Lower bounds on the performance of polynomial-time algorithms for sparse linear regression

Yuchen Zhang[⋆] Martin J. Wainwright[⋆,†] Michael I. Jordan[⋆,†]

{yuczhang,wainwrig,jordan}@berkeley.edu

[⋆]Department of Electrical Engineering and Computer Science [†]Department of Statistics
University of California, Berkeley

February 11, 2014

Abstract

Under a standard assumption in complexity theory ($\mathbf{NP} \not\subset \mathbf{P/poly}$), we demonstrate a gap between the minimax prediction risk for sparse linear regression that can be achieved by polynomial-time algorithms, and that achieved by optimal algorithms. In particular, when the design matrix is ill-conditioned, the minimax prediction loss achievable by polynomial-time algorithms can be substantially greater than that of an optimal algorithm. This result is the first known gap between polynomial and optimal algorithms for sparse linear regression, and does not depend on conjectures in average-case complexity.

1 Introduction

The past decade has witnessed a flurry of results on the performance of polynomial-time procedures, many of them based on convex relaxation, that aim at solving challenging optimization problems that arise in statistics. The large majority of these results have been of the positive variety, essentially guaranteeing that a certain polynomial-time procedure produces an estimate with low statistical error; see the overviews [8, 15] for results of this type. Moreover, in many cases, the resulting bounds have been shown to be minimax-optimal, meaning that no estimator can achieve substantially smaller error. More recently, however, this compelling story has begun to develop some wrinkles, in that gaps have been established between the performance of convex relaxations and the performance of optimal methods, notably in the context of sparse PCA and related sparse-low-rank matrix problems (e.g., [1, 3, 4, 11, 16]). The main contribution of this paper is to add an additional twist to this ongoing story, in particular by demonstrating a fundamental gap between the performance of polynomial-time methods and optimal methods for high-dimensional sparse linear regression. Notably, in contrast with the recent work of Rigollet and Berthet [4] on sparse PCA, and subsequent results on matrix detection [12], both of which are based on average-case complexity, our result is based only on a standard conjecture in worst-case complexity theory.

Linear regression is a canonical problem in statistics: it is based on observing a response vector $y \in \mathbb{R}^n$ and a design matrix $X \in \mathbb{R}^{n\times d}$ that are linked via the linear relationship

$$y = X\theta^* + w. \tag{1}$$

Here the vector $w \in \mathbb{R}^n$ is some form of observation noise, and our goal is to estimate the unknown vector $\theta^* \in \mathbb{R}^d$, known as the regression vector. Throughout this paper, we focus on the standard Gaussian model, in which the entries of the noise vector w are i.i.d. $N(0, \sigma^2)$ variates, and the case of deterministic design, in which the matrix X is viewed as non-random. In the sparse variant of this model, the regression vector is assumed to have a relatively small number of non-zero coefficients. In particular, for some positive integer $k < d$, the vector θ^* is said to be k-sparse if it has at most k non-zero coefficients. Thus, our model is parameterized by the triple (n, d, k) of sample size n, ambient dimension d, and sparsity k.

Given a k-sparse regression problem, the most direct approach would be to seek a k-sparse minimizer to the least-squares cost $\|y - X\theta\|_2^2$, thereby obtaining the ℓ_0-based estimator

$$\widehat{\theta}_{\ell_0} := \arg \min_{\theta \in \mathbb{B}_0(k)} \|y - X\theta\|_2^2. \tag{2}$$

Note that this estimator involves minimization over the ℓ_0-"ball"

$$\mathbb{B}_0(k) := \big\{\theta \in \mathbb{R}^d \mid \sum_{j=1}^{d} \mathbb{I}[\theta_j \neq 0] \leq k\big\} \tag{3}$$

of k-sparse vectors. This estimator is not easy to compute in a brute force manner, since there are $\binom{d}{k}$ subsets of size k to consider. More generally, it is known that computing a sparse solution to a set of linear equations is an NP-hard problem [14], and this intractability result has motivated the use of various heuristic algorithms and approximations. Recent years have witnessed an especially intensive study of methods based on ℓ_1-relaxation, including the basis pursuit and Lasso estimators [20, 9], as well as the Dantzig selector [7]. Essentially, these methods are based on replacing the ℓ_0-constraint (3) with its ℓ_1-equivalent, in either a constrained or penalized form. All of these estimators are based on relatively simple convex optimization problems (linear or quadratic programs), and so can be computed in polynomial time. Moreover, in certain cases, the performance of ℓ_1-based methods have been shown to meet minimax-optimal lower bounds [18].

Despite this encouraging progress, there remain some intriguing gaps in the performance of ℓ_1-based procedures, perhaps most notably when assessed in terms of their *mean-squared prediction error* $\frac{1}{n}\|X\widehat{\theta} - X\theta^*\|_2^2$. In order to bring this issue into sharper focus, given an estimator $\widehat{\theta}$, suppose that we evaluate its performance in terms of the quantity

$$\mathcal{M}_{n,k,d}(\widehat{\theta}; X) := \sup_{\theta^* \in \mathbb{B}_0(k)} \frac{1}{n} \mathbb{E}\big[\|X\widehat{\theta} - X\theta^*\|_2^2\big], \tag{4}$$

where the design matrix X remains fixed, and expectation is taken over realizations of the noise vector $w \sim N(0, \sigma^2 I_{n \times n})$.

The criterion (4) assesses the performance of the estimator $\widehat{\theta}$ uniformly over the set of all k-sparse regression vectors. In terms of this uniform measure, the ℓ_0-based estimator (2) is known [6, 18] to satisfy the bound

$$\mathcal{M}_{n,k,d}(\widehat{\theta}_{\ell_0}; X) \precsim \frac{\sigma^2 \, k \log d}{n}, \tag{5}$$

where $\precsim$ denotes an inequality up to a universal constant, meaning independent of all problem dependent quantities. A noteworthy point is that the upper bound (5) holds for *any* fixed design matrix X.

By way of contrast, most ℓ_1-based guarantees involve imposing certain conditions on the design matrix X. One of the most widely used conditions is the restricted eigenvalue (RE) condition [5, 21], which lower bounds the quadratic form defined by X over a subset of sparse vectors (see equation (8) to follow for a precise definition). Under such an RE condition, it can be shown that the Lasso-based estimator $\widehat{\theta}_{\ell_1}$ satisfies a bound of the form

$$\mathcal{M}_{n,k,d}(\widehat{\theta}_{\ell_1}; X) \precsim \frac{1}{\gamma^2(X)} \frac{\sigma^2 \, k \log d}{n}, \tag{6}$$

where $\gamma(X) \leq 1$ denotes the restricted eigenvalue constant (8). Comparison of this bound to the earlier ℓ_0-based guarantee (5) shows that the only difference is the RE constant, which is a measure of the conditioning of the matrix X. However, from a fundamental point of view, the conditioning of X has no effect on whether or not a good sparse predictor exists; for instance, a design matrix with two duplicated columns is poorly conditioned, but the duplication would have no effect on sparse prediction performance.

The difference between the bounds (5) and (6) leaves open various questions, both about the performance of the Lasso (and other ℓ_1-based methods), as well as polynomial-time methods more generally. Beginning with ℓ_1-based methods, one possibility is that existing analyses of prediction error are overly conservative, but that the Lasso can actually achieve the bound (5), without the additional RE term. When the regression vector θ^* has a bounded ℓ_1-norm, then it is possible to achieve a prediction error bound that does *not* involve the RE constant [6], but the resulting rate is “slow”, decaying as $1/\sqrt{n}$ instead of the rate $1/n$ given in equation (6). Foygel and Srebro [10] asked whether this slow rate could be improved without an RE condition, and gave a partial negative answer in the case $k = 2$, constructing a 2-sparse regression vector and a design matrix violating the RE condition for which the Lasso prediction error is lower bounded by $1/\sqrt{n}$. In this paper, we ask whether the same type of gap persists if we allow for *all polynomial-time estimators*, instead of just the Lasso. Our main result is to answer this question in the affirmative: we show that there is a family of design matrices X_{bad} such that, under a standard conjecture in computational complexity ($\mathbf{NP} \not\subset \mathbf{P/poly}$), for any estimator $\widehat{\theta}_{\text{poly}}$ that can be computed in polynomial time, its mean-squared error is lower bounded as

$$\mathcal{M}_{n,k,d}(\widehat{\theta}_{\text{poly}}; X_{\text{bad}}) \succsim \frac{1}{\gamma^2(X_{\text{bad}})} \frac{\sigma^2 k^{1-\delta} \log d}{n},$$

where $\delta > 0$ is an arbitrarily small positive scalar. Consequently, we see that there is a fundamental gap between the performance of polynomial-time methods and that of the optimal ℓ_0-based method.

The remainder of this paper is organized as follows. We begin in Section 2 with background on sparse linear regression and restricted eigenvalue conditions. We then introduce some background on complexity theory, followed by the statement of our main result in Section 3. The proof of the main theorem is given in Section 4, with more technical results deferred to the appendices.

2 Background and problem set-up

We begin with background on sparse linear regression, then introduce restricted eigenvalue conditions. These notions allow us to give a precise characterization of the mean-squared prediction error that can be achieved by the ℓ_0-based algorithm and by a thresholded version of the Lasso algorithm.

2.1 Estimators for sparse linear regression

As previously described, an instance of the sparse linear regression problem is based on observing a pair $(y, X) \in \mathbb{R}^n \times \mathbb{R}^{n \times d}$ that are linked via the linear equation (1), where the unknown regression vector $\theta^* \in \mathbb{R}^d$ is assumed to be k-sparse, and so belongs to the ℓ_0-ball $\mathbb{B}_0(k)$. An estimator $\widehat{\theta}$ of the regression vector is a (measurable) function $(y, X) \mapsto \widehat{\theta} \in \mathbb{R}^d$, and our goal is to determine an estimator that is both k-sparse, and has low prediction error $\frac{1}{n}\|X\widehat{\theta} - X\theta^*\|_2^2$. Accordingly, we let $\mathcal{A}(k)$ denote the family of all estimators that return vectors in $\mathbb{B}_0(k)$. Note that the ℓ_0-based estimator $\widehat{\theta}_{\ell_0}$, as previously defined in equation (2), belongs to the family $\mathcal{A}(k)$ of estimators. The following result provides a guarantee for this estimator:

Proposition 1 (Prediction error for $\widehat{\theta}_{\ell_0}$)**.** *There are universal constants $c_j, j = 1, 2$ such for any design matrix X, the ℓ_0-based estimator $\widehat{\theta}_{\ell_0}$ satisfies*

$$\frac{1}{n}\|X\widehat{\theta}_{\ell_0} - X\theta^*\|_2^2 \leq c_1 \frac{\sigma^2 k \log d}{n} \qquad \textit{for any } \theta^* \in \mathbb{B}_0(k) \tag{7}$$

with probability at least $1 - 2e^{-c_2 k \log d}$.

We also consider another member of the family $\mathcal{A}(k)$—namely, a thresholded version of the Lasso estimator [20, 9]. The ordinary Lasso estimate $\widehat{\theta}_{\lambda_n}$ based on regularization parameter $\lambda_n > 0$ is given by

$$\widehat{\theta}_{\lambda_n} := \arg\min_{\theta \in \mathbb{R}^d} \Big\{ \frac{1}{2n}\|y - X\theta\|_2^2 + \lambda_n \|\theta\|_1 \Big\}.$$

In general, this estimator need not be k-sparse, but a thresholded version of it can be shown to have similar guarantees. Overall, we define the *thresholded Lasso* estimator $\widehat{\theta}_{\mathrm{TL}}$ based on the following two steps:

(a) Compute the ordinary Lasso estimate $\widehat{\theta}_{\lambda_n}$ with $\lambda_n = 4\sigma\sqrt{\frac{\log d}{n}}$.

(b) Truncate $\widehat{\theta}_{\lambda_n}$ to its k entries that are the largest in absolute value, thereby obtaining the estimate $\widehat{\theta}_{\mathrm{TL}}$.

By construction, the estimator $\widehat{\theta}_{\mathrm{TL}}$ belongs to the family $\mathcal{A}(k)$. The choice of regularization parameter given in step (a) is a standard one for the Lasso.

