On learning sparse vectors from mixture of responses

Anonymous Author(s)
Affiliation
Address
email

Abstract

In this paper, we address two learning problems. Suppose a family of ℓ unknown sparse vectors is fixed, where each vector has k non-zero elements only. In the first problem, we aim at designing queries such that all sparse vectors from the family can be reconstructed using a sequence of responses. Each response to a query vector shows the sign of the inner product between a randomly chosen vector from the family and the query vector. In the second problem, we concentrate on robust learning only the supports of all vectors from the family in the setting in which a fraction of responses can be erroneously flipped. This model naturally generalizes its noiseless version discussed in the recent work of Gandikota et al., 2020, and the classical model of 1-bit compressed sensing with noisy measurements. As the main contribution, we present learning algorithms for both problems and rigorously analyze their query complexity.

1 Introduction

Many digital data acquisition methods can be expressed as recovering an unknown real-valued sparse vector from linear measurements. Compressed sensing is a signal processing methodology proposed in [8, 4] to make the process of vector reconstruction reliable and efficient. Criticism of classical compressed sensing is that it assumes infinite precision of real-valued measurements. One interesting setup for compressed sensing is that the measurement outcomes are quantized, e.g., only signs of measurements are captured. Different aspects of the 1-bit compressed sensing problem have been extensively investigated in recent years [1, 22, 2, 13, 16, 11]. One important sub-problem that often arises in 1-bit compressed sensing is to learn the support of an unknown sparse vector. A vast body of literature focuses on this task and gives bounds on the minimum number of measurements sufficient for support recovery [1, 13, 11].

In this paper, we investigate a model which can be seen as a generalization of 1-bit compressed sensing. In this model, a family of ℓ unknown sparse vectors is fixed, and the response to a query vector shows the sign of the inner product between the query vector and a vector picked uniformly at random from the family. The goal is to reconstruct the vectors (or only the supports of the vectors) using the least number of queries. In real-life applications, the measurements are suffered by noise, which causes sign flips. We also suggest considering a variation of the model that admits noisy measurements.

Our problem statement is in spirit of many machine learning models, e.g., a mixture of linear regression models [6, 17, 24, 15, 27, 26, 5] and a mixture of binary linear classifiers [26]. However, we stress an important difference from the classical machine learning setting: in our model, one constructs data points and an oracle provides the label for each of them; the goal is to find the minimal number of data points sufficient for learning all the components in a mixture of binary linear classifiers. Recent works on a mixture of sparse linear regressions considered a similar setting and studied query complexity of learning algorithms [28, 20, 18].

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

The most relevant research to our work is provided in [12], where the noiseless version of the proposed model was discussed and some single-stage and two-stage learning strategies were provided. The main technical contribution presented in that paper is an algorithm for finding the support of the sparse vectors. However, its main drawback is that for three or more unknown vectors, it was shown to work only under the support separability assumption: the support of any unknown vector in the family is not covered by the union of the supports of the other unknown vectors. The authors left the problem of support recovery without any assumptions as an open problem. We also highlight that for a family of two unknown sparse vectors, the authors [12] provided a polynomial-time algorithm for the approximate reconstruction of both vectors that works under a mild assumption.

1.1 Our contribution and technique

Our main contribution lies in a new support recovery algorithm that works without any assumptions and is resilient to noisy measurements. Suppose a family of ℓ unknown k-sparse vectors in $\mathbb{R}^n$ is fixed. For any given subset $Q \subset [n]$, we first design queries that enable us to learn the number of vectors whose support intersects Q non-trivially. Based on this knowledge, we show how to reduce the support recovery problem to a problem of quantitative group testing. In this problem, one needs to learn a hidden hypergraph defined over the set $[n]$ with ℓ edges each of size at most k. The response to a query $Q \subset [n]$ shows the piercing number, the number of edges overlapping with Q. Here, we arrive to the concept of (ℓ, k, α)-robust-resolvable matrices that allow us to uniquely reconstruct any hidden sparse hypergraph based on a sequence of responses. In the combinatorial coding theory literature, this concept was analyzed only when $k = 1$, $\alpha = 0$ [23]. We extend this line of research for $k \geq 1$ and $\alpha \geq 0$ and give an achievability bound for robust-resolvable matrices. In particular, we show a connection between robust-resolvable matrices and a notion that generalizes cover-free families which were previously investigated in extremal set theory [25, 10]. We highlight that the use of some specific subclasses of generalized cover-free families such as union-free families and pairwise independent union-free families was already discussed in the context of various 1-bit compressed sensing problems [12, 1, 13]. However, such a link was used before either in a very specific setting, e.g., $\ell = 1$, or under certain assumptions. We note that all queries of the proposed support recovery algorithm can be carried out in parallel (Theorem 1).

Once the supports of all unknown vectors in the family are reconstructed, it remains to recover the values of these vectors. We consider this task in the noiseless setting and make an assumption: (a) the support of any unknown vector is not fully contained in the support of any other unknown vector from the family, and (b) the magnitude of each entry can be bounded from below and above. It is known that $\mathcal{O}(k/\epsilon \log(n/\epsilon))$ simple Gaussian queries are sufficient to learn an unknown k-sparse vector in $\mathbb{R}^n$ using only signs of linear measurements [16]. Thereby, one natural idea to recover the family of vectors is to model a Gaussian query for every given unknown vector from the family. A similar technique was also used in [12], however, our solutions work under a milder assumption. In a simple two-stage recovery process, we carry out queries required for the approximate reconstruction at the second stage after getting responses to queries used for the support recovery (Theorem 2). In a more sophisticated non-adaptive solution, we re-use the concept of cover-free families that enable us to emulate individual Gaussian queries for any potential family of ℓ unknown sparse vectors (Theorem 3).

We emphasize that our contribution is primarily theoretical as the questions raised in our work are purely mathematical. For instance, we leave the problem of designing a query scheme that allows for an efficient reconstruction algorithm with running time polynomial in the number of queries as an open problem. However, our research has relevant practical applications since the considered model can be used for modeling complex systems with heterogeneous data. According to [12], studying different aspects in such a model would help analyze recommendation systems where the privacy of user's data has to be preserved.

