Testing of Index-Invariant Properties in the Huge Object Model

Sourav Chakraborty* Eldar Fischer† Arijit Ghosh‡ Gopinath Mishra§
Sayantan Sen¶

Abstract

The study of distribution testing has become ubiquitous in the area of property testing, both for its theoretical appeal, as well as for its applications in other fields of Computer Science and Machine Learning and various real-life statistical tasks.

The original distribution testing model relies on samples drawn independently from the distribution to be tested. However, when testing distributions over the n-dimensional Hamming cube $\{0,1\}^n$ for a large n, even reading a few samples is infeasible. To address this, Goldreich and Ron [ITCS 2022] have defined a model called the *huge object model*, in which the samples may only be queried in a few places. In this paper, we will address the following two fundamental questions about the huge object model:

- Characterization of properties that are *testable*. Note that a property is said to be testable with respect to a query model if it can be tested using a constant number of queries.
- Understanding query complexity of non-adaptive algorithms in terms of the query complexity of the adaptive algorithms.

In this work, we initiate a study of a general class of properties in the huge object model, those that are invariant under a permutation of the indices of the vectors in $\{0,1\}^n$, while still not being necessarily fully symmetric as per the definition used in traditional distribution testing.

We prove that every index-invariant property satisfying a bounded VC-dimension restriction admits a property tester with a number of queries independent of n. To complement this result, we argue that satisfying only index-invariance or only a VC-dimension bound is insufficient to guarantee a tester whose query complexity is independent of n. Moreover, we prove that the dependency of sample and query complexities of our tester on the VC-dimension is essentially tight. As a second part of this work, we address the question of the number of queries required for non-adaptive testing. We show that it can be at most quadratic in the number of queries required for an adaptive tester in the case of index-invariant properties. This is in contrast with the tight (easily provable) exponential gap between adaptive and non-adaptive testers for general non-index-invariant properties. Finally, we provide an index-invariant property for which the quadratic gap between adaptive and non-adaptive query complexities for testing is almost tight.

*Indian Statistical Institute, Kolkata, India. Email: chakraborty.sourav@gmail.com.

†Technion - Israel Institute of Technology, Israel. Email: eldar@cs.technion.ac.il. Research supported in part by an Israel Science Foundation grant number 879/22.

‡Indian Statistical Institute, Kolkata, India. Email: arijitiitkgpster@gmail.com.

§University of Warwick, UK. Email: gopianjan117@gmail.com. Research supported in part by the Centre for Discrete Mathematics and its Applications (DIMAP) and by EPSRC award EP/V01305X/1.

¶Indian Statistical Institute, Kolkata, India. Email: sayantan789@gmail.com.

Contents

1 Introduction **3**
- 1.1 Our Results . . . 4
- 1.2 Organization of the paper . . . 7

2 Technical overview of our results **8**
- 2.1 Overview of our upper bound results for index-invariant bounded VC-dimension property 8
- 2.2 Overview of the lower bound result for index-invariant bounded VC-dimension properties 9
- 2.3 Overview of adaptive vs non-adaptive query complexity results . . . 10

3 Preliminaries **11**
- 3.1 Definitions and relations of various distance measures of distributions . . . 12
- 3.2 Formal definitions of various kinds of property testers . . . 13
- 3.3 Distributions and properties with bounded VC-dimension . . . 14
- 3.4 Yao's lemma for the huge object model . . . 15

4 Learning clusterable distributions **16**
- 4.1 Useful lemmas and events to prove the correctness of Test-and-Learn . . . 18
- 4.2 Proof of Theorem 4.2 (Correctness of Test-and-Learn) . . . 21

5 Testing properties with bounded VC-dimension **26**
- 5.1 A corollary of Theorem 4.2 to prove Theorem 5.1 . . . 27
- 5.2 A corollary from VC theory required to prove Theorem 5.1 . . . 28
- 5.3 Proof of Theorem 5.1 and testing bounded VC-dimension properties . . . 29

6 Tightness of the bounds for bounded VC-dimension properties **30**

7 Exponential gap between adaptive and non-adaptive testers for general properties **35**
- 7.1 Relation between adaptive and non-adaptive testers for general properties . . . 35
- 7.2 Exponential separation between adaptive and non-adaptive query complexities . . . 37

8 Quadratic gap between adaptive and non-adaptive testers for index-invariant properties **40**
- 8.1 Quadratic relation between adaptive and non-adaptive testers for index-invariant properties . . . 40
- 8.2 Preliminaries towards proving a quadratic separation result . . . 42
- 8.3 Determining the permutation π . . . 46
- 8.4 The upper bound on adaptive testing for property $\mathcal{P}_{\mathrm{Gap}}$ . . . 48
- 8.5 Near-quadratic lower bound for non-adaptive testers for testing $\mathcal{P}_{\mathrm{Gap}}$ . . . 55

A Some probability results **62**

1 Introduction

The field of distribution testing is currently ubiquitous in property testing and also plays a central role in various outher areas, such as computational and statistical learning theory, probabilistic program checking. See the books and surveys of [Gol17, DK20, BY22, Fis01, Ron08, Ron09, CS10, RS11, Can20, Can22] for reference. Distribution testing has also found numerous applications in other areas of research, including topics that have real life applications (see [CM19, MPC20, CKS20, CDKS20, ACL$^+$21, ASZ21, PM21] for references).

In the original model of distribution testing, a distribution D defined over some set Ω can be accessed by obtaining independent samples from D, and the goal is to approximate various interesting properties of D. This model has been studied extensively over the last two decades, and many interesting results and techniques have emerged.

The majority of distribution testing research centers on the goal of minimizing the number of samples required to test for various properties of the underlying distribution. If the domain of the distribution is structured (for example, if the domain is the n-dimensional Hamming cube $\{0,1\}^n$), then designing efficient testers brings its own challenges. A number of papers have studied the problem of testing properties of distributions defined over the n-dimensional Hamming cube (see [ABR16, CDKS17, BC18, BGMV20, CCK$^+$21, CJLW21, BCY22]). With the rise of big data (translating to n being very large), even reading all the bits in the representation of a single sample might be very expensive. To address this issue, recently [GR22] studied distribution testing in a different setting.

In their model, called the *huge object model*, the distribution D is supported over the n-dimensional Hamming cube $\{0,1\}^n$, and the tester will obtain n-length Boolean strings as samples. However, as reading the sampled strings in their entirety might be infeasible when n is large, the authors in [GR22] considered query access to the samples along with standard sampling access. Note that without loss of generality, the number of samples will be upper-bounded by the number of queries. Thus, a desirable goal in this model is to optimize the number of queries for testing a given property. Restricting the algorithm to query access to the samples requires us to move from the Variation Distance measure to the Earth Mover Distance. A discussion of the reason for this, and of the features of this distance, is given below. [GR22] studied various natural properties like support size estimation, uniformity, identity, equality, and "grainedness" [1] in this model, providing upper and lower bounds on the sample and query complexities for these properties.

Index-Invariant Distribution Properties: In general, a distribution property is a collection of distributions over a fixed domain Ω [2]. Often the property in question has some other "symmetry". For example, a property is called *label-invariant* if any changes in the labels of the domain do not affect whether the distribution is in the property or not. Many of the well studied properties, such as uniformity, entropy estimation, support size estimation, and grainedness, are label-invariant properties. Label-invariant properties have been studied extensively in literature (see [BDKR05, Pan08, GR11, Val11, DKN14, CDVV14, ADK15, VV17, BC17, DKS18]).

In some cases, the distribution property is not fully label-invariant, but still has a certain amount of symmetry. For illustration, consider the following examples:

[1] A distribution D over $\{0,1\}^n$ is said to be m-grained if the probability mass of any element in its support is a multiple of $1/m$, where $m \in \mathbb{N}$.

[2] We use the phrases "a distribution is in the property" and "a distribution has the property" interchangeably to mean the same thing.

1. **Property** MONOTONE**:** Any distribution D over $\{0,1\}^n$ satisfies the MONOTONE property if

$$\mathbf{X} \preceq \mathbf{Y} \text{ implies } D(\mathbf{X}) \leq D(\mathbf{Y}), \text{ for any } \mathbf{X}, \mathbf{Y} \in \{0,1\}^n,$$

   where for two vectors $\mathbf{X}, \mathbf{Y} \in \{0,1\}^n$, $\mathbf{X} \preceq \mathbf{Y}$ if $x_i \leq y_i$ holds for every $i \in [n]$.

2. **Property** LOG-SUPER-MODULARITY**:** Any distribution D over $\{0,1\}^n$ satisfies the property LOG-SUPER- MODULARITY if

$$D(\mathbf{U})D(\mathbf{V}) \leq D(\mathbf{U} \wedge \mathbf{V})D(\mathbf{U} \vee \mathbf{V}), \text{ for any } \mathbf{U}, \mathbf{V} \in \{0,1\}^n,$$

   where the Boolean $\wedge$ and $\vee$ operations over the vectors are performed coordinate-wise.

3. **Property** LOW-VC-DIMENSION**:** For any $d \in \mathbb{N}$, a distribution D over $\{0,1\}^n$ is said to satisfy the LOW-VC-DIMENSION property, with parameter d, if the support of D has VC-dimension at most d.

Note that for the properties described above, a distribution satisfies the above properties even after the indices $\{1, \ldots, n\}$ of the vectors in $\{0,1\}^n$ are permuted by a permutation σ defined over $[n]$. To capture this structure, we introduce the notion of *index-invariant* properties.

Definition 1.1 (**Index-invariant property).** Let us assume that $D : \{0,1\}^n \to [0,1]$ is a distribution over the n-dimensional Hamming cube $\{0,1\}^n$. For any permutation $\sigma : [n] \to [n]$, let D_σ be the distribution such that $D(w_1, \ldots, w_n) = D_\sigma(w_{\sigma(1)}, \ldots, w_{\sigma(n)})$ for all $(w_1, \ldots, w_n) \in \{0,1\}^n$. A distribution property $\mathcal{P}$ is said to be *index-invariant* when D is in $\mathcal{P}$ if and only if D_σ is in $\mathcal{P}$, for any distribution D and any permutation σ.

Informally speaking, index-invariant properties refer to those properties that are invariant under the permutations of the indices $\{1, \ldots, n\}$. Note that this set of properties differs from the more common notion of label-invariant properties, since the total number of possible labels, for distributions over all n-length Boolean vectors, is 2^n. However, we are considering only permutations over $[n]$, thus in total only $n!$ permutations instead of $2^n!$ permutations. Index-invariant properties, while being able to have a richer structure as compared to label-invariance, allow for a thorough analysis with respect to learnability, as well as the gap between adaptive and non-adaptive testing.

1.1 Our Results

We will first study *testability* in the huge object model through the lens of index-invariance and bounded VC-dimension. Secondly, we study the gap between the query complexities of the adaptive and non-adaptive testers.

One important and technical difference between the huge object model and the standard distribution property testing model is the use of *Earth Mover Distance* (EMD) for the notion of "closeness" and "farness", instead of the more prevalent ℓ_1 or variation distance. Thus, in the rest of the paper, by an ε-tester for any property $\mathcal{P}$ of distributions over $\{0,1\}^n$, we mean an algorithm that given sample and query access (to the bits of the sampled vectors) to a distribution distinguishes (with probability at least 2/3) the case where the distribution D is in the property $\mathcal{P}$ from the case where the EMD of D from any distribution in $\mathcal{P}$ is at least ε, where $\varepsilon > 0$ is a proximity parameter.

Testable Properties: Testing by Learning, VC-dimension, and Index Invariance

We prove that a large class of distribution properties are all testable with a number of queries independent of n, using the *testing by learning paradigm* (see [DLM+07, GOS+09, Ser10]), where the distributions are supported over the n-dimensional Hamming cube $\{0,1\}^n$. More specifically, we prove that every distribution whose support has a bounded VC-dimension can be efficiently learnt up to a permutation, leading to efficient testers for index-invariant distribution properties that admit a global VC-dimension bound. Our main result regarding the learning of distributions in the huge object model is the following theorem.

Theorem 1.2 (**Main learning result (informal)**). *For any fixed constant $d \in \mathbb{N}$, given sample and query access to an unknown distribution D over $\{0,1\}^n$ and a proximity parameter $\varepsilon > 0$, there exists an algorithm that makes $\mathrm{poly}(\frac{1}{\varepsilon})$ queries [3], and either outputs the full description of a distribution or* FAIL *satisfying the following conditions:*

(i) If the support of D is of VC-dimension at most d, then with probability at least $2/3$, the algorithm outputs a full description of a distribution D' such that D is ε-close to D'_σ for some permutation $\sigma : [n] \to [n]$.

(ii) For any D, the algorithm will not output a distribution D' such that D'_σ is ε-far from D for all permutations $\sigma : [n] \to [n]$, with probability more than $1/3$. However, if the VC-dimension of the support of D is more than d, the algorithm may output FAIL *with any probability.*

In fact, our result holds for a general class of *clusterable* properties (stated in Theorem 4.2 and Corollary 5.3) that also covers the VC-dimension case as stated in the above theorem. Note that the above theorem corresponds to the learnability of any distribution when the VC-dimension of its support is bounded. As a corollary, it implies that any index-invariant distribution property admitting a global VC-dimension bound is testable with a constant number of queries, depending only on the proximity parameter ε and the VC-dimension d. The corollary is stated as follows:

Corollary 1.3 (**Testing (informal)**). *Let $\mathcal{P}$ be an index-invariant property such that any distribution $D \in \mathcal{P}$ has VC-dimension at most d, where d is some constant. There exists an algorithm, that has sample and query access to an unknown distribution D over $\{0,1\}^n$, takes a proximity parameter $\varepsilon > 0$, and distinguishes whether $D \in \mathcal{P}$ or D is ε-far from $\mathcal{P}$ with probability at least $2/3$, by making only $\mathrm{poly}(\frac{1}{\varepsilon})$ queries.*

It turns out that our tester for testing VC-dimension property takes $\exp(d)$ samples, and performs $\exp(\exp(d))$ queries for VC-dimension d. We show that this bound is tight, in the sense that there exists an index-invariant property with VC-dimension d such that any tester for the property requires an exponential number of samples and a doubly-exponential number of queries on d.

Theorem 1.4. *Let $d, n \in \mathbb{N}$. There exists an index-invariant property $\mathcal{P}_{\mathsf{vc}}$ with VC-dimension at most d such that any (non-adaptive) tester for $\mathcal{P}_{\mathsf{vc}}$ requires $2^{\Omega(d)}$ samples and $2^{2^{d-\mathcal{O}(1)}}$ queries.*

A natural question in this regard is whether the bounded VC-dimension and index-invariance assumptions are necessary for a property to be constantly testable, or just bounded VC-dimension is sufficient to guarantee testability. The following remark and proposition rule out respectively the possibility of only the bounded VC-dimension assumption, or only the index-invariance assumption, being sufficient for having an efficient tester.

[3] The degree of the polynomial in $\frac{1}{\varepsilon}$ depends on the parameter d.

[Index-invariance alone does not guarantee testability] From a result in [GR22], it follows that there exists an index-invariant property $\mathcal{P}$ such that any distribution $D \in \mathcal{P}$ has VC-dimension d and any algorithm that has sample access to a distribution D over $\{0,1\}^n$ requires $\Omega(2^d/d)$ samples [4]. Note that this essentially shows that index-invariance alone, without any bound on VC-dimension, can not guarantee testability.

Proposition 1.5 (**Necessity of index-invariance (informal)**). *There exists a non-index-invariant property $\mathcal{P}$ such that any distribution $D \in \mathcal{P}$ has VC-dimension $O(1)$ and the following holds. There exists a fixed $\varepsilon > 0$, such that distinguishing whether $D \in \mathcal{P}$ or D is ε-far from $\mathcal{P}$ requires $\Omega(n)$ queries, where the distributions in the property $\mathcal{P}$ are defined over the n-dimensional Hamming cube $\{0,1\}^n$.*

The above proposition is formally stated and proved at the end of Subsection 7.2.

Separation between adaptive and non-adaptive testers

Until now, all the upper bounds that we have discussed are designed for non-adaptive algorithms. The question how adaptivity helps in designing efficient testers is interesting in its own right. In the standard model of distribution testing, since the model is inherently non-adaptive, there is essentially no gap between adaptive and non-adaptive testers. However, in the related model of conditional sampling of distributions (see [CFGM16, CRS15]), there is a super-exponential separation (constant vs. $\mathrm{poly}(\log n)$) between the complexities of these two types of testers ([ACK18]).

In the context of graph testing in the dense graph model, it is known that the gap between the query complexities of adaptive and non-adaptive algorithms is at most quadratic (see [GT03]), which has recently been proved to be tight (see [GW21]). However, for bounded-degree graphs, the gap between the query complexities for some properties is constant vs. $\Omega(\sqrt{n})$, where n denotes the number of vertices of the graph (see [GR97]). For testing of Boolean strings, there is an exponential separation between the complexity of these two types of testers (see [RS15]).

Thus, a natural question to study in the huge object model is the gap between the query complexities of non-adaptive and adaptive algorithms. When considering general properties, we show that there can be an exponential gap in the query complexities between non-adaptive and adaptive testers (see Theorem 7.6).

Theorem 1.6. (Exponential gap between adaptive and non-adaptive testers for general properties) *There exists a distribution property for which there is an exponential gap in the query complexities of non-adaptive and adaptive testers.*

However, for index-invariant properties, this gap can be at most quadratic, as stated in the following theorem.

Theorem 1.7 (Adapt. vs. non-adapt. testers for index-invariant properties: upper bound)**.** *For any index-invariant property $\mathcal{P}$, there is at most a quadratic gap between the query complexities of adaptive and non-adaptive testers.*

We also prove that the above gap is almost tight, in the sense that there exists an index-invariant property which can be ε-tested using $\widetilde{\mathcal{O}}(n)$ adaptive queries, while $\widetilde{\Omega}(n^2)$ non-adaptive queries are required to ε-test it.

[4] Let $\mathcal{P}$ be the distribution property of having support size at most 2^d. Note that the VC-dimension of any member of $\mathcal{P}$ is at most d. By [GR22], for any small enough ε, an ε-test for this property requires at least $\Omega\left(2^d/d\right)$ samples.

Theorem 1.8. (Adaptive vs. non-adaptive testers for index-invariant properties: lower bound) *There exists an index-invariant property $\mathcal{P}_{\text{Gap}}$ that can be ε-tested adaptively using $\widetilde{\mathcal{O}}(n)$ queries for any $\varepsilon \in (0,1)$, while there exists an $\varepsilon \in (0,1)$ for which $\widetilde{\Omega}(n^2)$ queries are necessary for any non-adaptive ε-tester.*

Using EMD as the distance metric in conjunction with the notion of index-invariance

Recall that here we will use the Earth Mover Distance (EMD) as the distance metric defining ε-testing, in contrast to the stronger variation distance, the commonly studied distance measure in distribution testing literature. As discussed in [GR22], this is essential when we restrict ourselves to querying the samples obtained from the distribution. To illustrate this, consider two (say very sparse) distributions D_1 and D_2 whose supports are disjoint, yet admit a bijection such that every string from $\text{Supp}(D_1)$ is mapped to a string from $\text{Supp}(D_2)$ that is very close to it in terms of the Hamming distance. The variation distance between D_1 and D_2 would be large, and yet we would not be able to distinguish the two distributions without querying some samples in their entirety, that is, without using $\Theta(n)$ queries per sample. The EMD metric is the one incorporating the Hamming distance between strings (which comes to play when we are not performing many queries to the samples) into the notion of variation distance.

Another question involves what general statements can be said about testers in this model. If we do not restrict ourselves to properties satisfying any sort of invariance, then very little can be proved on testers in general, just as is the case with general string property testing under the Hamming distance (in fact, string testing can be reduced to testing in the huge object model [5]). On the other hand, if we were to restrict ourselves to label-invariant properties only, it would appear that we lose much of the rich structure offered by the ability to define distributions over strings. We believe that index-invariance is a natural middle-of-the-road restriction for the formulation of general statements about testing in the huge object model.

1.2 Organization of the paper

In Section **??** and Section **??**, we present brief overviews of our results on distribution learning and the relations between the query complexities of testers in the adaptive and non-adaptive models. We present the related definitions in the preliminaries section (Section 3). We present the results about learning and testing clusterable distributions in Section 4. After that, in Section 5, we move on to present algorithms for testing properties with bounded VC-dimension. We present lower bound results for bounded VC-dimension testing in Section 6. Later, we show the tight exponential separation between the query complexities of adaptive and non-adaptive algorithms for non-index-invariant properties in Section 7. In Section 8.1, we prove that for index-invariant properties, there is at most a quadratic gap between the query complexities of adaptive and non-adaptive testers. Finally, in Section 8.2, Section 8.3, Section 8.4, and in Section 8.5, we prove that the quadratic gap between adaptive and non-adaptive testers for index-invariant properties is almost tight, ignoring poly-logarithmic factors. In Appendix A, we state some useful concentration inequalities that are used in our proofs.

Due to lack of space, we move Section 3 to Section A to the Appendix and only the introduction, description of our results and overview of the proofs of our results are presented in the main body of the paper.

[5] We will use this reduction for proving exponential separation between adaptive and non-adaptive testers for non-index-invariant properties (see Subsection 7.2).

2 Technical overview of our results

In this section, we provide a brief overview of our results. We start by explaining our upper bounds.

2.1 Overview of our upper bound results for index-invariant bounded VC-dimension property

In our main upper bound result, we prove a learning result for a general class of distributions that covers the case of learning distributions with bounded VC-dimension. We say that a distribution D is (ζ, δ, r)-clusterable if we can partition the n-dimensional Hamming cube $\{0,1\}^n$ into $r+1$ parts $\mathcal{C}_0, \ldots, \mathcal{C}_r$, such that $D(\mathcal{C}_0) \leq \zeta$ and the diameter of $\mathcal{C}_i$ is at most δ for every $i \in [r]$ (see Definition 4.1). The main upper bound result (Theorem 4.2), that leads to Theorem 1.2, is the design of an algorithm for learning a (ζ, δ, r)-clusterable distribution up to permutations. That is, given sample and query access to a (ζ, δ, r)-clusterable distribution, we want to output a distribution D' such that the Earth Mover Distance between D and D'_σ is small for some permutation $\sigma : [n] \to [n]$, by performing number of queries independent of n.

The idea of learning (ζ, δ, r)-clusterable distributions: The formal algorithm is presented in Algorithm **??** in Section 4 as Test-and-Learn. The algorithm starts by taking $t_1 = \mathcal{O}(\frac{r}{\zeta} \log \frac{r}{\zeta})$ samples from the input distribution D. Let us denote them as $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$. If D is (ζ, δ, r)-clusterable, consider its clusters $\mathcal{C}_0, \ldots, \mathcal{C}_r$ as described above. We say that a cluster C_i is *large* if the probability mass of $\mathcal{C}_i$ is more than $\frac{\zeta}{10r}$, that is, $D(\mathcal{C}_i) \geq \frac{\zeta}{10r}$. As the size of $\mathcal{S}$ is sufficiently large, we know that $\mathcal{S}$ intersects every large cluster with probability at least 99/100 (see Lemma 4.5). In order to estimate the masses of $\mathcal{C}_i$, for each $i \in [t_1]$, we take another set of random samples $\mathcal{T} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_{t_2}\}$ from D where $t_2 = \mathcal{O}(\frac{t_1^2}{\zeta^2} \log t_1)$, and assign each of the vectors in $\mathcal{T}$ to some vector in $\mathcal{S}$ depending on their Hamming distance. However, since computing the exact distances between the vectors in $\mathcal{S}$ and $\mathcal{T}$ requires $\Omega(n)$ queries, we use random sampling.

We take a random set of indices $R \subset [n]$ of suitable size, and project the vectors in $\mathcal{S}$ and $\mathcal{T}$ on R to estimate their pairwise distances up to an additive factor of δ. R not only preserves the distances between all pairs of vectors between $\mathcal{S}$ and $\mathcal{T}$, but also the distances of a large fraction of the vectors in $\{0,1\}^n$ from all the vectors in $\mathcal{S}$ (see Lemma 4.6). Based on the estimated distances, we assign each vector of $\mathbf{T} \in \mathcal{T}$ to a vector in $\mathbf{S} \in \mathcal{S}$ such that the projected distance between them is at most 2δ. If there exists no such vector in $\mathcal{S}$ corresponding to a vector $\mathbf{T} \in \mathcal{T}$, then the vector $\mathbf{T}$ remains unassigned. Let us denote the fraction of vectors in $\mathcal{T}$ that are assigned to $\mathbf{X}_i$ as w_i, for every $i \in [t_1]$. Let w_0 be the fraction of vectors in $\mathcal{T}$ that are not assigned to any vector in $\mathcal{S}$. If D is (ζ, δ, r)-clusterable, then $w_0 \leq 3\zeta$ holds with high probability. These w_i's preserve the weights of some approximate clustering (which may not be the original one from which we started, but is close to it in some sense), see Lemma 4.7 for the details.

Consider a distribution D^* supported over $\mathcal{S}$ such that $D(\mathbf{X}_i) \geq w_i$ for every $i \in [t_1]$. Using a number of technical lemmas, we prove that the EMD between D and D^* is *small*. Note that we still can not report D^* as the output distribution, since to do so, we need to know the exact vectors in $\mathcal{S}$, which requires $\Omega(n)$ queries. To bypass this barrier, we use the provision that we are allowed to output any permutation of the distribution. More specifically, we construct vectors $\mathbf{S}_1, \ldots, \mathbf{S}_{t_1} \in \{0,1\}^n$ such that $d_H(\mathbf{X}_i, \sigma(\mathbf{S}_i))$ is small for every $i \in [t_1]$ and some permutation $\sigma : [n] \to [n]$. This is possible using the projections of the vectors in $\mathcal{S}$ to the random set of indices R for estimating the number of indices of each "type" with respect to $\mathcal{S}$ (see Lemma 4.11). Finally, we output the distribution D' supported over the newly constructed vectors $\mathbf{S}_1, \ldots, \mathbf{S}_{t_1}$ such that

$D'(\mathbf{S}_i) = D^*(\mathbf{X}_i)$ for every $i \in [t_1]$. The guarantee on the Hamming distance between $\mathbf{X}_i$ and $\sigma(\mathbf{S}_i)$ provides a bound on the EMD between D'_σ and D^*, and with the above mentioned EMD bound between D^* and D, we are done. To keep the discussion simple, we will not explain here the idea of the proof of Theorem 1.2(ii), which relies on a sort of converse to the above method of approximating cluster weights.

How learning (ζ, δ, r)-clusterable distribution implies Theorem 1.2: Let us define a distribution to be (α, r)-clusterable if it is $(0, \alpha, r)$-clusterable. The learning of (ζ, δ, r)-clusterable distribution implies a learning result for any distribution that is close to being (α, r)-clusterable (see Corollary 5.3) due to a technical lemma (see Lemma 5.4). If the support of a distribution has bounded VC-dimension, using standard results in VC theory, we can show that it is also (α, r)-clusterable, where r is a function of α and d. Thus the learning result of (α, r)-clusterable distributions implies a result allowing the learning of distributions with bounded VC-dimension.

2.2 Overview of the lower bound result for index-invariant bounded VC-dimension properties

To prove Theorem 1.4, let us define the property $\mathcal{P}_{\mathsf{vc}}$. Let $k = 2^d$, $\ell = 2^{2^{d-10}}$ and $\ell' = 2^{2^{d-20}}$. Consider a matrix A of dimension $k \times \ell$ whose column vectors are 1/3-far from each other. Let $\mathbf{V}_1, \ldots, \mathbf{V}_k \in \{0,1\}^n$ be k vectors that are formed by blowing up the row vectors of A in $\{0,1\}^\ell$ to $\{0,1\}^n$ by repeating each bit of the vectors n/ℓ times, and D_A be the uniform distribution over the support $\{\mathbf{V}_1, \ldots, \mathbf{V}_k\}$. Our property $\mathcal{P}_{\mathsf{vc}}$ is the collection of all distribution that can be obtained from D_A by permuting the indices. Let D_{yes} be the distribution obtained from D_A by randomly permuting the indices. Note that $D_{yes} \in \mathcal{P}_{\mathsf{vc}}$. As the support size of any distribution in $\mathcal{P}_{\mathsf{vc}}$ is at most 2^d, the VC-dimension of $\mathcal{P}_{\mathsf{vc}}$ is at most d.

To prove the lower bound on the query complexity, let us define the distribution D_{no}. Let us take ℓ' columns of A uniformly at random to form a matrix B of dimension $k \times \ell'$, and $\mathbf{W}_1, \ldots, \mathbf{W}_k \in \{0,1\}^n$ be k vectors that are formed by blowing up the row vectors of B in $\{0,1\}^{\ell'}$ to $\{0,1\}^n$ by repeating each bit of the vectors n/ℓ' times. Let D_B be the uniform distribution over the support $\{\mathbf{W}'_1, \ldots, \mathbf{W}'_k\}$. D_{no} is the distribution obtained from D_B by randomly permuting the indices. We show that the Earth Mover Distance between D_{no} and any distribution in $\mathcal{P}_{\mathsf{vc}}$ is at least 1/8 (see Lemma 6.3). Observe that D_{yes} divides the index set $[n]$ into ℓ equivalence classes and D_{no} divides the index set into ℓ' equivalence classes. The query complexity lower bound follows from the fact that, unless we query $2^{2^{d-\mathcal{O}(1)}}$ indices, we do not hit two indices from the same equivalence class, irrespective of whether the distribution is D_{yes} or D_{no} (see Lemma 6.8).

To prove the lower bound on the sample complexity, let us define another distribution D'_{no}. Let us take $k' = 2^{d-20}$ rows of A uniformly at random to form a matrix B' of dimension $k' \times \ell$. Let $\mathbf{W}'_1, \ldots, \mathbf{W}'_{k'} \in \{0,1\}^n$ be k' vectors that are formed by blowing up the row vectors of B' in $\{0,1\}^\ell$ to $\{0,1\}^n$ by repeating each bit of the vectors n/ℓ times. Let $D_{B'}$ be the uniform distribution with support $\{\mathbf{W}'_1, \ldots, \mathbf{W}'_{k'}\}$. D'_{no} is the distribution obtained from $D_{B'}$ by randomly permuting the indices. We show that the Earth Mover Distance between D'_{no} and any distribution in $\mathcal{P}_{\mathsf{vc}}$ is at least 1/8 (see Lemma 6.9). The sample complexity lower bound follows from the fact that, unless we take $2^{\Omega(d)}$ samples, all the samples are distinct with probability $1 - o(1)$, irrespective of whether the distribution is D_{yes} or D_{no} (see Lemma 6.11).

2.3 Overview of adaptive vs non-adaptive query complexity results

Finally, we explore the relationship between adaptive and non-adaptive testers in the huge object model. It turns out that there is a tight (easy to prove) exponential separation between the query complexities of adaptive and non-adaptive testers for non-index-invariant properties. Roughly, the simulation of an adaptive algorithm by a non-adaptive one follows from unrolling the decision tree of the adaptive algorithm. This is formally proved in Lemma 7.2. Moreover, we show that this separation is tight. For this purpose, we consider a property of strings $\mathcal{P}_{Pal}$, which exhibits an exponential gap between adaptive and non-adaptive testing in the string testing model. We show how to transform a string property $\mathcal{P}$ to a distribution property $1_{\mathcal{P}}$ such that the query bounds on adaptive and non-adaptive testing carry over. Thus, the separation result between adaptive and non-adaptive algorithms for $\mathcal{P}_{Pal}$ carries over to $1_{\mathcal{P}_{Pal}}$ (see Theorem 7.6). This technique, employed for a maximally hard to test string property, is also used for proving Proposition 1.5.

