Recovery of Sparse Signals from a Mixture of Linear Samples

Arya Mazumdar [1] **Soumyabrata Pal** [1]

Abstract

Mixture of linear regressions is a popular learning theoretic model that is used widely to represent heterogeneous data. In the simplest form, this model assumes that the labels are generated from either of two different linear models and mixed together. Recent works of Yin et al. and Krishnamurthy et al., 2019, focus on an experimental design setting of model recovery for this problem. It is assumed that the features can be designed and queried with to obtain their label. When queried, an oracle randomly selects one of the two different sparse linear models and generates a label accordingly. How many such oracle queries are needed to recover both of the models simultaneously? This question can also be thought of as a generalization of the well-known compressed sensing problem (Candès and Tao, 2005, Donoho, 2006). In this work we address this query complexity problem and provide efficient algorithms that improves on the previously best known results.

1. Introduction

Suppose, there are two unknown distinct vectors $\beta^1, \beta^2 \in \mathbb{R}^n$, that we want to recover. We can measure these vectors by taking linear samples, however the linear samples come without the identifier of the vectors. To make this statement rigorous, assume the presence of an oracle $\mathcal{O} : \mathbb{R}^n \to \mathbb{R}$ which, when queried with a vector $\mathbf{x} \in \mathbb{R}^n$, returns the noisy output $y \in \mathbb{R}$:

$$y = \langle \mathbf{x}, \beta \rangle + \zeta \tag{1}$$

where β is chosen uniformly from $\{\beta^1, \beta^2\}$ and ζ is additive Gaussian noise with zero mean and known variance $\sigma^2 > 0$.

[1]Computer Science Department at the University of Massachusetts Amherst, Amherst, MA01003, USA. Correspondence to: Arya Mazumdar <arya@cs.umass.edu>, Soumyabrata Pal <soumyabratap@umass.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

We will refer to the values returned by the oracle given these queries as *samples*.

For a $\beta \in \mathbb{R}^n$, the best k-sparse approximation $\beta_{(k)}$ is defined to be the vector obtained from β where all except the k-largest (by absolute value) coordinates are set to 0. For each $\beta \in \{\beta^1, \beta^2\}$, our objective in this setting is to return a *sparse approximation* of $\hat{\beta}$ using minimum number of queries such that

$$||\hat{\beta} - \beta|| \leq c||\beta - \beta_{(k)}|| + \gamma$$

where c is an absolute constant, γ is a user defined nonnegative parameter representing the precision up to which we want to recover the unknown vectors, and the norms are arbitrary. For any algorithm that performs this task, the total number of samples acquired from the oracle is referred to as the *query complexity*.

If we had one, instead of two unknown vectors, then the problem would exactly be that of *compressed sensing* (Candès et al., 2006; Donoho, 2006). However having two vectors makes this problem significantly different and challenging. Further, if we allow $\gamma = \Omega(\|\beta^1 - \beta^2\|)$, then we can treat all the samples to be coming from the same vector and output only a single vector as an approximation to both vectors. So in practice, obtaining $\gamma = o(\|\beta^1 - \beta^2\|)$ is more interesting.

On another technical note, under this setting it is always possible to make the noise ζ negligible by increasing the norm of the query $\mathbf{x}$. To make the problem well-posed, let us define the Signal-to-Noise Ratio (SNR) for a query $\mathbf{x}$:

$$\mathsf{SNR}(\mathbf{x}) \triangleq \frac{\mathbb{E}|\langle \mathbf{x}, \beta^1 - \beta^2 \rangle|^2}{\mathbb{E}\zeta^2}$$

where the expectation is over the randomness of the query. Furthermore define the overall SNR to be $\mathsf{SNR} := \max_{\mathbf{x}} \mathsf{SNR}(\mathbf{x})$, where the maximization is over all the queries used in the recovery process.

1.1. Most Relevant Works

Previous works that are most relevant to our problem are by Yin et al. (Yin et al., 2019) and Krishnamurthy et al. (Krishnamurthy et al., 2019). Both of these papers address the exact same problem as above; but provide results under

some restrictive conditions on the unknown vectors. For example, the results of (Yin et al., 2019) is valid only when,

- the unknown vectors are exactly k-sparse, i.e., has at most k nonzero entries;
- it must hold that,

  $$\beta^1_j \neq \beta^2_j \quad \text{for each} \quad j \in \text{supp}\big(\beta^1\big) \cap \text{supp}\big(\beta^2\big) \ ,$$

  where β_j denotes the jth coordinate of β, and $\text{supp}(\beta)$ denotes the set of nonzero coordinates of β;
- for some $\epsilon > 0$, $\beta^1, \beta^2 \in \{0, \pm\epsilon, \pm 2\epsilon, \pm 3\epsilon, \ldots\}^n$.

All of these assumptions, especially the later two, are severely restrictive. While the results of (Krishnamurthy et al., 2019) are valid without the first two assumptions, they fail to get rid of the third, an assumption of the unknown vectors always taking discrete values. This is in particular unfavorable, because the resultant query/sample complexities (and hence the time complexity) in both the above papers has an exponential dependence on $\frac{1}{\epsilon}$.

1.2. Our Main Result

In contrast to these earlier results, we provide a generic sample complexity result that does not require any of the assumptions used by the predecessor works. Our main result is following.

Theorem 1. *[Main Result] Let* $\mathsf{NF} := \frac{\gamma}{\sigma}$ *(the* noise factor*) where* $\gamma > 0$ *is a parameter representing the desired recovery precision and* $\sigma > 0$ *is the standard deviation of* ζ *in Eq. (1).*

Case 1. For any $\gamma < \left|\left|\beta^1 - \beta^2\right|\right|_2 /2$*, there exists an algorithm that makes*

$$O\left(k \log n \log k \left\lceil \frac{\log k}{\log(\sqrt{\mathsf{SNR}}/\mathsf{NF})} \right\rceil \left\lceil \frac{1}{\mathsf{NF}^4\sqrt{\mathsf{SNR}}} + \frac{1}{\mathsf{NF}^2} \right\rceil \right)$$

queries to recover $\hat{\beta}^1, \hat{\beta}^2$*, estimates of* β^1, β^2*, with high probability such that , for* $i = 1, 2$*,*

$$||\hat{\beta}^i - \beta^{\pi(i)}||_2 \leq \frac{c||\beta^i - \beta^i_{(k)}||_1}{\sqrt{k}} + O(\gamma)$$

where $\pi : \{1, 2\} \to \{1, 2\}$ *is some permutation of* $\{1, 2\}$ *and* c *is a universal constant.*

Case 2. For any $\gamma = \Omega(\left|\left|\beta^1 - \beta^2\right|\right|_2)$*, there exists an algorithm that makes* $O\left(k \log n \left\lceil \frac{\log k}{\mathsf{SNR}} \right\rceil\right)$ *queries to recover* $\hat{\beta}$*, estimates of both* β^1, β^2*, with high probability such that , for both* $i = 1, 2$*,*

$$||\hat{\beta} - \beta^i||_2 \leq \frac{c||\beta^i - \beta^i_{(k)}||_1}{\sqrt{k}} + O(\gamma)$$

where c *is a universal constant.*

For a $\gamma = \Theta(\left|\left|\beta^1 - \beta^2\right|\right|_2)$ the first case of the Theorem holds but using the second case may give better result in that regime of precision. The second case of the theorem shows that if we allow a rather large precision error, then the number of queries is similar to the required number for recovering a single vector. This is expected, because in this case we can find just one line approximating both regressions.

The recovery guarantee that we are providing (an ℓ_2-ℓ_1 guarantee) is in line with the standard guarantees of the compressed sensing literature. In this paper, we are interested in the regime $\log n \leq k \ll n$ as in compressed sensing. Note that, our number of required samples scales linearly with k and has only poly-logarithmic scaling with n, and polynomial scaling with the noise σ. In the previous works (Yin et al., 2019), (Krishnamurthy et al., 2019), the complexities scaled exponentially with noise.

Furthermore, the query complexity of our algorithm decreases with the Euclidean distance between the vectors (or the 'gap') - which makes sense intuitively. Consider the case when when we want a precise recovery (γ very small). It turns out that when the gap is large, the query complexity varies as $O((\log \text{gap})^{-1})$ and when the gap is small the query complexity scale as $O((\text{gap} \log \text{gap})^{-1})$.