2.2 Restricted eigenvalues and ℓ_1-guarantees

We now define the notion of a (sparse) restricted eigenvalue (RE), and then discuss guarantees on the Lasso-based estimator $\widehat{\theta}_{\mathrm{TL}}$ that hold under such an RE condition. Restricted eigenvalues are defined in terms of subsets S of the index set $\{1, 2, \ldots, d\}$, and a cone associated with any such subset. In particular, letting S^c denote the complement of S, we define the cone

$$\mathbb{C}(S) := \big\{\theta \in \mathbb{R}^d \mid \|\theta_{S^c}\|_1 \leq 3\|\theta_S\|_1\big\}.$$

Here $\|\theta_{S^c}\|_1 := \sum_{j \in S^c} |\theta_j|$ corresponds to the ℓ_1-norm of the coefficients indexed by S^c, with $\|\theta_S\|_1$ defined similarly. Note that any vector θ^* supported on S belongs to the cone $\mathbb{C}(S)$; in addition, it includes vectors whose ℓ_1-norm on the "bad" set S^c is small relative to their ℓ_1-norm on S.

Definition 1 (Restricted eigenvalue (RE) condition)**.** *Given triplet (n, d, k), the matrix $X \in \mathbb{R}^{n \times d}$ is said to satisfy a uniform γ-RE condition if*

$$\frac{1}{n}\|X\theta\|_2^2 \geq \gamma\|\theta\|_2^2 \qquad \textit{for all } \theta \in \bigcup_{|S|=k} \mathbb{C}(S). \tag{8}$$

Moreover, the restricted eigenvalue constant of X, denoted by $\gamma(X)$, is the greatest γ such that X satisfies the condition (8).

The RE condition (8) and related quantities have been studied extensively in past work on basis pursuit and the Lasso (e.g., [5, 13, 18]); see the paper [21] for an overview of the different types of RE parameters. Note that it characterizes the curvature of the quadratic form specified by X when restricted to a certain subset of relatively sparse vectors. When the RE constant $\gamma(X)$ is close to zero, there are relatively sparse vectors $\widetilde{\theta}$ such that $\|X\widetilde{\theta} - X\theta^*\|_2$ is small but $\|\widetilde{\theta} - \theta^*\|_2$ is large. Given that we observe only a noisy version of the product $X\theta^*$, it is then difficult to distinguish θ^* from other sparse vectors, which makes estimating θ^* from the data difficult. Thus, it is natural to impose an RE condition if the goal is to produce an estimate $\widehat{\theta}$ such that the ℓ_2-norm error $\|\widehat{\theta} - \theta^*\|_2$ is small. Indeed, for ℓ_2-norm estimation, Raskutti et al. [18] show that a closely related condition is necessary for any method.

In contrast, it is worth noting that the RE condition is not a necessary condition for minimizing the prediction loss $\|X\widehat{\theta} - X\theta^*\|_2$, since an estimator far apart from θ^* may still achieve small prediction error. However, the RE condition turns to be an important criterion for ℓ_1-based methods to guarantee good prediction performance. Under the normalization condition:

$$\frac{\|X\theta\|_2^2}{n} \leq \|\theta\|_2^2 \qquad \text{for all } \theta \in \mathbb{B}_0(2k), \tag{9}$$

the following result provides such a guarantee for the thresholded Lasso estimator:

Proposition 2 (Prediction error for thresholded Lasso)**.** *There are universal constants $c_j, j = 3, 4$ such that for any design matrix X satisfying the normalization condition (9) and having the RE constant $\gamma(X) > 0$, the thresholded Lasso estimator $\widehat{\theta}_{\scriptscriptstyle TL}$ satisfies*

$$\frac{1}{n}\|X\widehat{\theta}_{\scriptscriptstyle TL} - X\theta^*\|_2^2 \leq \frac{c_3}{\gamma^2(X)} \frac{\sigma^2 k \log d}{n} \qquad \textit{for any } \theta^* \in \mathbb{B}_0(k) \tag{10}$$

with probability at least $1 - 2e^{-c_4 k \log d}$.

See Appendix B for the proof of this claim. Apart from different numerical constants, the main difference between the guarantee (10) and our earlier bound (7) for the ℓ_0-based estimator is the $1/\gamma^2(X)$ term. The RE constant $\gamma(X)$ is a dimension-dependent quantity, since it is a function of the $n \times d$ design matrix.

3 Main result and its consequences

Thus far, we have considered two estimators in the class $\mathcal{A}(k)$—namely, the ℓ_0-constrained estimator $\widehat{\theta}_{\ell_0}$ and the thresholded Lasso estimator $\widehat{\theta}_{\mathrm{TL}}$. We also proved associated guarantees on their prediction error (Propositions 1 and 2 respectively), showing a $1/\gamma^2(X)$ gap between their respective guarantees. Our main result shows that this gap is *not a coincidence*: more fundamentally, it is a characterization of the gap between an optimal algorithm and the class of all polynomial-time algorithms.

In order to state our main result, we need to make precise a particular notion of a polynomial-efficient estimator. Since the observation (X, y) consists of real numbers, any efficient algorithm can only take a finite-length representation of the input. Consequently, we need to introduce an appropriate notion of discretization, as has been done in past work on matrix detection [12]. We begin by defining an appropriate form of *input quantization*: for any input value x and integer τ, the operator

$$\lfloor x \rfloor_\tau := 2^{-\tau} \lfloor 2^\tau x \rfloor$$

represents a $2^{-\tau}$-precise quantization of x. (Here $\lfloor u \rfloor$ denotes the largest integer smaller than or equal to u.) Given a real value x, an efficient estimator is allowed to take $\lfloor x \rfloor_\tau$ as its input for some finite choice τ. We denote by $\text{size}(x; \tau)$ the length of binary representation of $\lfloor x \rfloor_\tau$, and denote by $\text{size}(X, y; \tau)$ the total length of the discretized matrix vector pair (X, y).

The following definition of efficiency is parameterized in terms of three quantities: (i) a positive integer b, corresponding to the number of bits required to implement the estimator as a computer program; (ii) a polynomial function G of the triplet (n, d, k), corresponding to the discretization accuracy of the input, and (iii) a polynomial function H of input size, corresponding to the runtime of the program.

Definition 2 (Polynomial-efficient estimators)**.** *Given polynomial functions $G : (\mathbb{Z}_+)^3 \to \mathbb{R}_+$, $H : \mathbb{Z}_+ \to \mathbb{R}_+$ and a positive integer $b \in \mathbb{Z}_+$, an estimator $(y, X) \mapsto \widehat{\theta}(y, X)$ is said to be (b, G, H)-*efficient *if:*

- *It can be represented by a computer program that is encoded in b bits.*
- *For every problem of scale (n, d, k), it accepts inputs quantized to accuracy $\lfloor \cdot \rfloor_\tau$ where the quantization level is bounded as $\tau \leq G(n, d, k)$.*
- *For every input (X, y), it is guaranteed to terminate in time $H(\text{size}(X, y; \tau))$.*

According to this definition, if we choose a sufficiently large code length—say $b = 10^{16}$—and polynomial functions G and H that grow sufficiently fast—say $G(n, d, k) = (ndk)^{100}$ and $H(s) = s^{100}$—then the class of (b, G, H)-efficient algorithms allow estimators to take sufficiently accurate input, and covers all estimators that are reasonably efficient in terms of storage and running time.

To present the main result, we require a few more notions from complexity theory. See the book [2] for a more detailed introduction. In complexity theory, the class **P** corresponds to problems that are solvable in polynomial time by a Turing machine. A closely related class denoted by **P/poly**, corresponds to all problems solvable in polynomial time by a Turing machine with a so-called advice string (meaning a side-input to the machine) that is of polynomial length. Although it is known that the class **P/poly** is strictly bigger than the class **P** (e.g, [2]), it is widely believed that **NP** $\not\subset$ **P/poly**. Accordingly, throughout this paper, we impose the following assumption:

Assumption A. *The class* **NP** *is not contained in the class* **P/poly**.

Based on Assumption A, we are ready to present the main result. In this statement, we use $c_j, j = 5, 6$ to denote universal constants independent of (n, d, k), (F, G, H) and (γ, σ, δ).

Theorem 1. *If* $\mathbf{NP} \not\subset \mathbf{P/poly}$, *then for any positive integer* b, *any scalar* $\delta \in (0, 1)$, *any polynomial functions* $G : (\mathbb{Z}_+)^3 \to \mathbb{R}_+$ *and* $F, H : \mathbb{Z}_+ \to \mathbb{R}_+$, *there is a sparsity level* $k \geq 1$ *such that the following holds:*

For any dimension $d \in [4k, F(k)]$, *any sample size* n *in the interval* $[c_5 k \log d, F(k)]$, *and any scalar* $\gamma \in [2^{-G(n,d,k)}, \frac{1}{24\sqrt{2}})$, *there is a matrix* $X \in \mathbb{R}^{n \times d}$ *such that*

(a) *It satisfies the normalization condition* (9), *and has an RE constant* $\gamma(X)$ *that is bounded as* $|\gamma(X) - \gamma| \leq 2^{-G(n,d,k)}$.

(b) *For any* (b, G, H)*-efficient estimator* $\widehat{\theta} \in \mathcal{A}(k)$, *the mean-squared prediction error is lower bounded as*

$$\max_{\theta^* \in \mathbb{B}_0(k)} \mathbb{E}\Big[\frac{\|X(\widehat{\theta} - \theta^*)\|_2^2}{n}\Big] \geq \frac{c_6}{\gamma^2} \, \frac{\sigma^2 k^{1-\delta} \log d}{n}. \tag{11}$$

To understand the consequence of Theorem 1, suppose that we have chosen b, F, G, H sufficiently large and have chosen δ very close to zero. Under this setting, Theorem 1(b) shows that as long as the triplet (n, d, k) is sufficiently large, then there is a explicitly constructed design matrix X such that any (b, G, H)-polynomial-efficient estimator has prediction risk lower bounded by inequality (11). Part (a) guarantees that the constructed design matrix X satisfies the normalization (9) and RE conditions (8), so that Proposition 2 can be applied to the thresholded Lasso estimator for this instance of the sparse regression problem. Since the parameter $\delta \in (0, 1)$ may be chosen arbitrarily close to zero, the lower bound (11) essentially matches the upper bound of Proposition 2, thereby confirming that Theorem 1 gives a tight lower bound.

Overall, Theorem 1 establishes that the inverse dependence on the RE constant $\gamma(X)$ is unavoidable for the class of polynomial-time algorithms. In contrast, the ℓ_0-based estimator—a method that is not polynomially efficient—does *not* exhibit this dependence, as shown by Proposition 1.

It is worth contrasting Theorem 1 with past work on the computational hardness of sparse linear systems. As mentioned previously, Natarajan [14] showed that finding sparse solutions to linear systems is an NP-hard problem. To contrast with our result, this earlier result applies to the problem of solving $X\theta = y$, where X is the worst-case matrix selected by an adversary who knows the algorithm in advance. In contrast, in our Theorem 1, the design matrix X is fixed ahead of time, and the associated hardness result applies to all polynomial-time algorithms. Furthermore, Theorem 1 requires X to satisfy the normalization condition (9) as well as the RE condition (8), so that the lower bound is achievable by the Lasso-based estimator in Proposition 2. These two conditions are not satisfied in the earlier construction [14]. Finally, in Theorem 1 we have proved a lower bound of the order $\frac{\sigma^2 k \log d}{\gamma^2(X)\, n}$, which is stronger than the $\frac{1}{n}$ lower bound proved in the original paper [14].

4 Proof of Theorem 1

We now turn to the proof of our main result. In broad outline, the proof involves three main steps. We begin in Section 4.1 by constructing a particular matrix M for which the sparse linear regression problem is NP-hard, doing so by comparison to the exact 3-set cover problem. In Section 4.2, we then extend this worst-case hardness result to the probabilistic setting that is of interest in Theorem 1. Section 4.3 is devoted to the actual construction of the design matrix X, using the matrix M as a building block, and verifies that it satisfies the conditions in part (a). Sections 4.4 contains the proof of part (b) of the theorem. In all cases, we defer the proofs of more technical lemmas to the appendices.

4.1 An NP-Hard Problem

In this section, we construct a linear system and prove that solving it under a specific sparsity condition is NP-hard. The NP-hardness is establishing by reducing the exact 3-set cover problem [14] to this linear regression problem. The exact 3-set cover (X3C) problem is stated as follows: given an integer $m \geq 3$ divisible by three, a set $\mathcal{S} = \{1, \ldots, m\}$ and a collection $\mathcal{C}$ of 3-element subsets of $\mathcal{S}$, find $m/3$ subsets in $\mathcal{C}$ that exactly cover $\mathcal{S}$, assuming such an exact cover exists.