In contrast to the non-index-invariant properties, we prove that there can be at most a quadratic gap between the query complexities of adaptive and non-adaptive algorithms for testing index-invariant properties. The proof is very close in spirit to the proof of the quadratic relation between adaptive and non-adaptive testing of graphs in the dense model [GT03]. Given an adaptive algorithm $\mathcal{A}$ with sample complexity s and query complexity q, the main idea is to first simulate a *semi-adaptive* algorithm $\mathcal{A}'$ that queries q indices from each of the s samples and decides accordingly. Note that the sample complexity of $\mathcal{A}'$ remains s, whereas the query complexity becomes qs. Once we have the semi-adaptive algorithm $\mathcal{A}'$, we now simulate a non-adaptive algorithm $\mathcal{A}''$. As the property we are testing is index-invariant, we can first apply a uniformly random permutation σ over $[n]$, and then run the semi-adaptive algorithm $\mathcal{A}'$ over D_σ instead of D, where D is the input distribution to be tested. This makes the tester completely non-adaptive. Its correctness follows from the index-invariance of the property we are testing.

Quadratic separation between adaptive and non-adaptive testers: Before proceeding to present an overview of our quadratic separation result, let us first recall the support estimation result of [VV10], which will be crucially used in our proof. Roughly speaking, the result states that in the standard sampling model, given a distribution D over $[2n]$, in order to distinguish whether D has support size at most n, or D is far from all distributions with support size at least n, $\Theta(n/\log n)$ samples are required.

Theorem 2.1 (Support Estimation: Restatement of Corollary 9 of [VV10])**.** *Given a distribution D over $[2n]$, that can be accessed via independent samples and a proximity parameter $\varepsilon \in (0, 1/8)$, in order to distinguish, with probability at least $\frac{3}{4}$, whether D has support size at most n or D has at least $(1+\varepsilon)n$ elements with non-negligible weights in its support, $\Theta(\frac{n}{\log n})$ samples from D are necessary and sufficient.*

To construct the index-invariant property that provides a quadratic separation between the query complexities of adaptive and non-adaptive testers, we will use the above result. Let D^{Supp}_{yes} and D^{Supp}_{no} be the pair of hard distributions corresponding to the support estimation lower bound, from which we define our pair D_{yes} and D_{no} of hard distributions for our property. We will construct a huge object distribution property over a slightly larger domain $[N]$ with $N = \mathcal{O}(n \log n)$, where we will encode the elements of the support of the distributions D^{Supp}_{yes} and D^{Supp}_{no}. Additionally, we will include a set of "ordering" vectors to both D_{yes} and D_{no} that encode a permutation $\sigma : [n] \to [n]$. Our property will be defined as a permutation of a non-index-invariant property along with an encoding of the permutation itself.

For the non-index-invariant property, we use an encoding of the elements of $\{0,1\}^n$ that can be decoded only if a sample from a family of special small sets is read in its entirety. For constructing hard distributions, we consider (encodings of) $2n$ special elements of $\{0,1\}^n$, and use over them the hard distributions corresponding to Theorem 2.1.

The encoding vectors of D_{yes} and D_{no} are designed in such a fashion, that if we can know the index ordering (and thus the identity of the above mentioned small sets), the support size estimation problem becomes relatively easy. However, without knowing the ordering vectors, estimating the size of the support becomes harder. More specifically, if we already know the index ordering, then support size estimation can be done using $\text{poly}(\log n)$ queries from each sample, over the $\widetilde{\mathcal{O}}(n)$ samples that are sufficient for solving the support estimation problem.

On the other hand, an important feature of our property ensures that unless some of the special sets are successfully hit while querying a sampled vector, which is a low probability event without prior knowledge of the encoded index ordering unless we perform $\widetilde{\Omega}(n)$ queries to that vector, then the queries do not provide any useful information about the sampled vector to the tester. This is achieved by the encoding procedure of the vectors, which is motivated from [BFLR20]. However, it is not deployed here the same way as [BFLR20], since the surrounding proofs here are quite different (as well as the end-goal).

Since an adaptive algorithm can first learn the ordering vectors by performing $\widetilde{\mathcal{O}}(n)$ queries (as it takes $\text{poly}(\log n)$ samples to hit all the order encoding vectors), the adaptive tester requires $\widetilde{\mathcal{O}}(n)$ queries in total. However, for non-adaptive testers, since we have to perform all queries simultaneously, the tester would have to make $\widetilde{\Omega}(n)$ queries to each sampled vector to be able to utilize the support estimation procedure (since as explained above, fewer queries would give no useful information about the sample to the tester). As a result, $\widetilde{\Omega}(n^2)$ non-adaptive queries are required following the lower bound result in Theorem 2.1.

Another technical challenge is to construct the property in such a fashion that allows the crafting of "wrong distributions" which remain far from the property, even if we permute the support vectors. This is due to the fact that just replacing the vectors defining the index ordering does not require a change of large Earth Mover Distance. Thus we need the distributions to remain far from the property even if we reorder them. We ensure this by designing the hard distributions such that the support vectors of the distributions are far from each other. This in turn allows us to prove that the distribution D_{no} will remain far from the property, as the size of its support is too large. The arguments involving only the mutual Hamming distance between the vectors in the support and the size of the support are invariant with respect to the index ordering, and are thus not affected by the possibility of "cheaply" changing the index ordering vectors.

3 Preliminaries

For an integer n, we will denote the set $\{1,\ldots,n\}$ as $[n]$. Given two vectors $\mathbf{X}$ and $\mathbf{Y}$ in $\{0,1\}^n$, we will denote by $d_H(\mathbf{X},\mathbf{Y})$ the normalized Hamming distance between $\mathbf{X}$ and $\mathbf{Y}$, that is,

$$d_H(\mathbf{X},\mathbf{Y}) := \frac{|\{i \in [n] \,:\, \mathbf{X}_i \neq \mathbf{Y}_i\}|}{n}.$$

Unless stated otherwise, all the distance measures that we will be considering in this paper will be the normalized distances. For two vectors $\mathbf{X},\mathbf{Y} \in \{0,1\}^n$, $\delta_H(\mathbf{X},\mathbf{Y}) = n \cdot d_H(\mathbf{X},\mathbf{Y})$ will be used to denote the absolute Hamming distance between $\mathbf{X}$ and $\mathbf{Y}$ in the few places where we will need to refer to it. When we write $\widetilde{\mathcal{O}}(\cdot)$, it suppresses a poly-logarithmic term in n and the inverse of the proximity parameter.

We will also need the following observation from [ABEF17] which roughly states that given a sequence of non-negative real numbers that sum up to an integer n, there is a procedure that by choosing the floor or ceiling of these real numbers, one can obtain another sequence of integers that sum up to n. This observation will be used in our proof.

Observation 3.1 (**Restatement of [ABEF17, Lemma 4.8]).** Let $T, n \in \mathbb{N}$. Given T non-negative real numbers $\alpha_1, \ldots, \alpha_T$ such that $\sum_{i=1}^{T} \alpha_i = n$, there exists a procedure of choosing T integers $\beta_1, \ldots, \beta_T$ such that $\beta_i \in \{\lfloor \alpha_i \rfloor, \lceil \alpha_i \rceil\}$ for every $i \in [T]$ and $\sum_{i=1}^{T} \beta_i = n$.

3.1 Definitions and relations of various distance measures of distributions

We will first define ℓ_1 distance between two distributions.

Definition 3.2 (ℓ_1 **distance and variation distance between two distributions).** Let D_1 and D_2 be two probability distributions over a set S. The ℓ_1 distance between D_1 and D_2 is defined as

$$||D_1 - D_2||_1 = \sum_{a \in S} |D_1(a) - D_2(a)|.$$

The variation distance between D_1 and D_2 is defined as:

$$d_{TV}(D_1, D_2) = \frac{1}{2} \cdot ||D_1 - D_2||_1.$$

Throughout this paper, the Earth Mover Distance (EMD) is the central metric for testing "closeness" and "farness" of a distribution from a given property. It is formally defined below.

Definition 3.3 (**Earth Mover Distance (**EMD**)).** Let D_1 and D_2 be two probability distributions over $\{0,1\}^n$. The EMD between D_1 and D_2 is denoted by $d_{EM}(D_1, D_2)$, and defined as the solution to the following linear program:

$$\begin{aligned}
\text{Minimize} \quad & \sum_{\mathbf{X},\mathbf{Y} \in \{0,1\}^n} f_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) & \\
\text{Subject to} \quad & \sum_{\mathbf{Y} \in \{0,1\}^n} f_{\mathbf{XY}} = D_1(\mathbf{X}), & \forall\ \mathbf{X} \in \{0,1\}^n \\
& \sum_{\mathbf{X} \in \{0,1\}^n} f_{\mathbf{XY}} = D_2(\mathbf{Y}), & \forall\ \mathbf{Y} \in \{0,1\}^n \\
& 0 \leq f_{\mathbf{XY}} \leq 1, & \forall\, \mathbf{X}, \mathbf{Y} \in \{0,1\}^n
\end{aligned}$$

Intuitively, the variable $f_{\mathbf{XY}}$ stands for the amount of probability mass transferred from $\mathbf{X}$ to $\mathbf{Y}$.

Directly from the definitions of $d_{EM}(D_1, D_2)$ and $d_H(\mathbf{X}, \mathbf{Y})$, we get the following simple yet useful observation connecting ℓ_1 distance and EMD between two distributions.

Observation 3.4 (**Relation between EMD and** ℓ_1 **distance).** Let D_1 and D_2 be two distributions over the n-dimensional Hamming cube $\{0,1\}^n$. Then we have the following relation between the Earth Mover Distance and ℓ_1 distance between D_1 and D_2:

$$d_{EM}(D_1, D_2) \leq \frac{||D_1 - D_2||_1}{2}.$$

Now we formally define the notions of "closeness" and "farness" of two distributions with respect to the Earth Mover Distance.

Definition 3.5 (**Closeness and farness with respect to** EMD)**.** Given two proximity parameters ε_1 and ε_2 with $0 \leq \varepsilon_1 < \varepsilon_2 \leq 1$, two distributions D_1 and D_2 over the n-dimensional Hamming cube $\{0,1\}^n$ are said to be ε_1-*close* if $d_{EM}(D_1, D_2) \leq \varepsilon_1$, and ε_2-*far* if $d_{EM}(D_1, D_2) \geq \varepsilon_2$.

Now we proceed to define the notion of distribution properties over the Hamming cube below.

Definition 3.6 (**Distribution property over the Hamming cube).** Let $\mathcal{D}$ denote the set of all distributions over the n-dimensional Hamming cube $\{0,1\}^n$. A *distribution property* $\mathcal{P}$ is a topologically closed subset of $\mathcal{D}$. [6] A distribution $D \in \mathcal{P}$ is said to be *in the property* or to *satisfy* the property. Any other distribution is said to be *not in the property* or *to not satisfy* the property.

Now we are now ready to define the notion of distance of a distribution from a property.

Definition 3.7 (**Distance of a distribution from a property).** The distance of a distribution D from a property $\mathcal{P}$ is the minimum Earth Mover Distance between D and any distribution in $\mathcal{P}$. [7] For $\varepsilon \in [0,1]$, a distribution D is said to be ε-*close* to $\mathcal{P}$ if the distance of D from $\mathcal{P}$ is at most ε. Analogously, for $\varepsilon \in [0,1]$, a distribution D is said to be ε-*far* from $\mathcal{P}$ if the distance of D from $\mathcal{P}$ is more than ε.

3.2 Formal definitions of various kinds of property testers

First we define our query model below.

Definition 3.8 (**Query to sampled vectors).** Let $\mathcal{A}$ be a tester with a set of sampled vectors $\mathbf{V}_1, \ldots, \mathbf{V}_s$, drawn independently from an input distribution D over $\{0,1\}^n$, where $\mathbf{V}_i = (v_{i,1}, \ldots, v_{i,n})$ for every $i \in [s]$. In order to perform a query, the tester will provide i and j, and will receive $v_{i,j}$ as the answer to the query.

In the following, we formally describe the notion of a tester.

Definition 3.9 (ε**-test).** Let $\varepsilon \in (0,1)$ be a proximity parameter, and $\delta \in (0,1)$. A probabilistic algorithm $\mathcal{A}$ is said to ε-*test* a property $\mathcal{P}$ with probability at least $1-\delta$, if any input in $\mathcal{P}$ is accepted by $\mathcal{A}$ with probability at least $1-\delta$, and any input that is ε-far from $\mathcal{P}$ is rejected by $\mathcal{A}$ with probability at least $1-\delta$. Unless explicitly stated, we assume that $\delta = 1/3$.

Now we define two different types of testers, *adaptive* testers and *non-adaptive* testers, which will be used throughout the paper. We begin by describing the adaptive testers. Informally, adaptive testers correspond to algorithms that perform queries depending on the answers to previous queries. Formally:

Definition 3.10 (**Adaptive tester).** Let $\mathcal{P}$ be a property over $\{0,1\}^+ +$. An adaptive tester for $\mathcal{P}$ with query complexity q and sample complexity s is a randomized algorithm $\mathcal{A}$ that ε-tests $\mathcal{P}$ by performing the following:

- $\mathcal{A}$ first draws some random coins and samples s vectors from the unknown distribution D, denoted by $S = \{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$.

[6] We put this restriction to avoid formalism issues. In particular, the investigated distribution properties that we know of (such as monotonicity and being a k-histogram) are topologically closed.

[7] The assumption that $\mathcal{P}$ is closed indeed makes it a minimum rather than an infimum.

- $\mathcal{A}$ then queries the j_1-th index of $\mathbf{V}_{i_1}$, for some $j_1 \in [n]$ and $i_1 \in [s]$ depending on the random coins.
- Suppose that $\mathcal{A}$ has executed k steps and has queried the j_ℓ-th index of $\mathbf{V}_{j_\ell}$, where $1 \leq \ell \leq k$. At the $(k+1)$-th step, depending upon the random coins and the answers to the queries till the k-th step, $\mathcal{A}$ will perform a query for the j_{k+1}-th bit of $\mathbf{V}_{i_{k+1}}$, where $j_{k+1} \in [n]$ and $i_{k+1} \in S$.
- After q steps, $\mathcal{A}$ reports ACCEPT or REJECT depending on the random coins and the answers to all q queries.

Now we define the more restricted *non-adaptive testers*. Informally, non-adaptive testers decide the set of queries to be performed on the input even before performing the first query. Formally:

Definition 3.11 (**Non-adaptive tester).** Let $\mathcal{P}$ be a property over $\{0,1\}^n$. A non-adaptive tester for $\mathcal{P}$ with query complexity q and sample complexity s is a randomized algorithm $\mathcal{A}$ that ε-tests $\mathcal{P}$ by performing the following:

- $\mathcal{A}$ tosses some random coins, and depending on the answers constructs a sequence of subsets of indices $J_1, \ldots, J_s \subset [n]$ such that $\sum_{i=1}^{s} J_i \leq q$.
- $\mathcal{A}$ takes s samples $\mathbf{V}_1, \ldots, \mathbf{V}_s$ from the unknown distribution D.
- $\mathcal{A}$ queries for the coordinates of $\mathbf{V}_i$ that are in J_i, for each $i \in [s]$.
- $\mathcal{A}$ reports either ACCEPT or REJECT based on the answers from the queries to the vectors, that is, $\mathbf{V}_1 \mid_{J_1}, \mathbf{V}_2 \mid_{J_2}, \ldots, \mathbf{V}_s \mid_{J_s}$, and the random coins.

3.3 Distributions and properties with bounded VC-dimension

Now we move on to define a class of properties using the notion of the VC-dimension of the support of a distribution. Before proceeding to define the class of properties, let us recall the notions of *shattering* and *VC-dimension*.

Let V be a collection of vectors from $\{0,1\}^n$. For a sequence of indices $I = (i_1, \ldots, i_k)$, with $1 \leq i_j \leq n$, let $V \mid_I$ denote the set of *projections* of V onto I, that is,

$$V \mid_I = \{(v_{i_1}, \ldots, v_{i_k}) : (v_1, \ldots, v_n) \in V\}.$$

If $V \mid_I = \{0,1\}^k$, then it is said that V *shatters* the index sequence I. The *VC-dimension* of V is the size of the largest index sequence I that is shattered by V. VC-dimension was introduced by [VC15] in the context of learning theory, and has found numerous applications in other areas like approximation algorithms, discrete and computational geometry, discrepancy theory, see [Mat99, PA95, Mat02, Cha00].

We now give a natural extension of VC-dimension to distributions.

Definition 3.12 (**Distribution with VC-dimension** d**).** Let d, $n \in \mathbb{N}$ and D be a distribution over $\{0,1\}^n$. We say that D has VC-dimension at most d if the support of D has VC-dimension at most d. A distribution D is said to be β-*close to VC-dimension* d if there exists a distribution D_0 with VC-dimension d such that $d_{EM}(D, D_0) \leq \beta$, where $\beta \in (0,1)$.

Analogously, we can also define the notion of a (β, d)-VC-dimension property.

Definition 3.13 (**(β, d)-VC-dimension property).** Let $d, n \in \mathbb{N}$ and $\beta \in (0,1)$. A property $\mathcal{P}$ over $\{0,1\}^n$ is said to be a (β, d)-*VC-dimension property* if for any distribution $D \in \mathcal{P}$, D is β-close to VC-dimension d. When $\beta = 0$, we say that the VC-dimension of $\mathcal{P}$ is d. We also say that a $(0, d)$-VC-dimension property is a *bounded VC-dimension property.*

We now give examples of bounded VC-dimension properties.

Property CHAIN**:** For any distribution $D \in$ CHAIN, the support of D can be written as a sequence $\mathbf{X}_1, \ldots, \mathbf{X}_t \in \{0,1\}^n$ such that any two vectors with non-zero probability are comparable, that is,

$$D(\mathbf{X}_i) > 0 \text{ and } D(\mathbf{X}_j) > 0 \text{ implies either } \mathbf{X}_i \preceq \mathbf{X}_j \text{ or } \mathbf{X}_j \preceq \mathbf{X}_i, \text{ for every } i, j \in [t].$$

Property LOW-AFFINE-DIMENSION**:** A distribution D over $\{0,1\}^n$ is said to satisfy the LOW-AFFINE-DIMENSION property, with parameter $d \in \mathbb{N}$, if the *affine dimension*[8] of the support of D is at most d.

Observe that the VC-dimension of CHAIN is 1, and the VC-dimension of LOW-AFFINE-DIMENSION is d. [9] Moreover, note that both CHAIN and LOW-AFFINE-DIMENSION are examples of index-invariant properties.

3.4 Yao's lemma for the huge object model

Our lower bound proofs crucially use Yao's lemma ([Yao77]). Informally, it states that for any two distributions D_1 and D_2 such that D_1 satisfies some property, and D_2 is far from the property, if the variation distance between D_1 and D_2 with respect to q queries is small, then D_1 and D_2 remain indistinguishable with respect to q queries. In order to formally state the lemma, we need the following definitions.

Definition 3.14 (**Restriction).** Let D be a distribution over a collection of functions $f : \mathcal{D} \to \{0,1\}$, and Q be a subset of the domain $\mathcal{D}$ of D. The restriction $D\,|_Q$ of D to Q is the distribution over functions of the form $g : Q \to \{0,1\}$, which is obtained from choosing a random function $f : \mathcal{D} \to \{0,1\}$ according to the distribution D, and then setting $g = f\,|_Q$, where $f\,|_Q$ denotes the restriction of f to Q.

The following is the version of Yao's Lemma which is used for non-adaptive testers in the classical setting. The crucial observation that makes this lemma work is the observation that the deterministic version of a non-adaptive tester in the classical setting is characterized by a set of possible responses to a fixed query set $Q \subset \mathcal{D}$.

Lemma 3.15 (**Yao's lemma for non-adaptive testers, see [Fis04]).** *Let $\varepsilon \in (0,1)$ be a parameter and $q \in \mathbb{N}$ be an integer. Suppose there exists a distribution D_{yes} on inputs over $\mathcal{D}$ that satisfy a given property $\mathcal{P}$, and a distribution D_{no} on inputs that are ε-far from satisfying the property. Moreover, assume that for any set of queries $Q \subset \mathcal{D}$ of size q, the variation distance between $D_{yes}\,|_Q$ and $D_{no}\,|_Q$ is less than $\frac{1}{3}$. Then it is not possible for a non-adaptive tester performing q (or less) queries to ε-test $\mathcal{P}$.*

[8] A set $S \subseteq \mathbb{R}^n$ has *affine dimension* k if the dimension of the smallest *affine set* in $\mathbb{R}^n$ that contains S is k.

[9] In fact, the property LOW-AFFINE-DIMENSION is a sub-property of "support size is at most 2^d", which has VC-dimension d.

In this paper, we will prove lower bounds against non-adaptive distribution testers in the huge object model. Hence, D_{yes} and D_{no}, rather than being distributions over functions from $\mathcal{D}$ to $\{0,1\}$, are distributions over distributions over $\{0,1\}^n$ (since the basic input object is a distribution over $\{0,1\}^n$).

The deterministic version of a non-adaptive tester in this setting is characterized by a set of possible responses to a sequence of queries $\mathcal{J} = (J_1, \ldots, J_s)$ to the samples. We call s the *length* of $\mathcal{J}$, and call $q = \sum_{i=1}^{s} J_i$, the *size* of $\mathcal{J}$.

Given a distribution D over distributions over $\{0,1\}^n$, we denote by $D\mid_{\mathcal{J}}$ the distribution over $\{0,1\}^q$ that results from first picking a distribution $\widehat{D}$ over $\{0,1\}^n$ according to D, then taking s independent samples $\mathbf{X}_1, \ldots, \mathbf{X}_s$ according to $\widehat{D}$, and finally constructing the sequence $\mathbf{X}_1\mid_{J_1}, \ldots, \mathbf{X}_s\mid_{J_s}$. The huge object model version of Yao's lemma for non-adaptive testers is the following one.

Lemma 3.16 (Yao's lemma for non-adaptive testers in the huge object model). *Let $\varepsilon \in (0,1)$ be a parameter and $q, s \in \mathbb{N}$ be two integers. Suppose there exists a distribution D_{yes} over distributions over $\{0,1\}^n$ that satisfy a given property $\mathcal{P}$, and a distribution D_{no} over distributions over $\{0,1\}^n$ that are ε-far from satisfying the property $\mathcal{P}$. Moreover, assume that for any query sequence $\mathcal{J}$ of length s and size q, the variation distance between $D_{yes}\mid_{\mathcal{J}}$ and $D_{no}\mid_{\mathcal{J}}$ is less than $1/3$. Then it is not possible for a non-adaptive tester that takes at most s samples and performs at most q queries to ε-test $\mathcal{P}$.*

4 Learning clusterable distributions

In this section, we define the notion of a (ζ, δ, r)-clusterable distribution formally (see Definition 4.1), and prove that such distributions can be learnt (up to permutation) efficiently in Theorem 4.2. Intuitively, a distribution D defined over $\{0,1\}^n$ is called (ζ, δ, r)-clusterable if we can remove a subset of the support vectors of D whose probability mass is at most ζ, and we can partition the remaining vectors in the support of D into at most r parts, each with diameter at most δ. Theorem 4.2 states that, given a distribution D over $\{0,1\}^n$, we can learn it (up to permutation) if it is (ζ, δ, r)-clusterable, and otherwise, we either report FAIL or learn the input distribution (up to permutation). Note that learning the distribution up to permutation is sufficient to provide testing algorithms for index-invariant properties with bounded VC-dimension, which will be discussed in Section 5.

Definition 4.1 ((ζ, δ, r)-clusterable and (α, r)-clusterable distribution). (i) Let $\zeta, \delta \in (0,1)$ and $r, n \in \mathbb{N}$. A distribution D over $\{0,1\}^n$ is called (ζ, δ, r)-*clusterable* if there exists a partition $\mathcal{C}_0, \ldots, \mathcal{C}_s$ of $\{0,1\}^n$ such that $D(\mathcal{C}_0) \leq \zeta$, $s \leq r$, and for every $1 \leq i \leq s$, $d_H(\mathbf{U}, \mathbf{V}) \leq \delta$ for any $\mathbf{U}, \mathbf{V} \in \mathcal{C}_i$.

(ii) For $\alpha \in (0,1)$ and $r \in \mathbb{N}$, a distribution D over $\{0,1\}^n$ is called (α, r)-*clusterable* if it is $(0, \alpha, r)$-*clusterable*. For $\beta \in (0,1)$, a distribution D is called β-*close to being* (α, r)-*clusterable* if there exists an (α, r)-clusterable distribution D_0 such that $d_{EM}(D, D_0) \leq \beta$.

Theorem 4.2 (Learning (ζ, δ, r)-clusterable distributions). *There exists a (non-adaptive) algorithm* TEST-AND-LEARN*, as described in Algorithm* **??***, that has sample and query access to an unknown distribution D over $\{0,1\}^n$ for $n \in \mathbb{N}$, takes parameters ζ, δ, r as inputs such that, $\zeta, \delta \in (0,1)$ and $\varepsilon = 17(\delta + \zeta) < 1$ [10] and $r \in \mathbb{N}$, makes a number of queries that only depends on ζ, δ*

[10] The constant 17 is arbitrary, and can be improved to a smaller constant. We did not try to optimize.

and r, and either reports a full description of a distribution over $\{0,1\}^n$ or reports FAIL*, satisfying both of the following conditions:*

(i) If D (ζ, δ, r)-clusterable, then with probability at least $\frac{2}{3}$, the algorithm outputs a full description of a distribution D' over $\{0,1\}^n$ such that $d_{EM}(D, D'_\sigma) \leq \varepsilon$ for some permutation $\sigma : [n] \to [n]$.

(ii) For any D, the algorithm will not output a distribution D' such that $d_{EM}(D, D'_\sigma) > \varepsilon$ for every permutation $\sigma : [n] \to [n]$, with probability more than $\frac{1}{3}$. However, if the distribution D is not (ζ, δ, r)-clusterable, the algorithm may output FAIL *with any probability.*

The algorithm corresponding to learning (ζ, δ, r)-clusterable distributions is described in Algorithm **??** as TEST-AND-LEARN. It calls a subroutine APPROX-CENTERS, as described in Algorithm **??**.

Remark 1. The sample complexity of TEST-AND-LEARN is polynomial in r, and the query complexity of TEST-AND-LEARN is exponential in r. Moreover, for the case of query complexity, the exponential dependency in r is required. In particular, in Section 6, we construct a distribution with support size r that requires $2^{\Omega(r)}$ queries to test for the property of being a permutation thereof.

To prove the correctness of TEST-AND-LEARN (which we will do in Section 4.1 and Section 4.2), we will need the notion of an (η, ξ)-clustered distribution around a sequence of vectors $\mathcal{S}$ (see Definition 4.3), and an associated observation (see Observation 4.4).

Definition 4.3 (**(η, ξ)-clustered distribution around a sequence).** Let $\eta, \xi \in (0,1)$ and $n \in \mathbb{N}$. Also, for $\mathbf{X} \in \{0,1\}^n$, let $\mathrm{NGB}_\eta(\mathbf{X})$ denote the set of vectors in $\{0,1\}^n$ that are at a distance of at most η from $\mathbf{X}$. Let $\mathcal{S} = \{\mathbf{S}_1, \ldots, \mathbf{S}_t\}$ be a sequence of t vectors in $\{0,1\}^n$ and define $\mathrm{NGB}_\eta(\mathcal{S}) = \bigcup_{S \in \mathcal{S}} \mathrm{NGB}_\eta(S)$. Then:

(i) A distribution D over $\{0,1\}^n$ is called (η, ξ)-*clustered around $\mathcal{S}$ with weights* $w_0, \ldots, w_t \in [0,1]$ satisfying $\sum_{i=0}^{t} w_i = 1$ and $w_0 \leq \xi$, if there exist t pairwise disjoint sets $\mathcal{C}_i$, such that $\mathcal{C}_i \subseteq \mathrm{NGB}_\eta(\mathbf{S}_i)$ and $D(\mathcal{C}_i) \geq w_i$ for every $i \in [t]$.

(ii) A distribution D over $\{0,1\}^n$ is called (η, ξ)-*clustered around* $\mathcal{S}$ if D is (η, ξ)-clustered around $\mathcal{S}$ with weights $w_0, \ldots, w_t \in [0,1]$, for some $w_0, \ldots, w_t$ such that $\sum_{i=0}^{t} w_i = 1$ and $w_0 \leq \xi$.

Observation 4.4. Let D be any distribution over $\{0,1\}^n$ and $\mathcal{S}$ be a sequence of vectors in $\{0,1\}^n$ such that $\mathrm{NGB}_\eta(\mathcal{S}) \geq 1 - \xi$. Then D is (η, ξ)-clustered around $\mathcal{S}$.

Proof. Let us partition $\mathrm{NGB}_\eta(\mathcal{S})$ into t parts such that $\mathcal{C}_i = \mathrm{NGB}_\eta(\mathbf{X}_i) \setminus \bigcup_{j=1}^{i-1} \mathrm{NGB}_\eta(\mathbf{X}_j)$ for every $i \in [t]$. For every $i \in [t]$, note that $\mathcal{C}_i \subseteq \mathrm{NGB}_\eta(\mathbf{X}_i)$, and let us define $w_i = D(\mathcal{C}_i)$. Also, set $w_0 = 1 - \sum_{i=1}^{t} w_i$, and observe that $w_0 = 1 - \mathrm{NGB}_\eta(\mathcal{S}) \leq \xi$. This shows that D is (η, ξ)-clustered around $\mathcal{S}$ with weights $w_0, \ldots, w_t$, and we are done. □

The correctness proof of TEST-AND-LEARN is in Subsection 4.2. Leading to it, in Subsection 4.1, we consider some important lemmas and define a set of events. These lemmas, and the events whose probability they bound from below, provide the infrastructure for the proof of TEST-AND-LEARN in Subsection 4.2.

4.1 Useful lemmas and events to prove the correctness of TEST-AND-LEARN

The central goal of this section is to define an event GOOD and show that $\Pr(\text{GOOD}) \geq 2/3$. The event GOOD is defined in such a fashion that, if it holds, then the algorithm TEST-AND-LEARN produces the desired output as stated in Theorem 4.2. Note that this bounds the error probability of TEST-AND-LEARN. The event GOOD is formally defined in Definition 4.12. To define the event GOOD, we first consider four lemmas: Lemma 4.5, Lemma 4.6, Lemma 4.7 and Lemma 4.11.

We will first state a lemma (Lemma 4.5) which says that, with high probability, the first set of samples $\mathcal{S}$ (obtained in Step (i) of TEST-AND-LEARN) intersects all the large clusters when D is (ζ, δ, r)-clusterable.

Lemma 4.5 (Hitting large clusters). *Assume that the input distribution D over $\{0,1\}^n$ is (ζ, δ, r)-clusterable with the clusters $\mathcal{C}_1, \ldots, \mathcal{C}_r$. The cluster $\mathcal{C}_i$ is said to be* large *if $D(\mathcal{C}_i) \geq \frac{\zeta}{10r}$. With probability at least 99/100, the sequence of vectors $\mathcal{S} = \{\mathbf{X}_1, \ldots \mathbf{X}_{t_1}\}$ (found in Step (i) of* TEST-AND-LEARN*) contains at least one vector from every large cluster.*

Proof. Consider any large cluster $\mathcal{C}_i$. As $D(\mathcal{C}_i) \geq \frac{\zeta}{10r}$, the probability that no vector in $\mathcal{S}$ belongs to $\mathcal{C}_i$ is at most $(1 - \frac{\zeta}{10r})^{|\mathcal{S}|} \leq \frac{99}{100r}$. This follows for a suitable choice of the hidden coefficient since $|\mathcal{S}| = t_1 = \mathcal{O}\left(\frac{r}{\zeta} \log \frac{r}{\zeta}\right)$. Since there are at most r large clusters, using the union bound, the lemma follows. □

Recall that TEST-AND-LEARN obtains a second set of sample vectors $\mathcal{T}$ in Step (ii), takes a random set of indices $R \subset [n]$ without replacement in Step (iii), and tries to assign each vector in $\mathcal{T}$ to some vector in $\mathcal{S}$, based on the distance between the vectors when projected to the indices of R. Intuitively, the step of assigning vectors performs as desired if R preserves the distances between the vectors in $\mathcal{S}$ and $\mathcal{T}$. For technical reasons, we also need R to preserve most (but not all) distances between $\mathcal{S}$ and the entirety of $\{0,1\}^n$. The following lemma says that indeed R achieves this with high probability.

