Machine Learning
https://doi.org/10.1007/s10994-021-06040-4

Provable training set debugging for linear regression

Xiaomin Zhang[1] · **Xiaojin Zhu**[1] · **Po-Ling Loh**[2]

Received: 28 January 2021 / Revised: 27 May 2021 / Accepted: 5 July 2021

Abstract
We investigate problems in penalized *M*-estimation, inspired by applications in machine learning debugging. Data are collected from two pools, one containing data with possibly contaminated labels, and the other which is known to contain only cleanly labeled points. We first formulate a general statistical algorithm for identifying buggy points and provide rigorous theoretical guarantees when the data follow a linear model. We then propose an algorithm for tuning parameter selection of our Lasso-based algorithm with theoretical guarantees. Finally, we consider a two-person "game" played between a bug generator and a debugger, where the debugger can augment the contaminated data set with cleanly labeled versions of points in the original data pool. We develop and analyze a debugging strategy in terms of a Mixed Integer Linear Programming (MILP). Finally, we provide empirical results to verify our theoretical results and the utility of the MILP strategy.

Keywords Robust statistics · Outlier detection · Tuning parameter selection · Optimization

1 Introduction

Modern machine learning systems are extremely sensitive to training set contamination. Since sources of error and noise are unavoidable in real-world data (e.g., due to Mechanical Turkers, selection bias, or adversarial attacks), an urgent need has arisen to perform automatic debugging of large data sets. Cadamuro et al. (2016), Zhang et al. (2018) proposed a method called "machine learning debugging" to identify training

Editors: Annalisa Appice, Sergio Escalera, Jose A. Gamez, Heike Trautmann.

✉ Xiaomin Zhang
xzhang682@wisc.edu

Xiaojin Zhu
jerryzhu@cs.wisc.edu

Po-Ling Loh
pll28@cam.ac.uk

[1] Department of Computer Sciences, University of Wisconsin-Madison, Madison, USA

[2] Department of Pure Mathematics and Mathematical Statistics, University of Cambridge, Cambridge, UK

Published online: 16 August 2021

set errors by introducing new clean data. Consider the following real-world scenario: Company A collects movie ratings for users on a media platform, from which it learns relationships between features of movies and ratings in order to perform future recommendations. A competing company B knows A's learning method and hires some users to provide malicious ratings. Company A could employ a robust method for learning contaminated data—but in the long run, it would be more effective for company A to *identify* the adversarial users and prevent them from submitting additional buggy ratings in the future. This distinguishes debugging from classical learning. The debugging problem also assumes that company A can hire an expert to help rate movies, from which it obtains a second trusted data set which is generally smaller than the original data set due to budget limitations. In this paper, we will study a theoretical framework for the machine learning debugging problem in a linear regression setting, where the main goal is to identify bugs in the data. We will also discuss theory and algorithms for selecting the trusted data set.

Our *first contribution* is to provide a rigorous theoretical framework explaining how to identify errors in the "buggy" data pool. Specifically, we embed a squared loss term applied to the trusted data pool into the extended Lasso algorithm proposed by Nguyen and Tran (2013), and reformulate the objective to better service the debugging task. Borrowing techniques from robust statistics (Huber and Ronchetti 2011; She and Owen 2011; Nguyen and Tran 2013; Foygel and Mackey 2014; Slawski and Ben-David 2017) and leveraging results on support recovery analysis (Wainwright 2009; Meinshausen and Yu 2009), we provide sufficient conditions for successful debugging in linear regression. We emphasize that our setting, involving data coming from multiple pools, has not been studied in any of the earlier papers.

The work of Nguyen and Tran (2013), Foygel and Mackey (2014) [and more recently, Sasai and Fujisawa (2020)] provided results for the extended Lasso with a theoretically optimal choice of tuning parameter, which depends on the unknown noise variance in the linear model. Our *second contribution* is to discuss a rigorous procedure for tuning parameter selection which does not require such an assumption. Specifically, our algorithm starts from a sufficiently large initial tuning parameter that produces the all-zeros vector as an estimator. Assuming the sufficient conditions for successful support recovery are met, this tuning parameter selection algorithm is guaranteed to terminate with a correct choice of tuning parameter after a logarithmic number of steps. Note that when outliers exist in the training data set, it is improper to use cross-validation to select the tuning parameter due to possible outliers in the validation data set.

Our *third contribution* considers how to design a second clean data pool, which is an important but previously unstudied problem in machine learning debugging. We consider a two-player "game" between a bug generator and debugger, where the bug generator performs adversarial attacks (Chakraborty et al. 2018), and the debugger applies Lasso-based linear regression to the augmented data set. On the theoretical side, we establish a sufficient condition under which the debugger can always beat the bug generator, and show how to translate this condition into a debugging strategy based on mixed integer linear programming. Our theory is only derived in the "noiseless" setting; nonetheless, empirical simulations show that our debugging strategy also performs well in the noisy setting. We experimentally compare our method to two other algorithms motivated by the machine learning literature, which involve designing two neural networks, one to correct labels and one to fit cleaned data (Veit et al. 2017); and a method based on semi-supervised learning that weights the noisy and clean datasets differently and employs a similarity matrix based on the graph Laplacian (Fergus et al. 2009).

The remainder of the paper is organized as follows: Sect. 2 introduces our novel framework for machine learning debugging using weighted M-estimators. Section 3 provides theoretical guarantees for recovery of buggy data points. Section 4 presents our algorithm for tuning parameter selection and corresponding theoretical guarantees. Section 5 discusses strategies for designing the second pool. Section 6 provides experimental results. Section 7 concludes the paper.

Notation We write $\Lambda_{\min}(A)$ and $\Lambda_{\max}(A)$ to denote the minimum and maximum eigenvalues, respectively, of a matrix A. We use $Null(A)$ to denote the nullspace of A. For subsets of row and column indices S and T, we write $A_{S,T}$ to denote the corresponding submatrix of A. We write $\|A\|_{\max}$ to denote the elementwise ℓ_∞-norm, $\|A\|_2$ to denote the spectral norm, and $\|A\|_\infty$ to denote the ℓ_∞-operator norm. For a vector $v \in \mathbb{R}^n$, we write $\text{supp}(v) \subseteq \{1, \dots, n\}$ to denote the support of v, and $\|v\|_\infty = \max |v_i|$ to denote the maximum absolute entry. We write $\|v\|_p$ to denote the ℓ_p-norm, for $p \geq 1$. We write $\text{diag}(v)$ to denote the $n \times n$ diagonal matrix with entries equal to the components of v. For $S \subseteq \{1, \dots, n\}$, we write v_S to denote the $|S|$-dimensional vector obtained by restricting v to S. We write $[n]$ as shorthand for $\{1, \dots, n\}$.

2 Problem formulation

We first formalize the data-generating models analyzed in this paper. Suppose we have observation pairs $\{(x_i, y_i)\}_{i=1}^n$ from the contaminated linear model

$$y_i = x_i^\top \beta^* + \gamma_i^* + \epsilon_i, \quad 1 \leq i \leq n, \tag{1}$$

where $\beta^* \in \mathbb{R}^p$ is the unknown regression vector, $\gamma^* \in \mathbb{R}^n$ represents possible contamination in the labels, and the ϵ_i's are i.i.d. sub-Gaussian noise variables with variance parameter σ^2. We also assume the x_i's are i.i.d. and $x_i \perp\!\!\!\perp \epsilon_i$. This constitutes the "first pool." Note that the vector γ^* is unknown and may be generated by some adversary. If $\gamma_i^* = 0$, the ith point is uncontaminated and follows the usual linear model; if $\gamma_i^* \neq 0$, the ith point is contaminated/buggy. Let $T := \text{supp}(\gamma^*)$ denote the indices of the buggy points, and let $t := |T|$ denote the number of bugs.

We also assume we have a clean data set which we call the "second pool." We observe $\{(\widetilde{x}_i, \widetilde{y}_i)\}_{i=1}^m$ satisfying

$$\widetilde{y}_i = \widetilde{x}_i^\top \beta^* + \widetilde{\epsilon}_i, \quad 1 \leq i \leq m, \tag{2}$$

where the $\widetilde{\epsilon}_i$'s are i.i.d. sub-Gaussian noise variables with parameter $\widetilde{\sigma}^2$. Let $L := \frac{\sigma}{\widetilde{\sigma}}$, and suppose $L \geq 1$. Unlike the first pool, the data points in the second pool are all known to be uncontaminated.

For notational convenience, we also use $X \in \mathbb{R}^{n \times p}$, $y \in \mathbb{R}^n$, and $\epsilon \in \mathbb{R}^m$ to denote the matrix/vectors containing the x_i's, y_i's, and ϵ_i's, respectively. Similarly, we define the matrices $\widetilde{X} \in \mathbb{R}^{m \times p}, \widetilde{y} \in \mathbb{R}^m$, and $\widetilde{\epsilon} \in \mathbb{R}^m$. Note that β^*, γ^*, T, t, and the noise parameters σ and $\widetilde{\sigma}$ are all assumed to be unknown to the debugger. In this paper, we will work in settings where $X^\top X$ is invertible.

Goal: Upon observing $\{(x_i, y_i)\}_{i=1}^n$, the debugger is allowed to design m points $\widetilde{X}$ in a stochastic or deterministic manner and query their corresponding labels $\widetilde{y}$, with the goal of recovering the support of γ^*. We have the following definitions:

Definition 1 An estimator $\widehat{\gamma}$ satisfies **subset support recovery** if $\text{supp}(\widehat{\gamma}) \subseteq \text{supp}(\gamma^*)$. It satisfies **exact support recovery** if $\text{supp}(\widehat{\gamma}) = \text{supp}(\gamma^*)$.

In words, when $\widehat{\gamma}$ satisfies subset support recovery, all estimated bugs are true bugs. When $\widehat{\gamma}$ satisfies exact support recovery, the debugger correctly flags *all* bugs. We are primarily interested in exact support recovery.

Weighted M-estimation Algorithm: We propose to optimize the joint objective

$$(\widehat{\beta}, \widehat{\gamma}) \in \arg \min_{\beta \in \mathbb{R}^p, \gamma \in \mathbb{R}^n} \left\{ \frac{1}{2n} \|y - X\beta - \gamma\|_2^2 + \frac{\eta}{2m} \|\widetilde{y} - \widetilde{X}\beta\|_2^2 + \lambda \|\gamma\|_1 \right\}, \tag{3}$$

where the weight parameter $\eta > 0$ determines the relative importance of the two data pools. The objective function applies the usual squared loss to the points in the second pool and introduces the additional variable γ to help identify bugs in the first pool. Furthermore, the ℓ_1-penalty encourages $\widehat{\gamma}$ to be sparse, since we are working in settings where the number of outliers is relatively small compared to the total number of data points. Note that the objective function (3) may equivalently be formulated as a weighted sum of M-estimators applied to the first and second pools, where the loss for the first pool is the robust Huber loss and the loss for the second pool is the squared loss (cf. Proposition 4 in Appendix A).

Lasso Reformulation: Recall that our main goal is to estimate (the support of) γ^* rather than β^*. Thus, we will restrict our attention to γ^* by reformulating the objectives appropriately. We first introduce some notation: Define the stacked vectors/matrices

$$X' = \begin{pmatrix} X \\ \sqrt{\frac{\eta n}{m}} \widetilde{X} \end{pmatrix}, \quad y' = \begin{pmatrix} y \\ \sqrt{\frac{\eta n}{m}} \widetilde{y} \end{pmatrix}, \quad \epsilon' = \begin{pmatrix} \epsilon \\ \sqrt{\frac{\eta n}{m}} \widetilde{\epsilon} \end{pmatrix}, \tag{4}$$

where $X' \in \mathbb{R}^{(m+n) \times p}$ and $y', \epsilon' \in \mathbb{R}^{m+n}$. For a matrix A, let $P_A = A(A^\top A)^{-1} A^\top$ and $P_A^\perp = I - A(A^\top A)^{-1} A^\top$ denote projection matrices onto the range of the column space of A and its orthogonal complement, respectively. For a matrix $S \subseteq [n]$, let M_S denote the $(n+m) \times |S|$ matrix with ith column equal to the canonical vector $e_{S(i)}$. Thus, right-multiplying by M_S truncates a matrix to only include columns indexed by S. We have the following useful result:

Proposition 1 *The objective function*

$$\widehat{\gamma} \in \arg \min_{\gamma \in \mathbb{R}^n} \left\{ \frac{1}{2n} \|P_{X'}^\perp y' - P_{X'}^\perp M_{[n]} \gamma\|_2^2 + \lambda \|\gamma\|_1 \right\} \tag{5}$$

shares the same solution for $\widehat{\gamma}$ with the objective function (3).

Proposition 1, proved in Appendix B, translates the joint optimization problem (3) into an optimization problem only involving the parameter of interest γ. We provide a discussion regarding the corresponding solution $\widehat{\beta}$ in Appendix A for the interested reader. Note that the optimization problem (5) corresponds to linear regression of the vector/matrix pairs $(P_{X'}^\perp y', P_{X'}^\perp M_{[n]})$ with a Lasso penalty, inspiring us to borrow techniques from high-dimensional statistics.

3 Support recovery

The reformulation (5) allows us to analyze the machine learning debugging framework through the lens of Lasso support recovery. The three key conditions we impose to ensure support recovery are provided below. Recall that we use M_T to represent the truncation matrix indexed by T.

Assumption 1 *(Minimum Eigenvalue)* Assume that there is a positive number $b'_{\min}$ such that

$$\Lambda_{\min}\left(M_T^\top P_{X'}^\perp M_T\right) \geq b'_{\min}. \tag{6}$$

Assumption 2 *[Mutual Incoherence]* Assume that there is a number $\alpha' \in [0, 1)$ such that

$$\|M_{T^c}^\top P_{X'}^\perp M_T (M_T^\top P_{X'}^\perp M_T)^{-1}\|_\infty \leq \alpha'. \tag{7}$$

Assumption 3 *(Gamma-Min)* Assume that

$$\min_{i \in T} |\gamma_i^*| > G' := \|(M_T^\top P_{X'}^\perp M_T)^{-1} M_T^\top P_{X'}^\perp \epsilon'\|_\infty + n\lambda \left\|(M_T^\top P_{X'}^\perp M_T)^{-1}\right\|_\infty. \tag{8}$$

Assumption 1 comes from a primal-dual witness argument (Wainwright 2009) to guarantee that the minimizer $\hat{\gamma}$ is unique. Assumption 2 measures a relationship between the sets T^c and T, indicating that the large number of nonbuggy covariates (i.e., T^c) cannot exert an overly strong effect on the subset of buggy covariates (Ravikumar et al. 2010). To aid intuition, consider an orthogonal design, where $X = \begin{bmatrix} cI_{[t],[p]} \\ c'I_{p\times p} \end{bmatrix}$ and $\widetilde{X} = c''I_{p\times p}$, for some $t < p$, and $c, c', c'' > 0$. We use the notation $I_{[t],[p]}$ to denote a submatrix of $I_{p\times p}$ with rows indexed by the set $[t]$. Suppose the first t points are bugs, and for simplicity, let $\eta = m/n$. Then the mutual incoherence condition requires $c < c' + \frac{(c'')^2}{c'}$, meaning that in every direction e_i, the component of buggy data cannot be too large compared to the nonbuggy data and the clean data. Assumption 3 lower-bounds the minimum absolute value of elements of γ. Note that λ is chosen based on ϵ', so the right-hand expression is a function of ϵ'. This assumption indeed captures the intuition that the signal-to-noise ratio, $\frac{\min_{i\in T}|\gamma_i^*|}{\sigma}$, needs to be sufficiently large.

We now provide two general theorems regarding subset support recovery and exact support recovery.

Theorem 1 (Subset support recovery) *Suppose* $P_{X'}^\perp$ *satisfies Assumptions 1 and 2. If the tuning parameter satisfies*

$$\lambda \geq \frac{2}{1-\alpha'} \left\| M_{T^c} P_{X'}^\perp \left(I - P_{X'}^\perp M_T (M_T^\top P_{X'}^\perp M_T)^{-1} M_T^\top P_{X'}^\perp\right) \frac{\epsilon'}{n} \right\|_\infty, \tag{9}$$

then the objective (5) *has a unique optimal solution* $\hat{\gamma}$, *satisfying* $\operatorname{supp}(\hat{\gamma}) \subseteq \operatorname{supp}(\gamma^*)$ *and* $\|\hat{\gamma} - \gamma^*\|_\infty \leq G'$.

Theorem 2 (Exact support recovery) *In addition to the assumptions in Theorem 1, suppose Assumption 3 holds. Then we have a unique optimal solution* $\hat{\gamma}$, *which satisfies exact support recovery.*

Note that we additionally need Assumption 3 to guarantee exact support recovery. This follows the aforementioned intuition regarding the assumption. In particular, recall that ϵ and $\widetilde{\epsilon}$ are sub-Gaussian vectors with parameters σ^2 and σ^2/L, respectively, where $L \geq 1$ (i.e., the clean data pool has smaller noise). The minimum signal strength $\min_{i \in T} |\gamma_i^*|$ needs to be at least $\Theta(\sigma\sqrt{\log n})$, since $\mathbb{E}\left[\max_{i \in [n]} |\epsilon_i|\right] \leq \sigma\sqrt{2\log(2n)}$. Intuitively, if $\min_{i \in T} |\gamma_i^*|$ is of constant order, it is difficult for the debugger to distinguish between random noise and intentional contamination.

We now present two special cases to illustrate the theoretical benefits of including a second data pool. Although Theorems 1 and 2 are stated in terms of *deterministic* design matrices and error vectors ϵ and $\widetilde{\epsilon}$, the assumptions can be shown to hold with high probability in the example. We provide formal statements of the associated results in Appendix C.2 and Appendix C.3.

Example 1 *(Orthogonal design)* Suppose Q is an orthogonal matrix with columns $q_1, q_2, \dots, q_p$, and consider the setting where $X_T = RQ^\top \in \mathbb{R}^{t \times p}$ and $X_{T^c} = FQ^\top \in \mathbb{R}^{p \times p}$, where $R = \left[\operatorname{diag}(\{r_i\}_{i=1}^t) \mid \mathbf{0}_{t \times (p-t)}\right]$ and $F = \operatorname{diag}(\{f_i\}_{i=1}^p)$. Thus, points in the contaminated first pool correspond to orthogonal vectors. Similarly, suppose the second pool consists of (rescaled) columns of Q, so $\widetilde{X} = WQ^\top \in \mathbb{R}^{m \times p}$, where $W = \operatorname{diag}(\{w_i\}_{i=1}^p)$. (To visualize this setting, one can consider $Q = I$ as a special case.) The mutual incoherence parameter is $\alpha' = \max_{1 \leq i \leq t} \left| \frac{r_i f_i}{f_i^2 + \eta \frac{n}{m} w_i^2} \right|$. Hence, $\alpha' < 1$ if the weight of a contaminated point dominates the weight of a clean point in any direction, e.g., when $|r_i| > |f_i|$ and $w_i = 0$; in contrast, if the second pool includes clean points $w_i q_i$ with sufficiently large $|w_i|$, we can guarantee that $\alpha' < 1$. Furthermore,

$$\begin{aligned} G' \approx & \sigma\left(\sqrt{2\log t} + c\right)\sqrt{1 + \max_{1 \leq i \leq t} \frac{r_i^2(Lf_i^2 + \frac{\eta n}{m} w_i^2)}{L(f_i^2 + \frac{\eta n}{m} w_i^2)^2}} \\ & + \frac{2\sigma}{1-\alpha'}\left(\sqrt{\log 2(n-t)} + C\right)\left(1 + \max_{1 \leq i \leq t} \frac{r_i^2}{f_i^2 + \frac{\eta n}{m} w_i^2}\right) \end{aligned}$$

for some constant C. It is not hard to verify that G' decreases by adding a second pool. Further note that the behavior of the non-buggy subspace, $\operatorname{span}\{q_{t+1}, \dots, q_p\}$, is not involved in any conditions or conclusions. Thus, our key observation is that the theoretical results for support recovery consistency only rely on the addition of second-pool points in buggy directions.

Example 2 *(Random design)* Consider a random design setting where the rows of X and $\widetilde{X}$ are drawn from a common sub-Gaussian distribution with covariance Σ. The conditions in Assumptions 1–3 are relaxed in the presence of a second data pool when n and m are large compared to p: First, $b'_{\min}$ increases by adding a second pool. Second, $\alpha' \approx \frac{\|X_{T^c}\Sigma^{-1}X_T\|_\infty}{n-t+\eta n}$, so the mutual incoherence parameter also decreases by adding a second pool. Third,

$$G' \approx \frac{2\sigma\sqrt{\log t}}{b'_{\min}} + \frac{2\sigma}{1-\alpha'}\max\left\{1, \sqrt{\frac{\eta n}{mL}}\right\}\left\|(I_{t \times t} - \frac{X_T \Sigma^{-1} X_T^\top}{n + \eta n})^{-1}\right\|_\infty,$$

where X_T and X_{T^c} represent the submatrices of X with rows indexed by T and T^c, respectively. Note that the one-pool case corresponds to $\eta = 0$ and

$\left\|(I_{t\times t}-\frac{X_T\Sigma^{-1}X_T^\top}{n+\eta n})^{-1}\right\|_\infty < \left\|(I_{t\times t}-\frac{X_T\Sigma^{-1}X_T^\top}{n})^{-1}\right\|_\infty$, so if we choose $\eta \leq \frac{mL}{n}$, then G' decreases by adding a second pool. Therefore, all three assumptions are relaxed by having a second pool, making it easier to achieve exact support recovery.

We also briefly discuss the three assumptions with respect to the weight parameter η: Increasing η always relaxes the eigenvalue and mutual incoherence conditions, so placing more weight on the second pool generally helps with subset support recovery. However, the same trend does not necessarily hold for exact recovery. This is because a larger value of η causes the lower bound (9) on λ to increase, resulting in a stricter gamma-min condition. Therefore, there is a tradeoff for selecting η.

4 Tuning parameter selection

A drawback of the results in the previous section is that the proper choice of tuning parameter depends on a lower bound (9) which cannot be calculated without knowledge of the unknown parameters (T, α', ϵ'). The tuning parameter λ determines how many outliers a debugger detects; if λ is large, then $\hat{\gamma}$ contains more zeros and the algorithm detects fewer bugs. A natural question arises: *In settings where the conditions for exact support recovery hold, can we select a data-dependent tuning parameter that correctly identifies all bugs?* In this section, we propose an algorithm which answers this question in the affirmative.

4.1 Algorithm and theoretical guarantees

Our tuning parameter selection algorithm is summarized in Algorithm 1, which searches through a range of parameter values for λ, starting from a large value λ_u and then halving the parameter on each successive step until a stopping criterion is met. The intuition is as follows: First, let λ^* be the right-hand expression of inequality (9). Suppose that for any value in $I=[\lambda^*, 2\lambda^*]$, support recovery holds. Then given $\lambda_u > \lambda^*$, the geometric series $\Lambda = \left\{\lambda_u, \frac{\lambda_u}{2}, \frac{\lambda_u}{4}, \dots\right\}$ must contain at least one correct parameter for exact support recovery since $\Lambda \cap I \neq \emptyset$, guaranteeing that the algorithm stops. As for the stopping criterion, let X_S denote the submatrix of X with rows indexed by S for $T^c \subseteq S \subseteq [n]$. We have $P^\perp_{X_S} \xrightarrow{|S|\to\infty} \left(1-\frac{p}{|S|}\right)I$ under some mild assumptions on X, in which case $P^\perp_{X_S} y_S \to \left(1-\frac{p}{|S|}\right)(\gamma^*_S+\epsilon_S)$. When λ is large and the conditions hold for subset support recovery but not exact recovery, we have $S \cap T \neq \emptyset$, so

$$\min |P^\perp_{X_S} y_S| \geq \left(1-\frac{p}{|S|}\right)\left(\min|\gamma^*_T| - \max_{i\in[n]}|\epsilon_i|\right).$$

In contrast, when $S = T^c$, we have

$$\min |P^\perp_{X_S} y_S| \leq \left(1-\frac{p}{|S|}\right)\max_{i\in[n]}|\epsilon_i|.$$

When $\min|\gamma^*_T|$ is large enough, the task then reduces to choosing a proper threshold to distinguish the error $|\epsilon_{T^c}|$ from the bug signal $|\gamma^*_T|$, which occurs when the threshold is chosen between $\max_i|\epsilon_i|$ and $\min_{i\in T}|\gamma^*_i| - \max_i|\epsilon_i|$.

Algorithm 1 Regularizer selection

```
Input: $\lambda_u, \bar{c}$
Output: $\hat{\lambda}^k$
$C = 1, k = 1, \hat{\lambda}^k = \lambda_u$.
while $C = 1$ do
    $\widehat{\gamma}^k \in \arg\min_{\gamma \in \mathbb{R}^n} \left\{ \frac{1}{2n} \|P_{X'}^{\perp} y' - P_{X'}^{\perp} M_{[n]} \gamma\|_2^2 + \hat{\lambda}^k \|\gamma\|_1 \right\}$.
    Let $X^{(k)}, y^{(k)}$ consist of $x_i, y_i$ such that $i \notin \mathrm{supp}(\widehat{\gamma}^k)$. Let $l^{(k)}$ be the length of $y^{(k)}$.
    $\widehat{\sigma} = \frac{l^{(k)}}{l^{(k)} - p} \cdot \mathrm{median}\left( \left| P_{X^{(k)}}^{\perp} y^{(k)} \right| \right)$.
    $C = 0$ if $\|P_{X^{(k)}}^{\perp} y^{(k)}\|_\infty \le \frac{5}{2} \bar{c}^{-1} \sqrt{\log 2n}\, \widehat{\sigma}$.
    $k = k + 1, \hat{\lambda}^k = \hat{\lambda}^{k-1}/2$.
end while
```

With the above intuition, we now state our main result concerning exact recovery guarantees for our algorithm. Recall that the ϵ_i's are sub-Gaussian with parameter σ^2.

Let $c_t := \frac{t}{n} < \frac{1}{2}$ denote the fraction of outliers. We assume knowledge of a constant $\bar{c}$ that satisfies $c_t + \mathbb{P}[|\epsilon_i| \le \bar{c}\sigma] < \frac{1}{2}$. Note that a priori knowledge of $\bar{c}$ is a less stringent assumption than knowing σ, since we can always choose $\bar{c}$ to be close to zero. For instance, if we know the ϵ_i's are Gaussian, we can choose $\bar{c} < \mathrm{erf}^{-1}(\frac{1}{2} - c_t)$; in practice, we can usually estimate c_t to be less than $\frac{1}{3}$, so we can take $\bar{c} = \mathrm{erf}^{-1}(\frac{1}{6})$. As shown later, the tradeoff is that having a larger value of $\bar{c}$ provides the desired guarantees under weaker requirements on the lower bound of $\min_{i \in T} |\gamma_i^*|$. Hence, if we know more about the shape of the error distribution, we can be guaranteed to detect bugs of smaller magnitudes. We will make the following assumption on the design matrix:

Assumption 4 There exists a $p \times p$ positive definite matrix Σ, with bounded minimum and maximum eigenvalues, such that for all $X^{(k)}$ appearing in the while loop of Algorithm 1, we have

$$\begin{aligned} \left\| \frac{X^{(k)} \Sigma^{-1} X^{(k)\top}}{p} - I \right\|_{\max} &\le c \max\left\{ \sqrt{\frac{\log l^{(k)}}{p}}, \frac{\log l^{(k)}}{p} \right\}, \\ \left\| \frac{X^{(k)\top} X^{(k)}}{l^{(k)}} - \Sigma \right\|_2 &\le \frac{\lambda_{\min}(\Sigma)}{2}, \end{aligned} \tag{10}$$

where $l^{(k)}$ is the number of rows of the matrix $X^{(k)}$ and c is a universal constant.

This assumption is a type of concentration result, which we will show holds w.h.p. in some random design settings in the following proposition:

Proposition 2 *Suppose the x_i's are i.i.d. and satisfy any of the following additional conditions:*

(a) *the x_i's are Gaussian and the spectral norm of the covariance matrix is bounded;*

(b) *the x_i's are sub-Gaussian with mean zero and independent coordinates, and the spectral norm of the covariance matrix is bounded; or*

(c) *the x_i's satisfy the convex concentration property.*

Then Assumption 4 holds with probability at least $1 - O(n^{-1})$.

The Σ matrix can be chosen as the covariance of X. In fact, Assumption 4 shows that $P^{\perp}_{X^{(k)}}$ is approximately a scalar matrix. We now introduce some additional notation: For $\nu > 0$, define c_ν and C_ν such that $\nu = \mathbb{P}[|\epsilon_i| \leq c_\nu \sigma]$ and $\nu = \mathbb{P}[|\epsilon_i| \geq C_\nu \sigma]$. We write $G'(\lambda)$ to denote the function of λ in the right-hand expression of inequality (8). Proofs of the theoretical results in this section are provided in Appendix D.

Theorem 3 *Assume ν is a constant satisfying $\nu + c_t < \frac{1}{2}$. Suppose Assumption 4, the minimum eigenvalue condition, and the mutual incoherence condition hold. If*

$$n \geq \max\left\{ \left[\frac{24}{c_\nu}\right]^{\frac{1}{c_n}}, \left[\frac{C\log 2n}{1-c_t}(p^2 + \log^2 n)\right]^{\frac{1}{1-2c_n}} \right\}, \tag{11}$$

where C is an absolute constant, and

$$\begin{aligned} \min_{i\in T} |\gamma_i^*| &> \max\left\{ G'(2\lambda^*), 4\sqrt{\log(2n)}\sigma, \frac{5}{4}\sqrt{\log(2n)}\frac{c_\nu + 5C_\nu}{\bar{c}}\sigma \right\}, \\ \|\gamma^*\|_\infty &\leq \frac{\sqrt{C}c_\nu}{16\sqrt{2}}\sqrt{1-c_t}\sqrt{\log 2n}\,\frac{n^{1/2+c_n}}{t}\sigma, \end{aligned} \tag{12}$$

for some $c_n \in (0, \frac{1}{2})$, then Algorithm 1 with inputs $\bar{c} < c_\nu$ and $\lambda_u \geq \lambda^*$ will return a feasible $\widehat{\lambda}$ in at most $\log_2\left(\frac{\lambda_u}{\lambda^*}\right)$ iterations such that the Lasso estimator $\widehat{\gamma}$ based on $\widehat{\lambda}$ satisfies $\mathrm{supp}(\widehat{\gamma}) = \mathrm{supp}(\gamma^*)$, *with probability at least*

$$1 - \frac{3\log_2\left(\frac{\lambda_u}{\lambda^*}\right)}{n-t} - 2\log_2\left(\frac{\lambda_u}{\lambda^*}\right)\exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right).$$

Theorem 3 guarantees exact support recovery for the output of Algorithm 1 without knowing σ. Note that compared to the gamma-min condition (8) with $\lambda = \lambda^*$, the required lower bound (12) only differs by a constant factor. In fact, the constant 2 inside $G'(2\lambda^*)$ can be replaced by any constant $c > 1$, but Algorithm 1 will then update $\hat{\lambda}^k = \hat{\lambda}^{k-1}/c$ and require $\log_c\left(\frac{\lambda_u}{\lambda^*}\right)$ iterations. Further note that larger values of c_t translate into a larger sample size requirement, as $n = \Omega\left(\frac{1}{1-c_t}\right)$ for c_n being close to 0. A limitation of the theorem is the upper bound on $\|\gamma^*\|_\infty$, where t needs to be smaller than n in a nonlinear relationship. Also, n is required to be $\Omega(p^2)$. These two conditions are imposed in our analysis in order to guarantee that $P^{\perp}_{X_S} y_S \to \left(1 - \frac{p}{|S|}\right)(\gamma_S^* + \epsilon_S)$. We now present a result indicating a practical choice of λ_u:

Corollary 1 *Define*

$$\lambda(\sigma) := \frac{8\max\{1, \sqrt{\frac{\eta n}{Lm}}\}}{1-\alpha'}\sqrt{\log 2(n-t)}\frac{\|P^{\perp}_{X,T^c}\|_2}{n} \cdot c\sigma.$$

Suppose Assumption 4, the minimum eigenvalue condition, and the mutual incoherence condition hold. Also assume conditions (11) and (12) hold when replacing λ^* by $\lambda(\sigma)$. Taking the input $\lambda_u = \frac{2\|M_{[n]}P^{\perp}_{X'}y'\|_\infty}{n}$, Algorithm 1 outputs a parameter $\widehat{\lambda}$ in $O(\log n)$ iterations which provides exact support recovery, with probability at least

$$1 - \frac{4\left(c' \log_2 n + \max\left\{0, \frac{1}{2}\log_2 \frac{\eta n}{mL}\right\}\right)}{n-t} \\ - 2\left(c' \log_2 n + \frac{1}{2}\max\left\{0, \log_2 \frac{\eta n}{mL}\right\}\right)e^{-2\left(\frac{1}{2}-c_t-\nu\right)^2 n}.$$

Note that λ_u can be calculated using the observed data set. Further note that the algorithm is guaranteed to stop after $O(\log n)$ iterations, meaning it is sufficient to test a relatively small number of candidate parameters in order to achieve exact recovery.

