal point of $\mathcal{D}(\lambda)$. One way to get a lower bound γ is to find a dual point θ_s that is feasible for the dual problem $\mathcal{D}(\lambda)$, and then set $\gamma = G(\theta_s)$.

To obtain a dual feasible point, we can solve the problem for a higher value $\lambda_0 \geq \lambda$ of the penalty parameter. (In the specific case examined below, we will see how to set λ_0 so that the vector $w_0^\star = 0$ at optimum.) This provides a dual point $\theta_0^\star$ that is feasible for $\mathcal{D}(\lambda_0)$, which satisfies $\lambda_0 = \|X\theta_0\|_\infty$. In turn, $\theta_0^\star$ can be scaled so as to become feasible for $\mathcal{D}(\lambda)$. Precisely, we set $\theta_s = s\theta_0$, with $\|X\theta_s\|_\infty \leq \lambda$ equivalent to $|s| \leq \lambda/\lambda_0$. In order to find the best possible scaling factor s, we solve the one-dimensional, convex problem

$$\gamma(\lambda) := \max_s \ G(s\theta_0) \ : |s| \leq \frac{\lambda}{\lambda_0}. \tag{13}$$

Under mild conditions on the loss function f, the above problem can be solved by bisection in $O(m)$ time. By construction, $\gamma(\lambda)$ is a lower bound on $\phi(\lambda)$. We can generate an initial point $\theta_0^\star$ by solving $\mathcal{P}(\lambda_0)$ with $w_0 = 0$. We get

$$\min_{v_0} \sum_{i=1}^m f(b_i v_0 + c_i) = \min_{v_0} \max_{\theta_0} \theta_0^T(bv_0 + c) - \sum_{i=1}^m f^*\left((\theta_0)_i\right) = \max_{\theta_0 \, : \, b^T\theta_0 = 0} G(\theta_0).$$

Solving the one-dimensional problem above can be often done in closed-form, or by bisection, in $O(m)$. Choosing $\theta_0^\star$ to be any optimal for the corresponding dual problem (the one on the right-hand side) generates a point that is dual feasible for it, that is, $G(\theta_0^\star)$ is finite, and $b^T\theta_0 = 0$.

The point $\theta_0^\star$ satisfies all the constraints of problem $\mathcal{D}(\lambda)$, except perhaps for the constraint $\|X\theta\|_\infty \leq \lambda$, i.e. $\|X\theta_0^\star\|_\infty > \lambda$. Hence, if $\lambda \geq \lambda_0 := \|X\theta_0^\star\|_\infty$, then $\theta_0^\star$ is dual optimal for $\mathcal{D}(\lambda)$ and by the optimality condition (12) we have $w^\star = 0$. Note that, since $\theta_0^\star$ may not be uniquely defined, λ_0 may not necessarily be the smallest value for which $w^\star = 0$ is optimal for the primal problem.

4.3 SAFE method

Assume that a lower bound γ on the optimal value of the learning problem $\phi(\lambda)$ is known: $\gamma \leq \phi(\lambda)$. (Without loss of generality, we can assume that $0 \leq \gamma \leq \sum_{i=1}^m f(c_i)$). The test

$$\lambda > \max(P(\gamma, x_k), P(\gamma, -x_k)),$$

allows to eliminate the k-th feature from the feature matrix X, where $P(\gamma, x_k)$ is the optimal value of a convex optimization problem with two constraints:

$$P(\gamma, x_k) := \max_\theta \ \theta^T x_k \ : \ G(\theta) \geq \gamma, \ \ \theta^T b = 0. \tag{14}$$

Since $P(\gamma, x_k)$ decreases when γ increases, the closer $\phi(\lambda)$ is to its lower bound γ, the more aggressive (accurate) our test is.

By construction, the dual function G is decomposable as a sum of functions of one variable only. This particular structure allows to solve problem (14) very efficiently, using for example interior-point methods, for a large class of loss functions f. Alternatively, we can express the problem in dual form as a convex optimization problem with two scalar variables:

$$P(\gamma, x_k) = \min_{\mu > 0, \, \nu} \ -\gamma\mu + \mu \sum_{i=1}^m f\left(\frac{(x_k)_i + \mu c_i + \nu b_i}{\mu}\right). \tag{15}$$

Note that the expression above involves the perspective of the function f, which is convex (see Boyd and Vandenberghe (2004)). For many loss functions f, the above problem can be efficiently solved using a variety of methods for convex optimization, in (close to) $O(m)$ time. We can also set the variable $\nu = 0$, leading to a simple bisection problem over μ. This amounts to ignore the constraint $\theta^T b = 0$ in the definition of $P(\gamma, x)$, resulting in a more conservative test. More generally, any pair (μ, ν) with $\mu > 0$ generates an upper bound on $P(\gamma, x)$, which in turn corresponds to a valid, perhaps conservative, test.

4.4 SAFE for Sparse Support Vector Machine

We turn to the sparse support vector machine classification problem:

$$\mathcal{P}_{\rm hi}(\lambda) \;:\; \phi(\lambda) := \min_{w,v} \sum_{i=1}^{m} (1 - y_i(z_i^T w + v))_+ + \lambda \|w\|_1, \tag{16}$$

where $z_i \in \mathbb{R}^n$, $i = 1, \ldots, m$ are the data points, and $y \in \{-1, 1\}^m$ is the label vector. The above is a special case of the generic problem (9), where $f(\zeta) := (1 - \xi)_+$ is the hinge loss, $b = y$, $c = 0$, and the feature matrix X is given by $X = [y_1 z_1, \ldots, y_m z_m]^T$, so that $x_k = [y_1 z_1(k), \ldots, y_m z_m(k)]^T$.

We denote by $\mathcal{I}_+, \mathcal{I}_-$ the set of indicies corresponding to the positive and negative classes, respectively, and denote by $m_\pm = |\mathcal{I}_\pm|$ the associated cardinalities. We define $\underline{m} := \min(m_+, m_-)$. Finally, for a generic data vector x, we set $x^\pm = (x_i)_{i \in \mathcal{I}_\pm} \in \mathbb{R}^{m_\pm}$, $k = 1, \ldots, n$, the vectors corresponding to each one of the classes.

The dual problem takes the form

$$\mathcal{D}_{hi}(\lambda) \;:\; \phi(\lambda) := \max_{\theta} G_{\rm hi}(\theta) \;:\; -\mathbf{1} \leq \theta \leq 0, \;\; \theta^T y = 0, \;\; |\theta^T x_k| \leq \lambda, \;\; k = 1, \ldots, n. \tag{17}$$

with $G_{\rm hi}(\theta) = \mathbf{1}^T \theta$.

4.4.1 Test, γ given

Let γ be a lower bound on $\phi(\lambda)$. The optimal value obtained upon setting $w = 0$ in (16) is given by

$$\min_v \sum_{i=1}^{m} (1 - y_i v)_+ = 2 \min(m_+, m_-) := \gamma_{\rm max}. \tag{18}$$

Hence, without loss of generality, we may assume $0 \leq \gamma \leq \gamma_{\rm max}$.

The feature elimination test hinges on the quantity

$$\begin{aligned} P_{\rm hi}(\gamma, x) &= \max_{\theta} \theta^T x \;:\; \mathbf{1}^T \theta \geq \gamma, \;\; \theta^T y = 0, \;\; -\mathbf{1} \leq \theta \leq 0 \\ &= \min_{\mu > 0, \, \nu} -\gamma\mu + \mu \sum_{i=1}^{m} f_{\rm hi}\left(\frac{x_i - \nu y_i}{\mu}\right) \\ &= \min_{\mu > 0, \, \nu} -\gamma\mu + \sum_{i=1}^{m} (\mu + \nu y_i - x_i)_+. \end{aligned} \tag{19}$$

In appendix C.1, we show that for any x, the quantity $P(\gamma, x)$ is finite if and only if $0 \leq \gamma \leq \gamma_{\rm max}$, and can be computed in $O(m \log m)$, or less with sparse data, via a closed-form expression. That expression is simpler to state for $P_{\rm hi}(\gamma, -x)$:

$$\begin{aligned} P_{\rm hi}(\gamma, -x) \;\; = \;\; & \sum_{j=1}^{\lfloor \gamma/2 \rfloor} \bar{x}_j - (\frac{\gamma}{2} - \lfloor \frac{\gamma}{2} \rfloor)(\bar{x}_{\lfloor \gamma/2 \rfloor + 1})_+ + \sum_{j=\lfloor \gamma/2 \rfloor + 1}^{\underline{m}} (\bar{x}_j)_+, \;\; 0 \leq \gamma \leq \gamma_{\rm max} = 2\underline{m}, \\ & \bar{x}_j := x_{[j]}^+ + x_{[j]}^-, \;\; j = 1, \ldots, \underline{m}, \end{aligned}$$

with $x_{[j]}$ the j-th largest element in a vector x, and with the convention that a sum over an empty index set is zero. Note that in particular, since $\gamma_{\max} = 2\underline{m}$:

$$P_{\rm hi}(\gamma_{\max}, -x) = \sum_{i=1}^{m} (x_{[j]}^{+} + x_{[j]}^{-}).$$

4.4.2 SAFE-SVM theorem

Following the construction proposed in section 4.2 for the generic case, we select $\gamma = G_{\rm hi}(\theta)$, where the point θ is feasible for (17), and can found by the scaling method outlined in section 4.2, as follows. The method starts with the assumption that there is a value $\lambda_0 \ge \lambda$ for which we know the optimal value γ_0 of $\mathcal{P}_{\rm hi}(\lambda_0)$.

Specific choices for λ_0, γ_0. Let us first detail how we can find such values λ_0, γ_0.

