London
Business
School

LBS Research Online

Sparse Classification: A Scalable Discrete Optimization Perspective

Dimitris Bertsimas

Sloan School of Management and Operations Research Center, MIT, Cambridge, MA, USA, dbertsim@mit.edu

Jean Pauphilet

Operations Research Center, MIT, Cambridge, MA, USA, jpauph@mit.edu

Bart Van Parys

Operations Research Center, Sloan School of Management, MIT, Cambridge, MA, USA, vanparys@mit.edu

We formulate the sparse classification problem of n samples with p features as a binary convex optimization problem and propose a outer-approximation algorithm to solve it exactly. For sparse logistic regression and sparse SVM, our algorithm finds optimal solutions for n and p in the $10,000$s within minutes. On synthetic data our algorithm achieves perfect support recovery in the large sample regime. Namely, there exists an n_0 such that the algorithm takes a long time to find an optimal solution and does not recover the correct support for $n < n_0$, while for $n \geqslant n_0$, the algorithm quickly detects all the true features, and does not return any false features. In contrast, while Lasso accurately detects all the true features, it persistently returns incorrect features, even as the number of observations increases. Consequently, on numerous real-world experiments, our outer-approximation algorithms returns sparser classifiers while achieving similar predictive accuracy as Lasso. To support our observations, we analyze conditions on the sample size needed to ensure full support recovery in classification. For k-sparse classification, and under some assumptions on the data generating process, we prove that information-theoretic limitations impose $n_0 < C \left(2 + \sigma^2\right) k \log(p - k)$, for some constant $C > 0$.

Key words: Sparse classification; Binary convex optimization; Support recovery

Subject classifications: Programming: integer, nonlinear; Statistics: data analysis

1. Introduction

Sparse classification is a central problem in machine learning as it leads to more interpretable models. Given data $\{(\boldsymbol{x}^{(i)}, y_i)\}_{i=1,\dots,n}$ with $y_i \in \{-1, 1\}$ and $\boldsymbol{x}^{(i)} \in \mathbb{R}^p$, we aim at computing an estimator $\boldsymbol{w}$ which minimizes an empirical loss ℓ subject to the constraint that its number of

nonzero entries does not exceed k:

$$\min_{\boldsymbol{w}\in\mathbb{R}^p, b\in\mathbb{R}} \sum_{i=1}^n \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) \text{ s.t. } \|\boldsymbol{w}\|_0 \leqslant k. \tag{1}$$

Problem (1) is an NP-hard optimization problem (Natarajan 1995). Thus, much of the literature has focused on heuristic proxies and replaced the 0 pseudo-norm with so-called sparsity-inducing convex norms (Bach et al. 2012). Even though regularization enforces robustness more than sparsity (Bertsimas and Copenhaver 2018), the L_1-penalty formulation

$$\min_{\boldsymbol{w}\in\mathbb{R}^p, b\in\mathbb{R}} \sum_{i=1}^n \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) + \lambda \|\boldsymbol{w}\|_1, \tag{2}$$

known as Lasso (Tibshirani 1996) is abundantly used in practice. Efficient numerical algorithms exist (Friedman et al. 2010), off-the-shelf implementations are publicly available (Friedman et al. 2013) and recovery of the true sparsity is theoretically guaranteed under some assumptions on the data. In regression problems with i.i.d. Gaussian measurements for instance, Wainwright (2009b) proved that Lasso recovers the k correct features with high probability (w.h.p. in short) for $n > (2k + \sigma^2) \log p$ where σ^2 is the variance of the noise, a phenomenon they refer to as phase transition in accuracy. On the other hand, recent works (Fan et al. 2010, Bühlmann 2011, Su et al. 2017) highlighted the difficulty for L_1-regularized estimators to select correct features without making false discoveries, considering Lasso as a good feature screening but a poor feature selection procedure. The innate shortcomings of the convex heuristics such as Lasso are by now so well documented that in recent years a lot of attention has been diverted to sparsity-inducing nonconvex regularization methods. Such penalization formulations consider

$$\min_{\boldsymbol{w}\in\mathbb{R}^p, b\in\mathbb{R}} \sum_{i=1}^n \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) + \lambda \cdot R(\boldsymbol{w}) \tag{3}$$

where R is a nonconvex penalty function with a unique minimizer at zero. Frank and Friedman (1993) proposed for instance the use of an L_p pseudo-norm with $0 < p < 1$ as a continuous approximation to the 0 pseudo-norm for least-squares regression problems. Nonconvex penalty functions are very actively studied nowadays as they promise stronger statistical guarantees when compared

to their L_1-based convex counterparts. Fan and Li (2001) propose a smoothly clipped absolute deviation (SCAD) penalty while Zhang (2010a) advance their minimax concave penalty (MCP) for general convex loss functions. Both penalties and their associated regression models enjoy a so-called "oracle property", that is, in the asymptotic sense, they perform as well as if the support of the nonzero coefficients was known in advance. As the formulation (3) is non-convex numerical challenges can be expected. Indeed, Chen et al. (2019) show that similar to (1) the nonconvex formulation (3) is unfortunately NP-hard to solve (even approximately). Nevertheless, many empirical studies (Fan and Li 2001, Hunter and Li 2005, Zou and Li 2008, Breheny and Huang 2011, Mazumder et al. 2011) have shown that despite the nonconvexity of these formulations stationary points can be computed efficiently based on gradient-based optimization methods and consistently return better regressors than their L_1-based counterparts do. In an impressive line of work Loh and Wainwright (2015), Loh et al. (2017) establish that in fact all stationary points in (3) enjoy support recovery consistency even when the usual incoherence conditions required by L_1-based methods fail to hold. Adaptive approximation approaches which attempt to address the nonconvex formulation (3) by solving a sequence of convex relaxations are discussed by Zhang (2010b), Zhang et al. (2013), Fan et al. (2018) and likewise result in empirically strong sparse regressors.

New research in numerical algorithms for solving the exact sparse formulation (1) has also flourished and demonstrated significant improvement on existing heuristics. Bertsimas et al. (2016), Bertsimas and King (2017) made use of recent advances in mixed-integer optimization to solve sparse linear and logistic regression problems. Pilanci et al. (2015) applied a Lagrangian relaxation and random rounding procedure for linear regression and provide sufficient conditions for support recovery with their method. Hazimeh and Mazumder (2020) developed a cyclic coordinate descent strategy combined with local search to efficiently find local optima of the L_0-penalized ordinary least square regression problem, later extended to other loss functions (Dedieu et al. 2021). Recently, sparse linear regression for dimensions n and p in $100,000$s was exactly solved for the first time, using a cutting-plane algorithm (Bertsimas et al. 2020b). Their method demonstrates a clear phase

transition in accuracy as the sample size n increases and requires less data than Lasso to achieve full recovery. Simultaneously, they observed a phase transition in false discovery, that is, the number of incorrect features selected, and in computational time, which is unique to their method: they exhibited a threshold n_0 for which for $n < n_0$ their algorithm takes a long time to find the optimal solution and it is does not recover the correct support, while for $n \geqslant n_0$, the algorithm is very fast and accurately detects all the true features, but does not return any false features. We refer to Bertsimas et al. (2020a) for a comprehensive empirical evaluation of the relative merits of L_1-based, cardinality-constrained, and non-convex formulations in accurately selecting the true support.

Besides algorithm-specific performance, any support recovery algorithm faces information-theoretic limitations as well (Wainwright 2009a, Wang et al. 2010). In regression, recent work (Gamarnik and Zadik 2017) indeed proved the existence of a sharp information-theoretic threshold $n^\star$: If $n < n^\star$, exact support recovery by any algorithm is impossible, while it is theoretically achievable for $n > n^\star$. Such results call for further research in learning algorithms in the regime $n^\star < n < (2k + \sigma^2) \log p$ where Lasso fails but full recovery is achievable in principle.

Regarding accuracy, such phase transition phenomena are actually not specific to regression problems but are observed in many data analysis and signal processing contexts (Donoho and Stodden 2006, Donoho and Tanner 2009). Surprisingly, little if no work focused on classification problems specifically. Guarantees in terms of ℓ_2 error have been obtained in the so-called 1-bit compressed sensing setting (Boufounos and Baraniuk 2008, Gupta et al. 2010, Plan and Vershynin 2013a, Jacques et al. 2013) but they do not precisely address the question of support recovery. In recent work, Scarlett and Cevher (2017) offer a comprehensive treatment of information theoretic limitations in support recovery for both linear and 1-bit compressed sensing, in some regimes of noise and sparsity. Sparse classification in itself has mainly been regarded as a feature selection problem (Dash and Liu 1997) and greedy procedures such as Recursive Feature Elimination (Guyon et al. 2002) have shown the most successful.

In this paper, we formulate exact sparse classification as a binary convex optimization problem, propose a tractable outer-approximation algorithm and a stochastic variant to solve it in high

dimensions. To provide insights on the algorithm's performance and behavior, we also derive an information-theoretic sufficient condition for support recovery in classification that complements existing results from the 1-bit compressed sensing literature.

Contributions The contributions of the present paper can be summarized as follows:

1. Based on duality results for regularized classification, we formulate the exact sparse classification problem as a binary convex optimization problem and propose a tractable outer-approximation algorithm to solve it. Our approach generalizes the one in Bertsimas et al. (2020b), who address linear regression for which a closed-form solution exists and make extensive use of this closed-form solution. Our framework, however, extends to cases where a closed-form solution is not available and includes, in addition to linear regression, logistic regression and 1- or 2-norm SVM. We also propose a stochastic constraint generating process to improve scalability of the outer-approximation algorithm with respect to the number of samples n.

2. We demonstrate the tractability and relevance of our algorithm in solving large binary classification problems with logistic and Hinge loss. Among others, we solve a real-world gene-selection problem with $n = 1,145$, $p = 14,858$ and select four to ten times fewer genes than the L_1 heuristic with little compromise on the predictive power. On synthetic data, our algorithm can scale to data sets with up to $p = 50,000$ features and prove optimality in less than a minute for low sparsity regimes ($k = 5$). Our present algorithm and the concurrent paper of Dedieu et al. (2021) are, to the best of our knowledge, the only methods currently available that solve sparse classification problems to provable optimality in such high dimensions within minutes. We believe, however, that the tools developed in the present paper may be more versatile and broadly applicable than the tailored "integrality generation" technique of Dedieu et al. (2021). Finally, we observe that our stochastic constraint generating process reduces computational time by a factor $2 - 10$ compared with the standard outer-approximation procedure for the hardest instances, namely instances where the number of samples n is not large enough for Lasso to achieve perfect recovery.

3. We demonstrate empirically that cardinality-constrained estimators asymptotically achieve perfect support recovery on synthetic data: As n increases, the method accurately detects all

the true features, just like Lasso, but does not return any false features, whereas Lasso does. In addition, we observe that computational time of our algorithm decreases as more data is available. We exhibit a smooth transition towards perfect support recovery in classification settings, whereas it is empirically observed (Donoho and Stodden 2006, Bertsimas et al. 2020b) and theoretically defined (Wainwright 2009a, Gamarnik and Zadik 2017) as a sharp phase transition in the case of linear regression.

4. Intrigued by this smoother empirical behavior, we show that there exists a threshold n_0 on the number of samples such that if $n > n_0$, the underlying truth $\boldsymbol{w}^\star$ minimizes the empirical error with high probability. Assuming $p \geqslant 2k$ and data is generated by $y_i = \text{sign}\left(\boldsymbol{w}^{\star\top}\boldsymbol{x}^{(i)} + \varepsilon_i\right)$, with $\boldsymbol{x}^{(i)}$ i.i.d. Gaussian, $\boldsymbol{w}^\star \in \{0,1\}^p$, $\text{supp}(\boldsymbol{w}^\star) = k$ and $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, we prove that $n_0 < C\left(2 + \sigma^2\right)k\log(p-k)$, for some constant $C > 0$ (Theorem 3). Our information-theoretic sufficient condition on support recovery parallels the one obtained by Wainwright (2009a) in regression settings, although discreteness of the outputs substantially modify the analysis and the scaling of the bound. It also scales as $k\log p$, thus agreeing with the sufficient conditions required by non-convex formulations (Loh et al. 2017), although their generative process is different and their result does not explicitly capture the impact of noise. In recent work, Scarlett and Cevher (2017) performed a similar analysis of support recovery in 1-bit compressed sensing. For low signal-to-noise ratios (SNR), they exhibit matching necessary and sufficient conditions, which suggests the existence of a phase transition. In high signal-to-noise regimes, however, they proved necessary conditions only. Our result is novel, in that it is not only valid for specific sparsity and noise regimes, but for all values of k and σ (see Table 1 for a summary of their results and ours). In particular, our sufficient condition holds when the sparsity k scales linearly in the dimension p and the signal-to-noise ratio is high. In this regime, there is a $\sqrt{\log p}$ factor between our bound and the necessary condition from Scarlett and Cevher (2017). As represented in Figure 1, there is an intermediate sample size regime where support recovery is neither provably impossible nor provably achievable, and could explained the smooth phase transition observed. Such an observation is made possible by the combination of both our results. Of course, this observation only suggests that a lack of phase transition is plausible. So is the existence of one and future work might improve on these bounds and close this gap.

Table 1   Summary of the necessary and sufficient conditions provided in Scarlett and Cevher (2017), as compared to the sufficient condition we provide in Theorem 3, depending on the sparsity and signal-to-noise ratio (SNR) regime.

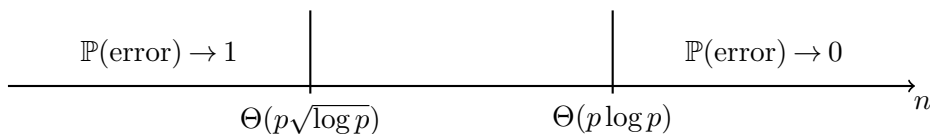| Parameters	$k = \Theta(1)$, Low SNR	$k = \Theta(p)$, High SNR
	Necessary condition for $\mathbb{P}(\text{error}) \nrightarrow 1$	
(Scarlett and Cevher 2017)	$\pi \sigma^2 \log p$	$\Omega(p\sqrt{\log p})$
	Sufficient condition for $\mathbb{P}(\text{error}) \to 0$	
(Scarlett and Cevher 2017)	$\pi \sigma^2 \log p$	$-$
Theorem 3	$(2 + \sigma^2)k \log(p - k)$	

Figure 1   Summary of known necessary (Scarlett and Cevher 2017, Cor. 4) and sufficient (see Theorem 3) conditions on the sample size n to achieve perfect support recovery in classification, when the sparsity k scales linearly in the dimension p and the signal-to-noise ratio is high. Thresholds are given up to a multiplicative constant.