1.2 Outline

The remainder of the paper is organized as follows. We introduce some notations and describe the problem statement in Sec. 2. Our results are summarized in Sec. 3. Sec. 4 describes a robust support recovery algorithm and introduces useful concepts required for the algorithm. We present our approximate recovery algorithms in Sec. 5. Finally, Sec. 6 concludes the paper and discusses

some open research directions. Proofs of several helpful statements and one theorem are delegated to Appendix.

2 Problem statement

For simplicity of presentation, hereafter one-based numbering is used and $\log n$ stands for the base-two logarithm of n. Let $H_2(\alpha)$ denote the binary entropy function, i.e., $H_2(\alpha) = -\alpha \log \alpha - (1-\alpha)\log(1-\alpha)$. The set of integers from 1 to n is abbreviated by $[n]$. A vector is denoted by bold lowercase letters, such as $\boldsymbol{x}$, and the i-th entry of the vector $\boldsymbol{x}$ is referred to as x_i. A binary matrix is denoted by uppercase letters, such as X, and the entry in the i-th row and the j-th column is written as $X_{i,j}$.

We say that a vector $\boldsymbol{x} \in \mathbb{R}^n$ is *k-sparse* if it contains at most k non-zero components. The *support* of a vector $\boldsymbol{x}$, written as $\mathrm{supp}(\boldsymbol{x})$, denotes the set of coordinates corresponding to non-zero components of $\boldsymbol{x}$. We write $\mathbb{1}\{A\}$ to denote the indicator function of an event A. For an integer vector $\boldsymbol{x} \in \mathbb{Z}^m$ and an integer $r \geq 0$, define the *Hamming ball* centered at $\boldsymbol{x}$ with radius r as $B_r(\boldsymbol{x}) := \{\boldsymbol{y} \in \mathbb{Z}^m : \, d_H(\boldsymbol{x}, \boldsymbol{y}) \leq r\}$.

Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors in $\mathbb{R}^n$ that we aim to recover by employing (noisy) binary responses to queries. Throughout the paper we assume that the number of vectors is not too large and the vectors are indeed sparse. Specifically, let $\ell k = o(n)$. Suppose that an oracle returns a (noisy) output $y_i \in \{-1, 1\}$ for a given query vector $\boldsymbol{x}^{(i)} \in \mathbb{R}^n$, $i \in [q]$:

$$y_i = \mathrm{sign}(\langle \boldsymbol{x}^{(i)}, \boldsymbol{\beta} \rangle) \cdot \eta_i, \tag{1}$$

where the random variable $\boldsymbol{\beta}$ is sampled uniformly at random from the set $\mathcal{B}$, the value $\eta_i \in \{-1, 1\}$ is noise, and the sign function is defined as follows

$$\mathrm{sign}(\alpha) = \begin{cases} 1, & \alpha \geq 0, \\ -1, & \alpha < 0. \end{cases}$$

Clearly, for $\eta_i = 1$, the resulting response is exactly the sign of the inner product between $\boldsymbol{x}^{(i)}$ and $\boldsymbol{\beta}$. For the approximate recovery problem, we assume the noiseless case, i.e., $\eta_i = 1$ for all $i \in [q]$. For the support recovery problem, we make the only structural assumption regarding the noise level, namely:

$$\sum_{i=1}^{q} \mathbb{1}\{\eta_i = -1\} \leq \tau q \tag{2}$$

for a given real value $\tau \geq 0$. When only signs of linear measurements are captured, it is not possible to reconstruct the length (or magnitude) of the unknown vectors. Thereby, our first task is to approximately recover the unordered set of directions of ℓ unknown vectors with a given precision. Formally, this problem can be stated as follows.

Problem 1 (ϵ-approximate recovery)**.** *Fix $\tau = 0$. Let $\epsilon > 0$ be a given real value and $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors in $\mathbb{R}^n$. Design queries $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(q)}$ such that with probability $1 - \mathcal{O}(1/n)$, based on a sequence of binary responses $y_1, \ldots, y_q$ provided that* (1)-(2)*, it is possible to find k-sparse vectors $\tilde{\boldsymbol{\beta}}^{(1)}, \ldots, \tilde{\boldsymbol{\beta}}^{(\ell)} \in \mathbb{R}^n$ satisfying the inequality*

$$\max_{i \in [\ell]} \left\| \frac{\boldsymbol{\beta}^{(i)}}{\left\| \boldsymbol{\beta}^{(i)} \right\|} - \tilde{\boldsymbol{\beta}}^{(\sigma(i))} \right\| \leq \epsilon,$$

where $\sigma : [\ell] \to [\ell]$ is some bijection (permutation).

When solving this problem, we follow one typical approach in one-bit compressed sensing that operates in two phases: first, the support of unknown vectors is reconstructed; second, the approximate values of vectors are recovered. Classical compressed sensing helps with the latter task. Compared to the work [12], our main contribution is constructing queries to handle the former task without any assumptions on the support of vectors. Further, our second line of research is devoted to a more difficult problem. We try reconstructing the supports of all vectors based on the noisy measurements.

Problem 2 (τ-robust support recovery)**.** *Let $\tau > 0$ be a given real value and $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \dots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors in $\mathbb{R}^n$. Design queries $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(q)}$ such that with probability $1 - \mathcal{O}(1/n)$, based on a sequence of noisy binary responses $y_1, \dots, y_q$ provided that* (1)-(2)*, it is possible to find k-sparse vectors $\tilde{\boldsymbol{\beta}}^{(1)}, \dots, \tilde{\boldsymbol{\beta}}^{(\ell)} \in \mathbb{R}^n$ such that*

$$\mathrm{supp}(\boldsymbol{\beta}^{(i)}) = \mathrm{supp}(\tilde{\boldsymbol{\beta}}^{(\sigma(i))}),$$

where $\sigma : [\ell] \to [\ell]$ is some bijection (permutation).