We can set a value λ_0 such that $\lambda > \lambda_0$ ensures that $w = 0$ is optimal for the primal problem (16). The value that results in the least conservative test is $\lambda_0 = \lambda_{\max}$, where $\lambda_{\max}$ is the smallest value of λ above which $w = 0$ is optimal:

$$\lambda_{\max} := \min_{\theta} \ \|X\theta\|_\infty \ : \ -\theta^T\mathbf{1} \ge \gamma_{\max}, \ \ \theta^T y = 0, \ \ -\mathbf{1} \le \theta \le 0. \tag{20}$$

Since $\lambda_{\max}$ may be relatively expensive to compute, we can settle for an upper bound $\overline{\lambda}_{\max}$ on $\lambda_{\max}$. One choice for $\overline{\lambda}_{\max}$ is based on the test derived in the previous section: we ask that it passes for all the features when $\lambda = \overline{\lambda}_{\max}$ and $\gamma = \gamma_{\max}$. That is, we set

$$\begin{array}{rcl} \overline{\lambda}_{\max} & = & \displaystyle\max_{1 \le k \le n} \ \max\left(P_{\rm hi}(\gamma_{\max}, x_k), P_{\rm hi}(\gamma_{\max}, -x_k)\right) \\ & = & \displaystyle\max_{1 \le k \le n} \ \max\left(\sum_{i=1}^{m} (x_k^{+})_{[j]} + (x_k^{-})_{[j]}, \sum_{i=1}^{m} (-x_k^{+})_{[j]} + (-x_k^{-})_{[j]} \right). \end{array} \tag{21}$$

By construction, we have $\overline{\lambda}_{\max} \ge \lambda_{\max}$, in fact:

$$\begin{array}{rcl} \overline{\lambda}_{\max} & = & \displaystyle\max_{1 \le k \le n} \ \max_{\theta} \ |x_k^T \theta| \ : \ -\theta^T\mathbf{1} \ge \gamma_{\max}, \ \ \theta^T y = 0, \ \ -\mathbf{1} \le \theta \le 0 \\ & = & \displaystyle\max_{\theta} \ \|X\theta\|_\infty \ : \ -\theta^T\mathbf{1} \ge \gamma_{\max}, \ \ \theta^T y = 0, \ \ -\mathbf{1} \le \theta \le 0, \end{array}$$

The two values $\lambda_{\max}, \overline{\lambda}_{\max}$ coincide if the feasible set is a singleton, that is, when $m_+ = m_-$. On the whole interval $\lambda_0 \in [\lambda_{\max}, \overline{\lambda}_{\max}]$, the optimal value of problem $\mathcal{P}_{\rm hi}(\lambda_0)$ is $\gamma_{\max}$.

Dual scaling. The remainder of our analysis applies to any value λ_0 for which we know the optimal value $\gamma_0 \in [0, \gamma_{\max}]$ of the problem $\mathcal{P}_{\rm hi}(\lambda_0)$.

Let θ_0 be a corresponding optimal dual point (as seen shortly, the value of θ_0 is irrelevant, as we will only need to know $\gamma_0 = \mathbf{1}^T\theta_0$). We now scale the point θ_0 to make it feasible for $\mathcal{P}_{\rm hi}(\lambda)$, where λ ($0 \le \lambda \le \lambda_0$) is given. The scaled dual point is obtained as $\theta = s\theta_0$, with s solution to (13). We obtain the optimal scaling $s = \lambda/\lambda_0$, and since $\gamma_0 = -\mathbf{1}^T\theta_0$, the corresponding bound is

$$\gamma(\lambda) = \mathbf{1}^T(s\theta_0) = s\gamma_0 = \gamma_0 \frac{\lambda}{\lambda_0}.$$

Our test takes the form

$$\lambda > \max\left(P_{\rm hi}(\gamma(\lambda), x), P_{\rm hi}(\gamma(\lambda), -x)\right).$$

Let us look at the condition $\lambda > P_{\rm hi}(\gamma(\lambda), -x)$:

$$\exists \, \mu \ge 0, \ \nu \ : \ \lambda > -\gamma(\lambda)\mu + \sum_{i=1}^{m} (\mu + \nu y_i + x_i)_+,$$

which is equivalent to:

$$\lambda > \min_{\mu \geq 0, \nu} \frac{\displaystyle\sum_{i=1}^m (\mu + \nu y_i + x_i)_+}{1 + (\gamma_0/\lambda_0)\mu}.$$

The problem of minimizing the above objective function over variable ν has a closed-form solution. In appendix C.2, we show that for any vectors $x^\pm \in \mathbb{R}^{m_\pm}$, we have

$$\Phi(x^+, x^-) := \min_\nu \sum_{i=1}^{m_+} (x_i^+ + \nu)_+ + \sum_{i=1}^{m_-} (x_i^- - \nu)_+ = \sum_{i=1}^{\underline{m}} (x_{[i]}^+ + x_{[i]}^-)_+,$$

with $x_{[j]}$ the j-th largest element in a vector x. Thus, the test becomes

$$\lambda > \min_{\mu \geq 0} \frac{\displaystyle\sum_{i=1}^{\underline{m}} (2\mu + x_{[i]}^+ + x_{[i]}^-)_+}{1 + (\gamma_0/\lambda_0)\mu}.$$

Setting $\kappa = \lambda_0/(\lambda_0 + \gamma_0 \mu)$, we obtain the following formulation for our test:

$$\lambda > \min_{0 \leq \kappa \leq 1} \sum_{i=1}^{\underline{m}} ((1-\kappa)\frac{2\lambda_0}{\gamma_0} + \kappa(x_{[i]}^+ + x_{[i]}^-))_+ = \frac{2\lambda_0}{\gamma_0} G(\frac{\gamma_0}{2\lambda_0}\overline{x}), \tag{22}$$

where $\overline{x}_i := x_{[i]}^+ + x_{[i]}^-$, $i = 1, \ldots, \underline{m}$, and for $z \in \mathbb{R}^{\underline{m}}$, we define

$$G(z) := \min_{0 \leq \kappa \leq 1} \sum_{i=1}^{\underline{m}} (1 - \kappa + \kappa z_i)_+.$$

We show in appendix C.3 that $G(z)$ admits a closed-form expression, which can be computed in $O(d \log d)$, where d is the number of non-zero elements in vector z. By construction, the test removes all the features if we set $\lambda_0 = \lambda_{\max}$, $\gamma_0 = \gamma_{\max}$, and when $\lambda > \lambda_{\max}$.

Theorem (SAFE-SVM) *Consider the SVM problem $\mathcal{P}_{\text{hi}}(\lambda)$ in (16). Denote by x_k the k-th row of the matrix $[y_1 z_1, \ldots, y_m z_m]$, and let $\mathcal{I}_\pm := \{i \ : \ y_i = \pm 1\}$, $m_\pm := |\mathcal{I}_\pm|$, $\underline{m} := \min(m_+, m_-)$, and $\gamma_{\max} := 2\underline{m}$. Let $\lambda_0 \geq \lambda$ be a value for which the optimal value $\gamma_0 \in [0, \gamma_{\max}]$ of $\mathcal{P}_{\text{sq}}(\lambda_0)$ is known. The following condition allows to remove the k-th feature vector x_k:*

$$\lambda > \frac{2\lambda_0}{\gamma_0} \max\left(G(\frac{\gamma_0}{2\lambda_0}\overline{x}_k), G(\frac{\gamma_0}{2\lambda_0}\underline{x}_k)\right), \tag{23}$$

where $(\overline{x}_k)_i := (x_k)_{[i]}^+ + (x_k)_{[i]}^-$, $(\underline{x}_k)_i := (-x_k)_{[i]}^+ + (-x_k)_{[i]}^-$, $i = 1, \ldots, \underline{m}$, and for $z \in \mathbb{R}^{\underline{m}}$:

$$G(z) = \min_z \frac{1}{1-z} \sum_{i=1}^p (z_i - z)_+ \ : \ z \in \{-\infty, 0, (z_j)_{j \,:\, z_j < 0}\}$$

A specific choice for λ_0 is $\overline{\lambda}_{\max}$ given by (21), with corresponding optimal value $\gamma_0 = \gamma_{\max}$. ■

4.5 SAFE for Sparse Logistic Regression

We now consider the sparse logistic regression problem:

$$\mathcal{P}_{\text{lo}}(\lambda) \ : \ \phi(\lambda) := \min_{w,v} \sum_{i=1}^m \log\left(1 + \exp(-y_i(z_i^T w + v))\right) + \lambda \|w\|_1, \tag{24}$$

with the same notation as in section 4.4. The dual problem takes the form

$$\mathcal{D}_{\text{lo}}(\lambda) \ : \ \phi(\lambda) := \max_\theta \sum_{i=1}^m \left(\theta_i \log(-\theta_i) - (1+\theta_i)^T \log(1+\theta_i)\right) \ : \ \begin{array}{l} -\mathbf{1} \leq \theta \leq 0, \ \ \theta^T y = 0, \\ |\theta^T x_k| \leq \lambda, \ \ k = 1, \ldots, n. \end{array} \tag{25}$$

4.5.1 Test, γ given

Assume that we know a lower bound on the problem, $\gamma \leq \phi(\lambda)$. Since $0 \leq \phi(\lambda) \leq m \log 2$, we may assume that $\gamma \in [0, m \log 2]$ without loss of generality. We proceed to formulate problem (15). For given $x \in \mathbb{R}^m$, and $\gamma \in \mathbb{R}$, we have

$$P_{\log}(\gamma, x) \quad = \quad \min_{\mu > 0,\, \nu} \; -\gamma\mu + \mu \sum_{i=1}^{m} f_{\log}\left(\frac{x_i + y_i\nu}{\mu}\right), \tag{26}$$

which can be computed in $O(m)$ by two-dimensional search, or by the dual interior-point method described in appendix. (As mentioned before, an alternative, resulting in a more conservative test, is to fix ν, for example $\nu = 0$.) Our test to eliminate the k-th feature takes the form

$$\lambda > T_{\log}(\gamma, x_k) := \max(P_{\log}(\gamma, x_k), P_{\log}(\gamma, -x_k)).$$

If γ is known, the complexity of running this test through all the features is $O(nm)$. (In fact, the terms in the objective function that correspond to zero elements of x are of two types, involving $f_{\log}(\pm\nu/\mu)$. This means that the effective dimension of problem (26) is the cardinality d of vector x, which in many applications is much smaller than m.)

4.5.2 Obtaining a dual feasible point

We can construct dual feasible points based on scaling one obtained by choice of a primal point (classifier weight) w_0. This in turn leads to other possible choices for the bound γ.