Structure   We derive the binary convex optimization formulation for exact sparse classification, present an outer-approximation algorithm to solve it, and propose a stochastic constraint generating process in Section 2. We evaluate the performance of our method, both in terms of tractability and support recovery ability, in Section 3. Finally, we prove information-theoretic sufficient condition for support recovery in Section 4.

Notations   We reserve bold characters ($\boldsymbol{x}$) to denote vectors. We denote by $\mathbf{e}$ the vector whose components are equal to one. If not specified, its dimension should be inferred from the context. The set $\mathcal{S}_k^p$ denotes the set

$$\mathcal{S}_k^p := \left\{ \boldsymbol{s} \in \{0,1\}^p \ : \ \mathbf{e}^\top \boldsymbol{s} \leqslant k \right\},$$

which contains all binary vectors $\boldsymbol{s}$ selecting k components from p possibilities. Assume $\boldsymbol{y} \in \mathbb{R}^p$ is a vector and $\boldsymbol{s} \in \mathcal{S}_k^p$, then $\boldsymbol{y_s}$ denotes the sub-vector or sub-collection of y_j's where $s_j = 1$. We use

$\|\boldsymbol{x}\|_0$ to denote the number of elements of a vector which are nonzero. We use bold capitalized letters ($\boldsymbol{X}$) for matrices. For a matrix $\boldsymbol{X}$, $\boldsymbol{X}_j$ denotes the jth column of $\boldsymbol{X}$, while $\boldsymbol{X_s}$ for $\boldsymbol{s} \in \mathcal{S}_k^p$ is the submatrix obtained from the k columns of $\boldsymbol{X}$ selected by $\boldsymbol{s}$.

2. Dual Framework

In this section, we use duality results to formulate a regularized version of the sparse classification problem (1) as a binary convex optimization problem and propose an outer-approximation approach to solve it efficiently.

2.1. Regularized Classification

We first introduce the basic notation and recall some well-known properties for the case of non-sparse classification. Two very popular classification methods in machine learning are logistic regression and Support Vector Machine (SVM). Despite different motivations and underlying intuition, both methods lead to a similar formulation which can be addressed under the unifying lens of regularized classification:

$$\min_{\boldsymbol{w}\in\mathbb{R}^p, b\in\mathbb{R}} \sum_{i=1}^n \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) + \frac{1}{2\gamma}\|\boldsymbol{w}\|_2^2, \tag{4}$$

where ℓ is an appropriate loss function and γ a regularization coefficient.

In the logistic regression framework, the loss function is the logistic loss

$$\ell(y, u) = \log\left(1 + e^{-yu}\right),$$

and the objective function can be interpreted as the negative log-likelihood of the data plus a regularization term, which ensures strict convexity of the objective and existence of an optimal solution.

In the SVM framework, the loss function ℓ is the hinge loss:

$$\ell(y, u) = \max(0, 1 - yu).$$

Under proper normalization assumptions, the square norm $\|\boldsymbol{w}\|_2^2$ relates to the notion of margin, which characterizes the robustness of the separating hyperplane $\{\boldsymbol{x} : \boldsymbol{w}^\top \boldsymbol{x} + b = 0\}$, while the loss

part penalizes the data points which do no satisfy $y_i(\boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) \geqslant 1$, that is points which are misclassified or lie within the margin (Vapnik 1998).

In addition, this general formulation (4) accounts for any loss function used in classification (e.g. 2-norm SVM) or even in regression problems. Throughout the paper we make the following assumption:

ASSUMPTION 1. *The loss function $\ell(y, \cdot)$ is convex for $y \in \{-1, 1\}$.*

Note that both the logistic and Hinge loss functions satisfy the convexity assumption 1. In classification, deeper results and insights can typically be obtained by adopting a dual perspective. Denoting $\boldsymbol{X} = (\boldsymbol{x}^{(i)^\top})_{i=1,\ldots,n} \in \mathbb{R}^{n \times p}$ the design matrix, we have:

THEOREM 1. *Under Assumption 1, strong duality holds for problem* (4) *and its dual is*

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^n : e^\top \boldsymbol{\alpha} = 0} - \sum_{i=1}^{n} \hat{\ell}(y_i, \alpha_i) - \frac{\gamma}{2} \boldsymbol{\alpha}^\top \boldsymbol{X} \boldsymbol{X}^\top \boldsymbol{\alpha} \,, \tag{5}$$

where $\hat{\ell}(y, \alpha) := \max_{u \in \mathbb{R}} u\alpha - \ell(y, u)$ is the Fenchel conjugate of the loss function ℓ (see Boyd and Vandenberghe 2004, Chap. 3.3).

Table 2 summarizes some popular loss functions in classification and their corresponding Fenchel conjugates.

Table 2   Examples of loss functions and their corresponding Fenchel conjugates, as defined in Theorem 1 (Bach 2009).

Method	Loss $\ell(y, u)$	Fenchel conjugate $\hat{\ell}(y, \alpha)$
Logistic loss	$\log\left(1 + e^{-yu}\right)$	$\begin{cases} (1 + y\alpha)\log(1 + y\alpha) - y\alpha\log(-y\alpha), & \text{if } y\boldsymbol{\alpha} \in [-1, 0], \\ +\infty, & \text{otherwise.} \end{cases}$
1-norm SVM	$\max(0, 1 - yu)$	$\begin{cases} y\alpha, & \text{if } y\boldsymbol{\alpha} \in [-1, 0], \\ +\infty, & \text{otherwise.} \end{cases}$
2-norm SVM	$\frac{1}{2}\max(0, 1 - yu)^2$	$\begin{cases} \frac{1}{2}\alpha^2 + y\alpha, & \text{if } y\alpha \leqslant 0, \\ +\infty, & \text{otherwise.} \end{cases}$

F or regularized classification, we have that

$$\min_{\boldsymbol{w},b} \sum_{i=1}^{n} \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) + \frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 = \min_{\boldsymbol{w},b,\boldsymbol{z}} \sum_{i=1}^{n} \ell(y_i, z_i) + \frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2$$

$$\text{s.t. } z_i = \boldsymbol{z}^\top \boldsymbol{x}^{(i)} + b.$$

By Assumption 1, the objective is convex, the optimization set is convex and Slater's conditions hold (Boyd and Vandenberghe 2004). Hence, strong duality holds and the primal is equivalent to the dual problem. To derive the dual formulation, we introduce Lagrange multipliers α_i associated with the equality constraints:

$$\min_{\boldsymbol{w},b,\boldsymbol{z}} \sum_{i=1}^{n} \ell(y_i, z_i) + \frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 \text{ s.t. } z_i = \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b$$

$$= \min_{\boldsymbol{w},b,\boldsymbol{z}} \sum_{i=1}^{n} \ell(y_i, z_i) + \frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 + \max_{\boldsymbol{\alpha} \in \mathbb{R}^n} \sum_{i=1}^{n} \alpha_i (\boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b - z_i)$$

$$= \min_{\boldsymbol{w},b,\boldsymbol{z}} \max_{\boldsymbol{\alpha}} \left(\sum_{i=1}^{n} \ell(y_i, z_i) - \alpha_i z_i \right) + \left(\frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 + \boldsymbol{w}^\top \left[\sum_i \alpha_i \boldsymbol{x}^{(i)} \right] \right) + b \mathbf{e}^\top \boldsymbol{\alpha},$$

$$= \max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \min_{z_i} (\ell(y_i, z_i) - \alpha_i z_i) + \min_{\boldsymbol{w}} \left(\frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 + \boldsymbol{w}^\top \boldsymbol{X}^\top \boldsymbol{\alpha} \right) + \min_b b \mathbf{e}^\top \boldsymbol{\alpha}.$$

Let us consider the three inner minimization problems separately. First,

$$\min_{z_i} (\ell(y_i, z_i) - \alpha_i z_i) = -\max_{z_i} (\alpha_i z_i - \ell(y_i, z_i)) = -\hat{\ell}(y_i, \alpha_i).$$

Then, $\frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 + \boldsymbol{w}^\top \boldsymbol{X}^\top \boldsymbol{\alpha}$ is minimized at $\boldsymbol{w}^\star$ satisfying: $\frac{1}{\gamma} \boldsymbol{w}^\star + \boldsymbol{X}^\top \boldsymbol{\alpha} = 0$. Hence

$$\min_w \left(\frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 + \boldsymbol{w}^\top \boldsymbol{X}^\top \boldsymbol{\alpha} \right) = -\frac{1}{2\gamma} \|\boldsymbol{w}^\star\|_2^2 = -\frac{\gamma}{2} \boldsymbol{\alpha}^\top \boldsymbol{X} \boldsymbol{X}^\top \boldsymbol{\alpha}.$$

Finally, $\min_b b \mathbf{e}^\top \boldsymbol{\alpha}$ is bounded if and only if $\mathbf{e}^\top \boldsymbol{\alpha} = 0$, thus we obtain (5).   Q.E.D.

The derivation of the dual (5) reveals that the optimal primal variables $\boldsymbol{w}^\star$ can be recovered from the dual variables $\boldsymbol{\alpha}^\star$ via the relationship $\boldsymbol{w}^\star = -\gamma \boldsymbol{X}^\top \boldsymbol{\alpha}^\star$. In other words, $\boldsymbol{w}^\star$ is a linear combination of the data points $\boldsymbol{X}$. Such an observation has historically led to the intuition that $\boldsymbol{w}^\star$ was *supported* by some observed vectors $\boldsymbol{x}^{(i)}$ and the name Support Vector Machine was coined (Cortes and Vapnik 1995). Conversely, $\boldsymbol{\alpha}^\star$ relates to the primal variables $\boldsymbol{w}^\star$ via the relationship

$\boldsymbol{\alpha}_i^\star \in \partial\ell(y_i, \boldsymbol{w}^{\star\top}\boldsymbol{x}^{(i)})$, where $\partial\ell(y_i, \boldsymbol{x}^{(i)\top}\boldsymbol{w}^\star)$ denotes the sub-differential of the loss function $\ell(y_i, \cdot)$ evaluated at $\boldsymbol{w}^{\star\top}\boldsymbol{x}^{(i)}$. If ℓ is differentiable, like the logistic loss, this relationship uniquely defines $\boldsymbol{\alpha}_i^\star$.

Moreover, the dual point of view opens the door to non-linear classification using kernels (Scholkopf and Smola 2001). The positive semi-definite matrix $\boldsymbol{X}\boldsymbol{X}^\top$, often referred to as the kernel or Gram matrix, is central in the dual problem (5) and could be replaced by any kernel matrix $\boldsymbol{K}$ whose entries K_{ij} encode some measure of similarity between inputs $\boldsymbol{x}^{(i)}$ and $\boldsymbol{x}^{(j)}$.

Numerical algorithms There is a rich literature on numerical algorithms for solving either the primal (4) or the dual (5) formulation for the regularized classification problem in the case of logistic regression and SVM. Gradient descent or Newton-Raphson methods are well-suited when the loss function is smooth. In addition, in the case where the dual problem is constrained to $\mathbf{e}^\top\boldsymbol{\alpha} = 0$, particular step size rules (Calamai and Moré 1987, Bertsekas 1982) or trust regions (Lin et al. 2008) can be implemented to cope with such linear constraints. When the loss function is not continuously differentiable, sub-gradient descent as proposed in the Pegasos algorithm (Shalev-Shwartz et al. 2011) provides an efficient optimization procedure. Within the machine learning community, coordinate descent methods have also received a lot of attention recently, especially in the context of regularized prediction, because of their ability to compute a whole regularization path at a low computational cost (Friedman et al. 2010). For coordinate descent algorithms specific to the regularized classification problem we address in this paper, we refer to (Hsieh et al. 2008, Yu et al. 2011, Keerthi et al. 2005).

REMARK 1. Theorem 1 does not hold when the loss is not convex, for instance in the case of the empirical misclassification rate

$$\min_{\boldsymbol{w}, b} \sum_{i=1}^{n} \mathbf{1}_{y_i(\boldsymbol{w}^\top\boldsymbol{x}^{(i)} + b) < 0} + \frac{1}{2\gamma}\|\boldsymbol{w}\|_2^2. \tag{6}$$

Indeed, strong duality does not hold. The objective value is clearly finite and nonnegative but since $\hat{\ell}(y, \alpha) = +\infty$, the dual problem has cost $-\infty$. In general, if the loss function ℓ is not convex, the dual problem (5) is not equivalent to (4) but provides a valid lower bound.

2.2. Dual Approach to Sparse Classification

Sparsity is a highly desirable property for statistical estimators, especially in high-dimensional regimes $(p \gg n)$ such as the ones encountered in biological applications, where interpretability is crucial. A natural way to induce sparsity is to add a constraint on the number of nonzero coefficients of $\boldsymbol{w}$ and solve:

$$\min_{\boldsymbol{w} \in \mathbb{R}^p, b \in \mathbb{R}} \sum_{i=1}^{n} \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) + \frac{1}{2\gamma} \|\boldsymbol{w}\|_2^2 \ \text{ s.t. } \ \|\boldsymbol{w}\|_0 \leqslant k. \tag{7}$$

Actually, (7) can be expressed as a convex binary optimization problem as stated in the following theorem:

THEOREM 2. *Problem* (7) *is equivalent to*

$$\min_{\boldsymbol{s} \in \mathcal{S}_k^p} c(\boldsymbol{s}), \tag{8}$$

where for any $\boldsymbol{s} \in \{0,1\}^p$ we have $c(\boldsymbol{s}) := \max_{\boldsymbol{\alpha} \in \mathbb{R}^n : e^\top \boldsymbol{\alpha} = 0} f(\boldsymbol{\alpha}, \boldsymbol{s})$ with

$$f(\boldsymbol{\alpha}, \boldsymbol{s}) := -\sum_{i=1}^{n} \hat{\ell}(y_i, \alpha_i) - \frac{\gamma}{2} \sum_{j=1}^{n} s_j \boldsymbol{\alpha}^\top \boldsymbol{X}_j \boldsymbol{X}_j^\top \boldsymbol{\alpha}. \tag{9}$$

In particular, $c(\boldsymbol{s})$ is convex over $[0,1]^p$.