Throughout our development, we make use of the convenient shorthand $p := \binom{m}{3}$, as well as $[m] := \{1, 2, \ldots, m\}$ and $[p] := \{1, 2, \ldots, p\}$. We begin by constructing a matrix $M \in \mathbb{R}^{(m+3p)\times 4p}$ in a blockwise manner, namely

$$M := \begin{bmatrix} A & 0 \\ B & C \end{bmatrix}$$

where the submatrices have dimensions $A \in \mathbb{R}^{m\times p}$, $B \in \mathbb{R}^{3p\times p}$ and $C \in R^{3p\times 3p}$.

We define the submatrices by considering all possible subsets of the form $(a, b, c) \in [m]^3$, where the elements are all required to be distinct. Since $p = \binom{m}{3}$, each such subset can be labeled with a unique index $j \equiv j_{abc} \in [p]$. For each $j \in [p]$, this indexing can be used to define the j^{th} column of the submatrix A as follows:

$$A_{aj} = A_{bj} = A_{cj} = 1 \quad \text{and} \quad A_{dj} = 0 \;\; \text{for all } d \in [m]\backslash\{a, b, c\}.$$

In other words, the j^{th} column of A is the binary indicator vector for membership in the subset $\{a, b, c\}$ indexed by j.

The submatrices B and C are defined in terms of their rows. For each $j \in [p]$, we define the following three rows of B

$$B_j = e_j, \;\; B_{p+j} = e_j \;\; \text{and} \;\; B_{2p+j} = 0,$$

where $e_j \in \mathbb{R}^p$ is j^{th} canonical basis vector (i.e., with j-th entry equal to 1 and all other entries set to 0). Similarly, we define the rows of C by

$$C_j = -f_j, \;\; C_{p+j} = f_{p+j} \;\; \text{and} \;\; C_{2p+j} = f_{2p+j},$$

where $f_j \in \mathbb{R}^{3p}$ is the j^{th} canonical basis vector in $\mathbb{R}^{3p}$. We also define the set

$$\mathcal{V} := \big\{ v \in \mathbb{R}^{m+3p} \;\big|\; v = Mu \text{ for some } u \in \{0, 1\}^{4p} \cap \mathbb{B}_0(m/3 + p) \big\}.$$

The following lemma shows that solving the linear system $Mu = v$ for all $v \in \mathcal{V}$ is NP-hard.

Lemma 1. *Given the matrix $M \in \mathbb{R}^{(m+3p)\times 4p}$ as previously defined and a vector $v \in \mathcal{V}$, the problem of computing a vector $u \in \{0, 1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ such that $\|Mu - v\|_2 < \frac{1}{2}$ is NP-hard.*

Proof. We reduce the X3C problem, specified by the set $\mathcal{S} = \{1, \ldots, m\}$ and triplet collection $\mathcal{C}$ to the sparse linear regression problem. In particular, given $(\mathcal{S}, \mathcal{C})$, we use it to construct a response vector $v \in \mathbb{R}^{m+3p}$, and then consider the linear system $Mu = v$. We then show that any method that leads to a sparse vector u such that $\|Mu - v\|_2 < \frac{1}{2}$ can also be used to solve the exact 3-set cover problem.

Constructing a response vector from X3C: Given $(\mathcal{S}, \mathcal{C})$, we now construct the response vector v. Recalling that $v \in \mathbb{R}^{m+3p}$, we let the first m coordinates of the vector v equal to 1. Since any triplet of distinct elements $(a, b, c) \in [m]^3$ can be associated with a unique index $j_{abc} \in [p]$. We use this correspondence to define the remaining $3p$ entries of v as follows:

$$\text{If } (a,b,c) \in \mathcal{C}: \qquad \text{set } v_{m+j_{abc}} = 0,\ v_{m+p+j_{abc}} = 1 \text{ and } v_{m+2p+j_{abc}} = 0, \tag{12a}$$

$$\text{If } (a,b,c) \notin \mathcal{C}: \qquad \text{set } v_{m+j_{abc}} = 0,\ v_{m+p+j_{abc}} = 0, \text{ and } v_{m+2p+j_{abc}} = 1. \tag{12b}$$

Assuming the existence of an exact cover, we now show that $v \in \mathcal{V}$, in particular by constructing a vector $u \in \{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ such that $Mu = v$. Given a fixed triplet (a, b, c), let us introduce the shorthand notation

$$\alpha := u_{j_{abc}}, \quad \beta := u_{p+i_{abc}}, \quad \delta := u_{2p+j_{abc}} \quad \text{and} \quad \gamma := u_{3p+j_{abc}}. \tag{13}$$

Observe that the linear equation $Mu = v$ holds if and only if the following conditions hold:

$$\text{For } (a,b,c) \in \mathcal{C}: \qquad \alpha - \beta = 0, \quad \alpha + \delta = 1, \text{ and } \gamma = 0. \tag{14a}$$

$$\text{For } (a,b,c) \notin \mathcal{C}: \qquad \alpha - \beta = 0, \quad \alpha + \delta = 0, \text{ and } \gamma = 1. \tag{14b}$$

$$\text{For any } i \in \mathcal{S}: \qquad \text{exactly one triplet } (a,b,c) \text{ satisfies } i \in \{a,b,c\} \text{ and } \alpha = 1. \tag{14c}$$

Thus, it is sufficient to construct a vector $u \in \{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ which satisfies the above conditions. If $(a, b, c) \notin C$, then we let $(\alpha, \beta, \delta, \gamma) = (0, 0, 0, 1)$. If $(a, b, c) \in C$ but it does not belong to the exact cover, then we let $(\alpha, \beta, \delta, \gamma) = (0, 0, 1, 0)$. Otherwise, if $(a, b, c) \in C$ and it is selected in the exact cover, then we let $(\alpha, \beta, \delta, \gamma) = (1, 1, 0, 0)$. Given these specifications, it is straightforward to verify that $u \in \{0,1\}^{4p}$ and exactly $m/3 + p$ entries of u are equal to 1. It is also easy to verify that u satisfies conditions (14a)-(14c). Overall, we conclude that $v \in \mathcal{V}$, and hence is a valid response vector for the X3C problem.

Solving X3C using a sparse linear system solver: Suppose that there is a sparse solution $u \in \{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ which satisfies $\|Mu - v\|_2 < 1/2$, where the response vector v was previously defined in equations (12a) and (12b). We now use u to construct an exact 3-set cover. Recalling the notation (13), we observe that the condition $\|Mu - v\|_2 < 1/2$ leads to the following restrictions:

$$\begin{aligned}
&\text{For } (a,b,c) \in \mathcal{C}: \qquad |\alpha - \beta| < 1/2, \quad |\alpha + \delta - 1| < 1/2, \text{ and } |\gamma| < 1/2. \\
&\text{For } (a,b,c) \notin \mathcal{C}: \qquad |\alpha - \beta| < 1/2, \quad |\alpha + \delta| < 1/2, \text{ and } |\gamma - 1| < 1/2.
\end{aligned}$$

Then, we prove the following two claims:

Claim 1. *There are exactly $m/3$ nonzero entries in $\{u_1, \ldots, u_p\}$. In addition, for any $i \in [p]$ such that $u_i > 0$, we must have $u_i > 1/2$.*

Proof. For any triplet (a, b, c), we claim that there is at least one nonzero entry in (β, δ, γ). Thus, the fact that $u \in \mathbb{B}_0(m/3 + p)$ implies that there are at most $m/3$ nonzero entries among $\{u_1, \ldots, u_p\}$. To prove this claim, consider two cases $(a, b, c) \notin C$ and $(a, b, c) \in C$. If $(a, b, c) \notin C$, then we have a restriction $|\gamma - 1| < 1/2$, which makes $\gamma \neq 0$. If $(a, b, c) \in C$, then we have restrictions $|\alpha - \beta| < 1/2$ and $|\alpha + \delta - 1| < 1/2$. Thus,

$$1 > |\alpha - \beta| + |\alpha + \delta - 1| \geq |\beta + \delta - 1|,$$

which implies that either β or δ must be nonzero.

On the other hand, we claim that the number of nonzero entries in $\{u_1, \ldots, u_p\}$ is at least $m/3$. Assume that there are k nonzero entries in $\{u_1, \ldots, u_p\}$. We notice that each nonzero entry in $\{u_1, \ldots, u_p\}$ selects a column in matrix A that has three nonzero coordinates. Thus, there are at most $3k$ nonzero entries in $\{(Mu)_1, \ldots, (Mu)_m\}$. Since $|(Mu)_i - 1| < 1/2$ holds for all $1 \leq i \leq m$, all elements in $\{(Mu)_1, \ldots, (Mu)_m\}$ must be greater than $1/2$. Thus, we have $3k \geq m$ which establishes the claim.

Hence, there are exactly $m/3$ nonzero entries in $\{u_1, \ldots, u_p\}$. Each nonzero entry covers exactly three coordinates of v. Since all elements in $\{(Mu)_1, \ldots, (Mu)_m\}$ must be greater than $1/2$, we conclude that the nonzero u_i's must be greater than $1/2$. □

Claim 2. *For any triplet $(a, b, c) \notin C$, we have $u_{i_{abc}} = 0$.*

Proof. We proceed via proof by contradiction. Suppose that $(a, b, c) \notin C$ and $u_{i_{abc}} \neq 0$. By Claim 1, we have $\alpha = u_{i_{abc}} > 1/2$. The condition $\|Mu - v\|_2 < 1/2$ imposes the restrictions $|\alpha - \beta| < 1/2$ and $|\gamma - 1| < 1/2$, which in turn imply that $\beta \neq 0$ and $\gamma \neq 0$. In the proof of Claim 1, we have shown that for any triplet (a, b, c) there is at least one nonzero entry in (β, δ, γ). For this specific triplet (a, b, c), there are at least two nonzero terms among (β, δ, γ). Hence, the total number of nonzero entries in $\{u_1, \ldots, u_p\}$ is at most $m/3 - 1$, which contradicts Claim 1. □

According to Claim 2, all nonzero entries in $\{u_1, \ldots, u_p\}$ correspond to 3-sets in $\mathcal{C}$. According to Claim 1, exactly $m/3$ of these 3-sets are selected. The condition $\|Mu - v\|_2 < 1/2$ implies that all entries in $\{(Mu)_1, \ldots, (Mu)_m\}$ are nonzero, which means that $\mathcal{S}$ is covered by the union of these 3-sets. Thus, these $m/3$ 3-sets form an exact cover of $\mathcal{S}$. □

4.2 Probabilistic Hardness

As in the previous section, we use the shorthand notation $p = \binom{m}{3}$ throughout. In this section, we extend the result of Section 4.1 to the probabilistic case. In order to do so, we require the following auxiliary lemma:

Lemma 2. *Let f, g be arbitrary polynomial functions, let $m_0 \in \mathbb{Z}_+$ be an arbitrary positive integer, and suppose that Assumption A holds. Then for any NP-hard problem $\mathcal{P}$, there is an input length $m > m_0$ such that any algorithm that can be encoded in $f(m)$ bits and that terminates in $g(m)$ time must fail on at least one input with probability greater than $1/2$.*

See Appendix A.1 for the proof.

We are now ready to state and prove the main result of this section:

Lemma 3. *Let f, g and h be arbitrary polynomial functions defined on $\mathbb{Z}_+ := \{1, 2, \ldots\}$. Then for any $m_0 \in \mathbb{Z}_+$, there is an integer $m > m_0$ and a distribution $\mathbb{Q}$ over $u \in \{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ such that for any solver $\widehat{u}$ that can be encoded in $f(m)$ bits and terminates in $g(m)$ time, we have*

$$\mathbb{Q}\Big[\, \|M\widehat{u} - Mu\|_2 < \frac{1}{2} \Big] \leq \frac{1}{h(m)}. \tag{15}$$

Proof. We proceed via proof by contradiction. In particular, for a given m_0, suppose that for every $m > m_0$ and for any distribution over $\{0,1\}^{4p} \cap \mathbb{B}_0(m/3+p)$, there is a solver $\widehat{u}$ that can be encoded in $f(m)$ bits and terminates in $g(m)$ time, and such that inequality (15) is violated.

