Swapping Variables for High-Dimensional Sparse Regression from Correlated Measurements

Divyanshu Vats and Richard Baraniuk
Rice University
{dvats,richb}@rice.edu

Abstract

We consider the high-dimensional sparse linear regression problem of accurately estimating a sparse vector using a small number of linear measurements that are contaminated by noise. It is well known that standard computationally tractable sparse regression algorithms, such as the Lasso, OMP, and their various extensions, perform poorly when the measurement matrix contains highly correlated columns. We develop a simple greedy algorithm, called SWAP, that iteratively *swaps* variables until a desired loss function cannot be decreased any further. SWAP is surprisingly effective in handling measurement matrices with high correlations. In particular, we prove that (i) SWAP outputs the true support, the location of the non-zero entries in the sparse vector, when initialized with the true support, and (ii) SWAP outputs the true support under a relatively mild condition on the measurement matrix when initialized with a support other than the true support. These theoretical results motivate the use of SWAP as a wrapper around various sparse regression algorithms for improved performance. We empirically show the advantages of using SWAP in sparse regression problems by comparing SWAP to several state-of-the-art sparse regression algorithms.

1. Introduction

An important problem that arises in many applications is that of recovering a high-dimensional sparse (or approximately sparse) vector given a small number of linear measurements. Depending on the problem of interest, the unknown sparse vector can encode relationships between genes (Segal et al., 2003), power line failures in massive power grid networks (Zhu and Giannakis, 2012), sparse representations of signals (Candès et al., 2006; Duarte et al., 2008), or edges in a graphical model (Meinshausen and Bühlmann, 2006; Ravikumar et al., 2010), to name just a few applications.

The simplest, but still very useful, setting is when the observations can be approximated by a *sparse* linear combination of the columns in a measurement matrix X weighted by the non-zero entries of the unknown sparse vector. Sparse regression algorithms can be used to estimate the sparse vector, and subsequently the location of the non-zero entries. In this paper, we study the sparse regression problem to accurately estimate the sparse vector in settings where current state-of-the-art methods fail. One of the key reasons why current methods fail is because of high correlations among the columns of the measurement matrix X. For example, if there exists a column, say X_i, that is nearly linearly dependent on the columns indexed by the locations of the non-zero entries, some sparse regression algorithms may falsely select X_i. Thus, the main problem we study in this paper is to perform sparse regression when the measurement matrix contains correlated columns.

There are several examples where it is natural for X to contain correlated columns. In signal processing, certain signals may admit a sparse representation in a basis, where the basis elements, which correspond to the columns of X, can be significantly correlated to each other (Elad and Aharon, 2006). Such sparse representations are useful in signal processing tasks including denoising and

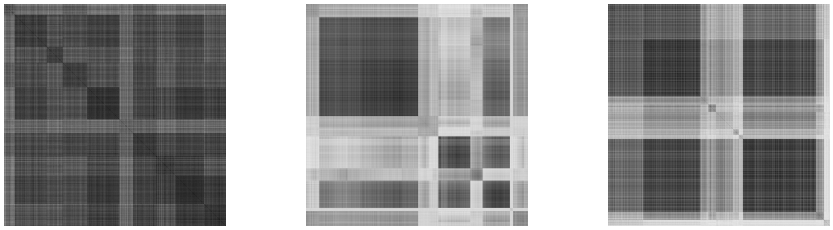

Figure 1: Pairwise correlations in gene expression data. The left figure corresponds to gene expressions from patients with small round blue cell (SRBC) tumor. The middle figure corresponds to gene expressions from patients with prostate cancer. The right figure corresponds to gene expressions from patients with leukemia. A higher intensity pixel values corresponds to the correlation being close to one. Subsequently, a lower intensity pixel corresponds to the correlations being close to zero.

compression (Elad, 2010). In functional magnetic resonance imaging (fMRI) data, measurements from neighboring voxels can be significantly correlated to each other (Varoquaux et al., 2012). This may lead to inaccurate understanding of the connectivity between regions of the brain. In gene expression data analysis, expression values of genes that are in the same pathway may be significantly correlated to each other (Segal et al., 2003). This may to lead to inaccurate identification of genes that may be most relevant to understanding diseases. For example, Figure 1 shows the pairwise correlations in three popular gene expression datasets (see caption in the figure for details about the data). Each pixel in the image is the inner product of one gene expression with another gene expression. The higher pixel intensities correspond to genes that have higher pairwise correlations.

1.1 Main Contributions

We analyze a simple method for sparse regression, called SWAP, that iteratively perturbs an initial estimate of the support, i.e., the location of the non-zero entries in the sparse vector, by swapping variables to output a final estimate of the support. The sparse vector is subsequently estimated using least-squares restricted to the estimated support. The main idea behind SWAP is that if swapping one variable from an estimated support with another variable not in the estimated support leads to a lower desired loss, then it is beneficial to perturb the estimated support to minimize the loss function. In this paper, we analyze SWAP under the least-squares loss function and prove the following two properties:

(P1) When initialized by the true support, or a support that misses one active variable (a variable in the true support), SWAP outputs the true support under very minimal conditions.

(P2) When initialized by a support that misses more than one active variable, under appropriate conditions that depend on certain correlations between among the columns of X, SWAP outputs the true support.

Property (P1) implies that SWAP can be used as a wrapper around any sparse regression algorithm without sacrificing for performance. This property alone shows that SWAP can potentially identify the true support when the correlations among the columns of X do not allow for accurate support recovery. Property (P2) identifies an explicit condition on the correlation between the active columns of X and the nonactive columns of X that are sufficient for accurate support recovery. The particular condition highlights the role of the initial support, wherein, a sparse regression algorithm that can identify more active variables, can potentially tolerate higher correlations among the columns of X. We demonstrate the empirical performance of SWAP on synthetic and gene expression data to show the advantages of using SWAP in practice. In particular, we use several state-of-the-art sparse regression algorithms to initialize SWAP. For every initialization, we show that SWAP leads to improved support recovery performance.

1.2 Related Work

Several methods have been proposed in the literature for sparse regression. The theoretical properties of these methods either depend on the irrepresentability condition (Zhao and Yu, 2006; Tropp and Gilbert, 2007; Meinshausen and Bühlmann, 2006; Wainwright, 2009a) and/or various forms of the restricted eigenvalue (RE) conditions (Meinshausen and Yu, 2009; Bickel et al., 2009). See Bühlmann and Van De Geer (2011) for a comprehensive review of such methods and the related conditions. To our knowledge, among the current methods in the literature, the least restrictive for simultaneous estimation of the sparse vector and its support are multi-stage methods, see Wasserman and Roeder (2009); Meinshausen and Yu (2009); van de Geer et al. (2011); Javanmard and Montanari (2013) for some examples. An example of a multi-stage method is thresholded Lasso (TLasso), which first applies Lasso, a popular sparse regression algorithm proposed in Tibshirani (1996), and then thresholds the absolute value of the estimated non-zero entries to output a final support. Theoretically, TLasso requires an RE based condition for accurate support recovery that is stronger than the RE based condition required by SWAP. Moreover, TLasso, and other multi-stage methods, typically require more computations since they require that computationally intensive model selection methods, such as cross-validation, be applied more than once to estimate a final regression vector.

When the columns in X are highly correlated to each other, exact support recovery may not be feasible, even when using SWAP. In such cases, it is instead desirable to obtain a superset of the true support so that the superset is as small as possible. Several methods have been proposed in the literature for estimating such a superset; see Zou and Hastie (2005); She (2010); Grave et al. (2011); Huang et al. (2011); Bühlmann et al. (2013) for examples. The main idea in these methods is to select all the highly correlated variables, even if only one of these variables is actually active. Just like SWAP improves the performance of standard sparse regression algorithms for exact support recovery, we believe that suitable modifications of SWAP can improve the various superset estimation methods to deal with highly correlated measurements.

The method SWAP itself is not new, and can be classified as a *genetic algorithm* for solving a combinatorial optimization problem (Melanie, 1999). In prior work,Fannjiang and Liao (2012) empirically show the superior performance of a slightly different version of SWAP for handing correlated measurements. However, no theory was given in Fannjiang and Liao (2012) to understand its superior performance. Our main contribution in this paper is to shed light into the performance guarantees of SWAP, and thereby show the advantages of using SWAP for sparse regression with correlated measurements. Part of the work in this paper appeared in Vats and Baraniuk (2013).

2. Problem Formulation

In this Section, we formulate the sparse regression problem and introduce relevant notations and assumptions that will be used throughout the paper. We assume that $y \in \mathbb{R}^n$, referred to as the observations, and $X \in \mathbb{R}^{n \times p}$, referred to as the measurement matrix, are known and related to each other by the linear model

$$y = X\beta^* + w, \tag{1}$$

where $\beta^* \in \mathbb{R}^p$ is the *unknown sparse vector* that we seek to estimate. Unless mentioned otherwise, we assume the following throughout this paper:

(A1) The matrix X is fixed with normalized columns, i.e., $\|X_i\|_2^2/n = 1$ for all $i \in [p]$, where $[p] = \{1, 2, \ldots, p\}$. In practice, normalization can easily be done by scaling X and β^* accordingly.

(A2) The entries of w are i.i.d. zero-mean sub-Gaussian random variables with parameter σ so that $\mathbb{E}[\exp(tw_i)] \leq \exp(t^2\sigma^2/2)$. The sub-Gaussian condition on w is common in the literature and allows for a wide class of noise models, including Gaussian, symmetric Bernoulli, and bounded random variables (Vershynin, 2010).

(A3) The vector β^* is k-sparse with the location of the non-zero entries given by the set S^*. It is common to refer to S^* as the *support* of β^* and we adopt this notation throughout the paper.

(A4) The number of observations n is greater than or equal to k, i.e., $n \geq k$. As we shall see later, this assumption is important to compute the estimates of β^*.

(A5) The number of observations n, the number of variables p, and the sparsity level k are all allowed to grow to infinity. In the literature, this is referred to as the *high-dimensional framework*.

For any set S, we associate a loss function, $\mathcal{L}(S; y, X)$, which represents the cost associated with estimating S^* by the set S. An appropriate loss function for the linear problem in (1) is the least-squares loss, which is given by

$$\mathcal{L}(S; y, X) := \min_{\alpha \in \mathbb{R}^{|S|}} \|y - X_S\alpha\|_2^2 = \left\|\Pi^{\perp}[S]y\right\|_2^2, \tag{2}$$

where X_S refers to an $n \times |S|$ matrix that only includes the columns indexed by S and $\Pi^{\perp}[S] = I - \Pi[S] = I - X_S(X_S^T X_S)^{-1}X_S^T$ is the orthogonal projection onto the kernel of the matrix X_S. Assumption (A4) is required so that the inverse of $X_S^T X_S$ exists. Throughout this paper, we mainly study the problem of estimating S^*, since, once S^* has been estimated, an estimate of β^* can be easily computed by solving a constrained least-squares problem. The subsequent error in estimating β^* is given by following Proposition.

Proposition 1 *Let $\widehat{S}$ be an estimate of S^* such that $\mathbb{P}(\widehat{S} \neq S^*) \leq e^{-f(n,p,k)}$. For any $\tau \geq 1$, the constrained least-squares estimator, $\widehat{\beta} = (X_{\widehat{S}}^T X_{\widehat{S}})^{-1} X_{\widehat{S}}^T y$, satisfies the following bound with probability at least $1 - e^{-\tau - f(n,p,k)}$:*

$$\|\widehat{\beta} - \widehat{\beta}^*\|_2^2 \leq ck\sigma^2\tau/(n\rho_k^2), \tag{3}$$

where c is a universal positive constant and ρ_k is the minimum eigenvalue of $X_{S^*}^T X_{S^*}/n$.

Algorithm 1: SWAP(y, X, S)

Inputs: Measurements y, design matrix X, and initial support S.

1 Let $r = 1$, $S^{(1)} = S$, and $L^{(1)} = \mathcal{L}(S^{(1)}; y, X)$

2 Swap $i \in S^{(r)}$ with $i' \in (S^{(r)})^c$ and compute the loss $L^{(r)}_{i,i'} = L(\{S^{(r)} \backslash i\} \cup i'; y, X)$.

3 **if** $\min_{i,i'} \mathcal{L}^{(r)}_{i,i'} < L^{(r)}$ **then**

4 $\quad \{\widehat{i}, \widehat{i'}\} = \text{argmin}_{i,i'} \mathcal{L}^{(r)}_{i,i'}$ (In case of a tie, choose a pair arbitrarily)

5 $\quad$ Let $S^{(r+1)} = \{S^{(r)} \backslash \widehat{i}\} \cup \widehat{i'}$ and $L^{(r+1)}$ be the corresponding loss.

6 $\quad$ Let $r = r + 1$ and repeat steps 2-4.

else

7 $\quad$ Return $\widehat{S} = S^{(r)}$.

Choosing $\tau = \log n$, Proposition 1 shows that if the support S^* can be estimated accurately with high probability, then the ℓ_2-error in estimating β^* can be upper bounded, with high probability, by $ck\sigma^2 \log n/(n\rho_k^2)$. Clearly, this bound converges to 0 as long as $n > ck\sigma^2 \log n/\rho_k^2$. The proof of Proposition 1 follows from standard analysis of sub-Gaussian random variables. We note that if the estimate $\widehat{S}$ is a superset of S^*, then the bound in (3) holds with k replaced by $|\widehat{S}|$.

Having established a bound for the estimation error when given the true support, we now solely focus on the problem of estimating the true support S^*. A classical method is to seek a support S that minimizes the loss $\mathcal{L}(S; y, X)$. Although this method, in general, may require a search over an exponential number of possible supports, the main goal in this paper is to design an algorithm that can search the space of all possible supports in a computationally tractable manner. Furthermore, we are interested in establishing conditions under which an algorithm can accurately estimate S^*, and as a result β^*, under much broader conditions on the measurement matrix X than those imposed by current state-of-the-art methods.

The rest of the paper is organized as follows. Section 3 presents the SWAP algorithm. Section 4 analyzes the performance of an exhaustive search decoder to understand the performance guarantees of sparse regression when given no restrictions on the computational complexity. Section 5 analyzes the SWAP algorithms and proves conditions under which SWAP leads to accurate support recovery. Section 6 presents numerical simulations. Section 7 concludes the paper.