W e introduce an additional binary variable $\boldsymbol{s} \in \{0,1\}^p$ encoding for the support of the sparse classifier $\boldsymbol{w}$. With these notations

$$\boldsymbol{w}^\top \boldsymbol{x}^{(i)} = \sum_{j=1}^{p} w_j x_j^{(i)} = \sum_{j:s_j=1} w_j x_j^{(i)} = \boldsymbol{w}_{\boldsymbol{s}}^\top \boldsymbol{x}_{\boldsymbol{s}}^{(i)},$$

$$\|\boldsymbol{w}\|_2^2 = \sum_{j=1}^{p} w_j^2 = \sum_{j:s_j=1} w_j^2 = \|\boldsymbol{w}_{\boldsymbol{s}}\|_2^2,$$

the cardinality constraint on $\boldsymbol{w}$ yields a linear constraint on $\boldsymbol{s}$

$$\boldsymbol{s}^\top \mathbf{e} \leqslant k,$$

and (7) can be equivalently written as

$$\min_{\boldsymbol{s} \in \mathcal{S}_k^p} \min_{w,b} \sum_{i=1}^{n} \ell(y_i, \boldsymbol{w}_{\boldsymbol{s}}^\top \boldsymbol{x}_{\boldsymbol{s}}^{(i)} + b) + \frac{1}{2\gamma} \|\boldsymbol{w}_{\boldsymbol{s}}\|_2^2.$$

Denoting $c(\boldsymbol{s})$ the inner minimization problem, we end up solving the pure binary to-be-proved-convex optimization problem

$$\min_{\boldsymbol{s} \in \mathcal{S}_k^p} c(\boldsymbol{s}).$$

In addition, $c(\boldsymbol{s}) = \min_{\boldsymbol{w}, b} \sum_{i=1}^n \ell(y_i, \boldsymbol{w}_{\boldsymbol{s}}^\top \boldsymbol{x}_{\boldsymbol{s}}^{(i)} + b) + \frac{1}{2\gamma} \|\boldsymbol{w}_{\boldsymbol{s}}\|_2^2$ is an unconstrained regularized classification problem based on the features selected by $\boldsymbol{s}$ only. Hence, Theorem 1 applies and

$$c(\boldsymbol{s}) = \max_{\boldsymbol{\alpha} \in \mathbb{R}^n} \sum_{i=1}^n -\hat{\ell}(y_i, \alpha_i) - \frac{\gamma}{2} \boldsymbol{\alpha}^\top \boldsymbol{X}_{\boldsymbol{s}} \boldsymbol{X}_{\boldsymbol{s}}^\top \boldsymbol{\alpha} \ \text{ s.t. } \ \mathbf{e}^\top \boldsymbol{\alpha} = 0.$$

Since $\boldsymbol{\alpha}^\top \boldsymbol{X}_{\boldsymbol{s}} \boldsymbol{X}_{\boldsymbol{s}}^\top \boldsymbol{\alpha} = \sum_{j:s_j=1} \boldsymbol{\alpha}^\top \boldsymbol{X}_j \boldsymbol{X}_j^\top \boldsymbol{\alpha} = \sum_{j=1}^p s_j \boldsymbol{\alpha}^\top \boldsymbol{X}_j \boldsymbol{X}_j^\top \boldsymbol{\alpha}$, we obtain the desired formulation.

Finally, let us denote

$$f(\alpha, s) := -\sum_{i=1}^n \hat{\ell}(y_i, \alpha_i) - \frac{\gamma}{2} \sum_{j=1}^n s_j \boldsymbol{\alpha}^\top \boldsymbol{X}_j \boldsymbol{X}_j^\top \boldsymbol{\alpha}.$$

The function f is convex –indeed linear– in $\boldsymbol{s}$ over $[0,1]^p$, so c is convex over $[0,1]^p$.   Q.E.D.

In practice, for a given support $\boldsymbol{s}$, we evaluate the function $c(\boldsymbol{s})$ by solving the maximization problem (9) with any of the numerical procedures presented in the previous section. In what follows, we need to calculate a sub-gradient of the function c as well. Using the dual maximizer $\boldsymbol{\alpha}^\star(\boldsymbol{s})$ in (9) at a support $\boldsymbol{s}$, we can compute one at no additional computational cost. Indeed, it follows that

$$\frac{\partial c(\boldsymbol{s})}{\partial s_j} = -\frac{\gamma}{2} \boldsymbol{\alpha}^\star(\boldsymbol{s})^\top \boldsymbol{X}_j \boldsymbol{X}_j^\top \boldsymbol{\alpha}^\star(\boldsymbol{s}).$$

2.3. Enhanced Outer-Approximation Algorithm

We solve the convex integer optimizaton (CIO) problem (8) taking into account that we can readily compute $c(\boldsymbol{s})$ and $\nabla c(\boldsymbol{s})$ for any given $\boldsymbol{s}$. None of the commercial solvers available are targeted to solve such CIO problems where there is no closed-form expression for $c(\boldsymbol{s})$. We propose to adopt an outer-approximation approach similar to the one introduced by Duran and Grossmann (1986) for linear mixed-integer optimization problems.

We first reformulate (8) as a mixed-integer optimization problem in epigraph form:

$$\min_{\boldsymbol{s} \in \mathcal{S}_k^p, \eta} \eta \ \text{ s.t. } \ \eta \geqslant c(\boldsymbol{s}).$$

By convexity, we can approximate $c(\boldsymbol{s})$ by the supremum of its tangents at all points $\tilde{\boldsymbol{s}} \in \mathcal{S}_k^p$, i.e.,

$c(\boldsymbol{s}) \geq \max_{\tilde{\boldsymbol{s}} \in \mathcal{S}_k^p} \left\{ c(\tilde{\boldsymbol{s}}) + \nabla c(\tilde{\boldsymbol{s}})^\top (\boldsymbol{s} - \tilde{\boldsymbol{s}}) \right\}$. For $\boldsymbol{s} \in \mathcal{S}_k^p$, this lower approximation matches $c(\boldsymbol{s})$ exactly.

Hence, we can equivalently solve

$$\min_{\boldsymbol{s} \in \mathcal{S}_k^p, \eta} \eta \ \text{ s.t. } \ \eta \geqslant c(\tilde{\boldsymbol{s}}) + \nabla c(\tilde{\boldsymbol{s}})^\top (\boldsymbol{s} - \tilde{\boldsymbol{s}}), \forall \tilde{\boldsymbol{s}} \in \mathcal{S}_k^p. \tag{10}$$

As described in Fletcher and Leyffer (1994), Bonami et al. (2008), we find a solution to (10) by iteratively introducing the linear constraints defining (10), i.e., iteratively refining a piece-wise linear lower approximation of c. The solver structure is given in pseudocode in Algorithm 1.

Algorithm 1 Outer-approximation algorithm

Require: $\boldsymbol{X} \in \mathbb{R}^{n \times p}$, $\boldsymbol{y} \in \{-1, 1\}^p$, $k \in \{1, ..., p\}$, γ

    $\boldsymbol{s}^{(1)} \leftarrow$ warm-start

    $\eta^{(1)} \leftarrow 0$

    $t \leftarrow 1$

    repeat

        $\boldsymbol{s}^{(t+1)}, \eta^{(t+1)} \leftarrow \operatorname{argmin}_{s,\eta} \left\{ \eta : \boldsymbol{s} \in \mathcal{S}_k^p, \eta \geqslant c(\boldsymbol{s}^{(i)}) + \nabla c(\boldsymbol{s}^{(i)})^\top (\boldsymbol{s} - \boldsymbol{s}^{(i)}), \ i = 1, \ldots, t \right\}$

        $t \leftarrow t + 1$

    until $\eta^{(t)} < c(\boldsymbol{s}^{(t)})$

    return $\boldsymbol{s}^{(t)}$

Under Assumption 1, Algorithm 1 is guaranteed to terminate in a finite, yet exponential in the worst-case, number of steps and return an optimal solution of (8) (Fletcher and Leyffer 1994). On this regard, Algorithm 1 resembles the simplex algorithm whose worst-case complexity is exponential, yet is very efficient in practice (Goldfarb 1994).

Several enhancements have been proposed to improve the convergence speed of Algorithm 1. First, in its original form, Algorithm 1 requires solving a mixed-integer linear optimization problem at each iteration, which is computationally inefficient. Modern solvers however, such as Gurobi (Gurobi Optimization 2016), IBM CPLEX (CPLEX 2011), or GLPK1. , can handle *lazy callbacks*, a feature that integrates the constraint generation procedure within a unique branch-and-bound

enumeration tree, shared by all subproblems. We implemented Algorithm 1 in this fashion. In addition, decomposition schemes as Algorithm 1 benefit from performing a rich root node analysis, as advocated by Fischetti et al. (2017). In essence, a "rich" root node analysis consists of a good initial feasible solution $\boldsymbol{s}^{(1)}$ (i.e, a good upper bound) and a set of initial constraints of the form $\eta \geqslant c(\boldsymbol{s}^{(i)}) + \nabla c(\boldsymbol{s}^{(i)})^\top (\boldsymbol{s} - \boldsymbol{s}^{(i)})$ to obtain tight lower bound as well. Regarding the warm-start $\boldsymbol{s}^{(1)}$, we recommend using the Lasso estimator provided by the `glmnet` package (Friedman et al. 2013) or estimators obtained by rounding the solution of the Boolean relaxation of (8). We refer to Pilanci et al. (2015) for a theoretical analysis of the latter, and Bertsimas et al. (2020a), Atamturk and Gomez (2019) for efficient numerical algorithms to solve it. Regarding the lower-bound and the initial constraints, Fischetti et al. (2017) suggests using the constraints obtained from solving the Boolean relaxation of (8) via a similar outer-approximation procedure - in which case there are no binary variables and the technique is often referred to as Kelley's algorithm (Kelley 1960). We refrain from implementing this strategy in our case. Indeed, computing $c(\boldsymbol{s})$ and $\nabla c(\boldsymbol{s})$ reduces to solving a binary classification problem over the k features encoded by the support of $\boldsymbol{s}$. As a result, the scalability of our approach largely relies on the fact that constraints are computed for sparse vectors $\boldsymbol{s}$. When solving the Boolean relaxation, however, $\boldsymbol{s}$ can be dense while satisfying the constraint $\boldsymbol{s}^\top \mathbf{e} \leqslant k$. Instead, we compute a regularization path for the Lasso estimator up to a sparsity level of $k + 1$ using the `glmnet` package, and initialize the outer approximation with the constraints obtained from these solutions.

Finally, we propose a stochastic version of the outer-approximation algorithm to improve the scalability with respect to n. At the incumbent solution $\boldsymbol{s}^{(t)}$, we observe that we do not need to solve Problem (9) to optimality to obtain a valid linear lower-approximation of $c(\boldsymbol{s})$. Indeed, any $\boldsymbol{\alpha} \in \mathbb{R}^n : \mathbf{e}^\top \boldsymbol{\alpha} = 0$ yields

$$c(\boldsymbol{s}) \geqslant -\sum_{i=1}^n \hat{\ell}(y_i, \alpha_i) - \frac{\gamma}{2} \sum_{j=1}^n s_j \boldsymbol{\alpha}^\top \boldsymbol{X}_j \boldsymbol{X}_j^\top \boldsymbol{\alpha}.$$

Hence, we propose a strategy (Algorithm 2) to find a good candidate solution α and the corresponding lower approximation $c(\boldsymbol{s}) \geqslant \tilde{c}(\boldsymbol{s}^{(t)}) + \nabla \tilde{c}(\boldsymbol{s}^{(t)})^\top (\boldsymbol{s} - \boldsymbol{s}^{(t)})$. Our strategy relies on the fact the

primal formulation in $\boldsymbol{w_s} \in \mathbb{R}^k$ only involves k decision variables, with $k < n$ or even $k \ll n$ in practice. As a result, one should not need the entire data set to estimate $\boldsymbol{w}^\star(\boldsymbol{s}^{(t)})$. Instead, we randomly select N out of n observations, estimate $\boldsymbol{w}^\star(\boldsymbol{s}^{(t)})$ on this reduced data set, and finally average the result over B subsamples. Typically, we take $N = \max(10\%n, 2k)$ and $B = 10$ in our experiments. Then, we estimate $\boldsymbol{\alpha} \in \mathbb{R}^n$ by solving the first-order optimality conditions $\alpha_i \in \partial\ell(y_i, \boldsymbol{w}^\top\boldsymbol{x}^{(i)})$. Since our objective is to generate a constraint that tightens the current lower-approximation of $c(\boldsymbol{s}^{(t)})$, we compare $\tilde{c}(\boldsymbol{s}^{(t)})$ with our current estimate of $c(\boldsymbol{s}^{(t)})$, $\eta^{(t)}$. If $\eta^{(t)} > \tilde{c}(\boldsymbol{s}^{(t)})$, i.e., if the approximate constraint does not improve the approximation, we reject it and compute $c(\boldsymbol{s}^{(t)}), \nabla c(\boldsymbol{s}^{(t)}))$ exactly instead.

Algorithm 2 Stochastic constraint generation procedure

Require: $\boldsymbol{s} \in \{0,1\}^p$, $\boldsymbol{X} \in \mathbb{R}^{n \times k}$, $\boldsymbol{y} \in \{-1,1\}^p$, B, N

  for $b = 1, \ldots, B$ **do**

    Define $\tilde{\boldsymbol{X}}^b \in \mathbb{R}^{N \times k}, \tilde{\boldsymbol{y}}^b \in \{-1,1\}^N$ obtained from $\boldsymbol{X}$ and $\boldsymbol{y}$ by randomly selecting N rows.

    Compute $\tilde{\boldsymbol{w}}^b$, solution of (4) with input data $(\tilde{\boldsymbol{X}}^b, \tilde{\boldsymbol{y}}^b)$.

  end for

  $\boldsymbol{w} \leftarrow \frac{1}{B}\sum_b \tilde{\boldsymbol{w}}_b$

  Find $\alpha_i \in \partial\ell(y_i, \boldsymbol{w}^\top\boldsymbol{x}^{(i)})$

  $\tilde{c} \leftarrow -\sum_i \hat{\ell}(y_i, \alpha_i) - \frac{\gamma}{2}\sum_j s_j(\boldsymbol{X}_j^\top\boldsymbol{\alpha})^2$, $\nabla\tilde{c}(\boldsymbol{s})_j \leftarrow -\frac{\gamma}{2}(\boldsymbol{X}_j^\top\boldsymbol{\alpha})^2$

  return $\tilde{c}(\boldsymbol{s}), \nabla\tilde{c}(\boldsymbol{s})$

In the next section, we provide numerical evidence that Algorithm 1, both for logistic and hinge loss functions, is a scalable method to compute cardinality constrained classifiers and select more accurately features than L_1-based heuristics.

2.4. Practical implementation considerations

Sparse regularized classification (7) involves two hyperparameters - a regularization factor γ and the sparsity k. For the regularization parameter γ, we fit its value using cross-validation among values uniformly distributed in the log-space: we start with a low value γ_0 - typically γ_0 scaling as

$1/\max_i \|\boldsymbol{x}^{(i)}\|^2$ as suggested in (Chu et al. 2015) - and inflate it iteratively by a factor two. We similarly tune k by simple hold-out cross-validation over a range of values. We use out-of-sample Area Under the receiving operator Curve as the validation criterion (to maximize). Although we did not implement it, Kenney et al. (2018) present a general blueprint consisting of warm-start strategies and a bisection search procedure that can tangibly accelerate the overall cross-validation loop.