For $w_0 \in \mathbb{R}^n$ given, we solve the one-dimensional, convex problem

$$v_0 := \arg\min_b \sum_{i=1}^{m} f_{\log}(y_i x_i^T w_0 + y_i b).$$

This problem can be solved by bisection in $O(m)$ time Kim et al. (2007). At optimum, the derivative of the objective is zero, hence $y^T\theta_0 = 0$, where

$$\theta_0(i) := -\frac{1}{1 + \exp(y_i x_i^T w_0 + y_i v_0)}, \quad i = 1, \ldots, m.$$

Now apply the scaling method seen before, and set γ by solving problem (13).

4.5.3 A specific example of a dual point

A convenient, specific choice in the above construction is to set $w_0 = 0$. Then, the intercept v_0 can be explicitly computed, as $v_0 = \log(m_+/m_-)$, where $m_\pm = |\{i : y_i = \pm 1\}|$ are the class cardinalities. The corresponding dual point θ_0 is

$$\theta_0(i) = \begin{cases} -\dfrac{m_-}{m} & (y_i = +1) \\ -\dfrac{m_+}{m} & (y_i = -1), \end{cases} \qquad i = 1, \ldots, m. \tag{27}$$

The corresponding value of λ_0 is (see Kim et al. (2007)):

$$\lambda_0 := \|X^T\theta_0\|_\infty = \max_{1 \leq k \leq n} |\theta_0^T x_k|.$$

We now compute $\gamma(\lambda)$ by solving problem (13), which expresses as

$$\gamma(\lambda) = \max_{|s| \leq \lambda/\lambda_0} G_{\log}(s\theta_0) = \max_{|s| \leq \lambda/\lambda_0} -m_+ f_{\log}^*(-s\frac{m_-}{m}) - m_- f_{\log}^*(-s\frac{m_+}{m}). \tag{28}$$

The above can be solved analytically: it can be shown that $s = \lambda/\lambda_0$ is optimal.

4.5.4 Solving the bisection problem

In this section, we are given $c \in \mathbb{R}^m$, $\gamma \in (0, m \log 2)$, and we consider the problem

$$F^* := \min_{\mu > 0} \; F(\mu) \quad := \quad -\gamma\mu + \mu \sum_{i=1}^m f_{\log}(c(i)/\mu). \tag{29}$$

Problem (29) corresponds to the problem (26), with ν set to a fixed value, and $c(i) = y_i x_i$, $i = 1, \ldots, m$. We assume that $c(i) \neq 0$ for every i, and that $\kappa := m \log 2 - \gamma > 0$. Observe that $F^* \leq F_0 := \lim_{\mu \to 0^+} \; F(\mu) = \mathbf{1}^T c_+$, where c_+ is the positive part of vector c.

To solve this problem via bisection, we initialize the interval of confidence to be $[0, \mu_u]$, with μ_u set as follows. Using the inequality $\log(1 + e^{-x}) \geq \log 2 - (1/2)x_+$, which is valid for every x, we obtain that for every $\mu > 0$:

$$F(\mu) \geq -\gamma\mu + \mu \sum_{i=1}^m \left(\log 2 - \frac{(c(i))_+}{2\mu} \right) = \kappa\mu - \frac{1}{2}\mathbf{1}^T c_+.$$

We can now identify a value μ_u such that for every $\mu \geq \mu_u$, we have $F(\mu) \geq F_0$: it suffices to ensure $\kappa\mu - (1/2)\mathbf{1}^T c_+ \geq F_0$, that is,

$$\mu \geq \mu_u := \frac{(1/2)\mathbf{1}^T c_+ + F_0}{\kappa} = \frac{3}{2} \frac{\mathbf{1}^T c_+}{m \log 2 - \gamma}.$$

4.5.5 Algorithm summary

An algorithm to check if a given feature can be removed from a sparse logistic regression problem works as follows.

Given: λ, k $(1 \leq k \leq n)$, $f_{\log}(x) = \log(1 + e^{-x})$, $f_{\log}^*(\vartheta) = (-\vartheta)\log(-\vartheta) + (\vartheta + 1)\log(\vartheta + 1)$.

1. Set $\lambda_0 = \max\limits_{1 \leq k \leq n} |\theta_0^T x_k|$, where $\theta_0(i) = -m_-/m$ $(y_i = +1)$, $\theta_0(i) = -m_+/m$ $(y_i = -1)$, $i = 1, \ldots, m$.

2. Set
$$\gamma(\lambda) := -m_+ f_{\log}^*(-\frac{\lambda}{\lambda_0}\frac{m_-}{m}) - m_- f_{\log}^*(-\frac{\lambda}{\lambda_0}\frac{m_+}{m}).$$

3. Solve via bisection a pair of one-dimensional convex optimization problems
$$P_\epsilon = \min_{\mu > 0} \; -\gamma(\lambda)\mu + \mu \sum_{i=1}^m f_{\log}(\epsilon y_i (x_k)_i / \mu) \;\; (\epsilon = \pm 1),$$
each with initial interval $[0, \mu_u]$, with
$$\mu_u = \frac{3}{2} \frac{\sum_{i=1}^m (\epsilon y_i (x_k)_i)_+}{m \log 2 - \gamma}.$$

4. If $\lambda > \max(P_+, P_-)$, the k-th feature can be safely removed.

5. Numerical results

In this section we explore the benefits of SAFE by running numerical experiments[1] with different LASSO solvers. We present two kinds of experiments to highlight the two main benefits of SAFE. One kind, in our opinion the most important, shows how memory limitations can be reduced, by allowing to treat larger data sets. The other focuses on measuring computational time reduction when using SAFE a priori to the LASSO solver.

We have used a variety of available algorithms for solving the LASSO problem. We use acronyms to refer to the following methods: IPM stands for the Interior-Point Method for LASSO described in Kim et al. (2007); GLMNET corresponds to the Generalized Linear Model algorithm described in Friedman et al. (2010); TFOCS corresponds to Templates for First-Order Conic Solvers described in Becker et al. (2010); FISTA and Homotopy stand for the Fast Iterative Shrinkage-Thresholding Algorithm and homotopy algorithm, described and implemented in Yang et al. (2010), respectively. Some methods (like IPM, TFOCS) do not return exact zeros in the final solution of the LASSO problem and the issue arises in evaluating the its cardinality. In appendix E, we discuss some issue related to the thresholding of the LASSO solution.

In our experiments, we use data sets derived from text classification sources in Frank and Asuncion (2010). We use medical journal abstracts from PubMed represented in a bag-of-words format, where stop words have been eliminated and capitalization removed. The dimensions of the feature matrix X we use from PubMed is $m = 1,000,000$ abstracts and $n = 127,025$ features (words). There is a total of $82,209,586$ non-zeros in the feature matrix, with an average of about 645 non-zeros per feature (word). We also use data-sets derived from the headlines of *The New York Times,* (NYT) spanning a period of about 20 years (from 1985 to 2007). The number of headlines in the entire NYT data-set is $m = 3,241,260$ and the number of features (words) is $n = 159,943$. There is a total of $14,083,676$ non-zeros in the feature matrix, with an average of about 90 non-zeros per feature.

In some applications such as Gawalt et al. (2010), the goal is to learn a short list of words that are predictive of the appearance of a given query term (say, “lung” or “china”) in the abstracts of medical journals or NYT news. The LASSO problem can be used to produce a summarization of the query term across the many abstracts or headlines considered. To be manageable by a human reader, the list of predictive terms should be very short (say at most 100 terms) with respect to the size of the dictionary n. To produce such a short list, we solve the LASSO problem (1) with different penalty parameters λ, and choose the appropriate penalty λ that would generate enough non-zeros in the LASSO solution (around 100 non-zeros in our case).

5.1 SAFE for reducing memory limit problems

We experiment with PubMed data-set which is too large to be loaded into memory, and thus not amenable to current LASSO solvers. As described before, we are interested in solving the LASSO problem for a regularization parameter that would result in about 100 non-zeros in the solution. We implement algorithm 1 with a memory limit $M = 1,000$ features, where we have observed that for the PubMed data loading more than $1,000$ features causes memory problems in the machine and platform we are using. The memory limit is approximately two orders of magnitudes less than the original number of features n, i.e. $M \approx 0.01n$. Using algorithm 1, we were able to solved the LASSO problem for $\lambda = 0.04\lambda_{max}$ using a sequence of 25 LASSO problem with each problem having a number of features less than $M = 1,000$. Figure 3 shows the simulation result for the PubMed data-set.

[1] In our experiments, we have used an Apple Mac Pro 64-bit workstation, with two 2.26 GHz Quad-Core Intel Xeon processors, 8 MB on-chip shared L3 cache per processor, with 6 GB SDRAM, operating at 1066 MHz.

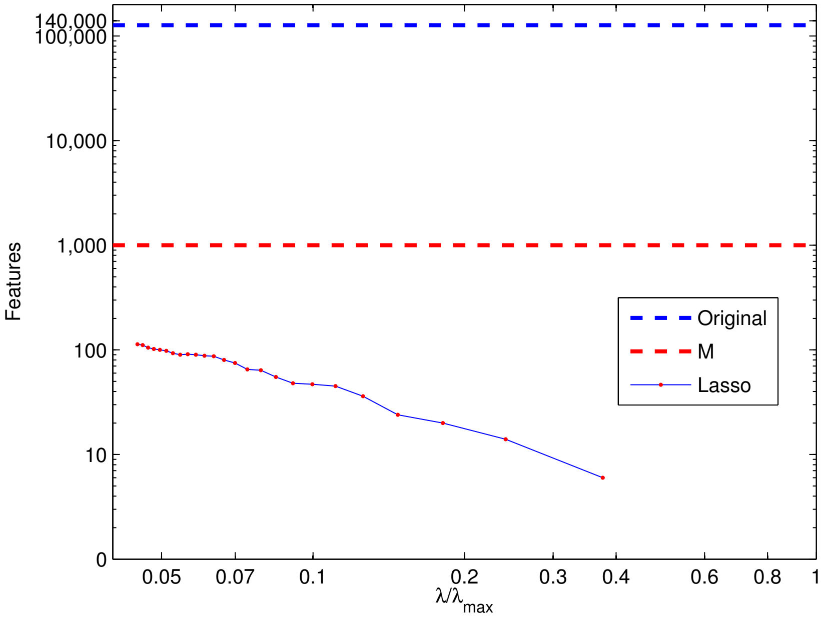
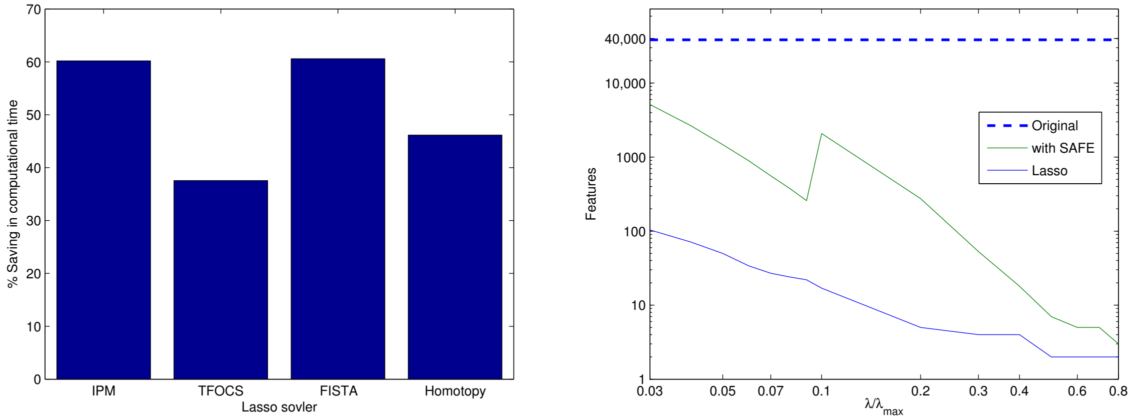