For a positive integer N to be specified, we now construct a sequence of distributions $\{\mathbb{P}_t\}_{t=1}^N$ that lead to a contradiction. Let $\mathbb{P}_1$ the uniform distribution over $\mathbb{S}_1 := \{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ and let $\widehat{u}_1$ be the solver which achieves $\mathbb{P}_1\big[\, \|M\widehat{u}_1 - Mu\|_2 < \frac{1}{2}\big] > \frac{1}{h(m)}$. For $t \geq 1$, we define the set $\mathbb{S}_{t+1}$ recursively as

$$\mathbb{S}_{t+1} = \mathbb{S}_t \cap \big\{u \in \mathbb{S}_0 \;\mid\; \|M\widehat{u}_t - Mu\|_2 \geq \frac{1}{2}\big\}.$$

When $\mathbb{S}_{t+1}$ is non-empty, we let $\mathbb{P}_{t+1}$ be the uniform distribution over $\mathbb{S}_{t+1}$, and let $\widehat{u}_{t+1}$ denote the solver that satisfies the bound $\mathbb{P}_{t+1}\big[\, \|M\widehat{u} - Mu\|_2 < \frac{1}{2}\big] > \frac{1}{h(m)}$. If $\mathbb{S}_{t+1}$ is empty, then we simply set $\widehat{u}_{t+1} = \widehat{u}_t$.

For any integer $N \geq 1$, this construction yields a sequence of solvers $\{\widehat{u}_t\}_{t=1}^N$, and we use them to define a combined solver u_N^* as follows:

$$u_N^* = \begin{cases} \widehat{u}_t & \text{for first } t \in \{1, \ldots, N\} \text{ such that } \|M\widehat{u}_t - Mu\| < \frac{1}{2} \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

If $\mathbb{S}_t$ is empty for some $t \in [N]$, then $\|Mu_N^* - Mu\|_2 < 1/2$ for all $u \in \mathbb{S}_0$. Otherwise, we may assume that $\mathbb{S}_t$ is non-empty for each $t \in [N]$. Since $\mathbb{S}_t$ is the set of vectors u for which all the solvers $\{\widehat{u}_1, \ldots, \widehat{u}_{t-1}\}$ fail, and $\mathbb{P}_t$ is the uniform distribution over $\mathbb{S}_t$, the chance of success for every solver $\widehat{u}_t$ in the construction (16) is at least $\frac{1}{h(m)}$. Consequently, if we consider the chance of success for the solver u_N^* under the uniform distribution over $\mathbb{S}_1$, we have

$$\mathbb{P}_1\big[\, \|Mu_N^* - Mu\|_2 < \frac{1}{2}\big] \overset{(i)}{\geq} 1 - \big(1 - \frac{1}{h(m)}\big)^N \overset{(ii)}{\geq} 1 - (1/2)^{4p+1}, \tag{17}$$

where inequality (ii) follows by choosing $N := \lceil \frac{-(4p+1)}{\log_2(1 - \frac{1}{h(m)})} \rceil$. Since the distribution $\mathbb{P}_1$ is uniform over a support of at most 2^{4p} points, every point has probability mass no less than $(1/2)^{4p}$. This fact, combined with the lower bound (17), implies that for any $u \in \mathbb{S}_1$ the solver u_N^* satisfies the bound $\|Mu_N^* - Mu\|_2 < 1/2$ with probability greater than 1/2.

Note that the solver u_N^* can be encoded in $N\, f(m)$ bits and it terminates in $N\, g(m)$ time, both are polynomial functions of m. According to Lemma 2 and Lemma 1, there is an integer $m > m_0$ such that on at least one input u, the solver u_N^* fails to achieve $\|Mu_N^* - Mu\|_2 < 1/2$ with probability 1/2. However, this contradicts the previous conclusion, which implies that our starting assumption was incorrect. □

4.3 Proof of part (a)

We now turn to the construction of the design matrix X specified in Theorem 1. Lemma 3 implies that given arbitrary polynomial functions f, g and h, there is a positive integer $m > m_0$ (where m_0 can be arbitrarily large), a matrix $M \in \mathbb{R}^{(m+3p)\times 4p}$ with $p = \binom{m}{3}$, and a distribution $\mathbb{Q}$ over the set of binary vectors $u \in \{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3} + p)$ such that solving for u based on observing (M, Mu) is hard in the average case. We leave f, g, h and m_0 to be specified later, assuming that they are sufficiently large. Based on integer m and for a second positive integer t to be chosen, we define the sparsity level

$$k := t\left(\frac{m}{3} + p\right) \;=\; t\left(\frac{m}{3} + \binom{m}{3}\right).$$

Using the matrix M as a building block, we construct a design matrix $X \in \mathbb{R}^{n\times d}$ that satisfies the conditions of Theorem 1. To this end, our first step is to construct a matrix A_k. Recall the matrix $M \in \mathbb{R}^{(m+3p)\times 4p}$ constructed in Section 4.1. For the given integer $t > 1$, we consider the rescaled matrix $\sqrt{t}M$ and replicate this matrix t times in diagonal blocks to build the following matrix:

$$A_k := \text{blkdiag}\underbrace{\left\{\sqrt{t}M, \sqrt{t}M, \ldots, \sqrt{t}M\right\}}_{t \text{ copies}} \in \mathbb{R}^{3k\times 4pt}.$$

By construction, the matrix A_k has $3k = t(m+3p)$ rows and $4pt$ columns. Since $4pt \leq 4k \leq d$, the matrix

$$B_k := \frac{1}{2}\begin{bmatrix} A_k & 0_{3k\times(d-4pt)}\end{bmatrix} \qquad \text{has dimensions } 3k \times d.$$

We now use B_k to construct the design matrix $X \in \mathbb{R}^{n\times d}$. (To simplify the argument, we assume that n is divisible by $6k$; when this condition does not hold, an obvious modification of the argument yields the same result modulo different constants.) Assume that $R \in \mathbb{R}^{(n/2)\times d}$ is a random Gaussian matrix whose rows are sampled i.i.d. from the Gaussian distribution $N(0, I_{d\times d})$. We define a parameterized family of random matrices $\{C_x \in \mathbb{R}^{n\times d}, x \geq 0\}$ such that:

- its top $n/2$ rows consist of $\frac{n}{6k}$ copies of the matrix B_k.
- for each $x \geq 0$, the bottom $n/2$ rows of C_x are equal to xR.

Given a matrix $C \in \mathbb{R}^{n\times d}$, let $C^{\uparrow}$ and $C^{\downarrow}$ represent its top $\frac{n}{2}$ rows and the bottom $\frac{n}{2}$ rows, respectively. In addition, we use $\lfloor C \rfloor_L$ to represent the matrix formed by quantizing every entry of C by the operator $\lfloor\cdot\rfloor_L$. In addition, we define $\varepsilon := 2^{-G(n,d,k)}$ as a shorthand quantity and note that $\gamma \in [\varepsilon, \frac{1}{24\sqrt{2}})$. The following lemma shows that there exists a particular value of x such that the matrix C_x has nice properties.

Lemma 4. *For any integer $L \geq \max\{\log(12\sqrt{d}), \log(\sqrt{nd}/\varepsilon)\}$, there is a realization of the random matrix R and a particular value of x such that the matrix $\lfloor C_x\rfloor_L$ has RE constant $|\gamma(\lfloor C_x\rfloor_L) - \gamma| \leq \varepsilon$, and satisfies the upper bounds*

$$\frac{\|\lfloor C_x\rfloor_L\theta\|_2}{\sqrt{n}} \leq \|\theta\|_2 \quad \textit{and} \quad \frac{\|\lfloor C_x^{\downarrow}\rfloor_L\theta\|_2}{\sqrt{n}} \leq 25\gamma\|\theta\|_2 \quad \textit{for all } \theta \in \mathbb{B}_0(2k). \tag{18}$$

According to Lemma 4, if we choose an integer $L \geq \max\{\log(12\sqrt{d}), \log(\sqrt{nd}/\varepsilon)\}$ and define the design matrix $X := \lfloor C_x\rfloor_L$, then the matrix X satisfies part (a) of the theorem. We leave the concrete value of L to be specified later, assuming now that it is sufficiently large.

4.4 Proof of part (b)

We begin with some notation for this section. Let $X^{\uparrow}$ and $X^{\downarrow}$ represent the top $\frac{n}{2}$ rows and the bottom $\frac{n}{2}$ rows of X, respectively. Given a vector $\theta \in \mathbb{R}^d$, we take its first $4pt$ coordinates and partition them evenly into t segments, each of length $4p$. For $i \in [t]$, we use $\theta_i \in \mathbb{R}^{4p}$ to denote the i^{th} segment, corresponding to the coordinates indexed $4p(i-1)+1$ to $4pi$. Using this notation, the proof involves the subvectors $\{\widehat{\theta}_i,\ i \in [t]\}$ and $\{\theta^*_i,\ i \in [t]\}$ defined by $\widehat{\theta}$ and θ^*, respectively.

Lemma 3 guarantees the existence of a distribution $\mathbb{Q}$ over $\{0,1\}^{4p} \cap \mathbb{B}_0(\frac{m}{3}+p)$ such that, if u^* is sampled from $\mathbb{Q}$, then solving $Mu = Mu^*$ is computationally hard. We now define the quantity

$$\rho := \left\lfloor \frac{r}{\sqrt{m/3+p}} \right\rfloor_L,$$

where $r > 0$ is a constant to be specified later. Suppose that each subvector θ^*_i/ρ is independently drawn from the distribution $\mathbb{Q}$—to be precise, a subvector is drawn from the distribution $\mathbb{Q}$, and is rescaled by a factor of ρ to obtain θ^*_i. Note that

$$\max_{\theta^* \in \mathbb{B}_0(k)} \mathbb{E}_{\theta^*}\left[\|X\widehat{\theta} - X\theta^*\|_2^2\right] \geq \sup_{\mathbb{Q}} \mathbb{E}_{\mathbb{Q}}\left[\|X\widehat{\theta} - X\theta^*\|_2^2\right],$$

so that it suffices to lower bound the right-hand side.

Denote by **P1** the problem defined in Lemma 3. Our strategy is to reduce problem **P1** to our regression problem, denoted **P2**. There are two main differences between the linear equation $Mu = Mu^*$ from Lemma 3 and the linear regression problem $y' = X\theta + w$ to be defined in this section. First, in addition to the basic matrix M, the design matrix X contains additional $n/2$ bottom rows consisting of independent Gaussian entries, Second, in the linear regression problem, the response vector y' is corrupted by Gaussian noise, whereas the original linear system $Mu = Mu^*$ is based on a noiseless response vector (namely, Mu^*). Consequently, we need to construct a reduction that bridges this gap.

Constructing a P2 instance from a P1 instance: Suppose that we are given an instance (M, Mu^*) of problem **P1**, where the vector u^* is sampled from $\mathbb{Q}$. We now show how to use it to construct a **P2** instance. Any **P2** instance consists of a design matrix X and a response vector y'. We have already discussed how to construct the design matrix X in Section 4.3, so that it remains to construct the response vector y'. Recall from the start of this subsection our notation $\{\theta_j, j \in [t]\}$ for the subvectors of the vector $\theta \in \mathbb{R}^d$. Our construction consists of the following three steps:

- First, we pick arbitrary index number $i \in [t]$ and construct a random vector $\tilde{\theta}^i \in \mathbb{R}^d$ as follows:
  - for each $j \in [t]\backslash\{i\}$, we draw the j^{th} sub-vector $\tilde{\theta}^i_j/\rho \in \mathbb{R}^{4p}$ independently from distribution $\mathbb{Q}$.
  - All other coordinates of $\tilde{\theta}^i$ are set to zero.
- Second, we form the vector $\tilde{\xi}^i \in \mathbb{R}^d$ with subvectors specified as

$$\tilde{\xi}^i_\ell := \begin{cases} \rho u^* & \text{if } \ell = i \\ \tilde{\theta}^i_\ell & \text{if } \ell \in [t]\backslash i, \end{cases}$$

with all other coordinates set to zero.

- Third, we form the response vector

$$y' = \begin{bmatrix} X^{\uparrow}\, \tilde{\xi}^{\,i} \\ X^{\downarrow}\, \tilde{\theta}^{\,i} \end{bmatrix} + w,$$

  where $w \sim N(0, \sigma^2 I_{n \times n})$ is a Gaussian noise vector,

It is important to note that even though $\tilde{\xi}^{\,i}$ contains ρu^* as a subvector, constructing the matrix vector product $X^{\uparrow}\tilde{\xi}^{\,i}$ only requires the knowledge of matrix vector product Mu^*. Thus, the response vector y' can be constructed from (M, Mu^*).