5 Strategy for second pool design

We now turn to the problem of designing a clean data pool. In the preceding sections, we have discussed how a second data pool can aid exact recovery under sub-Gaussian designs. In practice, however, it is often unreasonable to assume that new points can be drawn from an entirely different distribution. Specifically, recall the movie rating example discussed in Sect. 1: The expert can only rate movies in the movie pool, say $\{x_i\}_{i=1}^n$, whereas an arbitrarily designed $\widetilde{x}$, e.g., $\widetilde{x} = x_1/2$, is unlikely to correspond to an existing movie. Thus, we will focus on devising a debugging strategy where the debugger is allowed to choose points for the second pool which have the same covariates as points in the first pool.

In particular, we consider this problem in the "worst" case: suppose a bug generator can generate any $\gamma^* \in \Gamma := \{\gamma \in \mathbb{R}^n : \mathrm{supp}(\gamma)| \leq t\}$ and add it to the correct labels $X\beta^*$. We will also suppose the bug generator knows the debugger's strategy. The debugger attempts to add a second data pool which will ensure that all bugs are detected regardless of the choice of γ^*. Our theory is limited to the noiseless case, where $y = X\beta^* + \gamma^*$ and $\widetilde{y} = \widetilde{X}\beta^*$; the noisy case is studied empirically in Sect. 6.3.3.

5.1 Preliminary analysis

We denote the debugger's choice by $\widetilde{x}_i = X^\top e_{\nu(i)}$, for $i \in [m]$, where $e_{\nu(i)} \in \mathbb{R}^n$ is a canonical vector and $\nu : [m] \to [n]$ is injective. In matrix form, we write $\widetilde{X} = X_D$, where $D \subseteq [n]$ represents the indices selected by the debugger. Assume $m < p$, so the debugger cannot simply use the clean pool to obtain a good estimate of β. In the noiseless case, we can write the debugging algorithm as follows:

$$\begin{aligned} &\min_{\beta\in\mathbb{R}^p, \gamma\in\mathbb{R}^n} \|\gamma\|_1 \\ &\text{subject to } y = X\beta + \gamma, \ \widetilde{y} = \widetilde{X}\beta. \end{aligned} \tag{13}$$

Similar to Proposition 1, given a γ, we can pick β to satisfy the constraints, specifically $\beta = \left(X^\top X + \widetilde{X}^\top \widetilde{X}\right)^{-1} \left(X^\top (y-\gamma) + \widetilde{X}^\top \widetilde{y}\right)$. Eliminating β, we obtain the optimization problem

$$\begin{aligned} &\min_{\gamma \in \mathbb{R}^n} \&\|\gamma\|_1 \\ &\text{subject to } \begin{bmatrix} y \\ \widetilde{y} \end{bmatrix} = \begin{bmatrix} X \\ \widetilde{X} \end{bmatrix} \left(X^\top X + \widetilde{X}^\top \widetilde{X}\right)^{-1} \left(X^\top (y-\gamma) + \widetilde{X}^\top \widetilde{y}\right) + \begin{bmatrix} \gamma \\ \mathbf{0} \end{bmatrix}. \end{aligned} \tag{14}$$

Before presenting our results for support recovery, we introduce some definitions. Define the cone set $\mathbb{C}(K)$ for some subset $K \subseteq [n]$ and $|K| = t$:

$$\mathbb{C}(K) := \left\{ \Delta \in \mathbb{R}^n : \|\Delta_{K^c}\|_1 \leq \|\Delta_K\|_1 \right\}. \tag{15}$$

Further let $\mathbb{C}^A = \cup_{K \subseteq [n], |K|=t} \mathbb{C}(K)$, and define

$$\bar{P}(D) = \begin{bmatrix} I - X\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top \\ X_D\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top \end{bmatrix}.$$

Theorem 4 *Suppose*

$$Null(\bar{P}(D)) \cap \mathbb{C}^A = \{\mathbf{0}\}. \tag{16}$$

Then a debugger who queries the points indexed by D cannot be beaten by any bug generator who introduces at most t bugs.

Theorem 4 suggests that Eq. (16) is a sufficient condition for support recovery for an omnipotent bug generator who knows the subset D. As a debugger, the consequent goal is to find such a subset D which makes Eq. (16) true. Whether such a D exists and how to find it will be discussed in Sect. 5.2.

Remark 1 When $m = n$, we can verify that $Null(\bar{P}(D)) = \{\mathbf{0}\}$, which implies that Eq. (16) always holds. Indeed, in this case, we can simply take $\widetilde{X} = X$ and solve for β^* explicitly to recover γ^*.

Remark 2 As stated in Theorem 4, Eq. (16) is a sufficient condition for support recovery. In fact, it is an if-and-only-if condition for signed support recovery: When Eq. (16) holds, $\text{sign}(\widehat{\gamma}) = \text{sign}(\gamma^*)$; and when it does not hold, the bug generator can find a γ^* with $\text{supp}(\gamma^*) \leq t$ such that $\text{sign}(\widehat{\gamma}) \neq \text{sign}(\gamma^*)$.

Remark 3 We can also write $Null(\bar{P}(D))$ as

$$\{u \in \mathbb{R}^n \mid \exists v \in \mathbb{R}^p, \text{ s.t. } u = Xv, X_D v = 0\}.$$

Let $\widehat{\beta} = \beta^* + v$ for some vector $v \in \mathbb{R}^p$. From the constraint-based algorithm, we obtain

$$\begin{aligned} y_T &= X_T(\beta^* + v) + \widehat{\gamma}_T, \\ y_{T^c} &= X_{T^c}(\beta^* + v) + \widehat{\gamma}_{T^c},\ y_D = X_D(\beta^* + v), \end{aligned}$$

which implies that $\widehat{\gamma}_T = \gamma^*_T - X_T v$ and $\widehat{\gamma}_{T^c} = -X_{T^c} v$, $X_D v = 0$. Let $u = Xv$. Then we obtain $\widehat{\gamma} = \gamma^* - u$. As can be seen, Eq. (16) requires that $u = \mathbf{0}$, which essentially implies $\widehat{\gamma} = \gamma^*$, and thus $\operatorname{supp}(\widehat{\gamma}) = \operatorname{supp}(\gamma^*)$.

5.2 Optimal debugger via MILP

The above analysis is also useful in practice for providing a method for designing $\widetilde{X}$. Consider the following optimization problem:

$$\max_{K \subseteq [n], |K| \le t, u \in \mathbb{R}^n, v \in \mathbb{R}^d} \|u_K\|_1 - \|u_{K^c}\|_1, \tag{17a}$$

$$\text{subject to } u = Xv, X_D v = 0, \|u\|_\infty \le 1. \tag{17b}$$

By Theorem 4 and Remark 3, we immediately conclude that if the problem (17) has the unique solution $(u, v) = (\mathbf{0}, \mathbf{0})$, then a debugger who queries the points indexed by D cannot be beaten by a bug generator who introduces at most t bugs.

Based on this argument, we can construct a bilevel optimization problem for the debugger to solve by further minimizing the objective (17a) with respect to $D \subseteq [n]$ such that $|D| \le m$. The optimization problem can then be transformed into a minimax MILP:

$$\begin{aligned}
&\min_{\xi \in \{0,1\}^n} \quad \max_{\substack{a, a^+, a^- \in \mathbb{R}^n, \\ u, u^+, u^- \in \mathbb{R}^n, v \in \mathbb{R}^d, \\ z, w \in \{0,1\}^n}} \quad \sum_{j=1}^n a_j^+ - a_j^-, \\
&\text{subject to } \Big\{ u = Xv, u = u^+ - u^-, u^+, u^- \ge 0, \\
&\qquad a = u^+ + u^-, u^+ \le z,\ u^- \le (\mathbb{1}_n - z), \\
&\qquad \sum_{i=1}^n w_i \le t, a^+ \le Mw,\ a^- \le M(\mathbb{1}_n - w), \\
&\qquad a = a^+ + a^-, a^+ \ge 0, a^- \ge 0, \\
&\qquad \sum_{i=1}^n \xi_i \le m, u \le (\mathbb{1}_n - \xi), u \ge -(\mathbb{1}_n - \xi). \Big\}
\end{aligned} \tag{18}$$

Theorem 5 (MILP for debugging) *If the optimization problem* (18) *has the unique solution* $(u, v) = (\mathbf{0}, \mathbf{0})$, *then the debugger can add m points indexed by* $D = \operatorname{supp}(\xi)$ *to achieve support recovery.*

Remark 4 For more information on efficient algorithms for optimizing minimax MILPs, we refer the reader to the references Tang et al. (2016), Xu and Wang (2014), Zeng and An (2014).

6 Experiments

In this section, we empirically validate our Lasso-based debugging method for support recovery. The section is organized as follows:

- Section 6.1, corresponding to Sect. 3, contains a number of experiments which investigate the performance of our proposed debugging formulation.
- Section 6.2, corresponding to Sect. 4, studies the proposed tuning parameter selection procedure.
- Section 6.3 studies the Lasso-based debugging method with a clean data pool, including the proposed MILP algorithm from Sect. 5.

We also compare our proposed method to alternative methods motivated by existing literature.

We begin with an outline of the experimental settings used in most of our experiments:

S1 Generate the feature design matrix $X \in \mathbb{R}^{n \times p}$ by sampling each row i.i.d. from $\mathcal{N}(\mathbf{0}_p, I_{p \times p})$.

S2 Generate $\beta^* \in \mathbb{R}^p$, where each entry β_i^* is drawn i.i.d. from $Unif(-1, 1)$.

S3 Generate $\epsilon \in \mathbb{R}^n$, where each entry ϵ_i is drawn i.i.d. from $\mathcal{N}(0, \sigma^2)$.

S4 Generate the bug vector $\gamma^* \in \mathbb{R}^n$, where we draw $\gamma_i^* = (10\sqrt{\log(2n)}\sigma + Unif(0, 10)) \cdot Bernoulli(\pm 1, 0.5)$ for $i \in [t]$ and take $\gamma_i^* = 0$ for the remaining positions.

S5 Generate the labels by $y = X\beta^* + \epsilon + \gamma^*$.

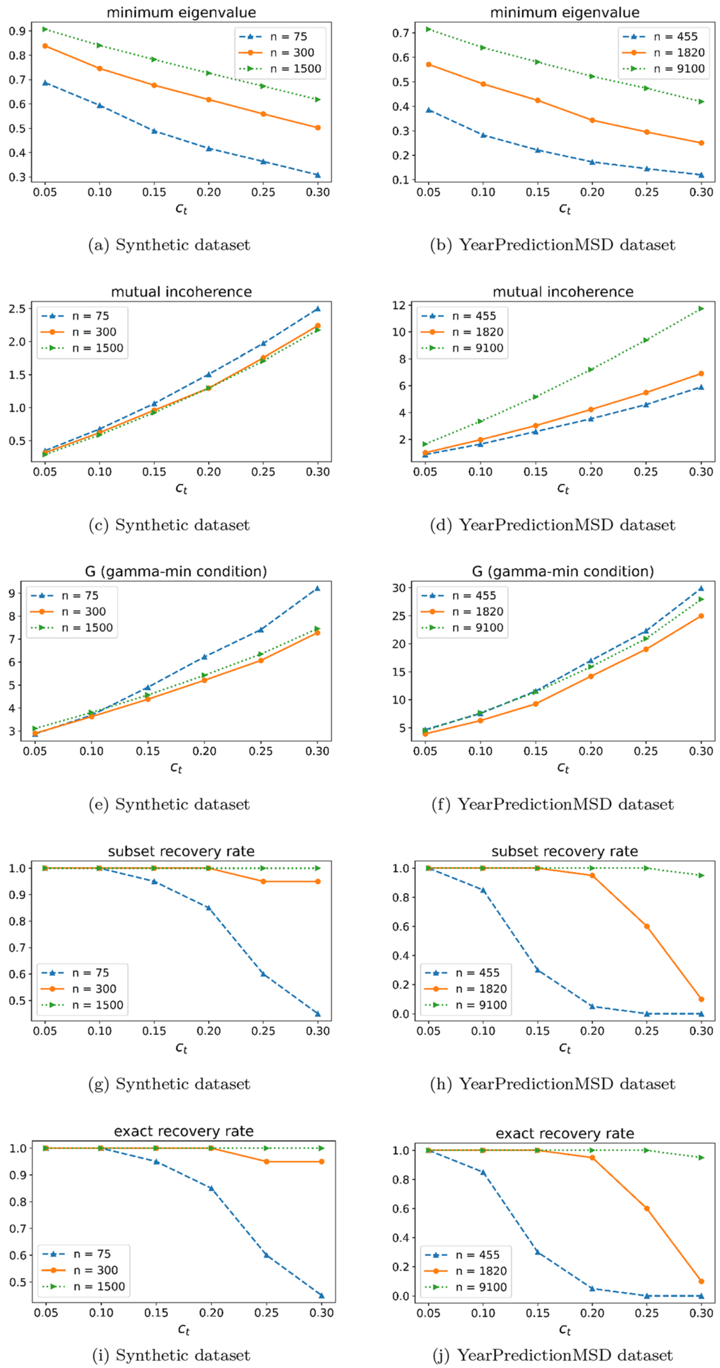
These five steps produce a synthetic dataset (X, y); we will specify the particular parameters (n, p, t, σ) in each task. If we use a real dataset, the first step changes to [S1']:

S1' Given the whole data pool X_{real}, uniformly sample n data points from it to construct X.

In the plot legends, we will refer to our Lasso-based debugging method as "debugging." We may also invoke a postprocessing step on top of debugging, called "debugging + postprocess," which first runs the Lasso optimization algorithm to obtain $\hat{\gamma}$ and an estimated support set $\hat{T}$, then removes the points $(X_{\hat{T},\cdot}, y_{\hat{T}})$ and runs ordinary least squares on the remaining points to obtain $\widehat{\beta}$.

6.1 Support recovery

In this section, we design two experiments. The first experiment investigates the influence of the fraction of bugs $c_t := \frac{t}{n}$ on the three assumptions imposed in our theory and the resulting recovery rates. We will vary the design of X using different datasets. The second experiment compares debugging with four alternative regression methods, using the precision-recall metric. Note that we will take the tuning parameter $\lambda = 2\frac{\sqrt{\log 2(n-t)}}{n}$ for these experiments, since the other outlier detection methods we use for comparison do not propose a way to perform parameter tuning. We will explore the performance of the proposed algorithm for parameter selection in the next subsection.



(a) Synthetic dataset

(b) YearPredictionMSD dataset

(c) Synthetic dataset

(d) YearPredictionMSD dataset

(e) Synthetic dataset

(f) YearPredictionMSD dataset

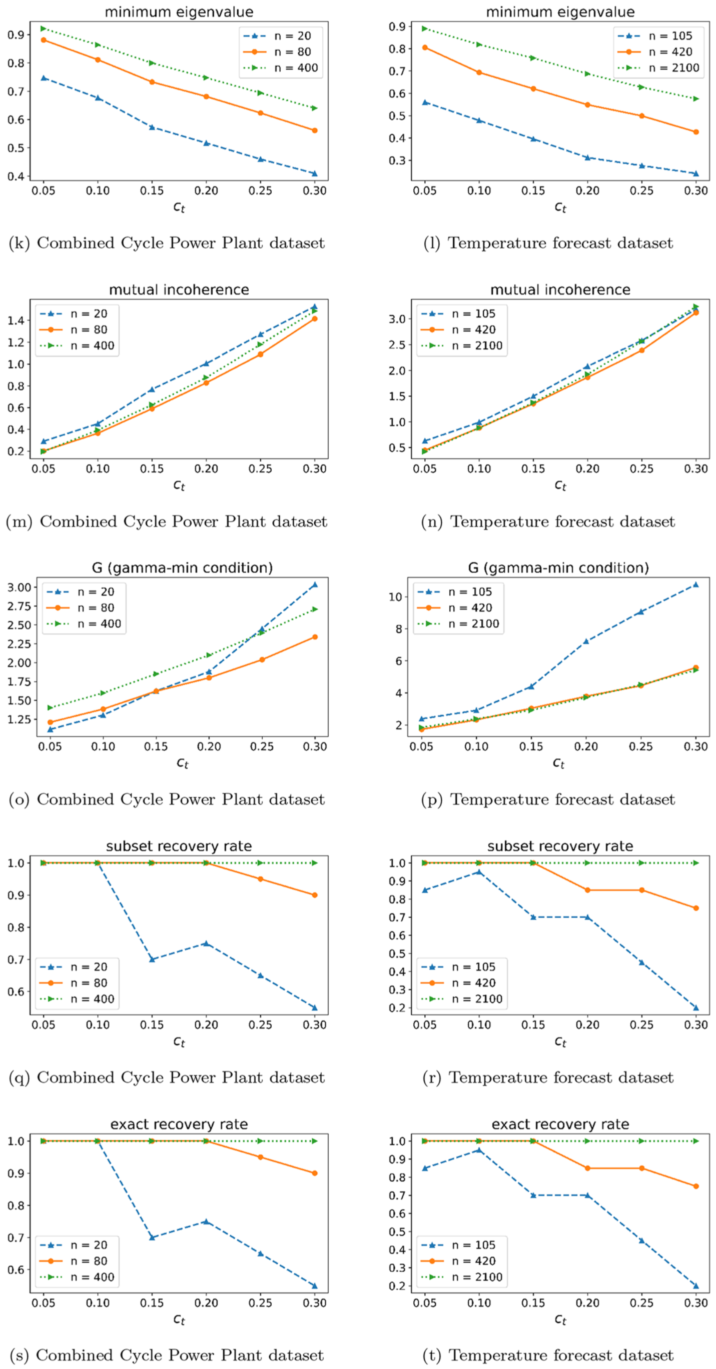
(g) Synthetic dataset

(h) YearPredictionMSD dataset

(i) Synthetic dataset

(j) YearPredictionMSD dataset

Fig. 1 Five Measurements on Four Datasets. Three different n's are of values $5p$, $20p$, and $100p$. The variance σ is set to 0.1. The tuning parameter is set to $\lambda = 2\frac{\sqrt{\log 2(n-t)}}{n}$. Each dot is an average value of 20 random trials



(k) Combined Cycle Power Plant dataset

(l) Temperature forecast dataset

(m) Combined Cycle Power Plant dataset

(n) Temperature forecast dataset

(o) Combined Cycle Power Plant dataset

(p) Temperature forecast dataset

(q) Combined Cycle Power Plant dataset

(r) Temperature forecast dataset

(s) Combined Cycle Power Plant dataset

(t) Temperature forecast dataset

Fig. 1 (continued)

6.1.1 Number of bugs versus different measurements

Our first experiment involves four different datasets with different values of n and c_t. We track the performance of the three assumptions (Assumptions 1–3) and the subset/exact recovery rates, which measure the fraction of experiments which result in subset/exact recovery. The first dataset is generated using the synthetic mechanism described at the beginning of Sect. 6, with $p = 15$. The other three datasets are chosen from the UCI Machine Learning Repository: Combined Cycle Power Plant,[1] temperature forecast,[2] and YearPredictionMSD.[3] They are all associated to regression tasks, with varying feature dimensions (4, 21, and 90, respectively). In the temperature forecast dataset, we remove the attribute of station and date from the original dataset, since they are discrete objects. For each of the UCI datasets, after randomly picking n data points from the entire data pool, we normalize the subsampled dataset according to $X_{\cdot,j} = \frac{X_{\cdot,j} - \frac{1}{n}\sum_{i\in[n]} X_{i,j}}{std[X_{\cdot,j}]}$, where std represents the standard deviation.

The results are displayed in Fig. 1. For the minimum eigenvalue assumption, a key observation from all datasets is that the minimum eigenvalue becomes larger (improves) as n increases, and becomes smaller as c_t increases. For the mutual incoherence assumption, the synthetic dataset satisfies the condition with less than 15% outliers. The Combined Cycle Power Plant dataset has mutual incoherence close to 1 when c_t is approximately 20%-25%, and the mutual incoherence condition of the YearPredictionMSD dataset approaches 1 when c_t is approximately 5%. Therefore, we see that the validity of the assumption highly depends on the design of X. For the gamma-min condition, as c_t increases, we need more obvious (larger $\min_i |\gamma_i^*|$) outliers. Finally, with larger n and smaller c_t, the subset/exact recovery rate improves.

6.1.2 Effectiveness for recovery

The second experiment compares our debugging method to other proposed methods in the robust statistics literature. We compare our method with the Fast LTS (Rousseeuw and Van Driessen 2006), E-lasso (Nguyen and Tran 2013), Simplified Θ-IPOD (She and Owen 2011), and Least Squares methods. E-lasso is similar to our formulation, except it includes an additional penalty with β. The Simplified Θ-IPOD method iteratively uses hard thresholding to eliminate the influence of outliers. For the experimental setup, we generate synthetic data with $n = 2000, t = 200, p = 15$, and $\sigma = 0.1$, but replace step [S4] by one of the following mechanisms for generating γ^*:

1. We generate $\gamma_i^*, i \in T$ by $Bernoulli(\pm 1, 0.5) \cdot (10\sqrt{\log(2n)}\sigma + Unif(0, 10))$.
2. We generate β' elementwise from $Unif(-10, 10)$ and take $\gamma_i^* = x_i^\top(\beta' - \beta^*), i \in T$.

The first adversary is random, whereas the second adversary aims to attack the data by inducing the learner to fit another hyperplane. The precision/recall for Fast LTS and Least Squares are calculated by running the method once and applying various thresholds to clip $\hat{\gamma}$. For the other three methods, we apply different tuning parameters, compute precision/recall for each result, and finally combine them to plot a macro precision-recall curve.

In the left panel of Fig. 2, Least Squares and Fast LTS reach perfect AUC, while the other three methods have slightly lower scores. In the right panel of Fig. 2, we see that

[1] http://archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant.

[2] http://archive.ics.uci.edu/ml/datasets/Bias+correction+of+numerical+prediction+model+temperature+forecast.

[3] http://archive.ics.uci.edu/ml/datasets/YearPredictionMSD.

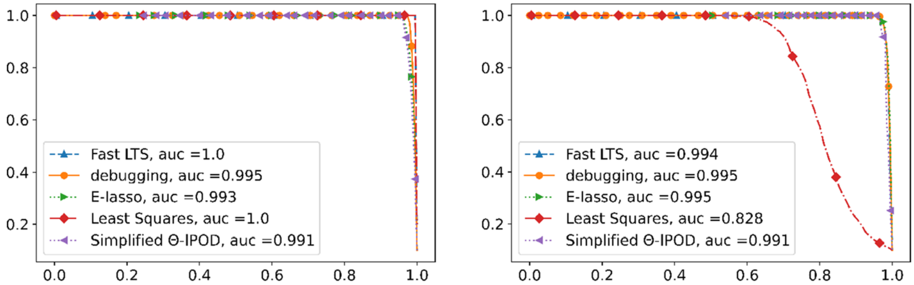


Fig. 2 Precision Recall Curves over Different Regression Methods. The two plots correspond to the two settings described in the text for generating γ^*. To better view the curves, we only show the dots for every c positions, where c is an interger and different for different methods

debugging, E-lasso, and Fast LTS perform comparably well, and slightly better than Simplified Θ-IPOD. Not surprisingly, Least Squares performs somewhat worse, since it is not a robust procedure.

6.2 Tuning parameter selection

We now present two experimental designs for tuning parameter selection. The first experiment runs Algorithm 1 for both one- and two-pool cases. We will present the recovery rates for a range of n's and c_t's, showing the effectiveness of our algorithm in a variety of situations. The second experiment compares Algorithm 1 in one- and two-pool cases to cross-validation, which is a popular alternative for parameter tuning. Our results indicate that Algorithm 1 outperforms cross-validation in terms of support recovery performance.

We begin by describing the method used to generate the second data pool. Given the first data pool (X, y) and the ground-truth parameters (β^*, σ), we describe two pipelines to generate the second pool. The first pipeline checks m random points of the first pool, with steps [T1-T3]:

T1 Select m points uniformly at random from the first pool to construct $\widetilde{X}$ for the second pool.
T2 Generate $\widetilde{\epsilon} \in \mathbb{R}^m$, where each entry $\widetilde{\epsilon}_i$ is drawn i.i.d. from $\mathcal{N}(0, \sigma^2/L)$.
T3 Generate the labels by $\widetilde{y} = \widetilde{X}\beta^* + \widetilde{\epsilon}$.

When the debugger is able to query features of clean points from a distribution $\mathcal{P}_X$, we can use a second pipeline, where [T1] is replaced by [T1']:

T1' Independently draw m points from $\mathcal{P}_X$ to construct $\widetilde{X}$.

6.2.1 Verification of Algorithm 1

We use the default procedure for generating the synthetic dataset, with parameters $p = 15$, $\sigma = 0.1$, and $t = c_t n$, where c_t ranges from 0.05 to 0.4 in increments of 0.05. In all cases, we input $\bar{c} = 0.2$ and $\lambda_u = \frac{2\|P_X^\perp y\|_\infty}{n}$ in Algorithm 1.

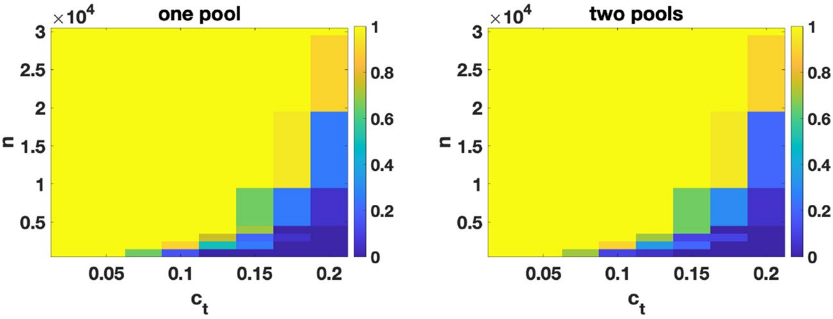


Fig. 3 Exact Recovery Rate over 20 Trials. The recovery rate is shown in different cases varying by fraction of outliers c_t and n. The left subfigure is for one-pool case and the right subfigure is for two-pool case. We set $m = 100, L = 5$ for the second pool

Figure 3 displays the results for $n \in \{1, 2, 3, 4, 5, 10, 20, 30\} \cdot 10^3$. First, we see that Algorithm 1 achieves exact support recovery in all 20 trials in the yellow area. Second, the exact recovery rate increases with increasing n and decreasing c_t, showing that the algorithm is particularly useful for large-scale data sets. This trend can also be seen from the requirement on n imposed in Theorem 3. In particular, we see that the contour curve for the exact recovery rate matches the curve of $\left(1 - c_t\right)^{-\frac{1}{1-2c_n}}$ for some constant $c_n \in (0, \frac{1}{2})$. However, a downside of Algorithm 1 is that it does not fully take advantage of the second pool in the two-pool case, as the left panel and the right panel display similar results.

6.2.2 Effectiveness of tuning parameter selection

We now compare our method for tuning parameter selection to cross-validation. We also use the postprocessing step described at the beginning of the section. Four measurements are presented, including two recovery rates, the ℓ_2-error of $\widehat{\beta}$, and the runtime. In both the one- and two-pool cases, we use our default methods for generating synthetic data, and we set $\bar{c} = 0.2$ for all the experiments.

The cross-validation method for the one-pool case splits the dataset into training and testing datasets with the ratio of 8:2, then selects λ with the smallest test error, $\|X_{test}\widehat{\beta} - y_{test}\|_2$. The procedure for the two-pool case is to run the Lasso-based debugging method with a list of candidate λ's and test it on the second pool. Finally, we select the λ value with the smallest test error, $\|\widetilde{X}\widehat{\beta} - \widetilde{y}\|_2$. We use 15 candidate values for λ, spaced evenly on a log scale between 10^{-6} and $\lambda_u = \frac{2\|P_X^{\perp} y\|_\infty}{n}$.

Figure 4 compares the results in the one-pool case. We note that cross-validation does not perform very well for all the measurements except $\|\widehat{\beta} - \beta^*\|_2$. Specifically, it does not work at all for subset support recovery, since cross-validation tends to choose very small λ values. For the ℓ_2-error, we see that for small values of c_t, our algorithm can select a suitable choice of λ, so that after removing outliers, we can fit the remaining points very well. This is why the debugging + postprecessing methods gives the lowest error. As c_t increases, our debugging method shows poorer performance in terms of support recovery, resulting in larger ℓ_2-error for $\widehat{\beta}$. Although cross-validation seems to perform well, carefully designed adversaries may still destroy the good performance of cross-validation, since its test dataset could be made to contain numerous buggy points.