Remark 1 (The zero noise case)**.** *When* $\sigma = 0$*, i.e., the samples are not noisy, the problem is still nontrivial, and is not covered by the statement of Theorem 1. However this case is strictly simpler to handle as it will involve only the alignment step (as will be discussed later), and not the mixture learning step. Recovery with* $\gamma = 0$ *is possible with only* $O(k \log n \log k)$ *queries (see Appendix F for a more detailed discussion on the noiseless setting).*

1.3. Other Relevant Works

The problem we address can be seen as the active learning version of learning mixtures of linear regressions. Mixture of linear regressions is a natural synthesis of mixture models and linear regression; a generalization of the basic linear regression problem of learning the best linear relationship between the labels and the feature vectors. In this generalization, each label is stochastically generated by picking a linear relation uniformly from a set of two or more linear functions, evaluating this function on the features and possibly adding noise; the goal is to learn the set of unknown linear functions. The problem has been studied at least for past three decades, staring with De Veaux (De Veaux, 1989) with a recent surge of interest (Chaganty & Liang, 2013; Faria & Soromenho, 2010; Städler et al., 2010; Kwon & Caramanis, 2018; Viele & Tong, 2002; Yi et al., 2014; 2016). In this literature a variety of algorithmic techniques to obtain polynomial sample complexity were proposed. To the best of our knowledge, Städler et al. (Städler et al., 2010)

were the first to impose sparsity on the solutions, where each linear function depends on only a small number of variables. However, many of the earlier papers on mixtures of linear regression, essentially consider the features to be fixed, i.e., part of the input, whereas recent works focus on the query-based model in the sparse setting, where features can be designed as queries (Yin et al., 2019; Krishnamurthy et al., 2019). The problem has numerous applications in modelling heterogeneous data arising in medical applications, behavioral health, and music perception (Yin et al., 2019).

This problem is a generalization of the compressed sensing problem (Candès et al., 2006; Donoho, 2006). As a building block to our solution, we use results from exact parameter learning for Gaussian mixtures. Both compressed sensing and learning mixtures of distributions (Dasgupta, 1999; Titterington et al., 1985) are immensely popular topics across statistics, signal processing and machine learning with a large body of prior work. We refer to an excellent survey by (Boche et al., 2015) for compressed sensing results (in particular the results of (Candes et al., 2008) and (Baraniuk et al., 2008) are useful). For parameter learning in mixture models, we find the results of (Daskalakis et al., 2017; Daskalakis & Kamath, 2014; Hardt & Price, 2015; Xu et al., 2016; Balakrishnan et al., 2017; Krishnamurthy et al., 2020) to be directly relevant.

1.4. Technical Contributions

If the responses to the queries were to contain tags of the models they are coming from, then we could use rows of any standard compressed sensing matrix as queries and just segregate the responses using the tags. Then by running a compressed sensing recovery on the groups with same tags, we would be done. In what follows, we try to infer this 'tag' information by making redundant queries.

If we repeat just the same query multiple time, the noisy responses are going to come from a mixture of Gaussians, with the actual responses being the component means. To learn the actual responses we rely on methods for parameter learning in Gaussian mixtures. It turns out that for different parameter regimes, different methods are best-suited for our purpose - and it is not known in advance what regime we would be in. The method of moments is a well-known procedure for parameter learning in Gaussian mixtures and rigorous theoretical guarantees on sample complexity exist (Hardt & Price, 2015). However we are in a specialized regime of scalar uniform mixtures with known variance; and we leverage these information to get better sample complexity guarantee for exact parameter learning (Theorem 3). In particular we show that, in this case the mean and variance of the mixture are sufficient statistics to recover the unknown means, as opposed to the first six moments of the general case (Hardt & Price, 2015). While recovery using other methods (Algorithms 1 and 4) are straight forward adaption of known literature, we show that only a small set of samples are needed to determine what method to use.

It turns out that method of moments still needs significantly more samples than the other methods. However we can avoid using method of moments and use a less intensive method (such as EM, Algorithms 1), provided we are in a regime when the gap between the component means is high. The only fact is that the Euclidean distance between β^1 and β^2 are far does not guarantee that. However, if we choose the queries to be Gaussians, then the gap is indeed high with certain probability. If the queries were to be generated by any other distribution, then such fact will require strong anti-concentration inequalities that in general do not hold. Therefore, we cannot really work with any standard compressed sensing matrix, but have to choose Gaussian matrices (which are incidentally also good standard compressed sensing matrices).

The main technical challenge comes in the next step, alignment. For any two queries $\mathbf{x}, \mathbf{x}'$, even if we know $y_1 = \langle \beta^1, \mathbf{x} \rangle, y_2 = \langle \beta^2, \mathbf{x} \rangle$ and $y_1' = \langle \beta^1, \mathbf{x}' \rangle, y_2' = \langle \beta^2, \mathbf{x}' \rangle$, we do not know how to club y_1 and y_1' together as their order could be different. And this is an issue with all pairs of queries which leaves us with exponential number of possibilities to choose form. We form a simple error-correcting code to tackle this problem.

For two queries, $\mathbf{x}, \mathbf{x}'$, we deal with this issue by designing two additional queries $\mathbf{x} + \mathbf{x}'$and $\mathbf{x} - \mathbf{x}'$. Now even if we mis-align, we can cross-verify with the samples from 'sum' query and the 'difference' query, and at least one of these will show inconsistency. We subsequently extend this idea to align all the samples. Once the samples are all aligned, we can just use some any recovery algorithm for compressed sensing to deduce the sparse vectors.

The rest of this paper is organized as follows. We give an overview of our algorithm in Sec. 2.1 , the actual algorithm is presented in Algorithm 8, which calls several subroutines. The process of denoising by Gaussian mixture learning is described in Sec. 2.2. The alignment problem is discussed in Sec. 2.3 and the proof of Theorem 1 is wrapped up in Sec. 2.4. Most proofs are delegated to the appendix in the supplementary material. Some 'proof of concept' simulation results are also in the appendix.

2. Main Results

2.1. Overview of Our Algorithm

Our scheme to recover the unknown vectors is described below. We will carefully chose the numbers m, m' so that the overall query complexity meets the promise of Theorem 1.

- We pick m query vectors $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^m$ independently,

each according to $\mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ where $\mathbf{0}$ is the n-dimensional all zero vector and $\mathbf{I}_n$ is the $n \times n$ identity matrix.

- (Mixture) We repeatedly query the oracle with $\mathbf{x}^i$ for T_i times for all $i \in [m]$ in order to offset the noise. The samples obtained from the repeated querying of $\mathbf{x}^i$ is referred to as a *batch* corresponding to $\mathbf{x}^i$. T_i is referred to as the *batchsize* of $\mathbf{x}^i$. Our objective is to return $\hat{\mu}_{i,1}$ and $\hat{\mu}_{i,2}$, estimates of $\langle \mathbf{x}^i, \beta^1 \rangle$ and $\langle \mathbf{x}^i, \beta^2 \rangle$ respectively from the batch of samples (details in Section 2.2). However, it will not be possible to label which estimated mean corresponds to β^1 and which one corresponds to β^2.
- (Alignment) For some $m' < m$ and for each $i \in [m], j \in [m']$ such that $i \neq j$, we also query the oracle with the vectors $\mathbf{x}^i + \mathbf{x}^j$ (*sum query*) and $\mathbf{x}^i - \mathbf{x}^j$ (*difference query*) repeatedly for $T^{\mathsf{sum}}_{i,j}$ and $T^{\mathsf{diff}}_{i,j}$ times respectively. Our objective is to cluster the set of estimated means $\{\hat{\mu}_{i,1}, \hat{\mu}_{i,2}\}_{i=1}^m$ into two equally sized clusters such that all the elements in a particular cluster are good estimates of querying the same unknown vector.
- Since the queries $\{\mathbf{x}^i\}_{i=1}^m$ has the property of being a good compressed sensing matrix (they satisfy δ-RIP condition, a sufficient condition for ℓ_2-ℓ_1 recovery in compressed sensing, with high probability), we can formulate a convex optimization problem using the estimates present in each cluster to recover the unknown vectors β^1 and β^2.