Lemma 4.6 (Distance preservation). *Let us consider the input distribution D over $\{0,1\}^n$, and $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$ and $R \subset [n]$ drawn in Step (i) and (iii) of* TEST-AND-LEARN*. R is said to be* distance preserving *if the following conditions hold:*

(i) $|d_H(\mathbf{S}, \mathbf{T}) - d_H(\mathbf{S}\,|_R, \mathbf{T}\,|_R)| \leq \delta$ *for every* $\mathbf{S} \in \mathcal{S}$ *and* $\mathbf{T} \in \mathcal{T}$.

(ii) Let $\mathcal{W} \subseteq \{0,1\}^n$ *be such that, for every* $\mathbf{W} \in \mathcal{W}$, $|d_H(\mathbf{W}, \mathbf{S}) - d_H(\mathbf{W}\,|_R, \mathbf{S}\,|_R)| \leq \delta$. *Then* $D(\mathcal{W}) \geq 1 - \frac{\zeta}{t_1}$.

The set R chosen in Step (iii) of TEST-AND-LEARN *is distance preserving with probability at least* 99/100.

Proof. For (i), consider a particular $\mathbf{S} \in \mathcal{S}$ and $\mathbf{T} \in \mathcal{T}$. Applying Observation A.5 with $K = R$, $\mathbf{U} = \mathbf{S}$ and $\mathbf{V} = \mathbf{T}$, the probability that $|d_H(\mathbf{S}, \mathbf{T}) - d_H(\mathbf{S}\,|_R, \mathbf{T}\,|_R)| \leq \delta$ is at least $1 - \frac{\zeta}{200 t_1^2 t_2}$. Applying the union bound over all possible choices over $(\mathbf{S}, \mathbf{T})$ pairs, we have Part (i) with probability at least 199/200.

To prove (ii), let us consider an arbitrary vector $\mathbf{V} \in \{0,1\}^n$. Similarly to (i), we know that $|d_H(\mathbf{V}, \mathbf{S}) - d_H(\mathbf{V}\,|_R, \mathbf{S}\,|_R)| \leq \delta$ holds with probability at least $1 - \frac{\zeta}{200 t_1^2 t_2}$. Applying the union bound, we can say that the same holds over all $\mathbf{S} \in \mathcal{S}$ with probability at least $1 - \frac{\zeta}{200 t_1}$. So, the expected value of $D(\{0,1\}^n \setminus \mathcal{W})$ is at most $\frac{\zeta}{200 t_1}$. By Markov's inequality, the probability that Part (ii) holds, that is, $D(\{0,1\}^n \setminus \mathcal{W}) \leq \frac{\zeta}{t_1}$ is at least 199/200. Putting everything together, we have the result. □

By Lemma 4.5, we know that $\mathcal{S}$ intersects with all large clusters with high probability, and we are trying to assign the vectors in $\mathcal{T}$ to some vectors in $\mathcal{S}$ based on their projected distances on the indices of R. To learn the input distribution, we want the second set of sample vectors $\mathcal{T}$ to preserve the mass of all the large clusters, and it is enough for us to approximate it, as well as be able to detect the case where approximation is impossible and we should output FAIL. The following lemma takes care of this.

Lemma 4.7 (Weight representation). *Let us consider the input distribution D over $\{0,1\}^n$ to* TEST-AND-LEARN*, $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$ in Step (i), $\mathcal{T} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_{t_2}\}$ in Step (ii), and consider fixed t_1 pairwise disjoint subsets $\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_{t_1}\}$ of $\{0,1\}^n$. $\mathcal{T}$ is said to be* weight preserving *for $\mathcal{S}$ and $\mathcal{C}$ if*

(i) $\frac{|\mathcal{T} \cap \mathrm{NGB}_\delta(\mathcal{S})|}{|\mathcal{T}|} \geq D(\mathrm{NGB}_\delta(\mathcal{S})) - \zeta.$

(ii) $\frac{|\mathcal{T} \cap \mathrm{NGB}_{3\delta}(\mathcal{S})|}{|\mathcal{T}|} \leq D(\mathrm{NGB}_{3\delta}(\mathcal{S})) + \zeta.$

(iii) for every $i \in [t_1]$, $\frac{|\mathcal{T} \cap \mathcal{C}_i|}{|\mathcal{T}|} \leq D(\mathcal{C}_i) + \frac{\zeta}{t_1}.$

Then with probability at least 99/100, $\mathcal{T}$ is weight reserving for $\mathcal{S}$ and $\mathcal{C}$.

Proof. To prove (i), let Z_j be the indicator random variable such that $Z_j = 1$ if and only if $\mathbf{Y}_j$ is in $\mathrm{NGB}_\delta(\mathcal{S})$, where $j \in [t_2]$. Observe that $|\mathcal{T} \cap NGB_\delta(\mathcal{S})| = \sum_{j=1}^{t_2} Z_j$. As $\Pr(Z_j = 1) = D(\mathrm{NGB}_\delta(\mathcal{S}))$, the expected value of $\frac{|\mathcal{T} \cap \mathrm{NGB}_\delta(\mathcal{S})|}{|\mathcal{T}|}$ is also $D(\mathrm{NGB}_\delta(\mathcal{S}))$. Applying Hoeffding's inequality (see Lemma A.3), we conclude that (i) holds with probability at least 299/300.

Proving (ii) is similar to (i). Again applying Hoeffding's inequality (Lemma A.3), we can show that (ii) holds with probability at least 299/300.

In order to prove (iii), we proceed in similar fashion as (i), and after applying Hoeffding's inequality (Lemma A.3), we apply the union bound over all $j \in [t_1]$ to get the desired result. □

Consider the weights $w_1, \ldots, w_{t_1}$ obtained in Step (vi) of TEST-AND-LEARN. To argue that these weights are good enough to report the desired distribution D' (if we know the vectors in $\mathcal{S}$ exactly), we consider the following observation which says that there exist t_1 pairwise disjoint subsets $\mathcal{C}_1^*, \ldots, \mathcal{C}_{t_1}^*$ such that w_i is the fraction of vectors in $\mathcal{T}$ that are in $\mathcal{C}_i^*$ for every $i \in [t_1]$. Also, let us define $\mathcal{C}^* = \{\mathcal{C}_1^*, \ldots, \mathcal{C}_{t_1}^*\}$.

Observation 4.8. Let us consider assigning each vector in $\{0,1\}^n$ either to some $S \in \mathcal{S}$ or not assigning to any vector in $\mathcal{S}$, using the same procedure that has been used to assign the set of vectors in $\mathcal{T}$ in Steps (iii) and (iv) of TEST-AND-LEARN. Let $\mathcal{C}_i^* \subseteq \{0,1\}^n$ be the set of all vectors that are assigned to $\mathbf{X}_i$, for every $i \in [t_1]$. Then, for every $i \in [t_1]$, we have $w_i = \frac{|\mathcal{T} \cap \mathcal{C}_i^*|}{|\mathcal{T}|}$.

Proof. This follows from the definition of $\mathcal{C}_i^*$. □

Note that $\mathcal{C}^*$ is formed following the procedure that TEST-AND-LEARN performs to assign the vectors of $\mathcal{T}$ to the vectors in $\mathcal{S}$. So, a vector far away from $\mathbf{X}_i \in \mathcal{S}$ might be assigned $\mathbf{X}_i$, and w_i is considered in this case. This is not a problem as the mass on $\mathcal{C}_i^*$ is close to being bounded by the total mass of the vectors in $\mathrm{NGB}_{3\delta}(\mathbf{X}_i)$. This follows from the fact that the set R is distance preserving (see Part (ii) of Lemma 4.6) with high probability. Now let us define $\mathcal{C}^{**} = \{\mathcal{C}_i^* \cap \mathrm{NBG}_{3\delta}(\mathbf{X}_i) : i \in [t_1]\}$. Finally, we will upper bound w_i by $D(\mathcal{C}_i^{**})$ in the following observation. This will be useful for proving the correctness of TEST-AND-LEARN in Section 4.2.

Observation 4.9. Let us assume that R is distance preserving and $\mathcal{T}$ is weight representative of $\mathcal{S}$ and $\mathcal{C}^*$. Then for every $i \in [t_1]$, $w_i \leq D(\mathcal{C}_i^*) + \frac{\zeta}{t_1} \leq D(\mathcal{C}_i^{**}) + \frac{2\zeta}{t_1}$, where we define $\mathcal{C}_i^{**} = \mathcal{C}_i^* \cap \mathrm{NBG}_{3\delta}(\mathbf{X}_i)$.

Proof. As R is distance preserving, consider $\mathcal{C}^* = \{\mathcal{C}_1^*, \ldots, \mathcal{C}_{t_1}^*\}$ as guaranteed by Observation 4.8. Now, as $\mathcal{T}$ is weight representative of $\mathcal{S}$ and C^* and $w_i = \frac{|\mathcal{T} \cap \mathcal{C}_i^*|}{|T|}$ for every $i \in [t_1]$, by Lemma 4.7 (iii), $w_i \leq D(\mathcal{C}_i^*) + \frac{\zeta}{t_1}$. By the definition of $\mathcal{C}_i^*$ and by Lemma 4.6 (ii), $D(\mathcal{C}_i^* \setminus \mathrm{NGB}_{3\delta}(\mathbf{X}_i)) \leq \frac{\zeta}{t_1}$, that is, $D(\mathcal{C}_i^*) \leq D(\mathcal{C}_i^{**}) + \frac{\zeta}{t_1}$. □

Note that the above observation only gives upper bounds on the set of weights $w_1, \ldots, w_{t_1}$. As Lemma 4.7 provides upper as well as lower bounds on the mass around $\mathcal{S}$, this will not be a problem.

Consider the distribution D^* supported over $\mathcal{S}$ such that $D(\mathbf{X}_i) \geq w_i$ for every $i \in [t_1]$, which we can view as an approximation of D. Note that we still can not report D^* as the output distribution, since in order to do so, we need to perform $\Omega(n)$ queries to know the exact vectors of $\mathcal{S}$. Instead we will report a distribution D' such that D'_σ is close to D^* for some permutation $\sigma : [n] \to [n]$. The idea is to construct a new set of vectors $\mathbf{S}_1, \ldots, \mathbf{S}_{t_1}$ in Step (vii) such that the Hamming distance between $\mathbf{X}_i$ and $\sigma(\mathbf{S}_i)$ is small for every $i \in [t_1]$ for some permutation $\sigma : [n] \to [n]$. Lemma 4.11 implies that this is possible from the projection of the vectors in $\mathcal{S}$ onto the indices of R (the implication itself will be proved later in Lemma 4.16). Before proceeding to Lemma 4.11, we need the following definition and observation.

Definition 4.10. Given any sequence of vectors $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\} \subseteq \{0,1\}^n$ and $j \in [n]$, we define the vector $C_j^{\mathcal{S}} \in \{0,1\}^{t_1}$ as

$$\text{for every } i \in [t_1],\ C_j^{\mathcal{S}}(i) = \mathbf{X}_i(j).$$

For any $J \in \{0,1\}^{t_1}$, we define

$$\alpha_J = \frac{|\{j \in [n] \mid C_j^{\mathcal{S}} = J\}|}{n}.$$

Intuitively, let us consider a matrix M of order $t_1 \times n$ such that the i-th row vector corresponds to the vector $\mathbf{X}_i$. Then observe that $C_j^{\mathcal{S}}$ represents the j-th column vector of the matrix M and α_J denotes the fraction of column vectors of M that are identical to J.

Lemma 4.11 (Structure preservation). *Let us consider the input distribution D over $\{0,1\}^n$, $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$ and $R \subset [n]$ drawn in Step (i) and (iii) of* Test-and-Learn*. Also, let us consider the values of Γ_J found in Step (iii) of* Approx-Centers *(called from Step (vii) of* Test-and-Learn*). The set R is said to be* structure preserving *for $\mathcal{S}$ if $\left|\alpha_J - \frac{\Gamma_J}{n}\right| \leq \frac{\delta}{10 \cdot 2^{t_1}}$ holds for every $J \in \{0,1\}^{t_1}$. Then the set R chosen in Step (iii) of* Test-and-Learn *is structure preserving for $\mathcal{S}$ with probability at least* 99/100*.*

Proof. Consider any particular $J \in \{0,1\}^{t_1}$ and γ_J determined by Step (ii) of Approx-Centers. By applying Hoeffding's bound for sampling without replacement (Lemma A.4), we obtain, for any $\eta > 0$,

$$\Pr\left[|\gamma_J - \alpha_J| \geq \frac{\eta}{20}\right] \leq e^{-2\eta^2 |R|/400}.$$

By substituting the value of $|R|$ (for a suitable choice of the hidden coefficient) and $\eta = \frac{\delta}{2^{t_1}}$, and using the union bound over all possible $J \in \{0,1\}^{t_1}$, we conclude that with probability at least 99/100, for all $J \in \{0,1\}^{t_1}$, $|\gamma_J - \alpha_J| \leq \frac{\delta}{20 \cdot 2^{t_1}}$.

Note that Approx-Centers constructs Γ_J's from γ_j's by applying Observation 3.1. From the way Observation 3.1 generates Γ_J's from γ_j's, we conclude that for all $J \in \{0,1\}^{t_1}$, $|\gamma_J - \frac{\Gamma_J}{n}| \leq \frac{1}{n}$, completing the proof, assuming that n is larger than $\frac{20 \cdot 2^{t_1}}{\delta}$. □

Now we are ready to define the event GOOD.

Definition 4.12 (Definition of the event GOOD). Let us define an event GOOD as $\mathcal{E}_1 \wedge \mathcal{E}_2 \wedge \mathcal{E}_3 \wedge \mathcal{E}_4$, where

(i) $\mathcal{E}_1$: If D is (ζ, δ, r)-clusterable with the clusters $\mathcal{C}_1, \ldots, \mathcal{C}_r$, then $\mathcal{S} = \{\mathbf{X}_1, \ldots \mathbf{X}_{t_1}\}$ (found in Step (i) of Test-and-Learn) contains at least one vector from every large cluster.

(ii) $\mathcal{E}_2 : R$ (picked in Step (ii) of Test-and-Learn) is distance preserving.

(iii) $\mathcal{E}_3 : R$ is structure preserving for $\mathcal{S}$.

(iv) $\mathcal{E}_4$: $\mathcal{T}$ is weight preserving for $\mathcal{S}$ and $\mathcal{C}^*$, where $\mathcal{C}^* = \{\mathcal{C}^*_1, \ldots, C^*_{t_1}\}$ is as defined in Observation 4.8.

Note that the event $\mathcal{E}_1$ follows from Lemma 4.5, $\mathcal{E}_2$ follows from Lemma 4.6, $\mathcal{E}_3$ follows from Lemma 4.11, and $\mathcal{E}_4$ follows from Lemma 4.7. Thus, from the respective guarantees of the aforementioned lemmas, we can say that $\Pr(\mathcal{E}_1), \Pr(\mathcal{E}_2), \Pr(\mathcal{E}_3), \Pr(\mathcal{E}_4) \geq \frac{99}{100}$. To address a subtle point, note that Lemma 4.6 gives a probability lower bound on R being distance preserving for any choice of $\mathcal{T}$, and hence the lower bound also holds for $\mathcal{T}$ sampled according to the distribution. Similarly, Lemma 4.7 provides a probability lower bound on $\mathcal{T}$ being weight representative for any choice of R (which affects C^*) regardless of whether R is distance preserving, and hence the lower bound also holds for the R chosen at random by the algorithm. So, we have the following lemma.

Lemma 4.13. $\Pr(\text{GOOD}) \geq \frac{2}{3}$.

4.2 Proof of Theorem 4.2 (Correctness of Test-and-Learn)

In the first three lemmas below (Lemma 4.14, Lemma 4.15 and Lemma 4.16), we prove the correctness of the internal steps of the algorithm. These lemmas are stated under the conditional space that the event GOOD defined in Definition 4.12 occurs. Using these lemmas along with Lemma 4.17, which helps us combine them, allows us to prove Theorem 4.2.

Lemma 4.14 (Guarantee till Step (v) of Test-and-Learn). *Assume that the event* GOOD *holds.*

(i) If D is (ζ, δ, r)-clusterable, then D is $(\delta, 2\zeta)$-clustered around $\mathcal{S}$, and the fraction of samples in $\mathcal{T}_y$ that are not assigned to any vector in $\mathcal{S}_x$ will be at most 3ζ. That is, Test-and-Learn *does not output* Fail *in Step (v) and proceeds to Step (vi).*

(ii) If D is not $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$, then the fraction of samples in $\mathcal{T}_y$ that are not assigned to any vector in $\mathcal{S}_x$ will be at least 3ζ. That is, Test-and-Learn *outputs* Fail *and does not proceed to Step (vi).*

Proof. (i) For the first part, as $\mathcal{E}_1$ holds (see Lemma 4.5), the set $\mathcal{S}$ contains at least one vector from every large cluster. Now, if we consider the δ-neighborhood of $\mathcal{S}$, that is, $\text{NGB}_\delta(\mathcal{S})$, we infer that all vectors in large clusters are in $\text{NGB}_\delta(\mathcal{S})$. By the definition of a large cluster, the mass on the vectors that are not in any large cluster is at most 2ζ. Hence, we conclude that $D(\text{NGB}_\delta(\mathcal{S})) \geq (1 - 2\zeta)$. Thus, by Observation 4.4, D is $(\delta, 2\zeta)$-clustered around $\mathcal{S}$. For the second part, as the event $\mathcal{E}_4$ holds (see Lemma 4.7 (i)), $\mathcal{T}$ is weight representative for $\mathcal{S}$. This follows since D is $(\delta, 2\zeta)$-clustered, and in particular is $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$. Thus,

$\frac{|\mathcal{T}\cap \mathrm{NGB}_\delta(\mathcal{S})|}{|\mathcal{T}|} \geq D(\mathrm{NGB}_\delta(\mathcal{S})) - \zeta$. Also, as the event $\mathcal{E}_2$ holds (see Lemma 4.6), R is distance preserving between $\mathcal{S}$ and $\mathcal{T}$, meaning that if $\mathbf{Y}_i$ in $\mathcal{C}_j$, then $\mathbf{y}_i$ is assigned to $\mathbf{x}_j$. Hence,

$$\sum_{i=1}^{t_1} w_i \geq \frac{|\mathcal{T} \cap \mathrm{NGB}_\delta(\mathcal{S})|}{|\mathcal{T}|} \geq D(\mathrm{NGB}_\delta(\mathcal{S})) - \zeta \geq 1 - 3\zeta.$$

That is, $w_0 \leq 3\zeta$, and the algorithm Test-and-Learn does not report Fail and proceeds to Step (vi).

(ii) Since the distribution D is not $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$, by Observation 4.4, $D(\mathrm{NGB}_{3\delta}(\mathcal{S})) < 1-5\zeta$. As the event $\mathcal{E}_4$ holds (see Lemma 4.7 (ii)), $\frac{|\mathcal{T}\cap \mathrm{NGB}_{3\delta}(\mathcal{S})|}{|\mathcal{T}|} \leq D(\mathrm{NGB}_{3\delta}(\mathcal{S}))+\zeta \leq 1-4\zeta$. Also, as the event $\mathcal{E}_2$ holds (see Lemma 4.6), R is distance preserving between $\mathcal{S}$ and $\mathcal{T}$. This implies that

$$\sum_{i=1}^{t_1} w_i \leq \frac{|\mathcal{T} \cap \mathrm{NGB}_{3\delta}(\mathcal{S})|}{|\mathcal{T}|} \leq D(\mathrm{NGB}_{3\delta}(\mathcal{S})) + \zeta < 1 - 3\zeta.$$

That is, $w_0 > 3\zeta$, and the algorithm Test-and-Learn reports Fail. So, Test-and-Learn does not proceed to Step (vi).

□

Lemma 4.15 (Guarantee from Step (vi) of Test-and-Learn). *Assume that the event* GOOD *holds. Also, assume that D is $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$ and $w_0 \leq 3\zeta$ holds in Step (vi) of Test-and-Learn. Consider the following distribution D'' over $\{0,1\}^n$, constructed from the weights obtained from Step (vi) of Test-and-Learn, such that*

(i) For each $i \in [t_1]$, $D''(\mathbf{X}_i) = w(\mathbf{x}_i) = w_i$.

(ii) $D''(\mathbf{X}_0) = 1 - \sum_{i=1}^{t_1} w(\mathbf{x}_i)$ for some arbitrary $\mathbf{X}_0$.

(iii) $D''(\mathbf{X}) = 0$ for every $\mathbf{X} \in \{0,1\}^n \setminus \{\mathbf{X}_0, \ldots, \mathbf{X}_{t_1}\}$.

Then D'' is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w_0, \ldots, w_{t_1}$, where $w_0 = 1 - \sum_{i=1}^{t_1} w_i$, and the EMD between D and D'' satisfies $d_{EM}(D, D'') \leq 10\delta + 12\zeta$.

We will prove Lemma 4.15 in Subsection 4.2. Now we proceed to prove the guarantee regarding Step (vii) of Test-and-Learn.

Lemma 4.16 (Guarantee from Step (vii) of Test-and-Learn). *Assume that the event* GOOD *holds. Then, in Step (vii), the algorithm* Approx-Centers *(if called as described in Algorithm **??**) outputs a sequence of vectors $\{\mathbf{S}_1, \ldots, \mathbf{S}_{t_1}\}$ in $\{0,1\}^n$, such that there exists a permutation $\sigma : [n] \to [n]$ for which $d_H(\sigma(\mathbf{X}_i), \mathbf{S}_i) \leq \frac{\delta}{10}$ holds for every $i \in [t_1]$.*

Proof. Here we assume that the event GOOD holds. In particular, we assume that the event $\mathcal{E}_3$ holds.

Let us consider a matrix M of order $t_1 \times n$ such that the i-th row vector corresponds to the vector $\mathbf{X}_i$. Then observe that $C_j^{\mathcal{S}}$ represents the j-th column vector of matrix M and α_J denotes the fraction of column vectors of M that are identical to the vector J.

Let us consider the matrix A of order $t_1 \times n$ constructed by our algorithm, by putting Γ_J many column vectors identical to J, for every $J \in \{0,1\}^{t_1}$. Note that $\{\mathbf{S}_1, \ldots, \mathbf{S}_{t_1}\}$ are the row vectors

corresponding to A. As we are assuming that the event $\mathcal{E}_3$ holds (see Lemma 4.11), $|\alpha_J - \frac{\Gamma_J}{n}| \leq \frac{\delta n}{10 \cdot 2^{t_1}}$ holds for every $J \in \{0,1\}^{t_1}$. Observe that we can permute the columns of the matrix M using a permutation $\sigma : [n] \to [n]$ and create a matrix M_σ, such that there exists a bad set $I \subset [n]$ of size at most $\frac{\delta \cdot n}{10}$, where after the removal of the columns corresponding to indices of I from both matrices M_σ and A become identical. Hence, we infer that $d_H(\sigma(\mathbf{X}_i), \mathbf{S}_i) \leq \frac{\delta}{10}$ for every $i \in [t_1]$, where σ is the permutation corresponding to M_σ. This completes the proof of Lemma 4.16. □

Finally, to prove Theorem 4.2, we need to show that the Earth Mover Distance between two distributions defined over close vectors is bounded when one distribution is clustered around a sequence of vectors and the other distribution has similar weights compared to the first distribution.

Lemma 4.17 (EMD between distributions having close cluster centers). *Let $\eta, \kappa, \xi \in (0,1)$ be three parameters such that $\eta + \kappa + \xi < 1$. Suppose that $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$ and $\mathcal{S}' = \{\mathbf{X}'_1, \ldots, \mathbf{X}'_{t_1}\}$ are two sequences of vectors over $\{0,1\}^n$ such that $d_H(\mathbf{X}_i, \mathbf{X}'_i) \leq \kappa$ for every $i \in [t_1]$. Moreover, let D be an (η, ξ)-clustered distribution around $\mathcal{S}$ with weights $w_0, \ldots, w_{t_1}$ and D' be another distribution such that $D'(\mathbf{X}'_i) \geq w_i$ for every $i \in [t_1]$. Then $d_{EM}(D, D') \leq \eta + \xi + \kappa$.*

Proof. Recall that the EMD between D and D' is the solution to the following LP:

$$\begin{aligned}
&\text{Minimize} && \sum_{\mathbf{X},\mathbf{Y} \in \{0,1\}^n} f_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \\
&\text{Subject to} && \sum_{\mathbf{Y} \in \{0,1\}^n} f_{\mathbf{XY}} = D(\mathbf{X}) \ \forall \mathbf{X} \in \{0,1\}^n \ , \ \sum_{\mathbf{X} \in \{0,1\}^n} f_{\mathbf{XY}} = D'(\mathbf{Y}) \ \forall \mathbf{Y} \in \{0,1\}^n \\
&\text{and} && 0 \leq f_{\mathbf{XY}} \leq 1, \quad \forall \mathbf{X}, \mathbf{Y} \in \{0,1\}^n.
\end{aligned}$$

Here D is (η, ξ)-clustered around $\mathcal{S}$. Let $\mathcal{C}_1, \ldots, \mathcal{C}_{t_1}$ be the pairwise disjoint subsets of $\{0,1\}^n$ such that $\mathcal{C}_i \subseteq \mathrm{NGB}_\eta(\mathbf{X}_i)$ and $D(\mathcal{C}_i) \geq w_i$ for every $i \in [t_1]$.

Consider a particular solution $\{f^*_{\mathbf{XY}} : \mathbf{X}, \mathbf{Y} \in \{0,1\}^n\}$ that also satisfies the constraint

$$\sum_{\mathbf{X} \in \mathcal{C}_i} f_{\mathbf{XX}'_i} \geq w_i \text{ for every } i \in [t_1].$$

The above constraint is feasible as $D(\mathcal{C}_i) \geq w_i$ and $D'(\mathbf{X}'_i) \geq w_i$, where $i \in [t_1]$.

Now,

$$\begin{aligned}
\mathrm{EMD}(D, D') &\leq \sum_{\mathbf{X},\mathbf{Y} \in \{0,1\}^n} f^*_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \\
&\leq \sum_{i=1}^{t_1} \sum_{\mathbf{X} \in \mathcal{C}_i} f^*_{\mathbf{XX}'_i} d_H(\mathbf{X}, \mathbf{X}'_i) + \sum_{\mathbf{X} \notin \bigcup_{i=1}^{t_1} \mathcal{C}_i, \mathbf{Y} \in \{0,1\}^n} f^*_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \\
&\leq \sum_{i=1}^{t_1} w_i \cdot (\eta + \kappa) + w_0 \cdot 1 \\
&\leq \eta + \kappa + \xi.
\end{aligned}$$

□

Proof of Theorem 4.2

To prove Theorem 4.2, we need the following lemma.

Lemma 4.18. *If D is $(3\delta, 5\zeta)$-clustered around S, and* Test-and-Learn *executes Step (vi), then $d_{EM}(D, D'_\sigma) \leq 17(\delta + \zeta)$ for some permutation $\sigma : [n] \to [n]$.*

Proof. As D is $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$, by Lemma 4.15, we have that D'' is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w_0, \ldots, w_{t_1}$ and $d_{EM}(D, D'') \leq 10\delta + 12\zeta$.

Now consider Step (vii) of Test-and-Learn, where we call Approx-Centers with R and $\mathbf{x}_1, \ldots, \mathbf{x}_{t_1}$ to obtain $\mathbf{S}_1, \ldots, \mathbf{S}_{t_1}$. By Lemma 4.16, $d_H(\sigma(\mathbf{X}_i), \mathbf{S}_i) \leq \frac{\delta}{10}$ for every $i \in [t_1]$ for some permutation $\sigma : [n] \to [n]$. Consider the sequence of vectors $\mathbf{X}_1^\sigma \ldots, \mathbf{X}_{t_1}^\sigma$ where $\mathbf{X}_i^\sigma = \sigma(\mathbf{X}_i)$ for every $i \in [t_1]$.

Let us now consider the distribution D''_σ over $\{0,1\}^n$ such that $D''_\sigma(\mathbf{X}) = D''(\sigma(\mathbf{X}))$ for every $\mathbf{X} \in \{0,1\}^n$. As D'' is $(5\delta, 5\zeta)$-clustered around $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$ with weights $w_0, \ldots, w_{t_1}$, D''_σ is $(5\delta, 5\zeta)$-clustered around $\{\mathbf{X}_1^\sigma, \ldots, \mathbf{X}_{t_1}^\sigma\}$ with weights $w_0, \ldots, w_{t_1}$. In the output distribution D', $D'(\mathbf{S}_i) \geq w_i$ for every $i \in [t_1]$. So, by Lemma 4.17, we have $d_{EM}(D', D''_\sigma) \leq 5\delta + \frac{\delta}{10} + 5\zeta$. Combining this with the fact that $d_{EM}(D, D'') \leq 10\delta + 12\zeta$, we conclude that $d_{EM}(D, D'_\sigma) \leq 17(\delta + \zeta)$. □

To prove Theorem 4.2, we first prove that the guarantees of the two parts follow assuming that the event GOOD holds. We will be done since $\Pr(\text{GOOD}) \geq 2/3$ (see Lemma 4.13). The query complexity of the algorithm follows from the parameters in its description.

Proof of Part (i): Here D is (ζ, δ, r)-clusterable. By Lemma 4.14, D is $(\delta, 2\zeta)$-clustered around $\mathcal{S}$ and the fraction of samples in $\mathcal{T}_y$ that are not assigned to any vector in $\mathcal{S}_x$ is at most 3ζ. That is, Test-and-Learn does not output Fail for D in Step (v). By Lemma 4.18, we conclude that $d_{EM}(D, D'_\sigma) \leq 17(\delta + \zeta)$ for some permutation $\sigma : [n] \to [n]$. This completes the proof of Part (i). □

Proof of Part (ii): Recall that we are working under the conditional space that the event GOOD holds. Now consider the following:

- If D is not $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$, then by Lemma 4.14, the algorithm Test-and-Learn reports Fail.
- If D is $(3\delta, 5\zeta)$-clustered around $\mathcal{S}$, then the algorithm Test-and-Learn either reports Fail in Step (v) or continues to Step (vi). In case we go to Step (vi), following Lemma 4.18, we again conclude that $d_{EM}(D, D'_\sigma) \leq \varepsilon$.

Observe that the above two statements imply Part (ii). This completes the proof of Theorem 4.2. □

Proof of Lemma 4.15

Here we assume that the event GOOD holds. In particular, the events $\mathcal{E}_2$ and $\mathcal{E}_4$ hold. To prove Lemma 4.15, we will prove some associated claims and lemmas about the weights $w_0, \ldots, w_{t_1}$ obtained in Step (vi) of Test-and-Learn, and the distribution D'' defined in Lemma 4.15. Let us start with the following claim.

Claim 4.19. *The distribution D'' (defined in the statement of Lemma 4.15) is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w_0, w_1 \ldots, w_{t_1}$, where $w_0 = 1 - \sum_{i=1}^{t_1} w_i$.*

Proof. This follows from the definition of D'', along with the fact that $w_0 \leq 3\zeta < 5\zeta$. □

Now we have the following claim.

Claim 4.20. *There exists a sequence of weights $w'_0, \ldots, w'_{t_1}$ such that D is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w'_0, \ldots, w'_{t_1}$, and $\sum_{i=1}^{t_1} |w_i - w'_i| \leq 2\zeta$.*

Proof. As events $\mathcal{E}_2$ and $\mathcal{E}_4$ hold, consider $\mathcal{C}^* = \{\mathcal{C}^*_1, \ldots, \mathcal{C}^*_{t_1}\}$ (as guaranteed by Observation 4.8) and $\mathcal{C}^{**} = \{\mathcal{C}^{**}_1, \ldots, \mathcal{C}^{**}_{t_1}\}$ such that, for every $i \in [t_1]$, $\mathcal{C}^{**}_i = \mathcal{C}^*_i \cap \mathrm{NGB}_{3\delta}(\mathbf{X}_i)$ and $w_i \leq D(\mathcal{C}^{**}_i) + \frac{2\zeta}{t_1}$ (see Observation 4.9).

