Journal of Machine Learning Research 26 (2025) 1-38 Submitted 1/24; Revised 2/25; Published 6/25

Simplex Constrained Sparse Optimization via Tail Screening

Peng Chen chenpeng1@mail.ustc.edu.cn
Department of Statistics and Finance, School of Management
University of Science and Technology of China

Jin Zhu j.zhu69@lse.ac.uk
Department of Statistics, London School of Economics and Political Science

Junxian Zhu junxian@nus.edu.sg
Saw Swee Hock School of Public Health, National University of Singapore

Xueqin Wang wangxq20@ustc.edu.cn
Department of Statistics and Finance/International Institute of Finance, School of Management
University of Science and Technology of China

Editor: Ali Shojaie

Abstract

We consider the probabilistic simplex-constrained sparse recovery problem. The commonly used Lasso-type penalty for promoting sparsity is ineffective in this context since it is a constant within the simplex. Despite this challenge, fortunately, simplex constraint itself brings a self-regularization property, i.e., the empirical risk minimizer without any sparsity-promoting procedure obtains the usual Lasso-type estimation error. Moreover, we analyze the iterates of a projected gradient descent method and show its convergence to the ground truth sparse solution in the geometric rate until a satisfied statistical precision is attained. Although the estimation error is statistically optimal, the resulting solution is usually more dense than the sparse ground truth. To further sparsify the iterates, we propose a method called PERMITS via embedding a tail screening procedure, i.e., identifying negligible components and discarding them during iterations, into the projected gradient descent method. Furthermore, we combine tail screening and the special information criterion to balance the trade-off between fitness and complexity. Theoretically, the proposed PERMITS method can exactly recover the ground truth support set under mild conditions and thus obtain the oracle property. We demonstrate the statistical and computational efficiency of PERMITS with both synthetic and real data. The implementation of the proposed method can be found in `https://github.com/abess-team/PERMITS`.

Keywords: simplex constrained sparse recovery, projected gradient descent, self-regularization, tail screening, special information criterion

1. Introduction

In widespread applications, we observe n samples $(\boldsymbol{x}_i, y_i)$ subject to the following generation process

$$y_i = \boldsymbol{x}_i^\top \boldsymbol{w}^* + \xi_i,\ i = 1, \cdots, n$$

*. Peng Chen, Jin Zhu, and Junxian Zhu contributed equally. Xueqin Wang is the corresponding author.

where $\boldsymbol{x}_i \in \mathbb{R}^p$ is the feature vector and $y_i \in \mathbb{R}$ is the response value, $\boldsymbol{w}^* \in \Delta := \{\boldsymbol{w} \in \mathbb{R}^p : \mathbf{1}^\top \boldsymbol{w} = 1, \boldsymbol{w} \geq \mathbf{0}\}$ is the unknown coefficient vector lying in the probabilistic simplex and ξ_i is some random noise. The objective is to recover the unknown coefficient $\boldsymbol{w}^*$. Furthermore, this model can be succinctly represented as follows:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}^* + \boldsymbol{\xi}$$

where $(\boldsymbol{X}, \boldsymbol{y}) \in \mathbb{R}^{n \times p} \times \mathbb{R}^n$ are the observed feature matrix and response vector and $\boldsymbol{\xi} \in \mathbb{R}^n$ is the noise vector. This modeling framework finds extensive applications in various domains such as economics, finance (Benidis et al., 2017; Zheng et al., 2020), and machine learning (Keshava, 2003; Limmer and Stańczak, 2018). Typically, the coefficient vector $\boldsymbol{w}^*$ embodies weights. For example, in finance, $\boldsymbol{w}^*$ corresponds to a market index or portfolio weights that are assigned to p assets (Du et al., 2022).

Nowadays, researchers often collect high-dimensional data with p generally associated with the same or even larger scale than n. In the high-dimensional statistics , a common assumption is that the unknown coefficient vector $\boldsymbol{w}^*$ is sparse. In other words, although the dimension p of $\boldsymbol{w}^*$ greatly surpasses the sample size n, the majority of its components w_i^* are precisely zero. Therefore, the true sparsity level defined as

$$s^* := \|\boldsymbol{w}^*\|_0$$

is relatively small compared to n. This assumption is rational in the context of the contemporary era of big data, where an extensive array of features $X_j \in \mathbb{R}^n$ can be collected, yet only a fraction of them significantly influences $\boldsymbol{y}$. Such a prevalent scenario enables the derivation of effective estimators for $\boldsymbol{w}^*$ and potentially allows for the recovery of the corresponding support set

$$S^* := \operatorname{supp}(\boldsymbol{w}^*).$$

In this paper, our primary interest is to recover S^* and estimate $\boldsymbol{w}^*$. To estimate the unknown $\boldsymbol{w}^*$, a common approach is to solve the following constrained least squares problem

$$\min_{\boldsymbol{w} \in \mathbb{R}^p} f(\boldsymbol{w}) = \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2 \quad \text{s.t. } \boldsymbol{w} \in \Delta. \tag{1}$$

Let $\widehat{\boldsymbol{w}}_n$ be any solution of (1), and it may not be unique since f is not necessarily strictly convex over Δ. The simplex constraint prevents us from obtaining an explicit formula for $\widehat{\boldsymbol{w}}_n$, and it can only be approximated through iterative optimization algorithms (Jaggi, 2013; Xiao and Bai, 2022; Li et al., 2023) producing outputs like $\boldsymbol{w}^t$. In real-world scenarios, especially those involving high dimensions ($p \gg n$), two significant challenges emerge. Firstly, the estimation error $\|\widehat{\boldsymbol{w}}_n - \boldsymbol{w}^*\|_2$ can be substantial due to the presence of noise $\boldsymbol{\xi}$. Secondly, optimization becomes challenging because $f(\boldsymbol{w})$ loses its strong convexity when $p \gg n$. Consequently, achieving a desirable linear convergence rate of $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$ is difficult for most algorithms under these circumstances.

1.1 Regularized methods

In the high-dimensional literature, regularization (Tibshirani, 1996; Meier et al., 2008; Tibshirani et al., 2005) is commonly used to promote sparsity and alleviate the defect of noise accumulation. However, in the presence of a simplex constraint, regularization is not as simple as that in the unconstrained setting.

The most explicit method to promote sparsity is to directly penalize the cardinality of $\boldsymbol{w}$ and solve

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|_0 \quad \text{s.t. } \boldsymbol{w}\in\Delta$$

where $\|\boldsymbol{w}\|_0$ counts the number of nonzero components and λ is a given penalty coefficient specified by the user or determined by some data-driven methods. However, this is computationally intractable since $\|\cdot\|_0$ is non-convex and thus intractable in practice, especially when both n and p are large. A greedy method, iterative hard thresholding (IHT), is a famous method to solve such sparse recovery problem (Blumensath and Davies, 2009). It performs a hard thresholding procedure after gradient descent to force the sparsity of iterates. Kyrillidis et al. (2013) extended this method to solve (1) by replacing the original hard thresholding procedure with the sparse projection onto the simplex, which can force sparsity and feasibility simultaneously.

Motivated by the Lasso-type methods, researchers may consider approximating the non-convex ℓ_0 penalty with a convex ℓ_1 penalty, and it becomes

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|_1 \quad \text{s.t. } \boldsymbol{w}\in\Delta,$$

where $\lambda > 0$ is also a hyper-parameter controlling the degree of penalty and may be slightly different from the above one. Unfortunately, this method fails in the presence of a simplex constraint since the added penalty term $\lambda\|\boldsymbol{w}\|_1 = \lambda$ is a constant (independent of $\boldsymbol{w}$) for any $\boldsymbol{w}\in\Delta$. That is, we cannot promote the sparsity of the solution by tuning λ as those Lasso-type methods do.

To deal with this undesirable property that $\|\boldsymbol{w}\|_1 = 1$ for any $\boldsymbol{w}\in\Delta$, some heuristic methods were considered by firstly ignoring the equality constraint $\mathbf{1}^\top\boldsymbol{w} = 1$ in the simplex and solving the following non-negative Lasso

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|_1 \quad \text{s.t. } \boldsymbol{w}\geq\mathbf{0}$$

or a modified version (Du et al., 2022)

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) + \lambda\|\boldsymbol{w}\|_1 \quad \text{s.t. } \boldsymbol{w}\geq\mathbf{0}, \|\boldsymbol{w}\|_1\leq R$$

and then scaling the solution such that $\mathbf{1}^\top\boldsymbol{w} = 1$ holds (dividing the solution by its ℓ_1-norm). In the context of portfolio selection, this method was shown to behave well from both theoretical and practical perspectives. However, the theoretical guarantee relied on some additional conditions, such as the single crossing assumption, which is hard to check in practice; see Du et al. (2022) for more details.

Besides these heuristic methods, Pilanci et al. (2012) proposed to replace the ℓ_0-norm by the inverse ℓ_∞-norm

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) + \lambda\frac{1}{\|\boldsymbol{w}\|_\infty} \quad \text{s.t. } \boldsymbol{w}\in\Delta$$

which is motivated by the fact that $1/\|\boldsymbol{w}\|_\infty$ is a lower bound of $\|\boldsymbol{w}\|_0$ for any $\boldsymbol{w} \in \Delta$. They also showed that it can be exactly solved via convex programming, although it is also a non-convex problem.

Xiao and Bai (2022) considered the Hadamard reparameterization $\boldsymbol{w} = \boldsymbol{u} \odot \boldsymbol{u}$ and the probabilistic simplex constraint $\boldsymbol{w} \in \Delta$ is reduced to the unit sphere constraint of $\boldsymbol{u} \in \mathbb{S}^{p-1} := \{\boldsymbol{u} \in \mathbb{R}^p : \|\boldsymbol{u}\|_2 = 1\}$. This reparameterization technique enables them to add the ℓ_1 penalty $\lambda\|\boldsymbol{u}\|_1$ to promote the sparsity of $\boldsymbol{u}$ and thus $\boldsymbol{w}$. Specifically, they considered the following ℓ_1 regularized non-convex problem:

$$\min_{\boldsymbol{u}\in\mathbb{R}^p} \frac{1}{2n}\|\boldsymbol{X}\left(\boldsymbol{u}\odot\boldsymbol{u}\right) - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{u}\|_1 \quad \text{s.t. } \boldsymbol{u} \in \mathcal{S}^{p-1}.$$

To solve this non-convex problem, they proposed a geometric projected gradient method with global convergence to a critical point. In fact, this reparameterization technique can also be applied directly to the ℓ_0-type sparse optimization problem. Combined with the reparameterization technique $\boldsymbol{w} = \boldsymbol{u} \odot \boldsymbol{u}$, the existing fast sparse solver (see, e.g., Wang et al., 2024) can solve the following sparse constraint optimization problem

$$\min_{\boldsymbol{u}\in\mathbb{R}^p} \frac{1}{2n}\left\|\frac{1}{\|\boldsymbol{u}\|_2^2}\boldsymbol{X}(\boldsymbol{u}\odot\boldsymbol{u}) - \boldsymbol{y}\right\|_2^2 \quad \text{s.t. } \|\boldsymbol{u}\|_0 \le s$$

where s is the pre-specified sparsity level. That is, it directly optimizes the normalized term $\tilde{\boldsymbol{u}} := (\boldsymbol{u}\odot\boldsymbol{u})/\|\boldsymbol{u}\|_2^2$ which is guaranteed to be feasible such that $\tilde{\boldsymbol{u}} \in \Delta$.

All the above methods are motivated by different procedures such as heuristic, norm approximation, parameterization, and sparse projection. Most of such procedures either lack statistical theory or pose an additional computation burden. Although the simplex constraint leads to such a significant flaw, we claim that it also brings some interesting blessings, as claimed in the next section.

1.2 Regularization-free methods

In this section, we introduce the notion of self-regularization and some interesting properties specialized to problem (1). By self-regularization, we mean that an estimator or the output of an optimization algorithm can obtain comparable performance as Lasso (w.r.t. ℓ_2 error) without an additional regularization term like $\lambda\|\boldsymbol{w}\|_1$. Specifically, under the commonly used conditions, the minimizer of the regularization-free problem (1) can actually obtain the Lasso-type ℓ_2 error $O\big(\sqrt{\log(p)/n}\big)$ with high probability (see, e.g., Meinshausen, 2013; Slawski and Hein, 2013; Li et al., 2020).

Self-regularization was first studied in Meinshausen (2013); Slawski and Hein (2013). They considered the non-negativity constrained least squares (NNLS) and the corresponding minimizer $\widehat{\boldsymbol{w}}_n^{NNLS}$, that is, the constrained set was the non-negative orthant $\{\boldsymbol{w} \in \mathbb{R}^p : \boldsymbol{w} \ge \boldsymbol{0}\}$ rather than the simplex Δ in (1). Under certain conditions, they proved that $\widehat{\boldsymbol{w}}_n^{NNLS}$ is consistent in the high-dimensional case $p \gg n$. Based on this self-regularization property, Slawski and Hein (2013) further considered recovering the support set by thresholding $\widehat{\boldsymbol{w}}_n^{NNLS}$ and taking the positions of the first $\widehat{s}$ largest components $[\widehat{w}_n]_j^{NNLS}$ as the estimated support set $\widehat{S}$. The theoretical guarantee was established in their work, but the choice of $\widehat{s}$ needed some prior knowledge of the noise level σ. These studies deviated from the paradigm in which sparsity-promoting regularization is regarded as a necessity in the high-dimensional estimation problem and provided a theoretical understanding of the previous empirical success of NNLS.

Li et al. (2020) recently extended the self-regularization property from the NNLS problem to the simplex constraint. They proved that the optimal solution $\widehat{\boldsymbol{w}}_n$ of (1) has a desirable statistical property similar to Lasso, that is, $\widehat{\boldsymbol{w}}_n$ is consistent as long as $s^* \log(p)/n \to 0$. Beyond this property, and observing that $\widehat{\boldsymbol{w}}_n$ was usually much denser than $\boldsymbol{w}^*$ (i.e. $\|\widehat{\boldsymbol{w}}_n\|_0 \gg s^*$), Li et al. (2020) proposed some modified techniques, including thresholding, weighted ℓ_1 and negative ℓ_2-norm to sparsify $\widehat{\boldsymbol{w}}_n$ further. The corresponding theoretical guarantee for support recovery was established for a special case where $\boldsymbol{X}^\top \boldsymbol{X}$ is an identity matrix, i.e., the least squares denoising problem.

Although $\widehat{\boldsymbol{w}}_n$ possesses the self-regularization property, some questions remain from the practical perspective. Notably, $\widehat{\boldsymbol{w}}_n$ is the theoretically optimal solution to (1), and it lacks a closed-form expression when $\boldsymbol{X}^\top \boldsymbol{X}$ is not an identity matrix. As a result, an optimization algorithm must be recruited to compute $\widehat{\boldsymbol{w}}$ that approximates $\widehat{\boldsymbol{w}}_n$. However, this introduces an optimization error $\|\widehat{\boldsymbol{w}} - \widehat{\boldsymbol{w}}_n\|_2$, whose impact remains unclear. To the best of our knowledge, existing theoretical analysis does not address the properties of $\widehat{\boldsymbol{w}}$, the output of an optimization algorithm. For instance, it is not evident whether the optimization error is negligible, especially in high-dimensional ($p \gg n$) settings where $f(\boldsymbol{w})$ is not strongly convex. This raises several important questions: (i) Does $\widehat{\boldsymbol{w}}$, as an approximation of $\widehat{\boldsymbol{w}}_n$, retain the self-regularization property? (ii) Since optimization algorithms are typically iterative, can we quantitatively assess the quality of the t-th iteration $\boldsymbol{w}^t$ of the algorithm, say $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$? (iii) Can we establish the linear convergence rate if we care about $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$ rather than $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$? (iv) In terms of the variable selection, under what conditions does the algorithm's solution accurately recover the true support set S^*? This paper aims to address these questions by bridging the gap between statistical and optimization properties. The answers to the above questions motivate us to propose a computationally feasible algorithm with rigorous statistical guarantees.

1.3 Proposal and contribution

In this paper, we mainly deal with problem (1) with the aim of estimating $\boldsymbol{w}^*$ and recovering S^*. Specifically, we directly analyze, from both statistical and optimization perspectives, the t-th iterate $\boldsymbol{w}^t$ of the projected gradient (PG) descent method applied to solving (1). The main contribution of this paper is two-fold and summarized as follows:

1. We fill the gap between statistics and optimization. Especially, we prove a linear convergence rate of the term $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$ to a statistically negligible error and thus claim a similar self-regularization property for t-th iteration $\boldsymbol{w}^t$ of the projected gradient descent method as $\widehat{\boldsymbol{w}}_n$. Note that this result is totally different from the counterpart in Li et al. (2020), which is only proved for the theoretical minimizer. The basic idea is that reducing optimization error below the order of statistical error is not only sufficient for self-regularization but also easy to compute.