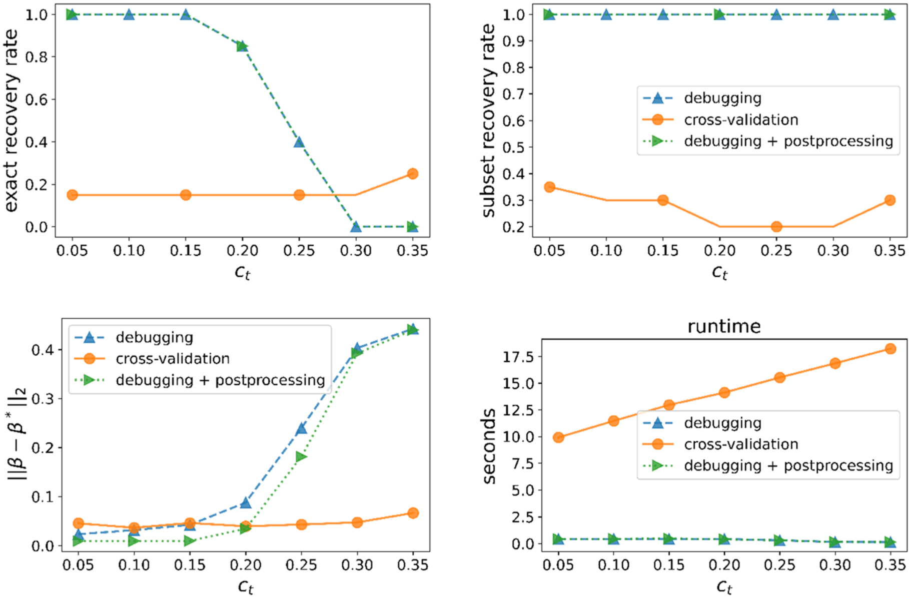


Fig. 4 Effectiveness of Tuning Parameter Selection (One Pool). Each dot is the average result of 20 random trials. We set $n = 2000, p = 15$, and $\sigma = 0.1$

Figure 5 displays the results for the two-pool experiments, which are qualitatively similar to the results of the one-pool experiments. We emphasize that our method works well for support recovery; furthermore, the methods exhibit comparable performance in terms of the ℓ_2-error. The slightly larger error of our debugging method can be attributed to the bias which arises from using an ℓ_1-norm instead of an ℓ_0-norm.

6.3 Experiments with clean points

We now focus on debugging methods involving a second clean pool. We have three experimental designs: First, we study the influence of m on support recovery. Second, we compare debugging with alternative methods suggested in the literature. Third, we study the performance of our proposed MILP debugger, where we compare it to three other simple strategies. Different strategies for selecting clean points correspond to changing step [T1] in the setup described above.

6.3.1 Number of clean points versus exact recovery

In this subsection, we present two experiments involving synthetic and YearPredictionMSD datasets, respectively, to see how m affects the exact recovery rate. Recall that the pipeline for generating the first pool is described at the beginning of Sect. 6. For the second pool, we use steps [T'1, T2, T3] for the synthetic dataset, where we assume $\mathcal{P}_X$ is standard Gaussian. We take steps [T1–T3] for YearPredictionMSD to check the sample points in the first pool.

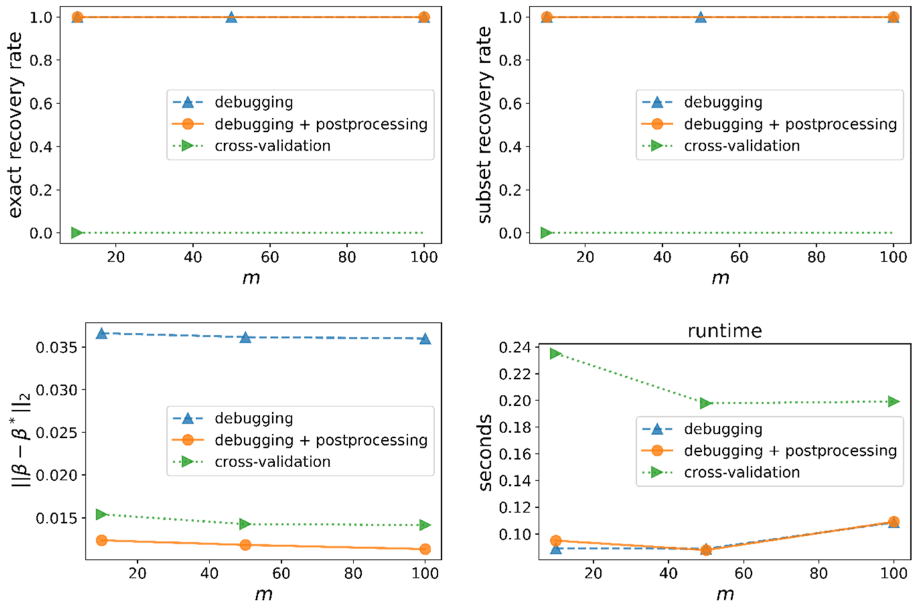


Fig. 5 Effectiveness on Tuning Parameter Selection (Two Pools). Each dot is the average result of 20 random trials. We set $n = 1000, p = 15, t = 100, L = 5$, and $\sigma = 0.1$

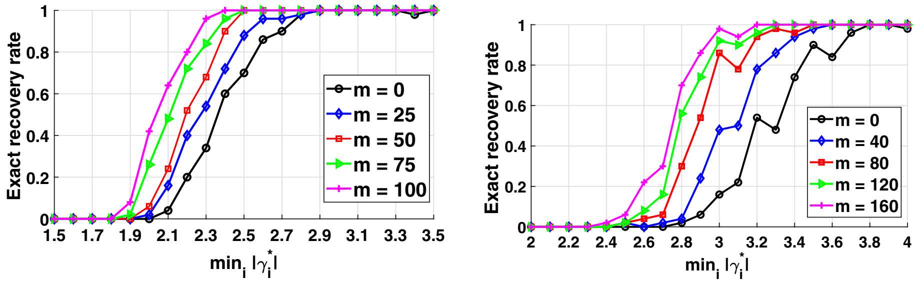


Fig. 6 Minimal Gamma versus Exact Recovery Rate on Synthetic Data. We run 50 trials for each dot and compute the average

Recall that the YearPredictionMSD dataset is designed to predict the release year of a song from audio features. The dataset consists of 515,345 songs, each with 90 audio features. Therefore, for both experiments, we set $n = 500, t = 50, p = 90, \sigma = 0.1$, and $L = 10$, and take $\lambda = 2.5\frac{\sqrt{\log(n-t)}}{n}$.

From Fig. 6, we see that the phenomena are similar for the two different design matrices. In particular, increasing the number of clean points helps with exact recovery. For instance, in the left subfigure, for $m = 0$, when $\min_i |\gamma_i^*| > 2.9$, the exact recovery rate goes to 1. For $m = 100$, the exact recovery rate goes to 1 when $\min_i |\gamma_i^*| > 2.4$. Also, the slope of the curve for larger m is sharper. Thus, adding a second pool helps relax the gamma-min condition.

6.3.2 Comparisons to methods with clean points

In this experiment, we compare the debugging method for two pools with other methods suggested by the machine learning literature. We generate synthetic data using the default first-pool setup with $n = 1000, p = 15, t = 100$, and $\sigma = 0.1$, and we run [T1–T3] to generate the second pool using different values of m. For our proposed debugging method, we use Algorithm 1 to select the tuning parameter. We compare the following methods: (1) debugging + postprocessing, (2) least squares, (3) simplified noisy neural network, and (4) semi-supervised eigvec. The least squares solution is applied using $\left\{ \begin{pmatrix} X \\ \tilde{X} \end{pmatrix}, \begin{pmatrix} y \\ \tilde{y} \end{pmatrix} \right\}$.

The simplified noisy neural network method borrows an idea from Veit et al. (2017), which is designed for image classification tasks for a datasets with noisy and clean points. This work introduced two kinds of networks and combines them together: the "Label Cleaning Network," used to correct the labels, and the "Image Classifier," which classifies images using CNN features as inputs and corrected labels as outputs. Each of them is associated with a loss, and the goal is to minimize the sum of the losses. Let $w \in \mathbb{R}$, $\beta_1 \in \mathbb{R}^d$, and $\beta_2 \in \mathbb{R}^d$ be the variables to be optimized. For our linear regression setting, we design the "Label Cleaning Network" by defining $\hat{c}_i = y_i w - x_i^\top \beta_1$ as the corrected labels for both noisy and clean datasets. Then we define the loss $\mathcal{L}_{clean} = \sum_{i \in cleanset} |\tilde{y}_i - y_i w - x_i^\top \beta_1|$. The "Image Classifier" is modified to the regression setting using predictions of $x_i^\top \beta_2$ and the squared loss. Therefore, the classification loss can be formalized as $\mathcal{L}_{classify} = \sum_{i \in cleanset} (x_i^\top \beta_2 - \tilde{y}_i)^2 + \sum_{i \in noisyset} (x_i^\top \beta_2 - \hat{c}_i)$. Together, the optimization problem becomes

$$\min_{\substack{\beta_1 \in \mathbb{R}^d, \beta_2 \in \mathbb{R}^d \\ w \in \mathbb{R}}} \sum_{i \in cleanset} \{(x_i^\top \beta_2 - \tilde{y}_i)^2 + |\tilde{y}_i - w y_i - x_i^\top \beta_1|\} + \sum_{i \in noisyset} (x_i^\top \beta_2 - w y_i - x_i^\top \beta_1)^2.$$

We use gradient descent to do the optimization, and initialize it with $w = 0$ and $\beta_1 = \beta_2 = \widehat{\beta}_{ls}$. The optimizer $\widehat{\beta}_2$ is used for further predictions. We then calculate $\widehat{\gamma} = y - X\widehat{\beta}_2$. For gradient descent, we will validate multiple step sizes and choose the one with the best performance on the squared loss of the clean pool.

The method "semi-supervised eigvec" is from Fergus et al. (2009), and is designed for the semi-supervised classification problem. It also contains an experimental setting that involves noisy and clean data. To further apply the ideas in our linear regression setting, we make the following modifications: Define the loss function as

$$J(f) = f^\top L f + \left(f - \begin{pmatrix} y \\ \tilde{y} \end{pmatrix} \right)^\top \Lambda \left(f - \begin{pmatrix} y \\ \tilde{y} \end{pmatrix} \right),$$

where $L = D - W(\epsilon)$ is the graph Laplacian matrix and Λ is a diagonal matrix whose diagonal elements are $\Lambda_{ii} = \lambda$ for clean points and $\Lambda_{ii} = \frac{\lambda}{c}$ for noisy points. In the classification setting, $f \in \mathbb{R}^{n+m}$ is to be optimized. The idea is to constrain the elements of f by injecting smoothness/similarity using the Laplacian matrix L. Since we assume the linear regression model, we can further plug in $f = \begin{pmatrix} X \\ \tilde{X} \end{pmatrix} \beta$. Our goal is then to estimate β by minimizing $J(\beta)$. As suggested in the original paper, we use the range of values $\epsilon \in [0, 1, 1, 5]$, $c \in [1, 10, 50]$, and $\lambda \in [1, 10, 100]$. We will evaluate all 36 possible combinations and pick the one with the smallest squared loss on the clean pool.

The results are shown in Fig. 7. We observe that only the debugging method is effective for support recovery, as we have carefully designed our method for this goal. The method

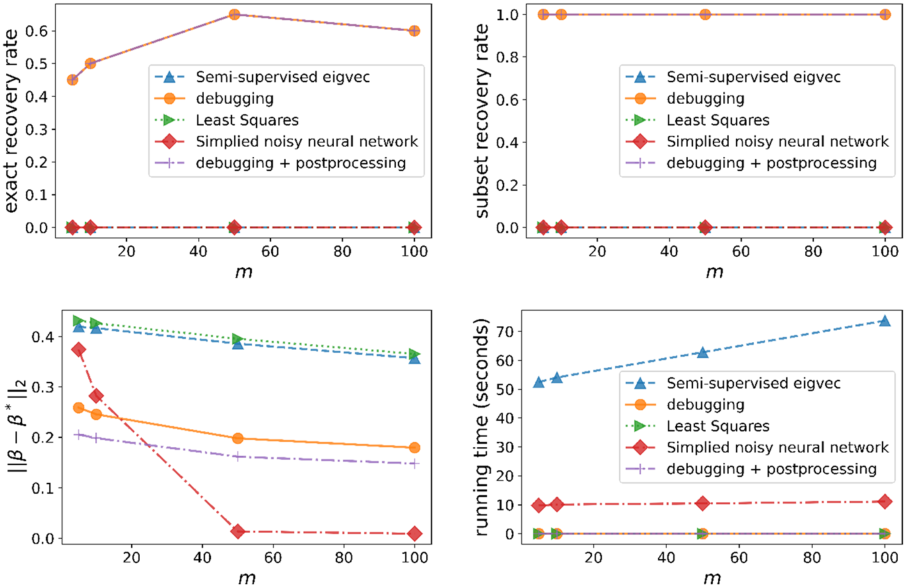


Fig. 7 Comparison to Methods involving Clean Points. Each dot is the average result of 20 random trials. We use the synthetic data setting, with $n = 500, p = 15, \sigma = 0.1, t = 0.1n$, and $\min_i |\gamma_i^*| = 10\sqrt{\log 2n}\sigma$. The clean data pool is randomly chosen from the first pool without replacement; we query the labels of these chosen points

from Veit et al. (2017) works best in terms of ℓ_2-error of β, especially when m is large. The semi-supervised method, like least squares, does not perform well, possibly because it does not consider replacing/removing the influence of the noisy dataset.

6.3.3 Effectiveness on second pool design

We now provide experiments to investigate the design of the clean pool, corresponding to Sect. 5. We use the Concrete Slump dataset,[4] where $p = 7$. We limit our study to small datasets, since the runtime of the MILP optimizer is quite long. We report the performance of the MILP debugging method in both noiseless and noisy settings. In our experiments, we compare the performance of the MILP debugger to a random debugger and a natural debugging method: adding high-leverage points into the second pool. In other words, D.milp selects m clean points to query from running the MILP (18); D.leverage selects the m points with the largest values of $x_i^\top(X^\top X)^{-1}x_i$; and D.random randomly chooses m points from the first pool without replacement. After choosing the clean pool, the debugger applies the Lasso-based algorithm. In Zhang et al. (2018), all the second pool points are chosen either randomly or artificially. Therefore, we may consider D.random as an implementation of the method in Zhang et al. (2018), which will be compared to our D.milp.

In the noiseless setting, we define β^* to be the least squares solution computed from all data points. We randomly select n data points as the x_i's. For D.milp and D.leverage,

[4] https://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test.

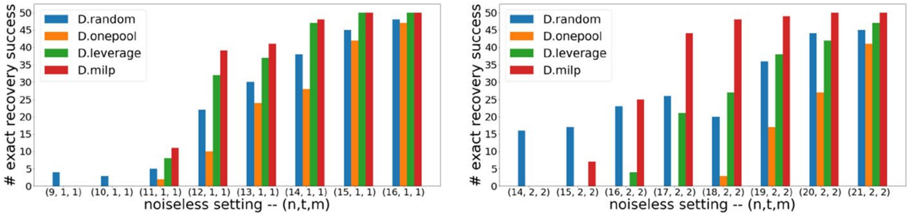


Fig. 8 Comparison between D.milp and other debugging strategies in noiseless settings. Each setting is an average over 50 random trials

since the bug generator knows their strategies or the selected D, it generates bugs according to the optimization problem (17). Let $T \subseteq [n]$ be the index set of the t largest $|u_i|$'s, for $i = 1, \dots, n$. The bug generator takes $\gamma^*_T = u_T$ if the solution u is nonzero, and otherwise randomly generates a subset T of size t to create $\gamma^*_T = \mathbf{1}$. Thus, $y_i = x_i^\top \beta^* + \gamma^*$. For D.onepool, the bug generator follows the above description with $D = \emptyset$. The orange bars indicate whether the bug generator succeeds in exact recovery in the one-pool case. For D.random, the bug generator generates bugs using the same mechanism as for D.onepool. Note the above bug generating methods are the "worst" in the sense of signed support recovery: The debuggers run (14) using their selected X_D. From Fig. 8, there is an obvious advantage of D.milp over D.onepool and D.leverage. This suggests improved performance of our MILP algorithm. D.random is sometimes successful even when n and t are small because the bug generator cannot control the randomness, but it performs worse than D.milp overall.

In the noisy setting, we define β^* to be the least squares solution computed using the entire data set. We randomly select n data points as the x_i's. For D.milp and D.leverage, since the bug generator knows their strategies or the selected D, it generates bugs via the optimization problem (17): taking $\gamma^*_T = u_T$ if the solution u is nonzero for T being the indices of the largest t elements of $|u|$, and otherwise randomly generating a subset T of size t to create $\gamma^*_T = \mathbf{1}$. Thus, $y_i = x_i^\top \beta^* + \gamma^* + \mathcal{N}(0, 0.01)$. Note that having $\gamma^*_T = u_T$ if the solution u is nonzero gives incorrect signed support recovery, which is proved in Appendix E.1. This is related to what we have claimed in Remark 2 above. For D.onepool, the bug generator follows the above description with $D = \emptyset$. The orange bars indicate whether the bug generator succeeds in exact recovery in the one-pool case. For D.random, since it is not deterministic, the bug generator does not know D and acts in the same way as in the one-pool case. Note that the above bug generating methods are the "worst" in the sense of signed support recovery. From Fig. 9, there is an obvious advantage of D.milp over D.onepool and D.leverage. Our theory only guarantees the success of D.milp in the *noiseless* setting, so the experimental results for the noisy setting are indeed encouraging.

Debugging in practice: The algorithm for minimax optimization has been executed by running all $\binom{n}{m}$ possible choices of clean points for the outer loop; for each outer loop, we then run the inner maximization. For optimal debugging in practice, i.e., n, t, and m being large, some recent work provides methods for efficiently solving the minimax MILP (Tang et al. 2016). Note that the MILP debugger can be easily combined to other heuristic methods: one can run the MILP, and if there is a nonzero solution, we can follow it to add clean points. Otherwise, we can switch to other methods, such as choosing random points or high-leverage points.

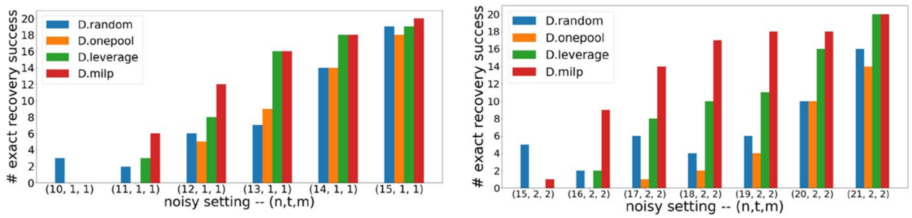


Fig. 9 Comparison between MILP Strategy and Others. In each setting, we run 20 random simulations

7 Conclusion

We have developed theoretical results for machine learning debugging via M-estimation and discussed sufficient conditions under which support recovery may be achieved. As shown by our theoretical results and illustrative examples, a clean data pool can assist debugging. We have also designed a tuning parameter algorithm which is guaranteed to obtain exact support recovery when the design matrix satisfies a certain concentration property. Finally, we have analyzed a competitive game between the bug generator and the debugger, and analyzed a mixed integer optimization strategy for the debugger. Empirical results show the success of the tuning parameter algorithm and proposed debugging strategy.

Our work raises many interesting future directions. First, the question of how to optimally choose the weight parameter η remains open. Second, although we have mentioned several efficient algorithms for bilevel mixed integer programming, we have not performed a thorough comparison of these algorithms for our specific problem. Third, although our MILP strategy for second pool design has been experimentally found to be effective in a noisy setting, we do not have corresponding theoretical guarantees. Fourth, our proposed debugging strategy is a one-shot method, and designing adaptive methods for choosing the second pool constitutes a fascinating research direction. Finally, the analysis of our tuning parameter algorithm suggests that a geometrically decreasing series might be used as a grid choice for more general tuning parameter selection methods, e.g., cross validation—in practice, one may not need to test candidate parameters on a large grid chosen linearly from an interval. Lastly, it would be very interesting to extend the ideas in this work to regression or classification settings where the underlying data do not follow a simple linear model.

The supplmentary materials is organized as follows: Sect. A presents some additional discussions on β. Sections B, C, D and E mainly provide proofs respectively for problem reformulation and support recovery, tuning parameter selection and strategy for second pool selection. They may also include additional discussions and formal statements as referred in the main text.

A Additional discussions

We present more miscellaneous discussions here to readers who may care about β.

Debugging connection to β. Throughout this paper, we have focused on estimating γ for the purpose of debugging. A result concerning how the second pool can be used to obtain a better estimate of β is as follows:

Proposition 3 *Let $X = USV^\top$ and $\widetilde{X} = \widetilde{S}V_0^\top$. Let $m < p$. It holds that*

$$\|V_0(\widehat{\beta} - \beta^*)\|_2 \leq \frac{c_1\sigma\sqrt{m}}{\sqrt{L}\sigma_{\min}(\tilde{S})} + \lambda n\|\tilde{S}^{-2}V_0VSUz_{\widehat{\gamma}}\|_2, \tag{19}$$

where $z_{\widehat{\gamma}}$ is the subgradient of $\|\widehat{\gamma}\|_1$.

Proof of Proposition 3 Recall the objective function (3) is

$$(\widehat{\beta}, \widehat{\gamma}) \in \arg \min_{\substack{\beta \in \mathbb{R}^p, \\ \gamma \in \mathbb{R}^n}} \left\{ \frac{1}{2n}\|y - X\beta - \gamma\|_2^2 + \frac{\eta}{2m}\|\widetilde{y} - \widetilde{X}\beta\|_2^2 + \lambda\|\gamma\|_1 \right\}.$$

By KKT conditions of the objective function,

$$\begin{aligned} \nabla_\beta &= -\frac{1}{n}X^\top(y - X\widehat{\beta} - \widehat{\gamma}) - \frac{\eta}{m}\widetilde{X}^\top(\widetilde{y} - \widetilde{X}\widehat{\beta}) = 0; \\ \nabla_\gamma &= -\frac{1}{n}(y - X\widehat{\beta} - \widehat{\gamma}) + \lambda\partial|\widehat{\gamma}| = 0. \end{aligned} \tag{20}$$

Plug $y = X\beta^* + \gamma^* + \epsilon$ and $\widetilde{y} = \widetilde{X}\beta^* + \widetilde{\epsilon}$ into (20) we obtain

$$-\left(\frac{1}{n}X^\top X + \frac{\eta}{m}\widetilde{X}^\top\widetilde{X}\right)(\beta^* - \widehat{\beta}) - \frac{1}{n}X^\top(\gamma^* - \widehat{\gamma}) - \frac{1}{n}X^\top\epsilon - \frac{\eta}{m}\widetilde{X}^\top\widetilde{\epsilon} = 0; \tag{21a}$$

$$-\frac{1}{n}X(\beta^* - \widehat{\beta}) - \frac{1}{n}(\gamma^* - \widehat{\gamma}) - \frac{1}{n}\epsilon + \lambda\partial|\widehat{\gamma}| = 0. \tag{21b}$$

Mutiply $X^\top$ on (21b) and plug it into (21a) we get

$$\widetilde{X}^\top\widetilde{X}(\widehat{\beta} - \beta^*) = \lambda\frac{m}{\eta}X^\top\partial|\widehat{\gamma}| + \widetilde{X}\widetilde{\epsilon}. \tag{22}$$

Given that $\widetilde{X} = \widetilde{S}V_0^\top$,

$$\widetilde{S}^\top\widetilde{S}V_0^\top(\widehat{\beta} - \beta^*) = \lambda\frac{m}{\eta}V_0^\top X^\top\partial|\widehat{\gamma}| + V_0^\top V_0\widetilde{S}\widetilde{\epsilon}.$$

Plugging into the SVD of $X = USV^\top$, we have

$$\begin{aligned}\left\|V_0^\top(\widehat{\beta}-\beta^*)\right\|_2 &\leq \lambda\frac{m}{\eta}\left\|(\widetilde{S}^\top\widetilde{S})^{-1}V_0^\top X^\top\partial|\widehat{\gamma}|\right\|_2 + \|(\widetilde{S}^\top\widetilde{S})^{-1}\widetilde{S}\|\|\widetilde{\epsilon}\|_2\\ &\leq \lambda\frac{m}{\eta}\left\|(\widetilde{S}^\top\widetilde{S})^{-1}V_0^\top VSU^\top\partial|\widehat{\gamma}|\right\|_2 + c_1\frac{\sqrt{m}\sigma}{\sqrt{L}\sigma_{\min}(\widetilde{S})}\\ &\leq \lambda\frac{m}{\eta}\left\|(\widetilde{S}^\top\widetilde{S})^{-1}V_0^\top VSU^\top\right\|_2\sqrt{n} + c_1\frac{\sqrt{m}\sigma}{\sqrt{L}\sigma_{\min}(\widetilde{S})}\\ &\leq c\sigma\sqrt{\frac{\log n}{n}}\frac{m}{\eta}\left\|(\widetilde{S}^\top\widetilde{S})^{-1}S_0^{1/2}\right\|_2 + c_1\frac{\sqrt{m}\sigma}{\sqrt{L}\sigma_{\min}(\widetilde{S})},\end{aligned}$$

with probability at least $1-\exp(-cm)$. The second step is because $\widetilde{\sigma}$ has subgaussian parameter σ^2/L. □

Note that when $\widetilde{S}$ is chosen large enough, then $\|V_0(\widehat{\beta}-\beta^*)\|_2$ is controlled to a small number. Besides, if the subspace V_0 contains the buggy subspace of X_T, then $\|y_T-y_T^*\|_2$ is well controlled and we can spot the contaminated points. This, together with the orthogonal design we will discuss in Sect. C.2, suggests that a successful debugging strategy may be obtained by producing a carefully chosen interaction between the non-buggy subspace (augmented using a second pool of clean data points) and the buggy subspace.

Related work She and Owen (2011). Without the second pool, She and Owen (2011) demonstrated the equivalence of the solution $\widehat{\beta}$ to the joint optimization of the objective (3) over (β,γ) to the optimum of a regression M-estimator in β with the Huber loss. This motivates the question of whether the optimizer $\widehat{\beta}$ of the objective (3) may similarly be viewed as the optimum of an M-estimation problem.

Proposition 4 *The solution $\widehat{\beta}$ of the joint optimization problem* (3) *is the unique optimum of the following weighted M-estimation problem:*

$$\min_{\beta\in\mathbb{R}^p}\left\{\frac{1}{n}\sum_{i=1}^{n}\ell_{n\lambda}\big(y_i-x_i^\top\beta\big)+\frac{\eta}{2m}\|\widetilde{y}-\widetilde{X}\beta\|_2^2\right\}. \tag{23}$$

Proof Recall the definition of the Huber loss function:

$$\ell_k(u)=\begin{cases}\lambda|u|-\frac{k^2}{2}, & \text{if } |u|>k,\\ \frac{u^2}{2}, & \text{if } |u|<k.\end{cases}$$

We will show the desired equivalence via the KKT conditions for both objective functions. Taking gradients with respect to β and γ for the original objective function (3), we obtain the following system of equations:

$$0=\frac{X^\top X}{n}\beta-\frac{X^\top(y-\gamma)}{n}+\eta\left(\frac{\widetilde{X}^\top\widetilde{X}}{m}\beta-\frac{\widetilde{X}^\top\widetilde{y}}{m}\right), \tag{24}$$

$$0=\frac{\gamma}{n}-\frac{y-X\beta}{n}+\lambda\,\mathrm{sign}\,(\gamma). \tag{25}$$

The second equation (25) has a unique solution, given by the soft-thresholding function:

$$\gamma = \mathrm{SoftThresh}_{n\lambda}(y - X\beta),$$

where for scalars $u, k \in \mathbb{R}$, we have

$$\mathrm{SoftThresh}_k(u) = \begin{cases} u - \lambda\,\mathrm{sign}\,(u), & \text{if } |u| \geq k, \\ 0, & \text{if } |u| < k, \end{cases}$$

and $\mathrm{SoftThresh}_k$ acts on vectors componentwise. Plugging back into Eq. (24), we obtain

$$0 = X^\top \left(\frac{X\beta - y}{n} + \frac{1}{n}\mathrm{SoftThresh}_{n\lambda}(y - X\beta) \right) + \eta \left(\frac{\widetilde{X}^\top \widetilde{X}}{m}\beta - \frac{\widetilde{X}^\top \widetilde{y}}{m} \right). \tag{26}$$

We now consider the KKT conditions for the weighted M-estimator (23). Taking a gradient with respect to β, we obtain

$$0 = -\sum_{i=1}^{n} \ell'_{n\lambda}\big(y_i - x_i^\top \beta\big)\frac{x_i}{n} + \eta \left(\frac{\widetilde{X}^\top \widetilde{X}}{m}\beta - \frac{\widetilde{X}^\top \widetilde{y}}{m} \right). \tag{27}$$

The key is to note that

$$u - \ell'_{n\lambda}(u) = \mathrm{SoftThresh}_{n\lambda}(u),$$

so

$$-\ell'_{n\lambda}\big(y_i - x_i^\top \beta\big)\frac{1}{n} = \frac{x_i^\top \beta - y_i}{n} + \frac{1}{n}\mathrm{SoftThresh}_{n\lambda}\big(y_i - x_i^\top \beta\big),$$

from which we may infer the equivalence of Eqs. (26) and (27). This concludes the proof. □

The proposition also illustrates that the objective uses Huber loss to get the robust estimation $\widehat{\beta}$, and then imply the estimation $\widehat{\gamma}$. Therefore, estimations of β and γ complement each other. Our reformulation more relies on giving a direct analysis of γ and its support.

B Appendix for Sect. 2

We show reformulation of the objective function in this section.

Proof of Proposition 1 Using the notation (4), we can translate (3) into

$$(\widehat{\beta}, \widehat{\gamma}) \in \arg\min_{\beta,\gamma} \left\{ \frac{1}{2n} \left\| y' - X'\beta - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right\|_2^2 + \lambda \|\gamma\|_1 \right\}, \tag{28}$$

First note that we can split $y' - X'\beta - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix}$ into two parts by projecting onto the column space of X' and the perpendicular space:

$$\left\| y' - X'\beta - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right\|_2^2 = \left\| P_{X'} \left(y' - X'\beta - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right) \right\|_2^2 + \left\| P_{X'}^{\perp} \left(y' - X'\beta - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right) \right\|_2^2$$

$$= \left\| P_{X'} \left(y' - X'\beta - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right) \right\|_2^2 + \left\| P_{X'}^{\perp} \left(y' - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right) \right\|_2^2.$$

For any value of $\hat{\gamma}$, we can choose $\hat{\beta}$ such that $\left\| P_{X'} \left(y' - X'\hat{\beta} - \begin{bmatrix} \gamma \\ \mathbf{0}_m \end{bmatrix} \right) \right\|_2^2 = 0$, simply by taking $\hat{\beta} = (X'^{\top}X')^{-1}X'^{\top} \left(y' - \begin{bmatrix} \hat{\gamma} \\ \mathbf{0}_m \end{bmatrix} \right)$. Hence, we get

$$\left\| y' - X'\beta - \begin{bmatrix} \hat{\gamma} \\ \mathbf{0}_m \end{bmatrix} \right\|_2^2 = \left\| P_{X'}^{\perp} \left(y' - \begin{bmatrix} \hat{\gamma} \\ \mathbf{0}_m \end{bmatrix} \right) \right\|_2^2 = \|P_{X'}^{\perp} y' - \bar{P}\hat{\gamma}\|_2^2,$$

and (28) becomes

$$\hat{\gamma} \in \frac{1}{2n} \|P_{X'}^{\perp} y' - \bar{P}\hat{\gamma}\|_2^2 + \lambda \|\hat{\gamma}\|_1,$$
$$\hat{\beta} = (X'^{\top}X')^{-1}X'^{\top} \left(y' - \begin{bmatrix} \hat{\gamma} \\ \mathbf{0}_m \end{bmatrix} \right).$$

Therefore, the two optimization problems share the same solution for $\hat{\gamma}$. □

C Appendix for Sect. 3

Notations in appendix: We write $P_{X',TT}^{\perp}$ to represent the submatrix of $P_{X'}^{\perp}$ with rows and column indexed by T. We write $P_{X',T\cdot}^{\perp}$ to represent the submatrix of $P_{X'}^{\perp}$ with rows indexed by T and $P_{X',\cdot T}^{\perp}$ to represent the submatrix of $P_{X'}^{\perp}$ with columns indexed by T. For simplicity, let $\bar{P} = P_{X'}^{\perp} M_{[n]}$. We slightly abuse notation by using $\bar{P}_T$ and $\bar{P}_{T^c}$ to denote $\bar{P}_{\cdot T}$ and $\bar{P}_{\cdot T^c}$, respectively.