3. Description of SWAP

In this section, we describe the SWAP algorithm to find a support that minimizes a desired loss function. Suppose that we are given an estimate, say $S^{(1)}$, of the true support and let $L^{(1)} = \mathcal{L}(S^{(1)}; y, X)$ be the corresponding least-squares loss (see (2)). We want to transition to another estimate $S^{(2)}$ that is closer (in terms of the number of true variables), or equal, to S^*. The SWAP algorithm transitions from $S^{(1)}$ to an appropriate $S^{(2)}$ in the following manner:

$$\textit{Swap every } i \in S^{(1)} \textit{ with } i' \in (S^{(1)})^c \textit{ and compute the loss } L^{(1)}_{i,i'} = \mathcal{L}\left(\{S^{(1)} \backslash i\} \cup i'; y, X\right).$$

If $\min_{i,i'} L^{(1)}_{i,i'} < L^{(1)}$, then there exists a support that has a lower loss than $L^{(1)}$. Subsequently, we find $\{\widehat{i}, \widehat{i'}\} = \arg\min_{i,i'} L^{(1)}_{i,i'}$ and let $S^{(2)} = \{S^{(1)} \backslash \widehat{i}\} \cup \{\widehat{i'}\}$. We repeat the above steps to find a sequence

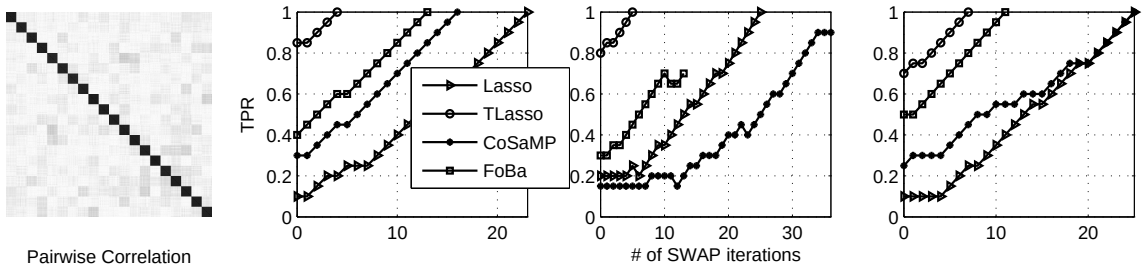

Figure 2: An example that illustrates the performance of SWAP for a matrix X with pairwise correlations given by the first figure. The second, third, and fourth figures illustrate the SWAP iterations using three realizations of the noise. The horizontal axis is the number of SWAP iterations (r in Algorithm 1) and the vertical axis is the true positive rate (TPR) of the intermediate estimates of the support.

of supports $S^{(1)}, S^{(2)}, \ldots, S^{(r)}$, where $S^{(r)}$ has the property that $\min_{i,i'} L^{(r)}_{i,i'} \geq L^{(r)}$. In other words, we stop SWAP when perturbing $S^{(r)}$ by one variable increases or does not change the resulting loss. These steps are summarized in Algorithm 1. We next make several remarks regarding SWAP.

3.1 Choosing the Initial support S.

The main input to SWAP is the initial support S. This parameter implicitly specifies the desired sparsity level of the estimated support. Recall that k is the unknown number of non-zero entries in β^*. If k is known, then SWAP can be initialized using the output of other sparse regression algorithms. For example, in Figure 2, we run SWAP using three different types of initializations on a synthetic sparse regression problem (see figure caption for details about the problem setup). The three plots in the figure show how the true positive rate (TPR) changes in the intermediate steps of the algorithm for three different realizations of the noise, where TPR is the total number of active variables in an estimate divided by the total number of active variables. For all the initializations, we see that the TPR eventually increases when SWAP stops. This signifies that SWAP estimates more active variables than the original method that SWAP is being applied in conjunction with. Furthermore, we generally observe that initializing SWAP with a support that contains more active variables is better for estimating the true support. For example, the TLasso method, short for thresholded Lasso, is able to estimate more than half of the active variables accurately. Using SWAP with TLasso leads to accurate support recovery such that SWAP terminates after only a few iterations. Our theoretical analysis in Section 5 sheds some light into this phenomenon. In particular, we show that the sufficient conditions for accurate support recovery for SWAP become weaker as the number of active variables in the initial support S increase.

When k is not known, which is the case in many applications, SWAP can be easily used in conjunction with other algorithms to compute a solution path, i.e., a list of all possible estimates of the support over different sparsity levels. Once a solution path is obtained, model selection methods, such as cross-validation and stability selection (Meinshausen and Bühlmann, 2010), can be applied to estimate the support.

3.2 Computational Complexity

The main computational step in Algorithm 1 is Line 2, where the loss $L_{i,i'}^{(r)}$ is computed for all possible swaps (i,i'). If $s = |S^{(r)}|$, then clearly $s(p-s)$ such computations need to be done in each iteration of the algorithm. Using properties of the orthogonal projection matrix, see Lemma 11, we have that for any S,

$$\Pi[S] = \Pi[S\backslash i] + \frac{(\Pi^{\perp}[S\backslash i]X_i)(\Pi^{\perp}[S\backslash i]X_i)^T}{X_i^T\Pi^{\perp}[S\backslash i]X_i}, \quad i \in S. \tag{4}$$

To compute $L_{i,i'}^{(r)}$, we need to compute the orthogonal projection matrix $\Pi^{\perp}[\{S^{(r)}\backslash\{i\}\}\cup\{i'\}]$. Once $\Pi^{\perp}[\{S^{(r)}\backslash\{i\}\}]$ has been computed, $\Pi^{\perp}[\{S^{(r)}\backslash\{i\}\}\cup\{i'\}]$ can be easily computed for all $i' \in (S^{(r)})^c$ using the rank one update equation in (4). Thus, effectively, the computational complexity of Line 2 is roughly $O(s(p-s)I_{s-1})$, where I_{s-1} is the complexity of computing a projection matrix of rank $s-1$. Using state-of-the-art matrix inversion algorithms (Coppersmith and Winograd, 1990), $I_s = O(s^{2.4})$.

There are several ways in which the computational complexity of SWAP can be significantly improved at the expense of degrading the performance of SWAP. One straightforward way, which was used in Fannjiang and Liao (2012), is to restrict the number of swaps using the correlations of the columns of X. Another method is to first estimate a superset of the support, using methods such as in Bühlmann et al. (2013); Vats (to appear), and then restrict the swaps to only lie in the estimated superset. Since the main goal in this paper is to study the statistical properties of SWAP, we address the computational aspects of SWAP in future work.

3.3 Generalization

A natural generalization of SWAP is to swap a group of m variables with another group of m variables. The computational complexity of generalized SWAP, which we refer to as SWAP^m, is roughly $O\left(\binom{s}{m}\binom{p-s}{m}I_{s-m}\right)$, where recall that I_{s-m} is the complexity of computing a rank $s-m$ projection matrix. Clearly, for m large, SWAP^m is not tractable to use in it's naiive form. In particular, as m increases, the complexity of SWAP^m approaches the complexity of the exponential search algorithm that searches among all possible supports of size s.

3.4 Comparison to Other Algorithms

SWAP differs significantly from other greedy algorithms in the literature. When k is known, the main distinctive feature of SWAP is that *it always maintains a k-sparse estimate of the support*. Note that the same is true for the computationally intractable exhaustive search algorithm. Other competitive algorithms, such as forward-backwards (FoBa) (Zhang, 2011) or CoSaMP (Needell and Tropp, 2009), usually estimate a sparse vector with higher sparsity level and iteratively remove variables until k variables are selected. The same is true for multi-stage algorithms (Zhang, 2009; Wasserman and Roeder, 2009; Zhang, 2010; van de Geer et al., 2011). Intuitively, as we shall see in Section 5, by maintaining a support of size k, the performance of SWAP only depends on correlations among the columns of the matrix X_A, where A is of size at most $2k$ and it includes the true support. In contrast, for other sparse recovery algorithms, $|A| > 2k$.

4. Theoretical Analysis of the Exhaustive Search Decoder

Recall that we are given observations y, a measurements matrix X, and we want to estimate the set S^* such that y is a linear combination of the columns in X_{S^*}. Before presenting a detailed theoretical analysis of SWAP in Section 5, we first study the exhaustive search decoder that estimates S^* and β^* as follows:

$$\text{Exaustive Search Decoder (ESD): } \widehat{S} = \min_{S \in \Omega_k} \mathcal{L}(S; y, X) \tag{5}$$

$$\widehat{\beta} = \left(X_{\widehat{S}}' X_{\widehat{S}}\right)^{-1} X_{\widehat{S}}^T y, \tag{6}$$

where $\Omega_k = \{S : S \subset [p], |S| \leq k\}$ is the set of all possible supports of size k. Thus, the ESD method searches among *all possible* supports of size k to find the support that minimizes the loss. In the literature, the ESD method is also referred to as the ℓ_0-norm solution. Understanding the performance of ESD is important to analyze the performance of SWAP, since, if S^* does *not* minimize the loss, then SWAP will most likely not estimate S^* accurately. Before stating the theorem, we define the following two parameters:

$$\rho_{k+\ell} := \inf \left\{ \frac{\|X\theta\|_2^2}{n\|\theta\|_2^2} : \|\theta\|_0 \leq k+\ell, S^* \subseteq \text{supp}(\theta) \right\} \tag{7}$$

$$\beta_{\min} := \min_{i \in S^*} |\beta_i|. \tag{8}$$

The parameter $\rho_{k+\ell}$ is the eigenvalue of certain diagonal blocks of the matrix $X^T X/n$ of size $k+\ell$ that includes the block $X_{S^*}^T X_{S^*}/n$. The parameter $\beta_{\min}$ is the minimum absolute value of the non-zero entries in β^*. Both $\rho_{k+\ell}$ (or it's different variations) and $\beta_{\min}$ are known to be crucial in determining the performance of sparse regression algorithms.

Proposition 2 *Consider the linear model in (1) and suppose (A1)-(A5) holds. Let $\widehat{S}$ be the ESD estimate with $s = k$ and let $\rho_{2k} > 0$. If $n > \frac{4+\log(k^2(p-k))}{c^2 \beta_{\min}^2 \rho_{2k}}$, where $0 < c^2 \leq 1/(18\sigma^2)$, then $\mathbb{P}(\widehat{S} = S^*) \to 1$ as $(n, p, k) \to \infty$.*

The proof of Theorem 2 is outlined in Appendix A. The steps in the proof mirror that of the ESD analysis in Wainwright (2009b). The main difference in the proof comes from the assumption that X is fixed, as opposed to being sampled from a Gaussian distribution. The dependence on ρ_{2k} in Theorem 2 is particularly interesting, since, to our knowledge, the performance of state-of-the-art computationally tractable methods for sparse regression depend either on certain correlations among the columns of X or on the restricted eigenvalues ρ_{k+s} for some $s > k$. Since $\rho_{2k} > \rho_{k+s}$ for $s > k$, ESD leads to accurate support recovery for a much broader class of measurement matrices X. However, ESD is computationally intractable, so it is desirable to devise algorithms that are computationally tractable and have similar performance guarantees. In the next section, we analyze the SWAP algorithm and show how its performance nearly mirrors that of the ESD algorithm.

5. Theoretical Analysis of SWAP

In this section, we present the main theoretical results of the paper that identifies the conditions under which SWAP performs accurate support recovery. Section 5.1 defines important parameters that are used to state the main theoretical results in Section 5.2.

5.1 Some Important Parameters

In this section, we collect some important parameters that determine the performance of SWAP. We have already defined the minimum absolute value of the non-zero entries, $\beta_{\min}$, in (8) and the restricted eigenvalue $\rho_{k+\ell}$ in (7). Another form of the restricted eigenvalue used in our analysis is defined as follows:

$$\rho_{k,\ell} := \inf \left\{ \frac{\|X\theta\|_2^2}{n\|\theta\|_2^2} : \|\theta\|_0 \leq k, |S^* \cap \text{supp}(\theta)| \geq \ell \right\}. \tag{9}$$

The parameter $\rho_{k,d}$ defined above is the minimum eigenvalue of certain blocks of the matrix $X^T X/n$ of size k that includes the blocks $X_A^T X_A/n$, where A is a subset of S^* of size at least ℓ. It is clear that smaller values of $\rho_{k+\ell}$ or $\rho_{k,d}$ correspond to correlated columns in the matrix X.

Next, we define two parameters that characterize the correlations between the columns of the matrix X_{S^*} and the columns of the matrix $X_{(S^*)^c}$, where recall that S^* is the true support of the unknown sparse vector β^*. For a set $\Omega_{k,d}$ that contains all supports of size k with atleast $k-d$ active variables from S^*, define γ_d as

$$\gamma_d := \max_{S \in \Omega_{k,d} \setminus S^*} \min_{i \in (S^*)^c \cap S} \frac{\left\| \Sigma_{i,\bar{S}}^{S \setminus i} \left(\Sigma_{\bar{S},S}^{S \setminus i} \right)^{-1} \right\|_1^2}{\Sigma_{i,i}^{S \setminus i}}, \bar{S} = S^* \setminus S, \tag{10}$$

where $\Sigma^B = X^T \Pi^{\perp}[B] X/n$. The matrix Σ^B is the pairwise correlations between vectors that are projected onto the kernel of the matrix X_B. When $B = \emptyset$, we write $\Sigma = \Sigma^B$. Popular sparse regression algorithms, such as the Lasso and the OMP, can perform accurate support recovery when $\zeta = \max_{i \in (S^*)^c} \|\Sigma_{i,S^*} \Sigma_{S^*,S^*}^{-1}\|_2^2 < 1$. We will show in Section 3.2 that SWAP can perform accurate support recovery when $\gamma_d < 1$. Although the form of γ_d is similar to ζ, there are several key differences, which we highlight as follows:

- Since $\Omega_{k,d}$ contains all supports such that $|S^* \setminus S| = d$, it is clear that γ_d is the ℓ_1 norm of a $d \times 1$ vector, where $d \leq k$. In contrast, ζ is the ℓ_1 norm of a $k \times 1$ vector. If indeed $\zeta < 1$, i.e., accurate support recovery is possible using the Lasso, then SWAP can be initialized by the output of the Lasso. In this case, $\gamma_d = 0$ and SWAP also outputs the true support as long as S^* minimizes the loss function. We make this statement precise in Theorem 3. Thus, it is only when $\zeta \geq 1$ that the parameter γ_d plays a role in the performance of SWAP.

- The parameter ζ directly computes correlations between the columns of X. In contrast, γ_d computes correlations between the columns of X when projected onto the null space of a matrix X_B, where $|B| = d-1$.