3. Numerical experiments: Scalability and Support Recovery

In this section, we evaluate the numerical performance of our method both in terms of scalability and quality of the features selected.

The computational tests were performed on a computer with Xeon @2.3GhZ processors, 1 core, 8GB RAM. Algorithms were implemented in Julia 1.0 (Lubin and Dunning 2015), a technical computing language, and Problem (8) was solved with Gurobi 8.1.0 (Gurobi Optimization 2016). We interfaced Julia with Gurobi using the modeling language JuMP 0.21.1 (Dunning et al. 2017). Our code is available on GitHub2. . Since our sparse regularized formulation (7) comprises 2 hyperparameters, γ and k, that control the degree of regularization and sparsity respectively, we compare our method with the ElasticNet formulation

$$\min_{\boldsymbol{w}\in\mathbb{R}^p, b\in\mathbb{R}} \sum_{i=1}^n \ell(y_i, \boldsymbol{w}^\top \boldsymbol{x}^{(i)} + b) + \lambda\left[(1-\alpha)\|\boldsymbol{w}\|_1 + \alpha\|\boldsymbol{w}\|_2^2\right], \tag{11}$$

which similarly contains 2 hyper parameters and can be computed efficiently by the `glmnet` package (Friedman et al. 2013). For a fair comparison, we cross-validate λ and α using the same procedure as γ and k described in Section 2.4. Our intention in this section is first and foremost to assess the validity of our numerical algorithms for computing the cardinality-constrained estimator (7). We refer to Bertsimas et al. (2020a), Hastie et al. (2020) for extensive numerical experiments and discussions on the statistical and practical relevance of each formulation, as well as comparison with non-convex penalties and alternatives.

3.1. Support recovery on synthetic data

We first consider synthesized data sets to assess the feature selection ability of our method compared it to a state-of-the-art L_1-based estimators.

3.1.1. Methodology   We draw $\boldsymbol{x}^{(i)} \sim \mathcal{N}(0_p, \Sigma), i = 1, \ldots, n$ independent realizations from a p-dimensional normal distribution with mean 0_p and covariance matrix $\Sigma_{ij} = \rho^{|i-j|}$. Columns of $\boldsymbol{X}$ are then normalized to have zero mean and unit variance. We randomly sample a weight vector $\boldsymbol{w}_{true} \in \{-1, 0, 1\}^p$ with exactly k nonzero coefficients. We draw $\varepsilon_i, i = 1, \ldots, n$, i.i.d. noise components from a normal distribution scaled according to a chosen signal-to-noise ratio

$$\sqrt{SNR} = \|\boldsymbol{X}\boldsymbol{w}_{true}\|_2 / \|\boldsymbol{\varepsilon}\|_2.$$

Finally, we construct y_i as $y_i = \text{sign}\left(\boldsymbol{w}_{true}^\top \boldsymbol{x}^{(i)} + \varepsilon_i > 0\right)$. This methodology enables us to produce synthetic data sets where we control the sample size n, feature size p, sparsity k, feature correlation ρ and signal-to-noise ratio SNR.

3.1.2. Support recovery metrics   Given the true classifier $\boldsymbol{w}_{true}$ of sparsity k_{true}, we assess the correctness of a classifier $\boldsymbol{w}$ of sparsity k by its accuracy, i.e., the number of true features it selects

$$A(\boldsymbol{w}) = |\{j : w_j \neq 0, w_{true,j} \neq 0\}| \in \{0, \ldots, k_{true}\},$$

and the false discovery, i.e., the number of false features it incorporates

$$F(\boldsymbol{w}) = |\{j : w_j \neq 0, w_{true,j} = 0\}| \in \{0, \ldots, p\}.$$

Obviously, $A(\boldsymbol{w}) + F(\boldsymbol{w}) = |\{j : w_j \neq 0\}| = k$. A classifier $\boldsymbol{w}$ is said to *perfectly recover* the true support if it selects the truth ($A(\boldsymbol{w}) = k_{true}$) and nothing but the truth ($F(\boldsymbol{w}) = 0$ or equivalently $k = k_{true}$).

3.1.3. Selecting the truth . . . We first compare the performance of our algorithm for sparse regression with ElasticNet, when both methods are given the true number of features in the support k_{true}. As mentioned in the introduction, a key property in this context for any best subset selection method, is that it selects the true support as sample size increases, as represented in Figure 2. From that perspective, both methods demonstrate a similar convergence: As n increases, both classifiers end up selecting the truth, with Algorithm 1 needing somewhat smaller number of samples than Lasso.
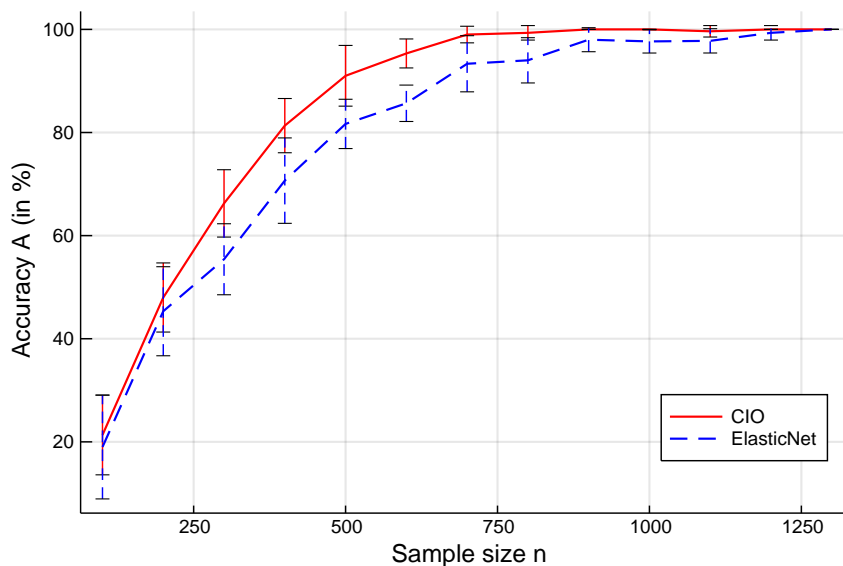

Figure 2   Accuracy as the sample size n increases, for ElasticNet with the logistic loss (dashed blue) and sparse SVM (solid red). Results correspond to average values obtained over 10 synthetic data sets with $p = 1,000$, $k_{true} = k = 30$, $\rho = 0.3$, $SNR \to \infty$ and increasing n from 100 to 1,300.

Apart from the absolute number of true/false features selected, one might wonder whether the features selected are actually good features in terms of predictive power. In this metric, sparse regression significantly outperforms the L_1-based classifier, both in terms of Area Under the Curve (AUC) and misclassification rate, as shown on Figure 3, demonstrating a clear predictive edge of exact sparse formulation.

In terms of computational cost, Lasso remains the gold standard: computational times for `glmnet` range from 0.05 to 0.48 seconds, while CIO terminates between 0.03 and 114.5 seconds on the same
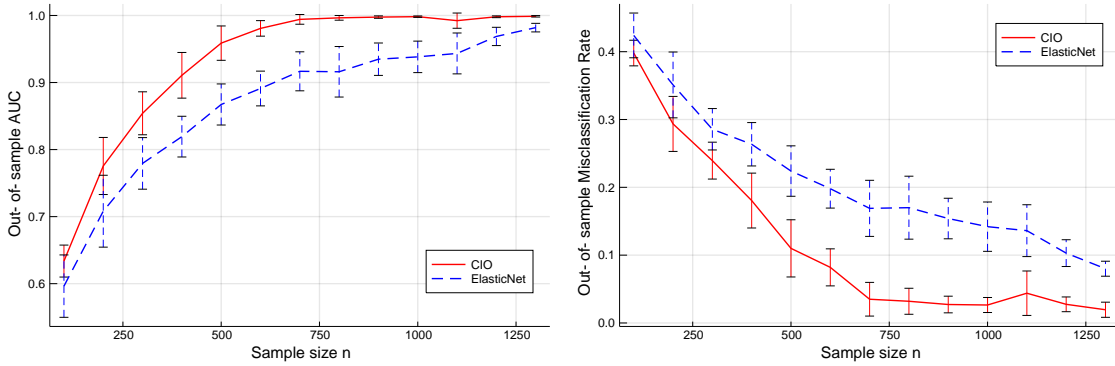
Figure 3   Out-of-sample AUC (left panel) and misclassification rate (right panel) as the sample size n increases, for ElasticNet with logistic loss (dashed blue) and sparse SVM (solid red). Results correspond to average values obtained over 10 synthetic data sets with $p = 1,000$, $k_{true} = k = 30$, $\rho = 0.3$, $SNR \to \infty$ and increasing n from 100 to 1,300.

instances. Figure 4 represents the number of constraints (left panel) and computational time (right panel) required by our outer-approximation algorithm as the problem size n increases, $\gamma \times n$ being fixed. For low values of n, the number of constraints is in the thousands. Surprisingly, computational effort does not increase with n. On the contrary, having more observations reduces the number of constraints down to less than twenty. Intuitively, as n grows, there is more signal and the problem becomes easier. Computational time evolves similarly: the algorithm reaches the time limit (here, 3 minutes) when n is low, but terminates in a few seconds for high values. For sparse linear regression, Bertsimas et al. (2020b) observed a similar, yet even sharper, phenomenon which they referred to as a phase transition in computational complexity. The threshold value, however, increases with γ. In other words, when n is fixed, computational time increases as γ increases (see Figure 9 in §3.2), which corroborates the intuition that in the limit $\gamma \to 0$, $\boldsymbol{w}^{\star} = 0$ is obviously optimal, while the problem can be ill-posed as $\gamma \to +\infty$. Since the right regularization parameter is unknown *a priori*, one needs to test high but sometimes relevant values of γ for which the time limit is reached and the overall procedures terminates in minutes, while `glmnet` computes the entire regularization path for Lasso in less than a second. As for the choice of the time limit, it does not significantly impact the performance of the algorithm: As shown on Figure 5, the algorithm quickly finds the

optimal solution and much of the computational time is spent improving the lower bound, i.e., proving the solution is indeed optimal. We will further explore the numerical scalability of the outer-approximation algorithm in Section 3.2.
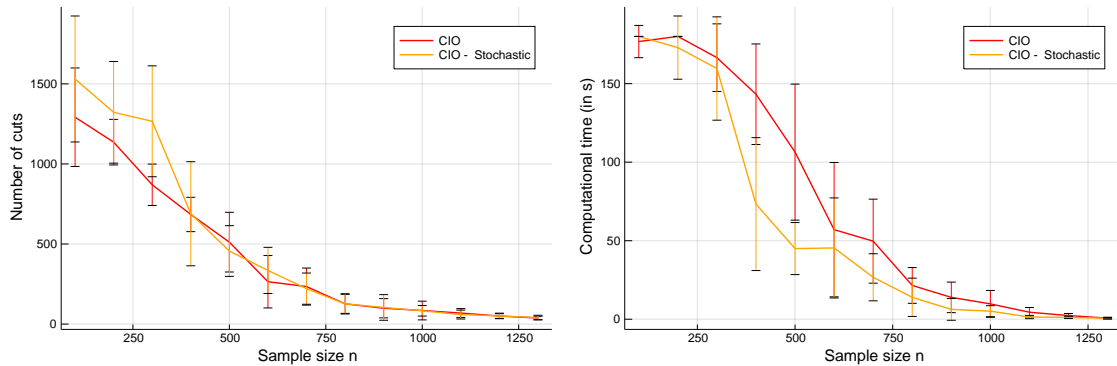

Figure 4   Number of constraints (left panel) and computational time (right panel) required by the outer-approximation algorithm with Hinge loss as sample size n increases. Results correspond to average values obtained over 10 synthetic data sets with $p = 1,000$, $k_{true} = k = 30$, $\rho = 0.3$, $SNR \to \infty$, $\gamma = \gamma_0/n$ with $\gamma_0 = 2^5 p/(k \max_i \|\boldsymbol{x}^{(i)}\|^2)$ and increasing n from 100 to 1,300. Algorithm 1 is initialized with the Lasso solution.
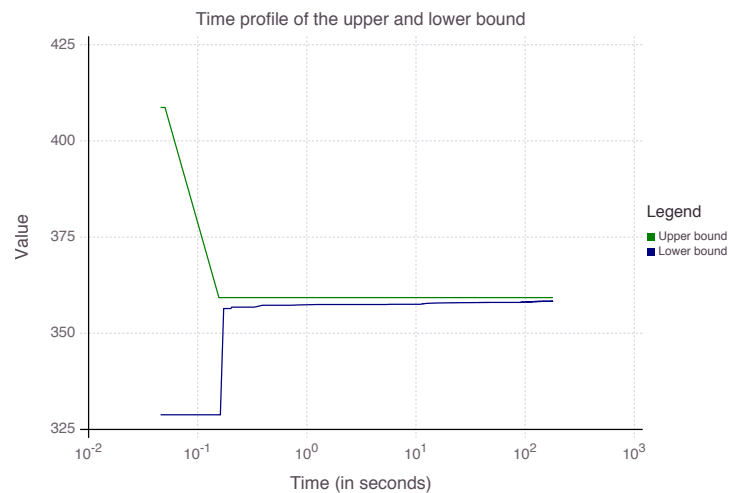

Figure 5   Typical evolution of the upper (best feasible solution, in green) and lower bounds (in blue) of Algorithm 1 as computational of time (in log scale) increases. Results for one problem instance with $p = 1,000$, $k_{true} = 30$, $\rho = 0.3$, $SNR \to \infty$, $\gamma = \gamma_0/n$ with $\gamma_0 = 2^7 p/(k \max_i \|\boldsymbol{x}^{(i)}\|^2)$ and $n = 600$.

3.1.4. ... and nothing but the truth In practice, however, the length of the true support k_{true} is unknown *a priori* and is to be determined using cross-validation.