3 Our results

In the following three statements, we summarize our main results concerning the above two problems. We show the existence of a robust support recovery algorithm without any assumptions.

Theorem 1 (Robust support recovery algorithm)**.** *Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \dots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors in $\mathbb{R}^n$. Let $\tau \in [0, \frac{\alpha_0}{32\ell})$, where $\alpha_0 := \alpha_0(\ell, k)$ is the least positive root of the function*

$$\begin{aligned} f(\alpha, \ell, k) &:= (2\alpha - 1)\log(1 - p') - 2\alpha \log p' - H_2(2\alpha), \\ p' &:= \left(\frac{\min(\ell, k)}{2k\ell}\right)^{\min(\ell,k)} \left(1 - \frac{\min(\ell, k)}{2k\ell}\right)^{2k\ell - \min(\ell,k)}. \end{aligned} \tag{3}$$

Then there exists an algorithm to learn the support of all vectors in $\mathcal{B}$ with probability at least $1 - \mathcal{O}(1/n)$ using $\mathcal{O}\left(\frac{k\ell^3 \log^2 n}{f(32\tau\ell, \ell, k)}\right)$ non-adaptive queries. For $\tau = 0$, the required number of queries can be estimated as $\mathcal{O}(k\ell^3 \max(2e\ell, 2ek)^{\min(\ell,k)} \log^2 n)$, where e is Euler's number.

Remark 1. *For some parameter regimes, e.g., $\tau = 0$, fixed $k \geq 4$ and large ℓ, Theorem 1 in [12] gives an algorithm with better query complexity. However, unlike [12], our algorithm does not require any structural assumptions for vectors in $\mathcal{B}$. Thereby, it can be considered as a solution for the open question posed in [12]. In addition, the proposed algorithm is robust to a (small) positive fraction of adversarial errors that occur in the measurements.*

The main term in query complexity comes from the derived existential result for robust cover-free families. In any solution that relies on the same concept, it is possible to estimate the number of queries using (almost matching) known converse results for cover-free families [10, 25]. However, we believe that it is possible to further improve query complexity since for the degenerate case $k = 1$ and $\tau = 0$, we also show the existence of an algorithm with $\mathcal{O}(\frac{\ell \log^2 n}{\log \ell})$ queries.

Before presenting approximate recovery algorithms, we introduce a mild assumption that is required for the proper working of the algorithms.

Assumption 1. *Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \dots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors in $\mathbb{R}^n$. For every $i \neq j$, the support of the i-th vector from $\mathcal{B}$ is not contained in the support of the j-th vector from $\mathcal{B}$, i.e., $\mathrm{supp}(\boldsymbol{\beta}^{(i)}) \setminus \mathrm{supp}(\boldsymbol{\beta}^{(j)}) \neq \emptyset$. Some bounds on the magnitude of non-zero entries in all vectors from $\mathcal{B}$ are given, e.g., $\exp(-\mathcal{O}(n)) \leq |\beta_j^{(i)}| \leq \exp(\mathcal{O}(n))$ for all $j \in [n]$ and $i \in [\ell]$ such that $\beta_j^{(i)} \neq 0$.*

Theorem 2 (Two-stage approximate recovery algorithm)**.** *Let $\epsilon > 0$ be a fixed Euclidean precision. Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \dots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of vectors as in Assumption 1. Then there exists an algorithm to recovery all vectors in $\mathcal{B}$ with precision ϵ with probability at least $1 - \mathcal{O}(1/n)$ using $\mathcal{O}(k\ell^3 \max(2e\ell, 2ek)^{\min(\ell,k)} \log^2 n + \ell^3 \frac{k}{\epsilon} \log \frac{k}{\epsilon} \log(n/\epsilon))$ queries that can be asked in two (non-adaptive) stages.*

Theorem 3 (Single-stage approximate recovery algorithm)**.** *Let $\epsilon > 0$ be a fixed Euclidean precision. Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \dots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of vectors as in Assumption 1. Then there exists an algorithm to recovery all vectors in $\mathcal{B}$ with precision ϵ with probability at least $1 - \mathcal{O}(1/n)$ using $\mathcal{O}(\frac{k^2}{\epsilon} \ell^3 \max(2e\ell, 2ek)^{\min(k,\ell)} \log n \log(\frac{n}{\epsilon}) \log(\frac{k}{\epsilon}))$ non-adaptive queries.*

4 Robust support recovery

For a set $Q \subseteq [n]$ and a family $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(\ell)}\}$ of ℓ unknown k-sparse vectors in $\mathbb{R}^n$, we define the *piercing number* of Q in $\mathcal{B}$ as

$$\mathrm{pir}(Q, \mathcal{B}) := \sum_{i=1}^{\ell} \mathbb{1}\{\mathrm{supp}(\boldsymbol{\beta}^{(i)}) \cap Q \neq \emptyset\}.$$

In Sec. 4.1 we first show how to learn this number with overwhelming probability by asking the same query multiple times. In Sec. 4.2 we rephrase the support recovery problem in the language of quantitative group testing and introduce the concept of robust resolvable matrices. The existence of such matrices is discussed in Sec. 4.3 and Sec. 4.4. Finally, in Sec. 4.5 we give a proof of Theorem 1.