- Notice that γ_d is computed by taking a maximum over supports in the set $\Omega_d \setminus S^*$ and a *minimum* over inactive variables in each support. The reason that the minimum appears in γ_d is because we choose to swap variables that result in the minimum loss. In contrast, ζ is computed by taking a *maximum* over all inactive variables. This minimum is what allows SWAP to tolerate higher correlations among columns of X when compared to other sparse regression algorithms.

In addition to characterizing the performance of SWAP using the parameter γ_d, we also make use of the following parameter:

$$\nu_d := \max_{S \in \Omega_{k,d} \setminus S^*} \min_i \max_{j,j'} \frac{\left\| \Sigma^{S\setminus\{i,j\}}_{i,S_j} \left(\Sigma^{S\setminus\{i,j\}}_{S_j,S_j} \right)^{-1} \right\|_1^2 + \left\| \Sigma^{S\setminus\{i,j\}}_{j',S_j} \left(\Sigma^{S\setminus\{i,j\}}_{S,S_j} \right)^{-1} \right\|_1^2}{\min\left\{ \Sigma^{S\setminus\{i,j\}}_{i,i}, \Sigma^{S\setminus\{i,j\}}_{j',j'} \right\}}, \tag{11}$$

where $i \in (S^*)^c \cap S$, $j \in S$, $j' \in S^c \cap (S^*)^c$, and $\bar{S} = \{S\setminus i\} \cup \{j\}$. We will see in Theorem 6 that in the noiseless case, i.e., $\sigma = 0$, $\nu_d < 1$ ensures that SWAP only swaps an active variable with another active variable or swaps an inactive variable with an inactive variables. This allows us to show that the sample complexity of SWAP can be at par with that of the exhaustive search decoder.

5.2 Statement of Main Results

In this Section, we state the main results that characterize the performance of SWAP. The first three Theorems that we state establish sufficient conditions under which SWAP estimates the true support S^* with high probability. Throughout this Section, we assume that SWAP is initialized with a support $S^{(1)}$ of size k and $\widehat{S}$ is the output of SWAP.

5.2.1 Initialization by the True Support S^*

We first show that SWAP does not spoil the results of other algorithms.

Theorem 3 *Consider the linear model in (1) and suppose (A1)-(A5) holds. Let the input $S^{(1)}$ to* SWAP *be such that $|S^{(1)}| = k$ and $|S^*\setminus S^{(1)}| \leq 1$. If*

$$n > \frac{4 + \log(k^2(p-k))}{c^2 \beta_{\min}^2 \rho_{2k}/2}, \tag{12}$$

where $0 < c^2 \leq 1/(18\sigma^2)$, then $\mathrm{P}(\widehat{S} = S^*) \to 1$ as $(n,p,k) \to \infty$.

Proof If S^* minimizes the loss among all supports of size k, then SWAP clearly outputs S^* when initialized with a support $S^{(1)}$ such that $|S^{(1)}| = 1$ and $|S^*\setminus S^{(1)}| = 1$. From Theorem 2, the conditions stated in the result guarantee that S^* minimizes the loss with high probability. ■

Theorem 3 states that if the input to SWAP falsely detects at most one variable, then SWAP is high-dimensional consistent when given a sufficient number of observations n. In particular, the condition on n in (12) is mainly enforced to guarantee that the true support S^* minimizes the loss function. This condition is weaker than the sufficient conditions required for computationally tractable sparse regression algorithms. For example, the method FoBa is known to be superior to other methods such as the Lasso and the OMP Zhang (2011) show that FoBa requires $n = \Omega(\log(p)/(\rho_{k+\ell}^3 \beta_{\min}^2))$ observations for high-dimensional consistent support recovery, where the choice of ℓ, which is greater than k, depends on the correlations among the matrix X. In contrast, the condition in (12), which reduces to $n = \Omega(\log(p-k)/(\rho_{2k}\beta_{\min}^2))$, is weaker since $1/\rho_{k+\ell}^3 < 1/\rho_{2k}$ for $\ell > k$ and $p - k < p$. This shows that if a sparse regression algorithm can accurately estimate the true support, then SWAP does not introduce any false positives and also outputs the true support. Furthermore, if a sparse regression algorithm falsely detects one variable, then SWAP can potentially recover the correct support.

5.2.2 Initialization by an Arbitrary Support

We now consider the more interesting case when SWAP is initialized by a support $S^{(1)}$ that falsely detects more than one variable. In this case, SWAP will clearly need more than one iteration to recover the true support. Furthermore, to ensure that the true support can be recovered, we need to impose some additional assumptions on the measurement matrix X. The particular assumption we enforce will depend on the parameter γ_k defined in (10). As mentioned in Section 5.1, γ_k captures the correlations between the columns of X_{S^*} and the columns of $X_{(S^*)^c}$. To simplify the statement in the next Theorem, define the function $g(\delta,\rho,c)$ as

$$g(\delta,\rho,c) = (\delta-1) + 2c(\sqrt{\delta}+1/\sqrt{\rho}) + 2c^2\,. \tag{13}$$

Theorem 4 *Let the input to $S^{(1)}$ to* SWAP *be such that $|S^{(1)}| = k$ and $|S^*\backslash S^{(1)}| > 1$. If for a constant c such that $0 < c^2 < 1/(18\sigma^2)$, $g(\gamma_k, \rho_{k,1}, c\sigma) < 0$, $\log\binom{p}{k} > 4 + \log(k^2(p-k))$, and $n > \frac{2\log\binom{p}{k}}{c^2\beta_{\min}^2\rho_{2k}^2}$, then $\mathrm{P}(\widehat{S} = S^*) \to 1$ as $(n,p,k) \to \infty$.*

Theorem 4 says that if SWAP is initialized by *any support* of size k, and γ_k satisfies the condition stated in the theorem, then SWAP will output the true support when given a sufficient number of observations. It is easy to see that in the noiseless case, i.e., when $\sigma = 0$, the condition required for accurate support recovery reduces to $\gamma_k < 1$. The proof of Theorem 4, outlined in Appendix B, relies on imposing conditions on each support $S \in \Omega_k\backslash S^*$ such that that there exists a swap so that the loss can be necessarily decreased. Clearly, if such a property holds for each support, except S^*, then SWAP will output the true support since (i) there are only a finite number of possible supports, and (ii) each iteration of SWAP results in a different support. The dependence on $\binom{p}{k}$ in the expression for the number of observations n arises from applying the union bound over all supports of size k.

5.2.3 Initialization by Other Sparse Regression Algorithms

The condition in Theorem 4 is independent of the initialization $S^{(1)}$, which is why the sample complexity, i.e., the number of observations n required for consistent support recovery, scales as $\log\binom{p}{k}$. To reduce the sample complexity, we can impose additional conditions on the support $S^{(1)}$ that is used to initialize SWAP. One such condition is to assume that $S^{(1)}$ has certain optimality conditions over a subset of the variables from the true support. In particular, define the event $\mathcal{E}_{k,d}$ as

$$\mathcal{E}_{k,d} = \left\{ \mathcal{L}(S^{(1)};y,X) < \min_{S\in\bar{\Omega}_{k,d}\backslash S^{(1)}} \mathcal{L}(S;y,X) \right\}, |S^*\backslash S^{(1)}| = d\,, \tag{14}$$

where $\bar{\Omega}_{k,d}^c = \{S : |S| = k, |S^*\backslash S| \geq d\}$ contains all supports of size k that contain at most $k-d$ active variables from S^*. The event $\mathcal{E}_{k,d}$ is the set of outcomes for which the loss associated with $S^{(1)}$ is less than the loss associated with all supports that contain a smaller or equal number of active variables than $S^{(1)}$. If we assume that $\mathbb{P}(\mathcal{E}_{S^{(1)}}) = 1$, then all iterations of SWAP will lie in the set $\Omega_{k,d+1}$. Furthermore, by a simple counting argument, $|\Omega_{k,d+1}| \leq \binom{p}{d}^3$. This leads to the following theorem.

Theorem 5 *Let $S^{(1)}$ be the input to* SWAP *such that $|S^{(1)}| = k$, $|S^* \backslash S^{(1)}| = d > 1$, and $\mathbb{P}(\mathcal{E}_{S^{(1)}}) \to 1$. If for a constant c such that $0 < c^2 < 1/(18\sigma^2)$, $g(\gamma_{d-1}, \rho_{k,1}, c\sigma) < 0$, $3\log\binom{p}{d} > 4 + \log(k^2(p-k))$, and $n > \frac{6\log\binom{p}{d}}{c^2\beta_{\min}^2\rho_{2k}^2}$, then $\mathrm{P}(\widehat{S} = S^*) \to 1$ as $(n,p,k) \to \infty$.*

The proof of Theorem 5 follows easily from the proof of Theorem 4. The main consequence of Theorem 5 is that if a sparse regression algorithm can achieve consistent partial support recovery, then the conditions needed for support recovery using SWAP are weakened. Moreover, as the number of active variables in $S^{(1)}$ increases, the sufficient conditions required for accurate support recovery using SWAP become weaker since $\gamma_{d-1} \leq \gamma_k$ for $d > 1$.

5.2.4 Achieving optimal sample complexity

One drawback of the theoretical analysis presented so far is that the conditions require that the number of observations n scale as $\log\binom{p}{d}$, where $d \leq k$. The reason for the dependence is that, once we assume that $\mathbb{P}(\mathcal{E}_{S,d}) = 1$, there are $O(\binom{p}{d})$ possible number of supports that SWAP can visit. To ensure that SWAP does not make an error, we use the union bound over all these sets to bound the probability of making an error. In practice, however, once the support set S is fixed, the total number of possible supports that SWAP can visit can be much less than $O(\binom{p}{d})$. The exact number of possible supports will depend on the correlations between the columns of X. In the next Theorem, we show that under additional assumptions on X, SWAP can achieve similar sample complexity as the exhaustive searcher decoder.

Theorem 6 *Let S be the input to* SWAP *such that $|S| = k$ and $|S^* \backslash S| \leq d$. If for a constant c such that $0 < c^2 < 1/(18\sigma^2)$, $g(\nu_d, \rho_{k-1,0}/2, c\sigma) < 0$, and $n > \frac{2k + \log(k(p-k))}{c^2\beta_{\min}^2\rho_{2k}^2/4}$, then $\mathrm{P}(\widehat{S} = S^*) \to 1$ as $(n,p,k) \to \infty$.*

The proof of Theorem 6 is similar to the proof of Theorem 4 and is outlined in Appendix C. The condition $g(\nu_d, \rho_{k-1,0}/2, c\sigma) < 0$ ensures that when SWAP is initialized with an appropriate support S of size k, then the SWAP iterations only swap an active variable with an active variable or swap an inactive variable with an active variable. This allows us to upper bound the total number of possible supports that SWAP can visit from $O\left(\binom{p}{d}\right)$ to 2^k. We note that in numerical simulations, we observed that even when the SWAP iterations swapped an inactive variable with an inactive variable, SWAP performed accurate support recovery. Thus, not allowing an inactive variable to be swapped with an inactive variable is rather restrictive. We believe that that a more complex analysis, that allows a constant number of swaps of an inactive with an inactive, can further weaken the condition in Theorem 6 with the number of observations satisfying $n > \frac{c'k + \log(k(p-k))}{c^2\beta_{\min}^2\rho_{2k}^2/4}$ for a constant c' that controls the maximum number of inactive with inactive swaps.

6. Numerical Simulations

In this section, we illustrate the performance of SWAP when used in conjunction with several sparse regression algorithms. Section 6.1 presents synthetic data results and Section 6.2 presents pseudo real data results.

6.1 Synthetic Data

To illustrate the advantages of SWAP, we use the following two examples:

(E1) We sample the rows of X from a Gaussian distribution with mean zero and covariance Σ. The covariance Σ is block-diagonal with blocks of size p/k. The entries in each block $\bar{\Sigma}$ are specified as follows: $\bar{\Sigma}_{ii} = 1$ for $i \in \{1, \ldots, p/k\}$ and $\bar{\Sigma}_{ij} = a$ for $i \neq j$. This construction of the design matrix is motivated from Bühlmann et al. (2013). The true support is chosen so that each variable in the support is assigned to a different block. The non-zero entries in β^* are chosen to have magnitude one with sign randomly chosen to be either positive or negative (with equal probability). We let $\sigma = 1$, $p = 1000$, $n \in \{100, 150, 200, 250, \ldots, 500\}$, $k = 20$, and $a \in [0.6, 0.99]$.

(E2) We sample X from the same distribution as described in (E1). The only difference is that the true support is chosen so that five different blocks contain active variables and each chosen block contains *four* active variables. The rest of the parameters are also the same.

In both (E1) and (E2), as a increases, the strength of correlations between the columns increase. Furthermore, from the construction, it is clear that the restricted eigenvalue parameter for (E1) is, in general, greater than the restricted eigenvalue parameter of (E2). Thus, (E1) requires less number of observations than (E2) for accurate sparse regression.

6.1.1 Sparse Regression Algorithms

We use the following sparse regression algorithms to initialize SWAP:

- Lasso (Tibshirani, 1996)
- Thresholded Lasso (TLasso) (van de Geer et al., 2011)
- Forward-Backward (FoBa) (Zhang, 2011)
- CoSaMP (Needell and Tropp, 2009)
- Marginal Regression (MaR)
- Random

TLasso is a two-stage algorithm where the first stage applies Lasso and the second stage selects the top k largest (in magnitude) variables from the Lasso estimate. In our implementation, we applied Lasso using 5-fold cross-validation. FoBa uses a combination of a forward and a backwards algorithm. CoSaMP is an iterative greedy algorithm. MaR selects the support by choosing the largest k variables in $|X^T y|$. Random selects a *random* subset of size k. We use the notation S-TLasso to refer to the algorithm that uses TLasso as an initialization for SWAP. A similar notation follows for other algorithms. Finally, when running the sparse regression algorithms, we assume that k is known. In practice, model selection algorithms can be used to select an appropriate k. Since our main goal is to illustrate the performance of SWAP, and thereby validate our theoretical results in Section 5, we only show results for the case when k is assumed to be known. Finally, all our results are reported over 100 trials.

6.1.2 DEPENDENCE ON THE DEGREE OF CORRELATION

Figure 3 and Figure 5 plot the mean TPR for (E1) and (E2), respectively, as the parameter a increases from 0.5 to 0.99. The dashed lines correspond to a standard sparse regression algorithm, while a solid line corresponds to a SWAP based algorithm. In most cases, we see that, for the same color, the solid lines are above the dashed lines. This shows, as predicted by our theory, that SWAP is able to improve the performance of other regression algorithms. Furthermore, we observe that TLasso has the best performance among all algorithms, and S-TLasso improves the performance of TLasso. However, the computational complexity of TLasso is more than that of other algorithms since it requires two stages of model selection.