Given a data set with a fixed number of samples n and features p, we compute classifiers with different values of sparsity parameter k and choose the value which leads to the best accuracy on a validation set. Irrespective of the method, AUC as a function of sparsity k should have an inverted-U shape: if k is too small, not enough features are taken into account to provide accurate predictions. If k is too big, the model is too complex and overfits the training data. Hence, there is some optimal value $k^\star$ which maximizes validation AUC (equivalently, one could use misclassification rate instead of AUC as a performance metric). Figure 6 represents the evolution of the AUC on a validation set as sparsity k increases for Lasso and the exact sparse logistic regression. The exact CIO formulation leads to an optimal sparsity value $k^\star_{CIO}$ which is much closer to the truth than $k^\star_{Lasso}$, and this observation remains valid when n increases as shown on the left panel of Figure 7. In addition, Figure 7 also exposes a major deficiency of Lasso as a feature selection method: even when the number of samples increases, Lasso fails to select the relevant features *only* and returns a support $k^\star_{Lasso}$ much larger than the truth whereas $k^\star_{CIO}$ converges to k_{true} quickly as n increases, hence selecting the truth and nothing but the truth.
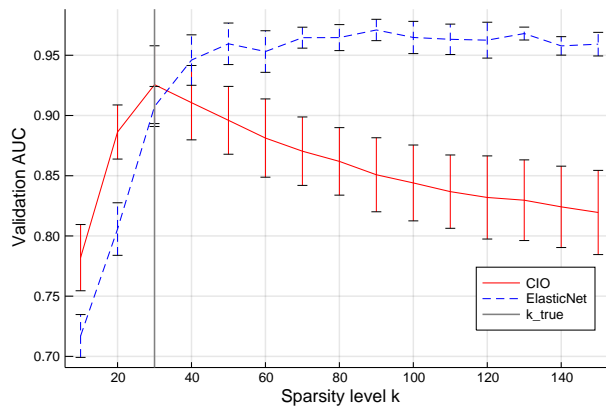

Figure 6   Validation AUC as sparsity of the classifier k increases, for ElasticNet (dashed blue) and sparse SVM (solid red). Results correspond to average values obtained over 10 synthetic data sets with $n = 700$, $p = 1,000$, $k_{true} = 30$ (black vertical line), $\rho = 0.3$, $SNR \to \infty$ and increasing k from 0 to 100.
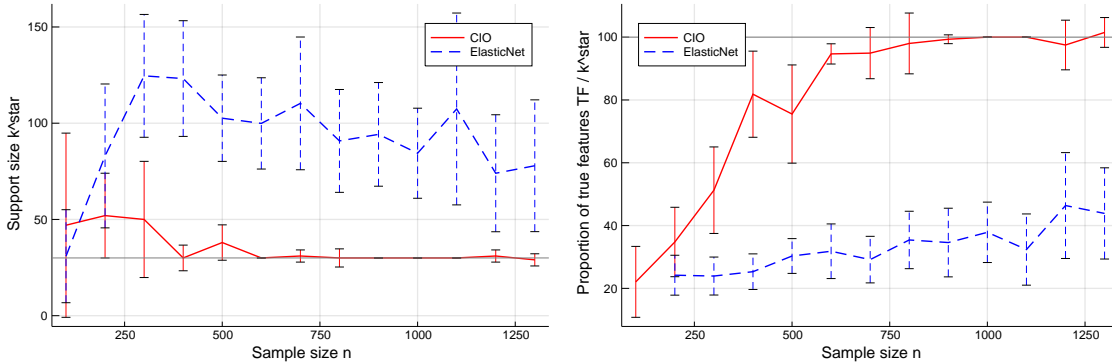
Figure 7   Optimal sparsity $k^\star$ (left) and accuracy rate $A/k^\star$ (right) as sample size n increases, for ElasticNet (dashed blue) and sparse SVM (solid red). Results correspond to average values obtained over 10 synthetic data sets with $p = 1,000$, $k_{true} = 30$, $\rho = 0.3$, $SNR \to \infty$ and increasing n from 100 to 1,900.

As mentioned in introduction, these shortcomings of Lasso-based feature selection have already been documented and motivated alternatives such as non-convex penalties. Extensive numerical comparisons, in regression and classification settings, presented Bertsimas et al. (2020a) corroborate our findings, and demonstrate that non-convex penalties and L_0-based formulations both provide a substantial improvement over L_1-regularization. Yet, in their analysis, discrete optimization methods do have some edge over the non-convex penalties SCAD and MCP in terms of false discoveries.

3.2. Scalability on synthetic data

We use the opportunity of working with synthetic data to numerically assess how computational time is impacted by the size of the problem - as measured by the number of samples, n, and features, p - and the two hyper parameters γ and k.

Impact of sample size n: As previously observed, computational time does not increase as n increases. On the contrary, for a fixed value of $\gamma \times n$, computational time decreases as n grows. As displayed on Figure 4 for Hinge loss, this phenomenon is observed for both the standard and stochastic version of the outer-approximation algorithm. Observe that the stochastic variant slightly increases the number of constraints for low values of n, yet, since generating a constraint is less expensive, computational time is reduced by a factor 2 for the hardest instances. On this regard, the

stochastic version of the outer-approximation algorithm is more beneficial when using the logistic loss compared with the Hinge loss, as depicted on Figure 8. Indeed, number of constraints are reduced by a factor 5 and computational time by a factor 10. In our understanding, this substantial improvement in the logistic case is due to the fact that the relationship $\alpha_i \in \partial \ell(y_i, {\boldsymbol{x}^{(i)}}^\top \boldsymbol{w})$ uniquely defines α and hence leads to higher quality constraints than in the case of the Hinge loss where ℓ is not differentiable at 0.
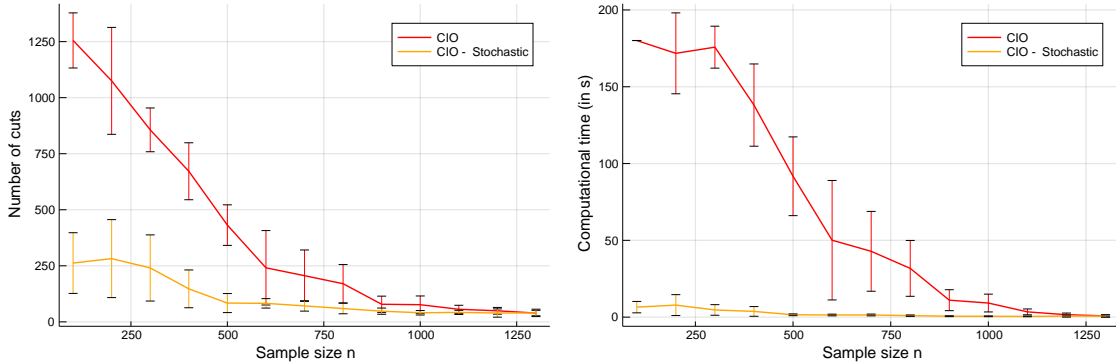

Figure 8   Number of constraints (left panel) and computational time (right panel) required by the outer-approximation algorithm with Logistic loss as sample size n increases. Results correspond to average values obtained over 10 data sets with $p = 1,000$, $k_{true} = 30$, $\rho = 0.3$, $SNR \to \infty$, $\gamma = \gamma_0/n$ with $\gamma_0 = 2^5 p/(k \max_i \|\boldsymbol{x}^{(i)}\|^2)$ and increasing n from 100 to 1,300. Algorithm 1 is initialized with the Lasso solution.

Impact of regularization γ: As γ increases, however, computational time and the number of constraints required sharply increases, as depicted in Figure 9. This phenomenon is consistently observed for all problem sizes and loss functions. Since the proper value of γ is unknown a priori and needs to be cross-validated, this could result in prohibitive overall computational time. However, two aspects need to be kept in mind: First, computational time, here, corresponds to time needed to certify optimality, although high-quality solutions - or even the optimal solution - can be found much faster. Second, we do not implement any warm-starting strategy between instances with different values of γ within the grid search. Smarter grid search procedures (Kenney et al. 2018) could further accelerate computations.
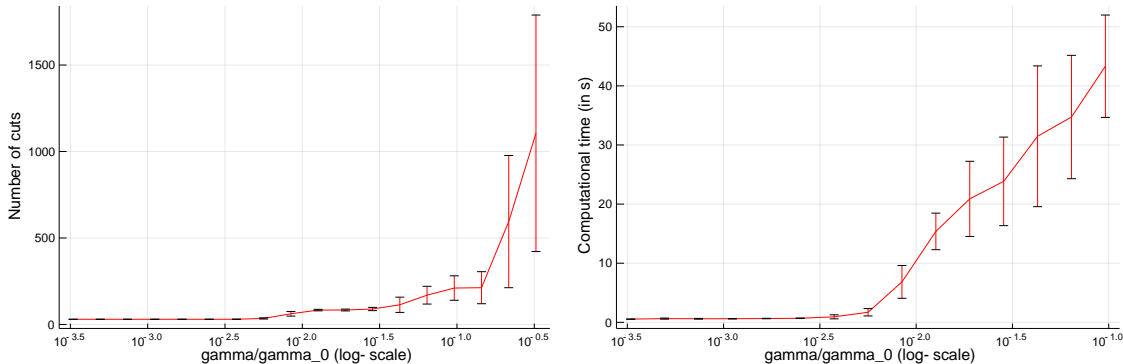
Figure 9   Number of constraints (left panel) and computational time (right panel) required by the outer-approximation algorithm with Hinge loss as the regularization parameter γ increases. Results correspond to average values obtained over 5 data sets with $p = 10,000$, $n = 1,000$, $k = k_{true} = 5$, $\rho = 0$, $SNR = 10$. The x-axis represents γ/γ_0 with $\gamma_0 = 10/n$. Algorithm 1 is initialized with the Lasso solution.

Impact of feature size p and sparsity level k: The master problem (8) obviously depends on the number of features p and the sparsity level k through the feasible set $\mathcal{S}_k^p$. As a first-order approximation, the combinatorial complexity of the problem can be captured by the total number of potential supports of size k, $\binom{p}{k}$. So one could expect a linear dependency on p for k fixed, and an exponential dependency on k, p being fixed. This intuition is largely confirmed empirically, as reported in Table 3: In low sparsity regimes ($k = 5$), we can solve instances with up to $50,000$ features under a minute, while denser problems are not solved to optimality after 30 minutes even for $p = 5,000$. Our experiments also highlight the benefits from having a good initialization solution s_1 and using the stochastic constraint generation technique. As displayed in Table 3, together, these enhancements reduce computational time up to a factor of 10.

3.3. Experiments on real-world data sets

We now illustrate the practical implications of sparse classification algorithms on real-world data, of various size and dimensions.

3.3.1. Over-determined regime $n > p$ We consider data sets from the UC Irvine Machine Learning Repository (available at `https://archive.ics.uci.edu/ml/datasets.html`), split

Table 3 Computational time (in seconds) of Algorithm 1 for large p and varying k. Instances are generated with $k = k_{true}$, $SNR = 10$, $n = 1,000$, and $\gamma = 10/n$ as in Dedieu et al. (2021). We indicate in parentheses the warm-start method used. SubsetSelection refers to a rounding of the solution of the Boolean relaxation of (7) as implemented in Bertsimas et al. (2020a). If the algorithm did not converge within the given time budget (1,800 seconds), we report optimality gap in parenthesis.

$k = 5$, $p =$	10,000	20,000	30,000	40,0000	50,000
glmnet	2.2	3.6	4.9	6.3	7.3
Hinge loss (glmnet)	18.2	29.4	29.3	30.7	148.7
Hinge loss (SubsetSelection)	15.2	19.8	42.2	43.3	40.8
Hinge loss (SubsetSelection) Stochastic	9.1	28.5	31.0	42.9	55.1

$p = 5,000$, $k =$	1	5	10	15	20
glmnet	1.8	2.3	2.3	2.7	2.3
Hinge loss (glmnet)	0.6	7.6	569.3	702.6	1800 (40%)
Hinge loss (SubsetSelection)	0.2	7.5	203.0	537.0	1800 (46%)
Hinge loss (SubsetSelection) Stochastic	0.3	7.6	65.8	425.6	1800 (41%)

them into a training and a test set (80%/20%), calibrate a classifier on the training data, using cross-validation to fit the hyper-parameters k and γ in the sparse case and λ and α in the ElasticNet (Enet) case, and compare AUC on the test set for both methods. Characteristics of these data sets and experimental results are given in Table 4. Experiments clearly demonstrate that (a) our outer-approximation algorithm scales to data sets of practical relevance, (b) cardinality constrained formulations generally lead to sparser classifiers than ElasticNet (on 7 out 8 data sets) with comparable predictive power, with ElasticNet being more accurate on 3 out of 8 data sets. It suggests that features selected by our discrete optimization carry more relevant information that those obtained by ElasticNet. Yet, since these data sets contain a limited number of features p, they may not make a strong case for exact sparse classification methods compared to Lasso, the original problem being relatively sparse already. Therefore, we investigate the under-determined regime $n < p$ in the next section.

Table 4   Comparative results of L_1-regularized and L_0-constrained estimators on data sets from UCI ML Repository. AUC are computed out-of-sample on a test set comprised of 20% of the initial data. Results are averaged over 10 different splits into train/validation data for cross-validation of the hyper-parameters.

			ElasticNet		CIO - Hinge		CIO - Logistic	
Data set	n	p	k	AUC	k	AUC	k	AUC
Banknote Authentication	1,372	5	4.0	1.000	3.7	1.000	**3.2**	1.000
Breast Cancer	683	10	7.8	**0.991**	**5.4**	0.990	5.5	0.989
Breast Cancer (Diagnostic)	569	31	21.1	**0.998**	12.0	0.997	**11.4**	0.995
Chess (Rook vs. Pawn)	3,196	38	13.8	0.867	15.6	0.862	**12.6**	**0.868**
Cylinder bands	277	484	**98**	0.701	111.0	**0.728**	139.0	0.719
Magic Telescope	19,020	11	9.2	0.840	6.4	0.843	**6.2**	**0.844**
QSAR Biodegradation	1,055	42	35.8	0.924	**18.5**	**0.950**	21.8	0.947
Spambase	4,601	58	56.4	**0.960**	27.2	0.957	**21.6**	0.954

3.3.2. Under-determined regime $p > n$ Performance of sparse classification in the under-determined regime is crucial for two reasons: Since the amount of data available is limited, such regime favors estimators which can efficiently recover the truth even when the sample size n is small with regard to p. More importantly, under-determined regimes occur in highly impactful applications, such as medical research. To show the direct implications of our method on this field of research, we used data from The Cancer Genome Atlas Research Network (available at `http://cancergenome.nih.gov`) on $n = 1,145$ lung cancer patients. Actually, tumor types often have distinct subtypes, each of them having its own genetic signature. In our sample for instance, 594 patients (51.9%) suffered from Adenocarcinoma while the remaining 551 patients (48.1%) suffered from Squamous Cell Carcinoma. The data set consists of gene expression data for $p = 14,858$ genes for each patient. We apply both sparse and Lasso classification to identify the most relevant genes to discriminate between the two subtypes and compile the results in Table 5. The first conclusion to be drawn from our results is that the exact sparse classification problem scales to problems of such size, which is far above data sets usually encountered in the gene selection academic literature. In addition, explicitly constraining sparsity of the classifier leads to much sparser, thus more interpretable results with little compromise on the predictive power: Sparse

SVM reaches an AUC of 0.977 with only 38 features while the L_1-regularized classifier selects ten times more genes for a +0.005 gain in AUC.