2. We establish the variable selection consistency. Combined with the information criterion, we propose a new algorithm referred to as PERMITS (short for projected gradient

method with tail screening) by embedding a tail screening procedure into PG. This algorithm fixes the smallest component of $\boldsymbol{w}^t$ to be zero when t is appropriately large and then keeps optimizing the remaining components, which helps sparsify the iterates $\boldsymbol{w}^t$. Under mild conditions, we prove that PERMITS is statistically guaranteed to recover the unknown support set, i.e., $\text{supp}(\boldsymbol{w}) = S^*$ for the output $\boldsymbol{w}$ of PERMITS. The pivotal factor influencing the algorithmic performance and theoretical characteristics is the special information criterion (SIC), which ultimately contributes to the variable selection consistency. Contrary to being a straightforward combination of the PG algorithm and variable selection procedure, PERMITS necessitates a comprehensive understanding and analysis of the self-regularization property inherent in the PG algorithm.

1.4 Outline of the paper

The remainder of this paper is organized as follows. In the next section, we introduce the self-regularization property of the PG algorithm and propose the PERMITS algorithm. In Section 3, we study the theoretical properties of the PERMITS algorithm from the perspectives of both computation and statistics. Extensive numerical experiments using both synthetic and real data are shown in Section 4 to validate our theoretical results. Section 5 closes this paper with a concluding remark. In the appendix, detailed proofs of all theoretical results are provided, together with additional numerical results.

1.5 Notations

Capital bold letters such as $\boldsymbol{X}$ represent matrices, and lowercase bold letters such as $\boldsymbol{w}, \boldsymbol{y}$, and $\boldsymbol{\xi}$ represent column vectors. The set $\{1, 2, \cdots, p\}$ is denoted as $[p]$. For any matrix $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ and a subset $S \subset [p]$, $\boldsymbol{X}_S \in \mathbb{R}^{n \times |S|}$ is the sub-matrix consisting of columns in S. Particularly, X_j is the j-th column of $\boldsymbol{X}$. For any vector $\boldsymbol{w}$, let $[\boldsymbol{w}]_i$ or w_i be its i-th coordinate or component and $\boldsymbol{w}_S \in \mathbb{R}^{|S|}$ is the sub-vector consisting of components in S. We denote $\|\boldsymbol{w}\|_1, \|\boldsymbol{w}\|_2, \|\boldsymbol{w}\|_\infty$ as the usual ℓ_1, ℓ_2 and ℓ_∞ norm of $\boldsymbol{w}$. The vector $\boldsymbol{x}_+$ is denoted as the vector with its i-th coordinate being $[\boldsymbol{x}_+]_i = x_i \vee 0$ where $a \vee b$ denotes the larger one between a and b. For any binary operation between a vector $\boldsymbol{x}$ and a scalar a, it means the same operation between a and each coordinate of $\boldsymbol{x}$, e.g., $[\boldsymbol{x} + a]_i = x_i + a$. C and c are denoted as the absolute constants. The symbol $\gtrsim$ means $\geq$ with some hidden absolute constant C.

2. Methodology

In this section, we first introduce the projected gradient descent method used to solve (1) and elucidate the self-regularization property of its iterates $\boldsymbol{w}^t$ in terms of statistical and computational performance. Based on this property, we further develop a variable selection procedure using the special information criteria.

2.1 Projected gradient method and self-regularization property

The constrained minimization (1) can be reformulated as the following unconstrained composite optimization problem

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) + \chi_\Delta(\boldsymbol{w}) \tag{2}$$

where $\chi_\Delta(\cdot)$ is the characteristic function of Δ defined as follows:

$$\chi_\Delta(\boldsymbol{w}) = \begin{cases} 0, & \boldsymbol{w}\in\Delta \\ \infty, & \boldsymbol{w}\notin\Delta \end{cases}.$$

The projected gradient (PG) method (Beck and Teboulle, 2009; Nesterov, 2013; Beck, 2017) is an efficient tool to solve (2). The basic idea of PG is illustrated as follows. We first replace the differentiable part $f(\boldsymbol{u})$ with its quadratic approximation evaluated at the current solution $\boldsymbol{w}$ that

$$m_M(\boldsymbol{u}) := f(\boldsymbol{w}) + \langle \nabla f(\boldsymbol{w}), \boldsymbol{u}-\boldsymbol{w}\rangle + \frac{M}{2}\|\boldsymbol{u}-\boldsymbol{w}\|_2^2$$

where $f(\boldsymbol{w}) = (2n)^{-1}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|_2^2$, $\nabla f(\boldsymbol{w}) = n^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y})$ and M is a guess value of the unknown Lipschitz constant L_f of f. Suppose that the current solution is $\boldsymbol{w}$, then we can make an update and obtain a new solution $\boldsymbol{w}^+$ by solving the following problem

$$\begin{aligned}\boldsymbol{w}^+ &= \operatorname*{argmin}_{\boldsymbol{u}\in\mathbb{R}^p}\{m_M(\boldsymbol{u}) + \chi_\Delta(\boldsymbol{u})\}\\ &= \mathcal{P}_\Delta\left[\boldsymbol{w} - M^{-1}\nabla f(\boldsymbol{w})\right],\end{aligned}$$

where $\mathcal{P}_\Delta(\cdot):\mathbb{R}^p\to\Delta$ is a projection operator that maps a vector to the $(p-1)$-dimensional simplex. The replacement of the quadratic approximation simplifies the minimization problem to the projection onto the simplex. There exist efficient algorithms to compute this exact projection (Duchi et al., 2008; Wang and Carreira-Perpinán, 2013; Condat, 2016; Perez et al., 2020). We iteratively perform quadratic approximation and projection until the difference between two consecutive iterations is small. This iterative procedure is summarized in Algorithm 1, which provides a meta PG algorithm.

Algorithm 1 Projected Gradient (PG) Method

```
Input: $\boldsymbol{w}^0 \in \Delta$, tolerance $\epsilon > 0$.
1: Initialize $t \leftarrow 0$.
2: while $\|\boldsymbol{w}^{t+1} - \boldsymbol{w}^t\|_2 \geq \epsilon$ do
3:     Select step-size $M^t > 0$                    ▷ see more details in Algorithm 3
4:     Set $\boldsymbol{w}^{t+1} \leftarrow \mathcal{P}_\Delta\left[\boldsymbol{w}^t - \frac{1}{M^t}\nabla f(\boldsymbol{w}^t)\right]$
5:     $t \leftarrow t + 1$
Output: $\boldsymbol{w}^t$.
```

Remark 1 *The convergence of Algorithm 1 relies on the choice of step size parameter M^t in Step 3. It is usually required that the selected M^t satisfies the following sufficient descent property*

$$f(\boldsymbol{w}^{t+1}) \leq f(\boldsymbol{w}^t) + \langle \nabla f(\boldsymbol{w}^t), \boldsymbol{w}^{t+1}-\boldsymbol{w}^t\rangle + \frac{M^t}{2}\|\boldsymbol{w}^{t+1}-\boldsymbol{w}^t\|_2^2 \tag{3}$$

which can be fulfilled via a backtracking procedure (Nesterov, 2013). The details of this procedure are provided in Algorithm 3 in the Appendix. The backtracking method introduces two hyperparameters: (i) L^0, the initial step size, and (ii) γ, the decay rate of step size. Given an input pair (L^0, γ), the backtracking strategy iteratively adjusts step size until condition (3) *holds. The values of $L^0 > 0$ and $\gamma > 1$ are arbitrary and do not affect the theoretical guarantees. In our implementation, we set $L^0 = 1, \gamma = 2$, which performed well across all numerical experiments. Furthermore, for Algorithm 1, the choice of initial parameter $\boldsymbol{w}^0$ does not influence theoretical results, provided that the tolerance parameter ϵ is selected appropriately. The criteria for setting ϵ will be discussed in Sections 2.3 and 3. Henceforth, for simplicity, we also use the notation* $\mathrm{PG}(\boldsymbol{w}^0, \epsilon)$ *to represent using Algorithm 1 to solve problem* (2)*.*

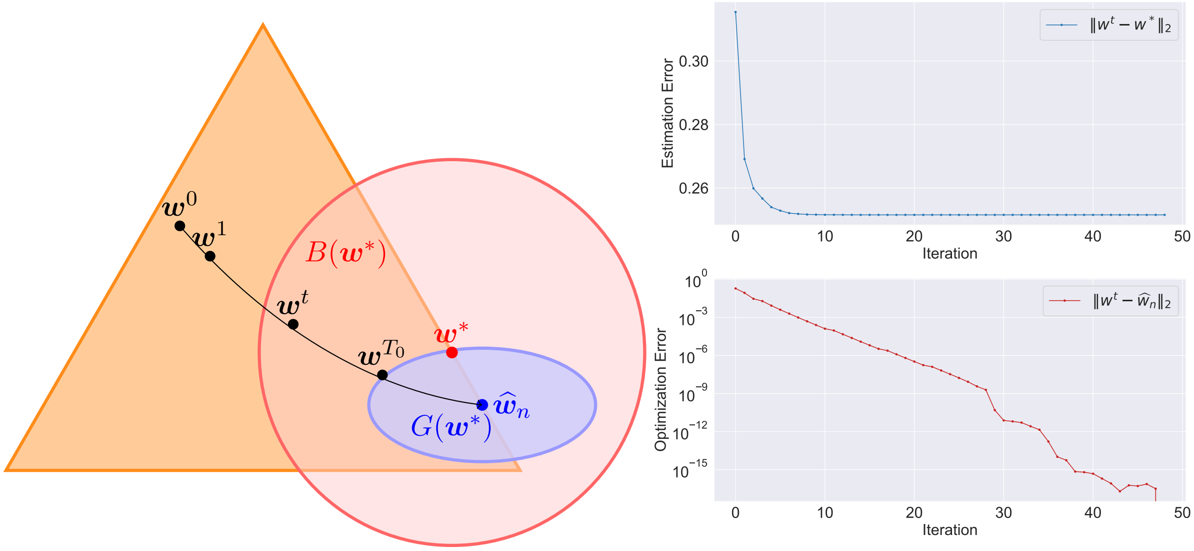
Under general convexity assumptions, standard analyses of the projected gradient (PG) method (Beck, 2017) guarantee convergence of iterates $\boldsymbol{w}^t$ to the solution $\widehat{\boldsymbol{w}}_n$ of problem (1). Specifically, the geometric convergence rate $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2 \le O(e^{-ct})$ holds for some $c > 0$, provided that $f(\boldsymbol{w})$ is strongly convex. However, this strong convexity condition is hard to guarantee in high-dimensional cases where $p \gg n$. In fact, the strong convexity parameter of $f(\boldsymbol{w})$ equals the least eigenvalue of $n^{-1}\boldsymbol{X}^\top\boldsymbol{X} \in \mathbb{R}^{p\times p}$, which is exactly 0 when $p \gg n$. This raises an apparent gap between $\boldsymbol{w}^t$ and $\widehat{\boldsymbol{w}}_n$, causing $\boldsymbol{w}^t$ may not inherit the self-regularization property of $\widehat{\boldsymbol{w}}_n$ in high-dimensional cases.

Different from the common analysis, we directly analyze the term $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|$ by merging the statistical error $\|\widehat{\boldsymbol{w}}_n - \boldsymbol{w}^*\|_2$ and the optimization error $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$. Then, an interesting self-regularization property shows that $\boldsymbol{w}^t$ converges to $\boldsymbol{w}^*$ up to an error of order $O(\sqrt{s^*\sigma^2\log(p)/n})$ as illustrated in Figure 1. Specifically, the orange triangle denotes the feasible region, i.e., the unit simplex Δ. The red region $B(\boldsymbol{w}^*)$ is a circle centered at $\boldsymbol{w}^*$ with the radius being the statistical error. The blue ellipse $G(\boldsymbol{w}^*)$ denotes the sub-level set whose elements have a loss value smaller than $f(\boldsymbol{w}^*)$. Definition 6 provides details of the two intersecting regions $B(\boldsymbol{w}^*)$ and $G(\boldsymbol{w}^*)$. In Section 3, we prove two critical results: (i) $G(\boldsymbol{w}^*) \subset B(\boldsymbol{w}^*)$ and (ii) $\boldsymbol{w}^t$ enters $G(\boldsymbol{w}^*)$ geometrically and then stay there. In other words, $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \le O\big(\sqrt{s^*\sigma^2\log(p)/n}\big)$ at a geometric rate. This explicitly reveals the self-regularization property of $\boldsymbol{w}^t$.

This property is highly useful for variable selection since it suggests that $\boldsymbol{w}^t$ in the PG algorithm converges to $\boldsymbol{w}^*$ at a fast rate. Specifically, the convergence rate matches that of the oracle estimator, $O(\sqrt{s^*/n})$, ignoring logarithmic and variance terms. In other words, a large component w_i^t corresponds to a large w_i^*, and a small w_i^t corresponds to small a w_i^*. This follows from the bound $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_\infty \le \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \le O(\sqrt{s^*\sigma^2\log(p)/n})$, which indicates the gap between w_i^t and w_i^* is small when n is sufficiently large. More importantly, this property helps distinguish between nonzero and zero components of $\boldsymbol{w}^*$, thereby aiding in the identification of the true support set S^*.

2.2 Projected gradient method with tail screening

To adapt the PG method to our sparse learning setting, we propose a screened PG algorithm: PERMITS, a shorthand for projected gradient method with tail screening. The complete algorithmic procedure is summarized in Algorithm 2. The core idea of PERMITS is simple: we recursively remove one element in the current support estimate to get a new support



(a) Illustrative trajectory

(b) Measurement of convergence.

Figure 1: Illustration of two-stage convergence property. (a) The convergence trajectory of $\boldsymbol{w}^t$ to $\widehat{\boldsymbol{w}}_n$ can be divided into two stages $[\boldsymbol{w}^0, \boldsymbol{w}^{T_0}]$ and $[\boldsymbol{w}^{T_0}, \widehat{\boldsymbol{w}}_n]$ for some T_0. The convergence rate of the first stage is geometric, and the point in the second stage is apart from $\boldsymbol{w}^*$ with distance at most $O(\sqrt{s^*\sigma^2 \log(p)/n})$, i.e., the statistical error. The whole second stage $[\boldsymbol{w}^{T_0}, \widehat{\boldsymbol{w}}_n]$ is included in the statistically satisfactory region $G(\boldsymbol{w}^*)$. (b) Two measurements of the convergence. The upper panel shows the estimation error $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$. In the lower panel, the optimization error $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$ decreases linearly and then behaves in a certain oscillating pattern, which may be caused by the non-strong-convexity.

estimate S and fit the reduced data $(\boldsymbol{X}_S, \boldsymbol{y})$ to obtain an updated coefficient $\boldsymbol{w}$ whose quality is measured by the special information criterion $\mathrm{SIC}(\boldsymbol{w}) = n \log f(\boldsymbol{w}) + \|\boldsymbol{w}\|_0 \log(p) \log\log n$. The output of PERMITS is the one that minimizes SIC.

Removing one element in the estimated support set leverages the self-regularization property of $\boldsymbol{w}^t$. Specifically, we set $\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2$ as a surrogate of $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$. When $\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2$ is tiny, we expect the self-regularization property $\|\boldsymbol{w}^t - \boldsymbol{w}^*\| \lesssim \sqrt{s^*\sigma^2 \log(p)/n}$ to hold. At this point, we discard a tail component of $\boldsymbol{w}^t$ and fix its value at 0. If the minimal signal $b^* = \min_{j \in S^*} w_j^*$ is large and $\mathrm{supp}(\boldsymbol{w}^t) \supset S^*$, discarding a tail component in $\boldsymbol{w}^t$ is safe, i.e., it removes a component in $(S^*)^c$. We refer to this process as tail screening. Tail screening produces a sequence of nested support sets:

$$S_p \supset S_{p-1} \supset \cdots \supset S_2 \supset S_1,$$

where $S_p = [p]$ is the full model, and $S_1 = \{j\}$ for some $j \in [p]$ is the singleton model. Notably, one of these sets coincides exactly with the true support set S^*. In other words, identifying S^* only requires evaluating p models rather than all 2^p possible models.

To select the correct model, we use the $\mathrm{SIC}(\boldsymbol{w})$, which adaptively determines the unknown sparsity level s^*. The penalty term $\|\boldsymbol{w}\|_0 \log(p) \log\log(n)$ in SIC balances the underfitting and over-fitting —increasing the model complexity (i.e., larger $\|\boldsymbol{w}\|_0$) reduces the empirical loss $f(\boldsymbol{w})$, but increases the risk of fitting noise, thereby degrading generalization.

By minimizing the SIC over the candidate models $\{S_p, S_{p-1}, \cdots, S_1\}$, we obtain a final estimate of the true support set S^*.