Let us define $w'_i = \max\{w_i - \frac{2\zeta}{t_1}, 0\}$ and $w'_0 = 1 - \sum_{i=1}^{t_1} w'_i$. So, $w'_i \leq D\left(\mathcal{C}^{**}_i\right)$.

Now

$$w'_0 = 1 - \sum_{i=1}^{t_1} w'_i \leq 1 - \sum_{i=1}^{t_1} \left(w_i - \frac{2\zeta}{t_1}\right) \leq (w_0 + 2\zeta) \leq 3\zeta + 2\zeta = 5\zeta.$$

Putting everything together, the above $\mathcal{C}^{**}$ satisfies $\mathcal{C}^{**}_i \subseteq \mathrm{NGB}_{3\delta}(\mathbf{X}_i) \subseteq \mathrm{NGB}_{5\delta}(\mathbf{X}_i)$ and has weights $w'_0, \ldots, w'_{t_1}$ such that $w'_0 \leq 5\zeta$ and $w'_i \leq D(\mathcal{C}^{**}_i)$ for every $i \in [t_1]$. Hence, D is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w'_0, \ldots, w'_{t_1}$. Moreover, $\sum_{i=1}^{t_1} |w_i - w'_i| \leq 2\zeta$ holds following the definition of w'_is. □

Lemma 4.21 (Comparison-by-weights). *Let D_1 and D_2 be two distributions defined over $\{0,1\}^n$ that are (η, ξ)-clustered around a sequence of vectors $\mathcal{S} = \{\mathbf{X}_1, \ldots, \mathbf{X}_{t_1}\}$ with weights $v_0, \ldots, v_{t_1}$ and $w_0, \ldots, w_{t_1}$, respectively. Then the Earth Mover Distance between D_1 and D_2 is $d_{EM}(D_1, D_2) \leq 2\eta + \sum_{i=1}^{t_1} |v_i - w_i| + 2\xi$.*

Proof. Let $\mathbf{U}$ be an arbitrary vector from $\{0,1\}^n$. Let us define a distribution D'_1 (supported over $\mathcal{S} \cup \{\mathbf{U}\}$) from the distribution D_1 as follows:

$$D'_1(\mathbf{Y}) = \begin{cases} v_i & \mathbf{Y} = \mathbf{X}_i \text{ for every } i \in [t_1] \\ 1 - \sum_{i=1}^{t_1} v_i & \mathbf{Y} = \mathbf{U} \\ 0 & \text{otherwise} \end{cases}$$

Similarly, we define a distribution D'_2 from D_2. First we have the following claim, which follows from the definitions. From the definitions of D'_1 and D'_2, we can say that

(i) $d_{EM}(D_1, D'_1) \leq \eta + \xi$ and $d_{EM}(D_2, D'_2) \leq \eta + \xi$ (by Lemma 4.17).

(iii) $d_{EM}(D'_1, D'_2) \leq \sum_{i=1}^{t_1} |v_i - w_i|$.

Using the triangle inequality, we have

$$\begin{aligned} d_{EM}(D_1, D_2) &\leq d_{EM}(D_1, D'_1) + d_{EM}(D'_1, D'_2) + d_{EM}(D_2, D'_2) \\ &\leq 2\eta + \sum_{i=1}^{t_1} |v_i - w_i| + 2\xi. \end{aligned}$$

This completes the proof of Lemma 4.21. □

Now we proceed to prove Lemma 4.15.

Proof of Lemma 4.15. By the description of D'' in Lemma 4.15, using Claim 4.19, we know that D'' is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w_1, \ldots, w_{t_1}$. By applying Claim 4.20, D is $(5\delta, 5\zeta)$-clustered around $\mathcal{S}$ with weights $w'_0, \ldots, w'_{t_1}$ such that $\sum_{i=1}^{t_1} |w_i - w'_i| \leq 2\zeta$. Now, by applying Lemma 4.21 with $\eta = 5\delta$, $\xi = 5\zeta$, we obtain that the Earth Mover Distance between D and D'' is bounded as follows:

$$d_{EM}(D, D'') \leq 10\delta + 2\zeta + 10\zeta \leq 10\delta + 12\zeta.$$

This completes the proof of Lemma 4.15. □

5 Testing properties with bounded VC-dimension

In this section, we will prove that distributions over $\{0,1\}^n$ whose support have bounded VC-dimension can be learnt (up to permutations) by performing a number of queries that is independent of the dimension n, and depends only on the proximity parameter ε and the VC-dimension d (Theorem 1.2). In fact, we will prove a generalization, that any distribution D that is β-close to bounded VC-dimension can be learnt efficiently up to permutations (with a proximity parameter depending on β) by performing a set of queries whose size is independent of n (Theorem 5.1). As a consequence of the learning result of Theorem 5.1, we also obtain a tester for properties having a bounded VC-dimension (Corollary 5.2) which is a restatement of Corollary 1.3.

In Subsection 5.1, we connect the notions of (ζ, δ, r)-clusterablity and being β-close to (α, r)-clusterablity (Definition 4.1) in Lemma 5.4 and prove Corollary 5.3 regarding learning distributions that are β-close to (α, r)-clusterable. Then, in Subsection 5.2, we recall some standard results from VC theory to connect the notions of bounded VC-dimension and clusterability, to obtain Corollary 5.12, which is crucially used in Subsection 5.3 to prove Theorem 5.1.

Theorem 5.1 (**Learning a distribution β-close to bounded VC-dimension).** *Let $d \in \mathbb{N}$ be a constant. There exists a (non-adaptive) algorithm, that given sample and query access to an unknown distribution D over $\{0,1\}^n$, takes $\alpha, \beta \in (0,1)$ with $\beta < \alpha$ as input such that $\varepsilon = 17(3\alpha + \beta/\alpha) < 1$, makes number of queries that depends only on α, β and d, and either reports a full description of a distribution, or* Fail*, satisfying both of the following conditions:*

(i) If D is β-close to VC-dimension d, then with probability at least $2/3$, the algorithm outputs a distribution D' such that $d_{EM}(D, D'_\sigma) \leq \varepsilon$ for some permutation $\sigma : [n] \to [n]$.

(ii) For any D, the algorithm will not output a distribution D' such that $d_{EM}(D, D'_\sigma) > \varepsilon$ for every permutation $\sigma : [n] \to [n]$ with probability more than $\frac{1}{3}$. However, if the distribution D is not β-close to VC-dimension d, the algorithm may output Fail *with any probability.*

Remark 2. Note that α above does not appear anywhere outside the expression for ε, and hence it is tempting to minimize ε by taking $\alpha = \sqrt{\beta/3}$. However, this is a bad strategy since the number of queries of the algorithm depends on $1/\alpha$. In the common scenario, we would be given β and $\varepsilon \geq 34\sqrt{3\beta}$, and solve for α.

Corollary 5.2 (**Testing properties with bounded VC-dimension).** *Let $d \in \mathbb{N}$ be a constant, and $\mathcal{P}$ be an index-invariant property with VC-dimension d. There exists an algorithm that has sample and query access to an unknown distribution D, takes a parameter $\varepsilon \in (0,1)$, and distinguishes whether $D \in \mathcal{P}$ or D is ε-far from $\mathcal{P}$ with probability at least $2/3$, where the total number of queries made by the algorithm is a function of only d and ε.*

Remark 3. Note that the algorithm for testing the index-invariant property with constant VC-dimension d takes $\exp(d)$ samples, and performs $\exp(\exp(d))$ queries. It turns out that similarly to the case of Test-and-Learn, the dependencies of the sample and query complexities on d are tight, in the sense that there exists a property of VC-dimension d such that testing it requires $2^{\Omega(d)}$ samples, and $\Omega(2^{2^{d-\mathcal{O}(1)}})$ queries. We will construct such a property and prove its lower bound in Section 6.

We will give the proof of Theorem 5.1 in Subsection 5.3.

5.1 A corollary of Theorem 4.2 to prove Theorem 5.1

In this subsection, we first connect the notions of (ζ, δ, r)-clusterablity and being β-close to (α, r)-clusterablity (Definition 4.1) in Lemma 5.4. Then using Lemma 5.4 with our algorithm for learning (ζ, δ, r)-clusterable distributions (Theorem 4.2), we prove Corollary 5.3 regarding learning distributions that are β-close to (α, r)-clusterable. This corollary will be used later to prove Theorem 5.1.

Corollary 5.3 (Learning distributions β-close to (α, r)-clusterable). *Let $n \in \mathbb{N}$. There exists a (non-adaptive) algorithm, that has sample and query access to an unknown distribution D over $\{0,1\}^n$, takes parameters α, β, r as inputs such that $\alpha > \beta$ and $\varepsilon = 17(3\alpha + \beta/\alpha) < 1$ and $r \in \mathbb{N}$, makes a number of queries that only depends on α, β and r, and either reports a full description of a distribution over $\{0,1\}^n$ or reports* Fail*, satisfying both of the following conditions:*

(i) If D is β-close to (α, r)-clusterable, then with probability at least $2/3$, the algorithm outputs a full description of a distribution D' over $\{0,1\}^n$ such that $d_{EM}(D, D'_\sigma) \leq \varepsilon$ for some permutation $\sigma : [n] \to [n]$.

(ii) For any D, the algorithm will not output a distribution D' such that $d_{EM}(D, D'_\sigma) > \varepsilon$ for every permutation $\sigma : [n] \to [n]$, with probability more than $1/3$. However, if the distribution D is not β-close to (α, r)-clusterable, the algorithm may output Fail *with any probability.*

To prove the above corollary, we need the following lemma, that connects the two notions of clusterability, that is, (ζ, δ, r)-clusterablity and being β-close to (α, r)-clusterability (see Definition 4.1).

Lemma 5.4. *Let $\alpha, \beta \in (0,1)$ be such that $\alpha > \beta$, and D be a distribution over $\{0,1\}^n$ that is β-close to being (α, r)-clusterable. Then D is $(3\alpha, r, \beta/\alpha)$-clusterable.*

Proof. Let D_0 be the distribution such that D_0 is (α, r)-clusterable and $d_{EM}(D, D_0) \leq \beta$. Let $\mathcal{C}_1, \ldots, \mathcal{C}_s$ be the partition of the support of D_0 that realizes the (α, r)-clusterability of D_0, and let $\{f_{\mathbf{XY}} : \mathbf{X}, \mathbf{Y} \in \{0,1\}^n\}$ be the flow that realizes $d_{EM}(D, D_0) \leq \beta$.

Let $\mathcal{C} = \bigcup_{i=1}^{s} \mathcal{C}_i$, and $\mathcal{C}_{>\alpha}$ be the set of vectors in $\{0,1\}^n$ that have distance of at least α from all the vectors in $\mathcal{C}$. Now we have the following claim.

Claim 5.5. $D(\mathcal{C}_{>\alpha}) \leq \frac{\beta}{\alpha}$.

Proof. By contradiction, let us assume that $D(\mathcal{C}_{>\alpha}) > \frac{\beta}{\alpha}$. Then we have the following:

$$d_{EM}(D, D_0) \quad \geq \quad \sum_{\mathbf{X} \in \mathcal{C}_{>\alpha}, \mathbf{Y} \in \mathcal{C}} f_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \geq \alpha \cdot D(\mathcal{C}_{>\alpha}) > \beta.$$

This is a contradiction as we have assumed $d_{EM}(D, D_0) \leq \beta$. □

Now for every i, let $\mathcal{C}_i^{\leq\alpha}$ be the vectors that have distance at most α from at least one vector $\mathcal{C}_i$, where $i \in [s]$. Let $\mathcal{C}'_i = \mathcal{C}_i^{\leq\alpha} \setminus \bigcup_{j=1}^{i-1} \mathcal{C}'_j$ for $1 \leq i \leq s$. Now we have the following observation.

Observation 5.6. For any $1 \leq i \leq s$, $d_H(\mathbf{U}, \mathbf{V}) \leq 3\alpha$ for any $\mathbf{U}, \mathbf{V} \in \mathcal{C}'_i$.

Proof. Since $\mathbf{U}, \mathbf{V} \in \mathcal{C}'_i$, let $\mathbf{U}'$ and $\mathbf{V}'$ be the vectors in $\mathcal{C}_i$ such that $d_H(\mathbf{U}, \mathbf{U}') \leq \alpha$, and $d_H(\mathbf{V}, \mathbf{V}') \leq \alpha$. As $\mathbf{U}', \mathbf{V}' \in \mathcal{C}_i$, and D_0 is (α, r)-clusterable, using the triangle inequality, we can say that $d_H(\mathbf{U}, \mathbf{V}) \leq d_H(\mathbf{U}, \mathbf{U}') + d_H(\mathbf{U}', \mathbf{V}') + d_H(\mathbf{V}', \mathbf{V}) \leq 3\alpha$. □

Consider $\mathcal{C}'_0 = \mathcal{C}_{>\alpha}$, and by Claim 5.5, note that $D(\mathcal{C}'_0) \leq \beta/\alpha$. The existence of $\mathcal{C}'_0, \mathcal{C}'_1, \ldots, \mathcal{C}'_s$ as above implies that D is $(3\alpha, r, \beta/\alpha)$-clusterable (see Definition 4.1). □

Proof of Corollary 5.3 using Theorem 4.2 and Lemma 5.4. The algorithm here (say ALG) calls algorithm Test-and-Learn (as described in Algorithm **??**) with parameters $\zeta = \beta/\alpha$ and $\delta = 3\alpha$, and reports the output returned by Test-and-Learn as the output of ALG. Now we prove the correctness of ALG.

Part (i): Here we consider the case where D is β-close to (α, r)-clusterable. By Lemma 5.4, D is (ζ, δ, r)-clusterable. By Theorem 4.2 (i), we get a distribution D' such that $d_{EM}(D, D'_\sigma) \leq 17(\zeta + \delta) = 17(3\alpha + \beta/\alpha) = \varepsilon$ for some permutation $\sigma : [n] \to [n]$, with probability at least $2/3$. This completes the proof of Part (i).

Part (ii): This follows from Theorem 4.2 (ii) along with our choices of $\delta = 3\alpha$ and $\zeta = \beta/\alpha$.

□

5.2 A corollary from VC theory required to prove Theorem 5.1

In this subsection, we recall some definitions from VC-dimension theory, and use a well known result of Haussler ([Hau95]) to obtain Corollary 5.12, which states that if the VC-dimension of a set of vectors V is bounded, then the vectors of V can be covered by bounded number of Hamming balls. This corollary will be crucially used to prove Theorem 5.1 in Subsection 5.3.

Let us start by defining the notion of an α-separated set.

Definition 5.7 (α**-separated set).** Let $\alpha \in (0, 1)$ and $W \subset \{0, 1\}^n$ be a set of vectors. W is said to be *α-separated* if for any two vectors $\mathbf{X}, \mathbf{Y} \in W$, $d_H(\mathbf{X}, \mathbf{Y}) \geq \alpha$.

Now let us define the notion of the α-packing number of a set of vectors.

Definition 5.8 (α**-packing number).** Let $\alpha \in (0, 1)$, and $V \subset \{0, 1\}^n$ be a set of vectors. The *α-packing number* of V, denoted by $\mathcal{M}(\alpha, V)$, is defined as the cardinality of the largest α-separated subset W of V.

Now we define the notion of an α-cover of a set of vectors.

Definition 5.9 (α**-cover).** Let $\alpha \in (0, 1)$ and $V \subset \{0, 1\}^n$ be a set of vectors. A set $M \subseteq V$ is an *α-cover* of V if $V \subseteq \bigcup_{\mathbf{p} \in M} \mathrm{NGB}_\alpha(\mathbf{p})$, where $\mathrm{NGB}_\alpha(\mathbf{p}) := \{\mathbf{q} : d_H(\mathbf{p}, \mathbf{q}) \leq \alpha\}$ denotes the set of vectors that are within Hamming distance α from the vector $\mathbf{p}$.

Now let us consider the following theorem from [Hau95], which says that if the VC-dimension of a set of vectors V is d, then the size of the α-packing number of V, that is, $\mathcal{M}(\alpha, V)$, is bounded by a function of d and α.

Theorem 5.10 (**Haussler's packing theorem [Hau95, Theorem 1]**). *Let $\alpha \in (0,1)$ be a parameter. If the VC-dimension of a set of vectors V is d, then the α-packing number of V is bounded as follows:*

$$\mathcal{M}(\alpha, V) \leq e(d+1)\left(\frac{2e}{\alpha}\right)^d$$

The following observation is immediate.

Observation 5.11. Let $\alpha \in (0,1)$ be a parameter and M be a maximal α-packing of a set of vectors $V \subset \{0,1\}^n$. Then M is also an α-cover of V.

With this observation, along with Theorem 5.10, we get the following bound on the size of a cover of a set of vectors in terms of its VC-dimension.

Corollary 5.12 (**Existence of a small α-cover**). *Let $d \in \mathbb{N}$. If the VC-dimension of a set of vectors V is d, then for all $\alpha \in (0,1)$, there exists a set $M \subseteq V$ such that M is an α-cover of V and $|M| \leq e(d+1)\left(\frac{2e}{\alpha}\right)^d$.*

5.3 Proof of Theorem 5.1 and testing bounded VC-dimension properties

In this subsection, using Corollary 5.3, we prove that any distribution that is β-close to bounded VC-dimension can be learnt (up to permutation) by performing a number of queries that depends only on the VC-dimension d and the proximity parameter ε, and is independent of the dimension of the Hamming cube $\{0,1\}^n$ (Theorem 5.1). The crucial ingredient of the proof is Theorem 5.10, through its Corollary 5.12. From Theorem 5.1, we obtain a tester for testing distribution properties with bounded VC-dimension (Corollary 5.2).

Proof of Theorem 5.1. We call the algorithm ALG corresponding to Corollary 5.3 with D as the input distribution, the same α and β as here, and $r = \lfloor e(d+1)\left(\frac{2e}{\alpha}\right)^d \rfloor$. Note that the output of ALG is either the full description of a distribution D' or FAIL. We output the same output returned by ALG. Now we prove the correctness of this procedure.

(i) Here D is β-close to having VC-dimension d. Let D_0 be the distribution such that D_0 has VC-dimension at most d and $d_{EM}(D, D_0) \leq \beta$. By Corollary 5.12, we can partition the support of D_0 into r parts $\mathcal{C}_1, \ldots, \mathcal{C}_r$ such that $r \leq e(d+1)\left(\frac{2e}{\alpha}\right)^d$ and the Hamming distance between any pair of vectors in the same cluster $\mathcal{C}_i$ is at most α. This means that D_0 is (α, r)-clusterable. So, with probability at least $2/3$, TEST-AND-LEARN outputs a distribution D' such that $d_{EM}(D, D'_\sigma) \leq 17(3\alpha + \beta/\alpha)$ for some permutation $\sigma : [n] \to [n]$, and we are done with the proof.

(ii) This follows from the guarantee provided by TEST-AND-LEARN, see Corollary 4.2 (ii).

□

We conclude this section with the proof of Corollary 5.2 regarding the testing of properties with bounded VC-dimension.

Proof of Corollary 5.2. We call the algorithm (say ALG) corresponding to Theorem 5.1 with the input distribution D, $\alpha = \varepsilon/102$, and $\beta = 0$. Let D' be the output of ALG. We check if there exists a distribution $D'' \in \mathcal{P}$ such that $d_{EM}(D', D'') \leq \varepsilon/2$. If yes, we accept D. Otherwise, we reject D.

Now we argue the correctness. For completeness, let us assume that $D \in \mathcal{P}$, hence D has VC-dimension d. By the guarantee for ALG following Theorem 5.1, with probability at least $2/3$, ALG does not report FAIL, and the output distribution D' by ALG satisfies $d_{EM}(D, D'_\sigma) \leq \varepsilon/2$ for some permutation $\sigma : [n] \to [n]$. Since $\mathcal{P}$ is an index-invariant property, D' and D'_σ have the same distance from the property $\mathcal{P}$. Also, as $D \in \mathcal{P}$, $D_\sigma \in \mathcal{P}$ as well. Hence, there exists a distribution $D'' \in \mathcal{P}$ (here D_σ in particular) such that $d_{EM}(D', D'') \leq \varepsilon/2$, and we accept D with probability at least $2/3$.

For soundness, consider the case where D is ε-far from $\mathcal{P}$. If ALG reports FAIL, we are done. Otherwise, by Theorem 5.1, the output distribution D' is such that $d_{EM}(D, D'_\sigma) \leq \varepsilon/2$ for some permutation $\sigma : [n] \to [n]$. Now we consider any distribution D'' with $d_{EM}(D', D'') \leq \varepsilon/2$ and argue that D'' is not in $\mathcal{P}$. By contradiction, let us assume that $D'' \in \mathcal{P}$. As $\mathcal{P}$ is index-invariant, $D''_\sigma \in \mathcal{P}$. Note that $d_{EM}(D'_\sigma, D''_\sigma) \leq \varepsilon/2$ as $d_{EM}(D', D'') \leq \varepsilon/2$. So, D'_σ is $\varepsilon/2$-close to property $\mathcal{P}$. As $d_{EM}(D, D'_\sigma) \leq \varepsilon/2$, by the triangle inequality, D is ε-close to $\mathcal{P}$, a contradiction. This completes the proof of Corollary 5.2. □

6 Tightness of the bounds for bounded VC-dimension properties

As mentioned in the introduction, our tester for testing a VC-dimension property takes $\exp(d)$ samples, and performs $\exp(\exp(d))$ queries for VC-dimension d. Now we show that there exists an index-invariant property of VC-dimension at most d which requires such sample and query complexities, proving Theorem 1.4.

Theorem 6.1 (Restatement of Theorem 1.4). *Let $d, n \in \mathbb{N}$. There exists an index-invariant property $\mathcal{P}_{\mathsf{vc}}$ with VC-dimension at most d such that any (non-adaptive) tester for $\mathcal{P}_{\mathsf{vc}}$ requires $2^{\Omega(d)}$ samples and $2^{2^{d-\mathcal{O}(1)}}$ queries.*

Since the query complexity of non-adaptive testers can be at most quadratic as compared to adaptive ones (Theorem 1.7), arguing only for non-adaptive testers is sufficient for our purpose. We would like to point out that the property of having support size at most 2^d is a property with VC-dimension bounded by d, for which the authors of [GR22] proved a lower bound of $\Omega(2^{(1-o(1))d})$ samples [GR22, Observation 2.7]. Although the sample lower bound of the property $\mathcal{P}_{\mathsf{vc}}$ of Theorem 6.1 is weaker in comparison to that of the support size property, here we prove both sample and query lower bounds for the same property $\mathcal{P}_{\mathsf{vc}}$. Moreover, $\mathcal{P}_{\mathsf{vc}}$ is defined by being a permutation of a single distribution.

Without loss of generality, in what follows, we assume that d is large enough.

Property $\mathcal{P}_{\mathsf{vc}}$: Let $k = 2^d$ and $\ell = 2^{2^{d-10}}$ be two integers and assume that ℓ divides n. Consider a matrix A of dimension $k \times \ell$ such that the Hamming distance between any pair of column vectors of A is at least $1/3$ [11]. Let D_A be a distribution supported over the vectors $\mathbf{V}_1, \ldots, \mathbf{V}_k$ such that, for every $i \in [k]$, the following holds:

[11] One way to construct such a matrix is to select 2^{d-10} vectors from $\{0,1\}^{2^d}$ uniformly at random, and let the columns of A be the set of all their linear combinations over the field $\mathbb{Z}_2$.

- $\mathbf{V}_i$ is the n/ℓ times "blow-up" of the i-th row of A, that is, for $j \in [\ell]$ and j' with $(j-1) \cdot \frac{n}{\ell} < j' \leq j \cdot \frac{n}{\ell}$, $(\mathbf{V}_i)_{j'} = a_{ij}$, where a_{ij} denotes the element of the matrix A present in the i-th row and the j-th column.
- $D_A(\mathbf{V}_i) = \frac{1}{k} = \frac{1}{2^d}$.

Now we are ready to define the property $\mathcal{P}_{\mathsf{vc}}$.

$$\mathcal{P}_{\mathsf{vc}} = \{D : D = D_A^\sigma \text{ for some permutation } \sigma : [n] \to [n]\}.$$

Now we have the following observation.

Observation 6.2. The VC-dimension of $\mathcal{P}_{\mathsf{vc}}$ is at most d.

This follows from the fact that the support size of the distribution D_A is 2^d. We will prove first the query complexity lower bound, and then prove the (easier) sample complexity lower bound.

Query complexity lower bound: Let us define the first pair of hard distributions over distributions over $\{0,1\}^n$, that is, D_{yes} and D_{no}.

Distribution D_{yes}: We choose a permutation $\sigma : [n] \to [n]$ uniformly at random, and pick the distribution D_A^σ over $\{0,1\}^n$.

The distribution D_{no} is constructed from the matrix A that is used to define D_{yes} as follows:

Distribution D_{no}: We first choose $\ell' = 2^{2^{d-20}}$ many column vectors uniformly at random from A and let B be the resulting matrix of dimension $k \times \ell'$. Let D_B be the distribution supported over the vectors $\mathbf{W}_1, \ldots, \mathbf{W}_k$ such that, for every $i \in [k]$, the following holds:

- $\mathbf{W}_i$ is the n/ℓ' times blow-up of the i-th row of B, that is, for $j \in [\ell']$ and j' with $(j-1) \cdot \frac{n}{\ell'} < j' \leq j \cdot \frac{n}{\ell'}$, $(\mathbf{W}_i)_{j'} = b_{ij}$, where b_{ij} denotes the element of matrix B present in the i-th row and the j-th column.
- $D_{no}(\mathbf{W}_i) = \frac{1}{k} = \frac{1}{2^d}$.

We choose a permutation $\sigma : [n] \to [n]$ uniformly at random, and pick the distribution D_B^σ over $\{0,1\}^n$.

Lemma 6.3. *D_{yes} is supported over $\mathcal{P}_{\mathsf{vc}}$ and D_{no} is supported over distributions that are $1/8$-far from $\mathcal{P}_{\mathsf{vc}}$.*

Proof. Following the definition of $\mathcal{P}_{\mathsf{vc}}$ and D_{yes}, it is clear that D_{yes} is supported over $\mathcal{P}_{\mathsf{vc}}$. To prove the claim about D_{no}, consider the following definition and observation.

Definition 6.4. Let us consider a distribution D over $\{0,1\}^n$. A matrix M of dimension $s \times n$ is said to be a *corresponding* matrix of D if D is the distribution resulting from picking uniformly at random a row of M. [12] For a permutation $\pi : [s] \to [s]$, M^π denotes the matrix obtained by permuting the rows of M according to the permutation π, that is, the $\pi(i)$-th row of M^π is same as the i-th row of M for every $i \in [s]$.

[12] Note that, if M has no duplicate rows, then D is a uniform distribution over its support.

Note that if M is a corresponding matrix of D with s rows and s' is a multiple of s, then the matrix M' constructed by repeating every row of M s'/s many times is also a corresponding matrix of D.

Now the following observation connects the Earth Mover Distance between two distributions with the Hamming distance between their corresponding matrices.

Claim 6.5. *Let D_1 and D_2 be two distributions over $\{0,1\}^n$. Also, let L and M be corresponding matrices of D_1 and D_2, respectively, both of dimension $s \times n$. Then the Earth Mover Distance between D_1 and D_2 is the same as the minimum Hamming distance between L and M over all row permutations.*

Formally, let the Hamming distance between L and M be defined as

$$d_H(L, M) = \frac{|\{(i,j) \in [s] \times [n] \ : \ l_{ij} \neq m_{ij}\}|}{s \cdot n}$$

Then

$$d_{EM}(D_1, D_2) = \min_{\pi:[s]\to[s]} d_H(L^\pi, M).$$

Proof. We first note that any solution $f_{\mathbf{XY}}$ for the EMD between D_1 and D_2 can be translated to a doubly stochastic matrix S of dimension $s \times s$ as follows:

For every i, let $\mathbf{L}_i$ be the i-th row of L and l_i be the number of rows of L that are identical to $\mathbf{L}_i$. Similarly, let $\mathbf{M}_i$ be the i-th row of M and m_i be the number of rows of M that are identical $\mathbf{M}_i$. To construct the matrix S, we set the value of its entry at i-th row and j-th column as follows:

$$s_{ij} = \frac{f_{\mathbf{L}_i\mathbf{M}_j} \cdot s}{l_i \cdot m_j}$$

Now we claim that the matrix S defined above is a doubly stochastic matrix.

Observation 6.6. The matrix S defined above is doubly stochastic.

Proof. We will prove that the every row of S sum to 1, and omit the identical proof for the columns of S. Note that if we sum the i-th row of S, we obtain the following:

$$\sum_{j=1}^{s} s_{ij} = \sum_{j=1}^{s} \frac{f_{\mathbf{L}_i\mathbf{M}_j} \cdot s}{l_i \cdot m_j} = \sum_{\mathbf{Y}\in\mathrm{Supp}(D_2)} \frac{f_{\mathbf{L}_i\mathbf{Y}} \cdot s}{l_i} = \frac{D_1(\mathbf{L}_i) \cdot s}{l_i} = 1$$

This completes the proof of the observation. □

Now we will apply the Birkhoff-Newmann theorem (see [Bir46, VN53]), which states that the doubly stochastic matrix S defined above can be expressed as a weighted average of permutation matrices. By translating the EMD expression from $f_{\mathbf{XY}}$ to S and using an averaging argument, we can infer that there exists a permutation π (among those in the representation of S) such that $d_H(L^\pi, M)$ is equal to $d_{EM}(D_1, D_2)$. This completes the proof of the claim. □

Note that D_{no} is supported over the set of distributions D_B^σ for any permutation σ and any matrix B which consists of $2^{2^{d-20}}$ columns of A. We will be done by showing that the Earth Mover Distance between D and D_B^σ is at least $1/8$, where $D \in \mathcal{P}_{\mathsf{vc}}$, $\sigma : [n] \to [n]$ is any permutation, and B is any matrix with $2^{2^{d-20}}$ columns.

Note that both D and D_B^σ admit respective corresponding matrices L and M, respectively, both of dimension $2^d \times n$, where the rows of L are the vectors $\mathbf{V}_i$, and the rows of M are the respective permutations of the vectors $\mathbf{W}_i$. By Claim 6.5, we note that:

$$d_{EM}(D_B^\sigma, D) = \min_{\pi:[2^d]\to[2^d]} d_H(L^\pi, M).$$

The following claim will imply that $d_{EM}(D_B^\sigma, D) \geq 1/8$.

Claim 6.7. *For any permutation $\pi : [2^d] \to [2^d]$, $d_H(L^\pi, M)$ is at least $1/8$.*

Proof. Let us partition the index set $[n]$ into ℓ' many equivalence classes $C_1, \ldots, C_{\ell'}$ such that two indices of $[n]$ belong to the same equivalence class if the corresponding column vectors in L^π are identical. Observe that

$$d_H(L^\pi, M) = \frac{\sum_{i\in[\ell']} \sum_{j\in C_i} d_H(\mathbf{L}_j^\pi, \mathbf{M}_j) \cdot k}{k \cdot n} = \frac{\sum_{i\in[\ell']} \sum_{j\in C_i} d_H(\mathbf{L}_j^\pi, \mathbf{M}_j)}{n},$$

where $\mathbf{L}_j^\pi$ and $\mathbf{M}_j$ denote the j-th column vectors of L^π and M, respectively.

Hence we will be done by showing $\sum_{j\in C_i} d_H\left(\mathbf{L}_j^\pi, \mathbf{M}_j\right) \geq \frac{n}{8\ell'}$, for every $i \in [\ell']$.

Note that $|C_i| = \frac{n}{\ell'}$. Also, all the columns in $\{\mathbf{L}_j^\pi : j \in C_i\}$ are identical. Consider a column vector $\mathbf{v} \in \{0,1\}^k$. Observe that there can be at most $\frac{n}{\ell}$ many columns in $\{\mathbf{M}_j : j \in C_i\}$ that are $1/7$-close to $\mathbf{v}$. This follows from the construction of $\mathcal{P}_{\mathsf{vc}}$, which implies that for every column $\mathbf{M}_j$ of M, there are no more than $n/\ell - 1$ many other columns of L^π whose distance from $\mathbf{M}_j$ is at most $2/7 < 1/3$.