It is evident that the sample (query) complexity will be $\sum_{i=1}^m T_i + \sum_{\substack{i \in [m], j \in [m'] \\ i \neq j}} T^{\mathsf{sum}}_{i,j} + T^{\mathsf{diff}}_{i,j}$. In the subsections below, we will show each step more formally and provide upper bounds on the sufficient batchsize for each query.

2.2. Recovering Unknown Means from a Batch

For a query $\mathbf{x} \in \{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^m\}$, notice that the samples from the batch corresponding to $\mathbf{x}$ is distributed according to a Gaussian mixture $\mathcal{M}$,

$$\mathcal{M} \triangleq \frac{1}{2}\mathcal{N}(\langle \mathbf{x}, \beta^1 \rangle, \sigma^2) + \frac{1}{2}\mathcal{N}(\langle \mathbf{x}, \beta^2 \rangle, \sigma^2),$$

an equally weighted mixture of two Gaussian distributions having means $\langle \mathbf{x}, \beta^1 \rangle, \langle \mathbf{x}, \beta^2 \rangle$ with known variance σ^2. For brevity, let us denote $\langle \mathbf{x}, \beta^1 \rangle$ by μ_1 and $\langle \mathbf{x}, \beta^2 \rangle$ by μ_2 from here on in this sub-section. In essence, our objective is to find the sufficient batchsize of $\mathbf{x}$ so that it is possible to estimate $\langle \mathbf{x}, \beta^1 \rangle$ and $\langle \mathbf{x}, \beta^2 \rangle$ upto an additive error of γ. Below, we go over some methods providing theoretical guarantees on the sufficient sample complexity for approximating the means that will be suitable for different parameter regimes.

2.2.1. Recovery using EM algorithm

The Expectation Maximization (EM) algorithm is widely known, and used for the purpose of parameter learning of Gaussian mixtures, cf. (Balakrishnan et al., 2017) and (Xu et al., 2016). The EM algorithm tailored towards recovering the parameters of the mixture $\mathcal{M}$ is described in Algorithm 1. The following result can be derived from (Daskalakis et al., 2017) (with our terminology) that gives a sample complexity guarantee of using EM algorithm.

Algorithm 1 EM$(\mathbf{x}, \sigma, T)$ Estimate the means $\langle \mathbf{x}, \beta^1 \rangle$, $\langle \mathbf{x}, \beta^2 \rangle$ for a query $\mathbf{x}$ using EM algorithm

Require: An oracle $\mathcal{O}$ which when queried with a vector $\mathbf{x} \in \mathbb{R}^n$ returns $\langle \mathbf{x}, \beta \rangle + \mathcal{N}(0, \sigma^2)$ where β is sampled uniformly from $\{\beta^1, \beta^2\}$.

for $i = 1, 2, \ldots, T$ **do**
  Query the oracle $\mathcal{O}$ with $\mathbf{x}$ and obtain a response y^i.
end for
Set the function $w : \mathbb{R}^3 \to \mathbb{R}$ as $w(y, \mu_1, \mu_2) = e^{-(y-\mu_1)^2/2\sigma^2}\left(e^{-(y-\mu_1)^2/2\sigma^2} + e^{-(y-\mu_2)^2/2\sigma^2}\right)^{-1}$.
Initialize $\hat{\mu}_1^0, \hat{\mu}_2^0$ randomly and $t = 0$.
while Until Convergence **do**
  $\hat{\mu}_1^{t+1} = \sum_{i=1}^T y_i w(y_i, \hat{\mu}_1^t, \hat{\mu}_2^t) / \sum_{i=1}^T w(y_i, \hat{\mu}_1^t, \hat{\mu}_2^t)$.
  $\hat{\mu}_2^{t+1} = \sum_{i=1}^T y_i w(y_i, \hat{\mu}_2^t, \hat{\mu}_1^t) / \sum_{i=1}^T w(y_i, \hat{\mu}_2^t, \hat{\mu}_1^t)$.
  $t \leftarrow t + 1$.
end while
Return $\hat{\mu}_1^t, \hat{\mu}_2^t$

Theorem 2 (Finite sample EM analysis (Daskalakis et al., 2017))**.** *From an equally weighted two component Gaussian mixture with unknown component means μ_1, μ_2 and known and shared variance σ^2, a total $O\Big(\Big\lceil \sigma^6/(\epsilon^2(\mu_1 - \mu_2)^4) \log 1/\eta \Big\rceil\Big)$ samples suffice to return $\hat{\mu}_1, \hat{\mu}_2$, such that for some permutation $\pi : \{1, 2\} \to \{1, 2\}$, for $i = 1, 2$,*

$$\left|\hat{\mu}_i - \mu_{\pi(i)}\right| \leq \epsilon$$

using the EM algorithm with probability at least $1 - \eta$.

This theorem implies that EM algorithm requires smaller number of samples as the separation between the means $|\mu_1 - \mu_2|$ grows larger. However, it is possible to have a better dependence on $|\mu_1 - \mu_2|$, especially when it is small compared to σ^2.

2.2.2. Method of Moments

Consider any Gaussian mixture with two components,

$$\mathcal{G} \triangleq p_1\mathcal{N}(\mu_1, \sigma_1^2) + p_2\mathcal{N}(\mu_2, \sigma_2^2),$$

where $0 < p_1, p_2 < 1$ and $p_1 + p_2 = 1$. Define the variance of a random variable distributed according to $\mathcal{G}$ to be

$$\sigma_{\mathcal{G}}^2 \triangleq p_1 p_2((\mu_1 - \mu_2)^2 + p_1\sigma_1^2 + p_2\sigma_2^2.$$

It was shown in (Hardt & Price, 2015) that $\Theta(\sigma_{\mathcal{G}}^{12}/\epsilon^{12})$ samples are both necessary and sufficient to recover the

unknown parameters $\mu_1, \mu_2, \sigma_1^2, \sigma_2^2$ upto an additive error of ϵ. However, in our setting the components of the mixture $\mathcal{M}$ have the same known variance σ^2 and further the mixture is equally weighted. Our first contribution is to show significantly better results for this special case.

Theorem 3. *With $O\left(\left\lceil \sigma_{\mathcal{M}}^2/\epsilon_1^2, \sigma_{\mathcal{M}}^4/\epsilon_2^2 \right\rceil \log 1/\eta\right)$ samples, Algorithm 3 returns $\hat{\mu}_1, \hat{\mu}_2$, such that for some permutation $\pi : \{1,2\} \to \{1,2\}$, we have, for $i = 1, 2$, $\left|\hat{\mu}_i - \mu_{\pi(i)}\right| \leq 2\epsilon_1 + 2\sqrt{\epsilon_2}$ with probability at least $1 - \eta$.*

This theorem states that $O(\sigma_{\mathcal{M}}^4)$ samples are sufficient to recover the unknown means of $\mathcal{M}$ (as compared to the $O(\sigma_{\mathcal{G}}^{12})$ result for the general case). This is because the mean and variance are sufficient statistics for this special case (as compared to first six excess moments in the general case). We first show two technical lemmas providing guarantees on recovering the mean and the variance of a random variable X distributed according to $\mathcal{M}$. The procedure to return $\hat{M}_1$ and $\hat{M}_2$ (estimates of $\mathbb{E}X$ and $\mathrm{var}X$ respectively) is described in Algorithm 2.

Lemma 1. *$O\left(\left\lceil \sigma_{\mathcal{M}}^2/\epsilon_1^2 \right\rceil \log \eta^{-1}\right)$ samples divided into $B = 36 \log \eta^{-1}$ equally sized batches are sufficient to compute $\hat{M}_1$ (see Algorithm 2) such that $\left|\hat{M}_1 - \mathbb{E}X\right| \leq \epsilon_1$ with probability at least $1 - 2\eta$.*

Lemma 2. *$O\left(\left\lceil \sigma_{\mathcal{M}}^4/\epsilon_2^2 \right\rceil \log \eta^{-1}\right)$ samples divided into $B = 36 \log \eta^{-1}$ equally sized batches is sufficient to compute $\hat{M}_2$ (see Algorithm 2) such that $\left|\hat{M}_2 - \mathrm{var}X\right| \leq \epsilon_2$ with probability at least $1 - 2\eta$.*

The detailed proofs of Lemma 1 and 2 can be found in Appendix A. We are now ready to prove Theorem 3.