Algorithm 2 ProjEcted gRadient Method wIth Tail Screening (PERMITS)

```
Input: $(\boldsymbol{X}, \boldsymbol{y}), \boldsymbol{w}_{\text{init}} \in \Delta, T_{\min} \geq 1, \epsilon > 0$.
1: Initialize $S = [p]$, $\text{SIC}_{\text{best}} = \infty$
2: while $|S| \geq 1$ do
3:     $\boldsymbol{w}_{\text{init}} \leftarrow \boldsymbol{w}_S / \boldsymbol{1}^\top \boldsymbol{w}_S$          ▷ normalization for warm start
4:     $\boldsymbol{w}_S \leftarrow \text{PG}(\boldsymbol{w}_{\text{init}}, \epsilon)$ with data $(\boldsymbol{X}_S, \boldsymbol{y})$ and at least $T_{\min}$ iterations
5:     $\boldsymbol{w}_{S^c} \leftarrow \boldsymbol{0}$
6:     $\text{SIC}(\boldsymbol{w}) \leftarrow n \log f(\boldsymbol{w}) + |S| \log(p) \log\log n$
7:     if $\text{SIC}(\boldsymbol{w}) < \text{SIC}_{\text{best}}$ then
8:         $\text{SIC}_{\text{best}} \leftarrow \text{SIC}(\boldsymbol{w})$, $\boldsymbol{w}_{\text{best}} \leftarrow \boldsymbol{w}$
9:     $j \leftarrow \arg\min_{i \in S} \boldsymbol{w}_i$, $S \leftarrow S \setminus \{j\}$
Output: $\boldsymbol{w}_{\text{best}}$
```

Remark 2 *In the fourth line of Algorithm 2, we apply the PG algorithm over the current support S. $T_{\min}$ is an additional hyper-parameter for the minimum number of iterations that are needed for the proof of statistical properties, and a small value (e.g., $T_{\min} = 5$ in our numerical experiment) is usually sufficient.*

Remark 3 *In the ninth line of Algorithm 2, we remove the component of $\boldsymbol{w}_S$ with the smallest value from the current estimated support S. If multiple components share the smallest value, we can randomly select one of them and discard it. In practice, we can also discard more components at each iteration; for instance, if there are many zero components in $\boldsymbol{w}_S$, we can discard all of them.*

Remark 4 *Intuitively speaking, PERMITS is better suited to the sparse learning problem because it integrates the standard PG algorithm and incorporates a tail screening procedure. Additionally, it offers two key advantages. First, the dimension of the optimization decreases as the projected gradient iterations proceed, since more components are excluded from the iteration. This may lead to acceleration, as will be confirmed by our numerical experiments. Second, PERMITS induces sparsity in the iterates $\boldsymbol{w}^t$ by setting more and more negligibly small components exactly to zero during the iterations. This also demonstrates why PERMITS has the capability to perform variable selection.*

Remark 5 *The information criterion typically has a form $n \log f(\boldsymbol{w}) + \lambda_{n,p} \|\boldsymbol{w}\|_0$ where $\lambda_{n,p}$ is a term scales with n and p. Selecting an appropriate value for $\lambda_{n,p}$ is critical to identify the true model S^*. Several choices of $\lambda_{n,p}$ have been proposed in the literature, and they correspond to different information criteria, including AIC, BIC, and some other ones in the high-dimensional setting (Akaike, 1998; Schwarz, 1978; Wang et al., 2013; Zhu et al., 2020). Algorithm 2 sets $\lambda_{n,p} = \log p \log\log n$, which corresponds to the SIC proposed in Zhu et al. (2020).*

2.3 Practical guidance for the tolerance parameter

In PERMITS, we terminate the sub-problem when the successive difference $r^t = \|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2$ is smaller than ϵ. In fact, the parameter ϵ balances the computation time and estimation accuracy. Specifically, as ϵ decreases, its estimation accuracy improves, while it needs more computation time. Hence, a moderate value ϵ may be a more appropriate choice.

From the theoretical perspective, we provide an upper bound for the consistency that

$$\epsilon \leq \frac{C(e^{\kappa_1} - 1)}{\mu_f} \sqrt{\frac{s^* \sigma^2 \log p}{n}}$$

contains some unknown quantities such as κ_1, μ_f, s^* and σ^2. These unknown quantities bring difficulties to the choice of ϵ and this problem is inevitable if we simultaneously take into account the statistical and optimization issues. However, compared to these unknown quantities, sample size n and feature dimension p may change more dramatically across different practical tasks, and thus, we mainly concern the order of n and p. In Fan et al. (2023), compared to the choice of ϵ here, a variety of hyper-parameters (e.g., number of iterations) need to be chosen manually as well. In their work, they suggested replacing the unknown part (similarly, κ_1, μ_f in our problem) of the theoretical result with a known small constant (e.g., 10^{-4}) and this method achieved good performance in practice. Hence, in our paper, we follow their suggestion and take ϵ to be a small multiple of $\sqrt{\log(p)/n}$. In our implementation, $\epsilon = 10^{-4}\sqrt{\log(p)/n}$ is the default choice. Furthermore, from the practical perspective, we also perform some additional experiments in Appendix A to show the robustness of PERMITS with respect to this tolerance parameter.

3. Theoretical Properties

3.1 Conditions

We list some conditions for establishing the main theoretical results.

(C1) f is L_f smooth over the simplex such that

$$\|\nabla f(\boldsymbol{w}) - \nabla f(\boldsymbol{u})\|_2 \leq L_f \|\boldsymbol{w} - \boldsymbol{u}\|_2, \quad \forall \boldsymbol{w}, \boldsymbol{u} \in \Delta.$$

Condition (C1) only requires that the gradient $\nabla f(\boldsymbol{w})$ is L_f Lipschitz over the simplex Δ which is weaker than the counterpart over the whole $\mathbb{R}^p$ (Beck, 2017).

(C2) f is restricted strongly convex with parameter μ_f such that

$$f(\boldsymbol{w}) \geq f(\boldsymbol{u}) + \langle \nabla f(\boldsymbol{u}), \boldsymbol{w} - \boldsymbol{u} \rangle + \frac{\mu_f}{2}\|\boldsymbol{w} - \boldsymbol{u}\|_2^2$$

holds for any $\boldsymbol{w}, \boldsymbol{u} \in \Delta$ and $\boldsymbol{w} - \boldsymbol{u} \in C(S^*)$ where

$$C(S) := \left\{ \boldsymbol{\delta} \in \mathbb{R}^p : \mathbf{1}^\top \boldsymbol{\delta} = 0, \boldsymbol{\delta}_{S^c} \geq \mathbf{0} \text{ or } \boldsymbol{\delta}_{S^c} \leq \mathbf{0} \right\}.$$

The restricted set $C(S)$ consists of elements having sum 0 and the same sign on S^c and Condition (C2) only requires that f is μ_f-strongly convex over the direction $\boldsymbol{\delta} \in C(S^*)$. In the linear model setting, this condition is equivalent to the following statement

$$\frac{1}{n}\|\boldsymbol{X}\boldsymbol{\delta}\|_2^2 \geq \mu_f \|\boldsymbol{\delta}\|_2^2, \quad \forall \boldsymbol{\delta} \in C(S^*),$$

which is weaker than the usual restricted eigenvalue condition (Bickel et al., 2009; van de Geer and Bühlmann, 2009; Bühlmann and van de Geer, 2011; Wainwright, 2019) that requires the above inequality to hold over the cone $C_\alpha(S) := \{\boldsymbol{\delta} \in \mathbb{R}^p : \|\boldsymbol{\delta}_{S^c}\|_1 \leq \alpha \|\boldsymbol{\delta}_S\|_1\}$ for some $\alpha \geq 1$. In fact, in this case, our restricted set is much smaller than $C(S^*) \subset C_1(S)$.

(C3) The *i.i.d.* random errors $\xi_1, \ldots, \xi_n$ have zero mean and sub-Gaussian tails: there exists a constant $\sigma > 0$ such that $\mathbb{P}(|\xi_i| \geq t) \leq 2\exp(-t^2/\sigma^2)$, for all $t \geq 0$.

Condition (C3) is a widespread condition in the high-dimensional literature (Wainwright, 2019) and is weaker than the conventional normality condition.

(C4) $b^* := \min_{j \in S^*} w_j^* \geq \sqrt{\frac{C\sigma^2 s^* \log p \log\log n}{n\mu_f}}$ for some constant $C > 0$.

Condition (C4) is necessary for the theoretical guarantee in the high-dimensional setting. It actually allows b^* to be small enough as the sample size n increases. Specifically, if we are only concerned with the sample efficiency and ignore some constants, this condition says that the minimal signal b^* can decrease with the order $O(\sqrt{\log p \log\log n/n})$. This condition is crucial for the support recovery since, if the minimal signal is smaller than the noise level, we cannot expect to recover that weak signal.

(C5) $\rho := \max_{i \neq j} \frac{|X_i^\top X_j|}{\|X_i\|_2 \|X_j\|_2} < \frac{c}{s^*}$ for some small enough constant $0 < c < 1$.

Condition (C5), as discussed in Wainwright (2019), is stronger than Condition (C2), but it is only needed for proving variable selection consistency. It is a technical assumption that might be relaxed in future work.

3.2 Self-regularization of projected gradient descent method

In our theoretical analysis, we directly analyze the quantity $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$, which merges the statistical error $\|\widehat{\boldsymbol{w}}_n - \boldsymbol{w}^*\|_2$ and optimization error $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$ and only a weaker restricted strongly convex condition (C2) is needed for establishing its geometric rate. First, we formally define two sets, $G(\boldsymbol{w}^*)$ and $B(\boldsymbol{w}^*)$, which represent computationally optimal points and statistically optimal points, respectively.

Definition 6 *Define two useful sets as follows*

$$\begin{aligned} G(\boldsymbol{w}^*) &:= \{\boldsymbol{w} \in \Delta : f(\boldsymbol{w}) \leq f(\boldsymbol{w}^*)\} \\ B(\boldsymbol{w}^*) &:= \left\{\boldsymbol{w} \in \Delta : \|\boldsymbol{w} - \boldsymbol{w}^*\|_2 \leq \frac{4\sqrt{s^*}\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty}{\mu_f}\right\} \end{aligned}$$

where μ_f is the restricted strongly convex constant defined in Condition (C2).

Remark 7 *On one hand, the set $G(\boldsymbol{w}^*)$ is a sub-level set containing all $\boldsymbol{w}$ with loss value smaller than the true parameter $\boldsymbol{w}^*$. Note that $f(\boldsymbol{w}^*)$ is usually much larger than the minimum value $f(\widehat{\boldsymbol{w}}_n)$. Thus, for iterate $\boldsymbol{w}^t$ of the PG algorithm, entering $G(\boldsymbol{w}^*)$ is considerably easier than approaching $\widehat{\boldsymbol{w}}_n$, and this is why we refer to it as a computationally satisfied set. On the other hand, the set $B(\boldsymbol{w}^*)$ is a ball with center $\boldsymbol{w}^*$ and radius $4\sqrt{s^*}\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty/\mu_f$ which can be viewed as the statistical error. If $\boldsymbol{\xi}$ is σ^2 sub-Gaussian, this statistical error, with high probability, is of order $O\big(\sqrt{s^*\sigma^2\log p/n}\big)$ which has a dependence on p only with a logarithmic scale. That is, any point $\boldsymbol{w} \in B(\boldsymbol{w}^*)$ is a satisfactory estimate of $\boldsymbol{w}^*$ even in the high-dimensional case $p \gg n$. This demonstrates the meaning of the statistically optimal set.*

We first claim the self-regularization property of the simplex-constrained minimization problem in the following proposition, which is an extension of the result in Li et al. (2020).

Proposition 8 *If Condition (C2) holds, then we have $G(\boldsymbol{w}^*) \subset B(\boldsymbol{w}^*)$.*

Remark 9 *As a special case, for any minimizer $\widehat{\boldsymbol{w}}_n$ of (1), since $f(\widehat{\boldsymbol{w}}_n) \le f(\boldsymbol{w}^*)$, we have $\widehat{\boldsymbol{w}}_n \in B(\boldsymbol{w}^*)$ and thus*

$$\|\widehat{\boldsymbol{w}}_n - \boldsymbol{w}^*\|_2 \le 4\sqrt{s^*}\|\nabla f(\boldsymbol{w}^*)\|_\infty/\mu_f$$

which was proved in Li et al. (2020). This property shows the benefits of simplex constraint in that it helps prevent noise accumulation in the high-dimensional case ($p \gg n$).

Theorem 10 (Linear Convergence Rate) *Suppose Conditions (C1)-(C3) hold, let $\boldsymbol{w}^t$ be the t-th iteration of Algorithm 1 with any $\boldsymbol{w}^0 \in \Delta, L^0 > 0, \gamma > 1$, then, with probability at least $1 - O(p^{-3})$, the inequality*

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \le \max\left\{\sqrt{2}\exp\left(-\frac{1}{2}\frac{t\mu_f}{L^0 \vee (\gamma L_f)}\right), \frac{C}{\mu_f}\sqrt{\frac{s^*\sigma^2\log p}{n}}\right\}$$

holds for all $t > 0$, where C is an absolute constant.

Remark 11 *The term $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$ encompasses the optimization error $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$ and the estimation error $\|\widehat{\boldsymbol{w}}_n - \boldsymbol{w}^*\|_2$ simultaneously. It is well known that the global linear convergence rate of the optimization error can not be achieved under the above conditions. However, if we only care about $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$ and the tolerance $\frac{C}{\mu_f}\sqrt{\frac{s^*\sigma^2\log p}{n}}$ is admissible, the corresponding linear convergence rate can be established. To the best of our knowledge, this is the first time such a theoretical result has been derived.*

As an immediate result of Theorem 10, we obtain the following corollary.

Corollary 12 *In the setting of Theorem 10, if we set*

$$T \ge \left[\log\left(\frac{n\mu_f^2}{s^*\sigma^2\log p}\right) + C\right]\frac{L^0 \vee (\gamma L_f)}{\mu_f},$$

then we have

$$\max\left\{\|\boldsymbol{w}^T - \widehat{\boldsymbol{w}}_n\|_2, \|\boldsymbol{w}^T - \boldsymbol{w}^*\|_2\right\} \le \frac{C}{\mu_f}\sqrt{\frac{s^*\sigma^2\log p}{n}}.$$

In fact, this corollary also provides valuable insights from an optimization perspective. For example, even though the full-stage geometric convergence rate of $\|\boldsymbol{w}^t - \widehat{\boldsymbol{w}}_n\|_2$ can not be obtained, geometric convergence during the first stage is feasible under mild conditions. That is, the number of iterations required is still of the logarithmic order, provided that the target precision is of the same order as the statistical error. Similar results were established for general unconstrained and regularized M-estimators in Agarwal et al. (2010).

The following theorem claims that, based on this stopping rule $r^t = \|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|$, PG algorithm can also output a good approximation of the true $\boldsymbol{w}^*$.

Theorem 13 (ℓ_2 Error Bound) *Suppose Conditions (C1)-(C3) hold. Let* $\kappa_1 := \frac{\mu_f}{2L^0 \vee (\gamma L_f)} > 0$*, if we set the tolerance parameter* ϵ *such that*

$$\epsilon \leq \frac{C(e^{\kappa_1} - 1)}{\mu_f} \sqrt{\frac{s^* \sigma^2 \log p}{n}},$$

then Algorithm 1 outputs a solution $\boldsymbol{w}$ *satisfying*

$$\|\boldsymbol{w} - \boldsymbol{w}^*\|_2 \leq \frac{C}{\mu_f} \sqrt{\frac{s^* \sigma^2 \log p}{n}}$$

where $C > 0$ *is an absolute constant.*

Remark 14 *Theorem 13 provides an* ℓ_2 *error bound for the output of the PG algorithm, which aligns with the Lasso-type bounds. Therefore, our theoretical analysis integrates both statistical and optimization perspectives, demonstrating the self-regularization property of the PG algorithm used to solve the problem* (1).

3.3 Variable selection consistency of PERMITS

We have demonstrated useful computational properties and statistical properties of the PG algorithm. Motivated by these advantageous properties, PERMITS integrates a straightforward tail screening procedure into the PG algorithm to seek a sparse solution and recover the true support S^*. The following theorem demonstrates the property of the support set of the solution.

Theorem 15 (Variable Selection Consistency) *Suppose Conditions (C1) and (C3)-(C5) hold, let* $\boldsymbol{w}$ *be the output of the PERMITS algorithm with*

$$\epsilon \leq \frac{C(e^{\kappa_1} - 1)}{\mu_f} \sqrt{\frac{s^* \sigma^2 \log p}{n}} \text{ and } T_{\min} \geq \log s^* / \log(\kappa_2^{-1}),$$

where $\kappa_1 = \frac{\mu_f}{2L^0 \vee (\gamma L_f)}$ *and* $\kappa_2 = 1 - \mu_f / L_f$*. Then, with probability at least* $1 - O(p^{-3})$*, the following event holds*

$$\operatorname{supp}(\boldsymbol{w}) = S^*.$$

Remark 16 *To facilitate the proof, we replace condition (C2) with the incoherence condition (C5), which is also a common requirement in high-dimensional data analysis (Wainwright, 2019; Wright and Ma, 2022; Tang et al., 2023). To the best of our knowledge,* *Theorem 15 is the first result providing the variable selection consistency using the self-regularization property of problem* (1). *It is important to note that our result considers the output of a specific algorithm rather than a theoretical minimizer. The distinction is crucial—the former is attainable in practice, while the latter cannot be.*

4. Numerical Experiments

In this section, we perform numerical experiments using both synthetic and real data to verify the self-regularization property and the advantage of PERMITS. The experiments on synthetic data and real-world data are presented in Section 4.1 and Section 4.2, respectively. The results show that the proposed PERMITS algorithm performs better than state-of-the-art methods.