Table 5   Comparative results of L_1-regularized and L_0-constrained estimators on the Lung Cancer data. AUC are computed out-of-sample on a test set comprised of 20% of the initial data. Results are averaged over 10 different splits into train/validation data for cross-validation of the hyper-parameters.

			ElasticNet		CIO - Hinge		CIO - Logistic	
Data set	n	p	k	AUC	k	AUC	k	AUC
Lung cancer	$1,145$	$14,858$	378.4	**0.982**	**38**	0.977	90	0.980

3.4. Summary and guidelines

All in all, these experiments demonstrate that (a) the L_2-regularized L_0-constrained formulation (7) provides an edge over L_1-based estimators such as Lasso or ElasticNet in terms of support recovery (accuracy and false detection rate) and downstream predictive power (AUC and misclassification rate), (b) though NP-hard in the worst case, solving the discrete optimization problem (7) can be routinely achieved within minutes for large problem sizes of interest (up to $p \approx 10,000$s). Though reasonable for most applications, these computational times are still one to two orders of magnitude higher than those of Lasso, which remains the gold standard in terms of scalability. In particular, the regularization parameter λ of (2) can be cross-validated at no additional computational cost. Designing efficient cross-validation procedures for cardinality-based estimators is an active area of research (Kenney et al. 2018, Hazimeh and Mazumder 2020). In practice, for a large data set, one could use Lasso as a feature screening method to exclude non-relevant covariates before applying a more computationally demanding feature selection method on a reduced subset of features.

4. Towards a theoretical understanding of asymptotic support recovery

As mentioned in the introduction, a large body of literature has provided information-theoretic limitations (Wainwright 2009a, Wang et al. 2010, Gamarnik and Zadik 2017) or theoretical guarantees of support recovery by some specific algorithms (Wainwright 2009b, Pilanci et al. 2015) for

sparse linear regression, which supported the empirical observation of a phase transition (Donoho and Stodden 2006, Bertsimas et al. 2020b). In classification, however, we did not observe such a sharp phenomenon. For instance, we observed a smooth convergence in accuracy in Figure 2, while Bertsimas et al. (2020a, Figure 1) experienced a sharp phase transition in low-noise linear regression. In this section, we provide some intuition on the theoretical mechanisms involved for perfect support recovery specific to classification. We prove an information-theoretic sufficient condition to achieve perfect support recovery and compare it with analogous bounds for linear regression, as well recent theoretical results on 1-bit compressed sensing (Jacques et al. 2013, Scarlett and Cevher 2017).

4.1. Notations and assumptions

To simplify the analysis, we consider a stylized framework where the data is generated according to the equation

$$y_i = \text{sign}\left(\boldsymbol{x}^{(i)^\top} \boldsymbol{w}^\star + \varepsilon_i\right),$$

where x_i are i.i.d. standard Gaussian random variables, $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, $\boldsymbol{w}^\star \in \mathbb{R}^p$ with $\|\boldsymbol{w}^\star\|_0 = k$ and $\text{sign}(\lambda) = 1$ if $\lambda > 0$, -1, otherwise. Given a classifier $\boldsymbol{w}$ predictions will be made according to the rule

$$\hat{y}_i(\boldsymbol{w}) = \text{sign}\left(\boldsymbol{x}^{(i)^\top} \boldsymbol{w}\right).$$

It is obvious from the definition that for any $\boldsymbol{w} \in \mathbb{R}^p$, $\text{sign}(\lambda \boldsymbol{x}^\top \boldsymbol{w}) = \text{sign}(\boldsymbol{x}^\top \boldsymbol{w})$, for any $\lambda > 0$. In other words, predictions made by a classifier are insensitive to scaling. As a consequence, the difference in prediction between two classifiers should demonstrate the same invariance and indeed only depends on the angle between the classifiers as formally stated in Lemma 1 (proof in Appendix A). This observation does not hold for $\text{sign}(\boldsymbol{x}^\top \boldsymbol{w} + \varepsilon)$, because of the presence of noise.

LEMMA 1. *Assume $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}_p, \boldsymbol{I}_p)$ and $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ are independent. Then, for any $\boldsymbol{w}, \boldsymbol{w}' \in \mathbb{R}^p$ we have that*

$$\mathbb{P}\left(sign\left(\boldsymbol{x}^\top \boldsymbol{w}\right) \neq sign\left(\boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon\right)\right) = \frac{1}{\pi} \arccos\left(\frac{\boldsymbol{w}^\top \boldsymbol{w}'}{\|\boldsymbol{w}\| \sqrt{\|\boldsymbol{w}'\|^2 + \sigma^2}}\right). \tag{12}$$

We consider classifiers with binary entries $\boldsymbol{w}^\star \in \{0,1\}^p$ only, similar to the work of Gamarnik and Zadik (2017) on sparse binary regression. Moreover, we learn the optimal classifier from the data by solving the minimization problem

$$\min_{\boldsymbol{w} \in \{0,1\}^p} \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\left(\hat{y}_i(\boldsymbol{w}) \neq y_i\right) \text{ s.t. } \|\boldsymbol{w}\|_0 = k, \tag{13}$$

where the loss function above corresponds to the empirical misclassification rate. Even though it is not a tractable loss function choice in practice, it demonstrates some interesting theoretical properties: it isolates the probabilistic model used to generate the data from the behavior of the optimal value. Indeed, for any classifier $\boldsymbol{w}$, the empirical misclassification rate $\sum_{i=1}^{n} \mathbf{1}\left(\hat{y}_i(\boldsymbol{w}) \neq y_i\right)$ follows a Binomial distribution, as the covariate data are independent. In addition, the problem (13) can be considered as the authentic formulation for binary classification, while other loss functions used in practice such as Hinge and logistic loss are only smooth proxies for the misclassification rate, used for their tractability and statistical consistency (Steinwart 2002, Zhang 2004).

4.2. Intuition and statement on sufficient conditions

For a given binary classifier $w \in \{0,1\}^p$ of sparsity k, the accuracy of the classifier (the number of true features it selects) is equal to the inner product of $\boldsymbol{w}$ with $\boldsymbol{w}^\star$:

$$A(\boldsymbol{w}) = |\{j : w_j \neq 0, \boldsymbol{w}_j^\star \neq 0\}| = |\{j : w_j = 1, \boldsymbol{w}_j^\star = 1\}| = \sum_j w_j w_j^\star = \boldsymbol{w}^\top \boldsymbol{w}^\star.$$

Consider a binary sparse classifier $\boldsymbol{w}$, i.e., $\|\boldsymbol{w}\|_0 = k$, with accuracy $\boldsymbol{w}^\top \boldsymbol{w}^\star = \ell$. Then, it follows that the indicators $\mathbf{1}\left(\hat{y}_i(\boldsymbol{w}) \neq y_i\right)$ are distributed as independent Bernoulli random variable sharing the success parameter

$$q(\ell; k, \sigma^2) := \mathbb{P}\left(\text{sign}\left(\boldsymbol{x}^{(i)\top} \boldsymbol{w}\right) \neq \text{sign}\left(\boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star + \varepsilon_i\right)\right),$$
$$= \frac{1}{\pi} \arccos\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right).$$

The success parameter $q_\ell = q(\ell; k, \sigma^2)$ can be checked to be a decreasing concave function of ℓ. That is, the more accurate our binary classifier $\boldsymbol{w}$, the smaller the probability of misclassification. The previous should come as no surprise to anybody. The central limit theorem states that

$$\sqrt{n}\left[\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\left(\hat{y}_i(\boldsymbol{w}) \neq y_i\right) - q_l\right] \to \mathcal{N}\left(0, q_\ell(1 - q_\ell)\right),$$

as $n \to \infty$. In words, asymptotically in n, a given classifier $\boldsymbol{w}$ will have an empirical misclassification rate close to q_ℓ. Since q_ℓ is decreasing in ℓ, the truth $\boldsymbol{w}^\star$ for which $\ell = k$ should minimize the misclassification error among all possible supports. As observed empirically, the number of true features selected corresponds to the true sparsity when n is sufficiently large (see Figures 2). Intuitively, n should be high enough such that the variance on the performance of each support $\frac{q_\ell(1-q_\ell)}{n}$ is small, taken into account that there are $\binom{k}{\ell}\binom{p-k}{k-\ell}$ possible supports with exactly ℓ correct features. In this case, it should be rather unlikely that the binary classifier with the smallest empirical misclassification rate is anything other than the ground truth $\boldsymbol{w}^\star$. We will now make the previous intuitive argument more rigorous.

Because we aim at minimizing the misclassification rate, we are guaranteed to recover the true support $\boldsymbol{w}^\star$ if there exists no other support $\boldsymbol{w}$ with an empirical performance at least as good as the truth.

THEOREM 3. *We assume the data is generated according to the equation $y_i = sign\left(\boldsymbol{x}^{(i)\top}\boldsymbol{w}^\star + \varepsilon_i\right)$, where x_i are i.i.d. standard random variables, $\varepsilon_i \sim \mathcal{N}(0,\sigma^2)$, $\boldsymbol{w}^\star \in \{0,1\}^p$ with $\|\boldsymbol{w}^\star\|_0 = k$. Given a classifier $\boldsymbol{w}$, we denote $\hat{y}_i(\boldsymbol{w}) := sign\left(\boldsymbol{x}^{(i)\top}\boldsymbol{w}\right)$. For any two binary classifiers $\boldsymbol{w}^1$ and $\boldsymbol{w}^2$, let $\Delta(\boldsymbol{w}^1, \boldsymbol{w}^2)$ denote the difference between their empirical performance, i.e.,*

$$\Delta(\boldsymbol{w}^1, \boldsymbol{w}^2) := \frac{1}{n}\sum_{i=1}^n \boldsymbol{1}\left(\hat{y}_i(\boldsymbol{w}^1) \neq y_i\right) - \boldsymbol{1}\left(\hat{y}_i(\boldsymbol{w}^2) \neq y_i\right).$$

Assume that $p \geqslant 2k$. Then there exist a threshold $n_0 > 0$ such that for any $n > n_0$,

$$\mathbb{P}\left(\exists \boldsymbol{w} \neq \boldsymbol{w}^\star, \, \boldsymbol{w} \in \{0,1\}^p, \, \|\boldsymbol{w}\|_0 = k \text{ s.t. } \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right) \leqslant e^{-\frac{n-n_0}{2\pi^2 k(\sigma^2+2)}}.$$

Moreover, we have $n_0 < C\left(2+\sigma^2\right)k\log(p-k)$ for some absolute constant $C > 0$.

In other words, if $n \geqslant C\left(2+\sigma^2\right)k\log(p-k) + 2\pi k(\sigma^2+2)\log(1/\delta)$ for some $\delta \in (0,1)$, then

$$\mathbb{P}\left(\exists \boldsymbol{w} \neq \boldsymbol{w}^\star, \, \boldsymbol{w} \in \{0,1\}^p, \, \|\boldsymbol{w}\|_0 = k \text{ s.t. } \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right) \leqslant \delta.$$

The proof of Theorem 3 is given in Appendix A. From a high-level perspective, our sufficient condition on n relies on two ingredients: (a) the union bound, which accounts for the log-dependence in $p - k$ and which is found in similar results for regression and signal processing (Wainwright 2009a, Wang et al. 2010) and (b) controlling of the individual probability $\mathbb{P}\left(\Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right)$ using large deviation bounds, which depends on the size of $\boldsymbol{w}$ and $\boldsymbol{w}^\star$, k, and the noise σ^2.

Before comparing the claim of Theorem 3 with similar results from statistics and signal processing, let us remember that k, the sparsity level of the true classifier $\boldsymbol{w}^\star$, is assumed to be known. To put this assumption in perspective with our previous simulations, our statement only concerns the best achievable accuracy when k is fixed.

4.3. Discussion

For regression, Gamarnik and Zadik (2017) proved that support recovery was possible from an information-theoretic point of view if

$$n > n^\star = \frac{2k \log p}{\log \left(\dfrac{2k}{\sigma^2} + 1 \right)}.$$

Note that our threshold n_0 for classification does not vanish in the low noise σ^2 setting. This observation is not surprising: the output y_i depending only on the sign of $\boldsymbol{w}^\top \boldsymbol{x}^{(i)}$ can be considered as inherently noisy. An observation already made by Scarlett and Cevher (2017).

As mentioned earlier, recent works in 1-bit compressed sensing have developed algorithms to recover sparse classifiers which provably recover the truth as the number of samples increases (Gupta et al. 2010, Plan and Vershynin 2013b,a). In particular, Plan and Vershynin (2013b) formulate the problem of learning $\boldsymbol{w}$ from the observations as a convex problem and establish bounds on the ℓ_2 error $\|\boldsymbol{w} - \boldsymbol{w}^\star\|_2$, in the case of logistic regression. In particular, they show that $n > C' k \log(2p/k)$ is sufficient to achieve an ℓ_2 reconstruction error which is bounded. In contrast with our result, they exhibited an algorithm able to achieve low error in terms of ℓ_2 distance with fewer samples than what we proved to be an information-theoretic sufficient condition. Yet, this does

Bertsimas, Pauphilet and Van Parys: *Sparse Classification: A Scalable Discrete Optimization Perspective*

33

not trivialize our result. ℓ_2 consistency is a related but distinct criterion for support recovery. Even a good estimate of $\boldsymbol{w}^\star$ in terms of ℓ_2 distance might have a very different support. The fact that their rate for ℓ_2 consistency is faster than the rate for support recovery suggests that achieving a good ℓ_2 error is presumably an easier problem. Similar observations were made in the case of linear regression in Wainwright (2009a) and intuitively explained: given a good support recovery procedure, one can restrict the number of features, use standard methods to estimate the values of the w_j's and hope to achieve a good ℓ_2 error, while the reverse might not be true. Loh et al. (2017) prove that non-convex regularization achieves perfect support recovery with high probability if $n \gtrsim k \log p$, which matches our bound up to a multiplicative constant. Together, our results suggest that the information-theoretic bounds are achievable in practice via a discrete optimization formulation (1) or non-convex penalties (3). Yet, they define $\boldsymbol{w}^\star$ implicitly as the solution $\min_{\boldsymbol{w}} \mathbb{E}[\mathcal{L}_n(\boldsymbol{w})]$, where $\mathcal{L}_n$ is the empirical loss over n observations, whereas we posit a generative model which explicitly depends on $\boldsymbol{w}^\star$. As a result, their model has the merit of being more general and covering many different statistical learning problems (e.g., linear regression, generalized linear models, sparse inverse covariance estimation). Unlike ours, however, it does isolate explicitly the impact of noise and does not capture the loss of information inherent in classification where discrete signals are generated from continuous inputs.