Algorithm 2 ESTIMATE$(\mathbf{x}, T, B)$ Estimating $\mathbb{E}X$ and $\mathrm{var}X$ for $X \sim \mathcal{M}$

Require: I.i.d samples $y^1, y^2, \ldots, y^T \sim \mathcal{M}$ where $\mathcal{M} = \frac{1}{2}\mathcal{N}(\langle \mathbf{x}, \beta^1 \rangle, \sigma^2) + \frac{1}{2}\mathcal{N}(\langle \mathbf{x}, \beta^2 \rangle, \sigma^2)$.

1: Set $t = T/B$
2: **for** $i = 1, 2, \ldots, B$ **do**
3: Set Batch i to be the samples y^j for $j \in \{it + 1, it + 2, \ldots, (i+1)t\}$.
4: Set $S_1^i = \sum_{j \in \text{ Batch } i} \frac{y^i}{t}$, $S_2^i = \sum_{j \in \text{ Batch } i} \frac{(y^i - S_1^i)^2}{t-1}$.
5: **end for**
6: $\hat{M}_1 = \mathsf{median}(\{S_1^i\}_{i=1}^B)$, $\hat{M}_2 = \mathsf{median}(\{S_2^i\}_{i=1}^B)$.
7: Return $\hat{M}_1, \hat{M}_2$.

Proof of Theorem 3. We will set up the following system of equations in the variables $\hat{\mu}_1$ and $\hat{\mu}_2$:

$$\hat{\mu}_1 + \hat{\mu}_2 = 2\hat{M}_1 \text{ and } (\hat{\mu}_1 - \hat{\mu}_2)^2 = 4\hat{M}_2 - 4\sigma^2$$

Recall that from Lemma 1 and Lemma 2, we have computed $\hat{M}_1$ and $\hat{M}_2$ with the following guarantees: $\left|\hat{M}_1 - \mathbb{E}X\right| \leq \epsilon_1$ and $\left|\hat{M}_2 - \mathrm{var}X\right| \leq \epsilon_2$. Therefore, we must have $|\hat{\mu}_1 + \hat{\mu}_2 - \mu_1 - \mu_2| \leq 2\epsilon_1$, $\left|(\hat{\mu}_1 - \hat{\mu}_2)^2 - (\mu_1 - \mu_2)^2\right| \leq 4\epsilon_2$. We can factorize the left hand side of the second equation in the following way: $|\hat{\mu}_1 - \hat{\mu}_2 - \mu_1 + \mu_2|\,|\hat{\mu}_1 - \hat{\mu}_2 + \mu_1 - \mu_2| \leq 4\epsilon_2$. Notice that one of the factors must be less than $2\sqrt{\epsilon_2}$. Without loss of generality, let us assume that $|\hat{\mu}_1 - \hat{\mu}_2 - \mu_1 + \mu_2| \leq 2\sqrt{\epsilon_2}$. This, along with the fact $|\hat{\mu}_1 + \hat{\mu}_2 - \mu_1 - \mu_2| \leq 2\epsilon_1$ implies that (by adding and subtracting) $|\hat{\mu}_i - \mu_i| \leq 2\epsilon_1 + 2\sqrt{\epsilon_2} \quad \forall i = 1, 2$. □

Algorithm 3 METHOD OF MOMENTS$(\mathbf{x}, \sigma, T, B)$ Estimate the means $\langle \mathbf{x}, \beta^1 \rangle$, $\langle \mathbf{x}, \beta^2 \rangle$ for a query $\mathbf{x}$

Require: An oracle $\mathcal{O}$ which when queried with a vector $\mathbf{x} \in \mathbb{R}^n$ returns $\langle \mathbf{x}, \beta \rangle + \mathcal{N}(0, \sigma^2)$ where β is sampled uniformly from $\{\beta^1, \beta^2\}$.

1: **for** $i = 1, 2, \ldots, T$ **do**
2: Query the oracle $\mathcal{O}$ with $\mathbf{x}$ and obtain a response y^i.
3: **end for**
4: Compute $\hat{M}_1, \hat{M}_2$ (estimates of $\mathbb{E}_{X \sim \mathcal{M}} X, \mathrm{var}_{X \sim \mathcal{M}} X$ respectively) using Algorithm ESTIMATE$(\mathbf{x}, T, B)$.
5: Solve for $\hat{\mu}_1, \hat{\mu}_2$ in the system of equations $\hat{\mu}_1 + \hat{\mu}_2 = 2\hat{M}_1$, $(\hat{\mu}_1 - \hat{\mu}_2)^2 = 4\hat{M}_2 - 4\sigma^2$.
6: Return $\hat{\mu}_1, \hat{\mu}_2$.

2.2.3. FITTING A SINGLE GAUSSIAN

In the situation when both the variance σ^2 of each component in $\mathcal{M}$ and the separation between the means $|\mu_1 - \mu_2|$ are very small, fitting a single Gaussian $\mathcal{N}(\hat{\mu}, \sigma^2)$ to the samples obtained from $\mathcal{M}$ works better than the aforementioned techniques. The procedure to compute $\hat{M}_1$, an estimate of $\mathbb{E}_{X \sim \mathcal{M}} X = (\mu_1 + \mu_2)/2$ is adapted from (Daskalakis et al., 2017) and is described in Algorithm 4. Notice that Algorithm 4 is different from the naive procedure (averaging all samples) described in Algorithm 2 for estimating the mean of the mixture. The sample complexity for the naive procedure (see Lemma 1) scales with the gap $|\mu_1 - \mu_2|$ even when the variance σ^2 is small which is undesirable. In stead we have the following lemma.

Lemma 3 (Lemma 5 in (Daskalakis et al., 2017))**.** *With Algorithm 4, $O\left(\left\lceil \sigma^2 \log \eta^{-1}/\epsilon^2 \right\rceil\right)$ samples are sufficient to compute $\hat{M}_1$ such that $\left|\hat{M}_1 - (\mu_1 + \mu_2)/2\right| \leq \epsilon$ with probability at least $1 - \eta$.*

In this case, we will return $\hat{M}_1$ to be estimates of both the means μ_1, μ_2.

Algorithm 4 FIT A SINGLE GAUSSIAN($\mathbf{x}, T$) Estimate the means $\langle \mathbf{x}, \beta^1 \rangle$, $\langle \mathbf{x}, \beta^2 \rangle$ for a query $\mathbf{x}$

Require: An oracle $\mathcal{O}$ which when queried with a vector $\mathbf{x} \in \mathbb{R}^n$ returns $\langle \mathbf{x}, \beta \rangle + \mathcal{N}(0, \sigma^2)$ where β is sampled uniformly from $\{\beta^1, \beta^2\}$.

for $i = 1, 2, \ldots, T$ **do**
  Query the oracle $\mathcal{O}$ with $\mathbf{x}$ and obtain a response y^i.
end for
Set $\hat{Q}_1$ and $\hat{Q}_3$ to be the first and third quartiles of the samples $y^1, y^2, \ldots, y^t$ respectively.
Return $(\hat{Q}_1 + \hat{Q}_3)/2$.

2.2.4. CHOOSING APPROPRIATE METHODS

Among the above three methods to learn mixtures, the appropriate algorithm to apply for each parameter regime is listed below.

Case 1 ($|\mu_1 - \mu_2| = \Omega(\sigma)$): We use the EM algorithm for this regime to recover μ_1, μ_2. Notice that in this regime, by using Theorem 2 with $\epsilon = \gamma$, we obtain that $O\Big(\Big\lceil (\sigma^2/\gamma^2) \log 1/\eta \Big\rceil\Big)$ samples are sufficient to recover μ_1, μ_2 up to an additive error of γ with probability at least $1 - \eta$.

Case 2 ($\sigma \geq \gamma, |\mu_1 - \mu_2| = O(\sigma)$): We use the method of moments to recover μ_1, μ_2. In this regime, we must have $\sigma^2_{\mathcal{M}} = O(\sigma^2)$. Therefore, by using Theorem 3 with $\epsilon_1 = \gamma/4, \epsilon_2 = \gamma^2/16$, it is evident that $O\Big(\Big\lceil (\sigma/\gamma)^4 \Big\rceil \log 1/\eta\Big)$ samples are sufficient to recover μ_1, μ_2 upto an additive error of γ with probability at least $1 - \eta$.