Figure 3: A LASSO problem solved for the PubMed data-set and $\lambda = 0.04\lambda_{max}$ using a sequence of 25 smaller size problems. Each LASSO problem in the sequence has a number of features L_F that satisfies the memory limit $M = 1,000$, i.e $L_F \leq 1,000$.



(a)

(b)

Figure 4: (a) Computational time savings. (b) Lasso solution for the sequence of problem between $0.03\lambda_{max}$ and λ_{max}. The green line shows the number of features we used to solve the LASSO problem after using algoirthm 3.

5.2 SAFE for LASSO run-time reduction

We have used a portion of the NYT data-set corresponding to all headlines in year 1985, the corresponding feature matrix has dimensions $n = 38,377$ features and $m = 192,182$ headlines, with an average of 21 non-zero per feature. We solved the plain LASSO problem and the LASSO problem with SAFE as outlined in algoirthm 3 for a sequence of λ logarithmically distributed between $0.03\lambda_{max}$ and λ_{max}. We have used four LASSO solvers, IPM, TFOCS, FISTA and Homotopy to solve the LASSO problem. Figure 4(a)shows the computational time saving when using SAFE. Figure 4(b) shows the number of features we used to solve the LASSO problem when using SAFE, and the number of non-zeros in the solution. We realize that when using algorithm 3 we solve problems with a number of features at most 10,000 instead of $n = 38,377$ features, this reduction has a direct impact on the solving time of the LASSO problem as demonstrated in figure 4(a).

5.3 SAFE for LASSO with intercept problem

We return to the LASSO with intercept problem discussed in section 2.5. We generate a feature matrix $X \in \mathbb{R}^{m\times n}$ with $m = 500$, $n = 10^6$. The entries of X has a $\mathcal{N}(0,1)$ normal distributed and sparsity density $d = 0.1$. We also generate a vector of coefficients $\omega \in \mathbb{R}^n$ with 50 non-zero entries. The response y is generated by setting $y = X\omega + 0.01\eta$, where η is a vector in $\mathbb{R}^m$ with $\mathcal{N}(0,1)$ distribution. We use GLMNET implemented in `R` to solve the LASSO problem with intercept. The generated data, X and y can be loaded into `R` , yet memory problems occur when we try to solve the LASSO problem. We use algorithm 1 with memory limit $M = 10,000$ features and $\lambda = 0.33\lambda_{max}$. Figure 5 shows the number of non-zeros in the solution of the 352 sequence of problems used to obtain the solution at $\lambda = 0.33\lambda_{max}$.

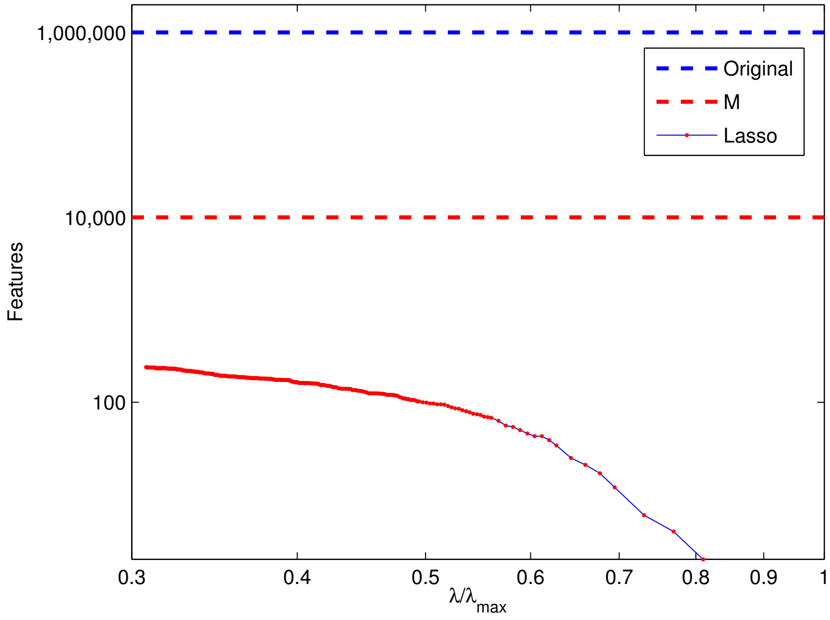


Figure 5: A LASSO problem with intercept solved for randomly generated data-set and $\lambda = 0.33\lambda_{max}$ using a sequence of 352 smaller size problems. Each LASSO problem in the sequence has a number of features L_F that satisfies the memory limit $M = 10,000$, i.e $L_F \leq 1000$.

Appendix A. Expression of $P(\gamma, x_k)$ (LASSO)

We can express problem (6) in dual form as a convex optimization problem with two scalar variables, μ_1 and μ_2:

$$\begin{aligned}
P(\gamma, x_k) &= \min_{\mu_1,\mu_2 \geq 0} \max_{\theta} x_k^T\theta + \mu_1 \left(G(\theta) - \gamma\right) + \mu_2 g^T \left(\theta - \theta_0^\star\right) \\
&= \min_{\mu_1,\mu_2 \geq 0} -\mu_1\gamma - \mu_2 g^T\theta_0^\star + \max_{\theta} x_k^T\theta + \mu_1 G(\theta) + \mu_2 g^T\theta \\
&= \min_{\mu_1,\mu_2 \geq 0} -\mu_1\gamma - \mu_2 g^T\theta_0^\star + \mu_1 \max_{\theta} \left(\frac{x_k^T - \mu_1 y^T + \mu_2 g^T}{\mu_1}\theta - \frac{1}{2}\left\|\theta\right\|_2^2 \right)
\end{aligned}$$

We obtain:

$$P(\gamma, x_k) = \min_{\mu_1,\mu_2 \geq 0} L(\mu_1, \mu_2) \tag{30}$$

with

$$L(\mu_1, \mu_2) = -x_k^T y + \frac{\mu_1}{2}D^2 + \frac{1}{2\mu_1}\left\|x_k\right\|_2^2 + \frac{\mu_2^2}{2\mu_1}\left\|g\right\|_2^2 + \frac{\mu_2}{\mu_1}x_k^T g - \mu_2\left\|g\right\|_2^2, \tag{31}$$

and $D := \left(\left\|y\right\|_2^2 - 2\gamma\right)^{1/2}$.

To solve (30), we take the derivative of (31) w.r.t μ_2 and set it to zero:

$$\mu_2\left\|g\right\|_2^2 + x_k^T g - \mu_1\left\|g\right\|_2^2 = 0.$$

This implies that $\mu_2 = \max(0, \mu_1 - \frac{x_k^T g}{\left\|g\right\|_2^2})$. When $\mu_1 \leq \frac{x_k^T g}{\left\|g\right\|_2^2}$, we have $\mu_2 = 0$, $\mu_1 = \frac{\left\|x_k\right\|_2}{D}$ and $P(\gamma, x_k)$ takes the value:

$$P(\gamma, x_k) = -y^T x_k + \left\|x_k\right\|_2 D.$$

On the other hand, when $\mu_1 \geq \frac{x_k^T g}{\left\|g\right\|_2^2}$, we take the derivative of (31) w.r.t μ_1 and set it to zero:

$$\tilde{D}^2\mu_1^2 = \Psi_k^2,$$

with $\Psi_k = \left(\left\|x_k\right\|_2^2 - \frac{\left(x_k^T g\right)^2}{\left\|g\right\|_2^2}\right)^{1/2}$ and $\tilde{D} = \left(D^2 - \left\|g\right\|_2^2\right)^{1/2}$. Substituting μ_1 and μ_2 in (30), $P(\gamma, x_k)$ takes the value:

$$P(\gamma, x_k) = \theta_0^{\star T} x_k + \Psi_k\tilde{D}.$$

Appendix B. Expression of $P(\gamma, x)$, general case

We show that the quantity $P(\gamma, x)$ defined in (14) can be expressed in dual form (15). This is a simple consequence of duality:

$$\begin{aligned}
P(\gamma, x) &= \max_{\theta} \theta^T x \ : \ G(\theta) \geq \gamma, \ \ \theta^T b = 0 \\
&= \max_{\theta} \min_{\mu > 0, \, \nu} \ \theta^T x + \mu(G(\theta) - \gamma) - \nu \theta^T b \\
&= \min_{\mu > 0, \, \nu} \max_{\theta} \ \theta^T x + \mu(-y^T \theta - \sum_{i=1}^m f^*(\theta(i)) - \gamma) - \nu \theta^T b \\
&= \min_{\mu > 0, \, \nu} \ -\gamma \mu + \max_{\theta} \ \theta^T (x - \mu y - \nu z) - \mu \sum_{i=1}^m f^*(\theta(i)) \\
&= \min_{\mu > 0, \, \nu} \ -\gamma \mu + \mu \left(\max_{\theta} \ \frac{1}{\mu} \theta^T (x - \mu y - \nu z) - \sum_{i=1}^m f^*(\theta(i)) \right) \\
&= \min_{\mu > 0, \, \nu} \ -\gamma \mu + \mu \sum_{i=1}^m f\left(\frac{x_i - \mu y(i) - \nu b_i}{\mu} \right).
\end{aligned}$$

Appendix C. SAFE test for SVM

In this section, we examine various optimization problems involving polyhedral functions in one or two variables, which arise in section 4.4.1 for the computation of $P_{\rm hi}(\gamma, x)$ as well as in the SAFE-SVM theorem of section 4.4.2.