So, in the expression $\sum_{j\in C_i} d_H\left(\mathbf{L}_j^\pi, \mathbf{M}_j\right)$, there are at least $(\frac{n}{\ell'} - \frac{n}{\ell})$ many terms that are at least $1/7$. Hence, $\sum_{j\in C_i} d_H\left(\mathbf{L}_j^\pi, \mathbf{M}_j\right) \geq \ell' \cdot \frac{1}{7}\left(\frac{n}{\ell'} - \frac{n}{\ell}\right) \geq \frac{n}{8\ell'}$. □

The above two claims conclude the proof of Lemma 6.3. □

Lemma 6.8 (Query complexity lower bound part of Theorem 6.1). *Any (non-adaptive) tester, that has sample and query access to either D_{yes} or D_{no} and performs $2^{2^{d-\omega(1)}}$ queries, can not distinguish between D_{yes} and D_{no}.*

Proof. Let A' and B' be the matrices of dimension $k \times n$ such that the i-th row of A' corresponds to the vector $\mathbf{V}_i^\sigma$ (for the permutation σ drawn according to D_{yes}) and the i-th row of B' corresponds to the vector $\mathbf{W}_i^\sigma$ (for the permutation σ drawn according to D_{no}), where $i \in [k]$.

Let us divide the index set $[n]$ into ℓ equivalence classes $C_1, \ldots, C_\ell$ such that two indices belong to the same equivalence class if the corresponding column vectors in A' are identical. Similarly, let us divide the index set $[n]$ into ℓ' equivalence classes $C'_1, \ldots, C'_{\ell'}$ such that two indices belong to the same equivalence class if the corresponding column vectors in B' are identical.

Let $Q \subseteq [n]$ be the set of all distinct indices queried by the tester to any sample (that is, the union of the sets $J_1, \ldots, J_s$ as they appear in Definition 3.11). If $|Q| = 2^{2^{d-\omega(1)}}$, then the probability that there exist two indices in Q that belong to the same C_i or the same C'_i is $o(1)$. Observe that, conditioned on the event that Q does not contain two indices from the same equivalence class C_i or C'_i, the distributions over the responses to the queries of the tester are identical for both D_{yes} and D_{no}. The reason is that in both the cases of D_{yes} and D_{no}, the distribution over the responses is identical to the one derived from picking a uniformly random subset of size $|Q|$ of the columns of the matrix A, and taking uniformly independent samples of the rows of the resulting matrix. □

Now we will prove the sample complexity lower bound for testing $\mathcal{P}_{\mathsf{vc}}$.

Sample complexity lower bound: Let us define the second pair of hard distributions over distributions over $\{0,1\}^n$, D'_{yes} and D'_{no}.

Distribution D'_{yes}: Identically to D_{yes} above, we choose a permutation $\sigma : [n] \to [n]$ uniformly at random, and pick the distribution D^{σ}_{A} over $\{0,1\}^n$.

The distribution D'_{no} is constructed from the matrix A used to define D'_{yes} as follows:

Distribution D'_{no}: We first choose $k' = 2^{d-20}$ many row vectors uniformly at random from A and construct a matrix B' of dimension $k' \times \ell$. Let $D_{B'}$ be the distribution supported over the vectors $\mathbf{W}'_1, \ldots, \mathbf{W}'_{k'}$ such that, for every $i \in [k']$, the following hold:

- $\mathbf{W}'_i$ is the n/ℓ times blow-up of the i-th row of B', that is, for $j \in [\ell]$ and j' with $(j-1) \cdot \frac{n}{\ell} < j' \leq j \cdot \frac{n}{\ell}$, $(\mathbf{W}'_i)_{j'} = b_{ij}$, where b_{ij} denotes the element of matrix B' present in the i-th row and the j-th column.
- $D_{no}(\mathbf{W}'_i) = \frac{1}{k'} = \frac{1}{2^{d-20}}$.

We choose a permutation $\sigma : [n] \to [n]$ uniformly at random, and pick the distribution $D^{\sigma}_{B'}$ over $\{0,1\}^n$.

Lemma 6.9. *D'_{yes} is supported over $\mathcal{P}_{\mathsf{vc}}$ and D'_{no} is supported over distributions that are $1/8$-far from $\mathcal{P}_{\mathsf{vc}}$.*

Proof. Following the definition of $\mathcal{P}_{\mathsf{vc}}$ and D'_{yes}, it is clear that D'_{yes} is supported over $\mathcal{P}_{\mathsf{vc}}$. To prove the claim about D'_{no}, we will apply Claim 6.5.

Note that D'_{no} is supported over the set of distributions $D^{\sigma}_{B'}$ for any permutation σ and any matrix B' which consists of 2^{d-20} rows of A. We will be done by showing the Earth Mover Distance between D and $D^{\sigma}_{B'}$ is at least $1/8$, where $D \in \mathcal{P}_{\mathsf{vc}}$ and $\sigma : [n] \to [n]$ be any permutation, and B' is any matrix with 2^{d-20} distinct rows.

Let L and M be corresponding matrices of D and $D^{\sigma}_{B'}$, respectively, of dimension $k \times n$, where $k = 2^d$ (where the rows of L are the vectors $\mathbf{V}_i$, and the rows of M are 2^{20}-fold repetitions of the respective permutations of the vectors $\mathbf{W'}_i$). By Claim 6.5, we know that

$$d_{EM}\left(D^{\sigma}_{B'}, D\right) = \min_{\pi:[k]\to[k]} d_H(L^{\pi}, M).$$

Thus, the following claim will imply that $d_{EM}(D^{\sigma}_{B'}, D) \geq 1/8$.

Claim 6.10. *For any permutation $\pi : [2^d] \to [2^d]$, $d_H(L^{\pi}, M)$ is at least $1/8$.*

Proof. Our proof will follow a similar vain to that of Claim 6.7. Let us first partition the index set $[n]$ into ℓ' many equivalence classes $C_1, \ldots, C_{\ell'}$ such that two indices of $[n]$ belong to the same equivalence class if the corresponding column vectors in L^{π} are identical. Observe that

$$d_H(L^{\pi}, M) = \frac{\sum_{i\in[\ell']} \sum_{j\in C_i} d_H(\mathbf{L}^{\pi}_j, \mathbf{M}_j) \cdot k}{k \cdot n} = \frac{\sum_{i\in[\ell']} \sum_{j\in C_i} d_H(\mathbf{L}^{\pi}_j, \mathbf{M}_j)}{n},$$

where $\mathbf{L}^{\pi}_j$ and $\mathbf{M}_j$ denote the j-th column vectors of L^{π} and M, respectively.

Since B' has only 2^{d-20} distinct rows, the number of its equivalence classes is bounded by $\ell' = 2^{2^{d-20}}$. Note that unlike the proof of the query lower bound, the sizes of the equivalence classes here may be different from each other. Also, note that the sizes of the equivalence classes of L are n/ℓ, as $D \in \mathcal{P}_{\mathsf{vc}}$. Thus we have the following:

$$d_H(L^\pi, M) \geq \frac{1}{7} \cdot \frac{\sum_{i=1}^{\ell'} \max\{0, |C_i| - n/\ell\}}{n} \geq \frac{1}{7} \cdot \left(1 - \frac{1}{2^{10}}\right) \cdot n > \frac{1}{8} n.$$

The inequality follows from the facts that $\ell = 2^{2^{d-10}}$ and $\ell' = 2^{2^{d-20}}$, and the columns of M corresponding to each C_i can be $1/7$-close to at most n/ℓ columns of L. □

This concludes the proof of Lemma 6.9. □

The sample lower bound for testing $\mathcal{P}_{\mathsf{vc}}$ now follows from the following lemma.

Lemma 6.11 (Sample complexity lower bound part of Theorem 6.1). *Any tester that takes at most $2^{o(d)}$ samples from the input distribution can not distinguish between the distributions D'_{yes} and D'_{no}.*

Proof. Let $\mathcal{S}$ be the set of samples taken by the algorithm. Note that if $|\mathcal{S}| = 2^{o(d)}$, then the probability that $\mathcal{S}$ contains two samples of the same $\mathbf{V}_i$ or the same $\mathbf{W}'_i$ is $o(1)$. Conditioned on the event that $\mathcal{S}$ does not contain two samples from the same vector ($\mathbf{V}_i$ or $\mathbf{W}'_i$), even if the tester queries the samples of $\mathcal{S}$ in their entirety, the distributions over the responses to the queries of the tester are identical for both D'_{yes} and D'_{no}. This follows from the fact that the distribution over the responses is identical to a distribution obtained by drawing uniformly without repetitions a sequence of row vectors from $\mathbf{V}_1, \ldots, \mathbf{V}_{2^d}$, and querying the row vectors completely. This completes the proof. □

7 Exponential gap between adaptive and non-adaptive testers for general properties

In this section, we prove that unlike the index-invariant properties, there can be an exponential gap between the query complexities of adaptive and non-adaptive algorithms for non-index-invariant properties. In Subsection 7.1, we prove an exponential upper bound on the relation between the non-adaptive and adaptive query complexities of general properties. In Subsection 7.2, we provide an exponential separation between them, and also use the same method to prove Proposition 1.5.

7.1 Relation between adaptive and non-adaptive testers for general properties

Let us assume that $\mathcal{A}$ is the adaptive algorithm that ε-tests $\mathcal{P}$ using s samples $\{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$ and q queries, along with tossing some random coins. Before directly proceeding to the description of the non-adaptive algorithm, let us first consider the following observation.

Observation 7.1. For any given outcome sequence of the random coin tosses of $\mathcal{A}$, there are at most $2^q - 1$ possible internal states of $\mathcal{A}$.

Proof. Consider the k-th step of $\mathcal{A}$, where $\mathcal{A}$ queries the j_k-th index of $\mathbf{V}_{i_k}$ for some $i_k \in [s]$, $j_k \in [n]$, and $k \in [q]$. Note that i_1 and j_1 are functions of only the random coins, and i_k and j_k are functions of the random coins, as well as $\mathbf{V}_{i_1}|_{j_1}, \ldots, \mathbf{V}_{i_{k-1}}|_{j_{k-1}}$, where $2 \leq k \leq q$. Due to the 2^{k-1}

possible values of $\mathbf{V}_{i_1}|_{j_1}, \ldots, \mathbf{V}_{i_{k-1}}|_{j_{k-1}}$, there are 2^k possible states of the algorithm $\mathcal{A}$ at Step k, for each $1 \leq k \leq q$. Finally, the state of $\mathcal{A}$ depending on the random coins and the values of $\mathbf{V}_{i_1}|_{j_1}, \ldots, \mathbf{V}_{i_q}|_{j_q}$ will decide the final output. This implies that for any fixed set of outcomes of the random coin tosses used by $\mathcal{A}$, there can be a total of at most $\sum_{i=0}^{q-1} 2^i = 2^q - 1$ internal states, each making one query, as well as 2^q final (non-query-making) states. □

Now we proceed to present the non-adaptive algorithm $\mathcal{A}'$ that simulates $\mathcal{A}$ by using s samples and at most 2^q queries.

Lemma 7.2. *Let $\mathcal{P}$ be any property that is ε-testable by an adaptive algorithm using s samples and q queries. Then $\mathcal{P}$ can be ε-tested by a non-adaptive algorithm using s samples and at most $2^q - 1$ queries, where s and q are integers.*

Proof. Let $\mathcal{A}$ be the adaptive algorithm that ε-tests $\mathcal{P}$ using s samples $\{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$ and q queries. Now we show that a non-adaptive algorithm $\mathcal{A}'$ exists that uses s samples and makes at most $2^q - 1$ queries, such that the output distributions of $\mathcal{A}$ and $\mathcal{A}'$ are identical for any unknown distribution D.

The idea of $\mathcal{A}'$ in a high level is to enumerate all possible internal steps of $\mathcal{A}$, and list all possible queries $\mathcal{Q}$ that might be performed by $\mathcal{A}$. Note that $\mathcal{Q}$ depends on the random coins used by $\mathcal{A}$. We then query all the indices of $\mathcal{Q}$ non-adaptively, and finally simulate $\mathcal{A}$ using the full information at hand, with the same random coins that were used to generate $\mathcal{Q}$. As $\mathcal{A}$ has query complexity q, the number of possible internal states of $\mathcal{A}$ is at most $2^q - 1$, and the query complexity of $\mathcal{A}'$ follows. Now we formalize the above intuition below.

The algorithm $\mathcal{A}'$ has two phases:

Phase 1:

(i) $\mathcal{A}'$ first takes s samples $\mathbf{V}_1, \ldots, \mathbf{V}_s$.

(ii) $\mathcal{A}'$ now tosses some random coins (same as $\mathcal{A}$) and determines the set of all possible indices J_i of $\mathbf{V}_i$ that might be queried by $\mathcal{A}$, for every $i \in [s]$. The sets of indices J_i's are well defined after we fix the random coins, and follows from Observation 7.1.

Thus at the end of Phase 1, $\mathcal{A}'$ has determined s sets of indices $J_1, \ldots, J_s$ of the vectors $\mathbf{V}_1, \ldots, \mathbf{V}_s$ such that $\sum_{i=1}^{s} |J_i| \leq 2^q - 1$. Now $\mathcal{A}'$ proceeds to the second phase of the algorithm.

Phase 2:

(i) For every $i \in [s]$ and $j \in J_i$, query the j-th index of $\mathbf{V}_i$, where J_i denotes the set of indices of $\mathbf{V}_i$ that might be queried at the internal states of $\mathcal{A}$, determined in Phase 1.

(ii) Simulate the algorithm $\mathcal{A}$ using the same random coins used in Phase 1, and report ACCEPT or REJECT according to the output of $\mathcal{A}$.

Note that the set of random coins that are used to determine $J_1, \ldots, J_s$ in Step (ii) of Phase 1 of the algorithm are the same random coins that are used to simulate $\mathcal{A}$ in Step (ii) of Phase 2. Thus the correctness of $\mathcal{A}'$ follows from to the correctness of $\mathcal{A}$ along with Observation 7.1. □

7.2 Exponential separation between adaptive and non-adaptive query complexities

Now we prove that the gap of Lemma 7.2 is almost tight, in the sense that there exists a property such that the adaptive and non-adaptive query complexities for testing it are exponentially separated.

Before proceeding to the proof, let us consider any property $\mathcal{P}$ of strings of length n over the alphabet $\{0,1\}$. Now we describe a related property $1_{\mathcal{P}}$ over distributions as follows:

Property $1_{\mathcal{P}}$: For any distribution $D \in 1_{\mathcal{P}}$, the size of the support of D is 1, and the single string in the support of D satisfies $\mathcal{P}$.

Let us first recall the following result from [GR22], which states that $\widetilde{\mathcal{O}}(\frac{1}{\varepsilon})$ queries are enough to ε-test whether any distribution has support size 1.

Lemma 7.3 (Restatement of Corollary 2.3.1 of [GR22]). *There exists a non-adaptive algorithm that ε-tests whether an unknown distribution D has support size 1 and uses $\widetilde{\mathcal{O}}(\frac{1}{\varepsilon})$ queries, for any $\varepsilon \in (0,1)$.*

We now prove that the query complexity of ε-testing $1_{\mathcal{P}}$ is at least the query complexity of ε-testing $\mathcal{P}$, and can be at most the query complexity of $\frac{\varepsilon}{2}$-testing of $\mathcal{P}$, along with an additional additive factor of $\widetilde{\mathcal{O}}(\frac{1}{\varepsilon})$ for testing whether the distribution has support size 1. The result is formally stated as follows:

Lemma 7.4. *Let q_N and q_A denote the non-adaptive and adaptive query complexities for ε-testing $\mathcal{P}$, respectively. Similarly, let Q_N and Q_A denote the non-adaptive and adaptive query complexities of ε-testing $1_{\mathcal{P}}$, respectively. Then the following hold:*

1. $q_A(\varepsilon) \leq Q_A(\varepsilon) \leq \widetilde{\mathcal{O}}(\frac{1}{\varepsilon}) + \mathcal{O}\left(q_A(\frac{\varepsilon}{2})\right)$ [13].
2. $q_N(\varepsilon) \leq Q_N(\varepsilon) \leq \widetilde{\mathcal{O}}(\frac{1}{\varepsilon}) + \mathcal{O}\left(q_N(\frac{\varepsilon}{2})\right)$.

Proof. We prove here (1), and omit the nearly identical proof of (2).

Proof of $q_A(\varepsilon) \leq Q_A(\varepsilon)$: Consider an adaptive algorithm $\mathcal{A}$ that ε-tests $1_{\mathcal{P}}$ by using $Q_A(\varepsilon)$ queries. We construct an algorithm $\mathcal{A}'$ that ε-tests $\mathcal{P}$ using the same number of queries. Let $\mathbf{V}$ be the unknown string of length n, where we want to test whether $\mathbf{V} \in \mathcal{P}$ or $\mathbf{V}$ is ε-far from $\mathcal{P}$.

Let us define an unknown distribution D' (over the Hamming cube $\{0,1\}^n$) such that we want to distinguish whether $D' \in 1_{\mathcal{P}}$ or D' is ε-far from $1_{\mathcal{P}}$. The distribution D' is defined as follows:

$$D'(\mathbf{X}) = \begin{cases} 1 & \mathbf{X} = \mathbf{V} \\ 0 & \text{otherwise} \end{cases}$$

Observe that $\mathbf{V} \in \mathcal{P}$ if and only if $D' \in 1_{\mathcal{P}}$. Similarly, it is not hard to see that $\mathbf{V}$ is ε-far from $\mathcal{P}$ if and only if D' is ε-far from $1_{\mathcal{P}}$. We simulate the algorithm $\mathcal{A}$ by $\mathcal{A}'$ as follows: when $\mathcal{A}$ takes a sample, $\mathcal{A}'$ does nothing, and when $\mathcal{A}$ queries an index $i \in [n]$ of any sample, $\mathcal{A}'$ queries the index i of $\mathbf{V}$. Finally, $\mathcal{A}'$ provides the output received from the simulation of $\mathcal{A}$.

From the description, it is clear that $\mathcal{A}'$ performs exactly $Q_{\mathcal{A}}(\varepsilon)$ queries and is indeed simulated by running $\mathcal{A}$ over D'.

[13] We are using $\mathcal{O}(\cdot)$ as we are amplifying the success probability of the tester for the property $\mathcal{P}$ to 9/10 as compared to the usual success probability of 2/3.

Proof of $Q_A(\varepsilon) \leq \widetilde{\mathcal{O}}\left(\frac{1}{\varepsilon}\right) + \mathcal{O}\left(q_A(\frac{\varepsilon}{2})\right)$: Let us consider an adaptive algorithm $\mathcal{A}_1$ that $\frac{\varepsilon}{2}$-tests $\mathcal{P}$ using $\mathcal{O}\left(q_A(\frac{\varepsilon}{2})\right)$ queries to the unknown string $\mathbf{X} \in \{0,1\}^n$, with success probability at least $\frac{9}{10}$. Now we design an adaptive algorithm $\mathcal{A}'_1$ that ε-tests $1_{\mathcal{P}}$ using $\widetilde{\mathcal{O}}\left(\frac{1}{\varepsilon}\right) + \mathcal{O}\left(q_A(\frac{\varepsilon}{2})\right)$ many queries.

Algorithm $\mathcal{A}'_1$: Assume that D is the distribution that we want to ε-test for $1_{\mathcal{P}}$. The algorithm $\mathcal{A}'_1$ performs the following steps:

(i) Run the tester corresponding to Lemma 7.3 to $\frac{\varepsilon}{20}$-test whether D has support size 1, with success probability at least $\frac{9}{10}$. If the tester decides that D has support size 1, then go to the next step. Otherwise, REJECT.

(ii) Take one more sample from D and let it be $\mathbf{U} \in \{0,1\}^n$. Run algorithm $\mathcal{A}_1$ to $\frac{\varepsilon}{2}$-test $\mathcal{P}$ considering $\mathbf{X} = \mathbf{U}$ as the unknown string. If $\mathcal{A}_1$ accepts, ACCEPT. Otherwise REJECT.

Note that the query complexity for performing Step (i) is $\widetilde{\mathcal{O}}(\frac{1}{\varepsilon})$, which follows from Lemma 7.3. Additionally, the number of queries performed in Step (ii) is $\mathcal{O}\left(q_A(\frac{\varepsilon}{2})\right)$, which follows from the assertion of the lemma. Thus, the algorithm $\mathcal{A}'_1$ performs $\widetilde{\mathcal{O}}(\frac{1}{\varepsilon}) + \mathcal{O}\left(q_A(\frac{\varepsilon}{2})\right)$ many queries in total.

Now we will argue the correctness of $\mathcal{A}'_1$. For completeness, assume that $D \in 1_{\mathcal{P}}$. Let $\mathbf{V} \in \{0,1\}^n$ be the string such that $D(\mathbf{V}) = 1$ and $\mathbf{V} \in \mathcal{P}$. Note that, by Lemma 7.3, $\mathcal{A}'_1$ proceeds to Step (ii) with probability at least $\frac{9}{10}$. In Step (ii), $\mathcal{A}'$ sets $\mathbf{U} = \mathbf{V}$, and runs algorithm $\mathcal{A}_1$ to $\frac{\varepsilon}{2}$-test $\mathcal{P}$ considering $\mathbf{X} = \mathbf{V}$ as the unknown string. Since $\mathbf{V} \in \mathcal{P}$, by the assumption on the algorithm $\mathcal{A}_1$, $\mathcal{A}'_1$ accepts with probability at least $\frac{9}{10}$, given that $\mathcal{A}'_1$ does not report REJECT in Step (i). Thus, by the union bound, $\mathcal{A}'_1$ accepts D with probability at least $\frac{4}{5}$.

Now consider the case where D is ε-far from $1_{\mathcal{P}}$. If D is $\frac{\varepsilon}{20}$-far from having support size 1, $\mathcal{A}'_1$ reports REJECT in Step (i) with probability at least $\frac{9}{10}$, and we are done. So, assume that D is $\frac{\varepsilon}{20}$-close to having support size 1. Then there exists a distribution D' with support size 1, and the distance between D and D' is at most $\frac{\varepsilon}{20}$. Let us assume that D' is supported on the string $\mathbf{V}$. By the Markov inequality, this implies that with probability at least $\frac{4}{5}$, a string $\mathbf{U}$ sampled according D will be $\frac{9\varepsilon}{20}$-close to $\mathbf{V}$.

(i) If $\mathbf{V}$ is $\frac{19\varepsilon}{20}$-close to $\mathcal{P}$, using the triangle inequality, this implies that D is ε-close to $1_{\mathcal{P}}$, which is a contradiction.

(ii) Now consider the case where $\mathbf{V}$ is $\frac{19\varepsilon}{20}$-far from $\mathcal{P}$. Recall that with probability at least $\frac{4}{5}$, the sample $\mathbf{U}$ taken at Step (ii) above is $\frac{9\varepsilon}{20}$-close to $\mathbf{V}$. As we are considering the case where $\mathbf{V}$ is $\frac{19\varepsilon}{20}$-far from $\mathcal{P}$, using the triangle inequality, $\mathbf{U}$ is $\frac{\varepsilon}{2}$-far from $\mathcal{P}$ with the same probability. In this case, the algorithm will REJECT in Step (ii), with probability at least $\frac{9}{10}$. Together, this implies that the algorithm will REJECT the distribution D, with probability at least $\frac{7}{10}$.

□

In the following, we will construct the property $\mathcal{P}_{Pal}$ of strings over the alphabet $\{0,1,2,3\}$. It will then be encoded as a property of strings over $\{0,1\}$ by using two bits per letter.

Property $\mathcal{P}_{Pal}$: A string $\mathbf{S}$ of length n is in $\mathcal{P}_{Pal}$ if $\mathbf{S} = \mathbf{XY}$, where $\mathbf{X}$ is a palindrome over the alphabet $\{0,1\}$, and $\mathbf{Y}$ is a palindrome over the alphabet $\{2,3\}$.

There is an exponential gap between the query complexities of adaptive and non-adaptive algorithms to ε-test $\mathcal{P}_{Pal}$. The result is stated as follows:

Lemma 7.5. *There exists an adaptive algorithm that ε-tests $\mathcal{P}_{Pal}$ by performing $\mathcal{O}(\log n)$ queries for any $\varepsilon \in (0,1)$. However, there exists an $\varepsilon \in (0,1)$ such that $\Omega(\sqrt{n})$ non-adaptive queries are necessary to ε-test $\mathcal{P}_{Pal}$.*

Proof. The lower bound proof (using Yao's lemma), which we omit here, is nearly identical to the one from [AKNS99] (see Theorem 2 therein).

Let us assume that $\mathbf{V}$ is the string that we want to ε-test for $\mathcal{P}_{Pal}$. The adaptive algorithm to ε-test $\mathcal{P}_{Pal}$ uses binary search, and is described below:

(i) Use binary search for an index of $\mathbf{V}$ that has "value 1.5" (which is not present in the input). This returns an index $0 \leq i \leq n$, such that (a) $\mathbf{V}_i \in \{0,1\}$ unless $i = 0$, and (b) $\mathbf{V}_{i+1} \in \{2,3\}$ unless $i = n$.

(ii) Repeat $\mathcal{O}(\frac{1}{\varepsilon})$ times:

  (a) Sample an index $j \in [n]$ uniformly at random.

  (b) If $j \leq i$, then query $\mathbf{V}_j$ and $\mathbf{V}_{i+1-j}$. REJECT if they are not both equal to the same value in $\{0,1\}$.

  (c) Otherwise query $\mathbf{V}_j$ and $\mathbf{V}_{n+i+1-j}$. REJECT if they are not both equal to the same value in $\{2,3\}$.

(iii) If the input has not been rejected till now, ACCEPT.

We first argue the completeness of the algorithm. Assume that $\mathbf{V}$ is a string such that $\mathbf{V} \in \mathcal{P}_{Pal}$, and i is the index returned by Step (i) of the algorithm. As $\mathbf{V} = \mathbf{XY}$ for some palindrome $\mathbf{X}$ over $\{0,1\}$ and palindrome $\mathbf{Y}$ over $\{2,3\}$, the index i will be equal to $|\mathbf{X}|$. This implies that the algorithm will ACCEPT $\mathbf{V}$ with probability 1.

Now consider the case where $\mathbf{V}$ is ε-far from $\mathcal{P}_{Pal}$. We call an index j *violating* if it does not satisfy the condition appearing either in Step (ii)(b) or Step (ii)(c) above, where i is the index returned in Step (i). The number of violating indices is at least εn, because otherwise we can change the violating indices such that the modified input is a string of the form $\mathbf{XY}$ following the definition of $\mathcal{P}_{Pal}$, where $|\mathbf{X}| = i$. Since the loop in Step (ii) runs for $\mathcal{O}(\frac{1}{\varepsilon})$ times, we conclude that with probability at least $\frac{2}{3}$ at least one such violating index will be found. So, the algorithm will REJECT $\mathbf{V}$ with probability at least $\frac{2}{3}$. □

Now we are ready to formally state and prove the main result of this section.

Proposition 7.6. *There exists a property of distributions over strings that can be ε-tested adaptively using $\mathcal{O}(\log n)$ queries for any $\varepsilon \in (0,1)$, but $\Omega(\sqrt{n})$ queries are necessary for any non-adaptive algorithm to ε-test it for some $\varepsilon \in (0,1)$.*

Proof. Consider the property $1_{\mathcal{P}_{Pal}}$. From Lemma 7.5, we know that $q_A(\frac{\varepsilon}{2}) = \mathcal{O}(\log n)$, for any fixed $\varepsilon \in (0,1)$. Using the upper bound of Lemma 7.4, we conclude that $Q_A(\varepsilon) = \mathcal{O}(\log n)$, for any fixed $\varepsilon \in (0,1)$, ignoring the additive $\widetilde{\mathcal{O}}(\frac{1}{\varepsilon})$ term.

On the other hand, according to Lemma 7.5, $q_N(\varepsilon) = \Omega(\sqrt{n})$ for some $\varepsilon \in (0,1)$. Thus, following Lemma 7.4, we conclude that $Q_N(\varepsilon) = \Omega(\sqrt{n})$ holds for some $\varepsilon \in (0,1)$. Together, Proposition 7.6 follows. □

Now we present a sketch of a proof of Proposition 1.5, which shows that for a property to be constantly testable, it is not sufficient that the property has constant VC-dimension, unless it is index-invariant as well.

Proposition 7.7 (**Restatement of Proposition 1.5**). *There exists a non-index-invariant property $\mathcal{P}$ such that any distribution $D \in \mathcal{P}$ has VC-dimension $O(1)$ and the following holds. There exists a fixed $\varepsilon > 0$, such that distinguishing whether $D \in \mathcal{P}$ or D is ε-far from $\mathcal{P}$, requires $\Omega(n)$ queries, where the distributions in the property $\mathcal{P}$ are defined over the n-dimensional Hamming cube $\{0,1\}^n$.*

Proof. Note that the VC-dimension of $1_{\mathcal{P}}$ is 0, where $1_{\mathcal{P}}$ is the property corresponding to $\mathcal{P}$ as defined before. String properties which are hard to test, for which there is a fixed $\varepsilon > 0$ such that ε-testing them requires $\Omega(n)$ queries, are known to exist. Examples are properties studied in the work of Ben-Eliezer, Fischer, Levi and Rothblum ([BFLR20]), and in the work of Ben-Sasson, Harsha and Raskhodnikova ([BHR05]). Defining $1_{\mathcal{P}}$ for such a property $\mathcal{P}$ provides us the example proving Proposition 7.7. □

8 Quadratic gap between adaptive and non-adaptive testers for index-invariant properties

In this section, we first prove Theorem 1.7 in Subsection 8.1, that is, there can be at most a quadratic gap between the query complexities of adaptive and non-adaptive algorithms for testing index-invariant properties. Then in Subsection 8.2, we prove Theorem 1.8, that is, we demonstrate a quadratic separation between them, which is one of the main results of the paper and the main content of this section.

8.1 Quadratic relation between adaptive and non-adaptive testers for index-invariant properties

Theorem 8.1 (**Restatement of Theorem 1.7**). *Let $\mathcal{P}$ be any index-invariant property that is ε-testable by an adaptive algorithm using s samples and q queries. Then $\mathcal{P}$ can be ε-tested by a non-adaptive algorithm using s samples and $sq \leq q^2$ queries, where s and q are integers.*

Proof. The main idea of the proof is to start with an adaptive algorithm $\mathcal{A}$ as stated above, and then argue for another semi-adaptive algorithm $\mathcal{A}'$ with sample complexity s but query complexity qs, such that the output distributions of $\mathcal{A}$ and $\mathcal{A}'$ are the same for any unknown distribution D. Finally, we construct a non-adaptive algorithm $\mathcal{A}''$ such that (i) the sample and query complexities of $\mathcal{A}''$ are the same as that of $\mathcal{A}'$, and (ii) the probability bounds of accepting and rejecting distributions depending on their distances to $\mathcal{P}$ are preserved from $\mathcal{A}'$ to $\mathcal{A}''$. Now we proceed to formalize this argument.

Let $\mathcal{A}$ be the adaptive algorithm that ε-tests $\mathcal{P}$ using s samples $\{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$ and q queries. Now we show that a two phase algorithm $\mathcal{A}'$ exists that takes s samples $\{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$ and proceeds as follows:

Phase 1: In this phase, $\mathcal{A}'$ queries in an adaptive fashion. If $\mathcal{A}$ queries the j_k-th index of $\mathbf{V}_{i_k}$ at its k-th step, for some $i_k \in [s]$ and $j_k \in [n]$, then we perform the following steps:

(i) If $\mathcal{A}'$ has queried the j_k-th index of all the samples before this step, then we reuse the queried value.

(ii) Otherwise, we query the j_k-th index from all the samples $\{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$.