In this appendix, we provide proofs and additional details for the results in Sect. 3. The proofs for fixed design are in Sect. C.1. We discuss orthogonal design in Sect. C.2 and sub-Gaussian design in Sect. C.3. In particular, we use the two special designs to better understand the three assumptions and see how having a clean pool helps with the support recovery. We will call one-pool case the setting with only contaminated pool and call two-pool case the setting with both data pools.

C.1 Proofs of Theorem 1 and Theorem 2

Proof of Theorem 1 We follow the usual Primal Dual Witness argument for support recovery in linear regression, which contains the following steps (Wainwright 2009):

1. Set $\hat{\gamma}_{T^c} = 0$.
2. Solve the oracle subproblem for $(\hat{\gamma}_T, \hat{z}_T)$:

$$\widehat{\gamma}_T \in \arg\min_{\gamma\in\mathbb{R}^t} \left\{ \frac{1}{2n}\left\|Ay' - B\gamma\right\|_2^2 + \lambda\|\gamma\|_1 \right\}, \tag{29}$$

and choose $\hat{z}_T \in \partial\left\|\widehat{\gamma}_T\right\|_1$. In the one data pool case, we have $A = P^{\perp}_{X,\cdot T}$ and $B = P^{\perp}_{X,\cdot T}$; in the two data pool case, we have $A = P^{\perp}_{X',\cdot T}$ and $B = \bar{P}_T$.

3. Solve $\hat{z}_{T^c}$ via the zero-subgradient equation, and check whether the strict dual feasibility condition holds: $\left\|\widehat{z}_{T^c}\right\|_\infty < 1$.

As in the usual Lasso analysis (Wainwright 2009), under the eigenvalue condition (6), $(\widehat{\gamma}_T, 0) \in \mathbb{R}^n$ is the unique optimal solution of the Lasso, where $\widehat{\gamma}_T$ is the solution obtained by solving the oracle subproblem (29).

The focus of our current analysis is to verify the conditions under which the strict dual feasibility condition holds. The KKT conditions for Eq. (5) may be rewritten as

$$\bar{P}_T^\top \bar{P}_T(\hat{\gamma}_T - \gamma_T^*) - \bar{P}_T^\top P^{\perp}_{X'}\epsilon' + n\lambda\hat{z}_T = 0, \tag{30}$$

$$\bar{P}_{T^c}^\top \bar{P}_T(\hat{\gamma}_T - \gamma_T^*) - \bar{P}_{T^c}^\top P^{\perp}_{X'}\epsilon' + n\lambda\hat{z}_{T^c} = 0, \tag{31}$$

where $\hat{z}_T \in \partial\left\|\widehat{\gamma}_T\right\|_1, \hat{z}_{T^c} \in \partial\left\|\widehat{\gamma}_{T^c}\right\|_1$.

We will use the following equations to simplify terms later:

$$\bar{P}_T^\top \bar{P}_T = (P^{\perp\top}_{X'} P^{\perp}_{X'})_{TT}, \quad \begin{pmatrix} \bar{P}_T^\top P^{\perp}_{X'}\epsilon' \\ \bar{P}_{T^c}^\top P^{\perp}_{X'}\epsilon' \end{pmatrix} = \bar{P}^\top P^{\perp}_{X'}\epsilon' = \bar{P}^\top\epsilon' = \begin{pmatrix} \bar{P}_T^\top\epsilon' \\ \bar{P}_{T^c}^\top\epsilon' \end{pmatrix}.$$

Since $\bar{P}_T^\top \bar{P}_T$ is invertible by condition (6), we can multiply Eq. (30) by $\left(\bar{P}_T^\top \bar{P}_T\right)^{-1}$ on the left to obtain

$$\hat{\gamma}_T - \gamma_T^* = (\bar{P}_T^\top \bar{P}_T)^{-1}\bar{P}_T^\top\epsilon' - n\lambda(\bar{P}_T^\top \bar{P}_T)^{-1}\hat{z}_T. \tag{32}$$

Plugging this into Eq. (31), we then obtain

$$\hat{z}_{T^c} = -\frac{1}{n\lambda}\bar{P}_{T^c}^\top \bar{P}_T\left[(\bar{P}_T^\top \bar{P}_T)^{-1}\bar{P}_T^\top\epsilon' - n\lambda(\bar{P}_T^\top \bar{P}_T)^{-1}\hat{z}_T\right] + \frac{1}{n\lambda}\bar{P}_{T^c}^\top\epsilon',$$

or

$$\hat{z}_{T^c} = \underbrace{\bar{P}_{T^c}^\top \bar{P}_T(\bar{P}_T^\top \bar{P}_T)^{-1}\hat{z}_T}_{\mu} + \underbrace{\bar{P}_{T^c}^\top\left(I - \bar{P}_T(\bar{P}_T^\top \bar{P}_T)^{-1}\bar{P}_T^\top\right)\frac{\epsilon'}{n\lambda}}_{V_{T^c}}. \tag{33}$$

We need to show that $\|\hat{z}_{T^c}\|_\infty < 1$.

Note that condition (7) gives us

$$\exists\alpha' \in [0,1),\ \|\mu\|_\infty = \max_{j\in T^c}\|\bar{P}_j^\top \bar{P}_T(\bar{P}_T^\top \bar{P}_T)^{-1}\|_1 \le \alpha'.$$

Furthermore, since

$$\lambda \ge \frac{1}{1-\alpha'}\left\|\bar{P}_{T^c}^\top\left(I - \bar{P}_T(\bar{P}_T^\top \bar{P}_T)^{-1}\bar{P}_T^\top\right)\frac{\epsilon'}{n}\right\|_\infty,$$

we have

$$\|V_{T^c}\|_\infty \leq \frac{1-\alpha'}{2}.$$

Combining these inequalities, we obtain strict dual feasibility:

$$\|\hat{z}_{T^c}\|_\infty \leq \|\mu\|_\infty + \|V_{T^c}\|_\infty < 1.$$

In addition, applying the triangle inequality to the RHS of Eq. (32), we obtain

$$G' = \|(\bar{P}_T^\top \bar{P}_T)^{-1}\bar{P}_T^\top \epsilon'\|_\infty + n\lambda\|(\bar{P}_T^\top \bar{P}_T)^{-1}\hat{z}_T\|_\infty \geq \|\hat{\gamma}_T - \gamma_T^*\|_\infty.$$

This concludes the proof. □

Proof of Theorem 2 Note that

$$\forall i \in T, \quad |\gamma_i^*| - |\hat{\gamma}_i| \leq \|\hat{\gamma}_T - \gamma_T^*\|_\infty \leq G',$$

where the last inequality uses Theorem 1. Thus, if condition (8) also holds, we have

$$\forall i \in T, \quad |\hat{\gamma}_i| \geq \min_{i\in T}|\gamma_i^*| - \|\hat{\gamma}_T - \gamma_T^*\|_\infty \geq \min_{i\in T}|\gamma_i^*| - G' > 0,$$

concluding the proof. □

C.2 Orthogonal design

C.2.1 Main results for orthogonal design

In this section, we focus on a special case, where our data have an orthogonal property. Let $X = \begin{bmatrix} RQ^\top \\ FQ^\top \end{bmatrix} \in \mathbb{R}^{(t+p)\times p}, \widetilde{X} = WQ^\top \in \mathbb{R}^{p\times p}$, where Q is an orthogonal matrix with columns $q_1, q_2, \cdots, q_p$, F, W are diagonal matrices with diagonals f_i's and w_i's separately ($i \in [p]$), and $R = \left[\begin{array}{cccc|c} r_1 & 0 & 0 & 0 & \\ 0 & r_2 & 0 & 0 & \\ 0 & 0 & \cdots & 0 & 0_{t\times(p-t)} \\ 0 & 0 & 0 & r_t & \end{array}\right]$. We assume for all $i \in [p]$, $r_i \neq 0$,$f_i \neq 0$. Consider the first t points are buggy and the rest p points are nonbuggy, i.e., $X_T = RQ^\top \in \mathbb{R}^{t\times p}, X_{T^c} = FQ^\top \in \mathbb{R}^{p\times p}$.

Applying Theorems 1 and 2, we obtain Propositions 5 and 6.

Proposition 5 *In the one-pool case, suppose we choose*

$$\lambda \geq \frac{2\sigma}{n(1-\alpha)}\left(\sqrt{\log 2(n-t)} + C\right), \tag{34}$$

for some constant $C > 0$, *and*

$$\alpha = \max_{1\leq i\leq t}\left|\frac{r_i}{f_i}\right| < 1. \tag{35}$$

Then the contaminated pool is capable of achieving subset support recovery with probability at least $1 - e^{-\frac{C^2}{2}}$.

In the two-pool case, suppose we choose

$$\lambda \geq \frac{2\sigma}{n(1-\alpha')} \max\left\{1, \sqrt{\frac{\eta n}{mL}}\right\}\left(\sqrt{\log 2(n-t)} + C'\right), \tag{36}$$

for some constant $C' > 0$, *and*

$$\alpha' = \max_{1\leq i\leq t}\left|\frac{r_i f_i}{f_i^2 + \eta\frac{n}{m}w_i^2}\right| < 1. \tag{37}$$

Then adding clean points will achieve subset support recovery with probability at least $1-e^{-\frac{c'^2}{2}}$.

As stated in Theorems 1 and 2, to ensure exact recovery, we also need to impose a gamma-min condition. This leads to the following proposition:

Proposition 6 *In the one-pool case, suppose inequality* (35) *holds. If also*

$$\min_{1\leq i\leq t}|\gamma_i^*| > \sigma(\sqrt{2\log t} + c)\max_{1\leq i\leq t}\sqrt{1+\frac{r_i^2}{f_i^2}} + \frac{2\sigma}{1-\alpha}\left(\sqrt{\log 2(n-t)} + C\right)\left(1+\max_{1\leq i\leq t}\frac{r_i^2}{f_i^2}\right), \tag{38}$$

then there exists a λ *to achieve exact recovery, with probability at least* $1-2e^{-\frac{c^2}{2}}-e^{-\frac{c^2}{2}}$.

In the two-pool case, suppose $\eta \leq \frac{mL}{n}$, *and inequality* (37) *holds. If also*

$$\begin{aligned}\min_{1\leq i\leq t}|\gamma_i^*| \geq{}& \sigma(\sqrt{2\log t}+c)\sqrt{1+\max_{1\leq i\leq t}\frac{r_i^2(Lf_i^2+\frac{\eta n}{m}w_i^2)}{L(f_i^2+\frac{\eta n}{m}w_i^2)^2}}\\ &+\frac{2\sigma}{1-\alpha'}\left(\sqrt{\log 2(n-t)}+C\right)\left(1+\max_{1\leq i\leq t}\frac{r_i^2}{f_i^2+\frac{\eta n}{m}w_i^2}\right),\end{aligned} \tag{39}$$

then there exists a λ *to achieve exact recovery, with probability at least* $1-2e^{-\frac{c^2}{2}}-e^{-\frac{c^2}{2}}$.

Compare (35) and (37). Mutual incoherence is decreased from $\frac{r_i^2}{f_i^2}$ to $\frac{r_i^2}{f_i^2+\frac{\eta n}{m}w_i^2}$. Compare (38) and (39). The second max term, $\max_{1\leq i\leq t}\frac{r_i^2}{f_i^2} \geq \max_{1\leq i\leq t}\frac{r_i^2(Lf_i^2+\frac{\eta n}{m}w_i^2)}{L(f_i^2+\frac{\eta n}{m}w_i^2)^2}$, because

$$\max_{1\leq i\leq t}\frac{r_i^2}{f_i^2} \geq \max_{1\leq i\leq t}\frac{r_i^2(f_i^2+\frac{\eta n}{m}w_i^2)}{(f_i^2+\frac{\eta n}{m}w_i^2)^2} \geq \max_{1\leq i\leq t}\frac{r_i^2(Lf_i^2+\frac{\eta n}{m}w_i^2)}{L(f_i^2+\frac{\eta n}{m}w_i^2)^2}$$

when $L \geq 1$. Also note that $\frac{1}{1-\alpha} > \frac{1}{1-\alpha'}$. Altogether, the requirement of $\min_{i\in[t]}|\gamma_i^*|$ is weakened by introducing clean points. Thus, we see that the mutual incoherence improves in two-pool setting. The gamma-min condition imposes a lower bound of $\Omega\left(\sqrt{\log(n-t)}\right)$ on the signal-to-noise ratio, $\frac{\min_{i\in[t]}|\gamma_i^*|}{\sigma}$, and including second pool reduces the prefactor.

As can be seen, we want $|w_i|$ to be sufficiently large compared to $|f_i|$. However, if $|w_i|$ is bounded, we may instead ensure support recovery by repeating points. In this section, we discuss the effect of repeating points and determine the number of points needed to guarantee correct support recovery. Suppose

$$W = \begin{bmatrix} \mathbf{w_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{w_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{w_p} \end{bmatrix},$$

where $\mathbf{w_i} = [w_{i1}, \dots, w_{il_i}]^\top$. For the ith direction q_i, we have k_i repeated points with respective weights $w_{i1}, w_{i2}, \dots, w_{il_i}$.

Proposition 7 *Suppose the scale of clean data points is bounded by* w_B. *Using* $w_{i1}, \dots, w_{il_i}$, *where* $l_i = \left\lceil \left(\frac{|w_i|}{w_B}\right)^2 \right\rceil$ *and* $|w_{ij}| = w_B$, $\forall j \in [l_i]$, *achieves the same effect on Conditions* 1, 2, *and* 3 *as adding a single point with scale* w_i.

From Proposition 7, we see that to correctly identify the bugs, we can also query multiple points in the same direction if the leverage of a single additional point is not large enough.

C.2.2 Proofs for orthogonal design

In this section, we first simplify the three conditions, and then provide the proofs of Propositions 5, 6, and 7.

In the one-pool case, we have

$$\begin{aligned} P^\perp_{X,TT} &= I_{t\times t} - X_T(X^\top X)^{-1}X_T^\top \\ &= I_{t\times t} - R(R^\top R + F^\top F)^{-1}R^\top \\ &= diag\left(\frac{f_1^2}{r_1^2+f_1^2}, \cdots, \frac{f_t^2}{r_t^2+f_t^2}\right). \end{aligned}$$

Note that $P^\perp_{X,TT}$ is a diagonal matrix. Thus, the eigenvalues are immediately obtained and

$$\lambda_{\min}(P^\perp_{X,TT}) = \min_{1\le i\le t} \frac{f_i^2}{r_i^2+f_i^2} = \min_{1\le i\le t} \frac{1}{\left(\frac{r_i}{f_i}\right)^2+1} = \frac{1}{\max_{1\le i\le t}\left(\frac{r_i}{f_i}\right)^2+1}.$$

The condition that $P^\perp_{X,TT}$ is invertible is therefore equivalent to the condition that $f_i \neq 0$ for all i. Assuming this is true, we have

$$\begin{aligned} P^\perp_{X,T^cT}(P^\perp_{X,TT})^{-1} &= -F(R^\top R + F^\top F)^{-1}R^\top \cdot (I_{t\times t} - R(R^\top R + F^\top F)^{-1}R^\top)^{-1} \\ &= \begin{bmatrix} diag\left(-\frac{r_1}{f_1}, \cdots, -\frac{r_t}{f_t}\right)_{t\times t} \\ \mathbf{0}_{(p-t)\times t} \end{bmatrix}. \end{aligned}$$

The mutual incoherence condition can then be written in terms of the quantity

$$\left\| P^{\perp}_{X,T^cT}(P^{\perp}_{X,TT})^{-1} \right\|_{\infty} = \max_{1\le i\le t} \left| \frac{r_i}{f_i} \right| = \max_{1\le i\le t} \left| \frac{r_i f_i}{f_i^2} \right|.$$

Note that the mutual incoherence condition also implies that $f_i \neq 0$, $\forall i$, since the mutual incoherence parameter will otherwise go to infinity.

The remaining condition is the gamma-min condition. Note that the upper bound on the ℓ_{∞}-error of γ consists of two parts:

$$\|\hat{\gamma} - \gamma^*\|_{\infty} \le \|(P^{\perp}_{X,TT})^{-1}(P^{\perp}_{X,T\cdot})\epsilon\|_{\infty} + n\lambda \left\| (P^{\perp}_{X,TT})^{-1} \right\|_{\infty}.$$

Regarding $P^{\perp}_{X,T\cdot}$ as two blocks, $\left(P^{\perp}_{X,TT},\ P^{\perp}_{X,TT^c} \right)$, we have

$$\|(P^{\perp}_{X,TT})^{-1}(P^{\perp}_{X,T\cdot})\epsilon\|_{\infty} = \left\| \left(I \ \ (P^{\perp}_{X,TT})^{-1} P^{\perp}_{X,TT^c} \right) \epsilon \right\|_{\infty}.$$

Altogether, we see that

$$G = \max_{1\le i\le t} \left| \epsilon_i - \frac{r_i}{f_i}\epsilon_{i+t} \right| + n\lambda \left(\max_{1\le i\le t} \left\{ \frac{r_i^2}{f_i^2} \right\} + 1 \right).$$

To summarize, the minimum eigenvalue condition becomes

$$\lambda_{\min}(P^{\perp}_{X,TT}) = \frac{1}{\max_{1\le i\le t} \left(\frac{r_i}{f_i}\right)^2 + 1} > 0; \tag{40a}$$

the mutual incoherence condition becomes

$$\left\| P^{\perp}_{X,T^cT}(P^{\perp}_{X,TT})^{-1} \right\|_{\infty} = \max_{1\le i\le t} \left| \frac{r_i}{f_i} \right| = \alpha \in [0,1); \tag{40b}$$

and the gamma-min condition becomes

$$\min_{1\le i\le t} |\gamma_i^*| \ge G = \max_{1\le i\le t} |\epsilon_i - \frac{r_i}{f_i}\epsilon_{i+t}| + n\lambda \left(\max_{1\le i\le t} \left\{ \frac{r_i^2}{f_i^2} \right\} + 1 \right). \tag{40c}$$

Similar calculations show that in the two-pool case, the minimum eigenvalue condition becomes

$$\lambda_{\min}(P^{\perp}_{X',TT}) = \min_{1\le i\le t} \frac{f_i^2 + \frac{\eta n}{m} w_i^2}{r_i^2 + f_i^2 + \frac{\eta n}{m} w_i^2} = \frac{1}{\max_{i\in[t]} \frac{r_i^2}{f_i^2 + \frac{\eta n}{m} w_i^2} + 1} > 0; \tag{41a}$$

the mutual incoherence condition becomes

$$\left\| P^{\perp}_{X',T^cT}(P^{\perp}_{X',TT})^{-1} \right\|_{\infty} = \max_{1\le i\le t} \left| \frac{r_i f_i}{f_i^2 + \frac{\eta n}{m} w_i^2} \right| = \alpha' \in [0,1); \tag{41b}$$

and the gamma-min condition becomes

$$\min_{1\le i\le t}|\gamma_i^*| \ge G', \tag{41c}$$

where

$$G' = \max_{1\le i\le t}\left|\epsilon_i - \frac{r_i f_i}{f_i^2 + \frac{\eta n}{m}w_i^2}\epsilon_{i+t} - \frac{\sqrt{\frac{\eta n}{m}}r_i w_i}{f_i^2 + \frac{\eta n}{m}w_i^2}\tilde{\epsilon}_i\right| + n\lambda\left(\max_{1\le i\le t}\left\{\frac{r_i^2}{f_i^2 + \frac{\eta n}{m}w_i^2}\right\} + 1\right).$$

Here is the proof of Proposition 5.

Proof of Proposition 5 According to Theorem 1, the subset support recovery result relies on two conditions: the minimum eigenvalue condition and the mutual incoherence condition. In the orthogonal design case, we will argue that both inequalities (40a) and (41a) hold in the one-pool case, and inequlaity (37) is sufficient for both inequalities (41a) and (41b) in the two-pool case.

For the one-pool case, the assumption (35) implies that $f_i \neq 0, \ ,\forall i \in [t]$. Note that the minimum eigenvalue condition (40a) is equivalent to $f_i \neq 0, \ ,\forall i \in [t]$. Hence, the minimum eigenvalue condition holds. Furthermore, the mutual incoherence condition (41a) clearly holds.

For the two-pool case, if $f_i = 0$ for some $i \in [t]$, then plugging into (37) implies that $w_i^2 > 0$. Thus, f_i and w_i cannot be zero at the same time, implying that the eigenvalue condition (41a) holds. Note that inequality (37) is equivalent to inequlaity (41b).

The remaining of the argument concerns the choice of λ. Note that Theorem 1 requires λ to be lower-bounded for subset recovery (see inequality (9)). Taking the two-pool case as an example, we will show that when inequality (36) holds, inequality (9) holds with high probability. Define

$$Z_j = \bar{P}_{\cdot j}^\top\left(I - \bar{P}_T(\bar{P}_T^\top\bar{P}_T)^{-1}\bar{P}_T^\top\right)\frac{\epsilon'}{n}, \quad j \in T^c.$$

Note that $\left\|\bar{P}_{\cdot j}^\top\left(I - \bar{P}_T(\bar{P}_T^\top\bar{P}_T)^{-1}\bar{P}_T^\top\right)\right\|_2 \le 1$ for all $j \in T^c$, and $\epsilon' = \begin{pmatrix}\epsilon \\ \sqrt{\frac{\eta n}{m}}\tilde{\epsilon}\end{pmatrix}$ has i.i.d. sub-Gaussian entries with parameter at most $\max\{1, \frac{\eta n}{mL}\}\sigma^2$. Thus, Z_j is sub-Gaussian with parameter at most $\max\{1, \frac{\eta n}{mL}\}\frac{\sigma^2}{n^2}$. By a sub-Gaussian tail bound (cf. Lemma 1), we then have

$$\mathbb{P}\left(\max_{j\in T^c}|Z_j| \ge \delta_0\right) \le 2(n-t)\exp\left(-\frac{n^2\delta_0^2}{2\max\{1, \frac{\eta n}{mL}\}\sigma^2}\right).$$

Let C' be a constant such that

$$2(n-t)\exp\left(-\frac{n^2\delta_0^2}{2\max\{1, \frac{\eta n}{mL}\}\sigma^2}\right) = \exp\left(-\frac{C'^2}{2}\right),$$

and define

$$\delta_0 := \frac{\sigma}{n}\max\{1, \sqrt{\frac{\eta n}{mL}}\}\sqrt{\log 2(n-t) + C'^2}.$$

Note that we want

$$\frac{2\max_{j\in T^c}|Z_j|}{1-\alpha'} \le \lambda,$$

which therefore occurs with probability at least $1-e^{-\frac{C'^2}{2}}$ when

$$\lambda \ge \frac{2\sigma}{n(1-\alpha')}\max\{1,\sqrt{\frac{\eta n}{mL}}\}\left(\sqrt{\log 2(n-t)}+C'\right) \ge \frac{2\delta_0}{1-\alpha'}.$$

The proof for the one-pool case is similar, so we omit the details. □

Here is the proof of Proposition 6.

Proof of Proposition 6 To simplify notation, define

$$u_i := \epsilon_i - \frac{r_i}{f_i}\epsilon_{i+t},$$

$$v_i := \epsilon_i - \frac{r_i f_i}{f_i^2+\frac{\eta n}{m}w_i^2}\epsilon_{i+t} - \frac{\sqrt{\frac{\eta n}{m}}r_i w_i}{f_i^2+\frac{\eta n}{m}w_i^2}\tilde{\epsilon}_i.$$

Note that u_i is σ_{u_i}-sub-Gaussian and v_i is σ_{v_i}-sub-Gaussian, with variance parameters

$$\sigma_{u_i} = \sqrt{1+\frac{r_i^2}{f_i^2}}\sigma, \quad \sigma_{v_i} = \sqrt{1+\frac{r_i^2(L^2f_i^2+\frac{\eta n}{m}w_i^2)}{L^2(f_i^2+\frac{\eta n}{m}w_i^2)^2}}\sigma.$$

We now prove two technical lemmas:

Lemma 1 (Concentration for non-identical sub-Gaussian random variables) *Suppose $\{u_i\}_{i=1}^t$ are σ_{u_i}-sub-Gaussian random variables and $\{v_i\}_{i=1}^t$ are σ_{v_i}-sub-Gaussian random variables. Then the following inequalities hold:*

$$P\left(\max_{1\le i\le t}|u_i| > \delta_1\right) \le 2t\exp\left(-\frac{\delta_1^2}{2\max_{1\le i\le t}\sigma_{u_i}^2}\right), \tag{42}$$

$$P\left(\max_{1\le i\le t}|v_i| > \delta_1\right) \le 2t\exp\left(-\frac{\delta_1^2}{2\max_{1\le i\le t}\sigma_{v_i}^2}\right). \tag{43}$$

Proof Note that

$$\max_{1\le i\le t}|u_i| = \max_{1\le i\le 2t} u_i,$$

where $u_{t+i} := -u_i$, for $1 \le i \le t$. By a union bound, we have

$$
\begin{aligned}
P\left(\max_{1\le i\le t}|u_i| > \delta_1\right) &= P\left(\bigcup_{1\le i\le 2t}\{u_i > \delta_1\}\right)\\
&\le \sum_{1\le i\le 2t} P\big(u_i \ge \delta_1\big)\\
&= \sum_{1\le i\le t} P\big(u_i \ge \delta_1\big) + \sum_{1\le i\le t} P\big(u_{t+i} \ge \delta_1\big)\\
&= \sum_{1\le i\le t} P\big(u_i \ge \delta_1\big) + \sum_{1\le i\le t} P\big(u_i \le -\delta_1\big).
\end{aligned}
$$

For each u_i, we have the tail bounds

$$
P(u_i > \delta_1) \le \exp\left(-\frac{\delta_1^2}{2\sigma_{u_i}^2}\right), \quad P(u_i < -\delta_1) \le \exp\left(-\frac{\delta_1^2}{2\sigma_{u_i}^2}\right).
$$

Altogether, we see that

$$
P\left(\max_{1\le i\le t}|u_i| > \delta_1\right) \le 2\sum_{1\le i\le t}\exp\left(-\frac{\delta_1^2}{2\sigma_{u_i}^2}\right) \le 2t\exp\left(-\frac{\delta_1^2}{2\max_{1\le i\le t}\sigma_{u_i}^2}\right).
$$

Similarly, we may obtain the desired concentration inequality for the v_i's:

$$
P\left(\max_{1\le i\le t}|v_i| > \delta_1\right) \le 2t\exp\left(-\frac{\delta_1^2}{2\max_{1\le i\le t}\sigma_{v_i}^2}\right).
$$

□

Lemma 2 *In the one-pool case, under the orthogonal design setting, suppose*

$$
\min_{1\le i\le t}|\gamma_i^*| > (\sqrt{2}\sqrt{\log t} + c_1)\max_{1\le i\le t}\sigma_{u_i} + n\lambda\left(1 + \max_{1\le i\le t}\frac{r_i^2}{f_i^2}\right), \tag{44}
$$

where $\sigma_{u_i} = \sqrt{1 + \frac{r_i^2}{f_i^2}}\sigma$. *Then the gamma-min condition holds with probability at least* $1 - 2e^{-c_1^2/2}$.

In the two-pool case, suppose

$$
\min_{1\le i\le t}|\gamma_i^*| > (\sqrt{2}\sqrt{\log t} + c_2)\max_{1\le i\le t}\sigma_{v_i} + n\lambda\left(1 + \max_{i\in[t]}\frac{r_i^2}{f_i^2 + \frac{\eta n}{m}w_i^2}\right), \tag{45}
$$

where $\sigma_{v_i} = \sqrt{1 + \frac{r_i^2(L^2 f_i^2 + \frac{\eta n}{m}w_i^2)}{L^2(f_i^2 + \frac{\eta n}{m}w_i^2)^2}}\sigma$. *Then the gamma-min condition holds with probability at least* $1 - 2e^{-c_2^2/2}$.

We use inequality (42) in Lemma 1. Let $\delta_1 = \sqrt{2\log t + c_1^2}\max_{1\le i\le t}\sigma_{u_i}$ where $c_1 \in (0, +\infty)$. Then with probability $1 - 2e^{-\frac{c_1^2}{2}}$, the following holds:

$$\max_{1\le i\le t}|u_i| \le \sqrt{2\log t + c_1^2}\max_{1\le i\le t}\sigma_{u_i} \le (\sqrt{2\log t}+c_1)\max_{1\le i\le t}\sigma_{u_i}.$$

In inequality (43), take $\delta_2 = \sqrt{2\log t + c_2^2}\max_{1\le i\le t}\sigma_{u_i}$ where $c_2 \in (0,+\infty)$. Then with probability $1-2e^{-\frac{c_2^2}{2}}$, the following holds:

$$\max_{1\le i\le t}|v_i| \le \sqrt{2\log t + c_2^2}\max_{1\le i\le t}\sigma_{v_i} \le (\sqrt{2\log t}+c_2)\max_{1\le i\le t}\sigma_{v_i}.$$

Combining these inequalities with conditions (40c) and (41c), we obtain $G \le \min_{i\in[t]}|\gamma_i^*|$ with probability at least $1-2e^{-\frac{c_1^2}{2}}$ or at least $1-2e^{-\frac{c_2^2}{2}}$. Specifically, when we choose $c_1 = c_2 = 2.72$, we can achieve a probability guarantee of at least 95% for the two statements.

Therefore, Proposition 6 is proved by plugging the results from Lemma 1 into Lemma 2. □

Here is the proof of Proposition 7.

Proof of Proposition 7 We will prove the proposition by comparing the three conditions in the two situations: adding one clean point and repeating multiple clean points. The conditions for adding one clean point are already provided in inequalities (41a), (41b) and (41c) above.