C.1 Computing $P_{\rm hi}(\gamma, x)$

We first focus on the specific problem of computing the quantity defined in (19). To simplify notation, we will consider the problem of computing $P_{\rm hi}(\gamma, -x)$, that is:

$$P_{\rm hi}(\gamma, -x) = \min_{\mu \geq 0, \, \nu} \ -\gamma \mu + \sum_{i=1}^m (\mu + \nu y_i + x_i)_+, \tag{32}$$

where $y \in \{-1, 1\}^m$, $x \in \mathbb{R}^m$ and γ are given, with $0 \leq \gamma \leq \gamma_0 := 2 \min(m_+, m_-)$. Here, $\mathcal{I}_\pm := \{i \ : \ y_i = \pm 1\}$, and $x^+ = (x_i)_{i \in \mathcal{I}_+}$, $x^- = (x_i)_{i \in \mathcal{I}_-}$, $m_\pm = |\mathcal{I}_\pm|$, and $\underline{m} = \min(m_+, m_-)$. Without loss of generality, we assume that both x^+, x^- are both sorted in descending order: $x_1^\pm \geq \ldots \geq x_{m_\pm}^\pm$.

Using $\alpha = \mu + \nu$, $\beta = \mu - \nu$, we have

$$\begin{aligned}
P_{\rm hi}(\gamma, -x) &= \min_{\alpha + \beta \geq 0} \ -\frac{\gamma}{2}(\alpha + \beta) + \sum_{i=1}^{m_+} (x_i^+ + \alpha)_+ + \sum_{i=1}^{m_-} (x_i^- + \beta)_+ \\
&= \min_{\alpha, \, \beta} \max_{t \geq 0} \ -\frac{\gamma}{2}(\alpha + \beta) + \sum_{i=1}^{m_+} (x_i^+ + \alpha)_+ + \sum_{i=1}^{m_-} (x_i^- + \beta)_+ - t(\alpha + \beta) \\
&= \max_{t \geq 0} \min_{\alpha, \, \beta} \ -(\frac{\gamma}{2} + t)(\alpha + \beta) + \sum_{i=1}^{m_+} (x_i^+ + \alpha)_+ + \sum_{i=1}^{m_-} (x_i^- + \beta)_+ \\
&= \max_{t \geq 0} \ F(\frac{\gamma}{2} + t, x^+) + F(\frac{\gamma}{2} + t, x^-),
\end{aligned} \tag{33}$$

where, for $h \in \mathbb{R}$ and $x \in \mathbb{R}^p$, $x_1 \geq \ldots \geq x_p$, we set

$$F(h, x) := \min_{z} \ -hz + \sum_{i=1}^p (z + x_i)_+, \tag{34}$$

Expression of the function F. If $h > p$, then with $z \to +\infty$ we obtain $F(h,x) = -\infty$. Similarly, if $h < 0$, then $z \to -\infty$ yields $F(h,x) = -\infty$. When $0 \le h \le p$, we proceed by expressing F in dual form:

$$F(h,x) = \max_u \; u^T x \; : \; 0 \le u \le \mathbf{1}, \;\; u^T \mathbf{1} = h.$$

If $h = p$, then the only feasible point is $u = \mathbf{1}$, so that $F(p,x) = \mathbf{1}^T x$. If $0 \le h < 1$, choosing $u_1 = h$, $u_2 = \ldots = u_p = 0$, we obtain the lower bound $F(h,x) \ge hx_1$, which is attained with $z = -x_1$.

Assume now that $1 \le h < p$. Let $h = q + r$, with $q = \lfloor h \rfloor$ the integer part of h, and $0 \le r < 1$. Choosing $u_1 = \ldots = u_q = 1$, $u_{q+1} = r$, we obtain the lower bound

$$F(h,x) \ge \sum_{j=1}^{q} x_j + r x_{q+1},$$

which is attained by choosing $z = -x_{q+1}$ in the expression (34).

To summarize:

$$F(h,x) = \begin{cases} hx_1 & \text{if } 0 \le h < 1, \\ \displaystyle\sum_{j=1}^{\lfloor h \rfloor} x_j + (h - \lfloor h \rfloor) x_{\lfloor h \rfloor + 1} & \text{if } 1 \le h < p, \\ \displaystyle\sum_{j=1}^{p} x_j & \text{if } h = p, \\ -\infty & \text{otherwise.} \end{cases} \tag{35}$$

A more compact expression, valid for $0 \le h \le p$ if we set $x_{p+1} = x_p$ and assume that a sum over an empty index sets is zero, is

$$F(h,x) = \sum_{j=1}^{\lfloor h \rfloor} x_j + (h - \lfloor h \rfloor) x_{\lfloor h \rfloor + 1}, \;\; 0 \le h \le p.$$

Note that $F(\cdot, x)$ is the piece-wise linear function that interpolates the sum of the h largest elements of x at the integer break points $h = 0, \ldots, p$.

Expression of $P_{\rm hi}(\gamma, -x)$. We start with the expression found in (33):

$$P_{\rm hi}(\gamma, -x) = \max_{t \ge 0} \; F(\frac{\gamma}{2} + t, x^+) + F(\frac{\gamma}{2} + t, x^-).$$

Since the domain of $F(\cdot, x^+) + F(\cdot, x^-)$ is $[0, \underline{m}]$, and with $0 \le \gamma/2 \le \gamma_0/2 = \underline{m}$, we get

$$P_{\rm hi}(\gamma, -x) = \max_{\gamma/2 \le h \le \underline{m}} \; G(h, x^+, x^-) := F(h, x^+) + F(h, x^-).$$

Since $F(\cdot, x)$ with $x \in \mathbb{R}^p$ is a piece-wise linear function with break points at $0, \ldots, p$, a maximizer of $G(\cdot, x^+, x^-)$ over $[\gamma/2, \underline{m}]$ lies in $\{\gamma/2, \lfloor \gamma/2 \rfloor + 1, \ldots, \underline{m}\}$. Thus,

$$P_{\rm hi}(\gamma, -x) = \max \left(G(\frac{\gamma}{2}, x^+, x^-), \max_{h \in \{\lfloor \gamma/2 \rfloor + 1, \ldots, \underline{m}\}} \; G(h, x^+, x^-) \right).$$

Let us examine the second term, and introduce the notation $\bar{x}_j := x_j^+ + x_j^-$, $j = 1, \ldots, \underline{m}$:

$$\begin{aligned} \max_{h \in \{\lfloor \gamma/2 \rfloor + 1, \ldots, \underline{m}\}} \; G(h, x^+, x^-) &= \max_{h \in \{\lfloor \gamma/2 \rfloor + 1, \ldots, \underline{m}\}} \; \sum_{j=1}^{h} (x_j^+ + x_j^-) \\ &= \sum_{j=1}^{\lfloor \gamma/2 \rfloor + 1} \bar{x}_j + \sum_{j = \lfloor \gamma/2 \rfloor + 2}^{\underline{m}} (\bar{x}_j)_+, \end{aligned}$$

with the convention that sums over empty index sets are zero. Since

$$G(\frac{\gamma}{2}, x^+, x^-) = \sum_{j=1}^{\lfloor \gamma/2 \rfloor} \bar{x}_j + (\frac{\gamma}{2} - \lfloor \frac{\gamma}{2} \rfloor) \bar{x}_{\lfloor \gamma/2 \rfloor + 1},$$

we obtain

$$P_{\rm hi}(\gamma, -x) = \sum_{j=1}^{\lfloor \gamma/2 \rfloor} \bar{x}_j + \max \left((\frac{\gamma}{2} - \lfloor \frac{\gamma}{2} \rfloor) \bar{x}_{\lfloor \gamma/2 \rfloor + 1}, \bar{x}_{\lfloor \gamma/2 \rfloor + 1} + \sum_{j=\lfloor \gamma/2 \rfloor + 2}^{\underline{m}} (\bar{x}_j)_+ \right).$$

An equivalent expression is:

$$\begin{array}{rcl} P_{\rm hi}(\gamma, -x) & = & \displaystyle\sum_{j=1}^{\lfloor \gamma/2 \rfloor} \bar{x}_j - (\frac{\gamma}{2} - \lfloor \frac{\gamma}{2} \rfloor)(-\bar{x}_{\lfloor \gamma/2 \rfloor + 1})_+ + \sum_{j=\lfloor \gamma/2 \rfloor + 1}^{\underline{m}} (\bar{x}_j)_+, \;\; 0 \leq \gamma \leq 2\underline{m}, \\ & & \bar{x}_j := x_j^+ + x_j^-, \;\; j = 1, \ldots, \underline{m}. \end{array}$$

The function $P_{\rm hi}(\cdot, -x)$ linearly interpolates the values obtained for $\gamma = 2q$ with q integer in $\{0, \ldots, \underline{m}\}$:

$$P_{\rm hi}(2q, -x) = \sum_{j=1}^{q} \bar{x}_j + \sum_{j=q+1}^{\underline{m}} (\bar{x}_j)_+.$$