4.1 Estimating the piercing number

Let $Q \subseteq [n]$ and $\boldsymbol{x} = (x_1, \ldots, x_n)$ be a vector such that the i-th entry is sampled independently from the continuous uniform distribution $U(0, 1)$ if $i \in Q$ and is 0 otherwise. For an integer m (divisible by 16ℓ), define $\boldsymbol{x}^{(1)} = \ldots = \boldsymbol{x}^{(m/2)} = \boldsymbol{x}$ and $\boldsymbol{x}^{(m/2+1)} = \ldots = \boldsymbol{x}^{(m)} = -\boldsymbol{x}$. For every $i \in [m]$, let y_i be a noisy response defined as in (1). Let $\mathrm{round}(\gamma)$ be a function that takes a real number γ as input and outputs the closest integer number. Define a sample statistic that estimates $\mathrm{pir}(Q, \mathcal{B})$

$$\widetilde{\mathrm{pir}}(Q, m) := \mathrm{round}\left(\frac{\ell}{m}\left(\sum_{i=1}^{m/2} y_i - \sum_{i=m/2+1}^{m} y_i\right)\right)$$

Lemma 1 (Piercing number estimator)**.** *Suppose* $\sum_{i=1}^{m} \mathbb{1}\{\eta_i = -1\} \leq \frac{m}{16\ell}$. *Then it holds that*

$$\Pr\left\{\widetilde{\mathrm{pir}}(Q, m) = \mathrm{pir}(Q, \mathcal{B})\right\} = 1 - \exp(-\Omega(m\ell^{-2})).$$

Before proceeding with a proof of this statement, we recall the famous Hoefding inequality [14] which can be applied to bounded independent random variables, regardless of their distribution.

Proposition 1 (Hoefding's inequality for bounded variables)**.** *Let* $\xi_1, \ldots, \xi_n$ *be independent random variables such that* $a \leq \xi_i \leq b$ *for all* $i \in [n]$. *Set* $\xi = \sum \xi_i$ *and* $\mu = \mathbf{E}[\xi]$. *Then, for all* $\delta > 0$

$$\Pr\{|\xi - \mu| \leq n\delta\} \geq 1 - 2\exp(-2\delta^2 n/(b-a)^2).$$

Proof. From the lemma condition, it follows that there exists a set $I \subseteq [m]$ of at least $(1 - \frac{1}{16\ell})m$ indices such that $y_i = \mathrm{sign}(\langle \boldsymbol{x}^{(i)}, \boldsymbol{\beta}\rangle)$ for all $i \in I$. Moreover, we can find a subset $I' \subseteq I$ such that $|\{i \in I' : i \leq m/2\}| = |\{i \in I' : i > m/2\}| = \frac{8\ell-1}{16\ell}m$. Then we apply Proposition 1 for a set $\{y_i : i \in I', i \leq m/2\} \cup \{y_i : i \in I', i > m/2\}$ with $\mu = \frac{(8\ell-1)m}{8\ell^2}\mathrm{pir}(Q, \mathcal{B})$, $a = -1$, $b = 1$ and $\delta = 1/(2\ell)$. Then we get

$$\Pr\left\{\mathrm{pir}(Q, \mathcal{B}) - \frac{1}{2} < \frac{\ell}{m}\left(\sum_{i=1}^{m/2} y_i - \sum_{i=m/2+1}^{m} y_i\right) < \mathrm{pir}(Q, \mathcal{B}) + \frac{1}{2}\right\}$$
$$\geq 1 - 2\exp\left(-\Omega(m\ell^{-2})\right).$$

This yields that $\widetilde{\mathrm{pir}}(Q, m)$ is a correct estimate of $\mathrm{pir}(Q, \mathcal{B})$ with probability $1 - \exp(-\Omega(m\ell^{-2}))$. □

4.2 Quantitative search model for sparse hypergraphs

A hypergraph is a pair $H = (V, E)$, where V is a set of vertices, and E is a multiset of non-empty subsets of V called hyperedges. A hypergraph $H = (V, E)$ is said to be (ℓ, k)-*sparse* if the number of hyperedges is ℓ and the size of each hyperedge is at most k. Let $E = \{e_1, \ldots, e_\ell\}$. For a subset $Q \subseteq V$, define the integer output $z(Q, H)$ as follows

$$z(Q, H) := \sum_{i=1}^{\ell} \mathbb{1}\{e_i \cap Q \neq \emptyset\}. \tag{4}$$

We now introduce the hypergraph $H_{\mathcal{B}} := (V, E_{\mathcal{B}})$, where the set of vertices $V = [n]$ and the set of hyperedges $E_{\mathcal{B}} = \{e_1, \ldots, e_\ell\}$ contains ℓ hyperedges with $e_i = \mathrm{supp}(\boldsymbol{\beta}^i)$ for all $i \in [\ell]$. Note that without any assumptions, the edge set $E_{\mathcal{B}}$ can possibly contain a hyperedge with large multiplicity if some vectors in $\mathcal{B}$ share the same support. Clearly, $H_{\mathcal{B}}$ is (ℓ, k)-sparse and $z(Q, H_{\mathcal{B}}) = \mathrm{pir}(Q, \mathcal{B})$. In next statement, we derive an obvious extension of Lemma 1.

Lemma 2. *Let $\mathcal{B}$ be a family of unknown sparse vectors and $Q_1, \ldots, Q_m \subseteq [n]$ be a sequence of subsets. Set $\boldsymbol{z} = (z(Q_1, H_{\mathcal{B}}), \ldots, z(Q_m, H_{\mathcal{B}}))$. Let τ be a noise parameter such that $0 \le \tau \le \frac{1}{16\ell}$. Then there exist queries $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(q)} \in \mathbb{R}^n$ with $q = \mathcal{O}(\ell^2 m \log(mn))$ such that with probability $1 - \mathcal{O}(1/n)$, based on a sequence of binary responses $y_1, \ldots, y_q$ provided that* (1)-(2)*, it is possible to reconstruct at least one $\tilde{\boldsymbol{z}} \in \mathbb{Z}^m$ such that $d_H(\tilde{\boldsymbol{z}}, \boldsymbol{z}) \le \alpha m$ with $0 \le \alpha \le 16\tau\ell \le 1$.*

Let X be a binary matrix of size $m \times n$, and $H = (V, E)$ be a hypergraph with $V = [n]$. Set $Q_i = \mathrm{supp}(X_i)$, where X_i is the i-th row of X. With a slight abuse of notation, we denote by

$$\boldsymbol{z}(X, H) := (z_1, \ldots, z_m)$$

the sequence of integer responses, where $z_i = z(Q_i, H)$ is computed as in (4). Lemma 2 naturally motivates us to introduce a notion of robust resolvable matrices.