We now provide the conditions for repeating multiple clean points. The minimum eigenvalue condition becomes

$$\lambda_{\min}(P^{\perp}_{X',TT}) = \min_{1\le i\le t}\frac{f_i^2+\sum_{j=1}^{l_i}w_{ij}^2}{r_i^2+f_i^2+\frac{\eta n}{m}\sum_{j=1}^{l_i}w_{ij}^2} = \frac{1}{\max_{1\le i\le t}\frac{r_i^2}{f_i^2+\sum_{j=1}^{l_i}w_{ij}^2}+1}; \tag{46a}$$

the mutual incoherence condition becomes

$$\left\|P^{\perp}_{X',T^cT}(P^{\perp}_{X',TT})^{-1}\right\|_{\infty} = \max_{1\le i\le t}\left|\frac{r_if_i}{f_i^2+\frac{\eta n}{m}\sum_{j=1}^{l_i}w_{ij}^2}\right|; \tag{46b}$$

and the gamma-min condition becomes

$$\begin{aligned}\|\hat{\gamma}-\gamma^*\|_{\infty} \le \max_{1\le i\le t}&\left|\epsilon_i + \frac{r_if_i}{f_i^2+\frac{\eta n}{m}\sum_{j=1}^{l_i}w_{ij}^2}\epsilon_{i+t} + \sum_{j=1}^{k_i}\frac{r_iw_{ij}}{f_i^2+\frac{\eta n}{m}\sum_{j=1}^{l_i}w_{ij}^2}\frac{\epsilon_{i+t+p+j}}{L}\right| \\ &+ n\lambda\left(\max_{1\le i\le t}\{\frac{r_i^2}{f_i^2+\frac{\eta n}{m}\sum_{j=1}^{l_i}w_{ij}^2}\}+1\right).\end{aligned} \tag{46c}$$

Compared with inequlaities (41a), (41b) and (41c), conditions (46a), (46b) and (46c) replace w_i^2 by $\sum_{j=1}^{l_i}w_{ij}^2$. Suppose the scale of the clean data points is bounded by w_B. Then adding one data point may not be enough to satisfy the three conditions. Thus, to achieve the same effect of a large scaled $|w_i|$ in inequalities (41a), (41b) and (41c), we need the number of repeated clean points to be at least $\left(\frac{|w_i|}{w_B}\right)^2$. □

Table 1 Comparison between the two cases

Condition	One-pool case	Two-pool case
Eigenvalue	$\lambda_{\min}\left(P^{\perp}_{X,TT}\right) = b_{\min}$	$\lambda_{\min}\left(P^{\perp}_{X',TT}\right) = b'_{\min} \geq b_{\min}$
Mutual incoherence	$\| -X_{T^c}((n-t)\Sigma)^{-1}X_T^{\top}\|_{\infty}$	$\frac{\|-X_{T^c}((n-t)\Sigma)^{-1}X_T^{\top}\|_{\infty}}{1+\eta\frac{n}{n-t}}$
Gamma-min	$\min_i \lvert\gamma_i^*\rvert \geq \frac{2\sigma\sqrt{\log t}+n\lambda\sqrt{t}}{b_{\min}}$	$\min_i \lvert\gamma_i^*\rvert \geq \frac{2\sigma\sqrt{\log t}+n\lambda\sqrt{t}}{b'_{\min}}$

C.3 Sub-Gaussian design

In this section, we will present the support recovery results for sub-Gaussian design in Proposition 8 and Proposition 9, and the comparisons of the three conditions in the one- and two-pool cases in Table 1. Later, we will provide the proofs of the propositions.

C.3.1 Main results for sub-Gausian design

Proposition 8 *Suppose* $\{x_j\}_{j\in T^c}$ *and* $\{\widetilde{x}_i\}_{i\in[m]}$, *are i.i.d. sub-Gaussian with parameter* σ_x^2 *and covariance matrix* $\Sigma \succ 0$. *Further assume that* $\|X_T\|_2 \leq B_T$. *For the one-pool case, suppose we choose* λ *to satisfy inequality* (34) *and the sample size satisfies*

$$
\begin{aligned}
n > t + \max\Big\{ & p + C_1, \frac{4c_1^2\sigma_x^4(p+C_1)\|\Sigma\|_2^2}{\lambda^2_{\min}(\Sigma)}, \\
& \sqrt{t}\Big(\sqrt{p\|\Sigma\|_2} + c_2\sigma_2^2(\log n + \sqrt{p\log n})\Big)\left(1 + \frac{2c_1\sigma_x^2\|\Sigma\|_2}{\lambda_{\min}(\Sigma)}\right)\frac{B_T}{\lambda_{\min}(\Sigma)}\Big\},
\end{aligned}
\tag{47}
$$

then the contaminated pool achieves subset support recovery with probability at least $1 - e^{-\frac{c^2}{2}} - 2e^{-C_1} - n^{-(c_2-1)}$.

For the two-pool case, assume we choose λ *to satisfy* (36) *and the sample sizes satisfy*

$$
n > \max\left\{t+m, \frac{t}{1+\eta} + \frac{\sqrt{t}}{1+\eta}\Big(\sqrt{p\|\Sigma\|_2} + c_2\sigma_2^2(\log n + \sqrt{p\log n})\Big)\left(1 + \frac{2c_1\sigma_x^2\|\Sigma\|_2}{\lambda_{\min}(\Sigma)}\right)\frac{B_T}{\lambda_{\min}(\Sigma)}\right\}
\tag{48}
$$

and

$$
m \geq \max\{1, 4c_1^2\sigma_x^4\|\Sigma\|_2^2\}(p + C_1').
$$

Then adding clean points achieves subset support recovery with probability at least $1 - e^{-\frac{c'^2}{2}} - 2e^{-C_1'} - n^{-(c_2-1)}$.

As seen in Proposition 8, the number of data points n may be reduced by $1+\eta$ with the introduction of a second data pool.

Note that when T is randomly chosen from $[n]$, we have $B_T = O(\sqrt{t}\|\Sigma\|_2)$, so inequalities (47) and (48) require $\frac{t}{n}$ to be upper-bounded, and adding a second pool may weaker the upper bound to be $(1+\eta)$ than the upper bound for one-pool case.

We now present a result concerning exact support recovery:

Proposition 9 *In the one-pool case, suppose inequality* (47) *holds. If*

$$\min_{i\in T} |\gamma_i^*| \geq \frac{1}{b_{\min}}\left(2\sigma\sqrt{\log t + c} + \frac{2\sigma\sqrt{t}}{(1-\alpha)}\left(\sqrt{\log 2(n-t)} + C\right)\right), \tag{49}$$

then there exists a λ to achieve exact recovery with probability at least $1 - 2e^{-c} - e^{-\frac{C^2}{2}} - 2e^{-C_1} - n^{-C_2}$.

For the two-pool case, suppose the assumptions in Proposition 8 *hold, and*

$$\min_{i\in T} |\gamma_i^*| \geq \frac{1}{b'_{\min}}\left(2\sigma\sqrt{\log t + c} + \frac{2\sigma\sqrt{t}}{(1-\alpha')}\max\{1, \sqrt{\frac{\eta n}{mL}}\}\left(\sqrt{\log 2(n-t)} + C'\right)\right). \tag{50}$$

Then there exists a λ to achieve exact recovery with probability at least $1 - 2e^{-c} - e^{\frac{-C'^2}{2}} - 2e^{-C'_1} - n^{-C_2}$.

Compared to Propositions 8, 9 additionally requires the "signal-to-noise" ratio to be large enough. We can show that $b_{\min} \leq b'_{\min}$; thus, for an appropriate choice of η, the lower bound (49) is smaller than the bound (50), so the gamma-min condition is improved.

We now briefly compare the three conditions for the one- and two-pool cases in the random design setting.

In general, the eigenvalue condition is improved by adding a second pool. The mutual incoherence condition is improved in the two-pool case with large m by a constant multiplier $\frac{1}{1+\eta\frac{n}{m}}$ (≤ 1), and the gamma-min condition lower bound is improved by a constant $\frac{b_{\min}}{b'_{\min}}$ (≤ 1).

For the **eigenvalue condition**, the key result is that adding clean data points will not hurt, i.e., it makes the minimum eigenvalue smaller. A formal statement is provided in Proposition 10. Recall that

$$P^{\perp}_{X',TT} = I - X'_T(X'^{\top}X')^{-1}X'^{\top}_T,$$
$$P^{\perp}_{X,TT} = I - X_T(X^{\top}X)^{-1}X^{\top}_T,$$

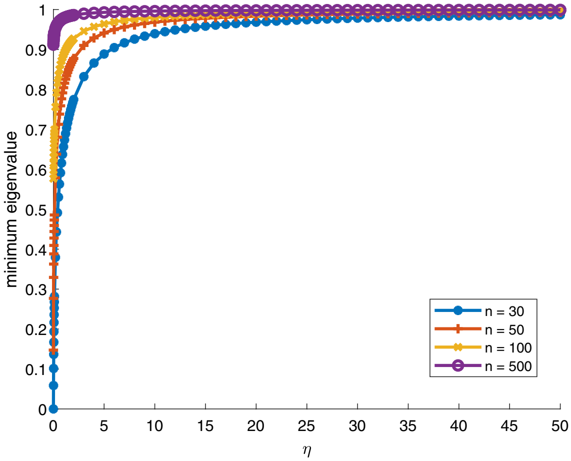
where $X' = \begin{pmatrix} X \\ \sqrt{\frac{\eta n}{m}}\widetilde{X} \end{pmatrix}$, and we assume that $X^{\top}X$ is invertible.

Proposition 10 (Comparison of minimum eigenvalue conditions) *We have*

$$\lambda_{\min}(P^{\perp}_{X',TT}) \geq \lambda_{\min}(P^{\perp}_{X,TT}).$$

Note that the result of Proposition 10 does not require any assumptions on $\widetilde{X}$ or η. However, the degree of improvement depends on η, as seen in the proof. Usually when n is small, increasing η leads to a big jump of the minimum eigenvalue; when n is large,

Fig. 10 How does η influence the minimum eigenvalue condition? The x-axis is the weight parameter η and the y-axis is $\lambda_{\min}(P^{\perp}_{X',TT})$. We take $t = 15, p = 20$, and $m = 5$, and vary n from 30 to 500. Both pools are drawn randomly from $\mathcal{N}(\mathbf{0}, \mathrm{I}_p)$



increasing η does not change the minimum eigenvalue much. A typical relationship between η and $\lambda_{\min}\left(P^{\perp}_{X',TT}\right)$ can be seen in Fig. 10.

For **mutual incoherence** condition, it is possible to find settings for small m that make the mutual incoherence condition worse. Consider the following example:

Example 3 *(Example where the mutual incoherence condition worsens)* Suppose

$$X_T = \begin{bmatrix} -1.8271 & -1.6954 & -1.1000 \\ 0.3020 & -1.4817 & -0.2284 \end{bmatrix},$$

$$X_{T^c} = \begin{bmatrix} -1.7680 & -0.0863 & 1.6822 \\ -0.5750 & -1.1013 & 0.4749 \\ -0.6693 & -0.6413 & 0.6126 \\ -0.3271 & 0.3060 & -1.0068 \\ 0.6177 & 0.3941 & -2.6407 \\ -0.7001 & 2.3465 & 0.4309 \end{bmatrix},$$

$$\widetilde{X} = \begin{bmatrix} -1.8722 & 0.5154 & 0.1560 \\ -0.9036 & 0.6064 & -0.2540 \end{bmatrix}.$$

Then

$$\|P^{\perp}_{X,T^cT}(P^{\perp}_{X,TT})^{-1}\|_{\infty} = 0.96 < 1 < \|P^{\perp}_{X',T^cT}(P^{\perp}_{X',TT})^{-1}\|_{\infty} = 1.28.$$

Despite this negative example, we can show that including a second pool helps when m is large compared to p. Recalling the assumption that $X_{T^c}^{\top} X_{T^c}$ is invertible, we can write

$$\begin{aligned}
P^{\perp}_{X,T^cT}(P^{\perp}_{X,TT})^{-1} &= -X_{T^c}\left(X_T^\top X_T + X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\Big(I - X_T\left(X_T^\top X_T + X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\Big)^{-1}\\
&= -X_{T^c}\left(X_T^\top X_T + X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\Big(I + X_T\left(X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\Big)\\
&= -X_{T^c}\left(X_{T^c}^\top X_{T^c}\right)^{-1}\left(X_T^\top X_T(X_{T^c}^\top X_{T^c})^{-1} + I\right)^{-1}\Big(I + X_T^\top X_T\left(X_{T^c}^\top X_{T^c}\right)^{-1}\Big)X_T^\top\\
&= -X_{T^c}(X_{T^c}^\top X_{T^c})^{-1}X_T^\top.
\end{aligned} \tag{51}$$

The first equality uses the definitions of $P^{\perp}_{X,T^cT}$ and $P^{\perp}_{X,TT}$, the second equality uses the Woodbury matrix identity (Henderson and Searle 1981), and the third equality follows from simple linear algebraic manipulations.

Similarly, we can simplify the mutual incoherence condition for the two-pool case, by replacing $X_{T^c}^\top X_{T^c}$ with $X_{T^c}^\top X_{T^c} + \eta\frac{n}{m}\widetilde{X}^\top\widetilde{X}$ in the inverse:

$$P^{\perp}_{X',T^cT}(P^{\perp}_{X',TT})^{-1} = -X_{T^c}\left(X_{T^c}^\top X_{T^c} + \eta\frac{n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1}X_T^\top, \tag{52}$$

where we know that $X_{T^c}^\top X_{T^c} + \eta\frac{n}{m}\widetilde{X}^\top\widetilde{X}$ must be invertible since $X_{T^c}^\top X_{T^c}$ is invertible.

Given these simplifications, it is easy to see that the difference between these two terms lies in the middle inverses. When m is large, we have $(X_{T^c}^\top X_{T^c})^{-1} \approx ((n-t)\Sigma)^{-1}$ and $\left(X_{T^c}^\top X_{T^c} + \eta\frac{n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1} \approx ((n-t+\eta n)\Sigma)^{-1}$, where Σ is the covariance matrix for the common distribution of X_{T^c} and $\widetilde{X}$. Therefore, the mutual incoherence parameter in the one-pool case is approximately equal to the mutual incoherence in the two-pool case scaled by $\left(1+\eta\frac{n}{n-t}\right)^{-1}$, which immediately implies that adding a second data pool improves the mutual incoherence condition. This is stated formally in the following proposition:

Proposition 11 (Comparison of mutual incoherence conditions) *Let $B_T = O(\sqrt{t})$. In the one-pool case, if $n \geq t + \frac{c_1^2\sigma_x^4(p+C_1)\|\Sigma\|^2}{\lambda^2_{\min}(\Sigma)}$,then*

$$\left|\left\|X_{T^c}\frac{\Theta}{n-t}X_T^\top\right\|_\infty - \left\|X_{T^c}\left(X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\right\|_\infty\right| = O\Big(t(n-t)^{-1}(\sqrt{p}+\sqrt{\log n})\Big),$$

with high probability.

In the two-pool case, if $n \geq t + \max\left\{\frac{c_1^2\sigma_x^4\|\Sigma\|^2}{\lambda^2_{\min}(\Sigma)}, 1\right\}m$ and $m \geq \max\{1, c_1^2\sigma_x^4(p+C_1')\|\Sigma\|_2^2\}$ $m \geq \max\{1, c_1^2\sigma_x^4(p+C_1')\|\Sigma\|_2^2\}$, then

$$\left|\left\|X_{T^c}\frac{\Theta}{n-t+\eta n}X_T^\top\right\|_\infty - \left\|X_{T^c}\left(X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1}X_T^\top\right\|_\infty\right| = O\Big(t(n-t+\eta n)^{-1}(\sqrt{p}+\sqrt{\log n})\Big),$$

with high probability.

Proposition 11 states that when m and n are sufficiently large, the one-pool mutual incoherence parameter is close to $\frac{\|X_{T^c}\Theta X_T^\top\|_\infty}{n-t}$ and the two-pool mutual incoherence parameter is close to $\frac{\|X_{T^c}\Theta X_T^\top\|_\infty}{n-t+\eta n}$. Since the second expression has a larger denominator, the mutual incoherence condition improves with the introduction of a second data pool with parameter $\eta > 0$.

For **gamma-min** condition, we need to compare the terms G and G'. Note that inequalities (49) and (50) are equivalent to lower-bounding the "signal-to-noise" ratio. The order of the lower bound for two-pool case is as same as the one-pool case, i.e., $\frac{\min_i |\gamma_i^*|}{\sigma} \geq O(\sqrt{t \log n})$. However, adding a second pool improves the constant by having a factor of $\frac{1}{b'_{\min}}$ instead of $\frac{1}{b_{\min}}$. As established in Proposition 10, we have $b_{\min} \leq b'_{\min}$. Therefore, the lower bound in the two-pool case is smaller than the lower bound in the one-pool case.

Note that the **weight parameter** η shows up in all the three conditions. However, recall that the mutual incoherence condition is not always improved by adding a second pool, unless m is sufficiently large. Therefore, an appropriate conclusion is that once we have a large clean data pool, it is reasonable to place arbitrarily large weight on the second pool. On the other hand, if we have fewer clean data points, we cannot be as confident about the estimator obtained using the second pool alone. For example, in the orthogonal design, if we obtain clean points in the non-buggy subspace, the mutual incoherence condition is not improved no matter how large we make η. In addition, the gamma-min condition involves the randomness from noise, and in order to control the sparsity of γ, we need the regularizer λ to match large η [cf. inequality (36)]. Based on inequality (50), we need the "signal-to-noise" ratio, i.e., $\frac{n\lambda\sqrt{t}}{\sigma}$, to be sufficient large. If η is too large, we cannot estimate relatively small components of γ^*. In summary, selecting η too large or too small is not wise: If η is too small, we do not improve the three conditions, whereas if η is too large, the range of controllable "signal-to-noise" ratios decays.

C.3.2 Proofs for sub-Gaussian design

In this section, we provide proofs of sub-Gaussian design. Here is the proof of Proposition 8.

Proof of Proposition 8 We prove the results for the one- and two-pool cases sequentially. In each case, we begin with background calculations, and then analyze the eigenvalue condition followed by the mutual incoherence condition.

For the one-pool case, we know that λ satisfies inequality (34) with probability at least $1 - e^{-\frac{c^2}{2}}$.

Note that x_j, $j \in T^c$ are sub-Gaussian random vectors with parameter σ_x. By Theorem 4.7.1 and Exercise 4.7.3 in Vershynin (2018) and our assumption of n, we have

$$\left\| \Sigma - \frac{X_{T^c}^\top X_{T^c}}{n-t} \right\|_2 \leq c_1 \sigma_x^2 \sqrt{\frac{p + C_1}{n-t}} \|\Sigma\|_2, \tag{53}$$

with probability at least $1 - e^{-C_1}$. We will later use this bound multiple times to establish the eigenvalue condition and the mutual incoherence condition.

We first consider the eigenvalue condition. By the dual Weyl's inequality (Horn and Johnson 1994), we have $\lambda_{\min}(A + B) \geq \lambda_{\min}(A) + \lambda_{\min}(B)$ for any square matrices A and B. Then

$$
\begin{aligned}
\lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right) &= \lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t} - \Sigma + \Sigma\right)\\
&\geq \lambda_{\min}(\Sigma) + \lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t} - \Sigma\right)\\
&\geq \lambda_{\min}(\Sigma) - \left\|\frac{X_{T^c}^\top X_{T^c}}{n-t} - \Sigma\right\|_2,
\end{aligned}
$$

where the second inequality follows from the fact that $\lambda_{\min}(A) \leq \lambda_{\max}(A)$ for any square matrix A. Combining this with inequality (53) and taking $n \geq t + 4\frac{c_1^2\sigma_x^4(p+C_1)\|\Sigma\|_2^2}{\lambda_{\min}^2(\Sigma)}$ by assumption (47), we have that

$$
\lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right) \geq \lambda_{\min}(\Sigma) - c_1\sigma_x^2\sqrt{\frac{p+C_1}{n-t}}\|\Sigma\|_2 \geq \frac{1}{2}\lambda_{\min}(\Sigma) > 0, \tag{54}
$$

with probability $1 - e^{-C_1}$. We now derive the following result:

Lemma 3 *Suppose $X_{T^c}^\top X_{T^c}$ is invertible, where $X_T \in \mathbb{R}^{t\times p}$ and $X_{T^c} \in \mathbb{R}^{(n-t)\times p}$. Then*

$$
\lambda_{\min}\left(P_{X,TT}^\perp\right) \geq 1 - \frac{\lambda_{\max}(X_T^\top X_T)}{\lambda_{\max}(X_T^\top X_T) + \lambda_{\min}(X_{T^c}^\top X_{T^c})} > 0,
$$

implying that the eigenvalue condition for the one-pool case holds.

Proof Define $C = Q(I + Q^\top Q)^{-1}Q^\top$ and $Q \in \mathbb{R}^{s\times p}$, and suppose $\text{rank}(Q) = r$. Let $Q = USV^\top$ be the SVD, where $U \in \mathbb{R}^{t\times p}$, $V \in \mathbb{R}^{p\times p}$, and $S = \begin{bmatrix} J_{r\times r} & 0_{r\times(p-r)}\\ 0_{(t-r)\times r} & 0_{(t-r)\times(p-r)}\end{bmatrix}$. Here, J is a diagonal matrix of positive singular values. Then

$$
\begin{aligned}
C &= USV^\top(I + VS^\top SV^\top)^{-1}VS^\top U^\top\\
&= US(I + S^\top S)^{-1}S^\top U^\top\\
&= U\begin{bmatrix} J_{r\times r} & 0_{r\times(p-r)}\\ 0_{(t-r)\times r} & 0_{(t-r)\times(p-r)}\end{bmatrix}\cdot\begin{bmatrix} (I+J^2)^{-1}_{r\times r} & 0_{r\times(p-r)}\\ 0_{(t-r)\times r} & I_{(p-r)\times(p-r)}\end{bmatrix}\cdot\begin{bmatrix} J_{r\times r} & 0_{r\times(p-r)}\\ 0_{(t-r)\times r} & 0_{(t-r)\times(p-r)}\end{bmatrix}U^\top\\
&= U\begin{bmatrix} (J(I+J^2)^{-1}J)_{r\times r} & 0_{r\times(p-r)}\\ 0_{(t-r)\times r} & 0_{(p-r)\times(p-r)}\end{bmatrix}U^\top.
\end{aligned}
\tag{55}
$$

Therefore, $\lambda_{\max}(C) = \frac{a_{\max}^2}{1+a_{\max}^2}$, where $a_{\max}$ is the maximum singular value appearing in J. Also note that $a_{\max}^2$ is the maximum eigenvalue of $Q^\top Q$.

Following (16.51) in Seber (2008), given $X_{T^c}^\top X_{T^c}$ is invertible, there exists a non-singular matrix A such that $AX_{T^c}^\top X_{T^c}A^\top = I$ and $AX_T^\top X_T A^\top = D$, where D is diagonal matrix.

Note that

$$\begin{aligned}X_T(X_T^\top X_T + X_{T^c}X_{T^c})^{-1}X_T^\top &= X_TA^\top(A(X_T^\top X_T + X_{T^c}X_{T^c})A^\top)^{-1}AX_T^\top\\ &= X_TA^\top(AX_T^\top X_TA^\top + I)AX_T^\top\\ &= Q(Q^\top Q + I)^{-1}Q^\top,\end{aligned}$$

where $Q := X_TA^\top$.

Based on our earlier arguments, we know that the matrix under consideration has maximum eigenvalue $\frac{\lambda_{\max}(AX_T^\top X_TA^\top)}{1+\lambda_{\max}(AX_T^\top X_TA^\top)}$. Since $AX_T^\top X_TA^\top$ is similar to $X_T^\top X_TA^\top A$, we have$\lambda_{\max}(AX_T^\top X_TA^\top) = \lambda_{\max}(X_T^\top X_TA^\top A)$. Furthermore, we have $A^\top A = (X_{T^c}^\top X_{T^c})^{-1}$, implying that

$$\begin{aligned}\lambda_{\max}(AX_T^\top X_TA^\top) &= \lambda_{\max}(X_T^\top X_T(X_{T^c}^\top X_{T^c})^{-1})\\ &\le \max_v \frac{\left\|X_T^\top X_T(X_{T^c}^\top X_{T^c})^{-1}v\right\|_2^2}{\left\|(X_{T^c}^\top X_{T^c})^{-1}v\right\|_2^2} \cdot \max_v \frac{\left\|(X_{T^c}^\top X_{T^c})^{-1}v\right\|_2^2}{\|v\|_2^2}\\ &\le \frac{\lambda_{\max}(X_T^\top X_T)}{\lambda_{\min}(X_{T^c}^\top X_{T^c})}.\end{aligned}$$

Altogether, we have

$$\begin{aligned}\lambda_{\max}\left(X_T\left(X_T^\top X_T + X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\right) &\le \frac{1}{1+\lambda_{\max}^{-1}\left(X_T^\top X_T(X_{T^c}^\top X_{T^c})^{-1}\right)}\\ &\le \frac{1}{1+\frac{\lambda_{\min}(X_{T^c}^\top X_{T^c})}{\lambda_{\max}(X_T^\top X_T)}}.\end{aligned} \tag{56}$$

Finally, we may conclude that

$$\begin{aligned}\lambda_{\min}\left(P_{X,TT}^\perp\right) &= \lambda_{\min}\left(I - X_T\left(X_T^\top X_T + X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\right)\\ &= 1-\lambda_{\max}\left(X_T\left(X_T^\top X_T + X_{T^c}^\top X_{T^c}\right)^{-1}X_T^\top\right)\\ &\ge 1-\frac{1}{1+\frac{\lambda_{\min}(X_{T^c}^\top X_{T^c})}{\lambda_{\max}(X_T^\top X_T)}}\\ &= 1-\frac{\lambda_{\max}(X_T^\top X_T)}{\lambda_{\max}(X_T^\top X_T)+\lambda_{\min}(X_{T^c}^\top X_{T^c})}.\end{aligned}$$

Since $\lambda_{\min}(X_{T^c}^\top X_{T^c}) > 0$, we have $\lambda_{\min}\left(P_{X,TT}^\perp\right) < 1$, implying the desired result. □

We now consider the mutual incoherence condition. By the triangle inequality, we have

$$\frac{1}{n-t}\left\|X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right)^{-1}X_T\right\|_\infty \le \underbrace{\frac{1}{n-t}\left\|X_{T^c}\Theta X_T^\top - X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right)^{-1}X_T^\top\right\|_\infty}_{①} + \underbrace{\frac{1}{n-t}\left\|X_{T^c}\Theta X_T^\top\right\|_\infty}_{②}.$$

We bound ① and ② separately. Note that

$$\begin{aligned} ① &= \frac{\max_{j\in T^c}\left\|x_j^\top\left(\Theta - \left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right)^{-1}\right)X_T^\top\right\|_1}{n-t} \\ &\le \frac{\sqrt{t}}{n-t}\max_{j\in T^c}\|x_j\|_2\left\|\Theta - \left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right)^{-1}\right\|_2\left\|X_T^\top\right\|_2. \end{aligned}$$

In order to bound ①, we bound three parts separately. By assumption, we have $\left\|X_T^\top\right\|_2 \le B_T$. For $\max_{j\in T^c}\|x_j\|_2$, we leverage the Hanson-Wright inequality (Theorem 6.2.1 in Vershynin (2018)) and a union bound. By the Hanson-Wright inequality, we see that for $t > 0$,

$$P\big(\|x_j\|_2^2 - \mathbb{E}[\|x_j\|_2^2] \ge t\big) \le \exp\left\{-c\min\left(\frac{t^2}{\sigma_x^4 p}, \frac{t}{\sigma_x^2}\right)\right\},$$

where c is an absolute constant.

By a union bound, we then have

$$\begin{aligned} P\left(\max_{j\in T^c}\|x_j\|_2 \ge \sqrt{\mathbb{E}[\|x_j\|_2^2] + \Delta}\right) &= P\left(\max_{j\in T^c}\|x_j\|_2^2 \ge \mathbb{E}[\|x_j\|_2^2] + \Delta\right) \\ &\le \sum_{j\in T^c} P\big(\|x_j\|_2^2 \ge \mathbb{E}[\|x_j\|_2^2] + \Delta\big) \\ &\le (n-t)\exp\left\{-c\min\left(\frac{\Delta^2}{\sigma_x^4 p}, \frac{\Delta}{\sigma_x^2}\right)\right\}. \end{aligned}$$

Setting $\Delta = c_2\sigma_x^2\max\{\sqrt{p\log n}, \log n\}$ with $c_2 \ge 1$ so that we have $\min\left\{\frac{\Delta^2}{\sigma_x^4 p}, \frac{\Delta}{\sigma_x^2}\right\} \ge c_2\log n$, we conclude that

$$\begin{aligned} \max_{j\in T^c}\|x_j\|_2 &\le \sqrt{\mathbb{E}[\|x_j\|_2^2] + \Delta} \\ &\le \sqrt{trace(\Sigma)} + \Delta \\ &\le \sqrt{p\|\Sigma\|_2} + c_2\sigma_x^2(\log n + \sqrt{p\log n}), \end{aligned} \tag{57}$$

with probability at least $1 - n^{-(c_2-1)}$, where $c_2 \ge \max\{2, 2/c\}$.

To bound $\left\|\Theta - \left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right)^{-1}\right\|_2$, note that for two matrices A and B, we have

$$\left\| A^{-1} - B^{-1} \right\|_2 \leq \frac{\|A - B\|_2}{\lambda_{\min}(A)\lambda_{\min}(B)}.$$

Combining this fact with inequalities (53) and (54), we obtain

$$\begin{aligned} \left\| \Theta - \left(\frac{X_{T^c}^\top X_{T^c}}{n-t} \right)^{-1} \right\|_2 &\leq \frac{\left\| \Sigma - \frac{X_{T^c}^\top X_{T^c}}{n-t} \right\|_2}{\lambda_{\min}(\Sigma)\lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t} \right)} \leq \frac{2\left\| \Sigma - \frac{X_{T^c}^\top X_{T^c}}{n-t} \right\|_2}{\lambda_{\min}^2(\Sigma)} \\ &\leq \frac{2c_1\sigma_x^2\sqrt{\frac{p+C_1}{n-t}}\|\Sigma\|_2}{\lambda_{\min}^2(\Sigma)}. \end{aligned} \tag{58}$$

Altogether, we obtain the bound

$$\textcircled{1} \leq \frac{\sqrt{t}}{n-t}\left(\sqrt{p\|\Sigma\|_2} + c_2\sigma_x^2(\log n + \sqrt{p\log n}) \right) \cdot \frac{2c_1\sigma_x^2\sqrt{\frac{p+C_1}{n-t}}\|\Sigma\|}{\lambda_{\min}^2(\Sigma)} B_T. \tag{59}$$

We now consider ②. Note that

$$\begin{aligned} \frac{\left\| X_{T^c}\Theta X_T^\top \right\|_\infty}{n-t} &= \frac{1}{n-t}\max_{j\in T^c} \|x_j^\top \Theta X_T^\top\|_1 \\ &\leq \frac{\sqrt{t}}{n-t}\max_{j\in T^c} \|x_j^\top\|_2\|\Theta\|_2\|X_T^\top\|_2 \\ &= \frac{\sqrt{t}}{n-t}\left(\sqrt{p\|\Sigma\|_2} + c_2\sigma_x^2(\log n + \sqrt{p\log n}) \right) \cdot \frac{1}{\lambda_{\min}(\Sigma)} B_T. \end{aligned} \tag{60}$$

Therefore,

$$\textcircled{1} + \textcircled{2} \leq \frac{\sqrt{t}}{n-t}\left(\sqrt{p\|\Sigma\|_2} + c_2\sigma_x^2(\log n + \sqrt{p\log n}) \right) \cdot \left(1 + \frac{2c_1\sigma_x^2\sqrt{\frac{p+C_1}{n-t}}\|\Sigma\|_2}{\lambda_{\min}(\Sigma)} \right) \frac{B_T}{\lambda_{\min}(\Sigma)}.$$

Finally, assuming n satisfies the bound (47), and taking a union bound over all the probabilistic statements appearing above, we conclude that the mutual incoherence condition holds with probability at least $1 - e^{-\frac{C^2}{2}} - 2e^{-C_1} - n^{-(c_2-1)}$. This concludes the proof.