C.2 Computing $\Phi(x^+, x^-)$

Let us consider the problem of computing

$$\Phi(x^+, x^-) := \min_{\nu} \sum_{i=1}^{m_+} (x_i^+ + \nu)_+ + \sum_{i=1}^{m_-} (x_i^- - \nu)_+,$$

with $x^\pm \in \mathbb{R}^{m_\pm}$, $x_1^\pm \geq \ldots \geq x_{m_\pm}^\pm$, given. We can express $\Phi(x^+, x^-)$ in terms of the function F defined in (34):

$$\begin{array}{rcl} \Phi(x^+, x^-) & = & \displaystyle\min_{\nu_+, \nu_-} \sum_{i \in \mathcal{I}_+} (x_i^+ + \nu^+)_+ + \sum_{i \in \mathcal{I}_-} (x_i^- - \nu^-)_+ \;:\; \nu^+ = \nu^- \\ & = & \displaystyle\max_h \min_{\nu^+, \nu^-} -h(\nu^+ - \nu^-) + \sum_{i \in \mathcal{I}_+} (x_i^+ + \nu^+)_+ + \sum_{i \in \mathcal{I}_-} (x_i^- - \nu^-)_+ \\ & = & \displaystyle\max_h \min_{\nu^+, \nu^-} -h\nu^+ + \sum_{i \in \mathcal{I}_+} (x_i^+ + \nu^+)_+ + h\nu^- + \sum_{i \in \mathcal{I}_-} (x_i^- - \nu^-)_+ \\ & = & \displaystyle\max_h \left(\min_{\nu} -h\nu + \sum_{i \in \mathcal{I}_+} (x_i^+ + \nu)_+ \right) + \left(\min_{\nu} -h\nu + \sum_{i \in \mathcal{I}_-} (x_i^- + \nu)_+ \right) \;\; (\nu_+ = -\nu_- = \nu) \\ & = & \displaystyle\max_h F(h, x^+) + F(h, x^-) \\ & = & \displaystyle\max_{0 \leq h \leq \underline{m}} F(h, x^+) + F(h, x^-) \\ & = & \max(A, B, C), \end{array}$$

where F is defined in (34), and

$$A = \max_{0 \leq h < 1} F(h, x^+) + F(h, x^-), \;\; B := \max_{1 \leq h < \underline{m}} F(h, x^+) + F(h, x^-)), \;\; C = F(\underline{m}, x^+) + F(\underline{m}, x^-).$$

We have

$$A := \max_{0 \leq h < 1} F(h, x^+) + F(h, x^-) = \max_{0 \leq h < 1} h(x_1^+ + x_1^-) = (x_1^+ + x_1^-)_+.$$

Next:

$$\begin{aligned} B &= \max_{1 \le h < \underline{m}} F(h, x^+) + F(h, x^-) \\ &= \max_{q \in \{1,\ldots,\underline{m}-1\}, r \in [0,1[} \sum_{i=1}^{q} (x_i^+ + x_i^-) + r(x_{q+1}^+ + x_{q+1}^-) \\ &= \max_{q \in \{1,\ldots,\underline{m}-1\}} \sum_{i=1}^{q} (x_i^+ + x_i^-) + (x_{q+1}^+ + x_{q+1}^-)_+ \\ &= (x_1^+ + x_1^-) + \sum_{i=2}^{\underline{m}} (x_i^+ + x_i^-)_+ . \end{aligned}$$

Observe that

$$B \ge C = \sum_{i=1}^{\underline{m}} (x_i^+ + x_i^-).$$

Moreover, if $(x_1^+ + x_1^-) \ge 0$, then $B = \sum_{i=1}^{\underline{m}} (x_i^+ + x_i^-)_+ \ge A$. On the other hand, if $x_1^+ + x_1^- \le 0$, then $x_i^+ + x_i^- \le 0$ for $2 \le j \le \underline{m}$, and $A = \sum_{i=1}^{\underline{m}} (x_i^+ + x_i^-)_+ \ge x_1^+ + x_1^- = B$. In all cases,

$$\Phi(x^+, x^-) = \max(A, B, C) = \sum_{i=1}^{\underline{m}} (x_i^+ + x_i^-)_+ .$$

C.3 SAFE-SVM test

Now we consider the problem that arises in the SAFE-SVM test (22):

$$G(z) := \min_{0 \le \kappa \le 1} \sum_{i=1}^{p} (1 - \kappa + \kappa z_i)_+ ,$$

where $z \in \mathbb{R}^p$ is given. (The SAFE-SVM condition (22) involves $z_i = \gamma_0/(2\lambda_0)(x_{[i]}^+ + x_{[i]}^-)$, $i = 1, \ldots, p := \underline{m}$.) We develop an algorithm to compute the quantity $G(z)$, the complexity of which grows as $O(d \log d)$, where d is (less than) the number of non-zero elements in z.

Define $\mathcal{I}_\pm = \{i \,:\, \pm z_i > 0\}$, $k := |\mathcal{I}_+|$, $h := |\mathcal{I}_-|$, $l = \mathcal{I}_0$, $l := |\mathcal{I}_0|$.

If $k = 0$, $\mathcal{I}_+$ is empty, and $\kappa = 1$ achieves the lower bound of 0 for $G(z)$. If $k > 0$ and $h = 0$, that is, $k + l = p$, then $\mathcal{I}_-$ is empty, and an optimal κ is attained in $\{0, 1\}$. In both cases ($\mathcal{I}_+$ or $\mathcal{I}_-$ empty), we can write

$$G(z) = \min_{\kappa \in \{0,1\}} \sum_{i=1}^{p} (1 - \kappa + \kappa z_i)_+ = \min\left(p, S_+\right), \quad S_+ := \sum_{i \in \mathcal{I}_+} z_i,$$

with the convention that a sum over an empty index set is zero.

Next we proceed with the assumption that $k \neq 0$ and $h \neq 0$. Let us re-order the elements of $\mathcal{I}_-$ in decreasing fashion, so that $z_i > 0 = z_{k+1} = \ldots = z_{k+l} > z_{k+l+1} \ge \ldots \ge z_p$, for every $i \in \mathcal{I}_+$. (The case when $\mathcal{I}_0$ is empty is handled simply by setting $l = 0$ in our formula.) We have

$$G(z) = k + l + \min_{0 \le \kappa \le 1} \left\{ \kappa \alpha + \sum_{i=k+l+1}^{p} (1 - \kappa + \kappa z_i)_+ \right\},$$

where, $\alpha := S_+ - k - l$. The minimum in the above is attained at $\kappa = 0, 1$ or one of the break points $1/(1 - z_j) \in (0, 1)$, where $j \in \{k + l + 1, \ldots, p\}$. At $\kappa = 0, 1$, the objective function of the original problem takes the values S_+, p, respectively. The value of the same objective function at the break point $\kappa = 1/(1-z_j)$, $j = k+l+1, \ldots, p$, is $k+l+G_j(z)$, where

$$\begin{aligned}
G_j(z) &:= \frac{\alpha}{1-z_j} + \sum_{i=k+l+1}^{p} \left(\frac{z_i - z_j}{1-z_j} \right)_+ \\
&= \frac{\alpha}{1-z_j} + \frac{1}{1-z_j} \sum_{i=k+l+1}^{j-1} (z_i - z_j) \\
&= \frac{1}{1-z_j} \left(\alpha - (j-k-l-1) z_j + \sum_{i=k+l+1}^{j-1} z_i \right) \\
&= \frac{1}{1-z_j} \left(S_+ - (j-1) z_j - (k+l)(1-z_j) + \sum_{i=k+l+1}^{j-1} z_i \right) \\
&= -(k+l) + \frac{1}{1-z_j} \left(\sum_{i=1}^{j-1} z_i - (j-1) z_j \right).
\end{aligned}$$

This allows us to write

$$G(z) = \min \left(p, \sum_{i=1}^{k} z_i, \min_{j \in \{k+l+1, \ldots, p\}} \frac{1}{1-z_j} \left(\sum_{i=1}^{j-1} z_i - (j-1) z_j \right) \right).$$

The expression is valid when $k+l = p$ ($h = 0$, $\mathcal{I}_-$ is empty), $l = 0$ ($\mathcal{I}_0$ is empty), or $k = 0$ ($\mathcal{I}_+$ is empty) with the convention that the sum (resp. minimum) over an empty index set is 0 (resp. $+\infty$).

We can summarize the result with the compact formula:

$$G(z) = \min_{z} \frac{1}{1-z} \sum_{i=1}^{p} (z_i - z)_+ \ : \ z \in \{-\infty, 0, (z_j)_{j \,:\, z_j < 0}\}.$$

Let us detail an algorithm for computing $G(z)$. Assume $h > 0$. The quantity

$$\underline{G}(z) := \min_{k+l+1 \le j \le p} (G_j(z))$$

can be evaluated in less than $O(h)$, via the following recursion:

$$\begin{aligned}
G_{j+1}(z) &= \frac{1-z_j}{1-z_{j+1}} G_j(z) - j \frac{z_{j+1} - z_j}{1-z_{j+1}} \quad , \quad j = k+l+1, \ldots, p, \\
\underline{G}_{j+1}(z) &= \min(\underline{G}_j(z), G_{j+1}(z))
\end{aligned} \tag{36}$$

with initial values

$$G_{k+l+1}(z) = \underline{G}_{k+l+1}(z) = \frac{1}{1-z_{k+l+1}} \left(\sum_{i=1}^{k+l} z_i - (k+l) z_{k+l+1} \right).$$

On exit, $\underline{G}(z) = \underline{G}_p$.

Our algorithm is as follows.

Algorithm for the evaluation of $G(z)$.

1. Find the index sets $\mathcal{I}_+$, $\mathcal{I}_-$, $\mathcal{I}_0$, and their respective cardinalities k, h, l.
2. If $k = 0$, set $G(z) = 0$ and exit.

3. Set $S_+ = \sum_{i=1}^k z_i$.
4. If $h = 0$, set $G(z) = \min(p, S_+)$, and exit.
5. If $h > 0$, order the negative elements of z, and evaluate $\underline{G}(z)$ by the recursion (36). Set $G(z) = \min(p, S_+, \underline{G}(z))$ and exit.

The complexity of evaluating $G(z)$ thus grows in $O(k + h \log h)$, which is less than $O(d \log d)$, where $d = k + h$ is the number of non-zero elements in z.