Definition 1 (Robust resolvable matrix)**.** Fix a real number $\alpha, 0 \le \alpha \le 1/2$, and integers $\ell, k \ge 1$. A matrix X of size $m \times n$ is said to be (ℓ, k, α)-*robust-resolvable* if for any two distinct (ℓ, k)-sparse hypergraphs H_1, H_2, the corresponding Hamming balls centered at $\boldsymbol{z}(X, H_1)$ and $\boldsymbol{z}(X, H_2)$ with radius αm are disjoint, i.e., $d_H(\boldsymbol{z}(X, H_1), \boldsymbol{z}(X, H_2)) \ge 2\alpha m + 1$.

By $m(n, \ell, k, \alpha)$ denote the minimal number of rows in an (ℓ, k, α)-robust-resolvable matrix with n columns. Up to our best knowledge, Definition 1 for $\alpha > 0$ (and $\alpha = 0$, $k \ge 2$) is new in the literature. We proceed our discussion with the simplest case $k = 1$ which might seem degenerate. However, the corresponding analysis is already non-trivial and sheds light on the existence of robust resolvable matrices.

4.3 Existence of $(\ell, 1, 0)$-robust-resolvable matrices

We now review literature on $(\ell, 1, 0)$-resolvable matrices. The best known explicit constructions come from number theory [3, 19] and are closely related to the concept of BCH codes. They have n columns and $m = \ell \log n(1 + o(1))$ rows. For $\ell \ge 3$, the best currently known resolvable matrices are all inexplicit and constructed by random coding as shown in [23, 9].

Lemma 3. *For any given integer ℓ, there exists an $(\ell, 1, 0)$-resolvable matrix of size $m \times n$ with*

$$m = \frac{2\ell - 1}{\log \binom{2^{2\ell}}{\binom{2\ell}{\ell}}} \log n(1 + o(1)).$$

Furthermore, $m = 4\ell \log n / \log \ell(1 + o(1))$ as $\ell \to \infty$.

Remark 2. *The presented existential result is order-optimal. By [9, 7], the number of rows in an $(\ell, 1, 0)$-robust-resolvable matrix with n columns has to be $m \ge 2\ell \log n / \log \ell(1 + o(1))$.*

4.4 Constructing robust resolvable matrices via robust cover-free matrices

It is challenging to find a sufficient (and necessary) condition for a matrix to be (ℓ, k, α)-robust-resolvable. In the following statement, we make one step toward understanding this problem.

Lemma 4. *Suppose $H_1 = (V, E_1)$ and $H_2 = (V, E_2)$ are two different (ℓ, k)-sparse hypergraphs. Then there exists a query $Q \subseteq V$ of size at most $\min(\ell, k)$ that separates the hypergraphs, i.e., $z(Q, H_1) \ne z(Q, H_2)$.*

Proof. Since two hypergraphs are distinct, there exists at least one hyperedge such that one edge set (E_1 or E_2) contains it with a larger multiplicity than the other one (E_2 or E_1). We take the largest hyperedge with this property. Without loss of generality, let m_1 and m_2 be multiplicities such that e is included, respectively, to E_1 with multiplicity m_1 and to E_2 with multiplicity m_2, $\ell \ge m_1 > m_2 \ge 0$. For a vertex v, define the set of hyperedges in a hypergraph $H = (V, E)$ that are incident to this vertex

$$A(v, H) := \{e' \in E : \ v \in h\}.$$

Consider the set of hyperedges $\bigcap_{v\in e} A(v, H_1)$. It includes m_1 copies of e and possibly some other hyperedges $E' \subseteq E_1$. We observe that the set of hyperedges $\bigcap_{v\in e} A(v, H_2)$ includes m_2 copies of e and precisely the same set of hyperedges E'. This happens due to the choice of e. Thus, we obtain

$$m_1 + |E'| = \left|\bigcap_{v\in e} A(v, H_1)\right| > \left|\bigcap_{v\in e} A(v, H_2)\right| = m_2 + |E'|. \tag{5}$$

Toward a contradiction, assume that there is no $Q \subseteq e$ such that

$$z(Q, H_1) = \left|\bigcup_{v\in Q} A(v, H_1)\right| \neq \left|\bigcup_{v\in Q} A(v, H_2)\right| = z(Q, H_2).$$

This means that the above unions for any $Q \subseteq e$ must have the same size. By the inclusion-exclusion principle

$$\left|\bigcup_{v\in Q} A(v, H_1)\right| = \sum_{\emptyset\neq Q'\subseteq Q} (-1)^{|Q'|+1} \left|\bigcap_{v\in Q} A(v, H_1)\right|.$$

Therefore, by induction on the size of Q, one can derive

$$\left|\bigcap_{v\in Q} A(v, H_1)\right| = \left|\bigcap_{v\in Q} A(v, H_2)\right|$$

for all $Q \subseteq e$. This contradicts to (5) when $Q = e$.

The above arguments show that a proper query Q could have size at most ℓ. Assume that $z(Q, H_1) > z(Q, H_2)$. Note that $z(Q, H_1) \leq k$ since H_1 contains k edges. Thus, one can remove at least $\max(0, |Q| - k)$ vertices from Q to get another subset of vertices Q' such that the output result for H_1 is not changed, i.e., $z(Q', H_1) = z(Q, H_1)$. On the other hand, it is clear that $z(Q', H_2) \leq z(Q, H_2)$. This completes the proof. □

Lemma 4 suggests a way for constructing (ℓ, k, α)-robust-resolvable matrices. We now introduce a concept that generalizes the definition of cover-free codes. This family of codes was originally suggested in [21] in connection with cryptographic applications.