Case 3 ($\sigma \leq \gamma, |\mu_1 - \mu_2| \leq \gamma$): In this setting, we fit a single Gaussian. Using Theorem 3 with $\epsilon = \gamma/2$, we will be able to estimate $(\mu_1 + \mu_2)/2$ up to an additive error of $\gamma/2$ using $O\Big(\Big\lceil (\sigma^2/\gamma^2) \log 1/\eta \Big\rceil\Big)$ samples. This, in turn implies

$$|\mu_i - \hat{M}_1| \leq \frac{|\mu_1 - \mu_2|}{2} + \left|\frac{\mu_1 + \mu_2}{2} - \hat{M}_1\right| \leq \gamma.$$

for $i \in \{1, 2\}$ and therefore both the means μ_1, μ_2 are recovered up to an additive error of γ. Note that these three cases covers all possibilities.

2.2.5. TEST FOR APPROPRIATE METHOD

Now, we describe a test to infer which parameter regime we are in and therefore which algorithm to use. The final algorithm to recover the means μ_1, μ_2 from $\mathcal{M}$ including the test is described in Algorithm 5. We have the following result, the proof of which is delegated to appendix B.

Lemma 4. *The number of samples required for Algorithm 5 to infer the parameter regime correctly with probability at least $1 - \eta$ is atmost $O(\log \eta^{-1})$.*

Algorithm 5 TEST AND ESTIMATE($\mathbf{x}, \sigma, \gamma, \eta$) Test for the correct parameter regime and apply the parameter estimation algorithm accordingly for a query $\mathbf{x}$

Require: An oracle $\mathcal{O}$ which when queried with a vector $\mathbf{x} \in \mathbb{R}^n$ returns $\langle \mathbf{x}, \beta \rangle + \mathcal{N}(0, \sigma^2)$ where β is sampled uniformly from $\{\beta^1, \beta^2\}$.

Set $T = O\Big(\Big\lceil \log \eta^{-1} \Big\rceil\Big)$.
for $i = 1, 2, \ldots, T$ **do**
  Query the oracle $\mathcal{O}$ with $\mathbf{x}$ and obtain a response y^i.
end for
Compute $\tilde{\mu}_1, \tilde{\mu}_2$ by running Algorithm METHOD OF MOMENTS ($\mathbf{x}, \sigma, T, 72 \log n$).
if $\sigma > \gamma$ and $|\tilde{\mu}_1 - \tilde{\mu}_2| \leq 15\sigma/32$ **then**
  Compute $\hat{\mu}_1, \hat{\mu}_2$ by running Algorithm METHOD OF MOMENTS ($\mathbf{x}, \sigma, O\Big(\Big\lceil (\sigma/\gamma)^4 \Big\rceil \log 1/\eta, 72 \log n\Big)$.
else if $\sigma \leq \gamma$ and $|\tilde{\mu}_1 - \tilde{\mu}_2| \leq 15\gamma/32$ **then**
  Compute $\hat{\mu}_1, \hat{\mu}_2$ by running Algorithm FIT A SINGLE GAUSSIAN ($\mathbf{x}, O\Big(\Big\lceil (\sigma^2/\gamma^2) \log 1/\eta \Big\rceil\Big)$.
else
  Compute $\hat{\mu}_1, \hat{\mu}_2$ by running Algorithm EM($\mathbf{x}, \sigma, O\Big(\Big\lceil (\sigma^2/\gamma^2) \log 1/\eta \Big\rceil\Big)$.
end if
Return $\hat{\mu}_1, \hat{\mu}_2$.

2.3. Alignment

For a query $\mathbf{x}^i, i \in [m]$, let us introduce the following notations for brevity:

$$\mu_{i,1} := \langle \mathbf{x}^i, \beta_1 \rangle \quad \mu_{i,2} := \langle \mathbf{x}^i, \beta_2 \rangle.$$

Now, using Algorithm 5, we can compute $(\hat{\mu}_{i,1}, \hat{\mu}_{i,2})$ (estimates of $\mu_{i,1}, \mu_{i,2}$) using a batchsize of T_i such that $\left|\hat{\mu}_{i,j} - \mu_{i,\pi_i(j)}\right| \leq \gamma \quad \forall\ i \in [m], j \in \{1, 2\}$, where $\pi_i : \{1, 2\} \to \{1, 2\}$ is a permutation on $\{1, 2\}$.

The most important step in our process is to separate the estimates of the means according to the generative unknown sparse vectors (β^1 and β^2) (i.e., alignment). Formally, we construct two m-dimensional vectors $\mathbf{u}$ and $\mathbf{v}$ such that, for all $i \in [m]$ the following hold:

- The i^{th} elements of $\mathbf{u}$ and $\mathbf{v}$, i.e., u_i and v_i, are $\hat{\mu}_{i,1}$ and $\hat{\mu}_{i,2}$ (but may not be respectively).
- Moreover, we must have the u_i and v_i to be good estimates of $\langle \mathbf{x}^i, \beta^{\pi(1)} \rangle$ and $\langle \mathbf{x}^i, \beta^{\pi(2)} \rangle$ respectively i.e. $\left|u_i - \langle \mathbf{x}^i, \beta^{\pi(1)} \rangle\right| \leq 10\gamma$; $\left|v_i - \langle \mathbf{x}^i, \beta^{\pi(2)} \rangle\right| \leq 10\gamma$ for all $i \in [m]$ where $\pi : \{1, 2\} \to \{1, 2\}$ is some permutation of $\{1, 2\}$.

In essence, for the alignment step, we want to find out all permutations $\pi_i, i \in [m]$. First, note that the aforementioned objective is trivial when $|\mu_{i,1} - \mu_{i,2}| \leq 9\gamma$. To see this, suppose π_i is the identity permutation without loss of

Algorithm 6 ALIGN PAIR$(\mathbf{x}^i, \mathbf{x}^j, \{\hat{\mu}_{s,t}\}_{\substack{s=i,j \\ t=1,2}}, \sigma, \gamma, \eta)$

Align the mean estimates for $\mathbf{x}^i$ and $\mathbf{x}^j$.

1: Recover $\hat{\mu}_{\mathsf{sum},1}, \hat{\mu}_{\mathsf{sum},2}$ using Algorithm TEST AND ESTIMATE $(\mathbf{x}^i + \mathbf{x}^j, \sigma, \gamma, \eta)$.
2: Recover $\hat{\mu}_{\mathsf{diff},1}, \hat{\mu}_{\mathsf{diff},2}$ using Algorithm TEST AND ESTIMATE $(\mathbf{x}^i - \mathbf{x}^j, \sigma, \gamma, \eta)$.
3: **if** $|\hat{\mu}_{\mathsf{sum},1} - \hat{\mu}_{i,p} - \hat{\mu}_{j,q}| \le 3\gamma$ such that $p, q \in \{1,2\}$ is unique **then**
4: **if** $p == q$ **then** Return TRUE **else** Return FALSE **end if**
5: **else**
6: Find p, q such that $|\hat{\mu}_{\mathsf{diff},1} - \hat{\mu}_{i,p} + \hat{\mu}_{j,q}| \le 3\gamma$ for $p, q \in \{1,2\}$.
7: **if** $p == q$ **then** Return TRUE **else** Return FALSE **end if**
8: **end if**

generality. In that case, we have for $\hat{\mu}_{i,1}$, $|\hat{\mu}_{i,1} - \mu_{i,1}| \le \gamma$ and $|\hat{\mu}_{i,1} - \mu_{i,2}| \le |\hat{\mu}_{i,1} - \mu_{i,1}| + |\mu_{i,1} - \mu_{i,2}| \le 10\gamma$. Similar guarantees also hold for $\hat{\mu}_{i,2}$ and therefore the choice of the i^{th} element of $\mathbf{u}, \mathbf{v}$ is trivial. This conclusion implies that the interesting case is only for those queries $\mathbf{x}^i$ when $|\mu_{i,1} - \mu_{i,2}| \ge 9\gamma$. In other words, this objective is equivalent to separate out the permutations $\{\pi_i\}_{i=1}^m$ for $i : |\mu_{i,1} - \mu_{i,2}| \ge 9\gamma$ into two groups such that all the permutations in each group are the same.