Phase 2: Let $\mathcal{Q} \subset [n]$ be the set of indices queried by $\mathcal{A}'$ while running the q querying steps of $\mathcal{A}$. If $|\mathcal{Q}| < q$, we arbitrarily pick $t = q - |\mathcal{Q}|$ distinct indices $\{j'_1, \ldots, j'_t\}$, disjoint from the set of indices $\mathcal{Q}$. We query the set of indices $j'_1, \ldots, j'_t$ from the entire set of sampled vectors $\mathbf{V}_1, \ldots, \mathbf{V}_s$.

The output (ACCEPT or REJECT) of $\mathcal{A}'$ is finally set to that of $\mathcal{A}$, and in particular depends only on the answers to the queries made in the first phase.

Now we have the following observation regarding the query complexity of $\mathcal{A}'$, which will be used to argue the query complexity of the non-adaptive algorithm later.

Observation 8.2. $\mathcal{A}'$ uses s samples and performs exactly qs queries. Moreover, for any distribution D, the output distribution of $\mathcal{A}'$ is the same as that of $\mathcal{A}$.

Let us assume that $\mathcal{A}'$ proceeds in q steps by querying indices $\ell_1, \ldots, \ell_q \in [n]$ in each of the s samples $\mathbf{V}_1, \ldots, \mathbf{V}_s$ (when the unknown distribution is D). Equivalently, we can think that the algorithm proceeds in q steps, where in Step k ($k \in [q]$), we query the ℓ_k-th index of $\{\mathbf{V}_1, \ldots, \mathbf{V}_s\}$, such that ℓ_k depends on $\ell_1, \ldots, \ell_{k-1}$, where $2 \leq k \leq q$.

Let us now consider an uniformly random permutation $\sigma : [n] \to [n]$ (unknown to $\mathcal{A}'$). Assume that the unknown distribution is D_σ instead of D. As $\mathcal{P}$ is index-invariant, we can assume that the algorithm $\mathcal{A}'$ runs on D_σ for q steps as follows. In Step k, $\mathcal{A}'$ queries the $\sigma(\ell_k)$-th index of each of the s samples, for $k \in [q]$. Now we have the following observation regarding the distribution of the indices queried, which follows from σ being uniformly random.

Observation 8.3. $\sigma(\ell_1)$ is uniformly distributed over $[n]$, and $\sigma(\ell_k)$ is uniformly distributed over $[n] \setminus \{\sigma(\ell_1), \ldots, \sigma(\ell_{k-1})\}$, where $2 \leq k \leq q$. Moreover, this holds even if we condition on the values $\ell_1, \ldots, \ell_k$ as well as $\sigma(\ell_1), \ldots, \sigma(\ell_{k-1})$.

Now the algorithm $\mathcal{A}''$ works as follows:

- First take a uniformly random permutation $\sigma : [n] \to [n]$.
- Run $\mathcal{A}'$ over D_σ instead of D.

From the above description, it does not immediately follow that $\mathcal{A}''$ is a non-adaptive algorithm. But from the description along with Observation 8.3, it follows that $\mathcal{A}''$ is the same as the following algorithm:

- First take s samples $\mathbf{V}_1, \ldots, \mathbf{V}_s$, and also pick a uniformly random non-repeating sequence of q indices $r_1, \ldots, r_q \in [n]$.
- Run $\mathcal{A}'$ such that, for every $i \in [q]$, when $\mathcal{A}'$ is about to query ℓ_i, query r_i from all samples instead. That is, we assume r_i to be the value of $\sigma(\ell_i)$.

The sample complexity and query complexity of algorithm $\mathcal{A}''$ are s and qs, respectively, which follows from Observation 8.2 and Observation 8.3. The correctness of the algorithm follows from Observation 8.2 and Observation 8.3 along with the fact that $\mathcal{P}$ is index-invariant. This completes the proof of Theorem 8.1. □

8.2 Preliminaries towards proving a quadratic separation result

In this subsection, we present some preliminary results required to prove that Theorem 1.7 is almost tight, that is, there exists an index-invariant property for which there is a nearly quadratic gap between the query complexities of adaptive and non-adaptive testers. The result is formally stated as follows.

Theorem 8.4 (Restatement of Theorem 1.8). *There exists an index-invariant property $\mathcal{P}_{\text{Gap}}$ that can be ε-tested adaptively using $\widetilde{\mathcal{O}}(n)$ queries for any $\varepsilon \in (0,1)$, while there exists an $\varepsilon \in (0,1)$ for which $\widetilde{\Omega}(n^2)$ queries are necessary for any non-adaptive ε-tester.*

In what follows throughout this section, we assume that the integer n is of the form $n = 2^l$ for some integer l, and that $k = \mathcal{O}(l)$ is another integer. We denote vectors in $\{0,1\}^N$ by capital bold letters (for example $\mathbf{X} \in \{0,1\}^N$) and vectors in $\{0,1\}^n$ by small bold letters (for example $\mathbf{x} \in \{0,1\}^n$). For two vectors $\mathbf{X}, \mathbf{Y} \in \{0,1\}^N$, we will use $\delta_H(\mathbf{X}, \mathbf{Y}) = N \cdot d_H(\mathbf{X}, \mathbf{Y})$ to denote the absolute Hamming distance between $\mathbf{X}$ and $\mathbf{Y}$.

To construct the property $\mathcal{P}_{\text{Gap}}$ (as stated in Theorem 8.4), we define two encodings $\text{SE} : \{0,1\}^\ell \to \{0,1\}^k$ and $\text{GE} : [n]^m \to [n]^n$ [14]. The encodings GE and SE follow from the construction of a Probabilistically Checkable Unveiling of a Shared Secret (PCUSS) in [BFLR20]. We can also construct such a function GE using the Reed-Solomon code, where we will assume that n is a prime power and use polynomials of degree $m-1$ over the field $\text{GL}(n)$ for $m = \Theta(n)$ [15].

Function SE: We will use a function SE of the form $\text{SE} : \{0,1\}^l \times \{0,1\} \to \{0,1\}^k$, where l and k are the integers defined above. In fact, SE takes an integer $i \in [n]$ in its Boolean encoding as an l bit Boolean string and a "secret" bit $a \in \{0,1\}$, and will output a Boolean string of length k. SE will have the following properties for some constant $\zeta \in (0, 1/2)$.

(i) Let $i, i' \in [n]$ be two integers encoded as binary strings of length l [16], and $a, a' \in \{0,1\}$. If $(i,a) \neq (i',a')$, then $\delta_H(\text{SE}(i,a), \text{SE}(i',a')) \geq \zeta \cdot k$.

(ii) Let $a \in \{0,1\}$ be a fixed bit, and suppose that i is an integer chosen uniformly at random from $[n]$. Then for any set of indices $I \subset [k]$ such that $|I| \leq \zeta \cdot k$, the restriction $\text{SE}(i,a)\,|_I$ is uniformly distributed over $\{0,1\}^{|I|}$.

Function GE: For our construction, we will use another function GE of the form $\text{GE} : [n]^m \to [n]^n$, where $n, m \in \mathbb{N}$ with the following properties for the same constant $\zeta \in (0, 1/2)$ as above.

(i) Let $\mathbf{z}, \mathbf{z}' \in [n]^m$ be two strings such that $\mathbf{z} \neq \mathbf{z}'$. For any two such strings $\mathbf{z}$ and $\mathbf{z}'$, $\delta_H(\text{GE}(\mathbf{z}), \text{GE}(\mathbf{z}')) = |\{i : \text{GE}(\mathbf{z})_i \neq \text{GE}(\mathbf{z}')_i\}| \geq \zeta \cdot n$.

(ii) Consider a string $\mathbf{z} \in [n]^m$ chosen uniformly at random. For any set of indices $I \subset [n]$ such that $|I| \leq \zeta \cdot n$, $\text{GE}(\mathbf{z})\,|_I$ is uniformly distributed over $[n]^{|I|}$.

From now on, we will use the following notation in this subsection: Let $n \in \mathbb{N}$ be such that $n = 2^l$ for some integer l, $k = \mathcal{O}(l)$ and $\zeta \in (0, 1/2)$ as above, $b = \lfloor \log(\lceil \log kn \rceil) \rfloor + 1$, $N = 1 + b + kn$ and $\alpha = 1/\log n$. Note that in particular $N = \mathcal{O}(n \log n)$. For a vector $\mathbf{X} \in \{0,1\}^N$ and a permutation

[14] SE stands for Secret Encoding, and GE stands for General Encoding.

[15] $\text{GL}(n)$ stands for the finite field with n elements.

[16] Binary strings of length $\log n$ can actually encode only integers from $\{0, \ldots, n-1\}$, so we use the encoding of 0 for the value n.

$\pi : [N] \to [N]$, $\mathbf{X}_\pi$ denotes the vector obtained from $\mathbf{X}$ by permuting the indices of $\mathbf{X}$ with π, that is, $\mathbf{X}_\pi = (\mathbf{X}_{\pi(1)}, \ldots, \mathbf{X}_{\pi(N)})$.

Let B be the sequence of integers $B = \{2, \ldots, b+1\}$, and for every $j \in [n]$, let C_j denote the sequence of integers $C_j = \{b+2+k(j-1), \ldots, b+1+kj\}$. For a sequence of integers A and a vector $\mathbf{X}$, we denote by $\mathbf{X}\mid_A$ the vector obtained by projecting $\mathbf{X}$ onto the set of indices of A preserving the sequence order. For a sequence $A \subseteq [N]$ and a permutation $\pi : [N] \to [N]$, we denote by $\pi(A)$ the sequence obtained after permuting every element of A with respect to the permutation π, that is, if $A = (a_1, \ldots, a_l)$, then $\pi(A) = (\pi(a_1), \ldots, \pi(a_l))$. In particular, we have $\mathbf{X}_\pi \mid_A = \mathbf{X} \mid_{\pi(A)}$. By abuse of notation and for simplicity, for a set of integers A and a vector $\mathbf{X}$, we denote by $\mathbf{X}\mid_A$ the vector obtained by projecting $\mathbf{X}$ onto the set of indices of A, whenever the ordering in which we consider the indices in A will be clear from the context [17].

In the following, we use string notation. For example, $\mathbf{1}^k\mathbf{0}^k$ denotes the vector in $\{0,1\}^{2k}$ whose first k coordinates are 1 and whose last k coordinates are 0. Now we formally define the notion of encoding of a vector which will be crucially used to define $\mathcal{P}_{\text{Gap}}$.

Definition 8.5 (Encoding of a vector). Let $n, k, b \in \mathbb{N}$, $N = 1 + b + kn$, and $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \{0,1\}^n$ and $\mathbf{Y} \in \{0,1\}^N$ be two vectors. $\mathbf{Y}$ is said to be an *encoding* of $\mathbf{x}$ with respect to the functions $\text{SE} : \{0,1\}^l \times \{0,1\} \to \{0,1\}^k$ and $\text{GE} : [n]^m \to [n]^n$ if the following hold:

(i) The first index of $\mathbf{Y}$ is 0.

(ii) $\mathbf{Y}\mid_B$ is the all-1 vector.

(iii) $\mathbf{Y}\mid_{[N]\setminus\{1\}\cup B}$ is of the form $\text{SE}(\text{GE}(\mathbf{z})_1, \mathbf{x}_1) \ldots \text{SE}(\text{GE}(\mathbf{z})_n, \mathbf{x}_n)$ for some string $\mathbf{z} \in [n]^m$. In other words, $\mathbf{Y} \mid_{C_j} = \text{SE}(\text{GE}(\mathbf{z})_j, \mathbf{x}_j)$ for every $j \in [n]$.

For ease of presentation, we will denote this encoding by FE, that is, $\text{FE} : [n]^m \times \{0,1\}^n \to \{0,1\}^N$ is the function [18] such that $\text{FE}(\mathbf{z}, \mathbf{x}) = \mathbf{0}(\mathbf{1}^b)\text{SE}(\text{GE}(\mathbf{z})_1, \mathbf{x}_1) \ldots \text{SE}(\text{GE}(\mathbf{z})_n, \mathbf{x}_n)$, for $\mathbf{z} \in [n]^m$ and $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \{0,1\}^n$. We also say that $\mathbf{X} \in \{0,1\}^N$ is a *valid encoding* of some $\mathbf{x} \in \{0,1\}^n$, if there exists some $\mathbf{z} \in [n]^m$ for which $\mathbf{X} = \text{FE}(\mathbf{z}, \mathbf{x})$. The image of FE will be called the set of all valid encodings.

Now let us infer two properties of the function FE, which will be crucial to our proofs, as stated in the following two claims. These properties of FE are analogous to the properties of SE and GE. As FE is formed by combining SE and GE, the proofs of these observations use their respective properties.

The following observation, particularly Items (i) and (ii), will allow us to prove that certain distributions are indeed far from the property $\mathcal{P}_{\text{Gap}}$ (to be defined later) in the EMD metric. Item (iii) will be useful to prove the soundness of our adaptive algorithm in Subsection 8.4, and in particular in Lemma 8.28.

Observation 8.6 (Distance properties of FE). Let $\text{FE} : [n]^m \times \{0,1\}^n \to \{0,1\}^N$ be the function from Definition 8.5. Then FE has the following properties:

(i) Let $\mathbf{x}, \mathbf{x}' \in \{0,1\}^n$ be any two strings and $\mathbf{z}, \mathbf{z}' \in [n]^m$ be two vectors such that $\mathbf{z} \neq \mathbf{z}'$. Then $\delta_H(\text{FE}(\mathbf{z}, \mathbf{x}), \text{FE}(\mathbf{z}', \mathbf{x}')) \geq \zeta^2 \cdot N/2$ holds.

(ii) Let $\mathbf{z}, \mathbf{z}' \in [n]^m$ be any two strings, and $\mathbf{x}, \mathbf{x}' \in \{0,1\}^n$ be two other strings such that $\mathbf{x} \neq \mathbf{x}'$. Then $\delta_H(\text{FE}(\mathbf{z}, \mathbf{x}), \text{FE}(\mathbf{z}', \mathbf{x}')) \geq \zeta k \cdot \delta_H(\mathbf{x}, \mathbf{x}')$.

[17] A common scenario is when the indexes of A are considered as a monotone increasing sequence.
[18] FE stands for Final Encoding.

(iii) Let $\mathbf{x}, \mathbf{x}' \in \{0,1\}^n$ be two strings and $\mathbf{z} \in [n]^m$ be a vector. Then $\delta_H(\mathrm{FE}(\mathbf{z},\mathbf{x}), \mathrm{FE}(\mathbf{z},\mathbf{x}')) \leq k \cdot \delta_H(\mathbf{x},\mathbf{x}')$, and in particular $d_H(\mathrm{FE}(\mathbf{z},\mathbf{x}), \mathrm{FE}(\mathbf{z},\mathbf{x}')) \leq d_H(\mathbf{x},\mathbf{x}')$ holds.

Proof. We prove each item separately below.

(i) Following the properties of GE (Property (i)), for $\mathbf{z} \neq \mathbf{z}'$, we know that $\delta_H(\mathrm{GE}(\mathbf{z}), \mathrm{GE}(\mathbf{z}')) \geq \zeta \cdot n \geq \zeta N/2k$. That is, the number of indices $j \in [n]$ such that $\mathrm{GE}(\mathbf{z})_j \neq \mathrm{GE}(\mathbf{z}')_j$, is at least $\zeta N/2k$. For every index $j \in [n]$ such that $\mathrm{GE}(\mathbf{z})_j \neq \mathrm{GE}(\mathbf{z}')_j$, we can say that $\delta_H(\mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}_j), \mathrm{SE}(\mathrm{GE}(\mathbf{z}')_j, \mathbf{x}'_j)) \geq \zeta \cdot k$. This is due to Property (i) of SE. Hence,

$$\begin{aligned}\delta_H(\mathrm{FE}(\mathbf{z},\mathbf{x}), \mathrm{FE}\big(\mathbf{z}',\mathbf{x}'\big)) &\geq \sum_{j \in [n]: \mathbf{z}_j \neq \mathbf{z}'_j} \delta_H(\mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}_j), \mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}'_j)) \\ &\geq \zeta N/2k \cdot \zeta k = \zeta^2 \cdot N/2.\end{aligned}$$

(ii) Consider two strings $\mathbf{x}, \mathbf{x}' \in \{0,1\}^n$ such that $\mathbf{x} \neq \mathbf{x}'$. Using Property (i) of SE, we know that $\delta_H(\mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}_j), \mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}'_j)) \geq \zeta \cdot k$ for every j for which $\mathbf{x}_j \neq \mathbf{x}'_j$. Note that the number of such indices j is $\delta_H(\mathbf{x},\mathbf{x}')$. Summing over them, we have the result.

(iii) Consider any two strings $\mathbf{x}, \mathbf{x}' \in \{0,1\}^n$. Observe that

$$\delta_H(\mathrm{FE}(\mathbf{z},\mathbf{x}), \mathrm{FE}\big(\mathbf{z},\mathbf{x}'\big)) = \sum_{j \in [n]} \delta_H(\mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}_j), \mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}'_j)).$$

Note that $\delta_H(\mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}_j), \mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}'_j))$ is at most k for every $j \in [n]$. Moreover, $\delta_H(\mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}_j), \mathrm{SE}(\mathrm{GE}(\mathbf{z})_j, \mathbf{x}'_j)) = 0$ for every $j \in [n]$ with $\mathbf{x}_j = \mathbf{x}'_j$. Since the number of indices j such that $\mathbf{x}_j \neq \mathbf{x}'_j$ is $\delta_H(\mathbf{x},\mathbf{x}')$, we conclude the following:

$$\delta_H(\mathrm{FE}(\mathbf{z},\mathbf{x}), \mathrm{FE}\big(\mathbf{z},\mathbf{x}'\big)) \leq k \cdot \delta_H(\mathbf{x},\mathbf{x}').$$

Note that this immediately implies $d_H(\mathrm{FE}(\mathbf{z},\mathbf{x}), \mathrm{FE}(\mathbf{z},\mathbf{x}')) \leq d_H(\mathbf{x},\mathbf{x}')$.

□

The following lemma will provide us a way to construct distributions that cannot be easily distinguished using non-adaptive queries (following a uniformly random index-permutation which we will deploy).

Lemma 8.7 (Projection property of FE). *Consider a fixed vector $\mathbf{x} \in \{0,1\}^n$, and let $\mathbf{z} \in [n]^m$ be a string chosen uniformly at random. For any set of indices $Q \subseteq [N]$ such that $|Q| \leq \zeta \cdot N/2k$ and $|Q \cap C_j| \leq \zeta \cdot k$ for every $j \in [n]$, the restriction of $\mathrm{FE}(\mathbf{z},\mathbf{x})\,|_{Q \setminus [b+1]}$ is uniformly distributed over $\{0,1\}^{|Q \setminus [b+1]|}$* [19].

Proof. For the set of indices Q, consider the set $J = \{j : Q \cap C_j \neq \emptyset\}$. From the statement of the lemma, we know that $|Q \cap C_j| \leq \zeta \cdot k$ for every $j \in J$. Noting that $|J| \leq |Q| \leq \zeta \cdot n$, if we consider the restriction $\mathrm{GE}(\mathbf{z})\,|_J$, following Property (ii) of the function GE, we know that $\mathrm{GE}(\mathbf{z})\,|_J$ is uniformly distributed over $[n]^{|J|}$.

Now when we call $\mathrm{SE}(i_j, \mathbf{x}_j)$ with $i_j \in [n]$ obtained from $\mathrm{GE}(\mathbf{z})\,|_J$, following the above argument, we can say that i_j has been chosen uniformly at random from $[n]$ (and independently from

[19] Recall that the restriction $\mathrm{FE}(\mathbf{z},\mathbf{x})\,|_{[b+1]}$ is always the vector $\mathbf{01}^b$.

the other $i_{j'}$). Since $|Q \cap C_j| \leq \zeta \cdot k$, applying Property (ii) of the function SE, we know that the corresponding restriction of $\mathrm{SE}(i_j, \mathbf{x}_j)$ will be uniformly distributed over $\{0,1\}^{|Q \cap C_j|}$. Since $\mathrm{FE}(\mathbf{z}, \mathbf{x}) = \mathbf{0}(\mathbf{1}^b)\mathrm{SE}(\mathrm{GE}(\mathbf{z})_1, \mathbf{x}_1) \ldots \mathrm{SE}(\mathrm{GE}(\mathbf{z})_n, \mathbf{x}_n)$, combining the above arguments, we conclude that $\mathrm{FE}(\mathbf{z}, \mathbf{x}) \mid_{Q \setminus [b+1]}$ is uniformly distributed over $\{0,1\}^{|Q \setminus [b+1]|}$. □

Now we are ready to formally define the property, first constructing a non-index-invariant version to be used in the next index-invariant definition.

Property $\mathcal{P}^0_{\mathrm{Gap}}$: A distribution D over $\{0,1\}^N$ is in $\mathcal{P}^0_{\mathrm{Gap}}$ if and only if D satisfies the following conditions:

(i) $D(\mathbf{U}) = \alpha$, where $\mathbf{U} = \mathbf{1}\mathbf{0}^{N-1}$ is the indicator vector for the index 1.

(ii) Consider the set of vectors $\mathcal{S} = \{\mathbf{V}_1, \ldots, \mathbf{V}_b\}$ in $\{0,1\}^N$ such that for every $i \in [b]$, the i-th vector $\mathbf{V}_i$ is of the form $\mathbf{1}^{i+1}\mathbf{0}^{N-1-i}$. Note that $\mathbf{V}_i \mid_B = 1^i 0^{b-i}$ for $B = \{2, \ldots, b+1\}$. We require that $D(\mathbf{V}_i) = \alpha/b$ for every $i \in [b]$.

(iii) Consider the set of vectors $\mathcal{T} = \{\mathbf{W}_0, \ldots, \mathbf{W}_{\lceil \log kn \rceil - 1}\}$ (disjoint from $\mathcal{S}$) in $\{0,1\}^N$ such that for every $\mathbf{W}_i \in \mathcal{T}$, $\mathbf{W}_i$ is of the form $\mathbf{0}(b(i))(\mathbf{0}^{2^i}\mathbf{1}^{2^i})^{kn/2^{i+1}}$, where $b(i)$ denotes the length b binary representation of i [20]. Note that for $i = b + 2 + j$, with $0 \leq j \leq kn - 1$, the sequence $(\mathbf{W}_0)_i, \ldots, (\mathbf{W}_{(\lceil \log kn \rceil - 1)})_i$ holds the binary representation of j. Also, note that there is an one-to-one correspondence between $\mathbf{W}_i \mid_B$ and $\mathbf{W}_i \mid_{[N] \setminus \{B\} \cup \{1\}}$. We require that $D(\mathbf{W}_i) = \alpha/|\mathcal{T}|$ for every $\mathbf{W}_i \in \mathcal{T}$.

(iv) $\mathrm{Supp}(D) \setminus (\{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T})$ consists of valid encodings of at most n vectors from $\{0,1\}^n$ with respect to the functions $\mathrm{SE} : \{0,1\}^l \times \{0,1\} \to \{0,1\}^k$ and $\mathrm{GE} : [n]^m \to [n]^n$, for the integers $l, m, k \in \mathbb{N}$ as defined in Definition 8.5. That is, there exist vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \{0,1\}^n$ for which $\mathrm{Supp}(D) \setminus (\{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T}) \subseteq \{\mathrm{FE}(\mathbf{z}, \mathbf{x}_i) : \mathbf{z} \in [n]^m, i \in [n]\}$. Note that for D to be a distribution, we must have $D(\mathrm{Supp}(D) \setminus (\{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T})) = 1 - 3\alpha$.

Property $\mathcal{P}_{\mathrm{Gap}}$: A distribution D over $\{0,1\}^N$ is said to be in the property $\mathcal{P}_{\mathrm{Gap}}$ if D_π is in $\mathcal{P}^0_{\mathrm{Gap}}$ for some permutation $\pi : [N] \to [N]$.

Remark 4 (Intuition behind the definition of $\mathcal{P}_{\mathrm{Gap}}$). If a distribution D is in $\mathcal{P}^0_{\mathrm{Gap}}$, then we can easily check (by querying the indexes in B) whether a sample from D would be equal to $\mathrm{FE}(\mathbf{z}, \mathbf{x})$ for some $\mathbf{z} \in [n]^m$ and $\mathbf{x} \in \{0,1\}^n$. In that case, individual bits of $\mathbf{x}$ can be decoded by querying the appropriate C_j and then passed to a tester of distributions over $\{0,1\}^n$.

On the other hand, if we take a uniformly random permutation of such a distribution D, which keeps it in $\mathcal{P}_{\mathrm{Gap}}$ (though no longer in $\mathcal{P}^0_{\mathrm{Gap}}$), a non-adaptive algorithm will need many queries to capture sufficiently many bits from any C_j, and this will enable us to fully hide the identity of $\mathbf{x}$ if fewer queries are performed.

By contrast, an adaptive tester will use relatively few samples that are queried in their entirety to obtain the (permutations of the) special vectors in Items (i) to (iii) of the definition of $\mathcal{P}^0_{\mathrm{Gap}}$, from which it will be able to fully learn the index-permutation applied to the distribution, and continue to successfully decode individual bits. A few further samples queried in their entirety will ensure that there is very little total weight on vectors that are neither special vectors nor equal to $\mathrm{FE}(\mathbf{z}, \mathbf{x})$ for some $\mathbf{z} \in [n]^m$ and $\mathbf{x} \in \{0,1\}^n$.

[20] If $kn/2^{i+1}$ is not an integer, we trim the rightmost copy of $\mathbf{0}^{2^i}\mathbf{1}^{2^i}$ so that the total length of "$(\mathbf{0}^{2^i}\mathbf{1}^{2^i})^{kn/2^{i+1}}$" is exactly kn.

Known useful results about support estimation: Now we state a lemma which will be required later to describe the adaptive tester for $\mathcal{P}_{\text{Gap}}$. Informally, it says that whether a distribution D over $\{0,1\}^n$ has support size s or is ε-far from any such distribution, can be tested by taking $\widetilde{\mathcal{O}}(s)$ samples from D, and performing $\widetilde{\mathcal{O}}(s)$ queries on them.

Lemma 8.8 (Support size estimation, Theorem 1.9 & Corollary 2.3 of [GR22]). *There exists an algorithm* Supp-Est(s,ε) *that uses $\widetilde{\mathcal{O}}(s/\varepsilon^2)$ queries to an unknown distribution D defined over $\{0,1\}^n$, and with probability at least $\frac{9}{10}$ distinguishes whether D has at most s elements in its support or D is ε-far from all such distributions with support size at most s.*

We will also use a lower bound on the support size estimation problem to prove the lower bound on non-adaptive testers for testing $\mathcal{P}_{\text{Gap}}$. Informally speaking, given a distribution D over $\{1,\dots,2n\}$, in order to distinguish in the traditional (non-huge-object) model whether the size of the support of D is n, or D is far from all such distributions, $\Omega(\frac{n}{\log n})$ samples are necessary. More formally, we have the following theorem.

Theorem 8.9 (Support size estimation lower bound, Corollary 9 of [VV10]). *There exist two distributions D^{Supp}_{yes} and D^{Supp}_{no} over distributions over $\{1,\dots,2n\}$, and an $\eta \in (0,1/8)$ such that the following holds:*

(i) The probability mass of every element in the support of D^{Supp}_{yes} as well as D^{Supp}_{no} is a multiple of $1/2n$.

(ii) D^{Supp}_{yes} is supported over distributions whose support size is n.

(iii) D^{Supp}_{no} is supported over distributions whose support size is at least $(1+2\eta)n$, and in particular are η-far in variation distance from any distribution defined over $\{1,\dots,2n\}$ whose support size is $(1+2\eta)n$.

(iv) If a sequence of $o(\frac{n}{\log n})$ samples from a distribution are drawn according to either D^{Supp}_{yes} or D^{Supp}_{no}, the resulting distributions over the sample sequences are $1/4$-close to each other.

We present an adaptive algorithm to test $\mathcal{P}_{\text{Gap}}$ in Subsection 8.4 and we prove the lower bound for non-adaptive testers in Subsection 8.5. In Subsection 8.3, we describe a subroutine to determine the unknown permutation that will be used in our adaptive algorithm in Subsection 8.4.

8.3 Determining the permutation π

Here we design an algorithm that, given a distribution $D \in \mathcal{P}_{\text{Gap}}$, can learn with high probability the permutation π for which $D_\pi \in P^0_{\text{Gap}}$.

The crux of the algorithm is that if $D \in \mathcal{P}_{\text{Gap}}$, then there exist $\mathbf{U}' = \mathbf{U}_\pi \in \{0,1\}^n$, $\mathcal{S}' = \mathcal{S}_\pi = \{\mathbf{V}'_i = (\mathbf{V}_i)_\pi : i \in [b]\}$ and $\mathcal{T}' = \mathcal{T}_\pi = \{\mathbf{W}'_j = (\mathbf{W}_j)_\pi : j \in \{0\} \cup [\lceil \log kn \rceil] - 1\}$ in the support of D such that $D(\mathbf{U}') = \alpha$, $D(\mathbf{V}'_i) = \alpha/b$ for every $i \in [b]$ and $D(\mathbf{W}'_j) = \alpha/\lceil \log kn \rceil$. Note that $\mathbf{U}$, $\mathcal{S}$ and $\mathcal{T}$ are as defined in the property $\mathcal{P}^0_{\text{Gap}}$.

The main observation is that, if we are given the set of special vectors $\{\mathbf{U}'\} \cup \mathcal{S}' \cup \mathcal{T}'$, then we can determine the permutation π. Our algorithm can find $\mathbf{U}', \mathcal{S}'$ and $\mathcal{T}'$ with high probability, if they exist, by taking $\mathcal{O}(\log^2 N/\alpha) = \mathcal{O}(\log^2 n/\alpha)$ samples and reading them in their entirety. This is due to the fact that the probability mass of every vector in the set of special vectors is at least $\Omega(\alpha/\log n)$.

The algorithm is described in the following subroutine Find-Permutation (see Algorithm 1) [21].

[21] This algorithm is not adaptive in itself, but its output is used adaptively in the testing algorithm described later.

Let us start by analyzing the query complexity of FIND-PERMUTATION.

Lemma 8.10 (**Query complexity of** FIND-PERMUTATION**).** *The query complexity of the above defined* FIND-PERMUTATION *is* $\widetilde{\mathcal{O}}(N)$.

Proof. Note that FIND-PERMUTATION takes a multi-set $\mathcal{X}$ of $\mathcal{O}(\log^2 N/\alpha)$ samples from D in Step (i), and queries them completely. So, FIND-PERMUTATION performs $\mathcal{O}(N \log^2 N/\alpha)$ queries in Step (i). FIND-PERMUTATION does not perform any new queries in Step (ii), Step (iii), Step (iv) and Step (v). Finally, FIND-PERMUTATION takes another multi-set $\mathcal{X}'$ of $\mathcal{O}(\log^2 N/\alpha)$ samples from D and queries them completely, similar to Step (i). Recalling that $\alpha = 1/\log n$, the query complexity of FIND-PERMUTATION is $\widetilde{\mathcal{O}}(N) = \widetilde{\mathcal{O}}(n)$ in total. □

Now we proceed to prove the correctness of FIND-PERMUTATION.

Lemma 8.11 (**Guarantee when** $D \in \mathcal{P}_{\text{Gap}}$**).** *If D is a distribution defined over $\{0,1\}^N$ such that $D \in \mathcal{P}_{\text{Gap}}$, then with probability at least $9/10$,* FIND-PERMUTATION *reports the permutation π such that $D_\pi \in \mathcal{P}^0_{\text{Gap}}$.*

We prove the above lemma by a series of intermediate lemmas. In the following lemmas, we consider $\mathbf{U}$, $\mathcal{S}$ and $\mathcal{T}$ as per the definition of $\mathcal{P}^0_{\text{Gap}}$. Also, consider the permutation π such that $D_\pi \in \mathcal{P}_{\text{Gap}}$.