For the two-pool case, we will use the following inequalities:

$$\left\| \Sigma - \frac{X_{T^c}^\top X_{T^c}}{n-t} \right\|_2 \leq c_1\sigma_x^2\sqrt{\frac{p+C_1'}{n-t}}\|\Sigma\|_2,$$

$$\left\| \Sigma - \frac{\widetilde{X}^\top \widetilde{X}}{m} \right\|_2 \leq c_1\sigma_x^2\sqrt{\frac{p+C_1'}{m}}\|\Sigma\|_2,$$

with probablity at least $1 - 2e^{-C_1'}$. Combining these inequalities and using the triangle inequality, we obtain

$$\begin{aligned}\left\|\Sigma - \frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right\|_2 &\le \frac{n-t}{n-t+\eta n}\left\|\Sigma - \frac{X_{T^c}^\top X_{T^c}}{n-t}\right\|_2 + \frac{\eta n}{n-t+\eta n}\left\|\Sigma - \frac{\widetilde{X}^\top\widetilde{X}}{m}\right\|_2\\ &\le c_1\sigma_x^2\|\Sigma\|_2\frac{n-t}{n-t+\eta n}\sqrt{\frac{p+C_1'}{n-t}} + c_1\sigma_x^2\|\Sigma\|_2\frac{\eta n}{n-t+\eta n}\sqrt{\frac{p+C_1'}{m}}\\ &\overset{n\ge t+m}{\le} 2c_1\sigma_x^2\|\Sigma\|_2\sqrt{\frac{p+C_1'}{m}},\end{aligned} \tag{61}$$

with probability at least $1-2e^{-C_1'}$.

Analogous to Lemma 3, we can conclude that if $X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}$ is invertible, the eigenvalue condition satisfies

$$\lambda_{\min}(P_{X',TT}^\perp) \ge 1 - \frac{\lambda_{\max}(X_T^\top X_T)}{\lambda_{\max}(X_T^\top X_T) + \lambda_{\min}\left(X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}\right)} > 0.$$

(This can be proved just by replacing $X_{T^c}^\top X_{T^c}$ with $X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}$ in the proof of Lemma 3.) However, since we further wish to bound the minimum eigenvalue from below by $\lambda_{\min}(\Sigma)/2$, to match the one-pool case and to be used in the proof for the mutual incoherence condition later, we will consider $X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}$ directly.

Note that

$$\begin{aligned}\lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right) &= \lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n} - \Sigma + \Sigma\right)\\ &\ge \lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n} - \Sigma\right) + \lambda_{\min}(\Sigma)\\ &\ge \lambda_{\min}(\Sigma) - \left\|\frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n} - \Sigma\right\|_2.\end{aligned}$$

Thus, if we choose $m \ge 4c_1^2\sigma_x^4(p+C_1')\|\Sigma\|_2^2$, we have

$$\lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right) \ge \frac{1}{2}\lambda_{\min}(\Sigma) > 0,$$

with probability at least $1-2e^{-C_1'}$.

We now consider the mutual incoherence condition. Similar to the derivation of inequality (58), we have that

$$\left\|\Theta-\left(\frac{X_{T^c}^\top X_{T^c}+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}\right\|_2 \le \frac{\left\|\Sigma-\frac{X_{T^c}^\top X_{T^c}+\eta\frac{n}{m}\widetilde{X}^\top\widetilde{X}}{(1+\eta)n-t}\right\|_2}{\lambda_{\min}(\Sigma)\lambda_{\min}\left(\frac{X_{T^c}^\top X_{T^c}+\eta\frac{n}{m}\widetilde{X}^\top\widetilde{X}}{(1+\eta)n-t}\right)}$$

$$\le 2c_1\sigma_x^2\frac{\|\Sigma\|_2}{\lambda_{\min}^2(\Sigma)}\sqrt{\frac{p+C_1'}{m}}.$$

Combining this with inequality (57), we obtain

$$\frac{\left\|X_{T^c}\Theta X_T^\top - X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c}+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}X_T^\top\right\|_\infty}{n-t+\eta n}$$

$$=\frac{\max_{j\in T^c}\left\|x_j^\top\left(\Theta-\left(\frac{X_{T^c}^\top X_{T^c}+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}\right)X_T^\top\right\|_1}{n-t+\eta n}$$

$$\le \frac{\sqrt{t}}{n-t+\eta n}\max_{j\in T^c}\|x_j\|_2\cdot\left\|\Theta-\left(\frac{X_{T^c}^\top X_{T^c}+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}\right\|_2\left\|X_T^\top\right\|_2$$

$$\le \frac{\sqrt{t}}{n-t+\eta n}\left(\sqrt{p\|\Sigma\|_2}+c_2\sigma_x^2(\log n+\sqrt{p\log n})\right)\cdot 2c_1\sigma_x^2\frac{\|\Sigma\|_2}{\lambda_{\min}^2(\Sigma)}\sqrt{\frac{p+C_1'}{m}}B_T.$$

Therefore, together with the triangle inequality and inequality (60), we can bound the mutual incoherence parameter as follows:

$$\frac{\left\|X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c}+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}X_T^\top\right\|_\infty}{n-t+\eta n}$$

$$\le \frac{\left\|X_{T^c}\Theta X_T^\top - X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c}+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}X_T^\top\right\|_\infty}{n-t+\eta n}+\frac{\left\|X_{T^c}\Theta X_T^\top\right\|_\infty}{n-t+\eta n}$$

$$\le \frac{\sqrt{t}}{n-t+\eta n}\left(\sqrt{p\|\Sigma\|_2}+c_2\sigma_x^2(\log n+\sqrt{p\log n})\right)\left(1+2c_1\sigma_x^2\frac{\|\Sigma\|_2}{\lambda_{\min}(\Sigma)}\sqrt{\frac{p+C_1'}{m}}\right)\frac{B_T}{\lambda_{\min}(\Sigma)}.$$

By the assumption on n in inequality (48), the mutual incoherence condition therefore holds with probability $1-e^{-\frac{c'^2}{2}}-2e^{-C_1'}-n^{-(c_2-1)}$. □

Here is the proof of Proposition 9.

Proof of Proposition 9 To achieve exact support recovery, we need all the three conditions to hold. The eigenvalue condition and the mutual incoherence condition have already been discussed in the analysis of subset support recovery in "Appendix 8", so it remains to analyze the gamma-min condition.

Recall that

$$G' = \|(P^{\perp}_{X',TT})^{-1}P^{\perp}_{X',T\cdot}\epsilon'\|_\infty + n\lambda\left\|(P^{\perp}_{X',TT})^{-1}\right\|_\infty.$$

To simplify notation, we define

$$A := \|(P^{\perp}_{X',TT})^{-1}P^{\perp}_{X',T\cdot}P^{\perp}_{X'}\epsilon'\|_\infty, \quad B := n\lambda\left\|(P^{\perp}_{X',TT})^{-1}\right\|_\infty.$$

We also define the random variables

$$Z_i := e_i^\top (P^{\perp}_{X',TT})^{-1}P^{\perp}_{X',T\cdot}P^{\perp}_{X'}\epsilon'.$$

Since $P^{\perp}_{X'}$ is a projection matrix and the maximum singular value of $P^{\perp}_{X',T\cdot}$ is smaller than the maximum singular value of $P^{\perp}_{X'}$'s, we have

$$\left\|(P^{\perp}_{X',TT})^{-1}P^{\perp}_{X',T\cdot}P^{\perp}_{X'}\right\|_2 \leq \left\|(P^{\perp}_{X',TT})^{-1}\right\|_2 \leq \left\|(P^{\perp}_{X',TT})^{-1}\right\|_2 \leq \frac{1}{b'_{\min}},$$

for all $i \in T$. Note that Z_i is a zero-mean sub-Gaussian random variable with parameter at most $\frac{\sigma}{b'_{\min}}$. By a sub-Gaussian tail bound, we then have

$$P\left(\max_{1\leq i\leq t}|Z_i| > \frac{\sigma}{b'_{\min}}\left(\sqrt{2\log t} + \Delta\right)\right) \leq 2e^{-\frac{\Delta^2}{2}}.$$

Therefore, with probability at least $1-2e^{-c}$, we have $A \leq \frac{2\sigma\sqrt{\log t + c}}{b'_{\min}}$. Note that $\|(P^{\perp}_{X',TT})^{-1}\|_\infty \leq \sqrt{t}\|(P^{\perp}_{X',TT})^{-1}\|_2 = \frac{\sqrt{t}}{b'_{\min}}$. We can then immediately obtain the bound $B \leq \frac{2n\lambda\sqrt{t}}{b'_{\min}}$.

Combined with the fact that $\lambda \geq \frac{2\sigma}{n(1-\alpha')}\max\left\{1, \sqrt{\frac{\eta n}{mL}}\right\}\left(\sqrt{\log 2(n-t)} + C'\right)$, we then obtain

$$G' \leq \frac{1}{b'_{\min}}\left(2\sigma\sqrt{\log t + c} + \frac{2\sigma\sqrt{t}}{(1-\alpha')}\max\left\{1, \sqrt{\frac{\eta n}{mL}}\right\}\left(\sqrt{\log 2(n-t)} + C'\right)\right).$$

Thus, as long as $\min_{i\in T}|\gamma_i^*|$ is greater than or equal to the RHS of the inequality above, the gamma-min condition holds with probability at least $1-2e^{-c}-e^{-\frac{C'^2}{2}}$. Consequently, the exact support recovery is achieved.

The proof of the one-pool case is similar as the proof of the two-pool case provided above, so we omit the details here. □

Here is the proof of Proposition 10

Proof Proof of Proposition 10

By the Sherman–Morrison–Woodbury formula (Henderson and Searle 1981), we have

$$\begin{aligned} &X_T\left(X^\top X + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1}X_T^\top \\ &\quad = X_T\left(X^\top X\right)^{-1}X_T^\top - \frac{\eta n}{m}X_T\left(X^\top X\right)^{-1}\widetilde{X}^\top(I + \frac{\eta n}{m}\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top)^{-1}\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top. \end{aligned} \tag{62}$$

We now state and prove two useful lemmas:

Lemma 4 *Assume $X^\top X$ is invertible. Define*

$$A := X_T\left(X^\top X\right)^{-1}\widetilde{X}^\top(I+\frac{\eta n}{m}\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top)^{-1}\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top.$$

Then $\lambda_{\min}(A) \geq 0$. Equality holds when $\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top$ is not full-rank.

Proof First note that since $X^\top X$ is invertible and $\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top \succ 0$, the matrix $I+\frac{\eta n}{m}\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top$ is invertible. Note that

$$\forall y \in \mathbb{R}^t \neq 0, \quad y^\top A y \geq 0,$$

so the minimum eigenvalue of A is nonnegative.

In order to study when the $\lambda_{\min}=0$, let $z=\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top y$. When $y \neq 0$ and $\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top$ is full-rank, we have $z \neq 0$. Thus, if $\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top$ is full-rank, we have $\lambda_{\min}(A) > 0$. When $y \neq 0$ and $\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top$ is not full-rank, there exists $y \neq 0$ such that $z = 0$, which causes $y^\top A y = 0$ and $\lambda_{\min}(A) = 0$. □

Lemma 5 *The following equations holds:*

$$\begin{aligned}
\lambda_{\min}(P^{\perp}_{X,TT}) =& 1-\lambda_{\max}(X_T\left(X^\top X\right)^{-1}X_T^\top),\\
\lambda_{\min}(P^{\perp}_{X',TT}) =& 1-\lambda_{\max}(X_T\left(X^\top X+\frac{\eta n}{m}\widetilde{X}\widetilde{X}^\top\right)^{-1}X_T^\top).
\end{aligned}$$

Proof Since $X_T\left(X^\top X\right)^{-1}X_T^\top$ is symmetric positive semidefinite, we can write $X_T\left(X^\top X\right)^{-1}X_T^\top = Q\Lambda Q^\top$, where Q is an orthogonal matrix and Λ is a diagonal matrix with nonnegative diagonals. Then

$$I - X_T\left(X^\top X\right)^{-1}X_T^\top = Q(I-\Lambda)Q^\top.$$

Furthermore, we have shown in inequality (56) that

$$\lambda_{\max}\left(X_T(X^\top X)^{-1}X_T^\top\right) \leq \frac{1}{1+\frac{\lambda_{\min}(X_{T^c}^\top X_{T^c})}{\lambda_{\max}(X_T^\top X_T)}}.$$

Hence, the maximum diagonal in Λ is upper-bounded by 1, and $I-\Lambda$ has all diagonal entries in the range [0, 1]. Thus, we have shown that $\min \operatorname{diag}(I-\Lambda) = \max(\operatorname{diag}(\Lambda))$, implying the conclusion of the lemma. □

Returning to the proof of the proposition, we have

$$\begin{aligned}
&\lambda_{\max}\left(X_T\left(X^\top X+\frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1}X_T^\top\right)\\
&\quad\leq \lambda_{\max}\left(X_T\left(X^\top X\right)^{-1}X_T^\top\right)\\
&\qquad - \frac{\eta n}{m}\lambda_{\min}\left((X_T\left(X^\top X\right)^{-1}\widetilde{X}^\top(I+\frac{\eta n}{m}\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top)^{-1}\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top\right)\\
&\qquad \overset{(i)}{\leq} \lambda_{\max}\left(X_T\left(X^\top X\right)^{-1}X_T^\top\right),
\end{aligned}$$

Here, (i) comes from the fact that

$$\lambda_{\min}\left(X_T\left(X^\top X\right)^{-1}\widetilde{X}^\top \cdot (I + \frac{\eta n}{m}\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top)^{-1}\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top\right) \geq 0,$$

which follows from Lemma 4. Furthermore, by Lemma 5, we have

$$\lambda_{\min}\left(P^{\perp}_{X',TT}\right) = 1 - \lambda_{\max}\left(X_T\left(X^\top X + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1}X_T^\top\right)$$

and

$$\lambda_{\min}\left(P^{\perp}_{X',TT}\right) = 1 - \lambda_{\max}\left(X_T\left(X^\top X + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}\right)^{-1}X_T^\top\right).$$

Altogether, we conclude that the minimum eigenvalue is at least improved by $\frac{\eta n}{m}\lambda_{\min}\left(X_T\left(X^\top X\right)^{-1}\widetilde{X}^\top(I + \frac{\eta n}{m}\widetilde{X}(X^\top X)^{-1}\widetilde{X}^\top)^{-1}\widetilde{X}\left(X^\top X\right)^{-1}X_T^\top\right)$. □

Here is the proof of Proposition 11.

Proof of Proposition 11 The proof leverages arguments from the proof of Proposition 8. The goal is to argue that when n and m are sufficiently large, the empirical quantities are close to their population-level versions. We will use Big-O notation to simplify our discussion.

As already stated in inequality (59), if $n \geq t + \frac{c_1^2\sigma_x^4\|\Sigma\|^2}{\lambda_{\min}^2(\Sigma)}(p + C_1)$, then

$$\frac{\left\|X_{T^c}\Theta X_T^\top - X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c}}{n-t}\right)^{-1}X_T^\top\right\|_\infty}{n-t}$$
$$\leq \frac{\sqrt{t}}{n-t}\left(\sqrt{p\|\Sigma\|_2} + c_2\sigma_x^2(\log n + \sqrt{p\log n})\right) \cdot \frac{2c_1\sigma_x^2\sqrt{\frac{p+C_1}{n-t}}\|\Sigma\|}{\lambda_{\min}^2(\Sigma)}B_T.,$$

with probability at least $1 - e^{-C_1} - n^{-1}$, where $c_2 > \max\{2, 2/c\}$.

Also for the two-pool case, if $n \geq t + \max\left\{\frac{c_1^2\sigma_x^4\|\Sigma\|^2}{\lambda_{\min}^2(\Sigma)}, 1\right\}m$ and $m \geq \max\{1, c_1^2\sigma_x^4(p + C_1')\|\Sigma\|_2^2\}$, we have

$$\frac{\left\|X_{T^c}\Theta X_T^\top - X_{T^c}\left(\frac{X_{T^c}^\top X_{T^c} + \frac{\eta n}{m}\widetilde{X}^\top\widetilde{X}}{n-t+\eta n}\right)^{-1}X_T^\top\right\|_\infty}{n-t+\eta n}$$
$$\leq \frac{\sqrt{t}}{n-t+\eta n}\left(\sqrt{p\|\Sigma\|_2} + c_2\sigma_x^2(\log n + \sqrt{p\log n})\right)\left(1 + 2c_1\sigma_x^2\frac{\|\Sigma\|_2}{\lambda_{\min}(\Sigma)}\sqrt{\frac{p+C_1'}{m}}\right)\frac{B_T}{\lambda_{\min}(\Sigma)},$$

with probability at least $1 - 2e^{-C_1'} - n^{-1}$, where c_2 is defined in the same way as above. Noting that $B_T \propto \sqrt{t}$ and using the triangle inequality, we conclude the proof. □

D Proofs for Sect. 4

In this section, we provide proofs and additional details for the results in Sect. 4. We will establish several auxiliary results in the process, which are stated and proved in "Appendix D.4". The flow of logic is outlined below:

Theorem 3 $\Leftarrow$ (Lemma 6, Lemma 12);
Lemma 6 $\Leftarrow$ Theorem 2;
Lemma 12 $\Leftarrow$ (Lemma 7, Lemma 11);
Lemma 11 $\Leftarrow$ (Lemma 8, Lemma 9);
Lemma 9 $\Leftarrow$ Lemma 7.
Corollary 1 $\Leftarrow$ (Theorem 3, Corollary 2).

We sometimes write $\hat{\gamma}(\lambda)$ to represent the estimator from Lasso-based debugging with tuning parameter λ.

D.1 Proof of Theorem 3

We will first argue that the algorithm will stop, and then argue that all bugs are identified correctly when the algorithm stops. Finally, we will take a union bound over all the iterations in the while loop to obtain a probabilistic conclusion.

Algorithm 1 stops: Note that if we have an iteration k such that $\hat{\lambda}^k > 2\lambda^*$ and $C = 0$, then the algorithm must stop after at most $\lfloor \log_2 \frac{\lambda^u}{\lambda^*} \rfloor$ iterations. Otherwise, we know that $C = 1$ for all iterations k such that $\hat{\lambda}^k \geq \lambda^*$. Thus, after $k = \lfloor \log_2 \frac{\lambda^u}{\lambda^*} \rfloor$ iterations, we have

$$\lambda^k = \frac{\lambda^u}{2^{\lfloor \log_2 \frac{\lambda^u}{\lambda^*} \rfloor}} \in \left[\frac{\lambda^u}{2^{\log_2 \frac{\lambda^u}{\lambda^*}}}, \frac{\lambda^u}{2^{\log_2 \frac{\lambda^u}{\lambda^*} - 1}} \right] = [\lambda^*, 2\lambda^*].$$

As established in Lemma 6, we know that all true bugs will be identified with such a value of λ^k, so the remaining points are $(X^{(k)}, y^{(k)}) = (X_{T^c}, y_{T^c})$. Also note that

$$\|P^{\perp}_{X_{T^c}} y_{T^c}\|_\infty = \|P^{\perp}_{X_{T^c}} (X_{T^c}\beta^* + \epsilon_{T^c})\|_\infty = \|P^{\perp}_{X_{T^c}} \epsilon_{T^c}\|_\infty.$$

Hence, by Lemma 12, we have

$$\|P^{\perp}_{X_{T^c}} \epsilon_{T^c}\|_\infty < \frac{5}{2} \frac{1}{\bar{c}} \sqrt{\log 2n}\, \hat{\sigma}.$$

Therefore, the stopping criteria takes effect and the algorithm stops.

Algorithm 1 correctly identifies all bugs: A byproduct of the preceding argument is that $\hat{\lambda} > \lambda^*$. By Theorem 1, we have $\text{supp}(\hat{\gamma}^k) \subseteq \text{supp}(\gamma^*)$. Now suppose we are at a stage where l of the t bugs are flagged, where $l \in \{0, 1, \ldots, t\}$.

If $l = t$, then $\bar{X} = X_{T^c}$. As argued preveiously, the algorithm stops with high probability. Hence, we output all of the bugs.

Otherwise, we have $l \leq t - 1$. Suppose this happens at the kth iteration. Then at least one bug remains in $(X^{(k)}, y^{(k)})$, and all the clean points are included. Let S denote the corresponding row indices of X and let γ^*_S denote the following subvector of γ^*. Since bugs still remain, we must have $\min_{i \in S} |\gamma^*_{S,i}| \geq \min_{i \in T} |\gamma^*_i|$. Furthermore,

$$\|P^{\perp}_{X^{(k)}} y^{(k)}\|_\infty = \|P^{\perp}_{X^{(k)}} (X^{(k)}\beta^* + \gamma^*_S + \epsilon_S)\|_\infty = \|P^{\perp}_{X^{(k)}} (\gamma^*_S + \epsilon_S)\|_\infty.$$

By Lemma 12, we have

$$\|P^{\perp}_{X^{(k)}}(\gamma^*_S + \epsilon_S)\|_\infty > \frac{5}{2}\frac{1}{c}\sqrt{\log 2n}\,\hat{\sigma},$$

implying that $C = 0$. Thus, the procedure proceeds to the $(k+1)^{\text{st}}$ iteration. If for all k such that $\hat{\lambda}^k \geq 2\lambda^*$, bugs still remain, then $\hat{\lambda}^k$ keeps shrinking until the $\lfloor \log_2 \frac{\lambda^u}{\lambda^*} \rfloor^{\text{th}}$ iteration. Then the tuning parameter must lie in the interval $(\lambda^*, 2\lambda^*]$, resulting in a value of $\hat{\gamma}$ such that $\text{supp}(\hat{\gamma}) = \text{supp}(\gamma^*)$.

Probability by union bound: Now we study the probability for this algorithm to output a value of $\hat{\gamma}$ that achieves exact recovery. Firstly, the algorithm stops as long as Lemmas 6 and 12 hold, which holds with probability at least $1 - \frac{3}{n-t} - 2\exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right)$.

Secondly, consider the argument that the algorithm correctly identifies all bugs. For each iteration, the events $\{C = 0$ if a bug still exists$\}$ and $\{C = 1$ if no bugs exist$\}$ hold as long as Lemmas 6 and 12 hold, which happens with probability at least $1 - \frac{3}{n-t} - 2\exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right)$. If the algorithm has K iterations, the probability that the algorithm flags all bugs is therefore at least $1 - \frac{3K}{n-t} - 2K\exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right)$ by a union bound. Since we have argued that $K \leq \log_2 \frac{\lambda^u}{\lambda(\sigma^*)}$, the desired statement follows.

D.2 Proof of Corollary 1

According to the PDW procedure, we can set $\hat{\gamma} = \mathbf{0}$, solve for $\hat{z}$ via the zero-subgradient equation, and check whether $\|\hat{z}\|_\infty < 1$, where $\hat{z}$ is a subgradient of $\|\hat{\gamma}\|_1$. The gradient of the loss function is equal to zero, which implies that

$$\hat{z} = \frac{1}{\lambda n}\|\bar{P}^\top P^{\perp}_{X'} y'\|_\infty.$$

Therefore, we see that $\|\hat{z}\|_\infty < 1$ for $\lambda > \frac{\|\bar{P}^\top P^{\perp}_{X'} y'\|_\infty}{n}$, which means the optimizer satisfies $\hat{\gamma} = \mathbf{0}$. Since $\lambda_u = \frac{2\|\bar{P}^\top P^{\perp}_{X'} y'\|_\infty}{n}$, the output with tuning parameter λ_u gives $\hat{\gamma}(\lambda_u) = 0$.

Note that

$$\|\bar{P}^\top P^{\perp}_{X'} y'\|_\infty = \|\bar{P}^\top \bar{P}\gamma^* + \bar{P}^\top P^{\perp}_{X'}\epsilon'\|_\infty \leq \|\bar{P}^\top \bar{P}\gamma^*\|_\infty + \|P^{\perp}_{X'}\epsilon'\|_\infty$$

by the triangle inequality. The second term is bounded by $2\max\{1, \sqrt{\frac{\eta n}{mL}}\}\sqrt{\log 2n}\,\sigma^*$ with probability at least $1 - \frac{1}{n}$, since $e_j^\top P^{\perp}_{X'}\epsilon'$ is Gaussian with variance at most $\max\{1, \sqrt{\frac{\eta n}{mL}}\}\sigma^*$. For the first term, we have

$$\begin{aligned}
\|\bar{P}^\top \bar{P}\gamma^*\|_\infty &= \left\|\bar{P}^\top \bar{P}\gamma^*\right\|_\infty \\
&\overset{(i)}{\leq} t\left\|\bar{P}^\top \bar{P}\right\|_{\max}\|\gamma^*_T\|_\infty \\
&\overset{(ii)}{\leq} t\|\gamma^*\|_\infty \\
&\leq \frac{Cc_\nu}{2}\sqrt{1 - c_t}\sqrt{\log 2n}\, n^{c_n + \frac{1}{2}}\sigma^*,
\end{aligned}$$

where (i) holds because $\|v^\top \gamma^*\|_1 = \sum_{i \in T} |v_i \gamma_i^*| \leq t\|v\|_\infty \|\gamma^*\|_\infty$ for any row v of the matrix $\bar{P}^\top \bar{P}$, and (ii) holds because $\bar{P}^\top \bar{P}$ is a submatrix of the projection matrix $P_{X'}^\perp$ and each entry of a projection matrix is upper-bounded by 1. Altogether, we obtain

$$\lambda_u \leq \left[\max\left\{1, \sqrt{\frac{\eta n}{mL}}\right\} \frac{2\sqrt{\log 2n}}{n} + \frac{Cc_v}{2}\sqrt{1-c_t}\sqrt{\log 2n}\, n^{c_n+\frac{1}{2}}\right]\sigma^*.$$

By a similar argument as in Theorem 3 and Corollary 2, we know that Algorithm 1 stops with at most $\log_2 \frac{\lambda_u}{\lambda(\sigma^*)}$ with probability at least $1 - \frac{1}{n-t}$. Hence,

$$\begin{aligned}
\log_2 \frac{\lambda_u}{\lambda(\sigma^*)} &= \log_2 \frac{\left[\max\{1, \sqrt{\frac{\eta n}{mL}}\} + \frac{Cc_v}{4}\sqrt{1-c_t} n^{c_n+\frac{3}{2}}\right] \frac{2\sqrt{\log 2n}}{n}\sigma^*}{\frac{4}{1-\alpha'}\sqrt{2\log 2n(1-c_t)}\frac{\|\bar{P}_{T^c}^\perp\|_2}{n}\sigma^*} \\
&\overset{(1)}{\leq} \log_2 \frac{\left[\max\{1, \sqrt{\frac{\eta n}{mL}}\} + \frac{C}{4} n^{c_n+\frac{3}{2}}\right] 2\sqrt{\log n}}{\frac{4}{1-\alpha'}\sqrt{2\log 2n}} \\
&\overset{(2)}{\leq} \log_2 \frac{\left[\max\{1, \sqrt{\frac{\eta n}{mL}}\} + \frac{C}{4} n^{c_n+\frac{3}{2}}\right]}{2} \\
&\leq c\left(\frac{3}{2} + c_n\right)\log_2 n + \max\left\{0, \frac{1}{2}\log_2 \frac{\eta n}{mL} - 1\right\},
\end{aligned}$$

where (1) comes from the fact that $\bar{P}_{T^c}^\perp$ is a submatrix of $P_{X'}^\perp$, which has spectral norm 1 when $n \geq t + p + 1$; and (2) holds because $1 - \alpha' < 1$. To illustrate that $\|\bar{P}_{T^c}^\perp\|_2 = 1$, note that it is sufficient to show $\|P_{X',T^cT^c}^\perp\|_2 = 1$ $P_{X',T^cT^c}^\perp$ is a principal matrix of $P_{X'}^\perp$. By interlacing theorem (Hwang 2004), we know that $\lambda_{\max}(P_{X',T^cT^c}^\perp)$ is no less than the $(t+1)^{\text{st}}$ largest eigenvalue of $P_{X'}^\perp$, which is a projection matrix and therefore has $n - p$ eigenvalues equal to 1. Thus, if $t + 1 \leq n - p$, i.e., $n \geq t + p + 1$, then $\|P_{X',T^cT^c}^\perp\|_2 = 1$.

Now that we have bounded the number of iterations, we consider probability that the statement holds. Note that ϵ' is sub-Gaussian and all the statements based on $\lambda(\sigma^*)$ hold with probability $1 - \frac{1}{n-t}$. Compared to Theorem 3, note that on each iteration, we have subset support recovery with probability $1 - \frac{1}{n-t}$; and on iteration $\log_2 \frac{\lambda_u}{\lambda(\sigma^*)}$, we have exact support recovery with probability $1 - \frac{1}{n-t}$. Thus, we conclude that Algorithm 1 outputs a value of $\widehat{\lambda}$ that achieves exact recovery with probability at least

$$1 - \frac{5\left(c\log_2 n + \max\left\{0, \frac{1}{2}\log_2 \frac{\eta n}{mL}\right\}\right)}{n-t} - 2\left(c\log_2 n + \max\left\{0, \frac{1}{2}\log_2 \frac{\eta n}{mL}\right\}\right)e^{-2\left(\frac{1}{2}-c_t-\nu\right)^2 n}.$$

Proof of Proposition 2

We consider the three cases in Appendices D.3.1, D.3.2, and D.3.3.

Let $\Sigma = \mathbb{E}[x_i x_i^\top]$ and $\Theta = \Sigma^{-1}$, and assume that $X^{(k)}$ corresponds to some X_S with rows indexed by S. Our goal is to prove that

$$\left\| \frac{X_S \Sigma^{-1} X_S^\top}{p} - I \right\|_{\max} \leq c \max \left\{ \sqrt{\frac{\log |S|}{p}}, \frac{\log |S|}{p} \right\}, \tag{63}$$

$$\left\| \frac{X_S \top X_S}{|S|} - \Sigma \right\|_2 \leq \frac{\lambda_{\min}(\Sigma)}{2}, \tag{64}$$

for at most $\log_2 \frac{\lambda_u}{\lambda^*}$ of such sets S. Note that $T^c \subseteq S \subseteq [n]$ holds with probability at least $1 - \frac{\log_2 \frac{\lambda_u}{\lambda^*}}{n-t}$.

D.3.1 Proof of Proposition 2 for Gaussian case

The spectral norm bound follows from standard results (Vershynin 2010), which holds for a fixed set S with probability at least $1 - e^{-|S|} \geq 1 - e^{-(n-t)}$. Note that Algorithm 1 runs for at most $\log_2 \frac{\lambda_u}{\lambda^*}$ iterations by Theorem 3. Taking a union bound over all sets S, we obtain an overall probability of $1 - \log_2 \frac{\lambda_u}{\lambda^*} e^{-(1-c_t)n} \geq 1 - e^{-\frac{n}{2} + \log \log_2 \frac{\lambda_u}{\lambda^*}}$.