2.3.1. ALIGNMENT FOR TWO QUERIES

Consider two queries $\mathbf{x}^1, \mathbf{x}^2$ such that $|\mu_{i,1} - \mu_{i,2}| \ge 9\gamma$ for $i = 1, 2$. In this section, we will show how we can infer if π_1 is same as π_2. Our strategy is to make two additional batches of queries corresponding to $\mathbf{x}^1 + \mathbf{x}^2$ and $\mathbf{x}^1 - \mathbf{x}^2$ (of size $T_{1,2}^{\mathsf{sum}}$ and $T_{1,2}^{\mathsf{diff}}$ respectively) which we shall call the *sum* and *difference* queries. Again, let us introduce the following notations: $\mu_{\mathsf{sum},1} = \langle \mathbf{x}^1 + \mathbf{x}^2, \beta_1 \rangle \quad \mu_{\mathsf{sum},2} = \langle \mathbf{x}^1 + \mathbf{x}^2, \beta_2 \rangle$, $\mu_{\mathsf{diff},1} = \langle \mathbf{x}^1 - \mathbf{x}^2, \beta_1 \rangle \quad \mu_{\mathsf{diff},2} = \langle \mathbf{x}^1 - \mathbf{x}^2, \beta_2 \rangle$. As before, using Algorithm 5, we can compute $(\hat{\mu}_{\mathsf{sum},1}, \hat{\mu}_{\mathsf{sum},2})$ (estimates of $\mu_{\mathsf{sum},1}, \mu_{\mathsf{sum},2}$) and $(\hat{\mu}_{\mathsf{diff},1}, \hat{\mu}_{\mathsf{diff},2})$ (estimates of $\mu_{\mathsf{diff},1}, \mu_{\mathsf{diff},2}$) using a batch-size of $T_{1,2}^{\mathsf{sum}}$ and $T_{1,2}^{\mathsf{diff}}$ for the *sum* and *difference* query respectively such that $\left|\hat{\mu}_{\mathsf{sum},j} - \mu_{\mathsf{sum},\pi_{\mathsf{sum}}(j)}\right| \le \gamma \quad$ for $j \in \{1,2\}$ and $\left|\hat{\mu}_{\mathsf{diff},j} - \mu_{\mathsf{diff},\pi_{\mathsf{diff}}(j)}\right| \le \gamma \quad$ for $j \in \{1,2\}$ where $\pi_{\mathsf{sum}}, \pi_{\mathsf{diff}} : \{1,2\} \to \{1,2\}$ are again unknown permutations of $\{1,2\}$. We show the following lemma.

Lemma 5. *We can infer, using Algorithm 6, if π_1 and π_2 are same using the estimates $\hat{\mu}_{\mathsf{sum},i}, \hat{\mu}_{\mathsf{diff},i}$ provided $|\mu_{i,1} - \mu_{i,2}| \ge 9\gamma$, $i = 1, 2$.*

The proof of this lemma is delegated to appendix C and we provide an outline over here. In Algorithm 6, we first choose one value from $\{\hat{\mu}_{\mathsf{sum},1}, \hat{\mu}_{\mathsf{sum},2}\}$ (say z) and we check if we can choose one element (say a) from the set $\{\hat{\mu}_{1,1}, \hat{\mu}_{1,2}\}$ and one element $\{\hat{\mu}_{2,1}, \hat{\mu}_{2,2}\}$ (say b) in exactly one way such that $|z - a - b| \le 3\gamma$. If that is true, then we infer that the tuple $\{a, b\}$ are estimates of the same unknown vector and accordingly return if π_1 is same as π_2. If not possible, then we choose one value from $\{\hat{\mu}_{\mathsf{diff},1}, \hat{\mu}_{\mathsf{diff},2}\}$ (say z') and again we check if we can choose one element (say c) from the set $\{\hat{\mu}_{1,1}, \hat{\mu}_{1,2}\}$ and one element from $\{\hat{\mu}_{2,1}, \hat{\mu}_{2,1}\}$ (say d) in exactly one way such that $|z' - c - d| \le 3\gamma$. If that is true, then we infer that $\{c, d\}$ are estimates of the same unknown vector and accordingly return if π_1 is same as π_2. It can be shown that we will succeed in this step using at least one of the sum or difference queries.

2.3.2. ALIGNMENT FOR ALL QUERIES

We will align the mean estimates for all the queries $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^m$ by aligning one pair at a time. This routine is summarized in Algorithm 7, which works when $\gamma \le \frac{13\sqrt{2}}{\sqrt{\pi}} \|\beta^1 - \beta^2\|_2 \approx 0.096 \|\beta^1 - \beta^2\|_2$. To understand the routine, we start with the following technical lemma:

Lemma 6. *Let, $\gamma \le \frac{13\sqrt{2}}{\sqrt{\pi}} \|\beta^1 - \beta^2\|_2$. For $m' = \left\lceil \log \eta^{-1} / \log \frac{\sqrt{\pi}||\beta^1 - \beta^2||_2}{13\sqrt{2}\gamma} \right\rceil$, there exists a query $\mathbf{x}^{i^\star}$ among $\{\mathbf{x}^i\}_{i=1}^{m'}$ such that $|\mu_{i^\star,1} - \mu_{i^\star,2}| \ge 13\gamma$ with probability at least $1 - \eta$.*

Algorithm 7 ALIGN ALL$(\{\mathbf{x}^i\}_{i\in[m]}, \{\hat{\mu}_{s,t}\}_{\substack{s\in[m] \\ t=1,2}}, \sigma, \gamma, \eta)$

Align mean estimates for all queries $\{\mathbf{x}^i\}_{i=1}^m$.

1: **Initialize:** $\mathbf{u}, \mathbf{v}$ to be m-dimensional all zero vector.
2: Set $m' = \left\lceil \log \eta^{-1} / \log \frac{\sqrt{\pi}||\beta^1 - \beta^2||_2}{13\sqrt{2}\gamma} \right\rceil$
3: **for** $i = 1, 2, \ldots, m$ **do**
4: **for** $j = 1, 2, \ldots, m', j \ne i$ **do**
5: Run Algorithm ALIGN PAIR $(\mathbf{x}^i, \mathbf{x}^j, \{\hat{\mu}_{s,t}\}_{\substack{s=i,j \\ t=1,2}}, \sigma, \gamma, \eta)$ and store the output.
6: **end for**
7: **end for**
8: Identify $\mathbf{x}^p$ from $p \in [m']$ such that $|\hat{\mu}_{p,1} - \hat{\mu}_{p,2}| \ge 11\gamma$.
9: Set $u_p := \mathbf{u}[p] = \hat{\mu}_{p,1}$ and $v_p := \mathbf{v}[p] = \hat{\mu}_{p,2}$
10: **for** $i = 1, 2, \ldots, m, i \ne p$ **do**
11: **if** Output of Algorithm 6 for $\mathbf{x}^i$ and $\mathbf{x}^p$ is TRUE **then**
12: Set $\mathbf{u}[i] = \hat{\mu}_{i,1}$ and $\mathbf{v}[i] = \hat{\mu}_{i,2}$.
13: **else**
14: Set $\mathbf{u}[i] = \hat{\mu}_{i,2}$ and $\mathbf{v}[i] = \hat{\mu}_{i,1}$.
15: **end if**
16: **end for**
17: Return $\mathbf{u}, \mathbf{v}$.