We compare the performance of PERMITS with the following methods:

1) Oracle: the oracle estimator, which is obtained by solving (1) subject to the constraint that $\text{supp}(\boldsymbol{w}) = S^*$.

2) IHT*: the iterative hard thresholding estimator proposed by Kyrillidis et al. (2013).

3) H-Lasso: the heuristic Lasso estimator that firstly solves the non-negative Lasso problem and then divides the solution by its ℓ_1 norm to fulfill the simplex constraint.

Note that Oracle is actually the best estimator for this sparse-constrained regression problem, and we include it just to exhibit the difference between different estimators and Oracle. The penalty parameter λ of H-Lasso is selected via cross-validation. For IHT*, the true sparsity s^* is provided since, otherwise, selecting the sparsity is time-consuming.

4.1 Synthetic data

In this subsection, we consider both statistical and computational performance. To validate the consistency of support set recovery, we fix $p = 1000$ and vary the sample size n from 100 to 600. The synthetic data is generated as follows. The rows of $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ are i.i.d. generated from a Gaussian distribution $\mathcal{N}_p(\boldsymbol{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is the covariance matrix and its elements being exponentially decreasing ($\Sigma_{ij} = \rho^{|i-j|}$). We consider three correlation cases such that $\rho \in \{-0.5, 0, 0.5\}$ and they are shown in different columns in Figures 2 and 3. Then, the true coefficient $\boldsymbol{w}^* \in \mathbb{R}^p$ takes value 0.1 on $s^* = 10$ randomly chosen positions S^* and 0 on the remaining positions $(S^*)^c$. The additive noise $\boldsymbol{\xi}$ is generated from $\mathcal{N}_n(\boldsymbol{0}, \sigma^2\boldsymbol{I})$ where σ^2 is the noise level. Here, σ^2 is chosen such that the signal-to-noise ratio (SNR), defined by $\text{Var}(\boldsymbol{X}\boldsymbol{w}^*)/\sigma^2$, belongs to $\{0.5, 1, 5\}$. Results with different SNRs are shown in different rows in Figures 2 and 3. Finally, the response $\boldsymbol{y}$ is generated through the linear model $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}^* + \boldsymbol{\xi}$. Our experiments will present the results of 50 repeated simulations where mean metric (accuracy, error, and time) values are plotted with n ranging from 100 to 600.

Statistical performance. We first present the statistical performance of methods in the following. Two metrics of statistical performance are considered here: accuracy and error.

For any given estimator $\boldsymbol{w}$, we define its accuracy and error as follows

$$\begin{aligned}\text{Accuracy} &:= \frac{|\operatorname{supp}(\boldsymbol{w}) \cap \operatorname{supp}(\boldsymbol{w}^*)|}{|\operatorname{supp}(\boldsymbol{w}) \cup \operatorname{supp}(\boldsymbol{w}^*)|},\\ \text{Error} &:= \|\boldsymbol{w} - \boldsymbol{w}^*\|_2.\end{aligned}$$

Firstly, we visualize the variation of accuracy as the sample size n increases in Figure 2. As n increases, the accuracy of PERMITS (the pink line in each sub-figure) approaches 1, which validates the variable selection consistency as theoretically shown in Theorem 15. As SNR increases, the sample size needed for high accuracy decreases. This can be seen by comparing different panels in each column in Figure 2 for each model. Similarly, each row shows the effect of different correlations for a fixed SNR, and high correlation ($\rho = \pm 0.5$) cases need a larger sample size to obtain high accuracy. Figure 2 also reveals that the case with $\rho = -0.5$ seems to be easier than that with $\rho = 0.5$. This is actually due to an informal fact that constraint $\Delta = \{\boldsymbol{w} : \boldsymbol{w} \geq \boldsymbol{0}, \boldsymbol{1}^\top \boldsymbol{w} = 1\}$ excludes those negatively correlated features, and only positively correlated features are under consideration. Thus, the case with $\rho = -0.5$ essentially has a lower feature dimension than that with $\rho = 0.5$. As sample size n increases, the accuracy of each method except H-Lasso tends to 1, and PERMITS universally outperforms other methods in each SNR and correlation. H-Lasso fails to be consistent since it includes too many irrelevant features, and this defect is also validated in the later real data experiment.

Then, we show the ℓ_2 error in Figure 3, which demonstrates that PERMITS greatly approaches the oracle ℓ_2 error when n is sufficiently large. As sample size n increases, each method tends to obtain the same ℓ_2 error as Oracle, and PERMITS outperforms other methods in the sense that we can obtain the same ℓ_2 error with a smaller sample size. The effects of SNR and correlation are similar to that in Figure 2. Note that, in contrast to accuracy, the H-Lasso outperforms IHT* in terms of ℓ_2 error, although it tends to select more features.

Computation performance. With respect to the computational performance, we mainly focus on the running time of different methods and their dependence on the dimension p. Here, we compare the mean running time of 50 repeated experiments with p ranging from 100 to 1000, and n is fixed at 500. Figure 4 shows the time demanded by these four methods. We aggregate three different SNRs with their means, and thus only one row appears in Figure 4, in contrast to three rows in Figure 2 and 3. Note that H-Lasso is much more time-consuming in contrast to other methods due to the cross-validation procedure.

4.2 Real data: DJIA constituents detection

In this subsection, we consider a real-world application that aims to detect the constituents of the Dow Jones Industrial Average (DJIA) index based on the daily return of the DJIA index and a pool of stocks. The daily return of the DJIA index y_t is a price-weighted daily return of $s^* = 30$ prominent companies. Here, we choose the stocks pool to be the constituents S&P 500, which contains $p = 490$ stocks after dropping those with missing data. The resulting stock pool actually contains the constituents of the DJIA as a subset, and we want to recover this subset using daily return data only.

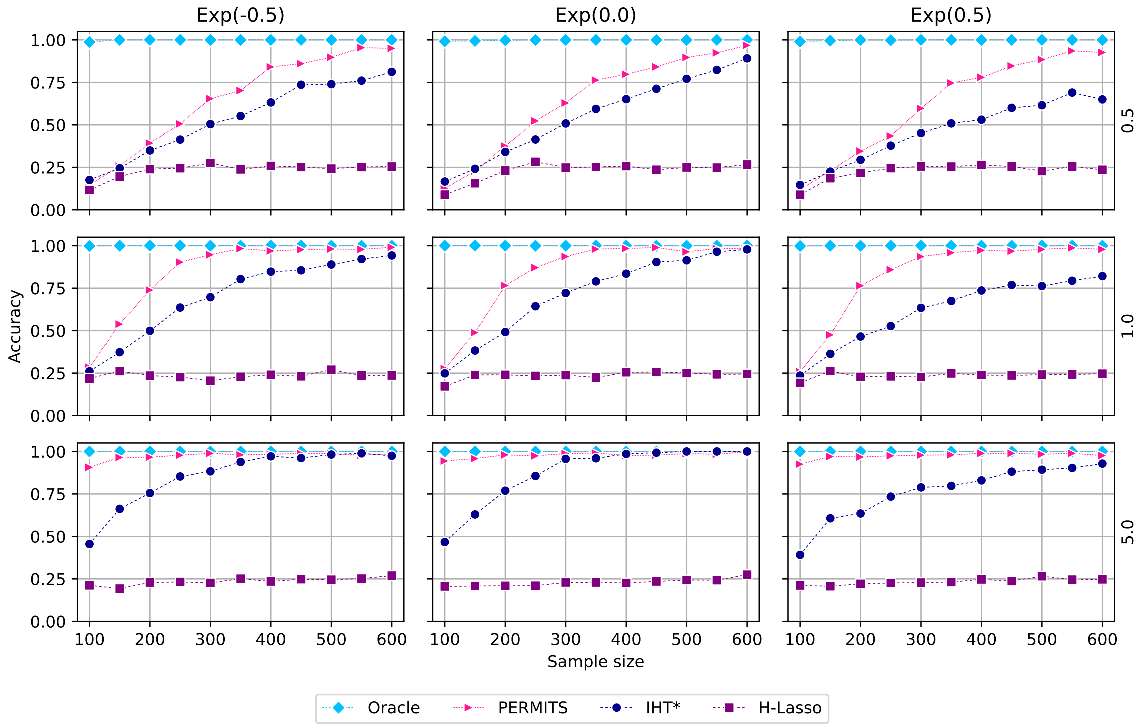


Figure 2: The sample size (x-axis) versus variable selection accuracy (y-axis). We fix $p = 1000, s^* = 10$. Each sub-figure corresponds to a different choice of SNR (in different rows) and correlation structure (in different columns). The sample size n ranges from 100 to 600 with step 50. 50 repeated experiments are performed, and the mean values are plotted.

We collect daily returns data of the year 2022, which contains $n = 251$ samples. Denote the daily return of DJIA by $y_t \in \mathbb{R}$ and the returns of the pool by $\boldsymbol{x}_t \in \mathbb{R}^p$, then we have

$$y_t = \boldsymbol{x}_t^\top \boldsymbol{w}_t, \quad t = 1, 2, \ldots, 251$$

where $\boldsymbol{w}_t \in \mathbb{R}^p$ is the weights of 490 stocks in the t-th day. Although 30 prominent constituents of DJIA are fixed for different t (i.e., $\mathrm{supp}(\boldsymbol{w}_t)$ is fixed), the daily weights of these 30 stocks change every day (i.e., $\boldsymbol{w}_t = \boldsymbol{w}_{t'}$ for $t, t' \in \{1, \ldots, 251\}$). To put this application into the usual setting of a signal-plus-noise linear model, we can construct a ground truth coefficient

$$\boldsymbol{w}^* := \frac{1}{n} \sum_{t=1}^{n} \boldsymbol{w}_t$$

and view $\xi_t = \boldsymbol{x}_t^\top (\boldsymbol{w}_t - \boldsymbol{w}^*)$ as the noise. Then we have the equivalent model that

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}^* + \boldsymbol{\xi}$$

where $\boldsymbol{X} \in \mathbb{R}^{251 \times 490}$, $\boldsymbol{y} \in \mathbb{R}^{251}$ are observed data and $(\boldsymbol{w}^*, \boldsymbol{\xi})$ are unobservable. The task is to recover the support set of $\boldsymbol{w}^*$ based on $(\boldsymbol{X}, \boldsymbol{y})$, i.e., detect the constituents of DJIA based only on the daily return of the DJIA index and 490 stocks. The right panel of Figure 5 shows the weight $\boldsymbol{w}^*$ defined as above. We emphasize that $\boldsymbol{w}^*$ is constructed manually and is indeed in the simplex; furthermore, we know 30 prominent constituents, i.e., the

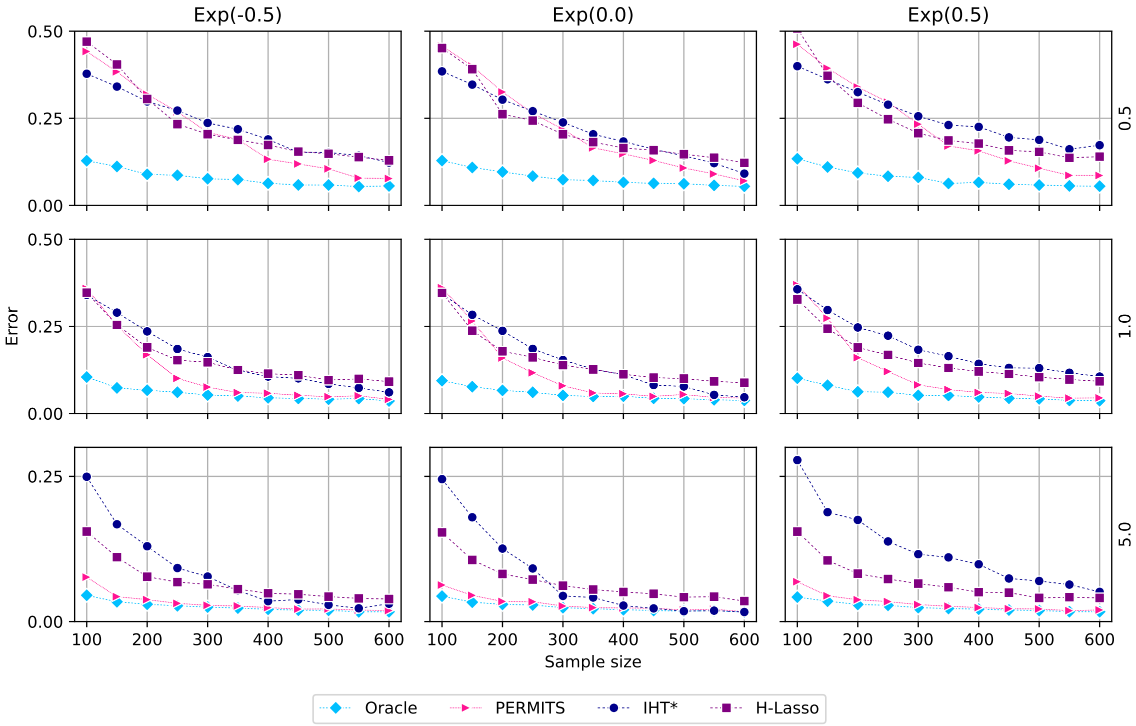


Figure 3: The sample size (x-axis) versus ℓ_2 error of parameter estimation (y-axis). The remaining settings are the same as Figure 2.

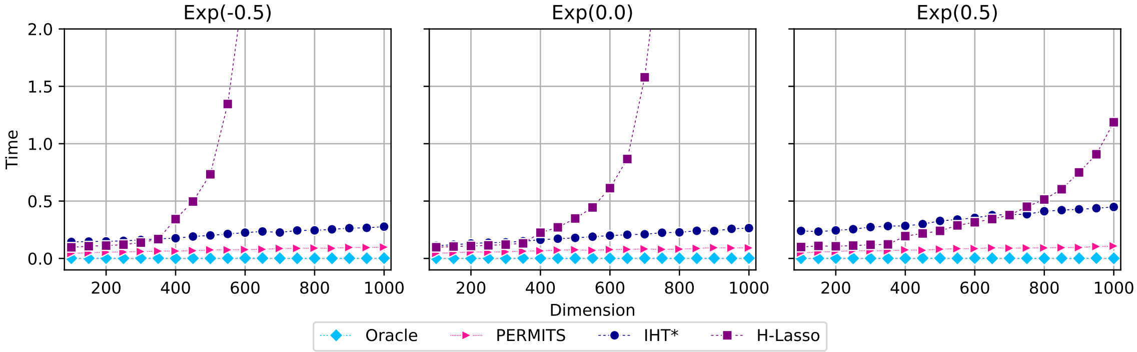


Figure 4: The dimensions (x-axis) versus running time (y-axis). Fix $n = 500$ and p ranges from 100 to 1000 with step size 50. The averaged running time of 50 repeated experiments is plotted.

support set $\text{supp}(\boldsymbol{w}^*)$. Therefore, this DJIA dataset serves as an appropriate benchmark dataset that enables the evaluation and comparison of methods for simplex-constrained sparse optimization. The optimization is affected by the correlation structure of the $\boldsymbol{X}$, which is visualized in the left panel of Figure 5. As we can see from Figure 5, the correlation structure is different from that in synthetic data, as the correlation may not exponentially decay. Hence, this real-world dataset serves as a complement to synthetic data analysis to help us understand the empirical performance of PERMIT at different and more challenging correlation settings.

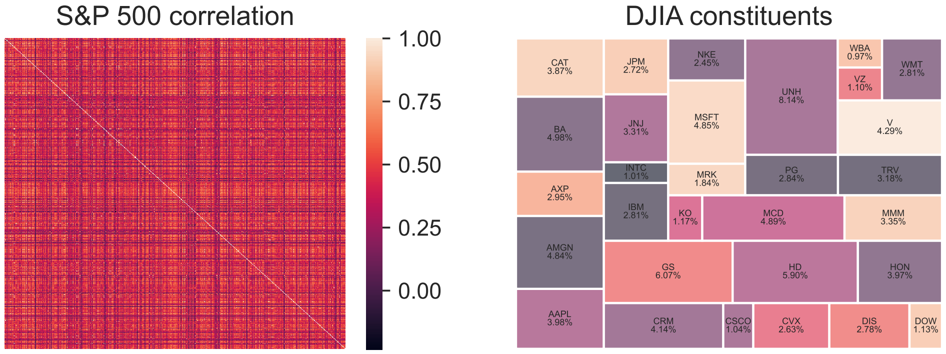


Figure 5: Basic information of the DJIA index. The left panel shows the correlation matrix of 490 stocks in our pool, and the right panel shows the constituents of DJIA whose sizes are determined by the weight $\boldsymbol{w}^*$.