We now consider (63). Define $z_i = \Theta^{1/2} x_i$ for $1 \leq i \leq n$, so that

$$X\Theta^{1/2} = \begin{bmatrix} -z_1^\top- \\ \dots \\ -z_n^\top- \end{bmatrix}.$$

We know the $\Theta^{1/2}x_i$'s are i.i.d. isotropic Gaussian random vectors. Hence, $z_i^\top z_i \sim \chi^2(p)$ satisfies

$$\frac{\|z_i\|_2^2}{p} - 1 \leq 4\sqrt{\frac{\log \frac{1}{\delta}}{p}},$$

with probability at least $1 - \delta$. Similarly, we can bound $z_k^\top z_k$ and $(z_i + z_k)^\top (z_i + z_k)$. Since $z_i^\top z_k = \frac{1}{2}[(z_i + z_k)^\top (z_i + z_k) - z_i^\top z_i - z_k^\top z_k]$, we then have

$$\frac{\langle z_i, z_k \rangle}{p} \leq 8\sqrt{\frac{\log \frac{1}{\delta}}{p}}, \quad \forall i \neq k,$$

with probability at least $1 - \delta$.

We now choose $\delta = \frac{1}{n^c}$ for some $c > 2$ and take a union bound over all n^2 entries of the matrix $X\Theta X^\top$, to obtain

$$\left\| \frac{X\Theta X^\top}{p} - I \right\|_{\max} \leq c \max \left\{ \sqrt{\frac{\log n}{p}}, \frac{\log n}{p}, \right\}$$

with probability at least $1 - \frac{1}{n^{c'-2}}$, where $c' > 2$.

Finally, note that for all $S \subseteq [n]$, we have

$$\left\|\frac{X_S\Theta X_S}{p}-I\right\|_{\max}\leq\left\|\frac{X\Theta X}{p}-I\right\|_{\max}.$$

D.3.2 Proof of Proposition 2 for sub-Gaussian case

By Lemma 14, inequality (64) holds for a fixed set S, with probability at least $1-e^{-c|S|}\geq 1-e^{-c(n-t)}$ for some $c>0$. Note that Algorithm 1 runs for at most $\log_2\frac{\lambda_u}{\lambda^*}$ iterations. We then take a union bound over the possible subsets $T^c\subseteq S\subseteq[n]$ to reach a probability of at least $1-\log_2\frac{\lambda_u}{\lambda^*}e^{-c(1-c_t)n}\geq 1-e^{-\frac{cn}{2}+\log\log_2\frac{\lambda_u}{\lambda^*}}$.

Next, we focus on verifying inequality (63). Assuming that the x_i's are independent random vectors and the components of the x_i's are independent of each other, our goal is to prove that

$$\left\|\frac{X\Theta X^\top}{p}-I\right\|_{\max}\lesssim\max\left\{\sqrt{\frac{\log n}{p}},\frac{\log n}{p}\right\},$$

w.h.p., where $\Sigma=\mathrm{Cov}(x_i)=\Theta^{-1}=:D^2$ is a diagonal matrix.

Define $z_i=D^{-1}x_i$. Since the z_i's are mutually independent with independent components, we know that the vector $g_{ij}=(z_{i1},...,z_{ip},z_{j1},...,z_{jp})^\top$, for $i\neq j$, also has independent components. Furthermore, the sub-Gaussian parameter of g_{ij} is bounded by $l_{\max}=\max_{q=1}^p\frac{K}{d_q^2}$, where K is the sub-Gaussian variance parameter of the x_i's. This is because for a unit vector u, we have

$$\begin{aligned}\mathbb{E}\left[e^{\lambda u^\top g_{ij}}\right]&=\Pi_{q=1}^p\mathbb{E}\left[e^{\lambda u_q z_{iq}}\right]\mathbb{E}\left[e^{\lambda u_{p+q}z_{jq}}\right]\\&=\Pi_{q=1}^p\mathbb{E}\left[e^{\lambda\frac{u_q}{d_q}x_{iq}}\right]\mathbb{E}\left[e^{\lambda\frac{u_{p+q}}{d_q}x_{jq}}\right]\\&\leq\Pi_{q=1}^p\mathbb{E}\left[e^{\lambda^2\frac{u_q^2}{2d_q^2}K}\right]\mathbb{E}\left[e^{\lambda^2\frac{u_{p+q}^2}{2d_q^2}K}\right]\\&=\mathbb{E}\left[e^{\sum_{q=1}^p\lambda^2\frac{u_q^2+u_{p+q}^2}{2d_q^2}K}\right]\\&\leq\mathbb{E}\left[e^{\sum_{q=1}^p(u_q^2+u_{p+q}^2)\frac{\lambda^2}{2}l_{\max}}\right]\\&=\mathbb{E}\left[e^{\frac{\lambda^2}{2}l_{\max}}\right].\end{aligned}$$

Since we have assumed that $\|\Sigma\|_2$ is bounded, the d_q's are all bounded for each q, so $l_{\max}$ is bounded, as well.

Now let $A=\begin{bmatrix}0_{p\times p}&I_{p\times p}\\0_{p\times p}&0_{p\times p}\end{bmatrix}$. By the Hanson-Wright inequality, with probability at least $1-\delta$, we have

$$\left|\frac{\langle z_i, z_j\rangle}{p}\right| = \frac{g_{ij}^\top A g_{ij}}{p} \leq c_1\sqrt{\frac{\log\frac{2}{\delta}}{p}}, \tag{65}$$

where c_1 is a constant related to $l_{\max}$.

Now applying the Hanson-Wright inequality to the vector z_i, we have

$$\left|\frac{\|z_i\|_2^2}{p} - \frac{\mathbb{E}[\|z_i\|_2^2]}{p}\right| \leq c_2 \max\left\{\sqrt{\frac{\log\frac{2}{\delta}}{p}}, \frac{\log\frac{2}{\delta}}{p}\right\}, \tag{66}$$

with probability at least $1-\delta$. Noting that $\mathbb{E}[\|z_i\|_2^2] = tr(\Theta\Sigma) = p$, we will finally have

$$\left|\frac{\|z_i\|_2^2}{p} - 1\right| \leq c_2 \max\left\{\sqrt{\frac{\log\frac{2}{\delta}}{p}}, \frac{\log\frac{2}{\delta}}{p}\right\}.$$

Plugging in $\delta = \frac{2}{n^3}$ and taking a union bound, we then conclude that

$$\left\|\frac{X\Theta X^\top}{p} - I\right\|_{\max} \leq 2\max\{c_1, c_2\}\max\left\{\sqrt{\frac{\log n}{p}}, \frac{\log n}{p}\right\},$$

with probability at least $1-\frac{2}{n}$.

D.3.3 Proof of Proposition 2 for convex concentration case

Recall the following definition:

Definition 2 *(Convex concentration property)* Let X be a random vector in $\mathbb{R}^d$. If for every 1-Lipschitz convex function $\varphi : \mathbb{R}^d \to \mathbb{R}$ such that $\mathbb{E}[\varphi(X)] < \infty$ and for every $t > 0$, we have

$$\mathbb{P}(|\varphi(X) - \mathbb{E}[\varphi(X)]| \geq t) \leq 2\exp(-t^2/K^2),$$

then X satisfies the convex concentration property with constant K.

Suppose x_i has the convex concentration property with parameter K. Note that

$$\begin{aligned}\left\|\frac{X\Theta X^\top}{p} - I\right\|_{\max} &= \max_{i,j}\left|e_i^\top\left(\frac{X\Theta X^\top}{p} - I\right)e_j\right| \\ &= \max_{i,j}\left|\frac{x_i^\top\Theta x_j}{p} - e_i^\top e_j\right|.\end{aligned}$$

By Lemma 13, we thus have the exponential tail bound

$$\mathbb{P}\left(\left|\frac{x_i^\top \Theta x_i}{p} - 1\right| \geq w\right) \leq 2\exp\left(-\frac{1}{C}\min\left\{\frac{w^2p^2}{2K^4\|\Theta\|_F}, \frac{wp}{K^2\|\Theta\|_2}\right\}\right),$$

for all $1 \leq i \leq p$, which implies that

$$\left|\frac{x_i^\top \Theta x_i}{p} - 1\right| \leq cK^2 \max\left\{\sqrt{\frac{\log\frac{2}{\delta}}{p}}, \frac{\log\frac{2}{\delta}}{p}\right\},$$

with probability at least $1-\delta$. Taking $\delta = 2/n^3$, we then obtain

$$\left|\frac{x_i^\top \Theta x_i}{p} - 1\right| \leq cK^2 \max\left\{\sqrt{\frac{\log n}{p}}, \frac{\log n}{p}\right\}, \tag{67}$$

with probability at least $1 - \frac{2}{n^3}$.

Now we consider the off-diagonals $\frac{x_i\Theta x_j}{p}$, for $i \neq j$. We first rewrite

$$\mathbb{P}\left(\left|\frac{x_i^\top \Theta x_j}{p}\right| \geq \Delta\right) = \mathbb{P}\left(\left|x_i^\top \frac{\Theta x_j}{\left\|\Theta x_j\right\|_2}\right| \geq \frac{\Delta p}{\left\|\Theta x_j\right\|_2}\right).$$

Conditioning on $\left\|\Theta x_j\right\|_2$ for some $w > 0$, we obtain

$$\begin{aligned}\mathbb{P}\left(\left|\frac{x_i^\top \Theta x_j}{p}\right| \geq \Delta\right) = {} & \mathbb{P}\left(\left|x_i^\top \frac{\Theta x_j}{\left\|\Theta x_j\right\|_2}\right| \geq \frac{\Delta p}{\left\|\Theta x_j\right\|_2} \,\middle|\, \left\|\Theta x_j\right\|_2 \geq w\right)\mathbb{P}\left(\left\|\Theta x_j\right\|_2 \geq w\right) \\ & + \mathbb{P}\left(\left|x_i^\top \frac{\Theta x_j}{\left\|\Theta x_j\right\|_2}\right| \geq \frac{\Delta p}{\left\|\Theta x_j\right\|_2} \,\middle|\, \left\|\Theta x_j\right\|_2 < w\right)\mathbb{P}\left(\left\|\Theta x_j\right\|_2 < w\right).\end{aligned}$$

Since we have a convex 1-Lipschitz function mapping from x_i to $x_i^\top \frac{\Theta x_j}{\|\Theta x_j\|_2}$, we can further upper-bound the probability using the convex concentration property:

$$
\begin{aligned}
\mathbb{P}\left(\left|\frac{x_i^\top \Theta x_j}{p}\right| \geq \Delta\right) &\leq \mathbb{P}\left(\left\|\Theta x_j\right\|_2 \geq w\right) + \mathbb{P}\left(\left|x_i^\top \frac{\Theta x_j}{\left\|\Theta x_j\right\|_2}\right| \geq \frac{\Delta p}{\left\|\Theta x_j\right\|_2} \,\middle|\, \left\|\Theta x_j\right\|_2 < w\right) \\
&\leq \mathbb{P}\left(\left\|x_j\right\|_2 \geq \frac{w}{\|\Theta\|_2}\right) + \mathbb{P}\left(\left|x_i^\top \frac{\Theta x_j}{\left\|\Theta x_j\right\|_2}\right| \geq \frac{\Delta p}{w}\right) \\
&\overset{(1)}{\leq} \mathbb{P}\left(\left\|x_j\right\|_2 - \mathbb{E}[\left\|x_j\right\|_2] \geq \frac{w}{\|\Theta\|_2} - \mathbb{E}[\left\|x_j\right\|_2]\right) + 2\exp\left(-\frac{\Delta^2 p^2}{w^2 K^2}\right) \\
&\overset{(2)}{\leq} \mathbb{P}\left(\left\|x_j\right\|_2 - \mathbb{E}[\left\|x_j\right\|_2] \geq \frac{w}{\|\Theta\|_2} - \sqrt{\mathbb{E}[\left\|x_j\right\|_2^2]}\right) + 2\exp\left(-\frac{\Delta^2 p^2}{w^2 K^2}\right) \\
&\overset{(3)}{\leq} 2\exp\left(-\frac{\left(\frac{w}{\|\Theta\|_2} - \sqrt{tr(\Sigma)}\right)^2}{K^2}\right) + 2\exp\left(-\frac{\Delta^2 p^2}{w^2 K^2}\right) \\
&\leq 2\exp\left(-\frac{\left(\frac{w}{\|\Theta\|_2} - \sqrt{p\|\Sigma\|_2}\right)^2}{K^2}\right) + 2\exp\left(-\frac{\Delta^2 p^2}{w^2 K^2}\right),
\end{aligned}
$$

where (1) and (3) use the convex concentration property and (2) uses Jensen's inequality. The last inequality assumes that $w \geq \sqrt{p\|\Sigma\|_2}$, can be guaranteed if we choose w sufficiently large.

Plugging $\Delta = c\max\left\{\frac{\log n}{p}, \sqrt{\frac{\log n}{p}}\right\}$ and $w = c'\left(\sqrt{p} + \sqrt{\log n}\right)$ into the above derivations, we then obtain

$$
\mathbb{P}\left(\left|\frac{x_i^\top \Theta x_j}{p}\right| \geq \Delta\right) \leq 2\exp\left(-\frac{c'' \log n}{K^2}\right) + 2\exp\left(-c''' \frac{\max\{(\log n)^2, p\log n\}}{(p+\log n)K^2}\right).
$$

If $p > \log n$, then $2\exp\left(-\frac{\max\{(\log n)^2, p\log n\}}{(p+\log n)K^2}\right) \leq 2\exp\left(-\frac{c'''' \log n}{K^2}\right)$; If $p \leq \log n$, then $2\exp\left(-\frac{\max\{(\log n)^2, p\log n\}}{(p+\log n)K^2}\right) \leq 2\exp\left(-\frac{c''''' \log n}{K^2}\right)$. Hence, we have

$$
\mathbb{P}\left(\left|\frac{x_i^\top \Theta x_j}{p}\right| \geq \Delta\right) \leq 2\exp\left(-C\log n\right).
$$

We can choose c and c' sufficiently large to ensure that $C > 2$. Combining this with inequality (67) using a union bound, we finally obtain the desired result.

D.4 Auxiliary lemmas

By Theorem 1, we have the following corollary:

Corollary 2 *For two data pools, suppose the eigenvalue and mutual incoherence conditions hold. Let* $\lambda \geq \lambda(\sigma^*)$. *Then with probability* $1 - \frac{1}{n-t}$, *we have* $\mathrm{supp}(\hat{\gamma}) \subseteq \mathrm{supp}(\gamma^*)$, *and*

$$
\left\|\hat{\gamma}(\lambda) - \gamma^*\right\|_\infty \leq G'(\lambda). \tag{68}
$$

Proof Recall that the rule for regularizer selection in Theorem 1 is

$$\lambda \geq \frac{2}{1-\alpha'} \left\| \bar{P}_{T^c}^\top \left(I - \bar{P}_T (\bar{P}_T^\top \bar{P}_T)^{-1} \bar{P}_T^\top \right) \frac{\epsilon'}{n} \right\|_\infty .$$

Note that $e_j^\top \bar{P}_{T^c}^\top \left(I - \bar{P}_T (\bar{P}_T^\top \bar{P}_T)^{-1} \bar{P}_T^\top \right) \frac{\epsilon'}{n}$ is sub-Gaussian with variance parameter $\max\{1, \frac{\eta n}{mL}\} \frac{\|\bar{P}_{T^c}^\perp\|_2^2 \sigma^{*2}}{n^2}$. We have

$$\max_{j \in T^c} \left| e_j^\top \bar{P}_{T^c}^\top \left(I - \bar{P}_T (\bar{P}_T^\top \bar{P}_T)^{-1} \bar{P}_T^\top \right) \frac{\epsilon'}{n} \right| \leq 4 \max\left\{ 1, \frac{\eta n}{mL} \right\} \sqrt{\log 2(n-t)} \frac{\|\bar{P}_{T^c}^\perp\|_2}{n} \sigma^{*2},$$

with probability at least $1 - \frac{1}{n-t}$. According to the definition of $\lambda(\sigma^*)$, we can further derive the bound for $\hat{\gamma}$, since

$$\|\hat{\gamma} - \gamma^*\|_\infty \leq \|(P_{X',TT}^\perp)^{-1} P_{X',T\cdot}^\perp \epsilon'\|_\infty + 2n\lambda(\sigma^*) \left\| (P_{X',TT}^\perp)^{-1} \right\|_\infty .$$

□

The following lemma suggests that if $\min_{i \in T} |\gamma_i^*| \geq G'(2\lambda^*)$, then $\mathrm{supp}(\hat{\gamma}(\lambda)) = \mathrm{supp}(\gamma^*)$ if we take $\lambda \in [\lambda^*, 2\lambda^*]$.

Lemma 6 *If* $\min_{i \in T} |\gamma_i^*| \geq G'(2\lambda^*)$, *then taking* $\lambda \in [\lambda^*, 2\lambda^*]$ *yields an estimator* $\hat{\gamma}(\lambda)$ *that satisfies* $\mathrm{supp}(\hat{\gamma}(\lambda)) = \mathrm{supp}(\gamma^*)$.

Proof According to Theorem 1, for a regularizer $\lambda \in [\lambda^*, 2\lambda^*]$, we have $\hat{\gamma}_{T^c} = 0$ and $\left\| \hat{\gamma}(\lambda) - \gamma^* \right\|_\infty \leq G'(\lambda)$. If $\min_{i \in T} |\gamma_i^*| \geq G'(2\lambda^*)$, then by the triangle inequality, we have

$$|\hat{\gamma}_i| > \min_{i \in T} |\gamma_i^*| - G'(\lambda) \geq G'(2\lambda^*) - G'(\lambda) \geq 0,$$

for all $i \in T$. □

We use X_S to represent some $X^{(k)}$ for $S \subseteq [n]$, as shown in Algorithm 3. In each loop of the algorithm, we know that the points in S^c all lie in T by the subset recovery result. Thus, $S \supseteq T^c$. Let $l = n - |S|$, and note that $0 \leq l \leq t$.

Lemma 7 *Suppose Assumption 4 holds. If* $\lambda_{\min}(\Sigma)$ *and* $\lambda_{\max}(\Sigma)$ *are bounded, then*

$$\left\| P_{X_S}^\perp - \left(1 - \frac{p}{n-l} \right) I \right\|_{\max} \leq C \frac{\max\{p, \sqrt{p \log(n-l)}, \log(n-l)\}}{n-l} .$$

Proof Using the notation $\Theta = \Sigma^{-1}$ and $\hat{\Sigma} = \frac{X_S^\top X_S}{|S|}$, we have

$$\begin{aligned} \left\| P_{X_S}^\perp - \left(1 - \frac{p}{|S|} \right) I_{|S| \times |S|} \right\|_{\max} &= \left\| X_S (X_S^\top X_S)^{-1} X_S^\top - \frac{p}{|S|} I \right\|_{\max} \\ &\leq \left\| \frac{X_S (\hat{\Sigma})^{-1} X_S^\top}{|S|} - \frac{X_S \Theta X_S^\top}{|S|} \right\|_{\max} + \left\| \frac{X_S \Theta X_S^\top}{|S|} - \frac{p}{|S|} I \right\|_{\max} . \end{aligned}$$

By assumption, we may bound the second term by

$$\left\| \frac{X_S \Theta X_S^\top}{|S|} - \frac{p}{|S|} I \right\|_{\max} \le \frac{p}{|S|} \cdot c \max \left\{ \sqrt{\frac{\log |S|}{p}}, \frac{\log |S|}{p} \right\} = \frac{c \max\{\sqrt{p \log |S|}, \log |S|\}}{|S|}.$$

For the first term, we have

$$\begin{aligned} \left\| \frac{X_S (\widehat{\Sigma})^{-1} X_S^\top}{|S|} - \frac{X_S \Theta X_S^\top}{|S|} \right\|_{\max} &= \frac{1}{|S|} \left\| X_S \Big((\widehat{\Sigma})^{-1} - \Theta \Big) X_S^\top \right\|_{\max} \\ &\le \left\| (\widehat{\Sigma})^{-1} - \Theta \right\|_2 \cdot \max_{1 \le i \le |S|} \frac{1}{|S|} \| X_S^\top e_i \|_2^2. \end{aligned}$$

We now have the bound

$$\begin{aligned} \left\| (\widehat{\Sigma})^{-1} - \Theta \right\|_2 &\le \frac{\frac{1}{2} \lambda_{\min}(\Sigma)}{\lambda_{\min}(\Sigma) \lambda_{\min}(\widehat{\Sigma})} \\ &\le \frac{\frac{1}{2} \lambda_{\min}(\Sigma)}{\lambda_{\min}(\Sigma)(\lambda_{\min}(\Sigma) - \frac{1}{2} \lambda_{\min}(\Sigma))} = \frac{1}{\lambda_{\min}(\Sigma)}, \end{aligned}$$

as well, where the second inequality holds by Weyl's Theorem (Horn and Johnson 1994): $\lambda(\widehat{\Sigma}) \ge \lambda(\Sigma) - \| \Sigma - \widehat{\Sigma} \|_2$. The basic idea for the first inequality is to use the multiplicativity of matrix norms to conclude that

$$\begin{aligned} \left\| A^{-1} - B^{-1} \right\|_2 &\le \left\| A^{-1} (A - B) B^{-1} \right\|_2 \\ &\le \left\| A^{-1} \right\|_2 \| A - B \|_2 \left\| B^{-1} \right\|_2 \\ &= \frac{\| A - B \|_2}{\lambda_{\min}(A) \cdot \lambda_{\min}(B)}. \end{aligned} \tag{69}$$

Hence, an upper bound on $\|A - B\|_2$—which we obtain from our assumptions—together with minimum eigenvalue bounds on A and B, implies an upper bound on $\left\| A^{-1} - B^{-1} \right\|_2$.

Finally, we have

$$\begin{aligned} \max_{1 \le i \le |S|} \frac{1}{|S|} \| X_S^\top e_i \|_2^2 &\le \max_{1 \le i \le |S|} \frac{1}{|S|} \cdot \frac{\| \Theta^{1/2} X_S^\top e_i \|_2^2}{\lambda_{\min}^2 (\Theta^{1/2})} \\ &= \frac{1}{\lambda_{\min}(\Theta)} \cdot \max_{1 \le i \le |S|} \frac{\| \Theta^{1/2} X_S^\top e_i \|_2^2}{|S|} \\ &= \lambda_{\max}(\Sigma) \cdot \max_{1 \le i \le |S|} \frac{e_i^\top X_S \Theta X_S^\top e_i}{|S|} \\ &\le \lambda_{\max}(\Sigma) \cdot \left\| \frac{X_S \Theta X_S^\top}{|S|} \right\|_{\max}. \end{aligned}$$

By assumption, we have

$$\left\| \frac{X_S \Theta X_S^\top}{p} - I \right\|_{\max} \leq c \max\left\{ \sqrt{\frac{\log |S|}{p}}, \frac{\log |S|}{p} \right\}.$$

Hence, rescaling and using the triangle inequality, we have

$$\left\| \frac{X_S \Theta X_S^\top}{|S|} \right\|_{\max} \leq \frac{p}{|S|} \left(\left\| \frac{X_S \Theta X_S^\top}{p} - I \right\|_{\max} + 1 \right) \leq \frac{p}{|S|} + \frac{p}{|S|} \max\left\{ \sqrt{\frac{\log |S|}{p}}, \frac{\log |S|}{p} \right\}.$$

Altogether, we have the bound

$$\left\| \frac{X_S (\hat{\Sigma})^{-1} X_S^\top}{|S|} - \frac{X_S \Theta X_S^\top}{|S|} \right\|_{\max} \leq \frac{\lambda_{\max}(\Sigma)}{\lambda_{\min}(\Sigma)} \cdot \frac{p}{|S|} \left(1 + \max\left\{ \sqrt{\frac{\log |S|}{p}}, \frac{\log |S|}{p} \right\} \right).$$

Finally, we have

$$\frac{c \max\{ \sqrt{p \log |S|}, \log |S| \}}{|S|} + c'' \frac{p}{|S|} \left(1 + \max\left\{ \sqrt{\frac{\log |S|}{p}}, \frac{\log |S|}{p} \right\} \right) \leq C \frac{\max\{p, \sqrt{p \log |S|}, \log |S| \}}{|S|}.$$

This finishes the proof. □

We use $\alpha(k)$ to represent the kth order statistics of $|\epsilon_i|$, for $i \in T^c$, where $\alpha_{(1)} \leq \alpha_{(2)} \leq \cdots \leq \alpha_{(n-t)}$.

Lemma 8 *For i.i.d. random variables* $\{|\epsilon_i|\}_{i \in T^c}$*, the kth order statistics, for any* $k \in \{\frac{n-t}{2}, \ldots, \frac{n}{2}\}$ *satisfy*

$$c_\nu \sigma^* \leq \alpha(k) \leq C_\nu \sigma^*,$$

with probability at least $1 - 2\exp\left(-2\left(\frac{1}{2} - c_t - \nu \right)^2 n \right)$*, for* $\nu \in (0, \frac{1}{2})$ *such that* $\nu < \frac{1}{2} - c_t$.

Proof By the assumptions on the noise distribution, we have

$$\nu = \mathbb{P}\left[|\epsilon_i| \leq c_\nu \sigma^* \right] \text{ and } \nu = \mathbb{P}\left[|\epsilon_i| \geq C_\nu \sigma^* \right].$$

Let ξ_i's be i.i.d. Bernoulli variables such that

$$\xi_i = \begin{cases} 1 & \text{if } |\epsilon_i| \leq c_\nu \sigma^*, \\ 0 & \text{otherwise.} \end{cases}$$

Note that $t = c_t n$ for some positive constant $c_t \in (0, \frac{1}{2})$. We have

$$k - \nu(n-t) \geq \frac{n-t}{2} - \nu(n-t) = \frac{(1-c_t)(1-2\nu)}{2}n > 0$$

and

$$\left(\frac{k}{n-t} - \nu\right)^2 (1-c_t) \geq \left(\frac{1}{2} - \nu\right)^2 (1-c_t) \geq \left(\frac{1-2\nu}{2}\right)\left(\frac{1-c_t-2\nu}{2}\right).$$

By Hoeffding's inequality (Hoeffding 1994), we then obtain

$$\begin{aligned}\mathbb{P}\left[\sum_{i=1}^{n-t} \xi_i \geq k\right] &= \mathbb{P}\left[\sum_{i=1}^{n-t} \xi_i - \nu(n-t) \geq k - \nu(n-t)\right] \\ &\leq \exp\left(-2\left(\frac{k}{n-t} - \nu\right)^2 (n-t)\right) \\ &\leq \exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right),\end{aligned}$$

implying that

$$\mathbb{P}\left[\alpha(k) \leq c_\nu \sigma^*\right] = \mathbb{P}\left[\sum_{i=1}^{n} \xi_i \geq k\right] \leq \exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right).$$

Similarly, let η_i's be i.i.d. Bernoulli variables such that

$$\eta_i = \begin{cases} 1 & \text{if } |\epsilon_i| \geq C_\nu \sigma^*, \\ 0 & \text{otherwise.} \end{cases}$$

Note that the assumption that $c_t < \frac{1}{2} - \nu$ gives us

$$n - t - k - \nu(n-t) > n - c_t n - \frac{n}{2} - \nu(1-c_t)n \geq \left(\frac{1}{2} - c_t - \nu\right)n > 0,$$

and

$$\left(1 - \frac{k}{n-t} - \nu\right)^2 (1-c_t) \geq \left(\frac{1}{2} - c_t - \nu\right)^2 \frac{n}{n-t} \geq \left(\frac{1}{2} - c_t - \nu\right)^2.$$

Then by Hoeffding inequality, we obtain

$$\begin{aligned}\mathbb{P}\left[\sum_{i=1}^{n-t} \eta_i \geq n - t - k\right] &= \mathbb{P}\left[\sum_{i=1}^{n-t} \eta_i - \nu(n-t) \geq n - t - k - \nu(n-t)\right] \\ &\leq \exp\left(-2\left(1 - \frac{k}{n-t} - \nu\right)^2 (n-t)\right) \\ &\leq \exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right),\end{aligned}$$

so that

$$\mathbb{P}\left[\alpha(k) \geq C_\nu \sigma^*\right] \leq \exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right).$$

□

Lemma 9 *Suppose the assumptions of Lemma 7 hold and*

$$n^{1-2c_n} \geq \max\left\{\frac{32C^2}{1-c_t}\log(2n)\,(p^2 + \log^2 n), \quad \left(\frac{24}{c_\nu}\right)^{\frac{1}{c_n}}\right\},$$

and

$$\max_{i\in S} |\gamma_S^*| \leq \frac{c_\nu C}{2}\sqrt{1-c_t}\sqrt{\log 2n}\,\frac{n^{1/2+c_n}}{t}\sigma^*,$$

for some constant $c_n \in (0, \frac{1}{2})$. *Then the kth order statistic of* $|P_{X_S}^{\perp}(\gamma_S^* + \epsilon_S)|$ *and the kth order statistic of* $\left|\left(1 - \frac{p}{|S|}\right)(\gamma_S^* + \epsilon_S)\right|$ *have differences of at most* $\frac{\bar{c}}{4}\sigma^*$, *for any* $k \in [|S|]$, *with probability at least* $1 - \frac{1}{n-t}$.

Proof Recall that $l = n - |S|$. Now consider the sequences $\{z_i = |e_i^\top P_{X_S}^{\perp}(\gamma_S^* + \epsilon_S)|\}_{i=1}^{n-l}$ and $\left\{w_i = \left|\left(1 - \frac{p}{n-l}\right)(\gamma_{S,i}^* + \epsilon_{S,i})\right|\right\}_{i=1}^{n-l}$. By the triangle inequality, we have

$$\begin{aligned}|z_i - w_i| &\leq \left|e_i^\top\left(P_{X_S}^{\perp} - \left(1 - \frac{p}{n-l}\right)I\right)(\gamma_S^* + \epsilon_S)\right| \\ &\leq \left|\underbrace{e_i^\top\left(P_{X_S}^{\perp} - \left(1 - \frac{p}{n-l}\right)I\right)\gamma_S^*}_{v_i}\right| + \left|\underbrace{e_i^\top\left(P_{X_S}^{\perp} - \left(1 - \frac{p}{n-l}\right)I\right)\epsilon_S}_{u_i}\right|,\end{aligned}$$

for $i = 1, \ldots, n-l$.

Since u_i is sub-Gaussian with parameter at most $\left\|(P_{X_S}^{\perp})_{i\cdot} - e_i^\top\left(1 - \frac{p}{n-l}\right)\right\|_2^2 \sigma^{*2}$, we can upper-bound the maximum of $\{|u_i|\}$. With probability at least $1 - \frac{1}{n-t}$, we have

$$\begin{aligned}\max_{i\in S} |u_i| &\leq 2\sqrt{\log 2(n-l)}\sigma^*\left\|(P_{X_S}^{\perp})_{i\cdot} - e_i^\top\left(1 - \frac{p}{n-l}\right)\right\|_2 \\ &\leq 2\sqrt{\log 2(n-l)}\sigma^*\sqrt{n-l}\left\|P_{X_S}^{\perp} - \left(1 - \frac{p}{n-l}\right)\right\|_{\max} \\ &\leq 2C\sqrt{\log 2(n-l)}\frac{(\sqrt{p} + \sqrt{\log(n-l)})^2}{\sqrt{n-l}}\sigma^*,\end{aligned}$$

where the last inequality follows by Lemma 7. Further note that since $n^{1-2c_n} \geq \frac{32C^2}{1-c_t}\log(2n)\,(p^2 + \log^2 n)$ for some $c_n \in (0, \frac{1}{2})$, we have $\max_{i\in S}|u_i| \leq \frac{1}{n^{c_n}}\sigma^*$.