It is also convenient to introduce the auxiliary response vector $y^{\diamond} := X\tilde{\xi}^{\,i} + w$. Since the random vectors $\tilde{\xi}^{\,i}$ and θ^* have the same distribution, the response vectors y and $y^{\diamond}$ have the same distribution. Moreover, by construction, the top $n/2$ coordinates of $y^{\diamond}$ are identical to those of y'. Using $y^{\diamond}_{\text{bot}}$ and y'_{bot} to denote the bottom $n/2$ co-ordinates of $y^{\diamond}$ and y' respectively, it is easy to check that $y^{\diamond}_{\text{bot}} \sim N(X^{\downarrow}\tilde{\xi}^{\,i}, \sigma^2 I_{n/2 \times n/2})$ and $y'_{\text{bot}} \sim N(X^{\downarrow}\tilde{\theta}^{\,i}, \sigma^2 I_{n/2 \times n/2})$.

Solving P1 via a method for P2 : Having constructed a **P2** instance (X, y') from (M, Mu^*), we now show how a method for solving a **P2** instance can be used to solve the original **P1** instance. More precisely, given an estimator $\widehat{\theta}$ of θ^* for problem **P2**, we output $u'_i := \widehat{\theta}_i(X, y')/\rho$ as a solution to **P1**. For the purposes of analysis, it is also convenient to define the estimator $u^{\diamond}_i := \widehat{\theta}_i(X, y^{\diamond})/\rho$. Since y and $y^{\diamond}$ share the same probability distribution, we note that the estimator $u^{\diamond}_i$ shares the same distribution as $\widehat{\theta}_i(X, y)/\rho$.

We now state two lemmas that characterize the basic properties of the estimators $\{u'_i, i \in [t]\}$. The first lemma upper bounds the prediction error of u'_i in terms of the mean-squared prediction error of the auxiliary solver $u^{\diamond}_i$.

Lemma 5. *As long as* $\frac{\gamma r \sqrt{n}}{\sigma} \leq 1/50$, *we are guaranteed that*

$$\big\|Mu'_i - Mu^*\big\|_2^2 \leq \frac{2e\sigma}{25\gamma r\sqrt{n}}\, \mathbb{E}\Big[\, \|Mu^{\diamond}_i - Mu^*\|_2^2\Big].$$

with probability at least $1 - \frac{50\gamma r\sqrt{n}}{\sigma}$.

See Appendix A.3 for the proof of this claim.

Our next lemma shows that a large subset of the estimators $\{u'_i, i \in [t]\}$ return sparse outputs with positive probability:

Lemma 6. *There is a set* $T \subset \{1, \ldots, t\}$ *with cardinality* $|T| \geq \frac{t^2}{2k+t}$ *such that*

$$\mathbb{P}\Big[u'_i \in \mathbb{B}_0(k/t)\Big] \geq e^{-1}\Big\{\frac{t}{2k+2t} - \frac{25\gamma r\sqrt{n}}{\sigma}\Big\} \qquad \textit{for each } i \in T.$$

See Appendix A.4 for the proof of this claim.

We now proceed by assigning particular values to the parameters t and r. In particular, we set

$$r := \frac{\sigma}{25 \times 16\gamma\sqrt{n}} \, \frac{t}{k+t}, \quad \text{and} \quad t := \lceil \big(\frac{m}{3} + p\big)^{\frac{1-\alpha}{\alpha}} \rceil, \tag{19a}$$

$$\text{and} \quad L := \Big\lceil \max\Big\{\log(12\sqrt{d}), \log(\sqrt{nd}/\varepsilon), \log(2\sqrt{k/t}/r)\Big\}\Big\rceil \tag{19b}$$

where $\alpha \in (0,1)$ is to be chosen later. Recalling that $k = t\,(\frac{m}{3} + p)$ by definition, we have the sandwich relation

$$k^{1-\alpha} \overset{(i)}{\leq} t \overset{(ii)}{\leq} k, \tag{20}$$

as well as the inequalities

$$e^{-1}\Big(\frac{t}{2k+2t} - \frac{25\gamma r\sqrt{n}}{\sigma}\Big) - \frac{50\gamma r\sqrt{n}}{\sigma} = \Big(\frac{7}{16e} - \frac{1}{8}\Big)\frac{t}{k+t} > 0.03\, k^{-\alpha}, \qquad \text{and} \tag{21a}$$

$$\frac{2e\sigma}{25\gamma r\sqrt{n}} = \frac{32k + 32t}{t} \leq 64k^{\alpha}. \tag{21b}$$

Combined with our two previous lemmas, we see that there is a subset T with cardinality $|T| \geq \frac{1}{3}k^{1-2\alpha}$ such that for each $i \in T$,

$$\mathbb{P}\Big[u_i' \in \mathbb{B}_0(k/t) \text{ and } \big\|Mu_i' - Mu^*\big\|_2^2 \leq 64ek^{\alpha}\mathbb{E}\big[\,\|Mu_i^{\diamond} - Mu^*\|_2^2\,\big]\Big] \geq 0.03\, k^{-\alpha}. \tag{22}$$

The following lemma transforms this inequality into a lower bound on the quantity $\mathbb{E}\big[\,\|Mu_i^{\diamond} - Mu^*\|_2^2\,\big]$. The proof explicitly relies on the hardness result of Lemma 3, and is deferred to Appendix A.5.

Lemma 7. *If inequality (22) holds, then* $\mathbb{E}\big[\|Mu_i^{\diamond} - Mu^*\|_2^2\big] \geq \frac{1}{256\,e}\, k^{-\alpha}$.

Note that the pair $(u_i^{\diamond}, u^*)$ has the same probability distribution as the pair $(\widehat{\theta}_i(X,y)/\rho, \theta_i^*/\rho)$. Therefore, under the conditions of Lemma 7, we are guaranteed that $\mathbb{E}\Big[\|M\widehat{\theta}_i - M\theta_i^*\|_2^2\Big] \geq \frac{\rho^2}{256\,e}\, k^{-\alpha}$. Since this lower bound holds for each index $i \in T$, we may sum over all such indices, thereby obtaining

$$\sum_{i \in T} \mathbb{E}\big[\|M\widehat{\theta}_i - M\theta_i^*\|_2^2\big] \geq |T|\,\frac{\rho^2}{256\,e}\, k^{-\alpha} \;\geq \frac{\rho^2}{768\,e}\, k^{1-3\alpha},$$

where we have recalled that $|T| \geq \frac{1}{3}k^{1-2\alpha}$. Substituting in

$$\rho = \Bigg\lfloor \frac{r}{\sqrt{m/3+p}} \Bigg\rfloor_L \geq \frac{r}{\sqrt{k/t}} - 2^{-L} \geq \frac{r}{2\sqrt{k/t}}$$

as well as our previous choice (19a) of the parameter r, we find that

$$\sum_{i \in T} \mathbb{E}\big[\|M\widehat{\theta}_i - M\theta_i^*\|_2^2 \geq \frac{r^2 t k^{-3\alpha}}{4 \times 768e} \;\geq \frac{1}{\gamma^2}\,\frac{1}{(25 \times 16)^2 \times 4 \times 768e}\,\frac{\sigma^2 k^{1-6\alpha}}{n}, \tag{23}$$

where we have also used the sandwich inequalities (20).

Our final step is to convert the inequality (23) into a lower bound on the quantity of interest, namely $\mathbb{E}\big[\frac{1}{n}\|X\widehat{\theta} - X\theta^*\|_2^2\big]$. In particular, we have

$$\mathbb{E}\big[\frac{1}{n}\|X\widehat{\theta} - X\theta^*\|_2^2\big] \geq \mathbb{E}\big[\frac{1}{n}\,\|X^{\uparrow}\widehat{\theta} - X^{\uparrow}\theta^*\|_2^2\big] \;\geq\; \frac{1}{n}\,\frac{n}{6k}\,\frac{t}{2}\sum_{i \in T}\mathbb{E}\big[\|M\widehat{\theta}_i - M\theta_i^*\|_2^2\big], \tag{24}$$

where $\frac{n}{6k}$ is the number of replicates of the submatrix B_k in the full design matrix X, and $\frac{t}{2}$ is the scale factor in the definition of B_k. Combining inequalities (23) and (24) yields

$$\mathbb{E}\Big[\frac{1}{n}\|X\widehat{\theta} - X\theta^*\|_2^2\Big] \geq \frac{c}{\gamma^2}\,\frac{\sigma^2 k^{1-7\alpha}}{n},$$

where $c > 0$ is a numerical constant. By assumption, the problem dimension is upper bounded as $d \leq F(k)$ where F is a polynomial function. Since Lemma 3 allows m_0 to be arbrtrary, we may choose it large enough so as to ensure that $k^\alpha \geq \log d$, or equivalently $k^{1-7\alpha} \geq k^{1-8\alpha}\log d$. Finally, setting $\alpha = \delta/8$ completes the proof of the theorem.

5 Conclusion

In this paper, under a standard conjecture in complexity theory, we have established a fundamental gap between the prediction error achievable by optimal algorithms, and that achievable by polynomial-time algorithms. In particular, whereas the prediction error of an optimal algorithm has no dependence on the restricted eigenvalue constant, our theory shows that the prediction error of any polynomial-time algorithm exhibits an inverse dependence on this quantity (for a suitably constructed design matrix). To the best of our knowledge, this is the first lower bound on the polynomially-constrained minimax rate of statistical estimation that depends only conjectures in worst-case complexity theory.

Acknowledgements

This work was supported in part by NSF CISE Expeditions award CCF-1139158, the U.S. Army Research Laboratory, the U.S. Army Research Office under grant number W911NF-11-1-0391, the Office of Naval Research MURI grant N00014-11-1-0688, and DARPA XData Award FA8750-12-2-0331, and by gifts from Amazon Web Services, Google, SAP, Apple, Inc., Cisco, Clearstory Data, Cloudera, Ericsson, Facebook, GameOnTalis, General Electric, Hortonworks, Huawei, Intel, Microsoft, NetApp, Oracle, Samsung, Splunk, VMware, WANdisco and Yahoo!.

A Technical lemmas for Theorem 1

In this appendix, we collect together the proofs of various technical lemmas involved in the proof of our main theorem.

A.1 Proof of Lemma 2

Consider a problem $\mathcal{P}$ for which, for any integer $m > m_0$, there is a Turing machine $\mathcal{T}_m$ such that:

(a) the code length of $\mathcal{T}_m$ is at most $f(m)$ bits, and;

(b) it solves every input of length m with probability at least $1/2$, and terminates in time at most $g(m)$.

Under Assumption A, it suffices to show that these two conditions certify that $\mathcal{P} \in \mathbf{P/poly}$.

In order to prove this claim, let $m > m_0$ be arbitrary. Given a sufficiently long random sequence, we may execute $\mathcal{T}_m$ a total of $m+1$ times, with the randomness in each execution being independent. By property (b), for any binary input of length m, each execution has success probability at least $1/2$. By independence, the probability that at least one of the $m+1$ executions is successful is $1-2^{-m-1}$. Since the total number of m-length inputs is at most 2^m, the union bound implies that the $m+1$ executions of $\mathcal{T}_m$ succeed on all inputs with probability at least $1-2^{-m-1}2^m = 1/2 > 0$. Consequently, the probabilistic method implies that there exists some realization of the random sequence—call it R_m—under which $m+1$ executions of $\mathcal{T}_m$ solves the problem for all inputs of length m.

Let C_m be the code defining machine $\mathcal{T}_m$, and consider a Turing machine $\mathcal{T}'_m$ that takes the pair (C_m, R_m) as an advice string, and then uses the string R_m to simulate the execution of $\mathcal{T}_m$ a total of $m+1$ times. From our previous argument, the Turing machine $\mathcal{T}'_m$ solves the problem on every input of length $m > m_0$. Notice that the length of string (C_m, R_m) is of the order $\mathcal{O}(f(m)+mg(m))$ bits, and the running time of $\mathcal{T}'_m$ is of the order $\mathcal{O}(mg(m))$. Finally, for all input lengths $m \le m_0$, a constant-size Turing machine $\mathcal{T}^\star_m$ can solve the problem in constant time. The combination of $\mathcal{T}'_m$ and $\mathcal{T}^\star_m$ provides a Turing machine that certifies $\mathcal{P} \in \mathbf{P/poly}$.

A.2 Proof of Lemma 4

Let $\tau = 8\sqrt{2}\gamma$, then let $C_\tau^\uparrow$ and $C_\tau^\downarrow$ represent the top $\frac{n}{2}$ rows and the bottom $\frac{n}{2}$ rows of C_τ, respectively. We first prove an auxiliary lemma about the random matrix C_τ.