Table 1 presents the performance of the three methods, evaluated using four metrics: (i) correctly detected stocks, i.e., stocks in the DJIA identified by the method; (ii) wrongly detected stocks, i.e., stocks not in the DJIA but selected by the method; (iii) accuracy, i.e., the proportion of correctly detected stocks relative to the total number of detected and missed stocks; and (iv) ℓ_2-error in estimation.

Table 1: Performance of detection of DJIA constituents.

Method	Correctly detected	Wrongly detected	Accuracy	ℓ_2 Error
H-Lasso	**30**	58	34% (30/88)	0.040
IHT*	9	21	18% (9/51)	0.226
PERMITS	28	**0**	**93% (28/30)**	**0.038**

As shown in Table 1, PERMITS detects 28 constituents of DJIA and misses 2 constituents: KO (with weight 1.17%), WBA (with weight 0.97%) whose weights are too small to be distinguished from noise; see Figure 5. Although H-Lasso detects all 30 stocks, it includes 58 in other wrong ones, and this is not an ideal result. Besides, IHT* only rightly detects a few constituents, reflecting that it has less power to identify stocks in the DJIA. This may be due to the fact that the signal is too weak. IHT* would also identify some constituents despite being less than H-Lasso. In summary, our method is the only one that finds almost all constituents without any false discovery. And notably, PERMITS also achieves the least estimated error among these methods.

4.3 Efficiency of tail screening procedure

In this subsection, we perform numerical experiments to show the efficiency of the tail screening procedure. Specifically, we compare the outputs of the following two methods.

i) Without tail screening (TS): the empirical risk minimizer of the original problem

$$\min_{\boldsymbol{w}\in\mathbb{R}^p} f(\boldsymbol{w}) = \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|_2^2 \quad \text{s.t. } \boldsymbol{w}\in\Delta.$$

The solution of this problem is obtained as the numerical solution of CVXPY (Diamond and Boyd, 2016), an open-source Python library for convex optimization problems. We

Table 2: Comparison of methods with and without the tail screening procedure across different values of n, p and s^*.

(n, p, s^*)	Method	Accuracy	Sparsity	ℓ_2 Error	Time
$(500, 500, 5)$	Without TS	0.13 (0.09)	148.44 (185.92)	0.10 (0.02)	1.52 (0.32)
	With TS	0.97 (0.06)	5.16 (0.37)	0.05 (0.02)	0.02 (0.00)
$(500, 1000, 10)$	Without TS	0.11 (0.08)	249.58 (230.69)	0.11 (0.02)	3.95 (1.04)
	With TS	0.98 (0.05)	10.26 (0.53)	0.05 (0.02)	0.30 (0.07)
$(1000, 1000, 20)$	Without TS	0.12 (0.09)	302.98 (183.42)	0.07 (0.01)	8.00 (2.16)
	With TS	0.99 (0.02)	20.10 (0.30)	0.03 (0.01)	0.32 (0.06)
$(1000, 2000, 30)$	Without TS	0.08 (0.05)	487.34 (168.99)	0.07 (0.01)	31.11 (7.68)
	With TS	0.97 (0.04)	29.38 (0.85)	0.04 (0.01)	0.57 (0.26)

solve the problem using CVXPY with its default settings. By default, CVXPY calls the solver most specialized to the problem type. Specifically, for the simplex constrained least squares problem here, CVXPY calls the Operator Splitting Quadratic Program (OSQP) solver (Stellato et al., 2020). We use the OSQP solver's default hyperparameters: a maximum number of 10^4 iterations, an absolute accuracy tolerance of 10^{-5}, a relative accuracy tolerance of 10^{-5}.

ii) With TS: the output of our proposed PERMITS algorithm.

The comparisons are conducted on the synthetic datasets, following the same data-generating process in Section 4.1. Four criteria are designed to assess the two methods. Two of these criteria (i.e., accuracy and ℓ_2 error) are adopted from Section 4.1. As for the remaining two, *sparsity* denotes the ℓ_0 norm of the output, and *time* is the runtime of methods, measured in seconds. The mean values (with their standard deviations in the parentheses) of 50 replicated simulated data are reported in Table 2.

Table 2 demonstrates that the proposed PERMITS algorithm is significantly faster than a standard convex optimization solver. This efficiency arises from two key factors: (i) leveraging the self-regularization property to eliminate unnecessary iterations and (ii) reducing the effective dimension of the iterates $\boldsymbol{w}^t$ through the tail screening procedure. Beyond computational benefits, the tail screening procedure also enhances estimation quality. By sparsifying the solution, it ensures that the estimated sparsity level closely matches the true support size s^*. As a result, both the accuracy of support recovery S^* and the ℓ_2-error in estimating $\boldsymbol{w}^*$ improve significantly. These findings highlight that tail screening not only accelerates computation but also strengthens statistical estimation.

5. Conclusion and Discussion

In this work, we study the statistical and computational properties of the PG method applied to solve the simplex-constrained least squares problem. By bridging the gap between statistics and optimization, we extend the self-regularization property from the minimizer to the iterate of the PG algorithm. Without any regularization, the iterate of the PG algorithm could approach the optimal statistical error at a geometric rate. Furthermore, we propose the PERMITS algorithm, which can accurately recover the true support set, demonstrating its widespread applicability in real-world tasks. Numerical experiments validate the effectiveness of both the statistical and computational performance of the PERMITS method.

Several potential extensions of this work may be considered. First, the variable selection consistency is proven under the mutual incoherence condition, which may be overly stringent. In the numerical experiments, we observe that PERMITS still performs well in the highly correlated setting, and thus, we anticipate relaxing this condition to a weaker one, such as the restricted strong convexity condition as used in the proof of the ℓ_2 error. Second, the self-regularization in our work is actually due to the geometric property of the simplex. We expect to extend the self-regularization to some more general constrained sets, such as polyhedra. Lastly, similar statistical properties may be extended to other models, such as the generalized linear model and group linear model (Zhu et al., 2022; Zhang et al., 2023).

Acknowledgments

We would like to thank the two referees for their constructive comments and valuable suggestions, which have substantially improved the paper.

Wang's research is partially supported by National Natural Science Foundation of China grants No.12231017, 72171216, 71921001, and 71991474.

Appendix A. Additional Experiments

In this part, we perform some extended experiments with the same setting as Section 4 to show the effect of the tolerance parameter ϵ. In Section 4, our proposed PERMITS method is implemented with default value $\epsilon = 10^{-4}\sqrt{\log(p)/n}$. Here, we show the results of two other choices such that

$$\epsilon \in \left\{10^{-3}\sqrt{\frac{\log p}{n}}, 10^{-4}\sqrt{\frac{\log p}{n}}, 10^{-5}\sqrt{\frac{\log p}{n}}\right\},$$

and thus these methods are referred to as PERMITS(e-3), PERMITS(e-4), and PERMITS(e-5), respectively. Three criteria (support accuracy, ℓ_2 error, and running time) compared with some benchmark methods are shown in Figures 6-8. We conclude that all these three choices of ϵ are appropriate for all simulation settings, while there is a slight difference in running time.

Appendix B. Details of the Backtracking Procedure

Recall that given the current iterate $\boldsymbol{w}$ and a guess value $M > 0$ of the Lipschitz constant L_f, we can make a projected gradient iteration by substituting $f(\boldsymbol{u})$ with a quadratic

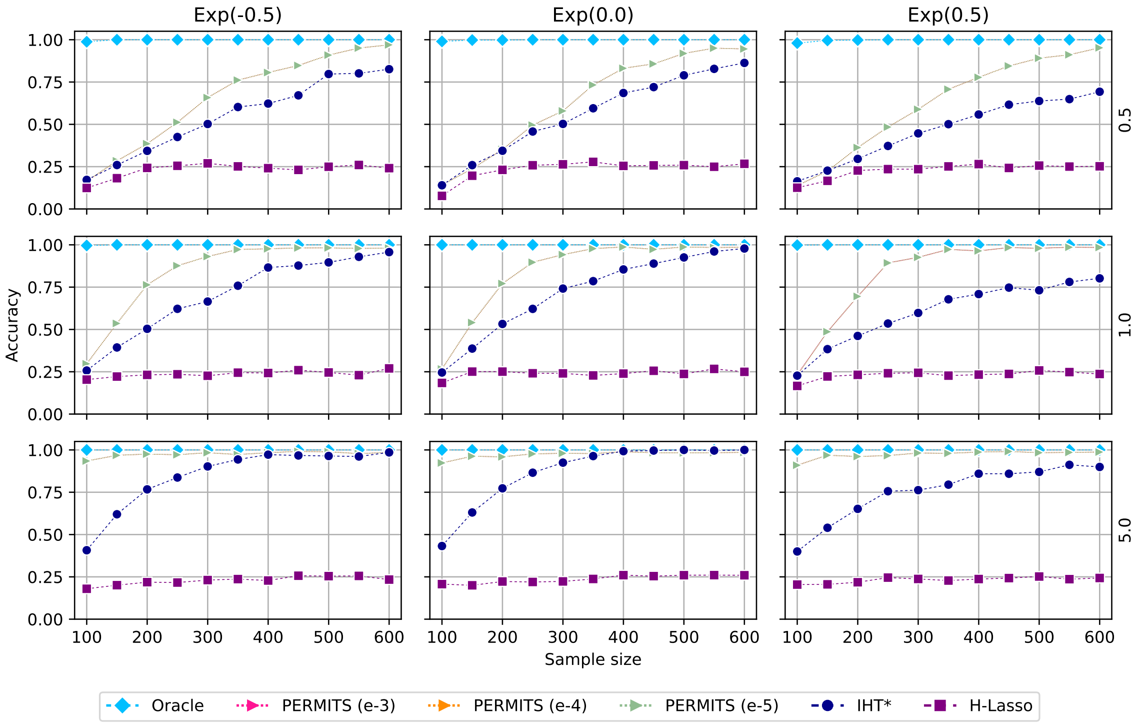


Figure 6: The sample size (x-axis) versus variable selection accuracy (y-axis). We fix $p = 1000, s^* = 10$. Each sub-figure corresponds to a specific choice of SNR (in different rows) and correlation structure (in different columns). The sample size n ranges from 100 to 600 with step 50. We perform 50 repeated experiments, and their mean values are plotted. PERMITS with three different tolerance parameters are compared with some benchmark methods and the oracle method.

approximation at $\boldsymbol{w}$ and solve the following minimization problem

$$\begin{aligned}\boldsymbol{w}^+ &= \arg\min_{\boldsymbol{u}\in\mathbb{R}^p}\left\{f(\boldsymbol{w}) + \langle\nabla f(\boldsymbol{w}), \boldsymbol{u}-\boldsymbol{w}\rangle + \frac{M}{2}\|\boldsymbol{u}-\boldsymbol{w}\|_2^2 + \chi_\Delta(\boldsymbol{u})\right\}\\ &= \mathcal{P}_\Delta(\boldsymbol{w} - M^{-1}\nabla f(\boldsymbol{w})).\end{aligned}$$

The quality of this new iteration $\boldsymbol{w}^+$ is affected by the step size M, i.e., the guess value of the Lipschitz constant. To obtain a sufficient decrease, we need to check whether M satisfies:

$$f(\boldsymbol{w}^+) \le f(\boldsymbol{w}) + \langle\nabla f(\boldsymbol{w}), \boldsymbol{w}^+ - \boldsymbol{w}\rangle + \frac{M}{2}\|\boldsymbol{w}^+ - \boldsymbol{w}\|_2^2. \tag{4}$$

If inequality (4) holds, we adopt $\boldsymbol{w}^+$ as the next iterate. Otherwise, we increase the guess M by a factor $\gamma > 1$ to a better guess γM and repeat the above projected gradient iteration until inequality (4) holds. This repetition will terminate in finite times by the Lipschitz smooth condition (C1) whenever M is larger than the Lipschitz constant L_f. We summarize the iterative procedure for finding M in Algorithm 3.

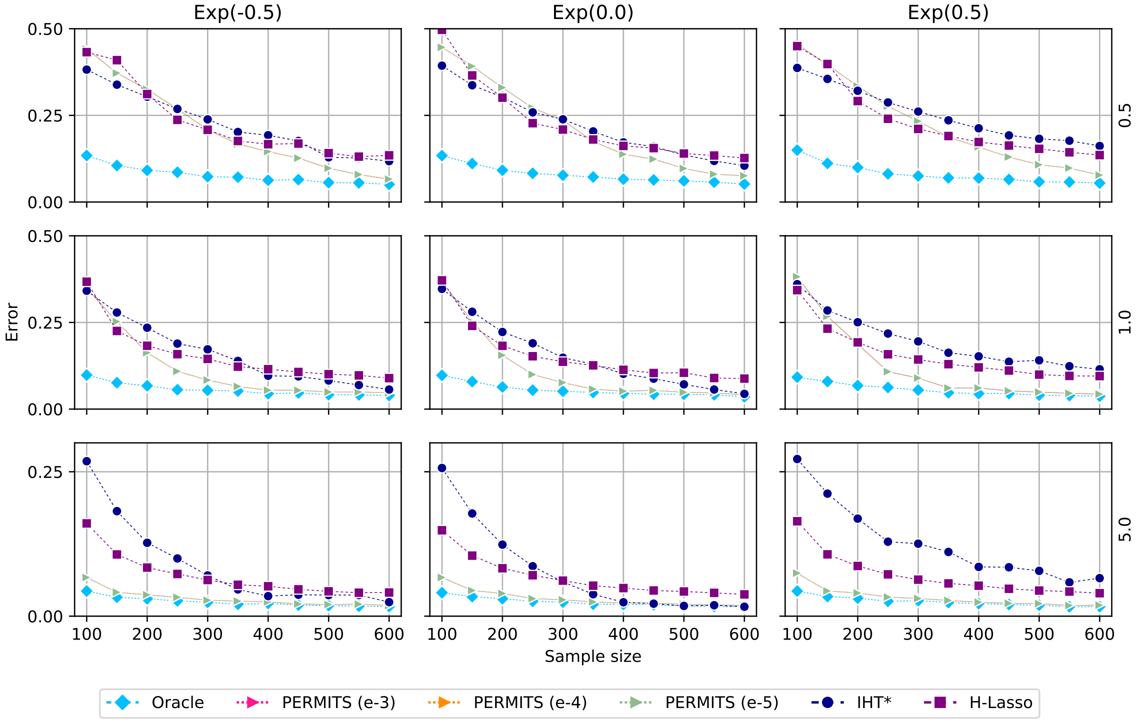


Figure 7: The sample size (x-axis) versus ℓ_2 error of parameter estimation (y-axis). The remaining settings are the same as Figure 6.

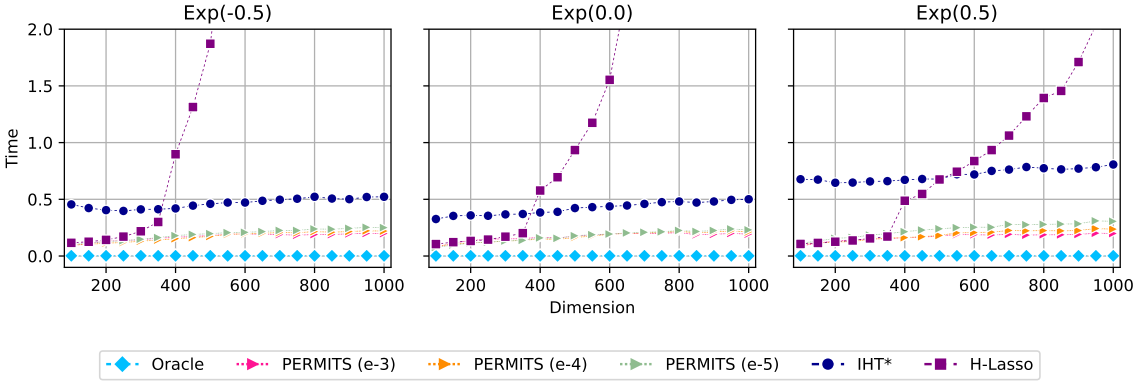


Figure 8: The dimensions (x-axis) versus running time (y-axis). Fix $n = 500$ and p ranges from 100 to 1000 with step size 50. The averaged running time of 50 repeated experiments are plotted.