Appendix D. Computing $P_{\log}(\gamma, x)$ via an interior-point method

We consider the problem (26) which arises with the logistic loss. We can use a generic interior-point method Boyd and Vandenberghe (2004), and exploit the decomposable structure of the dual function $G_{\log}$. The algorithm is based on solving, via a variant of Newton's method, a sequence of linearly constrained problems of the form

$$\min_\theta \ \tau x^T \theta + \log(G_{\log}(\theta) - \gamma) + \sum_{i=1}^m \log(-\theta - \theta^2) \ : \ z^T\theta = 0,$$

where $\tau > 0$ is a parameter that is increased as the algorithm progresses, and the last terms correspond to domain constraints $\theta \in [-1, 0]^m$. As an initial point, we can take the point θ generated by scaling, as explained in section 4.2. Each iteration of the algorithm involves solving a linear system in variable δ, of the form $H\delta = h$, with H is a rank-two modification to the Hessian of the objective function in the problem above. It is easily verified that the matrix H has a "diagonal plus rank-two" structure, that is, it can be written as $H = D - gg^T - vv^T$, where the $m \times m$ matrix D is diagonal and $g, v \in \mathbb{R}^m$ are computed in $O(m)$. The matrix H can be formed, as the associated linear system solved, in $O(m)$ time. Since the number of iterations for this problem with two constraints grows as $\log(1/\epsilon)O(1)$, the total complexity of the algorithm is $\log(1/\epsilon)O(m)$ (ϵ is the absolute accuracy at which the interior-point method computes the objective). We note that memory requirements for this method also grow as $O(m)$.

Appendix E. On thresholding methods for LASSO

Sparse classification algorithms may return a classifier vector w with many small, but not exactly zero, elements. This implies that we need to choose a thresholding rule to decide which elements to set to zero. In this section, we discuss an issue related to the thresholding rule originally proposed for the interior point method for Logistic algorithm in Koh et al. (2007), and propose a new thresholding rule.

The KKT thresholding rule. Recall that the primal problem for LASSO is

$$\phi(\lambda) = \min_w \frac{1}{2}\|X^T w - y\|_2^2 + \lambda \|w\|_1. \tag{37}$$

Observing that the KKT conditions imply that, at optimum, $(X(X^T w - y))_k = \lambda \text{sign}(w_k)$, with the convention $\text{sign}(0) \in [-1, 1]$, and following the ideas of Koh et al. (2007), the following thresholding rule can be proposed: at optimum, set component w_k to 0 whenever

$$|(X(X^T w - y))_k| \leq 0.9999\lambda. \tag{38}$$

We refer to this rule as the "KKT" rule.

The IPM-LASSO algorithm takes as input a "duality gap" parameter ϵ, which controls the relative accuracy on the objective. When comparing the IPM code results with other algorithms such as GLMNET, we observed chaotic behaviors when applying the KKT rule, especially when the duality gap parameter ϵ was not small enough. More surprisingly, when this parameter is not small enough, some components w_k with absolute values not close to 0 can be thresholded. This suggests that the KKT rule should only be used for problems solved with a small enough duality gap ϵ. However, setting the duality gap to a small value can dramatically slow down computations. In our experiments, changing the duality gap from $\epsilon = 10^{-4}$ to 10^{-6} (resp. 10^{-8}) increased the computational time by 30% to 40% (resp. 50 to 100%).

An alternative method. We propose an alternative thresholding rule, which is based on controlling the perturbation of the objective function that is induced by thresholding.

Assume that we have solved the LASSO problem above, with a given duality gap parameter ϵ. If we denote by w^* the classifier vector delivered by the IPM algorithm, w^* is ϵ-sub-optimal, that is, achieves a value

$$\phi^* = \frac{1}{2}\|Xw^* - y\|_2^2 + \lambda\|w^*\|_1,$$

with $0 \leq \phi^* - \phi(\lambda) \leq \epsilon\phi(\lambda)$.

For a given threshold $\tau > 0$, consider the thresholded vector $\tilde{w}(\tau)$ defined as

$$\tilde{w}_k(\tau) \quad = \quad \begin{cases} 0 & \text{if } |w_k^*| \leq \tau, \\ w_k^* & \text{otherwise,} \end{cases} \quad k = 1, \ldots, n.$$

We have $\tilde{w}(\tau) = w^* + \delta(\tau)$ where the vector of perturbation $\delta(\tau)$ is such that

$$\delta_k(\tau) \quad = \quad \begin{cases} -w_k^* & \text{if } |w_k^*| \leq \tau, \\ 0 & \text{otherwise,} \end{cases} \quad k = 1, \ldots, n.$$

Note that, by construction, we have $\|w^*\|_1 = \|w^* + \delta\|_1 + \|\delta\|_1$. Also note that if w^* is sparse, so is δ.

Let us now denote by ϕ_τ the LASSO objective that we obtain upon replacing the optimum classifier w^* with its thresholded version $\tilde{w}(\tau) = w^* + \delta(\tau)$:

$$\phi_\tau \quad := \quad \frac{1}{2}\|X(w^* + \delta(\tau)) - y\|_2^2 + \lambda\|w^* + \delta(\tau)\|_1.$$

Since $w(\tau)$ is (trivially) feasible for the primal problem, we have $\phi_\tau \geq \phi(\lambda)$. On the other hand,

$$\begin{aligned} \phi_\tau \quad &= \quad \frac{1}{2}\|Xw^* - y\|_2^2 + \lambda\|w^* + \delta(\tau)\|_1 + \frac{1}{2}\|X\delta(\tau)\|_2^2 + \delta(\tau)^T X^T (Xw^* - y) \\ &\leq \quad \frac{1}{2}\|Xw^* - y\|_2^2 + \lambda\|w^*\|_1 + \frac{1}{2}\|X\delta(\tau)\|_2^2 + \delta(\tau)^T X^T (Xw^* - y). \end{aligned}$$

For a given $\alpha > 1$, the condition

$$\mathcal{C}(\tau) := \frac{1}{2}\|X\delta(\tau)\|_2 + \delta(\tau)^T X^T (Xw^* - y) \leq \kappa\phi^*, \;\; \kappa := \frac{1 + \alpha\epsilon}{1 + \epsilon} - 1 \geq 0, \tag{39}$$

allows to write

$$\phi(\lambda) \leq \phi_\tau \leq \; (1 + \alpha\epsilon)\phi(\lambda).$$

The condition (39) then implies that the thresholded classifier is sub-optimal, with relative accuracy $\alpha\epsilon$.

Our proposed thresholding rule is based on the condition (39). Precisely, we choose the parameter $\alpha > 0$, then we set the threshold level τ by solving, via line search, the largest threshold τ allowed by condition (39):

$$\tau_\alpha = \arg\max_{\tau \geq 0} \left\{ \tau \;:\; \|X\delta(\tau)\|_2 \leq \left(\sqrt{\frac{1 + \alpha\epsilon}{1 + \epsilon}} - 1 \right) \|Xw^* - y\|_2 \right\}.$$

The larger α is, the more elements the rule allows to set to zero; at the same time, the more degradation in the objective will be observed: precisely, the new relative accuracy is bounded by $\alpha\epsilon$. The rule also depends on the duality gap parameter ϵ. We refer to the thresholding rule as $\mathrm{TR}(\alpha)$ in the sequel. In practice, we observe that the value $\alpha = 2$ works well, in a sense made more precise below.

The complexity of the rule is $O(mn)$. More precisely, the optimal dual variable $\theta^* = Xw^* - y$ is returned by IPM-LASSO. The matrix $X\theta^* = X(X^T w^* - y)$ is computed once for all in $O(mn)$. We then sort the optimal vector w^* so that $|w^*_{(1)}| \leq \ldots \leq |w^*_{(n)}|$, and set $\tau = \tau_0 = |w^*_{(n)}|$, so that $\delta_k(\tau_0) = -w^*_k$ and $\tilde{w}_k(\tau_0) = 0$ for all $k = 1, \ldots, n$. The product $X\delta(\tau_0)$ is computed in $O(mn)$, while the product $\delta(\tau_0)^T(X^T\theta^*)$ is computed in $O(n)$. If the quantity $\mathcal{C}(\tau_0) = \frac{1}{2}\|X\delta(\tau_0)\|_2 + \delta(\tau_0)^T(X^T\theta^*)$ is greater than $\kappa\phi^\star$, then we set $\tau = \tau_1 = |w^*_{(n-1)}|$. We have $\delta_k(\tau_1) = \delta_k(\tau_0)$ for any $k \neq (n)$ and $\delta_{(n)}(\tau_1) = 0$. Therefore, $\mathcal{C}(\tau_1)$ can be deduced from $\mathcal{C}(\tau_0)$ in $O(n)$. We proceed by successively setting $\tau_k = |w^*_{(n-k)}|$ until we reach a threshold τ_k such that $\mathcal{C}(\tau_k) \leq \kappa\phi^*$.

Simulation study. We conducted a simple simulation study to evaluate our proposal and compare it to the KKT thresholding rule. Both methods were further compared to the results returned by the `glmnet` `R` package. The latter algorithm returns hard zeros in the classifier coefficients, and we have chosen the corresponding sparsity pattern as the "ground truth", which the IPM should recover.

We first experimented with synthetic data. We generated samples of the pair (X, y) for various values of (m, n). We present the results for $(m, n) = (5000, 2500)$ and $(m, n) = (100, 500)$. The number s of relevant features was set to $\min(m, n/2)$. Features were drawn from independent $\mathcal{N}(0, 1)$ distributions and y was computed as $y = X^T w + \xi$, where $\xi \sim \mathcal{N}(0, 0.2)$ and w is a vector of $\mathbb{R}^n$ with first s components equal to $0.1 + 1/s$ and remaining $n - s$ components set to 0. Because `glmnet` includes an unpenalized intercept while IPM method does not, both y and X were centered before applying either methods to make their results comparable.

Results are presented on Figures 6. First, the KKT thresholding rule was observed to be very chaotic when the duality gap was set to $\epsilon = 10^{-4}$ (we recall here that the default value for the duality gap in IPM `MATLAB` implementation is $\epsilon = 10^{-3}$), while it was way better when duality gap was set to $\epsilon = 10^{-8}$ (somehow justifying our choice of considering the sparsity pattern returned by `glmnet` as the ground truth). Therefore, for applications where computational time is not critical, running IPM method and applying KKT thresholding rule should yield appropriate results. However, when computational time matters, passing the duality gap from, say, 10^{-4} to 10^{-8}, is not a viable option. Next, regarding our proposal, we observed that it was significantly better than KKT thresholding rule when the duality gap was set to 10^{-4} and equivalent to KKT thresholding rule for a duality gap of 10^{-8}. Interestingly, setting $\alpha = 1.5$ in (39) generally enabled to achieved very good results for low values of λ, but lead to irregular results for higher values of λ (in the case $m = 100$, results were unstable for the whole range of λ values we considered). Overall, the choices $\alpha = 2$, 3 and 4 lead to acceptable results. A little irregularity remained with $\alpha = 2$ for high values of λ, but this choice of α performed the best for lower values of λ. As for choices $\alpha = 3$ and $\alpha = 4$, it is noteworthy that the results were all the better as the dimension n was low.