As expected, the performance of the algorithms degrade as the correlations are large. Moreover, when the correlations are extremely high, then the difference between TLasso and S-TLasso is insignficant. This suggests that for such cases, it might be more suitable to use sparse regression algorithms, such as those in Zou and Hastie (2005); She (2010); Grave et al. (2011); Huang et al. (2011); Bühlmann et al. (2013), which are designed to estimate a superset of the true support.

Figure 4 and Figure 6 show the boxplots of the difference between the TPR of a SWAP based algorithm minus the TPR of a standard algorithm. A positive difference means that SWAP is able to improve the performance of a sparse regression algorithm. For Lasso, FoBa, and CoSaMP, we see that the difference is generally positive, even for large values of a. For TLasso, we notice that the difference is generally positive for $a = 0.90$ and $a = 0.93$. For greater values of a, there does not seem to be any advantages of using SWAP. This is likely due to the correlations being extremely high so that the true support no longer minimizes the loss function.

6.1.3 NUMBER OF ITERATIONS AND DEPENDENCE ON THE NUMBER OF OBSERVATIONS

Figure 7 plots the mean number of iterations required by SWAP based algorithms as the parameter a varies from 0.55 to 0.99. As expected, the number of iterations generally increase as the correlations among the columns increase. We also observe that FoBa and TLasso require the smallest number of iterations to converge. This is primarily because both these algorithms are able to estimate a large fraction of the true support.

Figure 8 plots the mean TPR for (E1) and (E2) when $a = 0.9$ and the number of observations vary from 100 to 500. We clearly see that SWAP based algorithms perform better than standard algorithms. For simplicity, we only plot results for TLasso and FoBa.

6.2 Pseudo Real Data

In this section, we present results for the case when the measurement matrix X corresponds to gene expression data. Since we do not have any ground truth, we simulate β^* and y. For this reason, we refer to this simulation as pseudo real data. The two simulation settings are as follows:

- Leukemia: This dataset contains 5147 gene expression values from 72 patients (Golub et al., 1999). For computational reasons, we only select $p = 2000$ genes to obtain a 72×2000 measurement matrix X. We let $k = 10$, $\beta_{\min} = 4$, and $\sigma = 1$. As seen in Section 6.1, the choice of the support plays an important role in determining the performance of a sparse regression algorithm. To select the support, we cluster the columns using kmeans to identify 20 clusters. Next, we obtain the support by random selecting five clusters and then selecting exactly two variabes from each cluster,

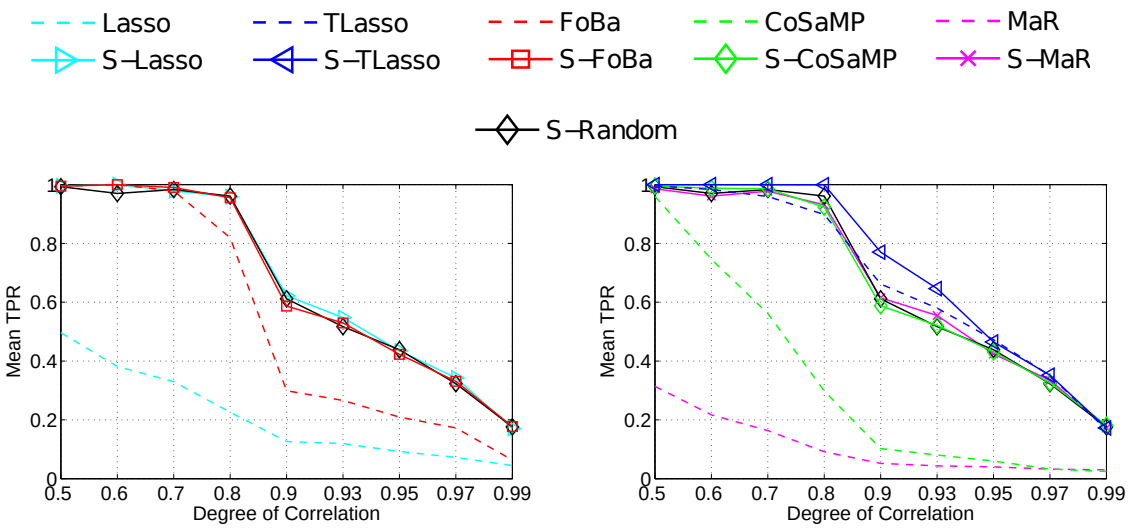

Figure 3: Results for (E1). Mean true positive rate (TPR) versus the degree of correlation (the parameter a) for several different sparse regression algorithms. The dashed lines correspond to standard sparse regression algorithms, while the solid lines with markers correspond to SWAP based regression algorithms.

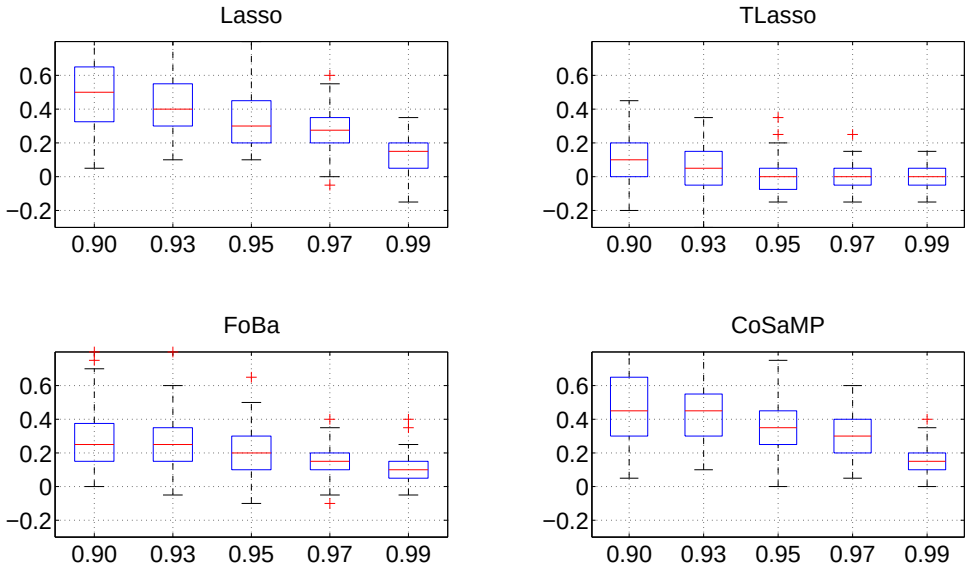

Figure 4: Results for (E1). Box plots of the TPR of SWAP based algorithms minus the TPR of a standard regression algorithm for five different values of the degree of correlation a. For example, top left is the TPR of SWAP based Lasso minus the TPR of Lasso.

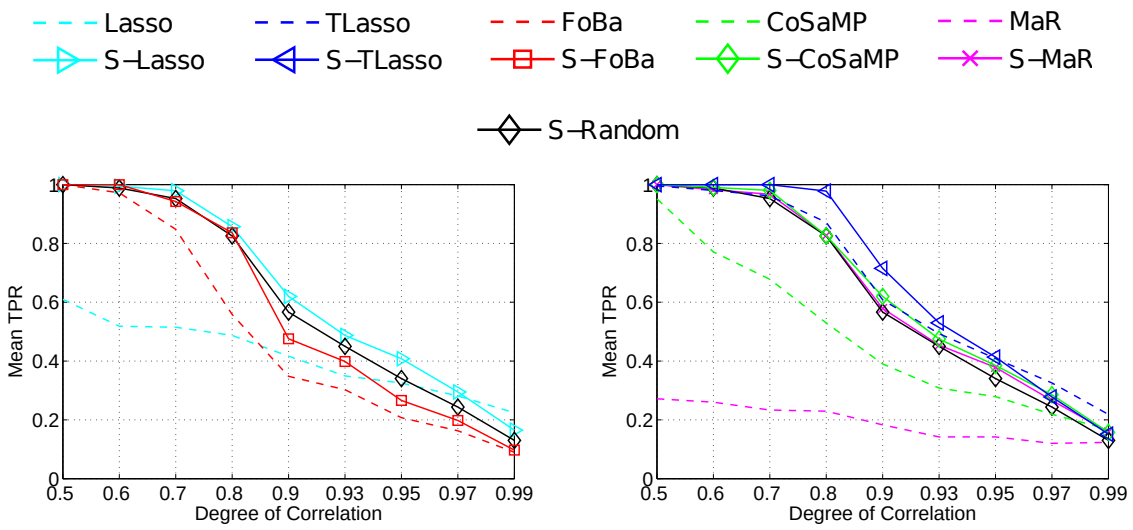

Figure 5: Results for (E2). Mean true positive rate (TPR) versus the degree of correlation (the parameter a) for several different sparse regression algorithms. The dashed lines correspond to standard sparse regression algorithms, while the solid lines with markers correspond to SWAP based regression algorithms.

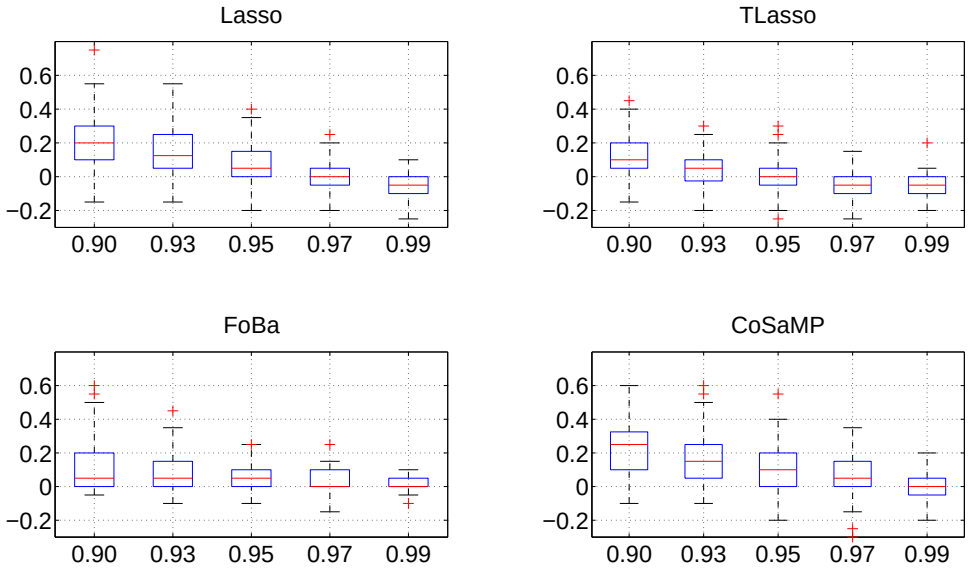

Figure 6: Results for (E2). Box plots of the TPR of SWAP based algorithms minus the TPR of a standard regression algorithm for five different values of the degree of correlation a. For example, top left is the TPR of SWAP based Lasso minus the TPR of Lasso.

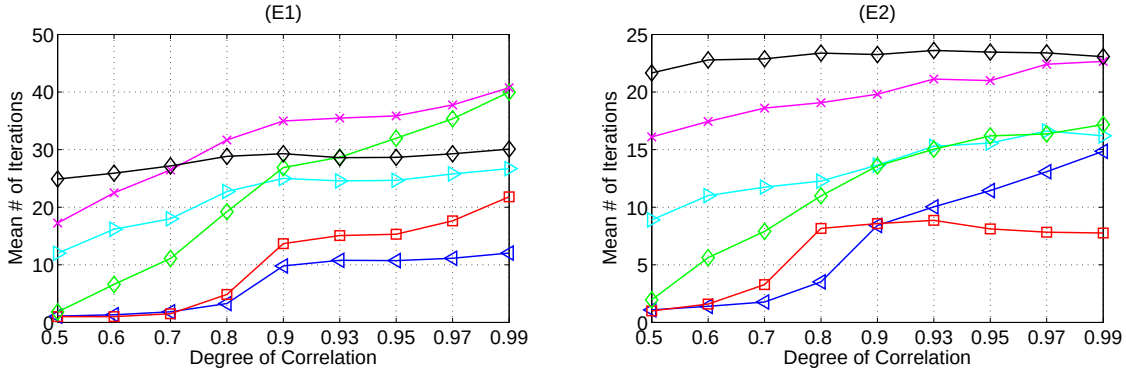

Figure 7: Mean number of iterations required by SWAP when used with different sparse regression algorithms for the synthetic examples (E1) and (E2). Recall that $p = 1000$, $n = 200$, and $k = 20$. See Figure 5 for legend.

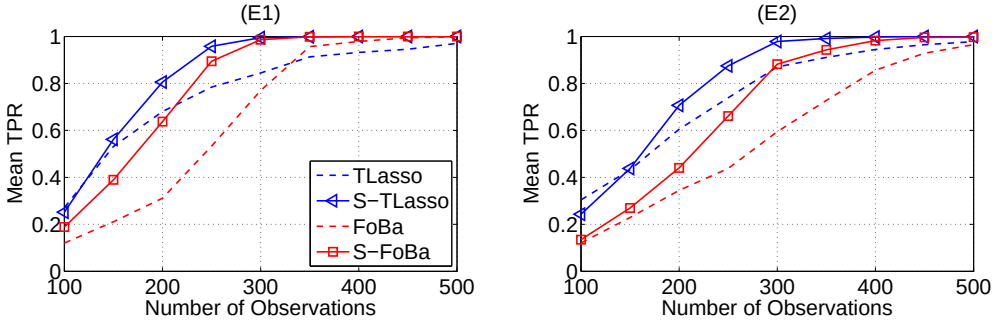

Figure 8: Mean TPR versus the number of observations n for (E1) and (E2).

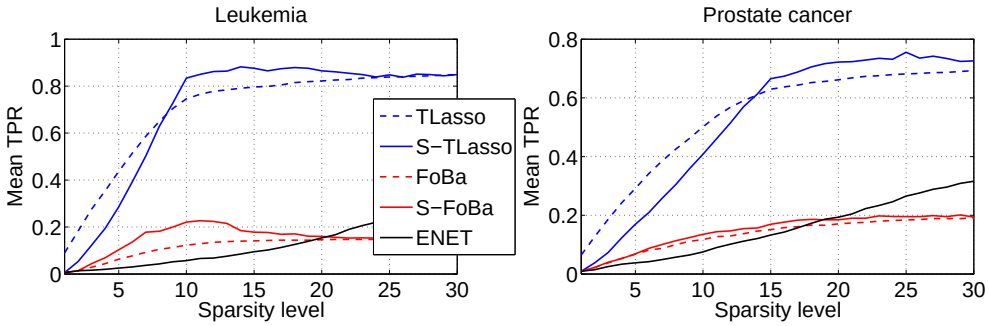

Figure 9: Mean TPR as the sparsity level of the estimated support increases for the Leukemia gene expression data (left) and the prostate cancer gene expression data (right). For the leukemia data, the true sparsity level is $k = 10$. For the prostate cancer data, the true sparsity level is $k = 15$.