Lemma 8.12 (**Correctly finding** $\pi^{-1}(1)$**).** *With probability at least $1 - 1/N^3$, $\mathcal{X}$ will contain the vector $\mathbf{U}'$ for which $\mathbf{U}'_\pi = \mathbf{U}$, and $i^* = \pi^{-1}(1)$ will be identified correctly. Moreover,* FIND-PERMUTATION *proceeds to Step (iii).*

Proof. By the definition of $\mathcal{P}^0_{\text{Gap}}$, the vector $\mathbf{U}'$ is the only vector in the support of D containing a single 1. Since $D(\mathbf{U}') = D(\mathbf{U}_{\pi^{-1}(1)}) = \alpha$, and we are taking $|\mathcal{X}|$ many samples from D, the probability that $\mathbf{U}'$ will not appear in $\mathcal{X}$ is at most $(1-\alpha)^{|X|} \leq \frac{1}{N^3}$. Thus, with probability at least $1 - \frac{1}{N^3}$, $\mathbf{U}' \in \mathcal{X}$ and FIND-PERMUTATION in Step (ii) proceeds to the next step. □

Lemma 8.13 (**Correctly finding** $B' = \pi^{-1}(B)$**).** *With probability at least $1 - 1/N^3$, the algorithm* FIND-PERMUTATION *will correctly identify $\mathbf{V}'_1, \ldots, \mathbf{V}'_b$ for which $\mathbf{V}'_{i,\pi} = \mathbf{V}_i$, and $B' = \pi^{-1}(2), \ldots, \pi^{-1}(b+1)$ will be identified correctly as well. Moreover,* FIND-PERMUTATION *proceeds to Step (iv).*

Proof. Let $\mathbf{V}'_1, \ldots, \mathbf{V}'_b$ denote the vectors for which $\mathbf{V}'_{i,\pi} = \mathbf{V}_i$ for every i. Note that these are the only vectors outside $\mathbf{U}'$ in the support of D that have 1 at the index i^*. As $D(\mathbf{V}'_i) = \frac{\alpha}{b}$, the probability that $\mathbf{V}'_i$ does not appear in $\mathcal{X}$ is at most $(1 - \frac{\alpha}{b})^{|X|}$. Since $|\mathcal{X}| = \mathcal{O}(\log^2 N/\alpha)$ and $b = \mathcal{O}(\log \log kn)$, the probability that $\mathbf{V}'_i \in \mathcal{X}$ is at least $1 - \frac{1}{N^4}$. Using the union bound over all the vectors of $\mathcal{S}'$, with probability at least $1 - 1/N^3$, we know that all of these vectors are in $\mathcal{X}$, in which case they are identified correctly, so B' is identified correctly as well, and FIND-PERMUTATION in Step (iii) proceeds to the next step. □

Lemma 8.14 (**Correctly identifying** $\pi^{-1}(b+2), \ldots, \pi^{-1}(N)$**).** *Let $\mathbf{W}'_1, \ldots, \mathbf{W}'_{\lceil \log kn \rceil - 1}$ denote the vectors for which $\mathbf{W}'_{j,\pi} = \mathbf{W}_j$ for every j. With probability at least $1 - 1/N^3$, all these vectors appear in $\mathcal{X}$, in which case they are identified correctly, and so are $\pi^{-1}(b+2), \ldots, \pi^{-1}(N)$. Moreover,* FIND-PERMUTATION *proceeds to Step (vi).*

The proof of the above lemma is similar to the proof of Lemma 8.13 and is omitted. Note that from Lemma 8.12, Lemma 8.13 and Lemma 8.14, we know that with probability at least $1 - o(1)$, the algorithm FIND-PERMUTATION has correctly determined the permutation π and proceeded to Step (vi). We will finish up the proof of Lemma 8.11 using the following lemma.

Lemma 8.15. *The probability that* Find-Permutation *outputs* Fail *in Step (vi) (instead of outputting π) is at most $1/N^3$.*

Proof. As $D \in \mathcal{P}_{\text{Gap}}$, from the description of the property, we know that $D(\{\mathbf{U}'\} \cup \mathcal{S}' \cup \mathcal{T}') = 3\alpha$. As $|\mathcal{X}'| = \mathcal{O}(\log^2 N/\alpha)$, using the Chernoff bound (Lemma A.1), we have the result. □

Combining the above lemmas, we conclude that with probability at least 9/10, the algorithm Find-Permutation outputs a correct permutation π, completing the proof of Lemma 8.11.

To conclude this section, we show that with high probability, we will not output π for which too much weight is placed outside the "encoded part" of the distribution.

Lemma 8.16. *For any distribution D (regardless of whether D is in $\mathcal{P}_{\text{Gap}}$ or not), the probability that* Find-Permutation *outputs a permutation π for which $D(\{\mathbf{X} : \mathbf{X} \mid_{\{i^*\} \cup B'} = \mathbf{01}^b\}) \leq 1 - 5\alpha$ is at most $1/10$.*

Proof. Recall the set of vectors $\mathcal{Y}$ as defined in Step (vi) of Find-Permutation: $\mathcal{Y} = \{\mathbf{Z} \in \mathcal{X}' : \mathbf{Z} \mid_{i^* \cup B'} \neq \mathbf{01}^b\}$, where $\mathcal{X}'$ is the multi-set of (new) samples obtained in Step (vi) of Find-Permutation. Consider a distribution D such that $D(\{\mathbf{X} : \mathbf{X}_{\{i^*\} \cup B'} = \mathbf{01}^b\}) \leq 1 - 5\alpha$. This implies that $\mathbb{E}\left[|\mathcal{Y}| \,/\, |\mathcal{X}'|\right] \geq 5\alpha$. As $|\mathcal{X}'| = \mathcal{O}(\log^2 N/\alpha)$, using the Chernoff bound (Lemma A.1), we obtain that with probability at least 9/10, the algorithm Find-Permutation outputs Fail in Step (vi), and does not output any permutation π. This completes the proof. □

8.4 The upper bound on adaptive testing for property $\mathcal{P}_{\text{Gap}}$

In this subsection, we design the adaptive tester for the property $\mathcal{P}_{\text{Gap}}$. Given a distribution D over $\{0,1\}^N$, with high probability, Alg-Adaptive outputs Accept when $D \in \mathcal{P}_{\text{Gap}}$, and outputs Reject when D is far from $\mathcal{P}_{\text{Gap}}$. The formal adaptive algorithm is presented in Alg-Adaptive (see Algorithm 2). Note that it has only two adaptive steps.

In the first adaptive step, our tester Alg-Adaptive starts by calling the algorithm Find-Permutation (as described in Subsection 8.3) whose query complexity is $\widetilde{\mathcal{O}}(n)$. If $D \in \mathcal{P}_{\text{Gap}}$, with high probability, Find-Permutation returns the permutation π such that $D_\pi \in \mathcal{P}^0_{\text{Gap}}$. Once π is known, when we obtain a sample $\mathbf{X}$ from D, we can consider it as $\mathbf{X}_\pi$ from D_π. Also, from the structure of the vectors in the support of the distributions in P^0_{Gap}, we can decide whether X_π is a special vector, that is, $X_\pi \in \{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T}$ or $\mathbf{X}_\pi$ is an encoding vector, that is, $\mathbf{X}_\pi = \text{FE}(\mathbf{z}, \mathbf{x})$ for some $\mathbf{z} \in [n]^m$ and $\mathbf{x} \in \{0,1\}^n$. Observe that, in the later case, we can decode any bit of $\mathbf{x}$ (say $\mathbf{x}_j$) by finding $\mathbf{X}_\pi$ projected into C_j, which can be done by performing $\mathcal{O}(\log n)$ queries.

As the second adaptive step, our algorithm asks for a sequence $\mathcal{Y}$ of $\mathcal{O}(n/\varepsilon)$ samples from D, that is, from D_π. Let $\mathcal{Y}' \subseteq \mathcal{Y}$ be the sequence of encoding vectors in $\mathcal{Y}$. We now call Supp-Est$(\mathcal{Y}', \varepsilon/3)$ (from Lemma 8.8), and depending on its output, Alg-Adaptive either reports Accept or Reject. Note that we can execute every query by Supp-Est$(\mathcal{Y}', \varepsilon/3)$, by performing $\mathcal{O}(\log n)$ queries to the corresponding sample in $\mathcal{Y}'$ as discussed above.

When $D \in \mathcal{P}_{\text{Gap}}$ (that is, $D_\pi \in \mathcal{P}^0_{\text{Gap}}$ for the permutation π), the set of encoding vectors in D_π is the encoding of at most n vectors in $\{0,1\}^n$. So, in that case, Alg-Adaptive reports Accept with high probability. Now consider the case where D is ε-far from $\mathcal{P}_{\text{Gap}}$. If Alg-Adaptive has not rejected D before calling Supp-Est$(\mathcal{Y}', \varepsilon/3)$, we will show that the distribution over $\{0,1\}^n$ induced by the vectors decoded from the encoding vectors in D_π is $\varepsilon/3$-far from having support size n. Then, Alg-Adaptive will still reject D with high probability.

Let us first discuss the query complexity of Alg-Adaptive.

Lemma 8.17 (**Query complexity of** Alg-Adaptive)**.** *The query complexity of the adaptive tester* Alg-Adaptive *for testing the property* $\mathcal{P}_{\mathrm{Gap}}$ *is* $\widetilde{\mathcal{O}}(N) = \widetilde{\mathcal{O}}(n)$.

Proof. Note that Alg-Adaptive calls the algorithm Find-Permutation in Step (i). Following the query complexity lemma of Find-Permutation (Lemma 8.10), we know that Find-Permutation performs $\widetilde{\mathcal{O}}(N)$ queries.

For every sample taken in Step (ii), the sampled vectors of the multi-set $\mathcal{X}$ are queried completely. Since we take $\mathcal{O}(1/\varepsilon)$ samples, this step requires $\mathcal{O}(N/\varepsilon) = \widetilde{\mathcal{O}}(n/\varepsilon)$ queries in total.

Then in Step (iii), Alg-Adaptive takes a multi-set $\mathcal{Y}$ of $\mathcal{O}(n/\varepsilon)$ samples, and queries for the indices in $\{i^*\} \cup B'$ to obtain the vectors in $\mathcal{Y}'$, which takes $\mathcal{O}(n \log\log kn/\varepsilon)$ queries. Finally, in Step (iv), Alg-Adaptive calls the algorithm Supp-Est, which performs $\widetilde{\mathcal{O}}(n)$ queries (following Lemma 8.8), each of them simulated by $\mathcal{O}(\log n)$ queries to some $\mathbf{Y}_i \mid_{C'_j}$. Thus, in total, Alg-Adaptive performs $\widetilde{\mathcal{O}}(N) = \widetilde{\mathcal{O}}(n)$ queries. □

Now we prove the correctness of Alg-Adaptive. We will start with the completeness proof.

Lemma 8.18 (**Completeness of** Alg-Adaptive)**.** *Let* D *be a distribution defined over* $\{0,1\}^N$. *If* $D \in \mathcal{P}_{\mathrm{Gap}}$, *then the algorithm* Alg-Adaptive *will output* Accept *with probability at least* $2/3$.

Proof. Consider a distribution $D \in \mathcal{P}_{\mathrm{Gap}}$. From the completeness lemma of Find-Permutation (Lemma 8.11), we infer that with probability at least $9/10$, Find-Permutation returns the correct permutation π in Step (i). Then, by the definition of $\mathcal{P}_{\mathrm{Gap}}$, the algorithm Alg-Adaptive can never encounter any samples with invalid encodings in Step (ii) which could cause it to Reject. Thus, with probability at least $9/10$, the algorithm proceeds, with the correct permutation π, to Step (iii) and Step (iv).

As $D \in \mathcal{P}_{\mathrm{Gap}}$, $D(\{\mathbf{U}'\} \cup \mathcal{S}' \cup \mathcal{T}') = 3\alpha$. Since $|\mathcal{Y}| = \mathcal{O}(n/\varepsilon)$, using the Chernoff bound (Lemma A.2 (ii)), we can say that, with probability at least $9/10$, $|\mathcal{Y}'| \,/\, |\mathcal{Y}| \geq 1/2$. Moreover, as the vectors in $\mathcal{Y}'$ are valid encodings with respect to the function FE of at most n vectors from $[2n]$, following the support estimation upper bound lemma (Lemma 8.8), we obtain that Supp-Est outputs Accept with probability at least $9/10$. Combining these, we conclude that Alg-Adaptive outputs Accept with probability at least $2/3$. □

Now we prove that when D is ε-far from $\mathcal{P}_{\mathrm{Gap}}$, Alg-Adaptive will output Reject with probability at least $2/3$.

Lemma 8.19 (**Soundness of** Alg-Adaptive)**.** *Let* $\varepsilon \in (0,1)$ *be a proximity parameter. Assume that* D *is a distribution defined over* $\{0,1\}^N$ *such that* D *is* ε*-far from* $\mathcal{P}_{\mathrm{Gap}}$. *Then* Alg-Adaptive *outputs* Reject *with probability at least* $2/3$.

From the description of Alg-Adaptive (Algorithm 2), if the tester reports Reject before executing all the steps of Supp-Est$(\mathcal{Y}', \varepsilon/3)$ in Step (iv), then we are done. So, let us assume that Alg-Adaptive executes all the steps of Supp-Est$(\mathcal{Y}', \varepsilon/3)$. Let $\mathcal{Y}'$ be the set of samples from a distribution $D^{\#}$ over $\{0,1\}^n$ as it is presented to Supp-Est$(\mathcal{Y}', \varepsilon/3)$. Note that $D^{\#}$ is unknown and we are accessing $D^{\#}$ indirectly via decoding samples from D over $\{0,1\}^N$. From the correctness Supp-Est$(\mathcal{Y}', \varepsilon/3)$, we will be done with the proof of Lemma 8.19 by proving the following lemma.

Lemma 8.20 (**Property of the decoded distribution**)**.** $D^{\#}$ *is* $\varepsilon/3$*-far from having support size at most* n.

We prove the above lemma using a series of claims. Let D be a distribution which is ε-far from $\mathcal{P}_{\mathrm{Gap}}$, and $\mathcal{V}$ denote the set $\{\mathbf{X} \in \mathrm{Supp}(D) : \mathbf{X} \mid_{\{i^*\} \cup B'} = \mathbf{01}^b\}$, and let us define $\mathcal{U} = \mathrm{Supp}(D) \setminus \mathcal{V}$. Let us start with the following observation.

Observation 8.21. $D(\mathcal{U}) \leq 5\alpha$, unless the algorithm Alg-Adaptive has rejected with probability at least $1 - 1/N^3$ in Step (i).

Proof. Since Alg-Adaptive in Step (i) invokes the algorithm Find-Permutation, this follows immediately from Lemma 8.16. □

Let π be the permutation returned by Find-Permutation. Now assume $\mathcal{V}^{\text{inv}} \subseteq \mathcal{V}$ denotes the following set of vectors:

$$\mathcal{V}^{\text{inv}} = \{\mathbf{X} \in \mathcal{V} : \mathbf{X}_\pi \neq \text{FE}(\mathbf{z}, \mathbf{x}) \text{ for all } \mathbf{z} \in [n]^m, \mathbf{x} \in \{0,1\}^n\}$$

For every vector $\mathbf{V} \in \mathcal{V}^{\text{inv}}$, let $\Gamma'_{\mathbf{V}} = \{j \in [n] : \mathbf{V}\,|_{C'_j}$ is not in the image of SE$\}$ denotes the set of indices in $[n]$ of chunks of all the "*locally invalid*" encodings in the vector $\mathbf{V}$ [22]. Now we have the following observation.

Observation 8.22. $D(\mathcal{V}^{\text{inv}}) \leq \varepsilon/10$.

The above observation holds as otherwise, Alg-Adaptive would have rejected in Step (ii) with probability at least $2/3$.

Let us define a distribution D_1 over $\{0,1\}^N$ using the following procedure:

(i) Set $D_1(\mathbf{X}) = D(\mathbf{X})$ for every $\mathbf{X} \in \mathcal{U}$.

(ii) Recall that $\Gamma'_{\mathbf{V}} = \{j \in [n] : \mathbf{V}\,|_{C'_j}$ is not in the image of SE$\}$ for every vector $\mathbf{V} \in \mathcal{V}^{\text{inv}}$. For every vector $\mathbf{V} \in \mathcal{V}^{\text{inv}}$, we perform the following steps:

  (a) For every $j \notin \Gamma'_{\mathbf{V}}$, decode the vector $\mathbf{V}\,|_{C'_j}$ using SE to obtain $\mathbf{x}_j \in \{0,1\}$.

  (b) For every $j \in \Gamma'_{\mathbf{V}}$, choose an arbitrary value $\mathbf{x}_j$ from $\{0,1\}$.

  (c) Using $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ obtained from (a) and (b), construct a new vector $\mathbf{V}'$ for which $\mathbf{V}'_\pi = \text{FE}(\mathbf{z}, \mathbf{x})$ for an arbitrary $\mathbf{z} \in [n]^m$, where π is the permutation obtained from Find-Permutation in Step (i) of Alg-Adaptive.

(iii) For every vector $\mathbf{V} \in \mathcal{V} \setminus \mathcal{V}^{\text{inv}}$, set $\mathbf{V}' = \mathbf{V}$.

(iv) Finally define $D_1(\mathbf{W}) = \sum_{\mathbf{V}:\mathbf{V}'=\mathbf{W}} D(\mathbf{V})$ for every $\mathbf{W} \in \mathcal{V}$.

Let $\mathcal{V}'$ be the set of vectors in $\{0,1\}^N$ that are in the support of D_1 but not in $\mathcal{U}$, that is, $\mathcal{V}' = \{\mathbf{X} : \mathbf{X} \in \text{Supp}(D_1) \setminus \mathcal{U}\}$. From the construction of D_1, the following observation follows.

Observation 8.23. $D_1(\mathcal{U}) = D(\mathcal{U}) \leq 5\alpha$ and $D_1(\mathcal{V}') = D(\mathcal{V}) = 1 - D(\mathcal{U}) \geq 1 - 5\alpha$.

Now we prove that the distributions D and D_1 are not far in Earth Mover Distance.

Lemma 8.24. *The Earth Mover Distance between D and D_1 is at most $\varepsilon/10$.*

[22] Note that it may be the case that $\Gamma'_{\mathbf{V}} = \emptyset$, for example when for every $j \in [n]$, we have $\mathbf{V}\,|_{C'_j} = \text{SE}(i_j, \mathbf{x}_j)$, for some $i_1, \ldots, i_n$ and $\mathbf{x}_1, \ldots, \mathbf{x}_n$ for which $i_1, \ldots, i_n$ are not in the image of GE.

Proof. Recall that the EMD between D and D_1 is the solution to the following LP:

$$\text{Minimize} \quad \sum_{\mathbf{X},\mathbf{Y}\in\{0,1\}^N} f_{\mathbf{XY}} d_H(\mathbf{X},\mathbf{Y})$$

$$\text{Subject to} \quad \sum_{\mathbf{Y}\in\{0,1\}^N} f_{\mathbf{XY}} = D(\mathbf{X}) \;\forall\; \mathbf{X}\in\{0,1\}^N \text{ and } \sum_{\mathbf{X}\in\{0,1\}^N} f_{\mathbf{XY}} = D_1(\mathbf{Y}) \;\forall\; \mathbf{Y}\in\{0,1\}^N.$$

Consider the flow f^* such that $f^*_{\mathbf{XX}} = D(\mathbf{X})$ for every $\mathbf{X}\in\mathcal{U}\cup(\mathcal{V}\setminus\mathcal{V}^{\text{inv}})$, $f^*_{\mathbf{VV'}} = D_1(\mathbf{V})$ for every $\mathbf{V}\in\mathcal{V}^{\text{inv}}$, and $f^*_{\mathbf{XY}} = 0$ for all other vectors. Then we have the following:

$$\begin{aligned} d_{EM}(D,D_1) &\le \sum_{\mathbf{X},\mathbf{Y}\in\{0,1\}^N} f^*_{\mathbf{XY}} d_H(\mathbf{X},\mathbf{Y}) \\ &\le \sum_{\mathbf{X}\in\{0,1\}^N\setminus\mathcal{V}^{\text{inv}}} f^*_{\mathbf{XX}} d_H(\mathbf{X},\mathbf{X}) + \sum_{\mathbf{V}\in\mathcal{V}^{\text{inv}}} f^*_{\mathbf{VV'}} d_H(\mathbf{V},\mathbf{V'}) \\ &\le 0 + \sum_{\mathbf{V}\in\mathcal{V}^{\text{inv}}} D(\mathbf{V}) d_H(\mathbf{V},\mathbf{V'}). \end{aligned}$$

To bound the second term of the last expression, note that

$$\sum_{\mathbf{V}\in\mathcal{V}^{\text{inv}}} D(\mathbf{V}) d_H(\mathbf{V},\mathbf{V'}) \le D(\mathcal{V}^{\text{inv}}) \le \varepsilon/10.$$

This follows from Observation 8.22. Thus, we conclude that $d_{EM}(D,D_1)\le\varepsilon/10$, completing the proof of the lemma. □

Now we have the following observation regarding the rejection probabilities of Alg-Adaptive for the distributions D and D_1. This will imply that, as we are executing all steps of Supp-Est$(\mathcal{Y}',\varepsilon/3)$, the steps of our algorithm are oblivious to both D and D_1. That is, we can assume that the input to the algorithm Alg-Adaptive is the distribution D_1 instead of D.

Observation 8.25. The probability that the tester Alg-Adaptive outputs Reject in Step (iv) where the input distribution is D is at least as large as the probability that Alg-Adaptive outputs Reject in Step (iv) when the input distribution is D_1.

Proof. Note that in the distribution D, there can be some vectors in Supp(D) that are not valid encodings with respect to the function FE. Thus during its execution, the tester Alg-Adaptive can Reject D by Condition (ii) and Condition (iv) (b). However, by the construction of D_1 from D, we have replaced the invalid encoding vectors with valid encoding vectors. Thus, the only difference it makes here is that Alg-Adaptive may eventually accept a sample from D_1 when encountering such a place where a sample from D would have been immediately rejected by Condition (ii) or Condition (iv) (b). Other than this difference, the distributions D and D_1 are identical. So, the probability that Alg-Adaptive will Reject D is at least as large as the probability that it Rejects D_1. □

Now, let us come back to the proof of Lemma 8.20. Recall that $\mathcal{V}' = \{\mathbf{X} : \mathbf{X}\in\text{Supp}(D_1)\setminus\mathcal{U}\}$. Let us define the distribution $D^{\#}$ over $\{0,1\}^n$ referred to in Lemma 8.20. For $\mathbf{x}\in\{0,1\}^n$, we have the following:

$$D^{\#}(\mathbf{x}) = D_1^{dec}(\mathbf{x}) = \frac{1}{D_1(\mathcal{V}')} \sum_{\substack{\mathbf{Y}_\pi=\text{FE}(\mathbf{z},\mathbf{x}) \\ \text{for some } \mathbf{z}\in[n]^m}} D_1(\mathbf{Y}) = \frac{1}{D_1(\mathcal{V}')} \sum_{\mathbf{z}\in[n]^m} D_1(\text{FE}(\mathbf{z},\mathbf{x})_{\pi^{-1}}). \qquad (1)$$

For the sake of contradiction, assume that $D^{\#} = D_1^{dec}$ is $\varepsilon/3$-close to having support size at most n. Let D_2 be a distribution over $\{0,1\}^n$ having support size at most n such that the Earth Mover Distance between D_2 and D_1^{dec} is at most $\varepsilon/3$.

Given the distribution D_2 over $\{0,1\}^n$, and the flow $f'_{\mathbf{xy}}$ from D_2 to D_1^{dec} realizing the EMD of at most $\varepsilon/3$ between them, let us consider the distribution D_2^{enc} over $\{0,1\}^N$ as follows:

(i) For any $\mathbf{X} \in \mathcal{V}'$, for which $\mathbf{X}_\pi = \mathrm{FE}(\mathbf{z}, \mathbf{x})$ for some $\mathbf{z} \in [n]^m$, set:

$$D_2^{enc}(\mathbf{X}) = \sum_{\mathbf{y} \in \{0,1\}^n} f'_{\mathbf{xy}} \frac{D_1(\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}})}{D_1^{dec}(\mathbf{y})}.$$

(ii) For every $\mathbf{X} \in \mathcal{U}$, set $D_2^{enc}(\mathbf{X}) = D_1(\mathbf{X})$.

The following observation follows from Observation 8.23 and the construction of D_2^{enc}.

Observation 8.26. $D_2^{enc}(\mathcal{U}) = D_1(\mathcal{U}) \leq 5\alpha$ and $D_2^{enc}(\mathcal{V}') = D_1(\mathcal{V}') = 1 - D_1(\mathcal{U}) \geq 1 - 5\alpha$.

The following two lemmas bound the distance of D_2^{enc} from $\mathcal{P}_{\mathrm{Gap}}$ and from D_1, where D_1^{dec} is $\varepsilon/3$-close to having support size at most n. We will prove these two lemmas later.

Lemma 8.27. *D_2^{enc} is 6α-close to $\mathcal{P}_{\mathrm{Gap}}$.*

Lemma 8.28. *The Earth Mover Distance between D_2^{enc} and D_1 is at most $\varepsilon/3$.*

Assuming Lemma 8.27 and Lemma 8.28 hold, now we proceed to prove Lemma 8.20.

Proof of Lemma 8.20. From Lemma 8.24, we know that $d_{EM}(D, D_1) \leq \varepsilon/10$. So, the above two lemmas imply that D is $(\varepsilon/3 + \varepsilon/10 + 6\alpha) = 2\varepsilon/3$-close to $\mathcal{P}_{\mathrm{Gap}}$, which contradicts the fact that D is ε-far from $\mathcal{P}_{\mathrm{Gap}}$. This completes the proof of the lemma. □

Now we will prove Lemma 8.27 and Lemma 8.28.

Proof of Lemma 8.27. We define another distribution D_3 over $\{0,1\}^N$ from D_2^{enc} such that D_3 is in $\mathcal{P}_{\mathrm{Gap}}$ and $d_{EM}(D_2^{enc}, D_3) \leq 6\alpha$ as follows:

(i) $D_3(\mathbf{U}') = \alpha$.

(ii) $D_3(\mathbf{X}) = \frac{\alpha}{b}$ for every $\mathbf{X} \in \mathcal{S}'$, $D_3(\mathbf{X}) = \frac{\alpha}{\lceil \log kn \rceil}$ for every $\mathbf{X} \in \mathcal{T}'$.

(iii) $D_3(\mathbf{X}) = (1 - 3\alpha) \cdot \frac{D_2^{enc}(\mathbf{X})}{D_2^{enc}(\mathcal{V}')}$ for every $\mathbf{X} \in \mathcal{V}'$.

Recall that D_2 is a distribution over $\{0,1\}^n$ that has support size at most n. This implies that the set of vectors in $\textsc{Supp}(D_2^{enc}) \setminus \mathcal{U}$ is the encoding of at most n vectors in $\{0,1\}^n$. So, from the definition of $\mathcal{P}_{\mathrm{Gap}}$ and D_3, it is clear that $D_3 \in \mathcal{P}_{\mathrm{Gap}}$.

Now we show that the Earth Mover Distance between the distributions D_3 and D_2^{enc} is not large.

Claim 8.29. *The Earth Mover Distance between D_2^{enc} and D_3 is at most 6α.*

Proof. We will bound the Earth Mover Distance between D_2^{enc} and D_3 in terms of the variation distance between them as follows:

$$\begin{aligned} d_{EM}(D_2^{enc}, D_3) &\leq \frac{1}{2} \cdot \sum_{\mathbf{X} \in \{0,1\}^N} |D_2^{enc}(\mathbf{X}) - D_3(\mathbf{X})| \\ &= \frac{1}{2} \cdot \sum_{\mathbf{X} \in \mathcal{V}'} |D_2^{enc}(\mathbf{X}) - D_3(\mathbf{X})| + \frac{1}{2} \cdot \sum_{\mathbf{X} \in \{0,1\}^N \setminus \mathcal{V}'} |D_2^{enc}(\mathbf{X}) - D_3(\mathbf{X})|. \quad (2) \end{aligned}$$

Let us bound the first term as follows:

$$\begin{aligned} \sum_{\mathbf{X} \in \mathcal{V}'} |D_2^{enc}(\mathbf{X}) - D_3(\mathbf{X})| &= \sum_{\mathbf{X} \in \mathcal{V}'} |(1-3\alpha)\frac{D_2^{enc}(\mathbf{X})}{D_2^{enc}(\mathcal{V}')} - D_2^{enc}(\mathbf{X})| \\ &= \sum_{\mathbf{X} \in \mathcal{V}'} \frac{D_2^{enc}(\mathbf{X})}{D_2^{enc}(\mathcal{V}')} |(1-3\alpha) - D_2^{enc}(\mathcal{V}')| \\ &= \sum_{\mathbf{X} \in \mathcal{V}'} \frac{D_2^{enc}(\mathbf{X})}{D_2^{enc}(\mathcal{V}')} |3\alpha - (1 - D_2^{enc}(\mathcal{V}'))| \\ &\leq \sum_{\mathbf{X} \in \mathcal{V}'} 3\alpha \frac{D_2^{enc}(\mathbf{X})}{D_2^{enc}(\mathcal{V}')} \leq 3\alpha. \quad \left(\because D_2^{enc}(\mathcal{V}') \geq 1 - 5\alpha, \text{ Observation 8.26}\right) \end{aligned}$$

From Observation 8.23, $D_2^{enc}(\mathcal{U}) \leq 5\alpha$. From the definition of D_3, $D_3(\mathcal{U}) = 3\alpha$, we have

$$\sum_{\mathbf{X} \in \{0,1\}^N \setminus \mathcal{V}'} |D_2^{enc}(\mathbf{X}) - D_3(\mathbf{X})| \leq 8\alpha.$$

Following Equation 2, we conclude that $d_{EM}(D_2^{enc}, D_3) \leq 6\alpha$, which completes the proof. □

Since $D_3 \in \mathcal{P}_{\text{Gap}}$, and $d_{EM}(D_2^{enc}, D_3) \leq 6\alpha$, we conclude that D_2^{enc} is 6α-close to $\mathcal{P}_{\text{Gap}}$. □

Proof of Lemma 8.28. Recall that the EMD between D_2^{enc} and D_1 is the solution to the following LP:

$$\begin{aligned} &\text{Minimize} \quad \sum_{\mathbf{X}, \mathbf{Y} \in \{0,1\}^N} f_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \\ &\text{Subject to} \quad \sum_{\mathbf{Y} \in \{0,1\}^N} f_{\mathbf{XY}} = D_2^{enc}(\mathbf{X}) \ \forall \mathbf{X} \in \{0,1\}^N \text{ and } \sum_{\mathbf{X} \in \{0,1\}^N} f_{\mathbf{XY}} = D_1(\mathbf{Y}) \ \forall \mathbf{Y} \in \{0,1\}^N. \end{aligned}$$

Let $f'_{\mathbf{xy}}$ be the flow realizing the EMD between D_2 and D_1^{dec}. Using f', we now construct a new flow $f^\star$ between D_2^{enc} and D_1 as follows:

(i) For vectors $\mathbf{X}, \mathbf{Y} \in \mathcal{U}$,

  (a) If $\mathbf{X} \neq \mathbf{Y}$, then set $f^\star_{\mathbf{XY}} = 0$.

  (b) If $\mathbf{X} = \mathbf{Y}$, then set $f^\star_{\mathbf{XY}} = D_2^{enc}(\mathbf{X}) = D_1(\mathbf{Y})$.

(ii) For two vectors $\mathbf{X}, \mathbf{Y} \in \mathcal{V}$, we take the vectors $\mathbf{x}, \mathbf{y} \in \{0,1\}^n$ such that $\mathbf{X}, \mathbf{Y} \in \{0,1\}^N$ are their valid encodings (by construction, if $\mathbf{X}$ and $\mathbf{Y}$ are in the support of D_2 and D_1^{enc} respectively, such vectors $\mathbf{x}, \mathbf{y}$ exist), and vectors $\mathbf{z}_1, \mathbf{z}_2$ such that $\mathbf{X}_\pi = \text{FE}(\mathbf{z}_1, \mathbf{x})$ and $\mathbf{Y}_\pi = \text{FE}(\mathbf{z}_2, \mathbf{y})$. Now we set the flow as follows:

(a) If $\mathbf{z}_1 \neq \mathbf{z}_2$, then set $f^\star_{\mathbf{XY}} = 0$.