Lemma 8. *There are universal positive constants c_1, c_2 such that, with probability at least $1 - c_1 \exp(-c_2 n)$, the random matrix C_τ has the following properties:*

- *It satisfies the RE condition (8) with parameter γ.*
- *It satisfies the following upper bounds:*

$$\frac{\|C_\tau^\uparrow \theta\|_2^2}{n} \le \frac{1}{3}\|\theta\|_2^2 \quad \textit{and} \quad \frac{\|C_\tau^\downarrow \theta\|_2^2}{n} \le (24\gamma)^2 \, \|\theta\|_2^2 \le \frac{1}{2}\|\theta\|_2^2 \quad \textit{for all } \theta \in \mathbb{B}_0(2k) \tag{25}$$

Proof. Dealing first with the upper block of the matrix C, we have

$$\left\| C_\tau^\uparrow \theta \right\|_2^2 = \frac{1}{4}\frac{n}{2}\frac{1}{3k} \, \|A_k \theta\|_2^2 \,. \tag{26}$$

Recall that the diagonal blocks of A_k are matrices M. The following claim provides an upper bound on its singular values:

Claim 3. *For any vector $u \in \mathbb{R}^{4p}$, we have $\|Mu\|_2^2 \le 8p\|u\|_2^2$.*

Proof. Since the entries of M all belong to $\{-1, 0, 1\}$, if we let M_i denote the i^{th} row of M, then

$$(M_i u)^2 \le \Big(\sum_{j \mid |M_{ij}|=1} |u_j| \Big)^2 \le \big| \{ j \mid |M_{ij}| = 1 \} \big| \, \|u\|_2^2.$$

Summing over all indices $i = 1, 2, \ldots, m + 3p$, we find that

$$\|Mu\|^2 \leq \big|\{(i,j) \mid |M_{ij}| = 1\}\big| \, \|u\|_2^2.$$

Since there are $8p$ nonzero entries in M, the claim follows. □

Returning to the main thread, Claim 3 implies that A_k has singular values bounded by $\sqrt{8tp}$. Putting together the pieces, we have

$$\|A_k \theta\|_2^2 \leq \; 8\, t\, p \, \|\theta\|_2^2 \; \leq \; 8\, k \|\theta\|_2^2, \tag{27}$$

where the final inequality follows since $tp \leq t(\frac{m}{3} + p) = k$. In conjunction, inequalities (26) and (27) imply that $\left\| C_\tau^{\uparrow} \theta \right\|_2^2 \leq n/3 \, \|\theta\|_2^2$.

Our next step is to prove that $\|C_\tau^{\downarrow}\theta\|_2^2 \leq n/2 \, \|\theta\|_2^2$ with high probability. Since $n \gtrsim k \log d$, standard results for Gaussian random matrices (see Lemma 10 in Appendix C) imply that

$$\left\| C_\tau^{\downarrow} \theta \right\|_2^2 \leq \tau^2 \, (3\sqrt{n/2}\, \|\theta\|_2)^2 \; \overset{(i)}{\leq} \; \frac{n}{2} \|\theta\|_2^2 \qquad \text{for all } \theta \in \mathbb{B}_0(2k) \tag{28}$$

with probability at least $1 - c_1 \exp(-c_2 n)$, where inequality (i) follows since $\tau = 8\sqrt{2}\gamma \leq 1/3$. Combining the upper bounds for $C_\tau^{\uparrow}$ and the upper bound for $C_\tau^{\downarrow}$, we find that the normalization conditions (25) hold with high probability.

It remains to show that the RE condition (8) holds with high probability. By Lemma 10, we have

$$\|C_\tau \theta\|_2^2 \geq \|C_\tau^{\downarrow}\theta\|_2^2 \; \geq \; \tau^2 \big(\frac{\sqrt{n/2}}{8} \|\theta\|_2 \big)^2 \; = \; \gamma^2 \, n \, \|\theta\|_2^2,$$

a bound that holds uniformly for all $\theta \in \bigcup_{\substack{S \subset \{1,\ldots,d\} \\ |S| = k}} \mathbb{C}(S)$, as required by the condition (8). □

According to Lemma 8, there is a realization of the random matrix R such that the matrix C_τ satisfies both the condition (25), and the RE condition (8) with parameter γ. We take this realization of R to define a concrete realization of C_τ. Consequently, the RE constant of this matrix, namely $\gamma(C_\tau)$, satisfies $\gamma(C_\tau) \geq \gamma$. For arbitrary matrices $X, Y \in \mathbb{R}^{n \times d}$, it is straightforward that

$$|\gamma(X) - \gamma(Y)| \leq \|X - Y\|_{\text{op}}, \tag{29}$$

showing that the function $X \mapsto \gamma(X)$ is a Lipschitz function with parameter 1, and hence continuous. Consequently, $\gamma(C_x)$ is a continous function of x. We also claim that it satisfies the condition

$$\gamma(C_0) = 0, \tag{30}$$

a claim proved at the end of this section. Based on the continuity property (29) and the initial condition (30), there is a constant $\tau' \in (0, \tau]$ such that $\gamma(C_{\tau'}) = \gamma$. Since C_τ satisfies the normalization condition (25) and $\tau' \leq \tau$, we have

$$\frac{\|C_{\tau'}\theta\|_2^2}{n} \leq \frac{\|C_\tau \theta\|_2^2}{n} \leq \frac{5}{6}\|\theta\|_2^2 \quad \text{and} \quad \frac{\|C_{\tau'}^{\downarrow}\theta\|_2^2}{n} \leq \frac{\|C_\tau^{\downarrow}\theta\|_2^2}{n} \leq (24\gamma)^2 \|\theta\|_2^2 \quad \text{for all } \theta \in \mathbb{B}_0(2k). \tag{31}$$

Next we consider the quantized matrix $\lfloor C_{\tau'} \rfloor_L$. Since the quantization operator approximates every entry of $C_{\tau'}$ to precision 2^{-L}, we have

$$|\!|\!| \lfloor C_{\tau'} \rfloor_L - C_{\tau'} |\!|\!|_{\mathrm{op}} \le |\!|\!| \lfloor C_{\tau'} \rfloor_L - C_{\tau'} |\!|\!|_{\mathrm{F}} \le 2^{-L} \sqrt{nd}. \tag{32}$$

Combining this inequality with the Lipschitz condition (29), we find that

$$|\gamma(\lfloor C_{\tau'} \rfloor_L) - \gamma| \le 2^{-L}\sqrt{nd},$$

showing that the quantized version satisfies the stated RE condition as long as $L \ge \log(\sqrt{nd}/\varepsilon)$. Turning to the normalization conditions, inequality (31) and inequality (32) imply that

$$\frac{\|\lfloor C_{\tau'} \rfloor_L \theta\|_2}{\sqrt{n}} \le \frac{\|C_{\tau'}\theta\|_2 + 2^{-L}\sqrt{nd}\,\|\theta\|_2}{\sqrt{n}} \le (\sqrt{5/6} + 2^{-L}\sqrt{d})\,\|\theta\|_2 \quad \text{and}$$
$$\frac{\|\lfloor C^{\downarrow}_{\tau'} \rfloor_L \theta\|_2}{\sqrt{n}} \le \frac{\|C^{\downarrow}_{\tau'}\theta\|_2 + 2^{-L}\sqrt{nd}\,\|\theta\|_2}{\sqrt{n}} \le (24\gamma + 2^{-L}\sqrt{d})\,\|\theta\|_2 \quad \text{for all } \theta \in \mathbb{B}_0(2k).$$

It is straightforward to verify that if $L \ge \max\{\log(12\sqrt{d}), \log(\sqrt{nd}/\varepsilon)\}$, then the normalization conditions (18) stated in the lemma are also satisfied.

Proof of equation (30): Notice that $\mathrm{rank(C_0)} = \mathrm{rank(A_k)} \le \mathrm{3k}$, but the number of nonzero columns of C_0 is greater than $3k$. Hence, there is a $(3k+1)$-sparse nonzero vector θ^+ such that

$$C_0 \theta^+ = 0. \tag{33}$$

Let S be a set of indices that correspond to the k largest entries of θ^+ in absolute value. Then $|S| = k$ and $\left\|\theta^+_{S^c}\right\|_1 \le \frac{2k+1}{k} \left\|\theta^+_S\right\|_1$. The later inequality implies that

$$|S| = k \quad \text{and} \quad \theta^+ \in \mathbb{C}(S). \tag{34}$$

Combining expressions (33) and (34) yields that $\gamma(C_0) = 0$.

A.3 Proof of Lemma 5

Letting q' and $q^\diamond$ denote the probability density functions of y' and $y^\diamond$, respectively, we have

$$\begin{aligned}
\mathbb{E}\big[\|Mu^\diamond_i - Mu^*\|_2^2\big] &= \int_{\mathbb{R}^n} \|M\widehat{\theta}_i(X,z) - Mu^*\|_2^2 \, q^\diamond(z)dz \\
&= \int_{\mathbb{R}^n} \frac{q^\diamond(z)}{q'(z)} \|M\widehat{\theta}_i(X,z) - Mu^*\|_2^2 \, q'(z)dz \\
&= \mathbb{E}\Big[\frac{q^\diamond(y')}{q'(y')} \, \|Mu'_i - Mu^*\|^2\Big] \\
&\ge e^{-1}\, \mathbb{E}\Big[\|Mu'_i - Mu^*\|_2^2 \mid \frac{q^\diamond(y')}{q'(y')} \ge e^{-1}\Big] \, \mathbb{P}\Big[\frac{q^\diamond(y')}{q'(y')} \ge e^{-1}\Big].
\end{aligned} \tag{35}$$

Recall that y' and $y^\diamond$ differ in distribution only in their last $n/2$ entries, denoted by y'_{bot} and $y^\diamond_{\mathrm{bot}}$ respectively. Since $y'_{\mathrm{bot}} \sim N(X^{\downarrow}\tilde{\theta}^{\,i}, \sigma^2 I)$ and $y^\diamond_{\mathrm{bot}} \sim N(X^{\downarrow}\tilde{\xi}^{\,i}, \sigma^2 I)$, we have

$$\frac{q^\diamond(y')}{q'(y')} = \exp\Big(\frac{(X^{\downarrow}\tilde{\theta}^{\,i} - y'_{\mathrm{bot}})^2}{2\sigma^2} - \frac{(X^{\downarrow}\tilde{\xi}^{\,i} - y'_{\mathrm{bot}})^2}{2\sigma^2}\Big). \tag{36}$$

Define the scalar $a := \left\| X^{\downarrow}(\tilde{\xi}^{\,i} - \tilde{\theta}^{\,i}) \right\|_2 / \sigma$, and define the event $\mathcal{E} := \{Z \geq \frac{a}{2} - \frac{1}{a}\}$ where $Z \sim N(0,1)$ is a standard normal variate. From the representation (36), some algebra shows that the probability that the lower bound $\frac{q^{\diamond}(y')}{q'(y')} \geq e^{-1}$ holds is equal to the probability that the event $\mathcal{E}$ holds. Letting Φ denote the CDF of a standard normal variate, we have $\mathbb{P}[\mathcal{E}] = 1 - \Phi\big(\frac{a}{2} - \frac{1}{a}\big)$. It can be verified that $\Phi\big(\frac{a}{2} - \frac{1}{a}\big) \leq a$ for all $a \geq 0$, whence

$$\mathbb{P}[\mathcal{E}] \geq 1 - a = 1 - \frac{\|X^{\downarrow}(\tilde{\xi}^{\,i} - \tilde{\theta}^{\,i})\|_2}{\sigma} \;\geq\; 1 - \frac{25\gamma\sqrt{n}\,\|\tilde{\xi}^{\,i} - \tilde{\theta}^{\,i}\|_2}{\sigma},$$

where the last step uses inequality (18). Since $\tilde{\xi}^{\,i}$ and $\tilde{\theta}^{\,i}$ differ only in one subvector—more precisely, by ρu^*—we have

$$\mathbb{P}[\mathcal{E}] \geq \frac{25\gamma\sqrt{n}\|\rho u^*\|_2}{\sigma} \overset{(i)}{\geq} 1 - \frac{25\gamma r\sqrt{n}}{\sigma} \overset{(ii)}{\geq} \frac{1}{2},$$

where step (i) follows since $\|\rho u^*\|_2 \leq r$, and step (ii) follows since $1 - \frac{25\gamma r\sqrt{n}}{\sigma} \geq 1/2$ by assumption. Plugging this lower bound into inequality (35) yields

$$\mathbb{E}\big[\|Mu_i' - Mu^*\|_2^2 \mid \mathcal{E}\big] \leq 2e\,\mathbb{E}\big[\|Mu_i^{\diamond} - Mu^*\|_2^2\big].$$

Applying Markov's inequality yields

$$\mathbb{P}\Big[\|Mu_i' - Mu^*\|_2^2 \leq \frac{\sigma}{25\gamma r\sqrt{n}} \cdot 2e\,\mathbb{E}\big[\|Mu_i^{\diamond} - Mu^*\|_2^2\big] \mid \mathcal{E}\Big] \geq 1 - \frac{25\gamma r\sqrt{n}}{\sigma}$$

Consequently, with probability $(1 - \frac{25\gamma r\sqrt{n}}{\sigma})^2 \geq 1 - \frac{50\gamma r\sqrt{n}}{\sigma}$, we have

$$\|Mu_i' - Mu^*\|_2^2 \leq \frac{2e\sigma}{25\gamma r\sqrt{n}}\,\mathbb{E}\big[\|Mu_i^{\diamond} - Mu^*\|_2^2\big],$$

as claimed.