- Prostate Cancer: This dataset contains 12533 gene expression values from 102 patients (Singh et al., 2002). The selection of X, σ, and $\beta_{\min}$ is the same as in the Leukemia data. The only difference is that $k = 15$ and the selected support contains three variables from one cluster (as opposed to the two chosen in the Leukemia data).

Figure 9 plots the mean TPR over 100 realizations versus the sparsity level of the estimated support for several sparse regression algorithms. In the figures, we also compare to the elastic net (ENET) algorithm (Zou and Hastie, 2005), which is known to be suitable for regression with correlated variables. Note that ENET requires two regularization parameters. In our simulations, we run ENET for a two-dimensional grid of regularization parameters and select a support for each sparsity level that results in the smallest loss. We only compare ENET to TLasso and FoBa since we know from Section 6.1 that both these algorithms are superior to other regression algorithms. From the figures, it is clear that after a certain sparsity level, the SWAP based algorithms perform better than non-SWAP based algorithms. Furthermore, we clearly see that SWAP performs significantly better than ENET.

7. Conclusion

We studied the sparse regression problem of estimating the support of a high-dimensional sparse vector when given a measurement matrix that contains correlated columns. We presented a simple algorithm, called SWAP, that iteratively swaps variables starting from an initial estimate of the support until an appropriate loss function can no longer be decreased further. We showed that SWAP is surprising effective in situations where the measurement matrix contains correlated columns. We theoretically quantified the conditions on the measurement matrix that guarantee accurate support recovery. Our theoretical results show that if SWAP is initialized with a support that contains some active variables, then SWAP can tolerate even higher correlations in the measurement matrix. Using numerical simulations on synthetic and real data, we showed how SWAP outperformed several sparse recovery algorithms.

Our work in this paper sets up a platform to study the following interesting extensions of SWAP. The first is a generalization of SWAP so that a group of variables can be swapped in a sequential manner. The second is a detailed analysis of SWAP when used with other sparse recovery algorithms. The third is an extension of SWAP to high-dimensional vectors that admit structured sparse representations.

Acknowledgement

The authors would like to thank Aswin Sankaranarayanan (Carnegie Mellon University) and Christoph Studer (Cornell University) for feedback and discussions. The work of D. Vats was partly supported by an Institute for Mathematics and Applications (IMA) Postdoctoral Fellowship.

Appendix A. Proof of Proposition 2

Recall that $\mathcal{L}(S;y,X) = \|\Pi^{\perp}[S]y\|_2^2$. Analyzing the exhaustive search decoder (ESD) for $s = k$ is equivalent to finding conditions under which the following holds:

$$||\Pi^{\perp}[S^*]y||_2^2 < \min_{S\in\Omega_k\backslash S^*} ||\Pi^{\perp}[S]y||_2^2 . \tag{15}$$

Using properties of the orthogonal projection, it is easy to see that $||\Pi^{\perp}[S^*]y||_2^2 = ||\Pi^{\perp}[S^*]w||_2^2$. Furthermore, we have

$$||\Pi^{\perp}[S]y||_2^2 = \xi^T\Pi^{\perp}[S]\xi + w^T\Pi^{\perp}[S]w + 2\xi^T\Pi^{\perp}[S]w ,$$

where $\xi = X\beta^*$. Substituting the above into (15), we have that (15) is equivalent to showing that the following holds:

$$\min_{S\in\Omega_k\backslash S^*} \Bigg[\underbrace{\xi^T\Pi^{\perp}[S]\xi + w^T(\Pi^{\perp}[S] - \Pi^{\perp}[S^*])w + 2\xi^T\Pi^{\perp}[S]w}_{\mathcal{W}(S)} \Bigg] > 0 .$$

To find conditions under which the above holds, we first lower bound $\mathcal{W}(S)$. Using properties of projection matrices, and using arguments in Wainwright (2009b), $\Pi^{\perp}[S] - \Pi^{\perp}[S^*]$ is a difference of two rank $\ell = |S^*\backslash S| = |S\backslash S^*|$ projection matrices. Using properties of sub-Gaussian random vectors in Lemma 9 and Lemma 10, we have

$$\mathrm{P}\left(|w^T(\Pi^{\perp}[S] - \Pi^{\perp}[S^*])w| > 4f_n\ell\sigma^2 \right) \leq 2e^{-\ell f_n/2} , \quad f_n \geq 1 ,$$

$$\mathrm{P}\left(|\xi^T\Pi^{\perp}[S]w| \geq \delta_n \right) \leq 2e^{-\delta_n^2/(2||\xi^T\Pi^{\perp}[S]||_2^2\sigma^2)} .$$

Using the above tail inequalities, we can write down a lower bound for $\mathcal{W}(s)$ such that

$$\mathcal{W}(s) \geq ||\Pi^{\perp}[S]\xi||_2^2 \left[1 - \frac{4\sigma^2\ell f_n}{||\Pi^{\perp}[S]\xi||_2^2} - \frac{2\sigma\delta_n}{||\Pi^{\perp}[S]\xi||_2} \right] ,$$

holds with probability at least $1-2e^{-\ell f_n/2}-2e^{-\delta_n^2/2}$. Using properties of the eigenvalues, we know that $||\Pi^{\perp}[S]\xi||_2^2 \geq \ell n \rho_{2k}\beta_{\min}^2$, where ρ_{2k} is defined in (7). Thus, we have the following lower bound:

$$\mathcal{W}(s) \geq ||\Pi^{\perp}[S]\xi||_2^2 \left[1-\frac{4\sigma^2 f_n}{n\rho_{2k}\beta_{\min}^2}-\frac{2\sigma\delta_n}{\sqrt{n\ell\rho_{2k}}\beta_{\min}}\right],$$

which holds with probability at least $1-2e^{-\ell f_n/2}-2e^{-\delta_n^2/2}$. Choosing $f_n=\delta_n^2/\ell$,

$$\mathcal{W}(s) \geq ||\Pi^{\perp}[S]\xi||_2^2 \left[1-\frac{2\sigma^2\delta_n^2}{n\ell\rho_{2k}\beta_{\min}^2}-\frac{2\sigma\delta_n}{\sqrt{n\ell\rho_{2k}}\beta_{\min}}\right],$$

with probability at least $1-4e^{-\delta_n^2/2}$. Choosing $\delta_n^2 = c^2 n\ell\rho_{2k}\beta_{\min}^2$,

$$\mathcal{W}(s) \geq ||\Pi^{\perp}[S]\xi||_2^2 \left(1-2c^2\sigma^2-2c\sigma\right).$$

with probability at least $1-4e^{-c^2 n\ell\rho_{2k}\beta_{\min}^2/2}$. Now, if c is chosen so that $(1-2(c\sigma)^2-2c\sigma)>0$, then

$$\mathrm{P}(\mathcal{W}(S)<0) \leq 4e^{-c^2 n\ell\rho_{2k}\beta_{\min}^2/2}. \tag{16}$$

Recall that we want to find conditions under which $\min_{S\in\Omega_k\setminus S^*}\mathcal{W}(S)>0$. Using standard arguments in probability theory, we have

$$\begin{aligned}
\mathrm{P}\left(\min_{S\in\Omega_k\setminus S^*}\mathcal{W}(S)>0\right) &= \mathrm{P}\left(\bigcap_{S\in\Omega_k\setminus S^*}\{\mathcal{W}(S)>0\}\right) = 1-\mathrm{P}\left(\bigcup_{S\in\Omega_k\setminus S^*}\{\mathcal{W}(S)\leq 0\}\right) \\
&\geq 1-\sum_{S\in\Omega_k\setminus S^*}\mathrm{P}(\mathcal{W}(S)\leq 0).
\end{aligned}$$

Let $N(\ell)$ be the number of supports of size k that differ from S^* by ℓ variables. From Wainwright (2009b), we have $N(\ell)=\binom{k}{\ell}\binom{p-k}{\ell}$. The summation above can now be upper bounded as follows:

$$\begin{aligned}
\sum_{S\in\Omega_k\setminus S^*}\mathrm{P}(\mathcal{W}(S)\leq 0) &\leq \sum_{\ell=1}^{k} 4N(\ell)\exp(-c^2 n\ell\rho_{2k}\beta_{\min}^2/2) \\
&\leq \sum_{\ell=1}^{k} 4\binom{k}{\ell}\binom{p-k}{\ell}\exp(-c^2 n\ell\rho_{2k}\beta_{\min}^2/2) \\
&\leq 4k\max_{\ell=1,\ldots,k}\binom{k}{\ell}\binom{p-k}{\ell}\exp(-c^2 n\ell\rho_{2k}\beta_{\min}^2/2) \\
&\overset{(a)}{\leq} 4k\max_{\ell=1,\ldots,k}\left(\frac{k(p-k)e^2}{\ell^2}\right)^{\ell}\exp(-c^2 n\ell\rho_{2k}\beta_{\min}^2/2) \\
&\leq 4k\max_{\ell=1,\ldots,k}\exp\left(\ell\log(k(p-k)e^2/\ell^2)-c^2 n\ell\rho_{2k}\beta_{\min}^2/2\right) \\
&\leq \max_{\ell=1,\ldots,k}\exp\left(\ell(\log(4k^2(p-k)e^2/\ell^2)-c^2 n\rho_{2k}\beta_{\min}^2/2)\right),
\end{aligned}$$

where (a) uses the upper bound $\binom{p}{k}=(pe/k)^k$. We want the above expression to diverge to 0 as $(n,p,k)\to\infty$. For this to happen, it is sufficient to require that

$$n > \frac{4+\log(k^2(p-k))}{c^2\beta_{\min}^2\rho_{2k}/2},$$

where 4 comes from $4 > \log(4e^2)$ and recall that c is chosen so that $1-2(c\sigma)^2-2c\sigma > 0$. It is easy to see that is sufficient to choose c such that $0 < c \leq 1/(3\sigma) \Longrightarrow 0 < c^2/2 \leq 1/(18\sigma^2)$.

Appendix B. Proof of Theorem 4

Suppose that for a set $S \in \Omega_k$, there exists a variable $i \in S$ that can be swapped with a variable $i' \in S^c$ such that the resulting support, $S^{(i,i')} = \{S\backslash i\} \cup \{i'\}$, leads to a lower loss than the loss associated with the support S. In this case, SWAP will not stop and find a suitable swapping to reduce the loss. If every set $S \in \Omega_k$, except S^*, has this property, then SWAP will only stop once it reaches S^*. Thus, a sufficient condition for SWAP to output the correct support is for the following two conditions to hold:

$$\forall\, S \in \Omega_k, \quad \min_{i \in S\cap(S^*)^c, i' \in S^c \cap S^*} \mathcal{L}(S^{(i,i')}; y, X) < \mathcal{L}(S; y, X)\,, \tag{17}$$

$$\min_{S \in \Omega_k \backslash S^*} \mathcal{L}(S; y, X) > \mathcal{L}(S^*; y, X)\,. \tag{18}$$

Equation (17) ensures that SWAP does not stop for all $S \in \Omega_k \backslash S^*$ and (18) ensures that SWAP stops once the true support has been identified. Note that (17) ensures that an inactive variable from S may be swapped with an active variable from S^c. Define the events

$$\begin{aligned}
\mathcal{E} &= \{\mathcal{L}(S; y, X) > \mathcal{L}(S^*; y, X)\}\,, \\
\mathcal{D}_S &= \{\min_{i \in S\cap(S^*)^c, i' \in S^c \cap S^*} \mathcal{L}(S^{(i,i')}; y, X) < \mathcal{L}(S; y, X)\},
\end{aligned}$$

where $S \in \Omega_k$. The probability of accurate support recovery can be lower bounded as follows:

$$\begin{aligned}
\mathbb{P}(\widehat{S} = S^*) &\geq \mathbb{P}\left(\widehat{S} = S^* | \left\{\cap_{S \in \Omega_k \backslash S^*} \mathcal{D}_S\right\} \cap \mathcal{E}\right) \mathbb{P}\left(\left\{\cap_{S \in \Omega_k \backslash S^*} \mathcal{D}_S\right\} \cap \mathcal{E}\right) \\
&= \mathbb{P}\left(\left\{\cap_{S \in \Omega_k \backslash S^*} \mathcal{D}_S\right\} \cap \mathcal{E}\right) \\
&= 1 - \mathbb{P}\left(\cup_{S \in \Omega_k \backslash S^*} \mathcal{D}_S^c\right) - \mathbb{P}(\mathcal{E}^c)\,.
\end{aligned} \tag{19}$$

Theorem 2 identifies conditions under which $\mathbb{P}(\mathcal{E}^c) \to 0$. Thus, we only need to specify conditions under which $\mathbb{P}\left(\cup_{S \in \Omega_k \backslash S^*} \mathcal{D}_S^c\right) \to 0$. To do so, we first analyze the event $\mathcal{D}_S^c$ for a fixed S. Using the definition of the least-squares loss, we have

$$\mathcal{D}_S^c = \left\{ \min_{i \in S\cap(S^*)^c, i' \in S^c \cap S^*} \|\Pi^{\perp}[S^{(i,i')}]y\|_2^2 - \|\Pi^{\perp}[S]y\|_2^2 > 0 \right\} \tag{20}$$

$$= \left\{ \min_{i \in S\cap(S^*)^c, i' \in S^c \cap S^*} \left[\xi^T \Gamma_S(i,i')\xi + w^T \Gamma_S(i,i')w + 2\xi^T \Gamma_S(i,i')w\right] > 0 \right\}, \tag{21}$$

where $\xi = X_{\bar{S}}\beta^*_{\bar{S}}$, $\bar{S} = S^*\backslash S$, $\Gamma_S(i,i') = \Pi[S] - \Pi[A_i, i']$, and $A_i = S\backslash\{i\}$. Recall that $i \in S^c \cap (S^*)^c$, which leads to the expression in (21). Note that the first term in the expression of (21) is deterministic, while the second and third terms are random (they depend on the noise w). To show that $\mathbb{P}(\mathcal{D}_S^c) \to 0$, we first upper bound $\mathbb{P}(\mathcal{D}_S^c)$ and then show that the upper bound converges to 0. Using properties of projection matrices, it is easy to see that $\Gamma_S(i,i')$ is a difference of two rank one projection matrices. Using Lemma 9 and Lemma 10, we have the following tail bounds:

$$\mathbb{P}\left(|w^T \Gamma_S(i,i')w| \geq 4 f_n^{(i,i')} \sigma^2\right) \leq 2e^{-f_n^{(i,i')}/2},\ f_n^{(i,i')} \geq 1\,, \tag{22}$$

$$\mathbb{P}\left(|\xi^T\Gamma_S(i,i')w| \geq \sigma\|\xi^T\Gamma_S(i,i')\|_2\delta_n^{(i,i')}\right) \leq 2e^{-\left(\delta_n^{(i,i')}\right)^2/2}. \tag{23}$$