The proof of this lemma is delegated to Appendix D. Now, for $i \in [m'], j \in [m]$ such that $i \ne j$, we will align $\mathbf{x}^i$ and $\mathbf{x}^j$ using Algorithm 6 and according to Lemma 5, this

alignment procedure will succeed for all such pairs where $|\mu_{i,1} - \mu_{i,2}|, |\mu_{j,1} - \mu_{j,2}| \geq 9\gamma$ with probability at least $1 - mm'\eta$ (using a union bound). Note that according to Lemma 6, there must exist a query $\mathbf{x}^{i^\star} \in \{\mathbf{x}^i\}_{i=1}^{m'}$ for which $|\mu_{i^\star,1} - \mu_{i^\star,2}| \geq 13\gamma$. This implies that for some $i^\star \in [m']$, we must have $|\hat{\mu}_{i^\star,1} - \hat{\mu}_{i^\star,2}| \geq |\mu_{i^\star,1} - \mu_{i^\star,2}| - |\hat{\mu}_{i^\star,1} - \mu_{i^\star,1}| - |\hat{\mu}_{i^\star,2} - \mu_{i^\star,2}| \geq 11\gamma$. Therefore, we can identify at least one query $\mathbf{x}^{\tilde{i}}$ for $\tilde{i} \in [m']$ such that $|\hat{\mu}_{\tilde{i},1} - \hat{\mu}_{\tilde{i},2}| \geq 11\gamma$. However, this implies that $|\mu_{\tilde{i},1} - \mu_{\tilde{i},2}| \geq |\hat{\mu}_{\tilde{i},1} - \hat{\mu}_{\tilde{i},2}| - |\mu_{\tilde{i},1} - \hat{\mu}_{\tilde{i},1}| - |\mu_{\tilde{i},2} - \hat{\mu}_{\tilde{i},2}| \geq 9\gamma$. Therefore we will be able to infer for every query $\mathbf{x}^i, i \in [m]$ for which $|\mu_{i,1} - \mu_{i,2}| \geq 9\gamma$ if π_i is same as $\pi_{\tilde{i}}$. Now, we are ready to put everything together and provide the proof for the main result (Thm. 1).

2.4. Proof of Theorem 1 ($\gamma < \left|\left|\beta^1 - \beta^2\right|\right|_2/2$)

The overall recovery procedure is described as Algorithm 8. Since this algorithm crucially uses Algorithm 7, it works only when $\gamma \leq 0.096 \left|\left|\beta^1 - \beta^2\right|\right|_2$; so assume that to hold for now. We will start by showing that for any two Gaussian queries, the samples are far enough (a simple instance of Gaussian anti-concentration).

Algorithm 8 RECOVER UNKNOWN VECTORS(σ, γ) Recover the unknown vectors β^1 and β^2

1: Set $m = c_s k \log n$.
2: Sample $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^m \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ independently.
3: **for** $i = 1, 2, \ldots, m$ **do**
4: Compute $\hat{\mu}_{i,1}, \hat{\mu}_{i,2}$ by running Algorithm TEST AND ESTIMATE$(\mathbf{x}^i, \sigma, \gamma, n^{-2})$.
5: **end for**
6: Construct $\mathbf{u}, \mathbf{v}$ by running Algorithm ALIGN ALL$(\{\mathbf{x}^i\}_{i\in[m]}, \{\hat{\mu}_{s,t}\}_{\substack{s\in[m]\\ t=1,2}}, \sigma, \gamma, \eta)$.
7: Set $\mathbf{A}$ to be the $m \times n$ matrix such that its i^{th} row is $\mathbf{x}^i$, with each entry normalized by $\sqrt{m}$.
8: Set $\hat{\beta}^1$ to be the solution of the optimization problem $\min_{\mathbf{z}\in\mathbb{R}^n} ||\mathbf{z}||_1$ s.t. $||\mathbf{A}\mathbf{z} - \frac{1}{\sqrt{m}}\mathbf{u}||_2 \leq 10\gamma$
9: Set $\hat{\beta}^2$ to be the solution of the optimization problem $\min_{\mathbf{z}\in\mathbb{R}^n} ||\mathbf{z}||_1$ s.t. $||\mathbf{A}\mathbf{z} - \frac{1}{\sqrt{m}}\mathbf{v}||_2 \leq 10\gamma$
10: Return $\hat{\beta}^1, \hat{\beta}^2$.

Lemma 7. *For all queries $\mathbf{x}$ designed in Algorithm 8, for any constant $c_1 > 0$, and some c_2 which is a function of c_1,*

$$\Pr(\left|\langle \mathbf{x}, \beta^1\rangle - \langle \mathbf{x}, \beta^2\rangle\right| \leq c_1\sigma)) \leq \frac{c_2\sigma}{||\beta^1 - \beta^2||_2}.$$

The proof of this lemma is delegated to Appendix D. Now the theorem is proved via a series of claims.

Claim 1. *The expected batchsize for any query designed in Algorithm 8 is $O\Big(\Big\lceil \frac{\sigma^5}{\gamma^4||\beta^1-\beta^2||_2} + \frac{\sigma^2}{\gamma^2}\Big\rceil \log \eta^{-1}\Big)$.*

Proof. In Algorithm 8, we make m batches of queries corresponding to $\{\mathbf{x}^i\}_{i=1}^m$ and mm' batches of queries corresponding to $\{\mathbf{x}^i + \mathbf{x}^j\}_{i=1,j=1,i\neq j}^{i=m,j=m'}$ and $\{\mathbf{x}^i - \mathbf{x}^j\}_{i=1,j=1,i\neq j}^{i=m,j=m'}$. Recall that the batchsize corresponding to $\mathbf{x}^i, \mathbf{x}^i + \mathbf{x}^j, \mathbf{x}^i - \mathbf{x}^j$ is denoted by T_i,$T_{i,j}^{\mathsf{sum}}$ and $T_{i,j}^{\mathsf{diff}}$ respectively. Recall from Section 2.2.4, we will use method of moments or or fit a single Gaussian (Case 2 and 3 in Section 2.2.4) to estimate the means when the difference between the means is $O(\sigma)$. By Lemma 7, this happens with probability $O(\sigma/||\beta^1 - \beta^2||_2)$. Otherwise we will use the EM algorithm (Case 1 in Section 2.2.4). or fit a single gaussian, both of which require a batchsize of at most $O\Big(\Big\lceil \sigma^2/\gamma^2 \Big\rceil \log \eta^{-1}\Big)$. We can conclude that the expected size of any of the aforementioned batchsize is bounded from above as the following: $\mathbb{E}T \leq O\Big(\Big\lceil \frac{\sigma^5}{\gamma^4||\beta^1-\beta^2||_2} + \frac{\sigma^2}{\gamma^2}\Big\rceil \log \eta^{-1}\Big)$ where $T \in \{T_i\} \cup \{T_{i,j}^{\mathsf{sum}}\} \cup \{T_{i,j}^{\mathsf{diff}}\}$ so that we can recover all the mean estimates upto an an additive error of γ with probability at least $1 - O(mm'\eta)$. □

Claim 2. *Algorithm 7 returns two vectors $\mathbf{u}$ and $\mathbf{v}$ of length m each such that*

$$\left|\mathbf{u}[i] - \langle \mathbf{x}^i, \beta^{\pi(1)}\rangle\right| \leq 10\gamma; \left|\mathbf{v}[i] - \langle \mathbf{x}^i, \beta^{\pi(2)}\rangle\right| \leq 10\gamma$$

for some permutation $\pi : \{1, 2\} \to \{1, 2\}$ for all $i \in [m]$ with probability at least $1 - \eta$.

The proof of this claim directly follows from the discussion in Section 2.3.2.

The matrix $\mathbf{A}$ is size $m \times n$ whose i^{th} row is the query vector $\mathbf{x}^i$ normalized by $\sqrt{m}$. .

Claim 3. *We must have*

$$||\mathbf{A}\beta^{\pi(1)} - \frac{\mathbf{u}}{\sqrt{m}}||_2 \leq 10\gamma \;\&\; ||\mathbf{A}\beta^{\pi(2)} - \frac{\mathbf{v}}{\sqrt{m}}||_2 \leq 10\gamma.$$

Proof. The proof of this claim follows from the fact that after normalization by $\sqrt{m}$, the error in each entry is also normalized by $\sqrt{m}$ and is therefore at most $10\gamma/\sqrt{m}$. Hence the ℓ_2 difference is at most 10γ. □

It is known that for $m \geq c_s k \log n$ where $c_s > 0$ is some appropriate constant, the matrix $\mathbf{A}$ satisfy *restricted isometric property* of order $2k$, which means for any exactly $2k$-sparse vector $\mathbf{x}$ and a constant δ, we have $|\|\mathbf{A}\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2| \leq \delta\|\mathbf{x}\|_2^2$ cf.(Baraniuk et al., 2008).