Definition 2. A binary matrix X of size $m \times n$ is said to be (c, f, α)-robust-cover-free if for any disjoint subsets of columns $U, W \subseteq [n]$ of size $|U| = c$ and $|W| = f$, there exists at least $2\alpha m + 1$ rows $I \subseteq [m]$ such that

$$X_{i,j} = 1 \quad \text{for all } i \in [I], j \in U, \qquad X_{i,j} = 0 \quad \text{for all } i \in [I], j \in W.$$

If $\alpha = 0$, then a (c, f, α)-robust-cover-free matrix is simply called (c, f)-cover-free.

An $(1, f, \alpha)$-robust-cover-free matrix is also called robust union-free. The latter has several applications in 1-bit compressed sensing [1, 13]. Now we state a straightforward corollary of Lemma 4 that enables us to construct robust resolvable matrices.

Lemma 5. *Suppose a binary matrix X of size $m \times n$ is $(\min(\ell, k), 2k\ell - \min(\ell, k), \alpha)$-robust-cover-free. Then X is (ℓ, k, α)-robust-resolvable.*

Proof. Let $H_1 = (V, E_1)$ and $H_2 = (V, E_2)$ be two distinct (ℓ, k)-sparse hypergraphs. By Lemma 4, there exists a query $Q \subseteq V$, $|Q| \leq \min(\ell, k)$, such that $z(Q, H_1) \neq z(Q, H_2)$. Define W to be $\{v \in e : e \in E_1 \cup E_2\}$. Since we associate V with $[n]$, we get $Q \subseteq [n]$, $|Q| \leq \min(\ell, k)$, and $W \subseteq [n]$, $|W| \leq 2\ell k - \min(\ell, k)$. Combining these arguments with the fact that X is $(\min(\ell, k), 2k\ell - \min(\ell, k), \alpha)$-robust-cover-free yields that X is (ℓ, k, α)-robust-resolvable. □

Fundamental limits of cover-free matrices have been a subject of extensive studies in many papers [10, 25]. However, we cannot directly apply known bounds since they correspond to the case $\alpha = 0$. Now we present an achievability bound for robust cover-free matrices.

Lemma 6. *Let $c \geq 1$ and $f \geq 1$ be integers, α be a real number from the interval $[0, \alpha_0)$, where $\alpha_0 := \alpha_0(\alpha, c, f)$ is the least positive root of the function*

$$g(\alpha, c, f) := (2\alpha - 1)\log(1 - \overline{p}) - 2\alpha \log \overline{p} - H_2(2\alpha),$$
$$\overline{p} := \left(\frac{c}{c+f}\right)^c \left(\frac{f}{c+f}\right)^f.$$

Then there exists an (c, f, α)-robust-cover-free matrix of size $m \times n$ with

$$m \leq \frac{(c + f - 1)\log n}{g(\alpha, c, f)}(1 + o(1)).$$

Now we are in a good position to present an existential result for robust resolvable matrices.

Lemma 7. *Let $\ell \geq 1$ and $k \geq 1$ be integers, α be a real number from the interval $[0, \min(\alpha_0, 1/4))$, where $\alpha_0 := \alpha_0(\alpha, \ell, k)$ is the least positive root of the function $f(\alpha, \ell, k)$ defined in* (3)*. Then there exists an (ℓ, k, α)-robust-resolvable matrix of size $m \times n$ with $m \leq \frac{(2k\ell - 1)\log n}{f(\alpha, \ell, k)}$. Furthermore, for $\ell \to \infty$ and fixed k, it holds that*

$$m(n, \ell, k, \alpha) \leq \frac{2k\ell \log n}{(1 - 2\alpha)(2\ell e)^{-k} \log e + 2\alpha(k \log(2\ell) + k \log e) - H_2(2\alpha)}(1 + o(1)).$$

For $\alpha = 0$, we have $m(n, \ell, k, 0) = \mathcal{O}(k\ell \max(2e\ell, 2ek)^{\min(\ell,k)} \log n)$.

Proof. The statement is implied by Lemma 5 and Lemma 6 with $c = \min(\ell, k)$, $f = 2\ell k - \min(\ell, k)$. □

4.5 Proof of Theorem 1

Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors. Let $H_{\mathcal{B}} = (V, E_{\mathcal{B}})$ be an (ℓ, k)-sparse hypergraph induced by $\mathcal{B}$, i.e., the set of edges $E_{\mathcal{B}}$ consists of the supports of vectors from $\mathcal{B}$. Set $\alpha = 16\tau\ell$. By Lemma 7, there exists an (ℓ, k, α)-robust-resolvable matrix X of size $m \times n$ with $m = \frac{(2k\ell - 1)\log n}{f(\alpha, \ell, k)}(1 + o(1))$. By Definition 1, $H_{\mathcal{B}}$ can be uniquely reconstructed based on any noisy version $\tilde{\boldsymbol{z}} \in \mathbb{Z}^m$ of the vector $\boldsymbol{z}(X, H)$ so that $d_H(\tilde{\boldsymbol{z}}, \boldsymbol{z}(X, H)) < \alpha m$. Applying Lemma 2, with overwhelming probability one can learn some $\tilde{\boldsymbol{z}}$ by asking $q = \mathcal{O}(\ell^2 m \log(nm))$ queries $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(q)}$ and obtaining responses $y_1, \ldots, y_q$ provided that (1)-(2). This implies that q queries are sufficient for solving Problem 2.

5 Approximate recovery

Throughout this section, we assume that there is no noise in measurements, i.e., $\tau = 0$. Theorem 1 enables us to assume that the supports of all sparse vectors are already known. Thus, it remains to approximately reconstruct the values of the vectors.

A query vector $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ is called *Gaussian* if each entry is sampled independently from a Gaussian distribution $\mathcal{N}(0, 1)$. We shall use the following result which says that Gaussian queries allow to efficiently reconstruct any sparse vector.