Algorithm 3 Backtracking procedure for finding M

```
Input: $M > 0, \gamma > 1$.
while sufficient decrease condition (4) is not satisfied do
    $\boldsymbol{w}^+ \leftarrow \mathcal{P}_\Delta \left[\boldsymbol{w} - M^{-1}\nabla f(\boldsymbol{w})\right]$
    $M \leftarrow \gamma M$
Output: $M$
```

Remark 17 *The input parameter M of Algorithm 3 is the initial guess of the t-th iteration and may vary across different iterations. We set it as $\max\{L^0, \tilde{M}/\gamma\}$ where $\tilde{M}$ is the last value of M in the $(t-1)$-th iteration in Algorithm 1 and more details could be found in Nesterov (2013).*

Appendix C. Technical Proofs

In this part, we prove the corresponding computational and statistical properties. The main body of this paper studies the case where f is the least squares loss function. Here, we give the proofs of the properties on ℓ_2 error and linear convergence rate for the general differentiable convex function f. So, for the least squares loss, i.e., $f(\boldsymbol{w}) = \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2$, the properties hold as special cases. Recall that $\boldsymbol{w}^*$ is the parameter of interest that we want to recover, and f satisfies Condition (C1) and Condition (C2), where the latter implies that for any $\boldsymbol{w} \in \Delta$, it holds that

$$\begin{aligned} f(\boldsymbol{w}^*) &\geq f(\boldsymbol{w}) + \langle \nabla f(\boldsymbol{w}), \boldsymbol{w}^* - \boldsymbol{w}\rangle + \frac{\mu_f}{2}\|\boldsymbol{w}^* - \boldsymbol{w}\|_2^2, \\ f(\boldsymbol{w}) &\geq f(\boldsymbol{w}^*) + \langle \nabla f(\boldsymbol{w}^*), \boldsymbol{w} - \boldsymbol{w}^*\rangle + \frac{\mu_f}{2}\|\boldsymbol{w} - \boldsymbol{w}^*\|_2^2. \end{aligned}$$

C.1 Proof of Proposition 8

The following proof extends Proposition 2 of Li et al. (2020).

Proof Let $\boldsymbol{w}$ be any point in $G(\boldsymbol{w}^*)$, that is $f(\boldsymbol{w}) \leq f(\boldsymbol{w}^*)$. Then

$$\begin{aligned} 0 &\geq f(\boldsymbol{w}) - f(\boldsymbol{w}^*) \\ &\geq \langle \nabla f(\boldsymbol{w}^*), \boldsymbol{w} - \boldsymbol{w}^*\rangle + \frac{\mu_f}{2}\|\boldsymbol{w} - \boldsymbol{w}^*\|_2^2 \\ &\geq \frac{\mu_f}{2}\|\boldsymbol{w} - \boldsymbol{w}^*\|_2^2 - \|\nabla f(\boldsymbol{w}^*)\|_\infty \|\boldsymbol{w} - \boldsymbol{w}^*\|_1 \\ &\geq \frac{\mu_f}{2}\|\boldsymbol{w} - \boldsymbol{w}^*\|_2^2 - 2\sqrt{s^*}\|\nabla f(\boldsymbol{w}^*)\|_\infty \|\boldsymbol{w} - \boldsymbol{w}^*\|_2 \end{aligned}$$

where the second inequality follows from Condition (C2), the third inequality is due to Holder's inequality, and the last one follows from the property of simplex that for any $\boldsymbol{w} \in \Delta$, let S^* be the support set of $\boldsymbol{w}^*$, then

$$\begin{aligned} \|\boldsymbol{w} - \boldsymbol{w}^*\|_1 &= \sum_{i \in S^*} |w_i - w_i^*| + \sum_{i \notin S^*} |w_i - w_i^*| \\ &= \sum_{i \in S^*} |w_i - w_i^*| + \sum_{i \notin S^*} w_i \\ &= \sum_{i \in S^*} |w_i - w_i^*| + 1 - \sum_{i \in S^*} w_i \\ &= \sum_{i \in S^*} |w_i - w_i^*| + \sum_{i \in S^*} (w_i^* - w_i) \\ &\leq 2\|\boldsymbol{w}_{S^*} - \boldsymbol{w}_{S^*}^*\|_1 \\ &\leq 2\sqrt{s^*}\|\boldsymbol{w}_{S^*} - \boldsymbol{w}_{S^*}^*\|_2 \\ &\leq 2\sqrt{s^*}\|\boldsymbol{w} - \boldsymbol{w}^*\|_2. \end{aligned}$$

Thus, $\|\boldsymbol{w} - \boldsymbol{w}^*\|_2 \leq \frac{4\sqrt{s^*}\|\nabla f(\boldsymbol{w}^*)\|_\infty}{\mu_f}$. Using the fact that

$$\|\nabla f(\boldsymbol{w}^*)\|_\infty = \|n^{-1}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{w}^* - \boldsymbol{y})\|_\infty = \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty,$$

we have $\boldsymbol{w} \in B(\boldsymbol{w}^*)$. Since $\boldsymbol{w} \in G(\boldsymbol{w}^*)$ is arbitrary, it holds that $G(\boldsymbol{w}^*) \subset B(\boldsymbol{w}^*)$. ■

C.2 Proof of Theorem 10

Proof First, we prove that before entering the region $G(\boldsymbol{w}^*)$, the convergence rate of $\boldsymbol{w}^t$ is geometric. Assume that $\boldsymbol{w}^t$ is outside of $G(\boldsymbol{w}^*)$, that is $f(\boldsymbol{w}^t) > f(\boldsymbol{w}^*)$. The projected gradient iteration performs the backtracking strategy for selecting the step length or, equivalently, M^t to guarantee the sufficient decrease that

$$f(\boldsymbol{w}^t) \leq f(\boldsymbol{w}^{t-1}) + \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^t - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2^2.$$

Thus, denote $\boldsymbol{z}^t = \boldsymbol{w}^{t-1} - \frac{1}{M^t}\nabla f(\boldsymbol{w}^{t-1})$, we have

$$\begin{aligned}
& f(\boldsymbol{w}^t) - f(\boldsymbol{w}^{t-1}) \\
\leq & \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^t - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2^2 \\
= & \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^t - \boldsymbol{w}^*\rangle + \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2^2 \\
= & M^t\langle \boldsymbol{w}^{t-1} - \boldsymbol{z}^t, \boldsymbol{w}^t - \boldsymbol{w}^*\rangle + \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2^2 \\
\leq & M^t\langle \boldsymbol{w}^{t-1} - \boldsymbol{w}^t, \boldsymbol{w}^t - \boldsymbol{w}^*\rangle + \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2^2 \\
= & \frac{M^t}{2}\left[\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 - \|\boldsymbol{w}^{t-1} - \boldsymbol{w}^t\|_2^2 - \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2^2\right] \\
& \quad + \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2^2 \\
= & \langle \nabla f(\boldsymbol{w}^{t-1}), \boldsymbol{w}^* - \boldsymbol{w}^{t-1}\rangle + \frac{M^t}{2}\left[\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 - \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2^2\right] \\
\leq & f(\boldsymbol{w}^*) - f(\boldsymbol{w}^{t-1}) - \frac{\mu_f}{2}\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 + \frac{M^t}{2}\left[\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 - \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2^2\right].
\end{aligned}$$

The first inequality is guaranteed by the sufficient decrease property of PG, and the second equality follows from the definition of $\boldsymbol{z}^t$. The second inequality is due to the convex projection property that $\langle \boldsymbol{w}^t - \boldsymbol{z}^t, \boldsymbol{w}^t - \boldsymbol{w}^*\rangle \leq 0$. The third equality uses the fact that $2\langle a, b\rangle = \|a\|_2^2 + \|b\|_2^2 - \|a - b\|_2^2$. The last inequality follows from the restricted strongly convex condition (C2) and the fact that $\boldsymbol{w}^*, \boldsymbol{w}^{t-1} \in \Delta, \mathbf{1}^\top(\boldsymbol{w}^* - \boldsymbol{w}^{t-1}) = 1 - 1 = 0$ and

$(\boldsymbol{w}^* - \boldsymbol{w}^{t-1})_{(S^*)^c} = \mathbf{0} - \boldsymbol{w}^{t-1}_{(S^*)^c} = -\boldsymbol{w}^{t-1}_{(S^*)^c} \leq \mathbf{0}$. By the assumption $f(\boldsymbol{w}^t) > f(\boldsymbol{w}^*)$, we have

$$\begin{aligned} 0 &< f(\boldsymbol{w}^t) - f(\boldsymbol{w}^*) \\ &\leq -\frac{\mu_f}{2}\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 + \frac{M^t}{2}\left[\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 - \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2^2\right] \\ &\leq \frac{M^t}{2}\left[(1 - \frac{\mu_f}{M^t})\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 - \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2^2\right]. \end{aligned}$$

Therefore, $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2^2 < \left(1 - \frac{\mu_f}{M^t}\right)\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2 \leq \exp\left(-\frac{\mu_f}{M^t}\right)\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2^2$. Since the sufficient decrease condition holds for any $L \geq L_f$, by the backtracking strategy, we have for any $t \geq 0$ that $M^t \leq L^0 \vee (\gamma L_f)$. Hence,

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq \exp\left(-\frac{1}{2}\frac{\mu_f}{L^0 \vee (\gamma L_f)}\right)\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2.$$

Applying the above result recursively gives

$$\begin{aligned} \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 &\leq \exp\left(-\frac{1}{2}\frac{t\mu_f}{L^0 \vee (\gamma L_f)}\right)\|\boldsymbol{w}^0 - \boldsymbol{w}^*\|_2 \\ &\leq \sqrt{2}\exp\left(-\frac{1}{2}\frac{t\mu_f}{L^0 \vee (\gamma L_f)}\right) \end{aligned}$$

where the last inequality uses the fact that $\|\boldsymbol{w} - \boldsymbol{u}\|_2^2 \leq \|\boldsymbol{w} - \boldsymbol{u}\|_1 \leq 2$ for any $\boldsymbol{w}, \boldsymbol{u} \in \Delta$. This means that when $f(\boldsymbol{w}^t) > f(\boldsymbol{w}^*)$, that is $\boldsymbol{w}^t$ is outside of $G(\boldsymbol{w}^*)$, $\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2$ converges at a linear rate. Since PG is a descent algorithm, we know that $\boldsymbol{w}^t$ approaches $G(\boldsymbol{w}^*)$ in a linear rate and will stay in $G(\boldsymbol{w}^*)$ henceforth. Therefore, there exists $T_0 > 0$ such that for any $t \geq T_0$, $\boldsymbol{w}^t \in G(\boldsymbol{w}^*)$,

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq \frac{4\sqrt{s^*}\|\nabla f(\boldsymbol{w}^*)\|_\infty}{\mu_f}.$$

Combine the above arguments, and we have that for any $t = 0, 1, 2, \cdots$

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq \max\left\{\sqrt{2}\exp\left(-\frac{1}{2}\frac{t\mu_f}{L^0 \vee (\gamma L_f)}\right), \frac{4\sqrt{s^*}\|\nabla f(\boldsymbol{w}^*)\|_\infty}{\mu_f}\right\}.$$

When f is the linear least squares loss, Lemma 18 gives that

$$\|\nabla f(\boldsymbol{w}^*)\|_\infty \leq C\sqrt{\frac{s^*\sigma^2 \log p}{n}}$$

holds with probability at least $1 - O(p^{-3})$. The proof is completed. ■

C.3 Proof of Theorem 13

Proof Before the statistical precision is attained, we can lower bound the difference $r^t := \|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2$ via $\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2$ as follows. From the proof of Theorem 10, when $\boldsymbol{w}^t$ is outside of $G(\boldsymbol{w}^*)$, we have the following one-step progress

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq e^{-\kappa_1}\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2$$

where $\kappa_1 = \frac{1}{2}\frac{\mu_f}{L^0 \vee (\gamma L_f)} > 0$. By the triangle inequality, we have

$$\begin{aligned} r^t = \|\boldsymbol{w}^t - \boldsymbol{w}^{t-1}\|_2 &\geq \|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2 - \|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \\ &\geq (1 - e^{-\kappa_1})\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2. \end{aligned}$$

For any tolerance parameter $\epsilon > 0$, if PG algorithm terminate ($r^t \leq \epsilon$), it must hold that either $\boldsymbol{w}^t$ is outside $G(\boldsymbol{w}^*)$ that

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq e^{-\kappa_1}\|\boldsymbol{w}^{t-1} - \boldsymbol{w}^*\|_2 \leq \frac{r^t}{e^{\kappa_1} - 1} \leq \frac{\epsilon}{e^{\kappa_1} - 1}$$

or $\boldsymbol{w}^t$ is inside $G(\boldsymbol{w}^*)$ that

$$\|\boldsymbol{w}^t - \boldsymbol{w}^*\|_2 \leq \frac{C}{\mu_f}\sqrt{\frac{s^*\sigma^2 \log p}{n}}.$$

Hence, if we choose $\epsilon \leq \frac{C(e^{\kappa_1}-1)}{\mu_f}\sqrt{\frac{s^*\sigma^2 \log p}{n}}$, the PG algorithms will terminate and output a solution $\boldsymbol{w}$ satisfying

$$\|\boldsymbol{w} - \boldsymbol{w}^*\|_2 \leq \frac{C}{\mu_f}\sqrt{\frac{s^*\sigma^2 \log p}{n}}.$$

This completes the proof of Theorem 13. ■

C.4 Proof of Theorem 15

Proof Let

$$\boldsymbol{w}^{\Diamond} := \arg\min_{\boldsymbol{w} \in \Delta} f(\boldsymbol{w}) \text{ s.t. } \boldsymbol{w}_{(S^*)^c} = \mathbf{0}$$

and

$$\bar{\boldsymbol{w}} := \arg\min_{\boldsymbol{w} \in \Delta} f(\boldsymbol{w}) \text{ s.t. } \boldsymbol{w}_{S^c} = \mathbf{0}$$

be the minimizers of the original optimization constrained on S^* and S respectively. Correspondingly, denote $\widetilde{\boldsymbol{w}}$ and $\widehat{\boldsymbol{w}}$ be the output of the PG algorithm supported on S^* and S respectively. By the definition of the minimizer, we have that

$$f(\widetilde{\boldsymbol{w}}) \geq f(\boldsymbol{w}^{\Diamond}), \quad f(\widehat{\boldsymbol{w}}) \geq f(\bar{\boldsymbol{w}}).$$

From the ℓ_2 error bound in Theorem 13 and the minimal signal condition (C4), we have that $S^t := \operatorname{supp}(\boldsymbol{w}^t) \supset S^*$ for small t and S^t decreases as t increases. To simplify the notation, we denote S as S^t. Hence, only three cases can happen in the iteration path of PERMITS: $S \supset S^*, S = S^*$ or $S \subset S^*$. To prove this theorem, it is sufficient to show that $\text{SIC}(S^*) < \text{SIC}(S)$ for any $S \neq S^*$.

We split our proof into the following two cases: (I) $|S| > |S^*|$ and (II) $|S| < |S^*|$. We only prove for the first case (I) since similar arguments hold for case (II). Now, suppose that $|S| > |S^*|$ holds. From Theorem 13, we have that all true variables are included in the current support set, that is, $S \supset S^*$. Denote $S = S^* \cup B$ with $|B| > 0$ and $S^* \cap B = \emptyset$. We only need to prove that

$$f(\widetilde{\boldsymbol{w}}) - f(\widehat{\boldsymbol{w}}) \lesssim |B| \frac{\sigma^2 \log p}{n}.$$

From Lemma 20, we have for the optimal solutions $\boldsymbol{w}^{\diamondsuit}$ and $\bar{\boldsymbol{w}}$ that

$$f(\boldsymbol{w}^{\diamondsuit}) - f(\bar{\boldsymbol{w}}) \lesssim |B| \frac{\sigma^2 \log p}{n}.$$

By the definition of minimizer $\bar{\boldsymbol{w}}$, it holds that $f(\bar{\boldsymbol{w}}) \leq f(\widehat{\boldsymbol{w}})$ and thus

$$f(\boldsymbol{w}^{\diamondsuit}) - f(\widehat{\boldsymbol{w}}) \lesssim |B| \frac{\sigma^2 \log p}{n}.$$

Hence, it remains to bound the optimization error over S^* such that

$$f(\widetilde{\boldsymbol{w}}) - f(\boldsymbol{w}^{\diamondsuit}) \leq O\big(\frac{\sigma^2 \log p}{n}\big).$$

By the warm start in the PERMITS algorithm, for each $t > 1$, we have that the initial error is no more than $O\left(\frac{s^* \sigma^2 \log p}{n}\right)$. Note that $\widetilde{\boldsymbol{w}}, \boldsymbol{w}^{\diamondsuit}$ are now constrained on the support S^* with size $|S^*| = s^* \ll n$. That is, the design matrix $\boldsymbol{X}$ has only s^* rather than $p \gg n$ columns. Therefore, the condition (C5) implies that f is **globally strongly convex** over this low-dimensional support S^*. Therefore, the iterate $\widetilde{\boldsymbol{w}}$ within this problem **converges exactly** in a linear rate to $\boldsymbol{w}^{\diamondsuit}$ rather than only to a neighbourhood (in contrast to the result of Theorem 10 where only the convergence to a neighbourhood is guaranteed since we only require a weaker **restricted strong convexity** holds there). Thus, the initial optimization error $O\left(\frac{s^* \sigma^2 \log p}{n}\right)$ can be decreased to 0 linearly and after $T_{\min} = \log s^* / \log(\kappa_2^{-1})$ iterations, the optimization error decreases to $O\left(\frac{\sigma^2 \log p}{n}\right)$. Hence, we have shown that

$$f(\widetilde{\boldsymbol{w}}) - f(\widehat{\boldsymbol{w}}) \lesssim |B| \frac{\sigma^2 \log p}{n}.$$

Using the inequality $\log(x) \leq x - 1$ and the fact $f(\widehat{\boldsymbol{w}}) > 0$, we have

$$\begin{aligned} \mathrm{SIC}(S^*) - \mathrm{SIC}(S) &= n \log \frac{f(\widetilde{\boldsymbol{w}})}{f(\widehat{\boldsymbol{w}})} - |B| \log(p) \log\log n \\ &\leq n \frac{f(\widetilde{\boldsymbol{w}}) - f(\widehat{\boldsymbol{w}})}{f(\widehat{\boldsymbol{w}})} - |B| \log(p) \log\log n \\ &\leq O\left(|B| \sigma^2 \log p\right) - |B| \log(p) \log\log n < 0 \end{aligned}$$

for some sufficiently large n. The proof is completed. ■

C.5 Technical Lemmas

Without loss of generality, we assume that $\|X_j\|_2/\sqrt{n} \le C$ holds for some constant $C > 0$ in this paper.