To simplify notation, define the events $\mathcal{E}_1 = \left\{|w^T\Gamma_S(i,i')w| < 4f_n^{(i,i')}\sigma^2\right\}$ and $\mathcal{E}_2 = \left\{|\xi^T\Gamma_S(i,i')w| < \sigma\|\xi^T\Gamma_S(i,i')\|_2\delta_n^{(i,i')}\right\}$. We emphasize that both $\mathcal{E}_1$ and $\mathcal{E}_2$ depend on S, i, and i'. Using the total probability theorem, we have the following upper bound for $\mathbb{P}(\mathcal{D}_S^c)$:

$$\mathbb{P}(\mathcal{D}_S^c) \leq \min_{i\in S\cap(S^*)^c, i'\in S^c\cap S^*} \mathbb{P}(\overbrace{\xi^T\Gamma_S(i,i')\xi + w^T\Gamma_S(i,i')w + 2\xi^T\Gamma_S(i,i')w > 0}^{\mathcal{H}_S^{(i,i')}}) \tag{24}$$

$$\mathbb{P}(\mathcal{H}_S^{(i,i')}) \leq \mathbb{P}(\mathcal{H}_S^{(i,i')}|\mathcal{E}_1\cap\mathcal{E}_2)\mathbb{P}(\mathcal{E}_1\cap\mathcal{E}_2) + \mathbb{P}(\mathcal{H}_S^{(i,i')}|\mathcal{E}_1^c\cup\mathcal{E}_2^c)\mathbb{P}(\mathcal{E}_1^c\cup\mathcal{E}_2^c) \tag{25}$$

$$\leq \mathbb{P}(\mathcal{H}_S^{(i,i')}|\mathcal{E}_1\cap\mathcal{E}_2) + \mathbb{P}(\mathcal{E}_1^c) + \mathbb{P}(\mathcal{E}_2^c). \tag{26}$$

Next, using the definition of $\mathcal{E}_1$ and $\mathcal{E}_2$, we have

$$\mathbb{P}(\mathcal{H}_S^{(i,i')}|\mathcal{E}_1\cap\mathcal{E}_2) \leq \mathbb{1}\left(\xi^T\Gamma_S\xi + 4f_n^{(i,i')}\sigma^2 + 2\sigma\|\xi^T\Gamma_S(i,i')\|_2\delta_n^{(i,i')} > 0\right), \tag{27}$$

where $\mathbb{1}(q>0) = 1$ if $q>0$ and $\mathbb{1}(q>0) = 0$ if $q\leq 0$. Choosing $f_n^{(i,i')} = (\delta_n^{(i,i')})^2 = \delta_n^2$ (i.e., independent of i and i'), and substituting (27) and (26) into (24), we have the following upper bound for $\mathbb{P}(\mathcal{D}_S^c)$:

$$\mathbb{1}\left(\min_{i,i'}\left(\xi^T\Gamma_S(i,i')\xi + 2\delta_n^2\sigma^2 + 2\sigma\|\xi^T\Gamma_S(i,i')\|_2\delta_n\right) > 0\right) + 4e^{-\delta_n^2/2} \tag{28}$$

where $i\in S\cap(S^*)^c$, $i'\in S^c\cap S^*$, and we use the tail bounds in (22) and (23). In Section B.1, we show that

$$\min_{i\in S\cap(S^*)^c, i'\in S^c\cap S^*} \left(\xi^T\Gamma_S(i,i')\xi + 2\sigma\|\xi^T\Gamma_S(i,i')\|_2\delta_n + 2\delta_n^2\sigma^2\right)$$
$$\leq C\left[(\gamma_k - 1) + \frac{2\sigma\delta_n}{\sqrt{n}\rho_{2k}\beta_{\min}}\left(\sqrt{\gamma_k} + 1/\sqrt{\rho_{k,1}}\right) + \frac{2\delta_n^2\sigma^2}{n\rho_{2k}^2\beta_{\min}^2}\right], \tag{29}$$

where $C>0$, γ_k is defined in (10), and $\rho_{k,1}$ is defined in (9). Choosing $\delta_n = c_1\sqrt{n}\rho_{2k}\beta_{\min}$, and using the bound in (29), we further upper bound (28) as

$$\mathbb{1}\left((\gamma_k-1) + 2c_1\sigma\left(\sqrt{\gamma_k} + 1/\sqrt{\rho_{k,1}}\right) + 2(c_1\sigma)^2 > 0\right) + 4e^{-c_1^2 n\beta_{\min}^2\rho_{2k}^2/2}$$

If c_1 is chosen so that $(\gamma_k-1) + 2c_1\sigma\left(\sqrt{\gamma_k} + 1/\sqrt{\rho_{k,1}}\right) + 2(c_1\sigma)^2 \leq 0$, then we have

$$\mathbb{P}(\mathcal{D}_S^c) \leq 4e^{-c_1^2 n\beta_{\min}^2\rho_{2k}^2/2}$$

Substituting the above into (19) and using the union bound, we have

$$\mathbb{P}(\widehat{S} = S^*) \geq 1 - \binom{p}{k}4e^{-c_1^2 n\beta_{\min}^2\rho_{2k}^2/2} - \mathbb{P}(\mathcal{E}^c)$$

Using Theorem 2, we know that if $n > \frac{4+\log(k^2(p-k))}{c^2\beta_{\min}^2\rho_{2k}/2}$, where $0 < c^2/2 \leq 1/(18\sigma^2)$, then $\mathbb{P}(\mathcal{E}^c)\to 0$. From the above expression, it is clear that if $n > \max\left\{\frac{\log\binom{p}{k}}{c_1^2 n\beta_{\min}^2\rho_{2k}^2/2}, \frac{4+\log(k^2(p-k))}{c^2\beta_{\min}^2\rho_{2k}/2}\right\}$, then $\mathbb{P}(\widehat{S} = S^*)\to 1$. Choosing $c_1 = c$ and realizing that $\rho_{2k}^2 < \rho_{2k}$, we get the desired result.

B.1 Proof of Equation (29)

Using Lemma 11, we have

$$\Gamma_S(i,i') = \Pi[S] - \Pi[A_i, i'] = \underbrace{\frac{(\Pi^\perp[A_i]X_i)(\Pi^\perp[A_i]X_i)^T}{\|\Pi^\perp[A_i]X_i\|_2^2}}_{P_i} - \underbrace{\frac{(\Pi^\perp[A_i]X_{i'})(\Pi^\perp[A_i]X_{i'})^T}{\|\Pi^\perp[A_i]X_{i'}\|_2^2}}_{P_{i'}}. \tag{30}$$

Next, we evaluate $\xi^T P_i \xi$:

$$\xi^T P_i \xi = \frac{(X_i^T \Pi^\perp[A_i]\xi)^2}{\|\Pi^\perp[A_i]X_i\|_2^2} \overset{(a)}{=} \frac{(X_i^T \Pi[A_i \cup \bar{S}]\Pi^\perp[A_i]\xi)^2}{\|\Pi^\perp[A_i]X_i\|_2^2} \tag{31}$$

$$\overset{(b)}{=} \frac{(X_i^T X_{A_i \cup \bar{S}}(X_{A_i \cup \bar{S}}^T X_{A_i \cup \bar{S}})^{-1} X_{A_i \cup \bar{S}}^T \Pi^\perp[A_i]\xi)^2}{\|\Pi^\perp[A_i]X_i\|_2^2} \tag{32}$$

$$\overset{(c)}{=} \frac{\left(X_i^T X_{A_i \cup \bar{S}}(X_{A_i \cup \bar{S}}^T X_{A_i \cup \bar{S}})^{-1} \left[0_{n \times |A_i|}\ X_{\bar{S}}\right]^T \Pi^\perp[A_i]\xi\right)^2}{\|\Pi^\perp[A_i]X_i\|_2^2}. \tag{33}$$

Recall that $\xi = X_{\bar{S}}\beta^*_{\bar{S}}$. Step (a) follow since $\Pi^\perp[A_i]\xi$ is in the span of $A_i \cup \bar{S}$. Step (b) uses the definition of the projection matrix $\Pi[A_i \cup \bar{S}]$. Step (c) uses the fact that X_{A_i} is orthogonal to the projection matrix $\Pi^\perp[A_i]$. The notation $0_{n \times |A_i|}$ refers to a matrix of size $|A_i| \times n$ with all zeros. We now want to find an upper bound for the last expression above. Using Holder's inequality, we have

$$\xi^T P_i \xi \le \frac{\left\| \left[X_i^T X_{A_i \cup \bar{S}}(X_{A_i \cup \bar{S}}^T X_{A_i \cup \bar{S}})^{-1} \right]_{\bar{S}} \right\|_1^2 \cdot \|X_S^T \Pi^\perp[A_i]\xi\|_\infty^2}{\|\Pi^\perp[A_i]X_i\|_2^2}, \tag{34}$$

where the notation $\|[v_{A_i \cup \bar{S}}]_{\bar{S}}\|_p$ only computes the norm over the variables in $\bar{S}$. For notational simplicity, define $\gamma_{i,S}$ as follows:

$$\gamma_{i,S} = \frac{\sqrt{n}\left\| \left[X_i^T X_{A_i \cup \bar{S}}(X_{A_i \cup \bar{S}}^T X_{A_i \cup \bar{S}})^{-1} \right]_{\bar{S}} \right\|_1^2}{\|\Pi^\perp[A_i]X_i\|_2^2}. \tag{35}$$

Recall that our original goal is to evaluate $\xi^T \Gamma_S(i,i')\xi$ and $\|\Gamma_S(i,i')\xi\|_2$. We first find an upper bound for $\|\Gamma_S(i,i')\|_2$:

$$\begin{aligned}
\|\Gamma_S(i,i')\xi\|_2 &\overset{(a)}{\le} \|P_i\xi\|_2 + \|P_{i'}\xi\|_2 \overset{(b)}{\le} \frac{\sqrt{\gamma_{i,S}}\|X_S^T\Pi^\perp[A_i]\xi\|_\infty}{\|\Pi^\perp[A_i]X_i\|_2} + \frac{|X_{i'}^T\Pi^\perp[A_i]\xi|}{\sqrt{n}} \\
&\overset{(c)}{\le} \frac{\sqrt{\gamma_{i,S}}\|X_S^T\Pi^\perp[A_i]\xi\|_\infty}{\sqrt{n}} + \frac{\|X_S^T\Pi^\perp[A_i]\xi|^2\|_\infty}{\|\Pi^\perp[A_i]X_{i'}\|_2} \\
&\overset{(d)}{\le} \|X_S^T\Pi^\perp[A_i]\xi\|_\infty \left(\frac{\sqrt{\gamma_{i,S}}}{\sqrt{n}} + \frac{1}{\|\Pi^\perp[A_i]X_{i'}\|_2}\right) \\
&\overset{(e)}{\le} \|X_S^T\Pi^\perp[A_i]\xi\|_\infty \left(\frac{\sqrt{\gamma_{i,S}}}{\sqrt{n}} + \frac{1}{\sqrt{n\rho_k}}\right)
\end{aligned}$$

Step (a) is the triangle inequality. Step (b) substitutes expressions for P_i and $P_{i'}$. Step (c) follows because $i' \in \overline{S}$. Step (d) is simple algebra. Finally, step (e) uses the fact that $\|\Pi^{\perp}[A_i]X_{j''}\| \geq \sqrt{n\rho_{k,1}}$ for all $j'' \in \overline{S}$. We now bound the minimum value of $\xi^T \Gamma_S(i,i')\xi$ for all $i' \in S$:

$$\min_{i' \in S} \xi^T \Gamma_S(i,i')\xi = \min_{i' \in S} \left[\xi^T P_i \xi - \xi^T P_{i'} \xi\right] \tag{36}$$

$$\overset{(a)}{\leq} \frac{\gamma_{i,S}\|X_S^T \Pi^{\perp}[A_i]\xi\|_{\infty}^2}{n} - \max_{i' \in \overline{S}} \frac{(X_{i'}\Pi^{\perp}[A_i]\xi)^2}{\|\Pi^{\perp}[A_i]X_{i'}\|_2^2} \tag{37}$$

$$\overset{(b)}{\leq} \frac{\gamma_{i,S}\|X_S^T \Pi^{\perp}[A_i]\xi\|_{\infty}^2}{n} - \frac{(X_{\overline{S}}\Pi^{\perp}[A_i]\xi)^2}{\|\Pi^{\perp}[A_i]X_{i'}\|_2^2} \tag{38}$$

$$\overset{(c)}{\leq} \|X_S^T \Pi^{\perp}[A_i]\xi\|_{\infty}^2 \left[\frac{\gamma_{i,S}}{n} - \frac{1}{n}\right] \tag{39}$$