For the v_i's, we have

$$\begin{aligned}\max_{i\in S}|v_i| &\overset{(i)}{\le} t\left\|P^{\perp}_{X_S} - \left(1-\frac{p}{n-l}\right)\right\|_{\max} \max_{i\in S}|\gamma^*_S| \\ &\overset{(ii)}{\le} \sqrt{\frac{t^2}{n(1-c_t)}}\frac{(\sqrt{p}+\sqrt{\log(n-l)})^2}{\sqrt{n-l}}\max_{i\in S}|\gamma^*_S| \\ &\overset{(iii)}{\le} \frac{1}{2C}\sqrt{\frac{1}{1-c_t}}\frac{t}{n^{1/2+c_n}}\frac{1}{\sqrt{\log 2n}}\max_{i\in S}|\gamma^*_S|,\end{aligned} \tag{70}$$

where (i) holds because $|a^\top\gamma^*_S| \le \|a\|_\infty\|\gamma^*_S\|_\infty|\text{supp}(\gamma^*_S)|$ for any vector a, (ii) holds by Lemma 7, and (iii) holds by our assumption on n. Combining this with the assumption that $\max_{i\in S}|\gamma^*_S| \le \frac{c_v C}{4}\sqrt{1-c_t}\sqrt{\log 2n}\,\frac{n^{1/2+c_n}}{t}\sigma^*$, we obtain $\max_{i\in S}|v_i| \le \frac{c_v}{8}\sigma^*$. Finally, using the fact that $n \ge \left(\frac{24}{c_v}\right)^{\frac{1}{c_n}}$, we obtain

$$|z_i - w_i| \le \frac{c_v}{6}\sigma^*,$$

with probability at least $1-\frac{1}{n-t}$.

We then use the following lemma:

Lemma 10 *For two sequences $a_1,\dots,a_n$ and $b_1,\dots,b_n$ such that $|a_i-b_i| \le c$ for some positive number c, the jth order statistics of $\{a_i\}$ and $\{b_i\}$, denoted by $\alpha_a(j)$ and $\alpha_b(j)$, satisfy*

$$|\alpha_a(j) - \alpha_b(j)| \le c. \tag{71}$$

Proof Without loss of generality, suppose $a_1 \le a_2 \le \cdots \le a_n$. If there exists $j \in [n]$ such that inequality (71) does not hold, then we have either $a_j > c + \alpha_b(j)$ or $a_j < \alpha_b(j) - c$. If the first case occurs, we have

$$a_n \ge \cdots \ge a_j > c + \alpha_b(j) \ge c + \alpha_b(j-1) \ge \cdots c + \alpha_b(1).$$

Pick a number z between $c+\alpha_b(j)$ and a_j. We see that at least j of the b_i's, denoted by $\mathbf{b}_\downarrow$, are smaller than $z-c$; and at least $n-j+1$ of a_i's, denoted by $\mathbf{a}_\uparrow$, are greater than z. This means that at most $j-1$ of a_i's are no larger than z. Note that for the $\mathbf{b}_\downarrow$, the components of the corresponding vector $\mathbf{a}_\downarrow$ are within a distance of c, so the elements of $\mathbf{a}_\downarrow$ must be at most z. However, this contradicts the fact that at most $j-1$ of the a_i's are at most z. This concludes the proof. □

From Lemma 10, we can compare the order statistics of sequences $\{z_i\}_{i=1}^n$ and $\{w_i\}_{i=1}^n$ and conclude that they have differences of at most $\frac{\bar{c}}{6}\sigma^*$, with probability at least $1-\frac{1}{n-t}$. □

Lemma 11 *Suppose the conditions of Lemmas 8 and 9 hold, and also* $\min_{i\in T}|\gamma^*_i| > 4\sqrt{\log(2n)}\sigma^*$. *Then*

$$\left(c_v - \frac{|S|}{|S|-p}\frac{c_v}{6}\right)\sigma^* \le \hat{\sigma} \le \left(\frac{|S|}{|S|-p}\frac{c_v}{6} + C_v\right)\sigma^*,$$

with probability at least $1 - 2\exp\left(-2\left(\frac{1}{2}-c_t-v\right)^2 n\right) - \frac{2}{n-t}$.

Proof Let $M_P(S)$ denote the median of $|P^{\perp}_{X_S}(\gamma^*_S + \epsilon_S)|$. By Lemma 9, we know that $M_P(S)$ is close to the median of $\left|\left(1 - \frac{p}{|S|}\right)(\gamma^*_S + \epsilon_S)\right|$. Thus, it remains to analyze the median of $\{|\gamma^*_i + \epsilon_i|\}_{i \in S}$.

Note that for $j \in T^c$, we have $|\gamma^*_j + \epsilon_j| = |\epsilon_j|$. Therefore, for all $j \in S \cap T^c = T^c$, we have $|\gamma^*_j + \epsilon_i|_\infty \leq 2\sqrt{\log 2n}\,\sigma^*$, with probability at least $1 - \frac{1}{n}$.

For $i \in T \cap S$, by the assumption that $\min_{i \in T} |\gamma^*_i| > 4\sqrt{\log 2n}\,\sigma^*$, we have $|\gamma^*_i + \epsilon_i| \geq |\gamma^*_i| - |\epsilon_i| > 2\sqrt{\log 2n}\,\sigma^*$. Therefore, the median of $|\gamma^*_S + \epsilon_S|$ is actually the *kth* order statistics of $|\epsilon_{T^c}|$ for some $\{k \in \frac{n-t}{2}, \ldots, \frac{n}{2}\}$. By Lemma 9, we have

$$\left(1 - \frac{p}{|S|}\right)\alpha(k) - \frac{c_\nu}{6}\sigma^* \leq M_P(S) \leq \left(1 - \frac{p}{|S|}\right)\alpha(k) + \frac{c_\nu}{6}\sigma^*.$$

In Algorithm 1, at some iteration k, we have $\hat{\sigma} = \frac{|S|}{|S|-p} M_P(S)$, where S is the corresponding set of indices of $\left(\mathrm{supp}(\hat{\gamma}^{(k)})\right)^c$. Thus,

$$\alpha(k) - \frac{|S|}{|S|-p}\frac{c_\nu}{6}\sigma^* \leq \hat{\sigma} \leq \alpha(k) + \frac{|S|}{|S|-p}\frac{c_\nu}{6}\sigma^*.$$

Combining this with Lemma 8, we have

$$\left(c_\nu - \frac{|S|}{|S|-p}\frac{c_\nu}{6}\right)\sigma^* \leq \hat{\sigma} \leq \left(\frac{|S|}{|S|-p}\frac{c_\nu}{6} + C_\nu\right)\sigma^*,$$

with probability at least $1 - 2\exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right) - \frac{2}{n-t}$. □

Lemma 12 *Suppose* $n \geq 12p$,

$$\min_{i \in T} |\gamma^*_i| \geq \frac{5}{4}\left(\frac{c_\nu + 5C_\nu}{\bar{c}}\right)\sqrt{\log 2n}\,\sigma^*,$$

and inequality (70) *holds. Then*

$$\|P^{\perp}_{X_{T^c}}\epsilon_{T^c}\|_\infty < \frac{5}{2\bar{c}}\sqrt{\log 2n}\hat{\sigma}, \tag{72}$$

and for any γ^*_S *such that* $S \cap T \neq \emptyset$, *we have*

$$\|P^{\perp}_{X_S}(\gamma^*_S + \epsilon_S)\|_\infty > \frac{5}{2\bar{c}}\sqrt{\log 2n}\hat{\sigma}, \tag{73}$$

with probability at least $1 - \frac{3}{n-t} - 2\exp\left(-2\left(\frac{1}{2} - c_t - \nu\right)^2 n\right)$.

Proof We first establish the bound on $\|P^{\perp}_{X_{T^c}}\epsilon_{T^c}\|_\infty$. Note that $e_j^\top P^{\perp}_{X_{T^c}}\epsilon_{T^c}$ is Gaussian with variance at most $\max_{j \in T^c}(P^{\perp}_{X_{T^c}})_{jj}$, so

$$\|P^{\perp}_{X_{T^c}}\epsilon_{T^c}\|_\infty = \max_{j \in T^c} |e_j^\top P^{\perp}_{X_{T^c}}\epsilon_{T^c}| \leq \max_j (P^{\perp}_{X_{T^c}})_{jj} 2\sqrt{\log 2(n-l)}\sigma^* \leq 2\sqrt{\log 2n}\,\sigma^*,$$

with probability at least $1 - \frac{1}{n-t}$. In addition, Lemma 11 implies that

$$\|P^{\perp}_{X_{T^c}}\epsilon_{T^c}\|_\infty \leq 2\sqrt{\log 2n}\frac{1}{\left(-\frac{c_\nu}{6}\frac{|S|}{|S|-p}+c_\nu\right)}\widehat{\sigma} \leq 2\sqrt{\log 2n}\frac{1}{\left(-\frac{1}{6}\frac{|S|}{|S|-p}+1\right)\bar{c}}\widehat{\sigma}.$$

For $n \geq 12p$, we therefore conclude the bound (72).

Now consider γ^*_S with nonzero elements, i.e., $S \supset T^c$. We have

$$\begin{aligned}\|P^{\perp}_{X_S}(\gamma^*_S+\epsilon_S)\|_\infty &\geq \max_{i\in S}|e_i^\top P^{\perp}_{X_S}\gamma^*_S| - \|P^{\perp}_{X_S}\epsilon_S\|_\infty\\ &\geq \max_{i\in S}|e_i^\top P^{\perp}_{X_S}\gamma^*_S| - 2\sqrt{\log 2n}\,\sigma^*,\end{aligned}$$

with probability at least $1-\frac{1}{n-t}$. We now split $P^{\perp}_{X_S}$ into $P^{\perp}_{X_S}-(1-\frac{p}{n-l})I$ and $(1-\frac{p}{n-l})I$. By the triangle inequality, we have

$$\begin{aligned}\max_{i\in[n-l]}\left|e_i^\top P^{\perp}_{X_S}\gamma^*_S\right| &\geq \max_{i\in[n-l]}\left|e_i^\top\left(1-\frac{p}{n-l}\right)I\gamma^*_S\right| - \max_{i\in[n-l]}\left|e_i^\top\left(P^{\perp}_{X_S}-\left(1-\frac{p}{n-l}\right)I\right)\gamma^*_S\right|\\ &\geq \left(1-\frac{p}{n-l}\right)\|\gamma^*_S\|_\infty - \max_{i\in[n-l]}\left|\underbrace{e_i^\top\left(P^{\perp}_{X_S}-\left(1-\frac{p}{n-l}\right)I\right)\gamma^*_S}_{v_i}\right|.\end{aligned}$$

Plugging this into the result from inequality (70), we then obtain

$$\max_{i\in[n-l]}\left|e_i^\top P^{\perp}_{X_S}\gamma^*_S\right| \geq \left(1-\frac{p}{n-l}\right)\|\gamma^*_S\|_\infty - \frac{c_\nu}{8}\sigma^*.$$

Therefore, we have

$$\|P^{\perp}_{X_S}(\gamma^*_S+\epsilon_S)\|_\infty \geq \left(1-\frac{p}{n-t}\right)\min_{i\in T}|\gamma^*_i| - (2\sqrt{\log 2n}+c_\nu/8)\sigma^*.$$

By the assumption that $n \geq 12p$ and Lemma 11, we then obtain

$$\begin{aligned}\|P^{\perp}_{X_S}(\gamma^*_S+\epsilon_S)\|_\infty &\geq \frac{5}{6}\min_{i\in T}|\gamma^*_i| - \frac{(2\sqrt{\log 2n}+c_\nu/8)}{c_\nu-\frac{|S|}{|S|-p}\frac{c_\nu}{6}}\widehat{\sigma}\\ &\geq \frac{5}{6}\min_{i\in T}|\gamma^*_i| - \frac{(2\sqrt{\log 2n}+c_\nu/8)}{c_\nu-\frac{c_\nu}{5}}\widehat{\sigma}\\ &\geq \frac{5}{6}\min_{i\in T}|\gamma^*_i| - \frac{13}{6}\frac{\sqrt{\log 2n}}{\frac{4c_\nu}{5}}\widehat{\sigma}.\end{aligned}$$

Thus, $\|P^{\perp}_{X_S}(\gamma^*_S+\epsilon_S)\|_\infty \geq \frac{5}{2\bar{c}}\sqrt{\log 2n}\,\widehat{\sigma}$ if $\min_{i\in T}|\gamma^*_i|$ satisfies

$$\min_{i\in T}|\gamma^*_i| \geq \sqrt{\log 2n}\,\widehat{\sigma}\left(\frac{3}{\bar{c}}+\frac{13}{4c_\nu}\right).$$

This can be further achieved according to Lemma 11 if

$$\min_{i\in T}|\gamma^*_i| \geq \sqrt{\log 2n}\,\sigma^*\left(\frac{3}{\bar{c}}+\frac{13}{4c_\nu}\right)\left(C_\nu+\frac{c_\nu}{6}\frac{|S|}{|S|-p}\right).$$

Also note that by the assumption of $\min_{i\in T}|\gamma_i|$, we have

$$\min_{i\in T}|\gamma_i^*| \geq \frac{5}{4}\left(\frac{c_v+5C_v}{\bar{c}}\right)\sqrt{\log 2n}\,\sigma^* \geq \sqrt{\log 2n}\,\sigma^*\left(\frac{3}{\bar{c}}+\frac{13}{5c_v-\bar{c}}\right)\left(C_v+\frac{c_v}{6}\frac{|S|}{|S|-p}\right).$$

This concludes the proof. □

Lemma 13 (Theorem 2.5 in Adamczak (2015)) *Suppose X is a zero-mean random vector in $\mathbb{R}^n$ satisfying the convex concentration property with constant K. Then for any fixed matrix $A\in\mathbb{R}^{n\times n}$ and any $w>0$, we have*

$$\mathbb{P}\big(|X^\top AX-\mathbb{E}[X^\top AX]|\geq w\big)\leq 2\exp\left(-\frac{1}{C}\min\left\{\frac{w^2}{2K^4\|A\|_F^2},\frac{w}{K^2\|A\|_2}\right\}\right).$$

Lemma 14 *Suppose $X\in\mathbb{R}^{n\times p}$ has i.i.d. rows from a zero-mean distribution satisfying the convex concentration property with constant K. Then*

$$\left\|\frac{X^\top X}{n}-\mathbb{E}\left[\frac{X^\top X}{n}\right]\right\|_2\leq c\frac{\lambda_{\min}(\Sigma)}{2},$$

with probability at least $1-\exp(-n)$.

Proof Note that for any fixed unit vector $u\in\mathbb{R}^p$, the map $\varphi: x\mapsto\langle x,u\rangle$ is convex and 1-Lipschitz. Hence, by the definition of the convex concentration property, each $x_i^\top u$ is sub-Gaussian with parameter proportional to K. In fact, this is enough to show the desired matrix concentration result [cf. Vershynin (2010)]. We omit the details. □

E Appendix for Sect. 5

In this sectopm, we provide proofs and additional details for the results in Sect. 5.

E.1 Proof of Theorem 4

We will prove a stronger results here, which implies Theorem 4. This is actually mentioned by Remark 2.

Theorem 6 *With respect to D, the bug generator, who has attacking budgets no more than t, cannot fail the sign support recovery if only if (16) holds. That failure of sign support recovery,* $\operatorname{sign}(\widehat{\gamma})\neq\operatorname{sign}(\gamma^*)$, *means either $\widehat{\gamma}_j\neq 0$ for some $j\in T^c$ or $\widehat{\gamma}_i\gamma_i^*\leq 0$ for some $i\in T$.*

Proof of Theorem 4 We will use the following lemma to prove Theorem 4.

Lemma 15 *The following two properties are equivalent:*

(a) *For any vector* $\gamma^* \in \mathbb{R}^d$ *with support* K, *the constraint-based optimization has all solutions* $\hat{\gamma}$ *satisfying* $\operatorname{sign}(\hat{\gamma}) = \operatorname{sign}(\gamma^*)$.

(b) *The matrix* $\bar{P}(D)$ *satisfies the restricted nullspace property with respect to* K.

Proof of Lemma 15 We first prove $(b) \implies (a)$. This immediately follows Theorem 7.8 in Wainwright (2019) since $(b) \implies \gamma^* = \hat{\gamma}$ for any vector γ^* with $\text{supp}(\gamma^*) = K$, it thus implies $(b) \implies \operatorname{sign}(\hat{\gamma}) = \operatorname{sign}(\gamma^*)$. Or we can show it directly as follow. Suppose (a) doesn't hold. Then, we have $\Delta := \gamma^* - \hat{\gamma} \neq 0$. By the constraint and the objective, it also needs to satisfy that $\Delta \in Null(\bar{P}(D))$ and

$$\|\gamma^* - \Delta\|_1 = \|\hat{\gamma}\|_1 \leq \|\gamma^*\|_1 = \|\gamma^*_K\|_1.$$

Therefore, we have

$$\|\gamma^*_K\|_1 - \|\Delta_K\|_1 + \|\Delta_{K^c}\|_1 \leq \|\gamma^*_K - \Delta_K\|_1 + \|\Delta_{K^c}\|_1 \leq \|\gamma^*_K\|_1,$$

which means a nonzero $\Delta \in Null(\bar{P}) \cap \mathbb{C}^A$ and causes a contradiction. Thus when (b) is true, (a) holds as well.

From now on to the end of the proof, we will abuse notation by using $\bar{P}$ to represent $\bar{P}(D)$. The remaining thing is to prove $(a) \implies (b)$. We will prove by contradiction. If (b) doesn't hold, then there exists a nonzero Δ such that $\bar{P}\Delta = 0$ and $\|\Delta_{K^c}\|_1 \leq \|\Delta_K\|_1$. We consider a γ^* with $\gamma^*_K = \Delta_K$ and $\gamma^*_{K^c} = \mathbf{0}$. Let $\hat{\gamma}$ be the optimizer given this γ^*. By (a), we shall have $\operatorname{sign}(\hat{\gamma}) = \operatorname{sign}(\gamma^*) = \operatorname{sign}\left(\begin{bmatrix} \Delta_K \\ \mathbf{0}_{(n-t)\times 1} \end{bmatrix}\right)$. The idea is to construct a γ' that has no larger ℓ_1 norm than $\hat{\gamma}$ and has support not equal to K, which contradicts with (a), and therefore, (b) must hold.

Consider $\gamma' = \hat{\gamma} - c \cdot \Delta$ where $c = \frac{\hat{\gamma}_i}{\Delta_i}$ for $i = \arg\min_{j \in K} \frac{\hat{\gamma}_j}{\Delta_j}$. Since Δ is a nonzero vector, we must have $\Delta_l \neq 0$ for some $l \in K$. Therefore, we have c being positive finite, $\gamma'_i = 0$ and $|\hat{\gamma}_j| \geq c|\Delta_j|$ for all $j \in K$. Therefore, we further get

$$\bar{P}(\gamma^* - \gamma') = \bar{P}(\gamma^* - \hat{\gamma} + c\Delta) = \bar{P}(\gamma^* - \hat{\gamma}) = 0,$$

as well as

$$\begin{aligned} \|\gamma'\|_1 &= \|\hat{\gamma}_K - c \cdot \Delta_K\|_1 + \|\hat{\gamma}_{K^c} - c \cdot \Delta_{K^c}\|_1 \\ &\overset{(i)}{=} \|\hat{\gamma}_K\|_1 - c\|\Delta_K\|_1 + c\|\Delta_{K^c}\|_1 \\ &\overset{(ii)}{\leq} \|\hat{\gamma}\|_1, \end{aligned}$$

where (i) is because $\operatorname{sign}(\hat{\gamma}_K) = \operatorname{sign}(\Delta_K), c > 0, |\hat{\gamma}_K| \geq c|\Delta_K|$ and $\hat{\gamma}_{K^c} = 0$, (ii) is because $\Delta \in \mathbb{C}(K)$. Hence, we find a γ' to have smaller or equal ℓ_1 norm than $\hat{\gamma}$. This contradicts with the fact that all the solutions have support K or $\hat{\gamma}$ is the optimal solution. Therefore, (b) must hold and $(a) \implies (b)$. □

We first prove that (16) is sufficient. For any $|K| \leq t$ and $K \subseteq [n]$, we know that $Null(\bar{P}(D)) \cap \mathbb{C}(K) = \{0\}$. Then by Proposition 15, we conclude that $sign(\hat{\gamma}) = sign(\gamma^*)$ with $\text{supp}(\gamma^*) = K$ for any subset K of size no more than t.

We second prove that (16) is necessary. Note that for any subset K of size less equal to t, we have $\operatorname{sign}(\hat{\gamma}) = \operatorname{sign}(\gamma^*)$ with $\text{supp}(\gamma^*) = K$. By Proposition 15, it means $\bar{P}(D)$ satisfies the restricted nullspace property for any such K. Therefore $Null(\bar{P}(D)) \cap \mathbb{C}^A = \{\mathbf{0}\}$. □

Theorem 4 immediately holds from Theorem 6.

E.2 Proof of Remark 3

We will prove the statement in Remark 3 here.

Proposition 12 *The subspace* $Null(\bar{P}(D))$ *is equivalent to* $\{u \in \mathbb{R}^n \mid \exists v \in \mathbb{R}^p, \text{ such that } u = Xv, X_D v = 0\}$.

Proof of Proposition 12 We first prove $Null(\bar{P}(D)) \supseteq \{u \in \mathbb{R}^n \mid \exists v \in \mathbb{R}^p, \text{ such that } u = Xv, X_D v = 0\}$. Let $u = \left(X + M^\top X_D\right)v$ for some $v \in \mathbb{R}^p$, where $M \in \mathbb{R}^{m \times p}$ contains m rows stacked with the canonical vectors indexed by D so that $MX = X_D$. We have

$$\begin{aligned}\left(I - X\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top\right)u &= u - X\left(X^\top X + \frac{\eta n}{m} X_D^\top X_D\right)^{-1} X^\top \left(X + \frac{\eta n}{m} M^\top X_D\right)v \\ &= \frac{\eta n}{m} M^\top X_D v.\end{aligned}$$

Besides, we have

$$\begin{aligned}X_D\left(X^\top X + \frac{\eta n}{m} X_D^\top X_D\right)^{-1} X^\top u &= X_D\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top \left(X + M^\top X_D\right)v \\ &= X_D v.\end{aligned}$$

Therefore $X_D v = 0, u = Xv \implies u \in Null(\bar{P}(D))$.

Secondly we prove $Null(\bar{P}(D)) \subseteq \{u \mid \exists v \in \mathbb{R}^d, \text{ such that } u = Xv, X_D v = 0\}$. Let u be some vector in $\mathbb{N}(X_D)$. Then we have

$$u = X\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top u, \tag{74}$$

and

$$X_D\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top u = 0. \tag{75}$$

By (75), we have $\left(X^\top X + X_D^\top X_D\right)^{-1} X^\top u = v$ for some $v \in Null(X_D)$. Plugging this back to (74), we have $u = Xv$. Hence, we have $u \in \{u \mid \exists v \in \mathbb{R}^d, \text{ such that } u = Xv, X_D v = 0\}$.

□

E.3 Proof of Theorem 5

Here we prove the proof of Theorem 5. We write the minimax MILP here again.

$$\min_{\xi \in \{0,1\}^n} \max_{\substack{a, a^+, a^-, u, u^+, u^- \in \mathbb{R}^n, v \in \mathbb{R}^d \\ z, w \in \{0,1\}^n}} \sum_{j=1}^{n} a_j^+ - a_j^-, \tag{76}$$

$$\text{subject to } u = Xv, \tag{77}$$

$$u = u^+ - u^-, a = u^+ + u^-, u^+, u^- \geq 0, u^+ \leq z,\ u^- \leq (\mathbb{1}_n - z), \tag{78}$$

$$\sum_{i=1}^{n} w_i \leq t, \tag{79}$$

$$a^+ \leq w,\ a^- \leq \mathbb{1}_n - w, a = a^+ + a^-, a^+ \geq 0, a^- \geq 0, \tag{80}$$

$$\sum_{i=1}^{n} \xi_i \leq m\ i = 1, \dots, n, \tag{81}$$

$$u \leq \mathbb{1}_n - \xi, u \geq -(\mathbb{1}_n - \xi). \tag{82}$$

Proof of Theorem 5 We first argue that if (83) has the unique solution of $(u, v) = (\mathbf{0}, \mathbf{0})$, then (16) holds and thus the debugger can add m points indexed by D to achieve support recovery.

$$\begin{aligned} \min_{\substack{D \in [n], \\ |D| \leq m}} \max_{K \subseteq [n], |K| \leq t, u \in \mathbb{R}^n, v \in \mathbb{R}^d} & \|u_K\|_1 - \|u_{K^c}\|_1, \\ \text{subject to } & u = Xv, X_D v = 0, \|u\|_\infty \leq 1. \end{aligned} \tag{83}$$

Suppose (16) doesn't hold. Then there exists $K \subseteq [n], |K| \leq t$ and a nonzero vector u' such that $u' = Xv, X_D v = 0$ and $\|u'_K\|_1 \geq \|u'_{K^c}\|_1$. And $\frac{u'}{\|u'\|_2}$ satisfies $\|u'\|_\infty \leq 1$. This contradicts with that (83) has the unique solution of $(u, v) = (\mathbf{0}, \mathbf{0})$, then (16) holds. This concludes our first part of the proof.

Now we argue that the MILP is equivalent to (83). Equation (77) is inherited from original constraint. Equations in (78) are equivalent to $a = |u|$. Note that u^+, u^- respectively correspond to the positive and negative parts of u. If $z_i = 0$, then $u_i^+ = 0, u_i^- \leq 1$ and $u_i^- = -u_i$. If $z_i = 1$, then $u_i^- = 0, u_i^+ \leq 1$ and $u_i^+ = u_i$. The vector w indicates K in (83). If $w_i = 1$, then $i \in K$ otherwise $i \in K^c$. Therefore, Eq. (79) restricts the attacking budget to t. Then, equations in (80) are equivalent to $a_i^+ = |u_i|, a_i^- = 0$ for $i \in K$ and $a_i^- = |u_i|, a_i^+ = 0$ for $i \in K^c$. Therefore, the objective function corresponds to $\|u_K\|_1 - \|u_{K^c}\|_1$.

Note that the variable in the first layer is ξ. If $\xi_i = 1$, it means the debugger queries the point x_i. And the constraint $X_D v = 0$ is replaced by (82). This is because $x_i^\top v = 0 \Leftrightarrow u_i = 0$. If $\xi_j = 0$, then u_j just needs to satisfy $|u_j| \leq 1$.

Therefore, we have shown that the MILP is equivalent to (83) and thus conclude Theorem 5. □

Funding Funding was provided by National Science Foundation: NSF (Grant Numbers DMS-1749857, CCF-1740707).

References

Adamczak, R. (2015). A note on the Hanson-Wright inequality for random vectors with dependencies. *Electronic Communications in Probability, 20,* 1–13.

Cadamuro, G., Gilad-Bachrach, R., & Zhu, X. (2016). Debugging machine learning models. In *ICML Workshop on Reliable Machine Learning in the Wild.*

Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., & Mukhopadhyay, D. (2018). *Adversarial attacks and defences: A survey*. arXiv preprint arXiv:1810.00069

Fergus, R., Weiss, Y., & Torralba, A. (2009). Semi-supervised learning in gigantic image collections. In *NIPS* (Vol. 1, p. 2), Citeseer.

Foygel, R., & Mackey, L. (2014). Corrupted sensing: Novel guarantees for separating structured signals. *IEEE Transactions on Information Theory, 60*(2), 1223–1247.

Henderson, H. V., & Searle, S. R. (1981). On deriving the inverse of a sum of matrices. *SIAM Review, 23*(1), 53–60.

Hoeffding, W. (1994). Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding* (pp. 409–426), Springer.

Horn, R. A., & Johnson, C. R. (1994). *Topics in matrix analysis*. Cambridge University Press.

Huber, P., & Ronchetti, E. (2011). *Robust statistics*. Wiley Series in Probability and Statistics. Wiley.

Hwang, S. G. (2004). Cauchy's interlace theorem for eigenvalues of Hermitian matrices. *The American Mathematical Monthly, 111*(2), 157–159.

Meinshausen, N., & Yu, B. (2009). Lasso-type recovery of sparse representations for high-dimensional data. *The Annals of Statistics, 37*(1), 246–270.

Nguyen, N. H., & Tran, T. D. (2013). Robust Lasso with missing and grossly corrupted observations. *IEEE Transactions on Information Theory, 4*(59), 2036–2058.

Ravikumar, P., Wainwright, M. J., & Lafferty, J. D. (2010). High-dimensional Ising model selection using ℓ_1 -regularized logistic regression. *The Annals of Statistics, 38*(3), 1287–1319.

Rousseeuw, P. J., & Van Driessen, K. (2006). Computing LTS regression for large data sets. *Data Mining and Knowledge Discovery, 12*(1), 29–45.

Sasai, T., & Fujisawa, H. (2020). *Robust estimation with Lasso when outputs are adversarially contaminated*. arXiv preprint arXiv:2004.05990

Seber, G. A. F. (2008). *A matrix handbook for statisticians* (Vol. 15). Wiley.

She, Y., & Owen, A. B. (2011). Outlier detection using nonconvex penalized regression. *Journal of the American Statistical Association, 106*(494), 626–639.

Slawski, M., & Ben-David, E. (2017). *Linear regression with sparsely permuted data*. arXiv preprint arXiv:1710.06030

Tang, Y., Richard, J. P. P., & Smith, J. C. (2016). A class of algorithms for mixed-integer bilevel min–max optimization. *Journal of Global Optimization, 66*(2), 225–262.

Veit, A., Alldrin, N., Chechik, G., Krasin, I., Gupta, A., & Belongie, S. (2017). Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 839–847).

Vershynin, R. (2010). *Introduction to the non-asymptotic analysis of random matrices*. arXiv preprint arXiv:1011.3027

Vershynin, R. (2018). *High-dimensional probability: An introduction with applications in data science* (Vol. 47). Cambridge University Press.

Wainwright, M. J. (2009). Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1 -constrained quadratic programming (Lasso). *IEEE Transactions on Information Theory, 55*(5), 2183–2202.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint* (Vol. 48). Cambridge University Press.

Xu, P., & Wang, L. (2014). An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. *Computers & Operations Research, 41,* 309–318.

Zeng, B., & An, Y. (2014). Solving bilevel mixed integer program by reformulations and decomposition. *Optimization Online* (pp. 1–34).

Zhang, X., Zhu, X., & Wright, S. (2018). Training set debugging using trusted items. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 32).