Lemma 18 *Suppose that* $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}^* + \boldsymbol{\xi}$ *where* $\xi_i, i \in [n]$ *are i.i.d. sub-Gaussian random variables and* $f(\boldsymbol{w}) = (2n)^{-1}\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2$ *is the least squares loss function. Then, we have with probability at least* $1 - (8p^3)^{-1}$ *that*

$$\|\nabla f(\boldsymbol{w}^*)\|_\infty = \|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty \lesssim \sqrt{\frac{\sigma^2 \log p}{n}}.$$

Proof By definition, $\frac{1}{\sqrt{n}} X_j^\top \boldsymbol{\xi}$ is a sub-Gaussian random variable. Using the union bound, we have

$$\mathbb{P}\left(\left\|\frac{1}{\sqrt{n}}\boldsymbol{X}^\top \boldsymbol{\xi}\right\|_\infty \ge t\right) \le \sum_{j=1}^{p} \mathbb{P}\left(\left|\frac{1}{\sqrt{n}} X_j^\top \boldsymbol{\xi}\right| \ge t\right) \le 2p \exp\left(-\frac{t^2}{2C^2\sigma^2}\right).$$

Setting $t = 2\sqrt{2C^2\sigma^2 \log(2p)}$, we have with probability at least $1 - (8p^3)^{-1}$ that

$$\|\nabla f(\boldsymbol{w}^*)\|_\infty = \|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty \le \frac{1}{\sqrt{n}} \cdot 2\sqrt{2C^2\sigma^2 \log(2p)} \lesssim \sqrt{\frac{\sigma^2 \log p}{n}}.$$

■

From Proposition 8, we can derive a ℓ_∞ bound that $\|\widehat{\boldsymbol{w}}_n - \boldsymbol{w}^*\|_\infty \lesssim \sqrt{\frac{\sigma^2 s^* \log p}{n}}$ which still depends on s^*. The following Lemma says that this bound can be improved to $\sqrt{\frac{\sigma^2 \log p}{n}}$ if the mutual incoherence condition (C5) holds.

Lemma 19 (ℓ_∞ error bound) *Suppose Conditions (C3)-(C5) hold. Let* $(\bar{\boldsymbol{w}}, \bar{\alpha}, \bar{\boldsymbol{\beta}})$ *be a KKT pair of problem* (1)*. Then, with probability at least* $1 - O(p^{-3})$*, we have*

$$\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_\infty \lesssim \sqrt{\frac{\sigma^2 \log p}{n}} \text{ and } |\bar{\alpha}| \lesssim \sqrt{\frac{\sigma^2 \log p}{n}}.$$

Proof of Lemma 19. We claim that the mutual incoherence parameter $\rho \le \frac{1}{16s^*}$ in (C5) is sufficient. Our proof is divided into two parts.
Part I: $\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_\infty \lesssim \|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty$.
Denote $\mathcal{J} \coloneqq \operatorname{supp}(\bar{\boldsymbol{w}})$ and we consider respectively the following three cases

$$\begin{aligned}
&\text{(a) } \mathcal{J} \cap (S^*)^c \ne \emptyset, \quad \mathcal{J}^c \cap S^* \ne \emptyset. \\
&\text{(b) } \mathcal{J} \cap (S^*)^c \ne \emptyset, \quad \mathcal{J}^c \cap S^* = \emptyset. \\
&\text{(c) } \mathcal{J} \cap (S^*)^c = \emptyset.
\end{aligned}$$

We only provide detailed proof for case (a), and similar arguments can be applied to cases (b) and (c).

We prove (a) by noting that it will lead to a contradiction if

$$\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_\infty \geq 8\|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty.$$

Since $\mathcal{J} \cap (S^*)^c \neq \emptyset$ and $\mathcal{J}^c \cap S^* \neq \emptyset$, i.e., $\mathcal{J}$ includes some noise and excludes some signals, then there exist two indexes $j, k \in [p]$ such that $j \in \mathcal{J} \cap (S^*)^c$ and $k \in \mathcal{J}^c \cap S^*$. The KKT point $(\bar{\boldsymbol{w}}, \bar{\alpha}, \bar{\boldsymbol{\beta}})$ necessarily satisfies the following conditions

$$\begin{aligned}
n^{-1}\boldsymbol{X}^\top \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi} - \bar{\alpha}\mathbf{1} - \bar{\boldsymbol{\beta}} = \mathbf{0}, & \qquad \text{(stationary)} \\
\bar{\boldsymbol{w}} \geq \mathbf{0}, \mathbf{1}^\top \bar{\boldsymbol{w}} = 1, & \qquad \text{(primal feasibility)} \\
\bar{\boldsymbol{\beta}} \geq \mathbf{0}, & \qquad \text{(dual feasibility)} \\
\bar{\boldsymbol{w}} \odot \bar{\boldsymbol{\beta}} = \mathbf{0}. & \qquad \text{(complementary slackness)}
\end{aligned}$$

Since $j \in \mathcal{J} \cap (S^*)^c \subset \mathcal{J}$, we have $\bar{w}_j > 0, w^*_j = 0$ and thus $\bar{\beta}_j = 0$ by the complementary slackness condition. Then, the stationary condition reads

$$\begin{aligned}
(\bar{w}_j - w^*_j) + \sum_{i \neq j} n^{-1} X_j^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_j^\top \boldsymbol{\xi} - \bar{\alpha} = 0 \\
\Longrightarrow \bar{\alpha} = \bar{w}_j + \sum_{i \neq j} n^{-1} X_j^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_j^\top \boldsymbol{\xi} \\
> \sum_{i \neq j} n^{-1} X_j^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_j^\top \boldsymbol{\xi}.
\end{aligned}$$

Owing to $k \in \mathcal{J}^c \cap S^*$, we have $\bar{w}_k = 0, w^*_k > 0$ and $\bar{\beta}_k \geq 0$ by dual feasibility condition. Then, the stationary condition reads

$$\begin{aligned}
(\bar{w}_k - w^*_k) + \sum_{i \neq k} n^{-1} X_k^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_k^\top \boldsymbol{\xi} - \bar{\alpha} - \bar{\beta}_k = 0 \\
\Longrightarrow \bar{\alpha} = -w^*_k + \sum_{i \neq k} n^{-1} X_k^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_k^\top \boldsymbol{\xi} - \bar{\beta}_k \\
\leq -w^*_k + \sum_{i \neq k} n^{-1} X_k^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_k^\top \boldsymbol{\xi}
\end{aligned}$$

Therefore, by combining the above two inequalities, we have

$$\sum_{i \neq j} n^{-1} X_j^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_j^\top \boldsymbol{\xi} < -w^*_k + \sum_{i \neq k} n^{-1} X_k^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_k^\top \boldsymbol{\xi}.$$

Therefore, we can conclude that

$$\begin{aligned}
w^*_k &< \sum_{i \neq k} n^{-1} X_k^\top X_i(\bar{w}_i - w^*_i) - n^{-1}X_k^\top \boldsymbol{\xi} - \sum_{i \neq j} n^{-1} X_j^\top X_i(\bar{w}_i - w^*_i) + n^{-1}X_j^\top \boldsymbol{\xi} \\
&\leq 2\rho\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_1 + 2\|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty \\
&\leq 4\rho\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_1 + 2\|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty \\
&\leq \frac{1}{4}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_\infty + \frac{1}{4}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_\infty \\
&= \frac{1}{2}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_\infty.
\end{aligned}$$

This says that for any $k \in \mathcal{J}^c \cap S^*$,

$$|\bar{w}_k - w_k^*| = w_k^* < \frac{1}{2}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty < \|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty.$$

So, the maximum $\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty$ is necessarily attained on some $l \in \mathcal{J} \cap S^* \neq \emptyset$ such that

$$|\bar{w}_l - w_l^*| = \|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty.$$

For any $m \in \mathcal{J}$, by the stationary and complementary slackness conditions we have that

$$\begin{aligned}\bar{\alpha} &= (\bar{w}_m - w_m^*) + \sum_{i \neq m} n^{-1} X_m^\top X_i (\bar{w}_i - w_i^*) - n^{-1} X_m^\top \boldsymbol{\xi} \\ &= (\bar{w}_l - w_l^*) + \sum_{i \neq l} n^{-1} X_l^\top X_i (\bar{w}_i - w_i^*) - n^{-1} X_l^\top \boldsymbol{\xi}.\end{aligned}$$

If $\bar{w}_l - w_l^* < 0$, we have $\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty = |\bar{w}_l - w_l^*| = -(\bar{w}_l - w_l^*)$. Then for any $m \in \mathcal{J}$,

$$\begin{aligned}\bar{w}_m - w_m^* &= (\bar{w}_l - w_l^*) + \sum_{i \neq l} n^{-1} X_l^\top X_i (\bar{w}_i - w_i^*) - n^{-1} X_l^\top \boldsymbol{\xi} \\ &\quad - \sum_{i \neq m} n^{-1} X_m^\top X_i (\bar{w}_i - w_i^*) + n^{-1} X_m^\top \boldsymbol{\xi} \\ &\leq (\bar{w}_l - w_l^*) + 2\rho\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_1 + 2\|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty \\ &\leq (\bar{w}_l - w_l^*) + \frac{1}{2}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty \\ &= (\bar{w}_l - w_l^*) - \frac{1}{2}(\bar{w}_l - w_l^*) \\ &= \frac{1}{2}(\bar{w}_l - w_l^*).\end{aligned}$$

Sum m over $\mathcal{J}$, we have

$$\frac{|\mathcal{J}|}{2}(\bar{w}_l - w_l^*) \geq \sum_{m \in \mathcal{J}} (\bar{w}_m - w_m^*) = 1 - \sum_{m \in \mathcal{J}} w_m^* \geq 0$$

which contradicts to the fact that $\bar{w}_l - w_l^* < 0$.

Hence, $\bar{w}_l - w_l^* \geq 0$, i.e., $\|\bar{\boldsymbol{w}}_S - \boldsymbol{w}_S^*\|_\infty = |\bar{w}_l - w_l^*| = (\bar{w}_l - w_l^*)$. By a similar argument we have for any $m \in \mathcal{J}$ that

$$\bar{w}_m - w_m^* \geq \frac{1}{2}(\bar{w}_l - w_l^*).$$

Again, sum m over $\mathcal{J}$, we have

$$\begin{aligned}\frac{|\mathcal{J}|}{2} \cdot (\bar{w}_l - w_l^*) &\leq \sum_{m \in \mathcal{J}} (\bar{w}_m - w_m^*) \\ &= 1 - \sum_{m \in \mathcal{J}} w_m^* \\ &= \sum_{m \in S^*} w_m^* - \sum_{m \in \mathcal{J}} w_m^* \\ &= \sum_{m \in \mathcal{J}^c \cap S^*} w_m^*.\end{aligned}$$

While, we have proved for any $k \in \mathcal{J}^c \cap S$ that

$$w_k^* < \frac{1}{2}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty = \frac{1}{2}(\bar{w}_l - w_l^*).$$

Therefore, we have:

$$\frac{|\mathcal{J}|}{2}(\bar{w}_l - w_l^*) < \frac{|\mathcal{J}^c \cap S^*|}{2}(\bar{w}_l - w_l^*) \Longrightarrow |\mathcal{J}| < |\mathcal{J}^c \cap S^*| \le |S^*| = s^*.$$

That is, at least one parameter in S^* is estimated as 0, then there exists $k \in S^*$ such that $|\bar{w}_k - w_k^*| = w_k^* \ge b^*$. However, Proposition 8 implies that $\sqrt{s^*}\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty \gtrsim \|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_2 \ge |\bar{w}_k - w_k^*| = w_k^* \ge b^*$ which contradicts to condition(C4). The proof of Part I is completed.
Part II: $\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_\infty \lesssim \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty$ and $|\bar{\alpha}| \lesssim \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty$.
It remains to prove that $|\bar{w}_k - w_k^*| \lesssim \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty$ for any $k \notin S^*$. Since $\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty \lesssim \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty$, there exists at least one $j \in S^* \cap \mathcal{J}$ such that

$$(\bar{w}_j - w_j^*) + \sum_{i \ne j} n^{-1}X_j^\top X_i(\bar{w}_i - w_i^*) - n^{-1}X_j^\top\boldsymbol{\xi} - \bar{\alpha} = 0.$$

For any $k \notin S^*$, either $\bar{w}_k = 0$ (then the error $|\bar{w}_k - w_k^*| = 0$) or $\bar{w}_k > 0$ (then $\bar{\beta}_k = 0$) and thus

$$\begin{aligned}
&(\bar{w}_k - w_k^*) + \sum_{i \ne k} n^{-1}X_k^\top X_i(\bar{w}_i - w_i^*) - n^{-1}X_k^\top\boldsymbol{\xi} - \bar{\alpha} = 0.\\
\Longrightarrow\quad &|\bar{w}_k - w_k^*| = \bar{w}_k\\
&=(\bar{w}_j - w_j^*) + \sum_{i \ne j} n^{-1}X_j^\top X_i(\bar{w}_i - w_i^*) - n^{-1}X_j^\top\boldsymbol{\xi} - \sum_{i \ne k} n^{-1}X_k^\top X_i(\bar{w}_i - w_i^*) + n^{-1}X_k^\top\boldsymbol{\xi}\\
&\le\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty + 2\rho\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_1 + 2\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty\\
&\le\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty + \frac{1}{4}\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}_{S^*}^*\|_\infty + 2\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty\\
&\lesssim\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty
\end{aligned}$$

where the last inequality is a direct implication of Part I. Therefore, it holds that $\|\bar{\boldsymbol{w}} - \boldsymbol{w}^*\|_\infty \lesssim \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty$. Similarly, the optimal dual variable $\bar{\alpha}$ satisfies the same bound $|\bar{\alpha}| \lesssim \|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty$. Finally, the probabilistic result holds since the event

$$\left\{\|n^{-1}\boldsymbol{X}^\top\boldsymbol{\xi}\|_\infty \lesssim \sqrt{\sigma^2 \log p/n}\right\}$$

holds with probability at least $1 - O(p^{-3})$ by Lemma 18. ■

Lemma 20 *Suppose that conditions in Theorem 15 hold. Let $\boldsymbol{w}^\Diamond, \bar{\boldsymbol{w}} \in \Delta$ be minimizers of $f(\boldsymbol{w}) = (2n)^{-1}\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2$ constrained on S^* and S respectively. With probability at least $1 - O(p^{-3})$, the following two statements hold:*

(I) If $S = S^* \cup B$ for some nonempty set B, then we have for some constant $C > 0$ that

$$0 \le f(\boldsymbol{w}^\Diamond) - f(\bar{\boldsymbol{w}}) \le C|B|\frac{\sigma^2 \log p}{n}.$$

(II) If $S^* = S \cup B$ for some nonempty set B, then we have for some constant $c > 0$ that

$$f(\bar{\boldsymbol{w}}) - f(\boldsymbol{w}^\Diamond) \geq c|B|b_*^2.$$

Proof of Lemma 20.
Part (I): $S = S^* \cup B$.
Note that all arguments in the proof of Lemma 19 hold true by substituting $\boldsymbol{X}$ with a lower dimensional $\boldsymbol{X}_S$. Meanwhile, the probability $1 - O(p^{-3})$ is uniform since the upper bound for the noise level is uniform that

$$\|n^{-1}\boldsymbol{X}_S^\top \boldsymbol{\xi}\|_\infty \leq \|n^{-1}\boldsymbol{X}^\top \boldsymbol{\xi}\|_\infty, \quad \forall S.$$

Let $(\boldsymbol{w}^\Diamond, \alpha^\Diamond, \boldsymbol{\beta}^\Diamond)$ and $(\bar{\boldsymbol{w}}, \bar{\alpha}, \bar{\boldsymbol{\beta}})$ be the KKT pairs of the constrained problems corresponding to S^* and S respectively. The task is to bound the difference $f_{S^*} - f_S \coloneqq f(\boldsymbol{w}^\Diamond) - f(\bar{\boldsymbol{w}})$. Note that, $\bar{\boldsymbol{w}}_S^* > 0$ is trivial by Lemma 19 and the minimal signal condition (C4). Moreover, without loss of generality, we assume that $\bar{\boldsymbol{w}}_B > 0$. In fact, otherwise, we can replace with $S = S^* \cup B$ with a smaller set $S' = S^* \cup B'$ where $B' \subset B$, $\bar{\boldsymbol{w}}_{B'} > 0$ and $f_S = f_{S'}$ until $S' = \emptyset$.