A.4 Proof of Lemma 6

As in the proof of Lemma 5, let q' and $q^{\diamond}$ denote the probability density functions of y' and $y^{\diamond}$. With this notation, we have

$$\begin{aligned}
\mathbb{P}\big[u_i' \in \mathbb{B}_0(k/t)\big] &= \int_{\mathbb{R}^n} \mathbb{I}\big[\widehat{\theta}(X,z) \in \mathbb{B}_0(k/t)\big]\, q'(z)dz \\
&= \int_{\mathbb{R}^n} \frac{q'(z)}{q^{\diamond}(z)} \mathbb{I}\big[\widehat{\theta}(X,z) \in \mathbb{B}_0(k/t)\big]\, q^{\diamond}(z)dz \\
&= \mathbb{E}\left[\frac{q'(y^{\diamond})}{q^{\diamond}(y^{\diamond})}\,\mathbb{I}\big[u_i^{\diamond} \in \mathbb{B}_0(k/t)\big]\right] \\
&\geq e^{-1}\,\mathbb{P}\Big[u_i^{\diamond} \in \mathbb{B}_0(k/t) \text{ and } \frac{q'(y^{\diamond})}{q^{\diamond}(y^{\diamond})} \geq e^{-1}\Big] \\
&\geq e^{-1}\,\Big\{\mathbb{P}\big[u_i^{\diamond} \in \mathbb{B}_0(k/t)\big] - \mathbb{P}\Big[\frac{q'(y^{\diamond})}{q'(y^{\diamond})} < e^{-1}\Big]\Big\}.
\end{aligned}$$

Following the same argument as in the proof of Lemma 5, we find that

$$\mathbb{P}\Big[\frac{q'(y^\diamond)}{q^\diamond(y^\diamond)} < e^{-1}\Big] \leq \frac{25\gamma r\sqrt{n}}{\sigma}.$$

Thus, if $\mathbb{P}\big[u_i^\diamond \in \mathbb{B}_0(k/t)\big] \geq \frac{t}{2k+2t}$, then we are guaranteed that

$$\mathbb{P}\big[u_i' \in \mathbb{B}_0(k/t)\big] \geq e^{-1}\Big(\frac{t}{2k+2t} - \frac{25\gamma r\sqrt{n}}{\sigma}\Big). \tag{37}$$

Finally, let $T \subseteq \{1,\ldots,t\}$ be the subset of indices for which $u_i^\diamond \in \mathbb{B}_0(k/t)$ with probability at least $\frac{t}{2k+2t}$. Using $\widehat{\theta}_i$ as a shorthand for $\widehat{\theta}_i(X,y)$, note that $u_i^\diamond$ and $\widehat{\theta}_i/\rho$ share the same distribution, and hence

$$\mathbb{P}\big[u_i^\diamond \in \mathbb{B}_0(k/t)\big] = \mathbb{P}\big[\widehat{\theta}_i \in \mathbb{B}_0(k/t)\big]. \tag{38}$$

Since $\widehat{\theta} \in \mathbb{B}_0(k)$ by construction, we have

$$k \geq \|\widehat{\theta}\|_0 \;\geq\; \sum_{i=1}^t \|\widehat{\theta}_i\|_0 \geq \big(\frac{k}{t}+1\big)\sum_{i=1}^t \mathbb{I}\big[\widehat{\theta}_i \notin \mathbb{B}_0(k/t)\big] \;=\; \frac{k+t}{t}\Big\{t - \sum_{i=1}^t \mathbb{I}\big[\widehat{\theta}_i \in \mathbb{B}_0(k/t)\big]\Big\}. \tag{39}$$

Following some algebra, we find that $\sum_{i=1}^t \mathbb{I}\big[\widehat{\theta}_i \in \mathbb{B}_0(k/t)\big] \geq \frac{t^2}{k+t}$, and hence

$$\sum_{i=1}^t \mathbb{P}[\widehat{\theta}_i \in \mathbb{B}_0(k/t)] \geq \frac{t^2}{k+t}.$$

Let N be the number of indices such that $\mathbb{P}[\widehat{\theta}_i \in \mathbb{B}_0(k/t)] \geq \frac{t}{2k+2t}$. By definition, there are $t-N$ indices for which $\mathbb{P}[\widehat{\theta}_i \in \mathbb{B}_0(k/t)] < \frac{t}{2k+2t}$, and hence

$$\frac{t^2}{k+t} \leq \sum_{i=1}^t \mathbb{P}\big[\widehat{\theta}_i \in \mathbb{B}_0(k/t)\big] \leq N + (t-N)\,\frac{t}{2k+2t},$$

which implies that $N \geq \frac{t^2}{2k+t}$, as claimed. By the equivalence (38), we have $N = |T|$, so that the proof is complete.

A.5 Proof of Lemma 7

Recall that the noise vector w follows the normal distribution $N(0,\sigma^2 I_{n\times n})$. Consequently, if we define the event $\mathcal{E} := \{\|w\|_\infty \leq \mu\}$, then standard tail bounds and the union bound imply that

$$\mathbb{P}[\mathcal{E}^c] \leq n\,\Big\{\frac{2}{\sqrt{2\pi}\mu}e^{-\mu^2/2}\Big\}.$$

Consequently, if we choose $\mu := \sqrt{2\log(100nk^\alpha)}$, then $\mathbb{P}[\mathcal{E}^c] \leq 0.01\,k^{-\alpha}$. Note that u_i' uses the random vectors $(\tilde{\theta}^i, w)$ as input; consequently, we can condition on the realization of $(\tilde{\theta}^i, w)$. The law of total probability and inequality (22) then imply that

$$\int_{\mathbb{R}^n}\sum_{\theta}\mathbb{P}\Big[u_i' \in \mathbb{B}_0(k/t) \text{ and } \big\|Mu_i' - Mu^*\big\|_2^2 \leq 64ek^\alpha\mathbb{E}\big[\,\|Mu_i^\diamond - Mu^*\|_2^2\,\big] \mid \tilde{\theta}^i = \theta, w = z\Big]$$
$$\times\,\mathbb{P}[\tilde{\theta}^i = \theta]\;\phi(z;0,\sigma^2 I_{n\times n})\,\mathrm{d}z \geq 0.03k^{-\alpha},$$

where $\phi(\cdot; 0, \sigma^2 I_{n\times n})$ denotes the density of the $N(0, \sigma^2 I_{n\times n})$ distribution. Splitting the integral up into two terms, indexed by $\mathcal{E}$ and $\mathcal{E}^c$ respectively, we find that

$$\int_{\mathcal{E}} \sum_{\theta} \mathbb{P}\Big[u_i' \in \mathbb{B}_0(k/t) \text{ and } \big\|Mu_i' - Mu^*\big\|_2^2 \leq 64ek^{\alpha}\mathbb{E}\big[\,\|Mu_i^{\diamond} - Mu^*\|_2^2\,\big] \mid \tilde{\theta}^{\,i} = \theta, w = z\Big] \\ \times \mathbb{P}[\tilde{\theta}^{\,i} = \theta]\; \phi(z; 0, \sigma^2 I_{n\times n})\, \mathrm{d}z \geq 0.03\, k^{-\alpha} - \mathbb{P}[\mathcal{E}^c] \geq 0.02\, k^{-\alpha}. \tag{40}$$

Consequently, there must be some specific values (θ_0, w_0) with $\|w_0\|_\infty \leq \mu$ such that

$$\mathbb{P}\Big[u_i' \in \mathbb{B}_0(k/t) \text{ and } \big\|Mu_i' - Mu^*\big\|_2^2 \leq 64\, e\, k^{\alpha}\mathbb{E}\big[\,\|Mu_i^{\diamond} - Mu^*\|_2^2\,\big] \mid \tilde{\theta}^{\,i} = \theta_0, w = w_0\Big] \geq 0.02\, k^{-\alpha}.$$

Let $\widehat{u}_i^*$ be the solver that uses the deterministic argument (θ_0, w_0) rather than drawing $(\tilde{\theta}^{\,i}, w)$ randomly according to the specified procedure. The remaining steps followed by $\widehat{u}_i^*$ are exactly the same as those of u_i'. By this definition, the above inequality implies that

$$\widehat{u}_i^* \in \mathbb{B}_0(k/t) \text{ and } \|M\widehat{u}_i^* - Mu^*\|_2^2 \leq 64ek^{\alpha}\mathbb{E}\big[\,\|Mu_i^{\diamond} - Mu^*\|_2^2\,\big] \tag{41}$$

with probability at least $0.02\, k^{-\alpha}$.

We now demonstrate that the solver $\widehat{u}_i^*$ has polynomial complexity in terms of code length and running time:

Code length: We begin by encoding the design matrix X and fixed vectors (θ_0, w_0) into the program of the solver $\widehat{u}_i^*$. Recall that both X and θ_0 are discretized to 2^{-L} precision. For any discrete value x of 2^{-L} precision, encoding it takes at most $\log(|x|+1)+L$ bits. Recalling our choice (19b) of L, is straightforward to verify that $L = \mathcal{O}(\log(\text{poly}(n,d,k)) + \log(1/\varepsilon))$. Moreover, the components of the matrix X and vector θ_0 are adequately controlled: in particular, by inequality (18) and equation (19a), each such entry x is bounded by $\log(|x|+1) = \mathcal{O}(\text{poly}(n,d,k)) + \log(1/\varepsilon))$, Since $\log(1/\varepsilon) = G(n,d,k)$ is a polynomial functions of m, we find that both X and θ_0 have polynomial code length.

Now the noise vector w_0 is involved in computing the response vector

$$y' = \begin{bmatrix} X^{\uparrow}\, \xi_0 \\ X^{\downarrow}\, \theta_0 \end{bmatrix} + w_0,$$

where ξ_0 is a realization of $\tilde{\xi}^{\,i}$ induced by θ_0. Since the estimator $\widehat{\theta}$ takes $\lfloor y' \rfloor_{G(n,d,k)}$ as input, it suffices to quantize w_0 sufficiently finely so as to ensure that the quantized response $\lfloor y' \rfloor_{G(n,d,k)}$ is not altered by the quantization. Note that the components of X, θ_0 and ξ_0 all have 2^{-L} precision. Consequently, their products have at most 2^{-2L} precision, and we have

$$\left\lfloor \begin{bmatrix} X^{\uparrow}\, \xi_0 \\ X^{\downarrow}\, \theta_0 \end{bmatrix} + w_0 \right\rfloor_G = \left\lfloor \begin{bmatrix} X^{\uparrow}\, \xi_0 \\ X^{\downarrow}\, \theta_0 \end{bmatrix} + \lfloor w_0 \rfloor_{\max\{2L, G\}} \right\rfloor_G,$$

showing that it suffices to quantize the noise w_0 at level $\lfloor \cdot \rfloor_{\max\{2L,G\}}$. On the other hand, by construction, the absolute values of w_0 are uniformly bounded by $\mu = \sqrt{2\log(100nk^{\alpha})}$. Hence, the vector w_0 can also be encoded in polynomial code length.

Finally, we have assumed that code length of $\widehat{\theta}$ is bounded by b, and observed that the program that constructs y' is of constant code length. Consequently, the total code length is bounded by a polynomial function of m, which we denote by $f(m)$.

Running time: Since the response vector y' can be constructed from $(M, Mu^*, X, \theta_0, w_0)$ in polynomial time and the solver $\widehat{\theta}$ terminates in polynomial time (bounded by $H(\text{size}(X, y'; G))$), we conclude that $\widehat{u}_i^*$ also terminates in polynomial time, say upper bounded by the polynomial $g(m)$.