Finally, Scarlett and Cevher (2017) proved similar sufficient conditions for support recovery in 1-bit compressed sensing and accompanied them with necessary conditions and constants $C > 0$ as tight as possible. To that extent, their result (Corollary 3) might appear stronger than ours. However, their condition is valid only for low sparsity and low signal-to-noise regimes. Theorem 3, on the other hand, remains valid for all values of k and σ. In particular, it holds even if k scales linearly in p and σ is low: a regime for which Scarlett and Cevher (2017) provide necessary (Corollary 4) but no sufficient conditions. More precisely, they prove that perfect support recovery cannot be achieved if the number of samples is below a threshold scaling as $p\sqrt{\log p}$, while our bound scales as $p \log p$ in this regime. Combined together, there is a $\sqrt{\log p}$ factor between necessary

and sufficient conditions, which hints at the absence of a clear phase transition in this setting. As illustrated in Figure 1, there is an intermediate sample size regime where support recovery is neither provably impossible nor achievable. Of course, this regime could be a deficiency of the proof techniques used, but in any case, we believe it constitutes an exciting direction for future research.

5. Conclusion

In this paper, we have proposed a tractable binary convex optimization algorithm for solving sparse classification. Although the problem is theoretically NP-hard, our algorithm scales for logistic regression and SVM in problems with n, p in $10,000$s within minutes. We also introduce a stochastic version of the constraint generation process which further reduces computational time by a factor 2 to 10 on numerical experiments. Comparing our method and Lasso-based estimates, we observe empirically that as n increases, the number of true features selected by both methods converges to the true sparsity. We support our observations with information-theoretic sufficient conditions, stating that support recovery is achievable as soon as $n > n_0$, with $n_0 < C \left(2 + \sigma^2\right) k \log(p - k)$ for some positive constant C. This sufficient information-theoretic condition echoes and complements existing results in the literature for 1-bit compressed sensing. Apart from accuracy, the exact sparse formulation has an edge over Lasso in the number of false features: as n increases, the number of false features selected by our method converges to zero, while this is not observed for Lasso. This phenomenon is also observed for classifying the type of cancer using gene expression data from the Cancer Genome Atlas Research Network with $n = 1,145$ lung cancer patients and $p = 14,858$ genes. Sparse classification using logistic and hinge loss returns a classifier based on 90 and 38 genes respectively compared with 378.4 genes for ElasticNet with similar predictive accuracy.

Appendix A: Proof of the sufficient condition for a support recovery

A.1. Preliminary results on orthant probabilities

Let us recall an analytical expression for the probability that a bivariate normal distribution assigns to the positive orthant.

LEMMA 2. *(Cramér 2016, p. 290) Assume we are given a zero mean bivariate normal random variable (n_1, n_2) with $\mathbb{E}[n_1^2] = \mathbb{E}[n_2^2] = 1$ and covariance $\rho_{12} = \mathbb{E}[n_1 n_2]$. Then,*

$$\mathbb{P}(n_1 \geqslant 0, n_2 \geqslant 0) = \frac{1}{2\pi}\left(\frac{\pi}{2} + \arcsin(\rho_{12})\right).$$

By continuity of the density function of normal distributions, the probability of the positive orthant and its interior are equivalent. Lemma 1 is an almost direct consequence of the previous result. We give here its proof.

Proof of Lemma 1 We can separate the event of interest in two disjunct cases as

$$\mathbb{P}\left(\operatorname{sign}\left(\boldsymbol{x}^\top \boldsymbol{w}\right) \neq \operatorname{sign}\left(\boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon\right)\right)$$
$$= \mathbb{P}\left(\boldsymbol{x}^\top \boldsymbol{w} \leqslant 0, \boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon > 0\right) + \mathbb{P}\left(\boldsymbol{x}^\top \boldsymbol{w} > 0, \boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon \leqslant 0\right).$$

Each term corresponds to the probability that a zero mean bivariate normal variable $(\boldsymbol{x}^\top \boldsymbol{w}, \boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon)$ realizes in an appropriate orthant. We define the random variables $n_1 := \boldsymbol{x}^\top \boldsymbol{w} / \|\boldsymbol{w}\|_2$ and $n_2 := (\boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon)/\sqrt{\|\boldsymbol{w}'\|_2^2 + \sigma^2}$ and obtain

$$\mathbb{P}\left(\operatorname{sign}\left(\boldsymbol{x}^\top \boldsymbol{w}\right) \neq \operatorname{sign}\left(\boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon\right)\right) = \mathbb{P}(n_1 \leq 0, n_2 > 0) + \mathbb{P}(n_1 > 0, n_2 \leq 0).$$

We have that $\mathbb{E}[n_1^2] = 1$, $\mathbb{E}[n_2^2] = 1$ and $\rho_{12} = \mathbb{E}[n_1 n_2] = \boldsymbol{w}^\top \boldsymbol{w}' / (\|\boldsymbol{w}\|_2 \sqrt{\|\boldsymbol{w}'\|_2^2 + \sigma^2})$. Using the analytical expressions of such orthant probabilities for bivariate normal random variables given in Lemma 2, we have hence $\mathbb{P}\left(\operatorname{sign}\left(\boldsymbol{x}^\top \boldsymbol{w}\right) \neq \operatorname{sign}\left(\boldsymbol{x}^\top \boldsymbol{w}' + \varepsilon\right)\right) = \frac{1}{\pi}\left(\frac{\pi}{2} - \arcsin(\rho_{12})\right) = \frac{1}{\pi}\arccos(\rho_{12})$. Q.E.D.

We will need a minor generalization of Lemma 2 to the three dimensional case in the proof of Theorem 3.

LEMMA 3. *(Cramér 2016, p. 290) Assume we are given a zero mean trivariate normal random variable (n_1, n_2, n_3) with $\mathbb{E}[n_1^2] = \mathbb{E}[n_2^2] = \mathbb{E}[n_3^2] = 1$ among which we have covariances $\rho_{12} = \mathbb{E}[n_1 n_2]$, $\rho_{13} = \mathbb{E}[n_1 n_3]$ and $\rho_{23} = \mathbb{E}[n_2 n_3]$. Then,*

$$\mathbb{P}(n_1 \geqslant 0, n_2 \geqslant 0, n_3 \geqslant 0) = \frac{1}{4\pi}\left(\frac{\pi}{2} + \arcsin(\rho_{12}) + \arcsin(\rho_{13}) + \arcsin(\rho_{23})\right).$$

A.2. Comparative performance of a given support with the truth

We first prove a large deviation bound for $\mathbb{P}(\Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0)$ for any given binary classifier $\boldsymbol{w}$, depending on the number of true features it selects. The following result can be derived using Hoeffdings inequality as illustrated in its proof.

LEMMA 4. *Let $\boldsymbol{w} \in \{0,1\}^p$ be a binary classifier such that $\|\boldsymbol{w}\|_0 = k$ and $\boldsymbol{w}^\top \boldsymbol{w}^\star = \ell \in \{0, \ldots, k\}$. Its misclassification rate with respect to the ground truth satisfies the exponential bound*

$$\mathbb{P}(\Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0) \leqslant \exp\left(-n \frac{(k-\ell)^2}{2\pi^2(k(k+\sigma^2) - \ell^2)}\right).$$

Proof of Lemma 4   Let us consider a binary classifier $\boldsymbol{w} \in \{0,1\}^p$ with sparsity $\|\boldsymbol{w}\|_0 = k$ and true features $\boldsymbol{w}^\top \boldsymbol{w}^\star = \ell$. We compare the empirical misclassification rate of $\boldsymbol{w}$ with the performance of the true support $\boldsymbol{w}^\star$. We take the misclassification rate with respect to the ground truth $\boldsymbol{w}^\star$ as

$$\Delta(\boldsymbol{w}, \boldsymbol{w}^\star) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(\hat{y}_i(\boldsymbol{w}) \neq y_i) - \mathbf{1}(\hat{y}_i(\boldsymbol{w}^\star) \neq y_i) =: \frac{1}{n} \sum_{i=1}^n Z_i$$

which is composed of the sum of independent random variables Z_i taking values in $\{-1, 0, 1\}$ such that

$$Z_i = \begin{cases} +1, & \text{if } y_i = \hat{y}_i(\boldsymbol{w}^\star) \neq \hat{y}_i(\boldsymbol{w}), \\ -1, & \text{if } y_i = \hat{y}_i(\boldsymbol{w}) \neq \hat{y}_i(\boldsymbol{w}^\star), \\ 0, & \text{otherwise} \end{cases}$$

Each random variable y_i, $\hat{y}_i(\boldsymbol{w}^\star)$, $\hat{y}_i(\boldsymbol{w})$ is the sign of the normally distributed quantities $\boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star + \varepsilon_i$, $\boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star$ and $\boldsymbol{x}^{(i)\top} \boldsymbol{w}$ respectively. Let us define three zero mean random variables $n_1 = (\boldsymbol{x}^{(i)\top} \boldsymbol{w} + e_i)/\sqrt{k+\sigma^2}$, $n_2 = \boldsymbol{x}^{(i)\top} \boldsymbol{w}/\sqrt{k}$ and $n_3 = \boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star/\sqrt{k}$. Their covariance structure is characterized as $\rho_{12} = \mathbb{E}[n_1 n_2] = k/\sqrt{k(k+\sigma^2)}$, $\rho_{23} = \mathbb{E}[n_2 n_3] = \ell/k$ and $\rho_{13} = \mathbb{E}[n_1 n_3] = \ell/\sqrt{k(k+\sigma^2)}$. We can then express the probabilities of each value of Z_i as tridimensional orthant probabilities for these three zero mean correlated normal random variables and use the analytical expression given in Lemma 3. We hence arrive at

$$\begin{aligned} \mathbb{P}(Z_i = 1) &= \mathbb{P}\left(\boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star + \epsilon_i \geq 0, \boldsymbol{x}^{(i)\top} \boldsymbol{w} \leqslant 0, \boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star \geqslant 0\right) \\ &\quad + \mathbb{P}\left(\boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star + \epsilon_i \leq 0, \boldsymbol{x}^{(i)\top} \boldsymbol{w} \geqslant 0, \boldsymbol{x}^{(i)\top} \boldsymbol{w}^\star \leqslant 0\right) \\ &= \mathbb{P}(n_1 \geq 0, n_2 \leq 0, n_3 \geqslant 0) + \mathbb{P}(n_1 \leqslant 0, n_2 \geq 0, n_3 \leq 0) \\ &= \frac{1}{2\pi}\left[\frac{\pi}{2} - \arcsin\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right) + \arcsin\left(\frac{k}{\sqrt{k(k+\sigma^2)}}\right) - \arcsin\left(\frac{\ell}{k}\right)\right], \end{aligned}$$

and equivalently

$$
\begin{aligned}
\mathbb{P}\left(Z_i = -1\right) &= \mathbb{P}\left({\boldsymbol{x}^{(i)}}^{\top}\boldsymbol{w}^{\star}+\epsilon_i \geq 0, {\boldsymbol{x}^{(i)}}^{\top}\boldsymbol{w} \geqslant 0, {\boldsymbol{x}^{(i)}}^{\top}\boldsymbol{w}^{\star} \leqslant 0\right) \\
&\quad + \mathbb{P}\left({\boldsymbol{x}^{(i)}}^{\top}\boldsymbol{w}^{\star}+\epsilon_i \leq 0, {\boldsymbol{x}^{(i)}}^{\top}\boldsymbol{w} \leqslant 0, {\boldsymbol{x}^{(i)}}^{\top}\boldsymbol{w}^{\star} \geqslant 0\right) \\
&= \mathbb{P}\left(n_1 \geq 0, n_2 \geq 0, n_3 \leq 0\right) + \mathbb{P}\left(n_1 \leq 0, n_2 \leq 0, n_3 \geq 0\right) \\
&= \frac{1}{2\pi}\left[\frac{\pi}{2} + \arcsin\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right) - \arcsin\left(\frac{k}{\sqrt{k(k+\sigma^2)}}\right) - \arcsin\left(\frac{\ell}{k}\right)\right].
\end{aligned}
$$

Evidently, we can characterize the probability of $Z_i = 0$ as $\mathbb{P}(Z_i = 0) = 1 - \mathbb{P}(Z_i = 1) - \mathbb{P}(Z_i = -1)$. The mean of Z_i is now easily found as the expression

$$
\mathbb{E}[Z_i] = \frac{1}{\pi}\left[\arccos\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right) - \arccos\left(\frac{k}{\sqrt{k(k+\sigma^2)}}\right)\right].
$$

Concavity of the arccos function on the interval $[0,1]$ enables us to state the gradient inequalities

$$
\begin{aligned}
\arccos\left(\frac{k}{\sqrt{k(k+\sigma^2)}}\right) &\leqslant \arccos\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right) + \arccos\boldsymbol{s}'\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right)\frac{k-\ell}{\sqrt{k(k+\sigma^2)}} \\
&= \arccos\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right) - \sqrt{\frac{k(k+\sigma^2)}{k(k+\sigma^2)-\ell^2}}\frac{k-\ell}{\sqrt{k(k+\sigma^2)}} \\
&= \arccos\left(\frac{\ell}{\sqrt{k(k+\sigma^2)}}\right) - \frac{k-\ell}{\sqrt{k(k+\sigma^2)-\ell^2}}.
\end{aligned}
$$

We thus obtain a somewhat simple lower bound on the mean of Z_i

$$
\mathbb{E}[Z_i] \geqslant \frac{1}{\pi}\frac{k-\ell}{\sqrt{k(k+\sigma^2)-\ell^2}}.
$$

We now have all the ingredients to upper-bound the probability that $\boldsymbol{w}$ performs strictly better than $\boldsymbol{w}^{\star}$, in other words that $\Delta(\boldsymbol{w},\boldsymbol{w}^{\star}) := \sum_{i=1}^{n} Z_i < 0$. Applying Hoeffding's inequality for independent random variables supported on $[-1,1]$, we have for any $t > 0$

$$
\mathbb{P}\left(\sum_{i=1}^{n}(Z_i - \mathbb{E}[Z_i]) < -nt\right) \leqslant \exp\left(-\frac{nt^2}{2}\right),
$$

and taking $t = \mathbb{E}[Z]$, which is non negative for $\ell < k$ because arccos is decreasing on $[0,1]$, leads to $\mathbb{P}\left(\Delta(\boldsymbol{w},\boldsymbol{w}^{\star}) < 0\right) \leqslant \exp\left(-\frac{n}{2}\mathbb{E}[Z_i]^2\right)$. Substituting in the previous expression our lower bound for the mean $\mathbb{E}[Z_i]$ gives the desired result. Q.E.D.