Step (a) uses the upper bound in (34) and the definition of $P_{i'}$. Step (b) uses the definition of the ℓ_{∞} norm. Step (c) uses some simple algebera and the bound $n\rho_k < \|\Pi^{\perp}[A_i]X_{j''}\|_2^2 \leq n$ for any $j'' \notin A_i$. We are now ready to find an upper bound for the expression of interest:

$$\min_{i \in (S^*)^c \cap S, i' \in \overline{S}} \left[\xi^T \Gamma_S(i,i')\xi + 2\sigma\delta_n \|\Gamma_S(i,i')\|_2 + 2\delta_n^2\sigma^2\right]$$

$$\overset{(a)}{\leq} \min_{i \in (S^*)^c \cap S} \left[\overbrace{\|X_S^T \Pi^{\perp}[A_i]\xi\|_{\infty}^2}^{R_i^2} \left[\frac{\gamma_{i,S}}{n} - \frac{1}{n}\right] + 2\sigma\delta_n \frac{\|X_S^T \Pi^{\perp}[A_i]\xi\|_{\infty}}{\sqrt{n}} \left(\sqrt{\gamma_{i,S}} + \frac{1}{\sqrt{\rho_k}}\right) + 2\delta_n^2\sigma^2\right]$$

$$\overset{(b)}{\leq} \min_{i \in (S^*)^c \cap S} R_i^2 \left[\left(\frac{\gamma_{i,S}}{n} - \frac{1}{n}\right) + \frac{2\sigma\delta_n}{\sqrt{n}R_i} \left(\sqrt{\gamma_{i,S}} + \frac{1}{\sqrt{\rho_k}}\right) + \frac{2\delta_n^2\sigma^2}{R_i^2}\right]$$

$$\overset{(c)}{\leq} \min_{i \in (S^*)^c \cap S} R_i^2 \left[\left(\frac{\gamma_{i,S}}{n} - \frac{1}{n}\right) + \frac{2\sigma\delta_n}{n\sqrt{n}\rho_{2k}\beta_{\min}} \left(\sqrt{\gamma_{i,S}} + \frac{1}{\sqrt{\rho_k}}\right) + \frac{2\delta_n^2\sigma^2}{n^2\rho_{2k}^2\beta_{\min}^2}\right]$$

$$\overset{(d)}{\leq} C \min_{i \in (S^*)^c \cap S} \left[(\gamma_{i,S} - 1) + \frac{2\sigma\delta_n}{\sqrt{n}\rho_{2k}\beta_{\min}} \left(\sqrt{\gamma_{i,S}} + \frac{1}{\sqrt{\rho_{k,1}}}\right) + \frac{2\delta_n^2\sigma^2}{n\rho_{2k}^2\beta_{\min}^2}\right]$$

$$\overset{(e)}{\leq} C \left[(\gamma_k - 1) + \frac{2\sigma\delta_n}{\sqrt{n}\rho_{2k}\beta_{\min}} \left(\sqrt{\gamma_k} + 1/\sqrt{\rho_{k,1}}\right) + \frac{2\delta_n^2\sigma^2}{n\rho_{2k}^2\beta_{\min}^2}\right]$$

Step (a) uses the bounds obtained in (36) and (39) and introduces the notation R_i for simplicity. The i' is captured in the ℓ_{∞} norm term. Step (b) is simple algebra. In step (c), we make use of the bound $R_i^2 = \|X_S^T \Pi^{\perp}[A_i]X_{\overline{S}}\beta_{\overline{S}}^*\|_{\infty}^2 \geq n^2\rho_{2k}^2\|\beta_{\overline{S}}^*\|_2^2/|\overline{S}| \geq n^2\rho_{2k}^2\beta_{\min}^2$. In Step (d), we factor out the terms that do not depend on i and represent them by C, where $C > 0$. In Step (e), we use the fact that $\min_{i \in (S^*)^c \cap S} \gamma_{i,S} = \gamma_k$, where γ_k is defined in (10). This equivalence can be easily established using the block-inversion formula.

Appendix C. Proof of Theorem 6

Suppose that after r iterations, SWAP outputs $\widehat{S} = S^{(r)}$. To ensure that $\widehat{S} = S^*$, we want to impose conditions so that with each iteration, SWAP takes positive steps towards the true support S^*. Let $S^{(1)}, S^{(2)}, \ldots, S^{(r)}$ be the intermediate supports computed in each iteration of SWAP. In what follows,

the conditions we impose will ensure that

$$|S^* \backslash S^{(1)}| \geq |S^* \backslash S^{(2)}| \geq \cdots \geq |S^* \backslash S^{(r-1)}| > |S^* \backslash S^{(r)}| = 0\,. \tag{40}$$

In other words, with each iteration, the number of active variables missed by SWAP will not increase and eventually decrease to 0. In order to ensure that (40) holds and r is much smaller than $\binom{p}{d}$, we define the following events:

$$\mathcal{E}_{S^*} = \{S^* = \arg\min_{S \in \Omega_k} \mathcal{L}(S; y, X)\}\,, \tag{41}$$

$$\mathcal{D}_S = \left\{ \min_{i \in S \cap (S^*)^c, i' \in S^c \cap S^*} \mathcal{L}(S^{(i,i')}; y, X) < \mathcal{L}(S; y, X) \right\}, \tag{42}$$

$$\mathcal{F}_S = \left\{ \min_{i \in S \cap (S^*)^c, i' \in S^c \cap S^*} \mathcal{L}(S^{(i,i')}; y, X) < \min_{j \in S, j' \in S^c \cap S^*} \mathcal{L}(S^{(j,j')}; y, X) \right\}. \tag{43}$$

The event $\mathcal{E}_{S^*}$ is the set of outcomes where S^* minimizes the loss. The event $\mathcal{D}_S$ is the set of outcomes where there exists at least one inactive variable in S that can be swapped with an active variable from S^c. Finally, the event $\mathcal{F}_S$ is the set of outcomes where an inactive variable in S can only be swapped with an active variable from S^c and an active variable from S cannot be swapped with an inactive variable from S^c. Before analyzing the events $\mathcal{E}_{S^*}$, $\mathcal{D}_S$, and $\mathcal{F}_S$, we establish an upper bound for the number of iterations in SWAP.

Lemma 7 *If for every $S \in \Omega_k$ such that $|S^* \backslash S| \leq d$, $\mathbb{P}(\mathcal{E}_{S^*}) = \mathbb{P}(\mathcal{D}_S) = \mathbb{P}(\mathcal{F}_S) = 1$, then r can be upper bounded as follows:*

$$r \leq \begin{cases} d\binom{k}{d} & d \leq \lceil k/2 \rceil \\ 2^k & d > \lceil k/2 \rceil \end{cases}\,. \tag{44}$$

Proof If $d \leq 1$, it is clear that $\mathbb{P}(\mathcal{E}_{S^*}) = 1$ ensures that $r = 1$ if $S = S^*$ and $r = 2$ if $|S \backslash S^*| = 1$. If $d > 1$, then the conditions $\mathbb{P}(\mathcal{D}_S) = \mathbb{P}(\mathcal{F}_S) = 1$ ensures that in each iteration, either an active variable is swapped with an active variable or an inactive variable is swapped with an active variable. For each $d' \leq d$, there are $\binom{k}{k-d'}$ possible supports once the inactive variables have been fixed. This means that the maximum possible value of r is $1 + \sum_{d'=2}^{d} \binom{k}{k-d'} = 1 + \sum_{d'=2}^{d} \binom{k}{d'}$. The upper bound in (7) follows using standard upper bounds of the binomial coefficients. ■

For notational convenience, define the event $\mathscr{E} = \mathcal{E}_{S^*} \bigcap_{\ell=1}^{r-1} \mathcal{D}_{S^{(\ell)}} \bigcap_{\ell=1}^{r-1} \mathcal{F}_{S^{(\ell)}}$. We have the following lower bound for the probability of correctly recovering the true support:

$$\mathbb{P}(\widehat{S} = S^*) \geq \mathbb{P}(\widehat{S} = S^* | \mathscr{E}) \mathbb{P}(\mathscr{E}) = \mathbb{P}(\mathscr{E}) \tag{45}$$

$$\geq 1 - \mathbb{P}(\mathcal{E}_{S^*}^c) - \sum_{\ell=1}^{r-1} \mathbb{P}(\mathcal{D}_{S^{(\ell)}}^c) - \sum_{\ell=1}^{r-1} \mathbb{P}(\mathcal{F}_{S^{(\ell)}}^c) \tag{46}$$

$$\geq 1 - \mathbb{P}(\mathcal{E}_{S^*}^c) - r \max_{S \in \Omega_{k,d} \backslash S^*} \mathbb{P}(\mathcal{D}_S^c) - r \max_{S \in \Omega_{k,d} \backslash S^*} \mathbb{P}(\mathcal{F}_S^c)\,. \tag{47}$$

From Theorem 3, we know that if $n > \frac{4 + \log(k^2(p-k))}{c^2 \beta_{\min}^2 \rho_{2k}/2}$, where $0 < c^2 \leq 1/(18\sigma^2)$, then $\mathbb{P}(\mathcal{E}_{S^*}^c) \to 0$. Furthermore, from the proof of Theorem 4 in Section B, we know that $\mathbb{P}(\mathcal{D}_S^c) \leq 4e^{-c^2 n \beta_{\min}^2 \rho_{2k}^2/2}$.

Thus, we only need to analyze $\mathbb{P}(\mathcal{F}_S^c)$. Using the definition of the least-squares loss, we have

$$\mathcal{F}_S^c = \left\{ \min_{i,i'} \max_{j,j'} \left[\xi^T \Theta_S(i,i',j,j') \xi + w^T \Theta_S(i,i',j,j') w + 2\xi^T \Theta_S(i,i',j,j') w \right] \geq 0 \right\},$$

where $\xi = X\beta^*$, $i \in (S^*)^c \cap S, i' \in S^c \cap S^*$, $j \in S$, and $j' \in S^c \cap (S^*)^c$. The matrix $\Theta_S(i,i',j,j')$ is defined as

$$\Theta_S(i,i',j,j') = \Pi^{\perp}[A_i, i'] - \Pi^{\perp}[A_j, j']. \tag{48}$$

Using similar methods as in Section B, we can write down an upper bound for $\frac{\mathbb{P}(\mathcal{F}_S^c)}{k(p-k)}$ as follows:

$$\mathbb{1}\left(\min_{i,i'} \max_{j,j'} \left[\xi^T \Theta_S(i,i',j,j') \xi + 2\delta_n^2 \sigma^2 + 2\sigma ||\Theta_S(i,i',j,j')^T \xi||_2 \delta_n \geq 0 \right] \right) + 4e^{-\delta_n^2/2},$$

where the $k(p-k)$ arises because $\mathcal{F}_S$ is defined by a term that takes a maximum over $k(p-k)$ possible number of elements. In Section C.1, we show that for some $C > 0$ and ν_d defined in (11),

$$\begin{aligned} &\min_{i,i'} \max_{j,j'} \left[\xi^T \Theta_S(i,i',j,j') \xi + 2\delta_n^2 \sigma^2 + 2\sigma ||\Theta_S(i,i',j,j')^T \xi||_2 \delta_n \right] \\ &\qquad \leq C \left[(\nu_d - 1) + \frac{2\sigma\delta_n}{\sqrt{\frac{dn}{d+1}} \rho_{2k} \beta_{\min}} \left(\sqrt{\nu_d} + \sqrt{\frac{2}{\rho_{k-1,0}}} \right) + \frac{2\delta_n^2}{\frac{d}{d+1} n \rho_{2k}^2 \beta_{\min}^2 / 2} \right] \end{aligned} \tag{49}$$

Choosing $\delta_n^2 = c_1^2 \frac{d}{d+1} n \rho_{2k}^2 \beta_{\min}^2$, we can upper bound $\mathbb{P}(\mathcal{F}_S^c)$ as

$$k(p-k)\mathbb{1}\left((\nu_d - 1) + 2\sigma c_1 \left(\sqrt{\nu_d} + \sqrt{\frac{2}{\rho_{k-1,0}}} \right) + 2(c_1\sigma)^2 \geq 0 \right) + 4k(p-k) e^{-c_1^2 \frac{d}{d+1} n \rho_{2k}^2 \beta_{\min}^2}$$

If $(\nu_d - 1) + 2\sigma c_1 \left(\sqrt{\nu_d} + \sqrt{\frac{2}{\rho_{k-1,0}}} \right) + 2(c_1\sigma)^2 < 0$, then $\mathbb{P}(\mathcal{F}_S^c) \leq 4k(p-k) e^{-c_1^2 \frac{d}{d+1} n \rho_{2k}^2 \beta_{\min}^2 / 2}$. Plugging into (47), we have

$$\begin{aligned} \mathbb{P}(\widehat{S} = S^*) &\geq 1 - \mathbb{P}(\mathcal{E}_{S^*}) - 4re^{-c^2 n \beta_{\min}^2 \rho_{2k}^2 / 2} - 4rk(p-k) e^{-c^2 \frac{d}{d+1} n \beta_{\min}^2 \rho_{2k}^2 / 2} && (50) \\ &\overset{(a)}{\geq} 1 - \mathbb{P}(\mathcal{E}_{S^*}) - 5rk(p-k) e^{-c^2 n \beta_{\min}^2 \rho_{2k}^2 / 4} && (51) \end{aligned}$$

In (a), we use the simple bound of $d/(d+1) \geq 1/2$ for $d \geq 1$ and let $c_1 = c$, where $0 < c^2/2 < 1/(18\sigma^2)$. From Lemma 7, $r \leq 2^k$, so it is sufficient to choose $n > (2k + \log(k(p-k)))/(c^2 \beta_{\min}^2 \rho_{2k}^2 / 4)$ for $\mathbb{P}(\widehat{S} = S^*) \to 1$.