Hence, as long as $S = S^* \cup B$ for some nonempty set B, we have

$$\begin{aligned}
f_{S^*} - f_S &= \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w}^\Diamond - \boldsymbol{y}\|_2^2 - \frac{1}{2n}\|\boldsymbol{X}\bar{\boldsymbol{w}} - \boldsymbol{y}\|_2^2 \\
&= \frac{1}{2n}(\boldsymbol{X}\boldsymbol{w}^\Diamond - \boldsymbol{X}\bar{\boldsymbol{w}})^\top(\boldsymbol{X}\boldsymbol{w}^\Diamond - \boldsymbol{y} + \boldsymbol{X}\bar{\boldsymbol{w}} - \boldsymbol{y}) \\
&= \frac{1}{2n}(\boldsymbol{w}_S^\Diamond - \bar{\boldsymbol{w}}_S)^\top \boldsymbol{X}_S^\top \left[\boldsymbol{X}(\boldsymbol{w}^\Diamond - \boldsymbol{w}^*) - \boldsymbol{\xi} + \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - \boldsymbol{\xi}\right] \\
&= \frac{1}{2n}(\boldsymbol{w}_{S^*}^\Diamond - \bar{\boldsymbol{w}}_{S^*})^\top \boldsymbol{X}_{S^*}^\top \left[\boldsymbol{X}(\boldsymbol{w}^\Diamond - \boldsymbol{w}^*) - \boldsymbol{\xi} + \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - \boldsymbol{\xi}\right] \\
&\quad + \frac{1}{2n}(\boldsymbol{w}_B^\Diamond - \bar{\boldsymbol{w}}_B)^\top \boldsymbol{X}_B^\top \left[\boldsymbol{X}(\boldsymbol{w}^\Diamond - \boldsymbol{w}^*) - \boldsymbol{\xi} + \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - \boldsymbol{\xi}\right].
\end{aligned}$$

In the following, we will bound these two terms respectively. The complementary slackness implies that $\bar{\boldsymbol{\beta}}_S = \boldsymbol{0}$, $\boldsymbol{\beta}_{S^*}^\Diamond = \boldsymbol{0}$. Therefore, by the stationary condition, we have

$$\begin{aligned}
&n^{-1}\boldsymbol{X}_{S^*}^\top \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - n^{-1}\boldsymbol{X}_{S^*}^\top \boldsymbol{\xi} = \bar{\alpha}\boldsymbol{1}, \\
&n^{-1}\boldsymbol{X}_B^\top \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - n^{-1}\boldsymbol{X}_B^\top \boldsymbol{\xi} = \bar{\alpha}\boldsymbol{1}, \\
&n^{-1}\boldsymbol{X}_{S^*}^\top \boldsymbol{X}(\boldsymbol{w}^\Diamond - \boldsymbol{w}^*) - n^{-1}\boldsymbol{X}_{S^*}^\top \boldsymbol{\xi} = \alpha^\Diamond\boldsymbol{1}.
\end{aligned}$$

As claimed in the Lemma 19, we have

$$|\bar{\alpha}| \lesssim \sqrt{\frac{\sigma^2 \log p}{n}}, |\alpha^\Diamond| \lesssim \sqrt{\frac{\sigma^2 \log p}{n}}.$$

Hence, on one hand, we can bound the first term as follows

$$\begin{aligned}
&\left|\frac{1}{2n}(\boldsymbol{w}^{\Diamond}_{S^*} - \bar{\boldsymbol{w}}_{S^*})^\top \boldsymbol{X}^\top_{S^*}\left[\boldsymbol{X}(\boldsymbol{w}^{\Diamond} - \boldsymbol{w}^*) - \boldsymbol{\xi} + \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - \boldsymbol{\xi}\right]\right| \\
\le &\left|\frac{\alpha^{\Diamond}}{2}(\boldsymbol{w}^{\Diamond}_{S^*} - \bar{\boldsymbol{w}}_{S^*})^\top \mathbf{1}\right| + \left|\frac{\bar{\alpha}}{2}(\boldsymbol{w}^{\Diamond}_{S^*} - \bar{\boldsymbol{w}}_{S^*})^\top \mathbf{1}\right| \\
\lesssim &|\alpha^{\Diamond}|\|\bar{\boldsymbol{w}}_B\|_1 + |\bar{\alpha}|\|\bar{\boldsymbol{w}}_B\|_1 \\
\lesssim &\sqrt{\frac{\sigma^2 \log p}{n}} \cdot |B|\sqrt{\frac{\sigma^2 \log p}{n}} \\
\lesssim &|B|\frac{\sigma^2 \log p}{n}.
\end{aligned}$$

On the other hand, the second term is bounded as follows

$$\begin{aligned}
&\left|\frac{1}{2n}(\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B})^\top \boldsymbol{X}^\top_{B}\left[\boldsymbol{X}(\boldsymbol{w}^{\Diamond} - \boldsymbol{w}^*) - \boldsymbol{\xi} + \boldsymbol{X}(\bar{\boldsymbol{w}} - \boldsymbol{w}^*) - \boldsymbol{\xi}\right]\right| \\
\le &\left|\frac{1}{2n}(\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B})^\top \boldsymbol{X}^\top_{B}\boldsymbol{X}_{S^*}(\boldsymbol{w}^{\Diamond}_{S^*} - \boldsymbol{w}^*_{S^*})\right| + \left|\frac{1}{2n}(\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B})^\top \boldsymbol{X}^\top_{B}\boldsymbol{\xi}\right| + \left|\frac{\bar{\alpha}}{2}(\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B})^\top \mathbf{1}\right| \\
\lesssim &\|\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B}\|_1\|n^{-1}\boldsymbol{X}^\top_B\boldsymbol{X}_{S^*}(\boldsymbol{w}^{\Diamond}_{S^*} - \boldsymbol{w}^*_{S^*})\|_\infty + \|\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B}\|_1\|n^{-1}\boldsymbol{X}^\top_B\boldsymbol{\xi}\|_\infty + |\bar{\alpha}|\|\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B}\|_1 \\
\lesssim &|B|\frac{\sigma^2 \log p}{n}
\end{aligned}$$

where we use the facts that

$$\|\boldsymbol{w}^{\Diamond}_{B} - \bar{\boldsymbol{w}}_{B}\|_1 \le |B|\left[\|\boldsymbol{w}^{\Diamond}_{B} - \boldsymbol{w}^*_{B}\|_\infty + \|\bar{\boldsymbol{w}}_{B} - \boldsymbol{w}^*_{B}\|_\infty\right] \lesssim |B|\sqrt{\frac{\sigma^2 \log p}{n}}$$

and

$$\begin{aligned}
\|n^{-1}\boldsymbol{X}^\top_B\boldsymbol{X}_{S^*}(\boldsymbol{w}^{\Diamond}_{S^*} - \boldsymbol{w}^*_{S^*})\|_\infty &= \max_{j\in B}\left|n^{-1}X_j^\top\boldsymbol{X}_{S^*}(\boldsymbol{w}^{\Diamond}_{S^*} - \boldsymbol{w}^*_{S^*})\right| \\
&\le \max_{j\in B}\|n^{-1}X_j^\top\boldsymbol{X}_{S^*}\|_\infty\|(\boldsymbol{w}^{\Diamond}_{S^*} - \boldsymbol{w}^*_{S^*})\|_1 \\
&\le \frac{c}{s^*}\cdot s^*\sqrt{\frac{\sigma^2 \log p}{n}} \\
&\lesssim \sqrt{\frac{\sigma^2 \log p}{n}}.
\end{aligned}$$

Hence, we prove the first part that $f_{S^*} - f_S \lesssim |B|\frac{\sigma^2 \log p}{n}$.
Part (II): $S^* = S \cup B$.

We lower bound $f(\bar{\boldsymbol{w}}) - f(\boldsymbol{w}^\Diamond) := f_S - f_{S^*}$ as follows

$$\begin{aligned}
f_S - f_{S^*} &= \frac{1}{2n}\|\boldsymbol{X}\bar{\boldsymbol{w}} - \boldsymbol{y}\|_2^2 - \frac{1}{2n}\|\boldsymbol{X}\boldsymbol{w}^\Diamond - \boldsymbol{y}\|_2^2 \\
&= \frac{1}{2n}\|\boldsymbol{X}_{S^*}(\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*})\|_2^2 - n^{-1}\boldsymbol{\xi}^\top \boldsymbol{X}_{S^*}(\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}) \\
&\quad - \frac{1}{2n}\|\boldsymbol{X}_{S^*}(\boldsymbol{w}^\Diamond_{S^*} - \boldsymbol{w}^*_{S^*})\|_2^2 + n^{-1}\boldsymbol{\xi}^\top \boldsymbol{X}_{S^*}(\boldsymbol{w}^\Diamond_{S^*} - \boldsymbol{w}^*_{S^*}) \\
&\geq c\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_2^2 - \|n^{-1}\boldsymbol{X}_{S^*}^\top\boldsymbol{\xi}\|_2\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_2 \\
&\quad - C\|\boldsymbol{w}^\Diamond_{S^*} - \boldsymbol{w}^*_{S^*}\|_2^2 - \|n^{-1}\boldsymbol{X}_{S^*}^\top\boldsymbol{\xi}\|_2\|\boldsymbol{w}^\Diamond_{S^*} - \boldsymbol{w}^*_{S^*}\|_2 \\
&\geq \sqrt{|B|}b_*\left(c\sqrt{|B|}b_* - \sqrt{\frac{s^*\sigma^2\log p}{n}}\right) - C\cdot s^*\frac{\sigma^2\log p}{n} \\
&\gtrsim |B|b_*^2.
\end{aligned}$$

In the first inequality, we use the implication of condition (C5) that $n^{-1}\boldsymbol{X}_{S^*}^\top\boldsymbol{X}_{S^*}$ has bounded eigenvalues. In the second inequality, we use the fact that $\|\bar{\boldsymbol{w}}_{S^*} - \boldsymbol{w}^*_{S^*}\|_2 \geq \sqrt{|B|}b_*$, $\|n^{-1}\boldsymbol{X}_{S^*}^\top\boldsymbol{\xi}\|_2 \leq \sqrt{\frac{s^*\sigma^2\log p}{n}}$ and $\|\boldsymbol{w}^\Diamond_{S^*} - \boldsymbol{w}^*_{S^*}\|_2 \leq \sqrt{\frac{s^*\sigma^2\log p}{n}}$. ■

References

Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. *Advances in Neural Information Processing Systems*, 23, 2010.

Hirotogu Akaike. Information theory and an extension of the maximum likelihood principle. In *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer, 1998.

Amir Beck. *First-order methods in optimization.* SIAM, 2017.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

Konstantinos Benidis, Yiyong Feng, and Daniel P Palomar. Sparse portfolios for high-dimensional financial index tracking. *IEEE Transactions on Signal Processing*, 66(1): 155–170, 2017.

Peter J Bickel, Ya'acov Ritov, and Alexandre B Tsybakov. Simultaneous analysis of lasso and dantzig selector. *Annals of Statistics*, pages 1705–1732, 2009.

Thomas Blumensath and Mike E Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

Peter Bühlmann and Sara van de Geer. Statistics for high-dimensional data: Methods, theory and applications. *Springer Series in Statistics*, 2011.

Laurent Condat. Fast projection onto the simplex and the ℓ_1 ball. *Mathematical Programming*, 158(1):575–585, 2016.

Steven Diamond and Stephen Boyd. CVXPY: A python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Jin-Hong Du, Yifeng Guo, and Xueqin Wang. High-dimensional portfolio selection with cardinality constraints. *Journal of the American Statistical Association*, pages 1–13, 2022.

John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, pages 272–279. PMLR, 2008.

Jianqing Fan, Yongyi Guo, and Kaizheng Wang. Communication-efficient accurate statistical estimation. *Journal of the American Statistical Association*, 118(542):1000–1010, 2023.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, pages 427–435. PMLR, 2013.

Nirmal Keshava. A survey of spectral unmixing algorithms. *Lincoln Laboratory Journal*, 14(1):55–78, 2003.

Anastasios Kyrillidis, Stephen Becker, Volkan Cevher, and Christoph Koch. Sparse projections onto the simplex. In *Proceedings of the 30th International Conference on Machine Learning*, pages 235–243. PMLR, 2013.

Ping Li, Syama Sundar Rangapuram, and Martin Slawski. Methods for sparse and low-rank recovery under simplex constraints. *Statistica Sinica*, 30(2):557–577, 2020.

Qiuwei Li, Daniel McKenzie, and Wotao Yin. From the simplex to the sphere: faster constrained optimization using the hadamard parametrization. *Information and Inference: A Journal of the IMA*, 12(3):1898–1937, 2023.

Steffen Limmer and Sławomir Stańczak. A neural architecture for bayesian compressive sensing over the simplex via laplace techniques. *IEEE Transactions on Signal Processing*, 66(22):6002–6015, 2018.

Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 70(1):53–71, 2008.

Nicolai Meinshausen. Sign-constrained least squares estimation for high-dimensional regression. *Electronic Journal of Statistics*, 7:1607 – 1631, 2013.

Yu Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

Guillaume Perez, Michel Barlaud, Lionel Fillatre, and Jean-Charles Régin. A filtered bucket-clustering method for projection onto the simplex and the ℓ_1 ball. *Mathematical Programming*, 182(1-2):445–464, 2020.

Mert Pilanci, Laurent Ghaoui, and Venkat Chandrasekaran. Recovery of sparse probability measures via convex programming. *Advances in Neural Information Processing Systems*, 25, 2012.

Gideon Schwarz. Estimating the dimension of a model. *Annals of Statistics*, pages 461–464, 1978.

Martin Slawski and Matthias Hein. Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization. *Electronic Journal of Statistics*, 7:3004 – 3056, 2013.

B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd. OSQP: an operator splitting solver for quadratic programs. *Mathematical Programming Computation*, 12(4):637–672, 2020.

Borui Tang, Jin Zhu, Junxian Zhu, Xueqin Wang, and Heping Zhang. A consistent and scalable algorithm for best subset selection in single index models. *arXiv preprint arXiv:2309.06230*, 2023.

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

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(1):91–108, 2005.

Sara A van de Geer and Peter Bühlmann. On the conditions used to prove oracle results for the lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009.

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Lan Wang, Yongdai Kim, and Runze Li. Calibrating non-convex penalized regression in ultra-high dimension. *Annals of Statistics*, 41(5):2505, 2013.

Weiran Wang and Miguel A Carreira-Perpinán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *arXiv preprint arXiv:1309.1541*, 2013.

Zezhi Wang, Junxian Zhu, Xueqin Wang, Jin Zhu, Huiyang Pen, Peng Chen, Anran Wang, and Xiaoke Zhang. skscope: Fast sparsity-constrained optimization in Python. *Journal of Machine Learning Research*, 25(290):1–9, 2024.

John Wright and Yi Ma. *High-dimensional data analysis with low-dimensional models: Principles, computation, and applications.* Cambridge University Press, 2022.

Guiyun Xiao and Zheng-Jian Bai. A geometric proximal gradient method for sparse least squares regression with probabilistic simplex constraint. *Journal of Scientific Computing*, 92(1):22, 2022.

Yanhang Zhang, Junxian Zhu, Jin Zhu, and Xueqin Wang. A splicing approach to best subset of groups selection. *INFORMS Journal on Computing*, 35(1):104–119, 2023.

Yu Zheng, Timothy M Hospedales, and Yongxin Yang. Diversity and sparsity: A new perspective on index tracking. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1768–1772. IEEE, 2020.

Jin Zhu, Xueqin Wang, Liyuan Hu, Junhao Huang, Kangkang Jiang, Yanhang Zhang, Shiyun Lin, and Junxian Zhu. abess: a fast best-subset selection library in Python and R. *Journal of Machine Learning Research*, 23(202):1–7, 2022.

Junxian Zhu, Canhong Wen, Jin Zhu, Heping Zhang, and Xueqin Wang. A polynomial algorithm for best-subset selection problem. *Proceedings of the National Academy of Sciences*, 117(52):33117–33123, 2020.