E.1 Real data examples

We also applied our proposal and compared it to KKT rule (38) on real data sets arising in text classification. More precisely, we used the New York Times headlines data set presented in the Numerical results Section. For illustration, we present here results we obtained for the topic "China" and the year 1985. We successively ran IPM-LASSO method with duality gap set to 10^{-4} and 10^{-8} and compare the number of active features returned after applying KKT thresholding rule (38) and TR (1.5), TR (2), TR (3) and TR (4). Results are presented on Figure 7. Because we could not applied `glmnet` on this data set, the ground truth was considered as the result of KKT rule, when applied to the model returned by IPM-LASSO ran with duality gap set to 10^{-10}. Applying KKT

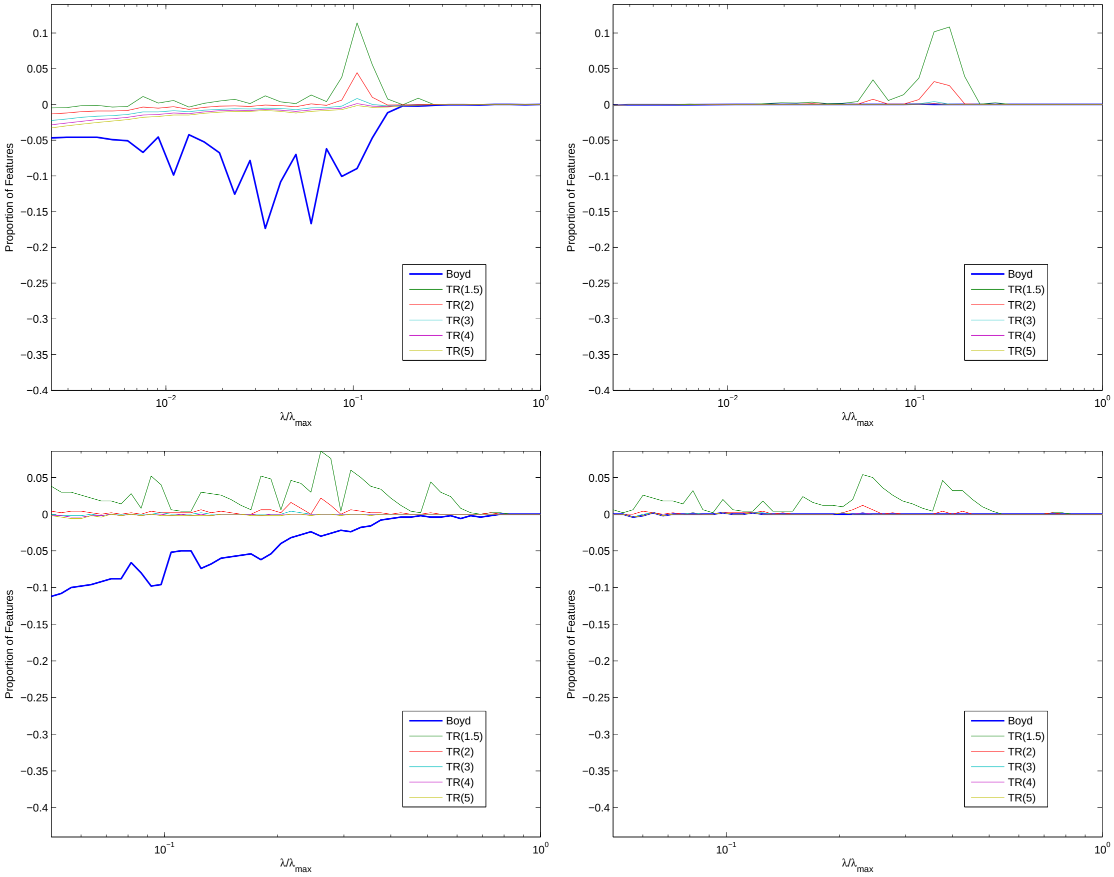


Figure 6: Comparison of several thresholding rules on synthetic data: the case $m = 5000$, $n = 100$ (*top panel*) and $m = 100$, $n = 500$ (*bottom panel*) with duality gap in IPM method set to (*i*) 10^{-4} (*left panel*) and (*iii*) 10^{-8} (*right panel*). The curves represent the differences between the number of active features returned after each thresholding method and the one returned by `glmnet` (this difference is further divided by the total number of features n). The graphs present the results attached to six thresholding rules: the one proposed by Koh et al. (2007) and five versions of our proposal, corresponding to setting α in (39) to 1.5, 2, 3, 4 and 5 respectively. Overall, these results suggest that by setting $\alpha \in (2, 5)$, our rule is less sensitive to the value of the duality gap parameter in IPM-LASSO than is the rule proposed by Koh et al. (2007).

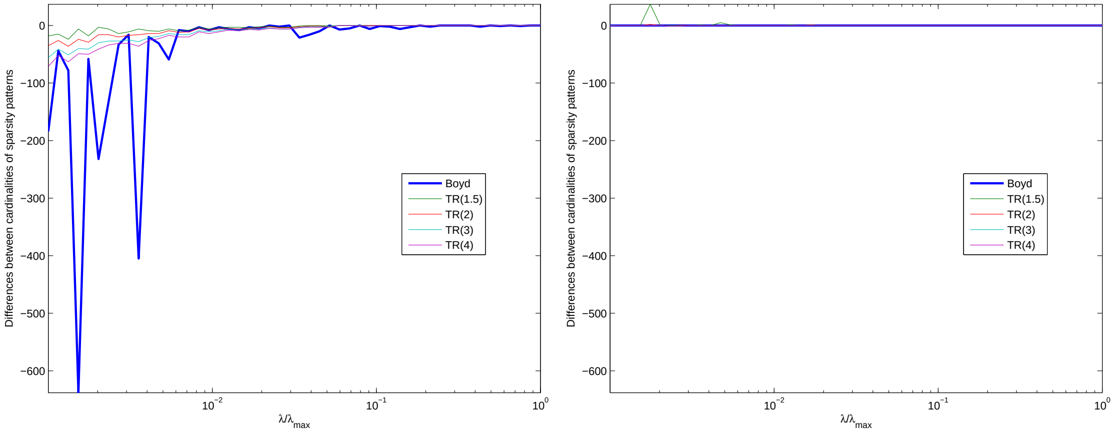


Figure 7: Comparison of several thresholding rules on the NYT headlines data set for the topic "China" and year 1985. Duality gap in IPM-LASSO was successively set to 10^{-4} (*left panel*) and 10^{-8} (*right panel*). The curves represent the differences between the number of active features returned after each thresholding method and the one returned by the KKT rule when duality gap was set to 10^{-10}. The graphs present the results attached to five thresholding rules: the KKT rule and four versions of our rule, corresponding to setting α in (39) to 1.5, 2, 3 and 4 respectively. Results obtained following our proposal appear to be less sensitive to the value of the duality gap used in IPM-LASSO. For instance, for the value $\lambda = \lambda_{\max}/1000$, the KKT rule returns 1758 active feature when the duality gap is set to 10^{-4} while it returns 2357 features for a duality gap of 10^{-8}.

rule on the model built with a duality gap of 10^{-4} lead to very misleading results again, especially for low values of λ. In this very high-dimensional setting ($n = 38377$ here), our rule generally resulted in a slight "underestimation" of the true number of active features for the lowest values of λ when the duality gap was set to 10^{-4}. This suggests that the “optimal” α for our rule might depend on both n and λ when the duality gap is not small enough. However, we still observed that our proposal significantly improved upon KKT rule when the duality gap was set to 10^{-4}.

References

S.R. Becker, E.J. Candes, and M. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Stanford University Technical Report*, 2010.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521833787.

S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43:129, 2001.

David L. Donoho and Yaakov Tsaig. Fast solution of l_1-norm minimization problems when the solution may be sparse. *IEEE Trans. Inform. Theory*, 54(11):4789–4812, 2008.

Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression (with discussion). *Ann. Statist.*, 32:407–499, 2004.

Jianqing Fan and Jinchi Lv. Sure independence screening for ultrahigh dimensional feature space. *J. Roy. Statist. Soc. Ser. B*, 70(5):849–911, 2008.

Jianqing Fan and Jinchi Lv. A selective overview of variable selection in high dimensional feature space. *Statist. Sinica*, 20:101–148, 2010.

George Forman. An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.*, 3:1289–1305, 2003.

A. Frank and A. Asuncion. UCI machine learning repository, 2010. URL `http://archive.ics.uci.edu/ml`.

Jerome Friedman, Trevor Hastie, Holger Höfling, and Robert Tibshirani. Pathwise coordinate optimization. *Ann. Appl. Statist.*, 1(2):302–332, 2007.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010. URL `http://www.jstatsoft.org/v33/i01/`.

Brian Gawalt, Jinzhu Jia, Luke Miratrix, Laurent El Ghaoui, Bin Yu, and Sophie Clavier. Discovering word associations in news media via feature selection and sparse classification. In *MIR '10: Proceedings of the international conference on Multimedia information retrieval*, pages 211–220, 2010.

Seung-Jean Kim, Kwangmoo Koh, Michael Lustig, Stephen Boyd, and Dimitry Gorinevsky. An interior-point method for large-scale l_1-regularized least squares. *IEEE J. Select. Top. Sign. Process.*, 1(4):606–617, 2007.

Kwangmoo Koh, Seung-Jean Kim, and Stephen Boyd. An interior-point method for large-scale l_1-regularized logistic regression. *JMLR*, 8:1519–1555, 2007.

Mee Young Park and Trevor Hastie. L_1-regularization path algorithm for generalized linear models. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 69(4):659–677, 2007.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 0035-9246.

A. Yang, A. Ganesh, Z. Zhou, S. Sastry, and Y. Ma. Fast l1-minimization algorithms and an application in robust face recognition: a review. *University of California at Berkeley Technical report UCB/EECS-2010-13*, 2010.