Lemma 8 (Approximate vector recovery, [16])**.** *Let $\boldsymbol{\beta}$ be an unknown vector in $\mathbb{R}^k$. For any $\epsilon > 0$, there exists an algorithm that takes responses to Gaussian queries $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(q)} \in \mathbb{R}^k$ with $q = \mathcal{O}(\frac{k}{\epsilon} \log \frac{k}{\epsilon})$ as input and outputs $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^k$ so that*

$$\left\| \frac{\boldsymbol{\beta}}{\|\boldsymbol{\beta}\|} - \tilde{\boldsymbol{\beta}} \right\| \leq \epsilon.$$

Recall the response to a query in our model is taken uniformly at random from a set of ℓ possible responses. Therefore, we need to show how to model a response to a Gaussian query for any given vector from the family. Here, we will use our structural assumption regarding the support of vectors (c.f. Assumption 1). In Sec. 5.1, we show how to emulate individual Gaussian queries in our model. In Sec. 5.2 we present a two-stage approximate recovery algorithm.

5.1 Modeling Gaussian query

Define Inf to be a very large positive real number whose lower bound will be clear from the context. For a query vector $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ and an integer $i \in [\ell]$, we write $\mathrm{Inf}^{(i)}(\boldsymbol{x})$ to denote the vector whose j-th entry is

$$\mathrm{Inf}^{(i)}(\boldsymbol{x})_j = \begin{cases} x_j, & j \in \mathrm{supp}(\boldsymbol{\beta}^{(i)}) \text{ or } j \notin \cup_{t \neq i} \mathrm{supp}(\boldsymbol{\beta}^{(t)}), \\ \mathrm{Inf}, & \text{otherwise.} \end{cases}$$

Let $\mathbf{0}$ denote an all-zero vector whose length will be clear from the context and $\boldsymbol{g} \in \mathbb{R}^n$ be a Gaussian query. Fix an integer $i \in [\ell]$ and some positive integer m. Set $\boldsymbol{x}^{(1,i)} = \ldots = \boldsymbol{x}^{(m/2,i)} = \mathrm{Inf}^{(i)}(\boldsymbol{g})$ and $\boldsymbol{x}^{(m/2+1,i)} = \ldots = \boldsymbol{x}^{(m,i)} = \mathrm{Inf}^{(i)}(\mathbf{0})$. Let $y_1, \ldots, y_m$ be responses to queries $\boldsymbol{x}^{(1,i)}, \ldots, \boldsymbol{x}^{(m,i)}$, i.e., $y_j = \mathrm{sign}(\langle \boldsymbol{x}^{(j,i)}, \boldsymbol{\beta} \rangle)$ with $\boldsymbol{\beta}$ being sampled uniformly at random from $\mathcal{B}$. Define a sample statistic that estimates the value $\mathrm{sign}(\langle \boldsymbol{g}, \boldsymbol{\beta}^{(i)} \rangle)$

$$\widetilde{\mathrm{sign}}(\boldsymbol{g}, \boldsymbol{\beta}^{(i)}, m) := \mathrm{round}\left(\frac{2\ell}{m} \left(\sum_{j=1}^{m/2} y_j - \sum_{j=m/2+1}^{m} y_j \right) + 1 \right)$$

Lemma 9 (Gaussian query estimator)**.** *Let $\mathcal{B}$ be a family of sparse vectors that satisfies Assumption 1. Then it holds that*

$$\Pr\left\{ \widetilde{\mathrm{sign}}(\boldsymbol{g}, \boldsymbol{\beta}^{(i)}, m) = \mathrm{sign}(\langle \boldsymbol{g}, \boldsymbol{\beta}^{(i)} \rangle) \right\} = 1 - \exp(-\Omega(\ell^{-2} m)).$$

5.2 Proof of Theorem 2

Let $\mathcal{B} = \{\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(\ell)}\}$ be a family of ℓ unknown k-sparse vectors that satisfy Assumption 1. Fix the Euclidean precision $\epsilon > 0$. First, we invoke Theorem 1 and with probability $1 - \mathcal{O}(1/n)$, the supports of all vectors in $\mathcal{B}$ can be reconstructed using $\mathcal{O}(k\ell^3 \max(2e\ell, 2ek)^{\min(\ell,k)} \log^2 n)$ queries. Note that the number of coordinates in the supports of the vectors is at most $k\ell$.

At the second stage, we first generate Gaussian vectors $g^{(1)}, \ldots, g^{(R)} \in \mathbb{R}^{k\ell}$ with $R = \mathcal{O}(\frac{k}{\epsilon} \log \frac{k}{\epsilon})$ being as in Lemma 8. Applying Lemma 9, with probability $1 - \mathcal{O}((n\ell \frac{k}{\epsilon} \log \frac{k}{\epsilon})^{-1})$ we can learn the value of $\mathrm{sign}(\langle \boldsymbol{g}^{(j)}, \boldsymbol{\beta}^{(i)} \rangle)$ for some $j \in [R]$ and $i \in [\ell]$, by making $\mathcal{O}(\ell^2 \log(nk\ell/\epsilon))$ queries. By the union bound, all values $\{\mathrm{sign}(\langle \boldsymbol{g}^{(j)}, \boldsymbol{\beta}^{(i)} \rangle) : j \in [R], i \in [\ell]\}$ are reconstructed with probability $1 - \mathcal{O}(1/n)$ by making $\mathcal{O}(\ell^3 \frac{k}{\epsilon} \log \frac{k}{\epsilon} \log(nk\ell/\epsilon))$ measurements. Finally, Lemma 8 guarantees that all sparse vectors from $\mathcal{B}$ can be recovered with the given precision ϵ based on the set of values $\{\mathrm{sign}(\langle \boldsymbol{g}^{(j)}, \boldsymbol{\beta}^{(i)} \rangle) : j \in [R], i \in [\ell]\}$. Since $k\ell = \mathcal{O}(n)$, the required claim follows.