REMARK 2. In the absence of noise ($\sigma = 0$), the bound in Lemma 4 can be improved. Indeed, the truth makes no mistakes ($\hat{y}_i(\boldsymbol{w}^{\star}) = y_i$, $\forall i$) and $\Delta(\boldsymbol{w},\boldsymbol{w}^{\star}) \geqslant 0$ for any $\boldsymbol{w}$. More precisely, here $\sum_{i=1}^{n} \mathbf{1}\left(\hat{y}_i(\boldsymbol{w}) \neq y_i\right)$ is a binomial random variable with parameters n and $q(\ell;k,0)$. Therefore,

$$
\mathbb{P}\left(\Delta(\boldsymbol{w},\boldsymbol{w}^{\star}) \leqslant 0\right) = \mathbb{P}\left(\Delta(\boldsymbol{w},\boldsymbol{w}^{\star}) = 0\right) = \left(1 - \frac{1}{\pi}\arccos\left(\tfrac{\ell}{k}\right)\right)^{n}.
$$

Using concavity of the logarithm and the arccos function successively yields the upper bound

$$\mathbb{P}\left(\Delta(\boldsymbol{w}, \boldsymbol{w}^\star) < 0\right) \leqslant \exp\left(-\frac{n}{\pi}\sqrt{\frac{k-\ell}{k+\ell}}\right).$$

However, such a refinement eventually modifies the result in Theorem 3 by a constant multiplicative factor only.

A.3. Proof Theorem 3

Proof of Theorem 3   We are interested in bounding the probability that the binary classifier with minimal empirical misclassification rate is any other than $\boldsymbol{w}^\star$. We can characterize the probability of such event as $\mathbb{P}\left(\exists \boldsymbol{w} \neq \boldsymbol{w}^\star \text{ s.t. } \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right)$. Evidently,

$$\mathbb{P}\left(\exists \boldsymbol{w} \neq \boldsymbol{w}^\star \text{ s.t. } \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right) = \sum_{\ell \in \{0,\dots,k-1\}} \mathbb{P}\left(\exists \boldsymbol{w} \text{ s.t. } \boldsymbol{w}^\top \boldsymbol{w}^\star = \ell, \ \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right).$$

Recall that there are exactly $\binom{k}{\ell}\binom{p-k}{k-l}$ distinct binary classifiers $\boldsymbol{w}$ with accuracy $\boldsymbol{w}^\top \boldsymbol{w}^\star = \ell$. Combining a union bound and the bound from Lemma 4 yields

$$\mathbb{P}\left(\exists \boldsymbol{w} \neq \boldsymbol{w}^\star \text{ s.t. } \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right) \leqslant \sum_{\ell=0}^{k-1} \binom{k}{\ell}\binom{p-k}{k-\ell} \exp\left(-n\frac{(k-\ell)^2}{2\pi^2(k(k+\sigma^2)-\ell^2)}\right), \qquad (14)$$

$$\leqslant k \cdot \max_{\ell=0,\dots,k-1} \binom{k}{\ell}\binom{p-k}{k-\ell} \exp\left(-n\frac{(k-\ell)^2}{2\pi^2(k(k+\sigma^2)-\ell^2)}\right). \quad (15)$$

In order for the previous error probability to be bounded away from one, it suffice to take n greater than a threshold T

$$n > T := \max_{\ell=0,\dots,k-1} \frac{2\pi^2(k(k+\sigma^2)-\ell^2)}{(k-\ell)^2}\left[\log k + \log\binom{k}{\ell} + \log\binom{p-k}{k-\ell}\right].$$

We can obtain a more interpretable sufficient condition by upper-bounding the threshold T. Assuming $p \geqslant 2k$, $\binom{k}{\ell} = \binom{k}{k-\ell} \leqslant \binom{p-k}{k-\ell}$ and $k \leqslant \binom{p-k}{k-\ell}$, so that

$$T \lesssim \max_{\ell=0,\dots,k-1} \frac{k(k+\sigma^2)-\ell^2}{(k-\ell)^2}\log\binom{p-k}{k-\ell},$$

where $\lesssim$ signifies that the inequality holds up to a multiplicative factor. Since $\log\binom{p-k}{k-\ell} \lesssim (k-\ell)\log\left(\frac{p-k}{k-\ell}\right)$, we now have

$$T \lesssim \max_{\ell=0,\dots,k-1} \frac{k(k+\sigma^2)-\ell^2}{(k-\ell)}\log\left(\frac{p-k}{k-\ell}\right),$$

The maximum over ℓ in right hand side of the previous inequality occurs when $\ell = k-1$. The previous observation yields hence that $T \lesssim (2+\sigma^2)k\log(p-k)$. Finally, it is easy to verify that when n exceeds some threshold value $n_0 \geqslant T$, the inequality (15) yields

$$\mathbb{P}\left(\exists \boldsymbol{w} \neq \boldsymbol{w}^\star \text{ s.t. } \Delta(\boldsymbol{w}, \boldsymbol{w}^\star) \leqslant 0\right) \leqslant \max_{\ell=0,\dots,k-1} \exp\left(-\frac{(k-\ell)^2}{2\pi^2(k(k+\sigma^2)-\ell^2)}(n-n_0)\right),$$

$$\leqslant \exp\left(-\frac{n-n_0}{2\pi^2 k(2+\sigma^2)}\right).$$

Q.E.D.

References

Atamturk A, Gomez A (2019) Rank-one convexification for sparse regression. *arXiv preprint arXiv:1901.10334* .

Bach F (2009) High-dimensional non-linear variable selection through hierarchical kernel learning. *arXiv preprint arXiv:0909.0844* .

Bach F, Jenatton R, Mairal J, Obozinski G (2012) Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning* 4(1):1–106.

Bertsekas DP (1982) Projected newton methods for optimization problems with simple constraints. *SIAM Journal on Control and Optimization* 20(2):221–246.

Bertsimas D, Copenhaver MS (2018) Characterization of the equivalence of robustification and regularization in linear, median, and matrix regression. *European Journal of Operations Research* 270(3):931–942.

Bertsimas D, King A (2017) Logistic regression: From art to science. *Statistical Science* 367–384.

Bertsimas D, King A, Mazumder R (2016) Best subset selection via a modern optimization lens. *Annals of Statistics* 44(2):813–852.

Bertsimas D, Pauphilet J, Van Parys B (2020a) Sparse regression: Scalable algorithms and empirical performance. *Statistical Science* 35(4):555–578.

Bertsimas D, Van Parys B, et al. (2020b) Sparse high-dimensional regression: Exact scalable algorithms and phase transitions. *Annals of Statistics* 48(1):300–323.

Bonami P, Biegler LT, Conn AR, Cornuéjols G, Grossmann IE, Laird CD, Lee J, Lodi A, Margot F, Sawaya N, et al. (2008) An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization* 5(2):186–204.

Boufounos PT, Baraniuk RG (2008) 1-bit compressive sensing. *Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on*, 16–21 (IEEE).

Boyd S, Vandenberghe L (2004) *Convex optimization* (Cambridge university press).

Breheny P, Huang J (2011) Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics* 5(1):232.

Bühlmann P (2011) Invited discussion on "regression shrinkage and selection via the lasso: a retrospective" (r. tibshirani). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73(3):273–282.

Calamai PH, Moré JJ (1987) Projected gradient methods for linearly constrained problems. *Mathematical Programming* 39(1):93–116.

Chen Y, Ye Y, Wang M (2019) Approximation hardness for a class of sparse optimization problems. *Journal of Machine Learning Research* .

Chu BY, Ho CH, Tsai CH, Lin CY, Lin CJ (2015) Warm start for parameter selection of linear classifiers. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 149–158 (ACM).

Cortes C, Vapnik V (1995) Support-vector networks. *Machine Learning* 20(3):273–297.

CPLEX II (2011) Cplex users manual.

Cramér H (2016) *Mathematical Methods of Statistics (PMS-9)*, volume 9 (Princeton university press).

Dash M, Liu H (1997) Feature selection for classification. *Intelligent data analysis* 1(1-4):131–156.

Dedieu A, Hazimeh H, Mazumder R (2021) Learning sparse classifiers: Continuous and mixed integer optimization perspectives. *Journal of Machine Learning Research* 22(135):1–47.

Donoho D, Stodden V (2006) Breakdown point of model selection when the number of variables exceeds the number of observations. *International Joint Conference on Neural Networks*, 1916–1921 (IEEE).

Donoho D, Tanner J (2009) Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 367(1906):4273–4293.

Dunning I, Huchette J, Lubin M (2017) JuMP: A modeling language for mathematical optimization. *SIAM Review* 59(2):295–320.

Duran MA, Grossmann IE (1986) An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming* 36(3):307–339.

Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association* 96(456):1348–1360.

Fan J, Liu H, Sun Q, Zhang T (2018) I-lamm for sparse learning: Simultaneous control of algorithmic complexity and statistical error. *Annals of Statistics* 46(2):814.

Fan J, Song R, et al. (2010) Sure independence screening in generalized linear models with np-dimensionality. *Annals of Statistics* 38(6):3567–3604.

Fischetti M, Ljubić I, Sinnl M (2017) Redesigning benders decomposition for large-scale facility location. *Management Science* 63(7):2146–2162.

Fletcher R, Leyffer S (1994) Solving mixed integer nonlinear programs by outer approximation. *Mathematical Programming* 66(1-3):327–349.

Frank L, Friedman J (1993) A statistical view of some chemometrics regression tools. *Technometrics* 35(2):109–135.

Friedman J, Hastie T, Tibshirani R (2010) Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software* 33(1):1.

Friedman J, Hastie T, Tibshirani R (2013) GLMNet: Lasso and elastic-net regularized generalized linear models. r package version 1.9–5.

Gamarnik D, Zadik I (2017) High-dimensional regression with binary coefficients. estimating squared error and a phase transition. *arXiv preprint arXiv:1701.04455* .

Goldfarb D (1994) On the complexity of the simplex method. *Advances in optimization and numerical analysis*, 25–38 (Springer).

Gupta A, Nowak R, Recht B (2010) Sample complexity for 1-bit compressed sensing and sparse classification. *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*, 1553–1557 (IEEE).

Gurobi Optimization I (2016) Gurobi optimizer reference manual. URL `http://www.gurobi.com`.

Guyon I, Weston J, Barnhill S, Vapnik V (2002) Gene selection for cancer classification using support vector machines. *Machine Learning* 46(1-3):389–422.

Hastie T, Tibshirani R, Tibshirani R (2020) Best subset, forward stepwise or lasso? analysis and recommendations based on extensive comparisons. *Statistical Science* 35(4):579–592.

Hazimeh H, Mazumder R (2020) Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms. *Operations Research* 68(5):1517–1537.

Hsieh CJ, Chang KW, Lin CJ, Keerthi SS, Sundararajan S (2008) A dual coordinate descent method for large-scale linear svm. *Proceedings of the 25th International Conference on Machine Learning*, 408–415 (ACM).

Hunter D, Li R (2005) Variable selection using MM algorithms. *Annals of Statistics* 33(4):1617.

Jacques L, Laska JN, Boufounos PT, Baraniuk RG (2013) Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. *IEEE Transactions on Information Theory* 59(4):2082–2102.

Keerthi SS, Duan KB, Shevade SK, Poo AN (2005) A fast dual algorithm for kernel logistic regression. *Machine Learning* 61(1-3):151–165.

Kelley JE Jr (1960) The cutting-plane method for solving convex programs. *Journal of the society for Industrial and Applied Mathematics* 8(4):703–712.

Kenney A, Chiaromonte F, Felici G (2018) Efficient and effective l_0 feature selection. *arXiv preprint arXiv:1808.02526* .

Lin CJ, Weng RC, Keerthi SS (2008) Trust region newton method for logistic regression. *Journal of Machine Learning Research* 9(Apr):627–650.

Loh PL, Wainwright M (2015) Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *The Journal of Machine Learning Research* 16(1):559–616.

Loh PL, Wainwright MJ, et al. (2017) Support recovery without incoherence: A case for nonconvex regularization. *Annals of Statistics* 45(6):2455–2482.

Lubin M, Dunning I (2015) Computing in operations research using Julia. *INFORMS Journal on Computing* 27(2):238–248.

Mazumder R, Friedman J, Hastie T (2011) Sparsenet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association* 106(495):1125–1138.

Natarajan BK (1995) Sparse approximate solutions to linear systems. *SIAM journal on computing* 24(2):227–234.

Pilanci M, Wainwright MJ, El Ghaoui L (2015) Sparse learning via boolean relaxations. *Mathematical Programming* 151(1):63–87.

Plan Y, Vershynin R (2013a) One-bit compressed sensing by linear programming. *Communications on Pure and Applied Mathematics* 66(8):1275–1297.

Plan Y, Vershynin R (2013b) Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach. *IEEE Transactions on Information Theory* 59(1):482–494.

Scarlett J, Cevher V (2017) Limits on support recovery with probabilistic models: An information-theoretic framework. *IEEE Transactions on Information Theory* 63(1):593–620.

Scholkopf B, Smola AJ (2001) *Learning with kernels: support vector machines, regularization, optimization, and beyond* (MIT press).

Shalev-Shwartz S, Singer Y, Srebro N, Cotter A (2011) Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical Programming* 127(1):3–30.

Steinwart I (2002) Support vector machines are universally consistent. *Journal of Complexity* 18(3):768–791.

Su W, Bogdan M, Candes E (2017) False discoveries occur early on the lasso path. *Annals of Statistics* 2133–2150.

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

Vapnik V (1998) The support vector method of function estimation. *Nonlinear Modeling*, 55–85 (Springer).

Wainwright MJ (2009a) Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting. *IEEE Transactions on Information Theory* 55(12):5728–5741.

Wainwright MJ (2009b) Sharp thresholds for high-dimensional and noisy sparsity recovery using-constrained quadratic programming (Lasso). *IEEE Transactions on Information Theory* 55(5):2183–2202.

Wang W, Wainwright MJ, Ramchandran K (2010) Information-theoretic limits on sparse signal recovery: Dense versus sparse measurement matrices. *IEEE Transactions on Information Theory* 56(6):2967–2979.

Yu HF, Huang FL, Lin CJ (2011) Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning* 85(1-2):41–75.

Zhang CH (2010a) Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics* 38(2):894–942.

Zhang T (2004) Statistical behavior and consistency of classification methods based on convex risk minimization. *Annals of Statistics* 56–85.

Zhang T (2010b) Analysis of multi-stage convex relaxation for sparse regularizationa. *Journal of Machine Learning Research* 11(3).

Zhang T, et al. (2013) Multi-stage convex relaxation for feature selection. *Bernoulli* 19(5B):2277–2293.

Zou H, Li R (2008) One-step sparse estimates in nonconcave penalized likelihood models. *Annals of Statistics* 36(4):1509.