We now solve the following convex optimization problems, standard recovery method called basis pursuit:

$$\hat{\beta}^{\pi(1)} = \min_{\mathbf{z}\in\mathbb{R}^n} ||\mathbf{z}||_1 \text{ s.t. } ||\mathbf{A}\mathbf{z} - \frac{\mathbf{u}}{\sqrt{m}}||_2 \leq 10\gamma$$
$$\hat{\beta}^{\pi(2)} = \min_{\mathbf{z}\in\mathbb{R}^n} ||\mathbf{z}||_1 \text{ s.t. } ||\mathbf{A}\mathbf{z} - \frac{\mathbf{v}}{\sqrt{m}}||_2 \leq 10\gamma$$

to recover $\hat{\beta}^{\pi(1)}, \hat{\beta}^{\pi(2)}$, estimates of β^1, β^2 having the guarantees given in Theorem 1 (see, Thm. 1.6 in (Boche et al., 2015)). The expected query complexity is $O\Big(mm' \log \eta^{-1} \Big\lceil \frac{\sigma^5}{\gamma^4 ||\beta^1 - \beta^2||_2} + \frac{\sigma^2}{\gamma^2} \Big\rceil\Big)$ Substituting $m = O(k \log n)$, $m' = O\Big(\Big\lceil \frac{\log \eta^{-1}}{\log \frac{||\beta^1 - \beta^2||_2}{\gamma}} \Big\rceil\Big)$ and $\eta = (mm' \log n)^{-1}$, we obtain the total query complexity

$$O\Bigg(k \log n \log k \Big\lceil \frac{\log k}{\log(\|\beta^1 - \beta^2\|_2/\gamma)} \Big\rceil \times \Big\lceil \frac{\sigma^5}{\gamma^4 ||\beta^1 - \beta^2||_2} + \frac{\sigma^2}{\gamma^2} \Big\rceil \Bigg)$$

and the error probability to be $o(1)$. We can just substitute the definition of NF and notice that $\mathsf{SNR} = \|\beta^1 - \beta^2\|_2^2/\sigma^2$ to obtain the query complexity promised in Theorem 1. Note that, we have assumed $k = \Omega(\log n)$ above.

It remains to be proved that the same (orderwise) number of samples is sufficient to recover both unknown vectors with high probability. For each query $\mathbf{x}$ designed in Algorithm 8, consider the indicator random variable $Y_i = \mathbb{1}[|\mu_{i,1} - \mu_{i,2}| = \Omega(\sigma)]$. The total number of queries for which this event is true (given by $\sum_i Y_i$) is sampled according to the binomial distribution $\mathsf{Bin}(mm', O(\sigma/||\beta^1 - \beta^2||_2))$ and therefore concentrates tightly around its mean. A simple use of Chernoff bound leads to the desired result.

While we have proved the theorem for any $\gamma \leq 0.096 \left|\left|\beta^1 - \beta^2\right|\right|_2$, it indeed holds for any $\gamma = c' \left|\left|\beta^1 - \beta^2\right|\right|_2$, where c' is a constant strictly less than 1. If the desired $\gamma > 0.096 \left|\left|\beta^1 - \beta^2\right|\right|_2$, then one can just define $\gamma' = 0.096 \left|\left|\beta^1 - \beta^2\right|\right|_2$ and obtain a precision γ' which is a constant factor within γ. Since the quantity NF defined with γ' is also within a constant factor of the original NF, the sample complexity can also change by at most a constant factor.

2.5. Proof of Theorem 1 ($\gamma = \Omega\left(\left|\left|\beta^1 - \beta^2\right|\right|_2\right)$)

The proof of Theorem 1 for the case when recovery precision $\gamma = \Omega(\left|\left|\beta^1 - \beta^2\right|\right|_2)$ follows by fitting a single Gaussian through all the samples. The algorithm for this case, and the proof, are delegated to Appendix E.

3. Conclusion

In this paper we have improved the recent results by (Yin et al., 2019) and (Krishnamurthy et al., 2019) for learning a mixture of sparse linear regressions when features can be designed and queried with for the labels. While our results are rigorously proved for two unknown sparse models, we believe extending to more than two models will be possible, and the key components are already present in our paper. Whether it will be an exercise in technicality or some key insights can be gained is unclear.

While our paper is theoretical, an important future work will be to find interesting use cases. A potential application of the query-based setting is to recommendation systems, where the goal is to identify the factors governing the preferences of individual members of a group via crowdsourcing while also preserving the anonymity of their responses. We are currently pursuing this line of applications.

Acknowledgements: This work is supported in parts by NSF awards 1642658, 1909046 and 1934846.

References

Balakrishnan, S., Wainwright, M. J., Yu, B., et al. Statistical guarantees for the EM algorithm: From poulation to sample-based analysis. *The Annals of Statistics*, 45(1): 77–120, 2017.

Baraniuk, R., Davenport, M., DeVore, R., and Wakin, M. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3): 253–263, 2008.

Boche, H., Calderbank, R., Kutyniok, G., and Vybíral, J. A survey of compressed sensing. In *Compressed Sensing and its Applications*, pp. 1–39. Springer, 2015.

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

Candes, E. J. et al. The restricted isometry property and its implications for compressed sensing. *Comptes rendus mathematique*, 346(9-10):589–592, 2008.

Chaganty, A. T. and Liang, P. Spectral experts for estimating mixtures of linear regressions. In *International Conference on Machine Learning*, pp. 1040–1048, 2013.

Dasgupta, S. Learning mixtures of Gaussians. In *Foundations of Computer Science*, pp. 634–644, 1999.

Daskalakis, C. and Kamath, G. Faster and sample near-optimal algorithms for proper learning mixtures of Gaussians. In *Conference on Learning Theory*, 2014.

Daskalakis, C., Tzamos, C., and Zampetakis, M. Ten steps of em suffice for mixtures of two Gaussians. In *Conference on Learning Theory*, pp. 704–710, 2017.

De Veaux, R. D. Mixtures of linear regressions. *Computational Statistics & Data Analysis*, 8(3):227–245, 1989.

Donoho, D. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

Faria, S. and Soromenho, G. Fitting mixtures of linear regressions. *Journal of Statistical Computation and Simulation*, 80(2):201–225, 2010.

Hardt, M. and Price, E. Tight bounds for learning a mixture of two Gaussians. In *Symposium on Theory of Computing*, 2015.

Krishnamurthy, A., Mazumdar, A., McGregor, A., and Pal, S. Sample complexity of learning mixture of sparse linear regressions. In *Advances in Neural Information Processing Systems 32: NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pp. 10531–10540, 2019.

Krishnamurthy, A., Mazumdar, A., McGregor, A., and Pal, S. Algebraic and analytic approaches for parameter learning in mixture models. In *Proceedings of the 31st International Conference on Algorithmic Learning Theory*, volume 117 of *Proceedings of Machine Learning Research*, pp. 468–489, San Diego, California, USA, 2020.

Kwon, J. and Caramanis, C. Global convergence of em algorithm for mixtures of two component linear regression. *arXiv preprint arXiv:1810.05752*, 2018.

Städler, N., Bühlmann, P., and Van De Geer, S. l1-penalization for mixture regression models. *Test*, 19(2): 209–256, 2010.

Titterington, D. M., Smith, A. F., and Makov, U. E. *Statistical analysis of finite mixture distributions*. Wiley, 1985.

Viele, K. and Tong, B. Modeling with mixtures of linear regressions. *Statistics and Computing*, 12(4):315–330, 2002.

Xu, J., Hsu, D. J., and Maleki, A. Global analysis of expectation maximization for mixtures of two Gaussians. In *Advances in Neural Information Processing Systems*, pp. 2676–2684, 2016.

Yi, X., Caramanis, C., and Sanghavi, S. Alternating minimization for mixed linear regression. In *International Conference on Machine Learning*, pp. 613–621, 2014.

Yi, X., Caramanis, C., and Sanghavi, S. Solving a mixture of many random linear equations by tensor decomposition and alternating minimization. *arXiv preprint arXiv:1608.05749*, 2016.

Yin, D., Pedarsani, R., Chen, Y., and Ramchandran, K. Learning mixtures of sparse linear regressions using sparse graph codes. *IEEE Transactions on Information Theory*, 65(3):1430–1451, 2019.