6 Conclusion

In this paper, we studied two problems that arise in recovering unknown sparse vectors from a mixture of sign responses. For the support recovery problem, we found a completely non-adaptive solution that does not use any assumptions. This problem was posed in [12]. In addition, the proposed algorithm is resilient to noisy measurements. When describing our result, we introduced the notion of a robust resolvable matrix and derived an existential result for such matrices. We leave the problem of deriving fundamental limits of this combinatorial object as an open challenging problem that can be of independent interest in computer science, coding theory, and combinatorics. For the approximate recovery problem, we suggested single-stage and two-stage reconstruction algorithms working under Assumption 1. Finding an algorithm for robust approximate recovery of a family of unknown sparse vectors without any assumptions remains a major open problem.

References

[1] Jayadev Acharya, Arnab Bhattacharyya, and Pritish Kamath. Improved bounds for universal one-bit compressive sensing. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 2353–2357. IEEE, 2017.

[2] Albert Ai, Alex Lapanowski, Yaniv Plan, and Roman Vershynin. One-bit compressed sensing with non-gaussian measurements. *Linear Algebra and its Applications*, 441:222–239, 2014.

[3] Raj C Bose and Sarvadaman Chowla. Theorems in the additive theory of numbers. *Commentarii Mathematici Helvetici*, 37(1):141–147, 1962.

[4] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

[5] Arun Tejasvi Chaganty and Percy Liang. Spectral experts for estimating mixtures of linear regressions. In *International Conference on Machine Learning*, pages 1040–1048, 2013.

[6] Yudong Chen, Xinyang Yi, and Constantine Caramanis. A convex formulation for mixed regression with two components: Minimax optimal rates. In *Conference on Learning Theory*, pages 560–604, 2014.

[7] Arkady Djackov. On a search model of false coins. In *Topics in Information Theory (Colloquia Mathematica Societatis Janos Bolyai 16). Budapest, Hungary: Hungarian Acad. Sci*, pages 163–170, 1975.

[8] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[9] Arkadii D'yachkov and Vladimir Rykov. On a coding model for a multiple-access adder channel. *Problemy Peredachi Informatsii*, 17(2):26–38, 1981.

[10] Arkady D'yachkov, Pavel Vilenkin, David Torney, and Anthony Macula. Families of finite sets in which no intersection of l sets is covered by the union of s others. *Journal of Combinatorial Theory, Series A*, 99(2):195–218, 2002.

[11] Larkin Flodin, Venkata Gandikota, and Arya Mazumdar. Superset technique for approximate recovery in one-bit compressed sensing. In *Advances in Neural Information Processing Systems*, pages 10387–10396, 2019.

[12] Venkata Gandikota, Arya Mazumdar, and Soumyabrata Pal. Recovery of sparse linear classifiers from mixture of responses. In *Advances in Neural Information Processing Systems*, 2020.

[13] Sivakant Gopi, Praneeth Netrapalli, Prateek Jain, and Aditya Nori. One-bit compressed sensing: Provable support and vector recovery. In *International Conference on Machine Learning*, pages 154–162, 2013.

[14] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding*, pages 409–426. Springer, 1994.

[15] Mian Huang, Runze Li, and Shaoli Wang. Nonparametric mixture of regression models. *Journal of the American Statistical Association*, 108(503):929–941, 2013.

[16] Laurent Jacques, Jason N Laska, Petros T Boufounos, and Richard G Baraniuk. Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. *IEEE Transactions on Information Theory*, 59(4):2082–2102, 2013.

[17] Abbas Khalili and Jiahua Chen. Variable selection in finite mixture of regression models. *Journal of the american Statistical association*, 102(479):1025–1038, 2007.

[18] Akshay Krishnamurthy, Arya Mazumdar, Andrew McGregor, and Soumyabrata Pal. Sample complexity of learning mixtures of sparse linear regressions. In *Advances in Neural Information Processing Systems*, pages 10532–10541, 2019.

[19] Bernt Lindström. Determination of two vectors from the sum. *Journal of Combinatorial Theory*, 6(4):402–407, 1969.

[20] Arya Mazumdar and Soumyabrata Pal. Recovery of sparse signals from a mixture of linear samples. In *International Conference on Machine Learning*, pages 7466–7475, 2020.

[21] Chris J Mitchell and Fred C Piper. Key storage in secure networks. *Discrete applied mathematics*, 21(3):215–228, 1988.

[22] Yaniv Plan and Roman Vershynin. One-bit compressed sensing by linear programming. *Communications on Pure and Applied Mathematics*, 66(8):1275–1297, 2013.

[23] Grigory Poltyrev. Improved upper bound on the probability of decoding error for codes of complex structure. *Problemy Peredachi Informatsii*, 23(4):5–18, 1987.

[24] Weixing Song, Weixin Yao, and Yanru Xing. Robust mixture regression model fitting by laplace distribution. *Computational Statistics & Data Analysis*, 71:128–137, 2014.

[25] Douglas R Stinson, Ruizhong Wei, and Lie Zhu. Some new bounds for cover-free families. *J. Comb. Theory, Ser. A*, 90(1):224–234, 2000.

[26] Yuekai Sun, Stratis Ioannidis, and Andrea Montanari. Learning mixtures of linear classifiers. In *International Conference on Machine Learning*, pages 721–729, 2014.

[27] Taiyao Wang and Ioannis Ch Paschalidis. Convergence of parameter estimates for regularized mixed linear regression models. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3664–3669. IEEE, 2019.

[28] Dong Yin, Ramtin Pedarsani, Yudong Chen, and Kannan Ramchandran. Learning mixtures of sparse linear regressions using sparse graph codes. *IEEE Transactions on Information Theory*, 65(3):1430–1451, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Please see the last paragraph of Sec. 1.1, Sec. 6, Remark 1.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We don't see any potential negative societal impacts
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Please see Sec. 3.
   (b) Did you include complete proofs of all theoretical results? [Yes] Please check Appendix (as a part of supplementary materials) for the proofs missing in the conference version of the paper.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] Not relevant
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [No] Not relevant
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Not relevant
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] Not relevant
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [No] Not relevant
   (b) Did you mention the license of the assets? [No] Not relevant
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] Not relevant

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] Not relevant

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] Not relevant

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [No] Not applicable

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [No] Not applicable

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [No] Not relevant