C.1 Proof of Equation (49)

Using Lemma 11, we have

$$\Theta_S(i,i',j,j') = \underbrace{\Pi^{\perp}[A_{ij}] X_{ij'} \left(X_{ij'}^T \Pi^{\perp}[A_{ij}] X_{ij'} \right)^{-1} X_{ij'}^T \Pi^{\perp}[A_{ij}]}_{P_{ij'}}$$

$$- \underbrace{\Pi^{\perp}[A_{ij}]X_{i'j}\left(X_{i'j}^T\Pi^{\perp}[A_{ij}]X_{i'j}\right)^{-1}X_{i'j}^T\Pi^{\perp}[A_{ij}]}_{P_{i'j}}. \tag{52}$$

Using the notation $Z_l = X_l^T\Pi^{\perp}[A_{ij}]X_l$, it is easy to see that the following holds:

$$\begin{aligned}\Lambda_{\min}\left(X_{ij'}^T\Pi^{\perp}[A_{ij}]X_{ij'}\right) &= (\|Z_i\|_2^2 + \|Z_{j'}\|_2^2 - \sqrt{(\|Z_i\|_2^2 - \|Z_{j'}\|_2^2)^2 - 4(Z_i^TZ_{j'})^2})/2 \\ &\geq \min\{\|Z_{i'}\|_2^2, \|Z_j\|_2^2\} \qquad (53)\\ \Lambda_{\max}\left(X_{ij'}^T\Pi^{\perp}[A_{ij}]X_{ij'}\right) &= (\|Z_i\|_2^2 + \|Z_{j'}\|_2^2 + \sqrt{(\|Z_i\|_2^2 - \|Z_{j'}\|_2^2)^2 - 4(Z_i^TZ_{j'})^2})/2 \\ &\leq \max\{\|Z_i\|_2^2, \|Z_{j'}\|_2^2\} \leq n. \qquad (54)\end{aligned}$$

Next, we have the following upper bound for $\|P_{ij'}\xi\|_2^2$:

$$\begin{aligned}\|P_{ij'}\xi\|_2^2 &\leq \left\|\left(X_{ij'}^T\Pi^{\perp}[A_{ij}]X_{ij'}\right)^{-1}\right\|_2 \cdot \|X_{ij'}^T\Pi^{\perp}[A_{ij}]\xi\|_2^2 \qquad (55)\\ &\leq \frac{\|X_{ij'}^T\Pi^{\perp}[A_{ij}]\xi\|_2^2}{\Lambda_{\min}\left(X_{ij'}^T\Pi^{\perp}[A_{ij}]X_{ij'}\right)} \leq \frac{(X_i^T\Pi^{\perp}[A_{ij}]\xi)^2 + (X_{j'}^T\Pi^{\perp}[A_{ij}]\xi)^2}{\min\{\|\Pi^{\perp}[A_{ij}]X_i\|_2^2, \|\Pi^{\perp}[A_{ij}]X_{j'}\|_2^2\}}. \qquad (56)\end{aligned}$$

Using similar steps as in Section B.1, we have

$$\begin{aligned}(X_i^T\Pi^{\perp}[A_{ij}]\xi)^2 &= \left\|X_i^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j}(X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j})^{-1}\right\|_1^2 \cdot \|X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2, \qquad (57)\\ (X_{j'}^T\Pi^{\perp}[A_{ij}]\xi)^2 &= \left\|X_{j'}^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j}(X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j})^{-1}\right\|_1^2 \cdot \|X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2, \qquad (58)\end{aligned}$$

where $\bar{S_j} = \{S^*\backslash S\}\cup\{j\}$. Defining $\nu_{i,j,j',S}$ as

$$\nu_{i,j,j',S} = \frac{\left\|X_i^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j}(X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j})^{-1}\right\|_1^2 + \left\|X_{j'}^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j}(X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]X_{\bar{S}_j})^{-1}\right\|_1^2}{\min\{\|\Pi^{\perp}[A_{ij}]X_i\|_2^2, \|\Pi^{\perp}[A_{ij}]X_{j'}\|_2^2\}/n}$$

we have $\|P_{ij'}\xi\|_2^2 \leq \nu_{i,j,j',S}\|X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2/n$. Next, upper and lower bounds for $\|P_{i'j}\xi\|_2^2$ are given as follows:

$$\begin{aligned}\|P_{i'j}\xi\|_2^2 &\leq \frac{(X_{i'}^T\Pi^{\perp}[A_{ij}]\xi)^2 + (X_j^T\Pi^{\perp}[A_{ij}]\xi)^2}{\min\{\|\Pi^{\perp}[A_{ij}]X_{i'}\|_2^2, \|\Pi^{\perp}[A_{ij}]X_j\|_\infty^2\}} \leq \frac{2\|X_{\bar{S}\cup j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2}{n\rho_{k-1,0}} \qquad (59)\\ \max_{i'\in\bar{S}}\|P_{i'j}\xi\|_2^2 &\geq \max_{i'\in S}\|X_{i'j}\Pi^{\perp}[A_{ij}]\xi\|_2^2/n = \|X_S^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2/n + (X_j^T\Pi^{\perp}[A_{ij}]\xi)^2/n \\ &\geq \|X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2/n \qquad (60)\end{aligned}$$

We can now upper bound $\xi^T\Theta_S(i,i',j,j')\xi$ and $\|\Theta_S(i,i',j,j')\xi\|$:

$$\begin{aligned}\min_{i'\in\bar{S}}\xi^T\Theta_S(i,i',j,j')\xi &= \|P_{ij'}\|_2^2 - \max_{i'\in\bar{S}}\|P_{i'j}\|_2^2 \qquad (61)\\ &\leq \nu_{i,j,j',S}\|X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2/n - \|X_{\bar{S}_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2/n \qquad (62)\end{aligned}$$

$$\|\Theta_S(i,i',j,j')\xi\| \le \|P_{ij'}\xi\|_2 + \|P_{i'j}\xi\|_2 \tag{63}$$

$$\le \sqrt{\nu_{i,j,j',S}}\|X_{S_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty/\sqrt{n} + \frac{\sqrt{2}\|X_{\bar{S\cup j}}\Pi^{\perp}[A_{ij}]\xi\|_\infty}{\sqrt{n\rho_{k-1,0}}} \tag{64}$$

Putting everything together, we can upper bound the expression of interest as follows:

$$\frac{\|X_{S_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2}{n}\left[(\nu_{i,j,j',S}-1) + \frac{2\sigma\delta_n\sqrt{n}}{\|X_{S_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty}\left(\sqrt{\nu_{i,j,j',S}} + \frac{\sqrt{2}}{\sqrt{\rho_{k-1,0}}}\right) + \frac{2\delta_n^2 n}{\|X_{S_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2}\right]$$

$$\le C\left[(\nu_{i,j,j',S}-1) + \frac{2\sigma\delta_n}{\sqrt{\frac{dn}{d+1}}\rho_{2k}\beta_{\min}}\left(\sqrt{\nu_{i,j,j',S}} + \frac{\sqrt{2}}{\sqrt{\rho_{k-1,0}}}\right) + \frac{2\delta_n^2}{\frac{d}{d+1}n\rho_{2k}^2\beta_{\min}^2}\right], \tag{65}$$

where $C>0$ and we use the inequality $\|X_{S_j}^T\Pi^{\perp}[A_{ij}]\xi\|_\infty^2 \ge \frac{d}{d+1}n^2\rho_{2k}^2\beta_{\min}^2$. Taking the minimum over $i\in(S^*)^c$ and maximum over $j\in S$ and $j'\in S^c\cap(S^*)^c$, we get the desired upper bound of

$$C\left[(\nu_d-1) + \frac{2\sigma\delta_n}{\sqrt{\frac{dn}{d+1}}\rho_{2k}\beta_{\min}}\left(\sqrt{\nu_d} + \frac{\sqrt{2}}{\sqrt{\rho_{k-1,0}}}\right) + \frac{2\delta_n^2}{\frac{dn}{d+1}\rho_{2k}^2\beta_{\min}}\right],$$

where ν_d is defined in (11).

Appendix D. Some Important Lemmas

In this Section, we collect some important lemmas we used in the proofs. For the first three lemmas, let w be a random vector with parameter σ so that the entries of w are i.i.d. zero mean sub-Gaussian random variables with parameter σ.

Lemma 8 (Hsu et al. (2012)) *If A is a rank ℓ projection matrix, then $\mathbb{P}(\|Aw\|_2^2/\sigma^2 - \ell > 2\sqrt{\ell t} + 2t) \le e^{-t}$ for all $t\ge 0$.*

Lemma 9 *If A_1 and A_2 are rank ℓ projection matrices, then for $t\ge 1$,*

$$\mathbb{P}\left(\left|\|A_1w\|_2^2 - \|A_2w\|_2^2\right| \ge 8\sigma^2\ell t\right) \le 2e^{-\ell t}$$

Proof Using the triangle inequality and the union bound, we have

$$\mathbb{P}\left(\left|\|A_1w\|_2^2 - \|A_2w\|_2^2\right|/\sigma^2 > 2x\right) \le \mathbb{P}\left(\left|\|A_1w\|_2^2/\sigma^2 - \ell\right| > x\right) + \mathbb{P}\left(\left|\|A_2w\|_2^2/\sigma^2 - \ell\right| > x\right)$$
$$\le 2\mathrm{P}\left(\left|\|A_1w\|_2^2/\sigma^2 - \ell\right| > x\right).$$

Analyzing $\mathbb{P}\left(\left|\|A_1w\|_2^2/\sigma^2 - \ell\right| > x\right)$, we have

$$\mathbb{P}\left(\left|\|A_1w\|_2^2/\sigma^2 - \ell\right| > x\right) \le \mathbb{P}\left(\|A_1w\|_2^2/\sigma^2 - \ell > x\right) + \mathbb{P}\left(\|A_1w\|_2^2/\sigma^2 - \ell < -x\right)$$
$$\le \mathbb{P}\left(\|A_1w\|_2^2/\sigma^2 - \ell > x\right), \text{ when } x > \ell.$$

Substituting, we have that for $x > \ell$,

$$\mathbb{P}\left(\left|\|A_1 w\|_2^2 - \|A_2 w\|_2^2\right| / \sigma^2 > 2x\right) \leq 2\mathbb{P}\left(\|A_1 w\|_2^2 / \sigma^2 - \ell > x\right).$$

Let $x = 2\sqrt{\ell \cdot \ell t} + 2\ell t$. Since $4t \geq 2\sqrt{t} + 2t$, we get the desired result using Lemma 8. ■

Lemma 10 *For any vector $a \in \mathbb{R}^n$, $\mathbb{P}(|a^T w| > t) \leq 2e^{-t^2/(2\|a\|_2^2\sigma^2)}$.*

Lemma 11 $\Pi[A_1, A_2] = \Pi[A_1] + \Pi^{\perp}[A_1] X_{A_2} (X_{A_2}^T \Pi^{\perp}[A_1] X_{A_2})^{-1} X_{A_2}^T \Pi^{\perp}[A_1]$

Proof Follows from the block-inversion formula. ■

In the next lemma, let $\Lambda_{\min}(W)$ and $\Lambda_{\max}(W)$ denote the minimum and maximum eigenvalues of a matrix W.

Lemma 12 *Let $A_1, A_2 \subset [p]$ that are disjoint. We have the following results:*

$$\Lambda_{\min}((X_{A_1 \cup A_2}^T X_{A_1 \cup A_2})^{-1}) \leq \Lambda_{\min}((X_{A_1}^T \Pi^{\perp}[A_2] X_{A_1})^{-1}) \tag{66}$$

$$\Lambda_{\max}((X_{A_1}^T \Pi^{\perp}[A_2] X_{A_1})^{-1}) \leq \Lambda_{\max}((X_{A_1 \cup A_2}^T X_{A_1 \cup A_2})^{-1}) \tag{67}$$

Proof Follows from the block-inversion formula and extensions of the Cauchy's interlacing theorem. ■

References

P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of Lasso and Dantzig selector. *Ann. Stat.*, 37(4):1705–1732, 2009.

P. Bühlmann and S. Van De Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer-Verlag New York Inc, 2011.

P. Bühlmann, P. Rütimann, S. van de Geer, and C. Zhang. Correlated variables in regression: Clustering and sparse estimation. *Journal of Statistical Planning and Inference*, 143(11):1835–1858, 2013.

E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52 (2):489–509, 2006.

D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *Journal of symbolic computation*, 9(3):251–280, 1990.

M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83–91, Mar. 2008.

M. Elad. *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer, 2010.

M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

A. Fannjiang and W. Liao. Coherence pattern-guided compressive sensing with unresolved grids. *SIAM Journal on Imaging Sciences*, 5(1):179–202, 2012.

T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, 286(5439):531–537, 1999.

E. Grave, F. Obozinski, and F. Bach. Trace Lasso: a trace norm regularization for correlated designs. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

D. Hsu, S. Kakade, and T. Zhang. A tail inequality for quadratic forms of subgaussian random vectors. *Electronic Communications in Probability*, 17:no. 52, 1–6, 2012.

J. Huang, S. Ma, H. Li, and C.H. Zhang. The sparse laplacian shrinkage estimator for high-dimensional regression. *Ann. Stat.*, 39(4):2021, 2011.

A. Javanmard and A. Montanari. Model selection for high-dimensional regression under the generalized irrepresentability condition. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the Lasso. *Ann. Stat.*, 34(3):1436, 2006.

N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.

N. Meinshausen and B. Yu. Lasso-type recovery of sparse representations for high-dimensional data. *The Annals of Statistics*, pages 246–270, 2009.

M. Melanie. An introduction to genetic algorithms. *Cambridge, Massachusetts London, England, Fifth printing*, 3, 1999.

D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.

P. Ravikumar, M. J. Wainwright, and J. D. Lafferty. High-dimensional ising model selection using ℓ_1-egularized logistic regression. *Ann. Stat.*, 38(3):1287–1319, 2010.

M.R. Segal, K.D. Dahlquist, and B.R. Conklin. Regression approaches for microarray data analysis. *Journal of Computational Biology*, 10(6):961–980, 2003.

Y. She. Sparse regression with exact clustering. *Electronic Journal of Statistics*, 4:1055–1096, 2010.

D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, et al. Gene expression correlates of clinical prostate cancer behavior. *Cancer cell*, 1(2):203–209, 2002.

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

J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.

S. van de Geer, P. Bühlmann, and S. Zhou. The adaptive and the thresholded lasso for potentially misspecified models (and a lower bound for the lasso). *Electronic Journal of Statistics*, 5:688–749, 2011.

G. Varoquaux, A. Gramfort, and B. Thirion. Small-sample brain mapping: sparse recovery on spatially correlated designs with randomization and clustering. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, pages 1375–1382, 2012.

D. Vats. High-dimensional screening using multiple grouping of variables. *IEEE Transactions on Signal Processing*, to appear.

D. Vats and R. G. Baraniuk. When in doubt swap: High-dimensional sparse recovery from correlated measurements. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

R. Vershynin. *Compressed Sensing: Theory and Applications*, chapter Introduction to the non-asymptotic analysis of random matrices. Cambridge University Press, 2010.

M. J. Wainwright. Sharp thresholds for noisy and high-dimensional recovery of sparsity using ℓ_1-constrained quadratic programming (Lasso). *IEEE Transactions on Information Theory*, 55(5), May 2009a.

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

L. Wasserman and K. Roeder. High dimensional variable selection. *The Annals ofstatistics*, 37(5A): 2178, 2009.

T. Zhang. Some sharp performance bounds for least squares regression with l1 regularization. *The Annals of Statistics*, 37(5A):2109–2144, 2009.

T. Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11:1081–1107, March 2010.

T. Zhang. Adaptive forward-backward greedy algorithm for learning sparse representations. *IEEE Transactions on Information Theory*, 57(7):4689–4708, 2011.

P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.

H. Zhu and G.B. Giannakis. Sparse overcomplete representations for efficient identification of power line outages. *IEEE Transactions on Power Systems*, 27(4):2215 –2224, Nov. 2012.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.