To complete the argument, we proceed via proof by contradiction. If it were the case that $\mathbb{E}\big[\|Mu_i^\diamond - Mu^*\|_2^2\big] < \frac{1}{256e}\, k^{-\alpha}$, then inequality (41) would imply that

$$\widehat{u}_i^* \in \mathbb{B}_0(m/3 + p) \text{ and } \|M\widehat{u}_i^* - Mu^*\|_2 < 1/2 \tag{42}$$

with probability at least $0.02\, k^{-\alpha}$. However, this bound contradicts the hardness result of Lemma 3. Indeed, this lemma guarantees that a polynomial-complexity solver cannot achieve probability of success greater than $1/h(m)$. If we choose the polynomial h such that $1/h(m) < 0.02\, k^{-\alpha}$, then we have obtained the desired contradiction with expression (42). Hence, we conclude that $\mathbb{E}\big[\, \|Mu_i^\diamond - Mu^*\|_2^2\,\big] \geq \frac{1}{256e}\, k^{-\alpha}$, as claimed.

B Proof of Proposition 2

We begin by defining the error vectors $\widehat{\Delta} := \widehat{\theta}_{\lambda_n} - \theta^*$ of the ordinary Lasso, as well as that $\widetilde{\Delta} := \widehat{\theta}_{\text{TL}} - \theta^*$ of the thresholded Lasso. By construction, the error vector $\widetilde{\Delta}$ is at most $2k$-sparse, so that the normalization condition (9) guarantees that

$$\frac{\|X\widetilde{\Delta}\|_2^2}{n} \leq \|\widetilde{\Delta}\|_2^2. \tag{43}$$

The following lemma shows that the truncated Lasso is essentially as good as the non-truncated Lasso:

Lemma 9. *The error of the truncated Lasso is bounded as* $\|\widetilde{\Delta}\|_2^2 \leq 5\|\widehat{\Delta}\|_2^2$.

We return to prove this intermediate lemma at the end of this section.

Taking Lemma 9 as given for the moment, we complete the proof by bounding the error of the ordinary Lasso using Corollary 2 in [15]. With the specified choice of λ_n, this corollary implies that

$$\|\widehat{\Delta}\|_2^2 \leq \frac{64}{\gamma^2(X)} \, \frac{\sigma^2 k \log d}{n}, \tag{44}$$

with probability at least $1 - 2e^{-c_4 k \log d}$. Combining the bound (44), Lemma 9 and inequality (43) yields the claim of the proposition.

Finally, we return to prove Lemma 9. Throughout this proof, we use $\widehat{\theta}$ as a shorthand for the thresholded estimator $\widehat{\theta}_{\text{TL}}$. Since $\widehat{\theta}$ is the k-truncated version of $\widehat{\theta}_{\lambda_n}$, if $|\text{supp}(\widehat{\theta}_{\lambda_n})| \leq k$, we must have $\widehat{\theta} = \widehat{\theta}_{\lambda_n}$, in which case the proof is complete. Otherwise, we may assume that $|\text{supp}(\widehat{\theta}_{\lambda_n})| > k$, and then define the set $G := \big\{\text{supp}(\theta^*) \cap \text{supp}(\widehat{\theta}_{\lambda_n})\big\} \backslash \text{supp}(\widehat{\theta})$, corresponding to "good" entries in the estimate $\widehat{\theta}_{\lambda_n}$ that were lost in the truncation. Introduce the notation $G = \{i_1, \ldots, i_t\}$ where $t = |G|$. Since $|\text{supp}(\theta^*)| \leq k$, the set $\text{supp}(\widehat{\theta}_{\lambda_n}) \backslash \text{supp}(\theta^*)$ must contain at least t indices, say $B = \{j_1, \ldots, j_t\}$, corresponding to "bad" entries of $\widehat{\theta}_{\lambda_n}$ that were preserved in the truncation. By definition of the truncation operation, they satisfy the bound

$$|(\widehat{\theta}_{\lambda_n})_{i_s}| \leq |(\widehat{\theta}_{\lambda_n})_{j_s}| \qquad \text{for all } s = 1, \ldots, t. \tag{45}$$

Now consider the decomposition

$$\|\widehat{\theta}-\theta^*\|_2^2 = \sum_{i=1}^{d}(\widehat{\theta}_i-\theta^*_i)^2 = \sum_{s=1}^{t}\Big((\widehat{\theta}_{i_s}-\theta^*_{i_s})^2+(\widehat{\theta}_{j_s}-\theta^*_{j_s})^2\Big)+\sum_{i\notin B\cup G}(\widehat{\theta}_i-\theta^*_i)^2. \tag{46}$$

The proof will be complete if we can establish the two inequalities

$$(\widehat{\theta}_i-\theta^*_i)^2 \le ((\widehat{\theta}_{\lambda_n})_i-\theta^*_i)^2 \quad \text{if } i\notin B\cup G, \text{ and} \tag{47a}$$

$$(\widehat{\theta}_{i_s}-\theta^*_{i_s})^2+(\widehat{\theta}_{j_s}-\theta^*_{j_s})^2 \le 5((\widehat{\theta}_{\lambda_n})_{i_s}-\theta^*_{i_s})^2+5((\widehat{\theta}_{\lambda_n})_{j_s}-\theta^*_{j_s})^2 \qquad \text{for } s=1,\ldots,t. \tag{47b}$$

Inequality (47a) is straightforward, since $i\notin B\cup G$ implies either $\widehat{\theta}_i=(\widehat{\theta}_{\lambda_n})_i$ or $\widehat{\theta}_i=\theta^*_i=0$.

Turning to the inequality (47b), note that $\widehat{\theta}_{i_s}=\theta^*_{j_s}=0$, so that it is equivalent to upper bound $(\theta^*_{i_s})^2+(\widehat{\theta}_{\lambda_n})^2_{j_s}$. We divide the proof of inequality (47b) into two cases:

Case 1: First, suppose that $|(\widehat{\theta}_{\lambda_n})_{i_s}-\theta^*_{i_s}|\le\frac{1}{2}|\theta^*_{i_s}|$. In this case, we have $|(\widehat{\theta}_{\lambda_n})_{i_s}|\ge\frac{1}{2}|\theta^*_{i_s}|$, and hence by inequality (45), we obtain $|(\widehat{\theta}_{\lambda_n})_{j_s}|\ge\frac{1}{2}|\theta^*_{i_s}|$. Combining the pieces yields

$$(\theta^*_{i_s})^2+((\widehat{\theta}_{\lambda_n})_{j_s})^2 \le 5((\widehat{\theta}_{\lambda_n})_{j_s})^2 \;\le\; 5((\widehat{\theta}_{\lambda_n})_{i_s}-\theta^*_{i_s})^2+5((\widehat{\theta}_{\lambda_n})_{j_s}-\theta^*_{j_s})^2.$$

Case 2: On the other hand, if $|(\widehat{\theta}_{\lambda_n})_{i_s}-\theta^*_{i_s}|>\frac{1}{2}|\theta^*_{i_s}|$, then we have

$$\begin{aligned}(\widehat{\theta}_{i_s}-\theta^*_{i_s})^2+(\widehat{\theta}_{j_s}-\theta^*_{j_s})^2 &= (\theta^*_{i_s})^2+((\widehat{\theta}_{\lambda_n})_{j_s})^2\\ &< 4((\widehat{\theta}_{\lambda_n})_{i_s}-\theta^*_{i_s})^2+((\widehat{\theta}_{\lambda_n})_{j_s})^2\\ &\le 5((\widehat{\theta}_{\lambda_n})_{i_s}-\theta^*_{i_s})^2+5((\widehat{\theta}_{\lambda_n})_{j_s}-\theta^*_{j_s})^2.\end{aligned}$$

Combining the two cases completes the proof of inequality (47b).

C Singular values of random matrices

In this section, we provide some background on the singular value of Gaussian random matrices, required in the proof of Theorem 1(a). Our results apply to a random matrix $A\in\mathbb{R}^{n\times d}$ formed of i.i.d. $N(0,1)$ entries. Recall the set $\mathbb{C}(S)=\big\{\theta\in\mathbb{R}^d \mid \|\theta_{S^c}\|_1\le 3\|\theta_S\|_1\big\}$ that is involved in the RE condition (8).

Lemma 10. *Suppose that $n>c_0\,k\log d$ for a sufficiently large constant c_0. Then there are universal constants $c_j, j=1,2$ such that*

$$\frac{\|A\theta\|_2}{\sqrt{n}}\le 3\|\theta\|_2 \qquad \textit{for all } \theta\in\mathbb{B}_0(2k), \textit{ and} \tag{48a}$$

$$\frac{\|A\theta\|_2}{\sqrt{n}}\ge\frac{\|\theta\|_2}{8} \qquad \textit{for all } \theta\in\bigcup_{\substack{S\subset\{1,\ldots,d\}\\|S|=k}}\mathbb{C}(S), \tag{48b}$$

where both bounds hold with probability at least $1-c_1\exp(-c_2 n)$.

Proof. To prove the upper bound (48a), it suffices to show that $\max_{|S|=2k} \frac{\|A_T\|_{\text{op}}}{\sqrt{n}} \leq 3$, where A_T denotes the columns of A whose indices are in set T, and $\|\cdot\|_{\text{op}}$ denotes the maximum singular value of a matrix. By known bounds on singular values of Gaussian random matrices [19], for any subset T of cardinality $2k$, we have

$$\mathbb{P}\big[\|A_T\|_{\text{op}} \geq \sqrt{n} + \sqrt{2k} + \delta\big] \leq 2\exp(-\delta^2/2) \qquad \text{for all } \delta > 0.$$

If we let $\delta = \sqrt{n}$ and use the assumption that $2k \leq n$, the above bound becomes

$$\mathbb{P}\big[\|A_T\|_{\text{op}} \geq 3\sqrt{n}\big] \leq 2\exp(-n/2).$$

Noting that there are $\binom{d}{2k} \leq e^{2k\log(\frac{ed}{2k})}$ subsets of cardinality $2k$, the claim thus follows by union bound.

On the other hand, the lower bound (48b) is implied by the main result of Raskutti et al. [17]. □

References

[1] A. A. Amini and M. J. Wainwright. High-dimensional analysis of semidefinite relaxations for sparse principal component analysis. *Annals of Statistics*, 5:2877–2921, 2009.

[2] S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[3] Q. Berthet and P. Rigollet. Optimal detection of sparse principal components in high dimensions. Technical report, Princeton University, 2012. arxiv1202.5070.

[4] Q. Berthet and P. Rigollet. Computational lower bounds for sparse PCA. Technical report, Princeton University, April 2013. arxiv1304.0828.

[5] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of Lasso and Dantzig selector. *Annals of Statistics*, 37(4):1705–1732, 2009.

[6] F. Bunea, A. Tsybakov, and M. Wegkamp. Aggregation for Gaussian regression. *Annals of Statistics*, 35(4):1674–1697, 2007.

[7] E. Candes and T. Tao. The Dantzig selector: statistical estimation when p is much larger than n. *Annals of Statistics*, 35(6):2313–2351, 2007.

[8] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012.

[9] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

[10] R. Foygel and N. Srebro. Fast rate and optimistic rate for ℓ_1-regularized regression. Technical report, Toyoto Technological Institute, 2011. arXiv:1108.037v1.

[11] R. Krauthgamer, B. Nadler, and D. Vilenchik. Do semidefinite relaxations really solve sparse PCA? Technical report, Weizmann Institute of Science, June 2013. arXiv:1306.3690v1.

[12] Z. Ma and Y. Wu. Computational barriers in minimax submatrix detection. *arXiv preprint arXiv:1309.5914*, 2013.

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

[14] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, 1995.

[15] S. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. *Statistical Science*, 27 (4):538–557, 2012.

[16] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi. Simultaneously structured models with applications to sparse and low-rank matrices. Technical report, Caltech, 2012. arxiv1212.3753.

[17] G. Raskutti, M. J. Wainwright, and B. Yu. Restricted eigenvalue properties for correlated Gaussian designs. *Journal of Machine Learning Research*, 99:2241–2259, 2010.

[18] G. Raskutti, M. J. Wainwright, and B. Yu. Minimax rates of estimation for high-dimensional linear regression over ℓ_q-balls. *IEEE Transactions on Information Theory*, 57(10):6976–6994, 2011.

[19] M. Rudelson and R. Vershynin. Non-asymptotic theory of random matrices: extreme singular values. *arXiv:1003.2990*, 2010.

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

[21] S. A. Van De Geer and P. Bühlmann. On the conditions used to prove oracle results for the Lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009.