(b) If $\mathbf{z}_1 = \mathbf{z}_2$, then set $f^\star_{\mathbf{XY}} = f'_{\mathbf{xy}} \cdot \frac{D_1(\mathbf{Y})}{D_1^{dec}(\mathbf{y})}$.

(iii) If one of $\mathbf{X}$ and $\mathbf{Y}$ is in $\mathcal{U}$ and the other one is in $\mathcal{V}$, then $f^\star_{\mathbf{XY}} = 0$.

We first argue that the flow $f^*_{\mathbf{XY}}$ constructed as above is indeed a valid flow, that is, we have: $\sum_{\mathbf{Y}\in\{0,1\}^N} f^\star_{\mathbf{XY}} = D_2^{enc}(\mathbf{X})$ and $\sum_{\mathbf{X}\in\{0,1\}^N} f^\star_{\mathbf{XY}} = D_1(\mathbf{Y})$.

To prove $\sum_{\mathbf{Y}\in\{0,1\}^N} f_{\mathbf{XY}} = D_2^{enc}(\mathbf{X})$, first observe that it holds when $\mathbf{X} \in \mathcal{U}$ from (i) and (iii) in the description of $f^\star_{\mathbf{XY}}$. Now consider the case where $\mathbf{X} \in \mathcal{V}$. Assume $\mathbf{X}_\pi = \mathrm{FE}(\mathbf{z}, \mathbf{x})$, where $\mathbf{z} \in [n]^m$ and $\mathbf{x} \in \{0,1\}^n$. So, from (ii) in the description of $f^\star_{\mathbf{XY}}$, we have

$$\sum_{\mathbf{Y}\in\{0,1\}^N} f^\star_{\mathbf{XY}} = \sum_{\mathbf{y}\in\{0,1\}^n} f^\star_{\mathbf{X}\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}}} = \sum_{\mathbf{y}\in\{0,1\}^n} f'_{\mathbf{xy}} \frac{D_1(\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}})}{D_1^{dec}(\mathbf{y})} = D_2^{enc}(\mathbf{X}).$$

For $\sum_{\mathbf{X}\in\{0,1\}^N} f^\star_{\mathbf{XY}} = D_1(\mathbf{Y})$, consider $\mathbf{Y} \in \mathcal{V}$ for which $\mathbf{Y}_\pi = \mathrm{FE}(\mathbf{z}, \mathbf{y})$ for some $\mathbf{z} \in [n]^m$. Then we have the following:

$$\sum_{\mathbf{X}\in\{0,1\}^N} f^\star_{\mathbf{XY}} = \sum_{\mathbf{x}\in\{0,1\}^n} f'_{\mathbf{xy}} \frac{D_1(\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}})}{D_1^{dec}(\mathbf{y})} = D_1^{dec}(\mathbf{y}) \frac{D_1(\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}})}{D_1^{dec}(\mathbf{y})} = D_1(\mathbf{Y}).$$

In the above, we have used the fact that $f'_{\mathbf{xy}}$ is a valid flow from D_2 to D_1^{dec}.

Now, to bound EMD between D_2^{enc} and D_1, let us bound the sum $\sum_{\mathbf{X},\mathbf{Y}\in\{0,1\}^N} f^\star_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y})$.

$$\begin{aligned}
&\sum_{\mathbf{X},\mathbf{Y}\in\{0,1\}^N} f^\star_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \\
&= \sum_{\mathbf{X},\mathbf{Y}\in\mathcal{V}} f^\star_{\mathbf{XY}} d_H(\mathbf{X}, \mathbf{Y}) \quad \text{(From (i) and (iii) in the description of } f^\star) \\
&= \sum_{\mathbf{x},\mathbf{y}\in\{0,1\}^n} \sum_{\mathbf{z}\in[n]^m} f^\star_{\mathrm{FE}(\mathbf{z},\mathbf{x})_{\pi^{-1}}\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}}} \cdot d_H(\mathrm{FE}(\mathbf{z},\mathbf{x})_{\pi^{-1}}, \mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}}) \\
&\qquad\qquad \text{(From (ii) in the description of } f^\star) \\
&\leq \sum_{\mathbf{x},\mathbf{y}\in\{0,1\}^n} \sum_{\mathbf{z}\in[n]^m} f^\star_{\mathrm{FE}(\mathbf{z},\mathbf{x})_{\pi^{-1}}\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}}} \cdot d_H(\mathbf{x}, \mathbf{y}) \quad \text{(Observation 8.6 (iii))} \\
&= \sum_{\mathbf{x},\mathbf{y}\in\{0,1\}^n} \sum_{\mathbf{z}\in[n]^m} f'_{\mathbf{xy}} \frac{D_1(\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}})}{D_1^{dec}(\mathbf{y})} \cdot d_H(\mathbf{x}, \mathbf{y}) \quad \text{(From (ii) in the description of } f^\star) \\
&= \sum_{\mathbf{x},\mathbf{y}\in\{0,1\}^n} \left(f'_{\mathbf{xy}} d_H(\mathbf{x}, \mathbf{y}) \cdot \sum_{\mathbf{z}\in[n]^m} \frac{D_1(\mathrm{FE}(\mathbf{z},\mathbf{y})_{\pi^{-1}})}{D_1^{dec}(\mathbf{y})} \right) \\
&= D_1(\mathcal{V}') \sum_{\mathbf{x},\mathbf{y}\in\{0,1\}^n} f'_{\mathbf{xy}} d_H(\mathbf{x}, \mathbf{y}) \quad \text{(By Equation (1))} \\
&\leq \sum_{\mathbf{x},\mathbf{y}\in\{0,1\}^n} f'_{\mathbf{xy}} d_H(\mathbf{x}, \mathbf{y}) \leq \frac{\varepsilon}{3}.
\end{aligned}$$

The last inequality follows from the fact that f' realizes the assumed EMD between D_1 and D_2^{dec}. □

8.5 Near-quadratic lower bound for non-adaptive testers for testing $\mathcal{P}_{\text{Gap}}$

Lemma 8.30 (Lower bound on non-adaptive testers). *Given sample and query access to an unknown distribution D, in order to distinguish whether D satisfies $\mathcal{P}_{\text{Gap}}$ or is ε-far from satisfying it, any non-adaptive tester must perform $\widetilde{\Omega}(n^2)$ queries to the samples obtained from D, for some $\varepsilon \in (0,1)$.*

To prove the above lemma, we will construct two hard distributions over distributions, D_{yes} which is supported over $\mathcal{P}_{\text{Gap}}$, and D_{no} which is supported over distributions far from $\mathcal{P}_{\text{Gap}}$, where to distinguish them, any non-adaptive tester must perform $\widetilde{\Omega}(n^2)$ queries. Recall from Theorem 8.9 that D_{yes}^{Supp} and D_{no}^{Supp} are two distributions defined over distributions over $\{1,\ldots,2n\}$, where D_{yes}^{Supp} provides distributions whose support sizes are n, and D_{no}^{Supp} provides distributions that are η-far from distributions whose support size is $(1+2\eta)n$, for some constant $\eta \in (0,1/8)$. We will use these two distributions to construct the hard distributions D_{yes} and D_{no} for the property $\mathcal{P}_{\text{Gap}}$.

The hard distributions D_{yes} and D_{no}: We describe the distributions D_{yes} and D_{no} over distributions over $\{0,1\}^N$ such that D_{yes} is supported over $\mathcal{P}_{\text{Gap}}$ and D_{no} is supported over distributions that are $\zeta^2\cdot\eta/5$-far from $\mathcal{P}_{\text{Gap}}$. In what follows, we describe a distribution D ($D = D_{yes}$ or $D = D_{no}$) with D^{Supp} as parameter, where D^{Supp} is a distribution defined over distributions over $[2n]$. In particular, D^{Supp} is either D_{yes}^{Supp} or D_{no}^{Supp}, where $D = D_{yes}$ when $D^{\text{Supp}} = D_{yes}^{\text{Supp}}$, or $D = D_{no}$ when $D^{\text{Supp}} = D_{no}^{\text{Supp}}$. To generate D, we first construct a distribution over distributions D^0 as follows. We denote by $\widehat{D}$ the distribution over $\{0,1\}^N$ that we draw according to D^0.

(i) Set $\widehat{D}(U) = \alpha$, where $\mathbf{U} = \mathbf{10}^{N-1}$ is the indicator vector for the index 1.

(ii) Take a set of vectors $\mathcal{S} = \{\mathbf{V}_1,\ldots,\mathbf{V}_b\}$ in $\{0,1\}^N$ such that for every $i \in [b]$, the i-th vector $\mathbf{V}_i$ is of the form $\mathbf{1}^{i+1}\mathbf{0}^{N-1-i}$. Set $\widehat{D}(\mathbf{V}_i) = \alpha/b$ for every $i \in [b]$.

(iii) Take another set of vectors $\mathcal{T} = \{\mathbf{W}_0,\ldots,\mathbf{W}_{\lceil \log kn\rceil - 1}\}$ (disjoint from $\mathcal{S}$) in $\{0,1\}^N$ such that for every $\mathbf{W}_i \in \mathcal{T}$, $\mathbf{W}_i$ is of the form $\mathbf{0}(b(i))(\mathbf{0}^{2^i}\mathbf{1}^{2^i})^{kn/2^{i+1}}$, where $b(i)$ denotes the length b binary representation of i [23]. Set $\widehat{D}(\mathbf{W}_i) = \alpha/|\mathcal{T}|$ for every $\mathbf{W}_i \in \mathcal{T}$.

(iv) Take a set of vectors $\mathcal{Y} \subseteq \{0,1\}^n$ such that $|\mathcal{Y}| = 2n$, and for any two vectors $\mathbf{y}_i, \mathbf{y}_j \in \mathcal{Y}$, $i \neq j$, $\delta_H(\mathbf{y}_i,\mathbf{y}_j) \geq n/3$. Also, draw a distribution $\widetilde{D}$ over $[2n]$ according to D^{Supp}.

(v) Define $\widehat{D}(\text{FE}(\mathbf{z},\mathbf{y}_i)) = (1-3\alpha)\widetilde{D}(i)/n^m$ for every $i \in [2n]$ and $\mathbf{z} \in [n]^m$, where $\text{FE} : [n]^m \times \{0,1\}^n \to \{0,1\}^N$ is the encoding function from Definition 8.5.

(vi) For all other remaining vectors that are not assigned probability mass in the above description, set their probabilities to 0.

We define D as the process of drawing a distribution $\widehat{D}$ according to D^0, and permuting it using a uniformly random permutation $\pi : [N] \to [N]$.

[23] If $kn/2^{i+1}$ is not an integer, we trim the rightmost copy of $\mathbf{0}^{2^i}\mathbf{1}^{2^i}$ so that the total length of "$(\mathbf{0}^{2^i}\mathbf{1}^{2^i})^{kn/2^{i+1}}$" is exactly kn.

Remark 5 (Intuition behind the above hard distributions). Unlike our adaptive algorithm to test $\mathcal{P}_{\text{Gap}}$ (Algorithm 2 in Subsection 8.4), we can not determine the permutation π first, and then perform queries depending on the permutation π. When the permutation π is not known, even if we obtain a sample $\mathbf{X}$ and know that it is equal to $\text{FE}(\mathbf{z}, \mathbf{x})_{\pi^{-1}}$ for some $\mathbf{x} \in \{0,1\}^n$ and $\mathbf{z} \in [n]^m$, we can not even decode a single bit of $\mathbf{x}$, unless we query too many of the indices of $\mathbf{X}$. This follows from the properties of our encodings functions SE and GE, used to construct FE (see Lemma 8.7), which "hides" $\mathbf{x}$ inside $\mathbf{X}$. Intuitively, this says that we have to query a quasilinear number of the coordinates of the sample. Since the support estimation problem admits a sample complexity lower bound of $\Omega(n/\log n)$, the non-adaptive query complexity of $\widetilde{\Omega}(n^2)$ follows for non-adaptive algorithms. We will formalize this intuition below.

We will start with the following simple observation.

Observation 8.31. The distribution D_{yes} is supported over $\mathcal{P}_{\text{Gap}}$.

Proof. From the construction of D_{yes}, which is constructed by encoding the elements of the support of the distribution D_{yes} drawn from D_{yes}^{Supp}, it is clear that $D_{yes} \in \mathcal{P}_{\text{Gap}}$. □

Now we show that the distribution D_{no} is supported over distributions that are far from the property $\mathcal{P}_{\text{Gap}}$.

Lemma 8.32 (Farness lemma). *D_{no} is supported over distributions that are $\zeta^2 \cdot \eta/5$-far from $\mathcal{P}_{\text{Gap}}$.*

Before directly proceeding to the proof, let us first prove an additional lemma which will be used in the proof of Lemma 8.32.

Lemma 8.33. *For any two distinct vectors $\mathbf{X}_1$ and $\mathbf{X}_2$ where $\mathbf{X}_{1,\pi}$, $\mathbf{X}_{2,\pi} \in \text{Supp}(\widehat{D}) \setminus (\{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T})$ for $\widehat{D} \in \text{Supp}(D_{no})$, and π is the permutation for which $\widehat{D}_\pi \in D_{no}^0$, we have $\delta_H(\mathbf{X}_1, \mathbf{X}_2) \geq \zeta^2 \cdot N/2$.*

Proof. We will use the properties of the function FE as mentioned in Observation 8.6. Recall that for a string $\mathbf{z} \in [n]^m$, and a vector $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \{0,1\}^n$, we have $\text{FE}(\mathbf{z}, \mathbf{x}) = \mathbf{0}(\mathbf{1}^b)\text{SE}(\text{GE}(\mathbf{z})_1, \mathbf{x}_1) \ldots \text{SE}(\text{GE}(\mathbf{z})_n, \mathbf{x}_n)$. Now we have the following two cases:

(a) Suppose that for some vectors $\mathbf{x} \in \{0,1\}^n$, and $\mathbf{z}_1, \mathbf{z}_2 \in [n]^m$ such that $\mathbf{z}_1 \neq \mathbf{z}_2$, we have $\mathbf{X}_{1,\pi} = \text{FE}(\mathbf{z}_1, \mathbf{x})$ and $\mathbf{X}_{2,\pi} = \text{FE}(\mathbf{z}_2, \mathbf{x})$. Then following Property (i) of FE in Observation 8.6, we know that $\delta_H(\mathbf{X}_1, \mathbf{X}_2) \geq \zeta^2 \cdot N/2$ (noting that permuting the two vectors by the permutation π preserves their pairwise distance).

(b) Suppose that for some vectors $\mathbf{z} \in [n]^m$, and $\mathbf{x}_1, \mathbf{x}_2 \in \{0,1\}^n$ such that $\mathbf{x}_1 \neq \mathbf{x}_2$, we have $\mathbf{X}_{1,\pi} = \text{FE}(\mathbf{z}, \mathbf{x}_1)$ and $\mathbf{X}_{2,\pi} = \text{FE}(\mathbf{z}, \mathbf{x}_2)$. Then following Property (ii) of FE in Observation 8.6, we know that $\delta_H(\mathbf{X}_1, \mathbf{X}_2) \geq \zeta \cdot \delta_H(\mathbf{x}_1, \mathbf{x}_2)$. From the choice of the vectors $\mathbf{y}_1, \ldots, \mathbf{y}_{2n}$, we know that $\delta_H(\mathbf{x}_1, \mathbf{x}_2) \geq n/3$. Thus, we can say that in this case $\delta_H(\mathbf{X}_1, \mathbf{X}_2) \geq \zeta \cdot nk/3 > \zeta^2 \cdot N/2$ (recalling that $\zeta < 1/2$).

Combining the above, we conclude that $\delta_H(\mathbf{X}_1, \mathbf{X}_2) \geq \zeta^2 \cdot N/2$, for any two distinct vectors $\mathbf{X}_1, \mathbf{X}_2$ as above. □

Proof of Lemma 8.32. Suppose that $\widehat{D} \in \text{Supp}(D_{no})$, and π is the permutation for which $\widehat{D}_\pi \in \text{Supp}(D_{no}^0)$. We will bound $d_{EM}(\widehat{D}, \mathcal{P}_{\text{Gap}})$. Let us denote the distribution $D_Y \in \mathcal{P}_{\text{Gap}}$ that is closest

to $\widehat{D}$, where π_Y is the permutation for which $D_{Y,\pi_Y} \in \mathcal{P}^0_{\text{Gap}}$. Let us first define a new distribution $\widetilde{D}_Y$ over $\{0,1\}^N$ as follows:

$$\widetilde{D}_Y(\mathbf{X}) = \begin{cases} \frac{1}{(1-3\alpha)} D_Y(\mathbf{X}) & \mathbf{X}_{\pi_Y} \notin (\{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T}) \\ 0 & \text{otherwise} \end{cases}$$

Similarly, we also define another distribution $\widetilde{D}$ from $\widehat{D}$, using π instead of π_Y.

Now we have the following claim that bounds the distance between $\widetilde{D}_Y$ and $\widetilde{D}$.

Claim 8.34. $d_{EM}(\widetilde{D}, \widetilde{D}_Y) \geq \zeta^2 \cdot \eta/4$.

Proof. Following the definition of the property $\mathcal{P}_{\text{Gap}}$, we know that $\text{Supp}(\widetilde{D}_Y)$ consists of possible encodings of n distinct vectors from $\{0,1\}^n$, and there are at most n^m valid encodings of every such vector (as per the number of possible vectors $\mathbf{z} \in [n]^m$ that are given as input to GE). This implies that the size of the support of the distribution $\widetilde{D}_Y$ is at most n^{m+1}.

Since any distribution in the support of D^{Supp}_{no} has support size at least $(1+2\eta)n$, following a similar argument as above, we infer that the size of the support of $\widetilde{D}$ is at least $(1+2\eta)n^{m+1}$. Moreover, by Lemma 8.33, we know that any pair of vectors there has distance at least $\zeta^2/2$ (in relative distance). Also, as any vector in the support of any distribution in the support of D^{Supp}_{no} has probability mass that is multiple of $1/2n$, we infer that every vector in the support of $\widetilde{D}$ has probability mass at least $n^{-m-1}/2$ (as per Item (v) in the definition of D^0).

Summing up, we obtain that there are at least $2\eta \cdot n^{m+1}$ many vectors in $\text{Supp}(\widetilde{D})$ that are $\zeta^2/4$-far (in relative distance) from any vector in $\text{Supp}(\widetilde{D}_Y)$, all of whose weights are at least $n^{-m-1}/2$ [24]. Thus, the Earth Mover Distance of $\widetilde{D}$ from $\widetilde{D}_Y$ is at least $\zeta^2 \cdot \eta/4$. □

Recall that we need to bound the distance between $\widehat{D}$ and D_Y. From Claim 8.34, we know that $d_{EM}(\widetilde{D}, \widetilde{D}_Y) \geq \zeta^2 \cdot \eta/4$, where the distributions $\widetilde{D}$ and $\widetilde{D}_Y$ are defined over the encoding vectors. From the definition of $\widetilde{D}$ and $\widetilde{D}_Y$ from $\widehat{D}$ and D_Y, we conclude that $d_{EM}(D_Y, \widehat{D}) = (1-3\alpha) d_{EM}(\widetilde{D}, \widetilde{D}_Y) \geq \zeta^2 \cdot \eta/5$. □

Now we prove that the distributions D_{yes} and D_{no} remain indistinguishable to any non-adaptive tester, unless it performs $\widetilde{\Omega}(n^2)$ queries. We start with some definitions that will be required for the proof. Recall that $N = \mathcal{O}(n \log n)$.

Definition 8.35 (Large and small query set). A set of indices $I \subseteq [N]$ is said to be a *large* if $|I| > n/\log^{10} n$. Otherwise, I is said to be a *small*.

Now we show that for a uniformly random permutation σ, and any C_j as defined in the property $\mathcal{P}_{\text{Gap}}$, with high probability the size of the set of indices $|I \cap \sigma(C_j)|$ will be small, unless I is a large query set.

Observation 8.36. Let $\sigma : [N] \to [N]$ be a uniformly random permutation, and C_j correspond to a "bit encoding set" of size k (as per the definition of $\mathcal{P}_{\text{Gap}}$) for an arbitrary $j \in [n]$. For a fixed small query set $I \subseteq [N]$, the probability that $|I \cap \sigma(C_j)|$ is at least $\zeta \cdot k$ is at most $1/n^{10}$.

Proof. Let us define a collection of binary random variables $\langle X_i : i \in I \rangle$ such that the following holds:

$$X_i = \begin{cases} 1 & i \in \sigma(C_j) \\ 0 & \text{otherwise} \end{cases}$$

[24] By the triangle inequality, if we consider a Hamming ball of radius $\zeta^2/4$ around every vector in $\text{Supp}(\widetilde{D}_Y)$, there can be at most one vector from $\text{Supp}(\widetilde{D})$ inside the ball.

Then as σ is a uniformly random permutation, $\Pr(X_i = 1) = \frac{|\sigma(C_j)|}{N} = \mathcal{O}(\frac{1}{n})$ for any $i \in [n]$. Now let us define another random variable $X = \sum_{i=1}^{n} X_i$. Noting that $X = |I \cap \sigma(C_j)|$, we obtain $\mathbb{E}[X] = \mathcal{O}(1/\log^{10} n)$. By applying Hoeffding's bound for sampling without replacement (Lemma A.4), we can say that $\Pr(X \geq \zeta \cdot k) \leq 1/n^{10}$. This completes the proof. □

Now let us define an event $\mathcal{E}_{I,j}$ as follows:

$$\mathcal{E}_{I,j} := \text{The query set } I \text{ satisfies } |I \cap \sigma(C_j)| \leq \zeta \cdot k.$$

Now we are ready to prove that unless $\widetilde{\Omega}(n^2)$ queries are performed, no non-adaptive tester can distinguish D_{yes} from D_{no}.

Lemma 8.37 (Indistinguishability lemma). *With probability at least $2/3$, in order to distinguish D_{yes} from D_{no}, $\widetilde{\Omega}(n^2)$ queries are necessary for any non-adaptive tester.*

Proof. From our result on the adaptive ε-tester for $\mathcal{P}_{\text{Gap}}$, we know that $\widetilde{\mathcal{O}}(n)$ queries are sufficient for adaptively testing $\mathcal{P}_{\text{Gap}}$. Without loss of generality, let us assume that the non-adaptive tester takes at most n^2 samples from the unknown distribution D (since we can assume that at least one query is performed in every sample). As per the definition of a non-adaptive tester, assume that the samples taken are $\mathbf{X}_1, \ldots, \mathbf{X}_s$, and their respective query sets are $I_1, \ldots, I_s$ for some integer s.

Consider an event $\mathcal{E}$ as follows:

$$\mathcal{E} := \text{For every } \ell \in [s] \text{ for which } I_\ell \text{ is small and every } j \in [n], \text{ the event } \mathcal{E}_{I_\ell,j} \text{ occurs.}$$

Since the non-adaptive tester takes at most n^2 samples, there can be at most n^2 samples for which a small set was queried, that is, $s \leq n^2$. Moreover, there are n possible sets C_j present in a sample. Using the union bound, along with Observation 8.36, we can say that the event $\mathcal{E}$ holds with probability at least $1 - 1/n^7$. Given that the event $\mathcal{E}$ holds, we will now show that the induced distributions of D_{yes} and D_{no} on small query sets are identical and independent of the samples with large query sets.

Claim 8.38. *Assume that the event $\mathcal{E}$ holds. Then a non-adaptive tester that uses at most $o(n/\log n)$ large query sets, can not distinguish D_{yes} from D_{no} with probability more than $1/4$.*

Proof. Since the distributions produced by D_{yes} and D_{no} are identical over the respective permutations of $(\{\mathbf{U}\} \cup \mathcal{S} \cup \mathcal{T})$, it is sufficient to prove indistinguishability over the restrictions to the valid encodings of $\mathbf{y}_1, \ldots, \mathbf{y}_{2n}$ (as they appear in the definition of D^0). Furthermore, we argue that this claim holds even if for every large query set, the tester is provided with the entire vector that was sampled.

Given that the event $\mathcal{E}$ holds, regardless of whether the distribution was produced by D_{yes} or D_{no}, the restriction of the samples to the small queried sets are completely uniformly distributed, even when conditioned on the samples with large query sets (which are taken independently of them). Thus we may assume that all samples with small query sets are ignored by the tester, since the answers to these queries can be simulated without taking any samples at all.

Finally, we appeal to the construction of the hard distributions D_{yes} and D_{no} from D_{yes}^{Supp} and D_{no}^{Supp}. By Theorem 8.9, the distance between these two distributions over the sample sequence is at most $1/4$, unless there were more than $o(n/\log n)$ samples with large sets. This completes the proof of the claim. □

Combining Claim 8.38 with the above bound on the probability of the event $\mathcal{E}$, we conclude that $\widetilde{\Omega}(n^2)$ queries are necessary for any non-adaptive tester to distinguish D_{yes} from D_{no} with probability at least $2/3$, that is, with a probability difference of at least $1/3$. This concludes the proof of the lemma. □

References

[ABEF17] Noga Alon, Omri Ben-Eliezer, and Eldar Fischer. Testing hereditary properties of ordered graphs and matrices. In *FOCS*, 2017. 12

[ABR16] Maryam Aliakbarpour, Eric Blais, and Ronitt Rubinfeld. Learning and testing junta distributions. In *COLT*, 2016. 3

[ACK18] Jayadev Acharya, Clément L Canonne, and Gautam Kamath. A chasm between identity and equivalence testing with conditional queries. *Theory Of Computing*, 2018. 6

[ACL+21] Jayadev Acharya, Clément L Canonne, Yuhan Liu, Ziteng Sun, and Himanshu Tyagi. Interactive inference under information constraints. *IEEE Transactions on Information Theory*, 2021. 3

[ADK15] Jayadev Acharya, Constantinos Daskalakis, and Gautam Kamath. Optimal testing for properties of distributions. *NIPS*, 2015. 3

[AKNS99] Noga Alon, Michael Krivelevich, Ilan Newman, and Mario Szegedy. Regular languages are testable with a constant number of queries. In *FOCS*, 1999. 39

[ASZ21] Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Robust testing and estimation under manipulation attacks. In *ICML*. PMLR, 2021. 3

[BC17] Tugkan Batu and Clément L Canonne. Generalized uniformity testing. In *FOCS*, 2017. 3

[BC18] Rishiraj Bhattacharyya and Sourav Chakraborty. Property testing of joint distributions using conditional samples. *ACM Trans. Comput. Theory*, 2018. 3

[BCY22] Arnab Bhattacharyya, Clément L. Canonne, and Joy Qiping Yang. Independence testing for bounded degree bayesian network. *CoRR*, abs/2204.08690, 2022. 3

[BDKR05] Tugkan Batu, Sanjoy Dasgupta, Ravi Kumar, and Ronitt Rubinfeld. The complexity of approximating the entropy. *SIAM Journal on Computing*, 2005. 3

[BFLR20] Omri Ben-Eliezer, Eldar Fischer, Amit Levi, and Ron D. Rothblum. Hard properties with (very) short pcpps and their applications. In *ITCS*, 2020. 11, 40, 42

[BGMV20] Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, and N. V. Vinodchandran. Efficient distance approximation for structured high-dimensional distributions via learning. In *NeurIPS*, 2020. 3

[BHR05] Eli Ben-Sasson, Prahladh Harsha, and Sofya Raskhodnikova. Some 3cnf properties are hard to test. *SIAM J. Comput.*, 2005. 40

[Bir46] Garrett Birkhoff. Three observations on linear algebra. *Univ. Nac. Tacuman, Rev. Ser. A*, 1946. 32

[BY22] Arnab Bhattacharyya and Yuichi Yoshida. *Property Testing - Problems and Techniques*. Springer, 2022. 3

[Can20] Clément L. Canonne. A Survey on Distribution Testing: Your Data is Big. But is it Blue? *Theory of Computing*, (9), 2020. 3

[Can22] Clément L. Canonne. Topics and Techniques in Distribution Testing: A Biased but Representative Sample. March 2022. 3

[CCK+21] Clément L. Canonne, Xi Chen, Gautam Kamath, Amit Levi, and Erik Waingarten. Random restrictions of high dimensional distributions and uniformity testing with subcube conditioning. In *SODA*, 2021. 3

[CDKS17] Clément L. Canonne, Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Testing bayesian networks. In *COLT*, 2017. 3

[CDKS20] Clément L Canonne, Ilias Diakonikolas, Daniel M Kane, and Alistair Stewart. Testing bayesian networks. *IEEE Transactions on Information Theory*, 2020. 3

[CDVV14] Siu-On Chan, Ilias Diakonikolas, Paul Valiant, and Gregory Valiant. Optimal algorithms for testing closeness of discrete distributions. In *SODA*, 2014. 3

[CFGM16] Sourav Chakraborty, Eldar Fischer, Yonatan Goldhirsh, and Arie Matsliah. On the power of conditional samples in distribution testing. *SIAM J. Comput.*, 2016. 6

[Cha00] B. Chazelle. *The Discrepancy Method: Randomness and Complexity*. Cambridge University Press, Cambridge, New York, 2000. 14

[CJLW21] Xi Chen, Rajesh Jayaram, Amit Levi, and Erik Waingarten. Learning and testing junta distributions with sub cube conditioning. In *COLT*, 2021. 3

[CKS20] Clément L. Canonne, Gautam Kamath, and Thomas Steinke. The discrete gaussian for differential privacy. In *NeurIPS*, 2020. 3

[CM19] Sourav Chakraborty and Kuldeep S. Meel. On testing of uniform samplers. In *AAAI*, 2019. 3

[CRS15] Clément L. Canonne, Dana Ron, and Rocco A. Servedio. Testing probability distributions using conditional samples. *SIAM J. Comput.*, 2015. 6

[CS10] Artur Czumaj and Christian Sohler. Sublinear-time algorithms. In *Property Testing - Current Research and Surveys*. 2010. 3

[DK20] Ilias Diakonikolas and Daniel M Kane. Robust high-dimensional statistics., 2020. 3

[DKN14] Ilias Diakonikolas, Daniel M Kane, and Vladimir Nikishkin. Testing identity of structured distributions. In *SODA*, 2014. 3

[DKS18] Ilias Diakonikolas, Daniel M Kane, and Alistair Stewart. Sharp bounds for generalized uniformity testing. In *NeurIPS*, 2018. 3

[DLM+07] Ilias Diakonikolas, Homin K Lee, Kevin Matulef, Krzysztof Onak, Ronitt Rubinfeld, Rocco A Servedio, and Andrew Wan. Testing for concise representations. In *FOCS*, 2007. 5

[DP09] D.P. Dubhashi and A. Panconesi. Concentration of Measure for the Analysis of Randomized Algoritms. In *Cambridge*, 2009. 62

[Fis01] Eldar Fischer. The art of uninformed decisions. *Bull. EATCS*, page 97, 2001. 3

[Fis04] Eldar Fischer. The art of uninformed decisions: A primer to property testing. In *Current Trends in Theoretical Computer Science: The Challenge of the New Century Vol 1: Algorithms and Complexity Vol 2: Formal Models and Semantics*. World Scientific, 2004. 15

[Gol17] Oded Goldreich. *Introduction to Property Testing*. Cambridge University Press, 2017. 3

[GOS+09] Parikshit Gopalan, Ryan O'Donnell, Rocco A. Servedio, Amir Shpilka, and Karl Wimmer. Testing fourier dimensionality and sparsity. In *ICALP*, 2009. 5

[GR97] Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. In *STOC*, 1997. 6

[GR11] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*. Springer, 2011. 3

[GR22] Oded Goldreich and Dana Ron. Testing distributions of huge objects. In *ITCS*, 2022. 3, 6, 7, 30, 37, 46

[GT03] Oded Goldreich and Luca Trevisan. Three theorems regarding testing graph properties. *Random Structures & Algorithms*, 2003. 6, 10

[GW21] Oded Goldreich and Avi Wigderson. Non-adaptive vs adaptive queries in the dense graph testing model. In *FOCS*, 2021. 6

[Hau95] David Haussler. Sphere packing numbers for subsets of the boolean n-cube with bounded vapnik-chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 1995. 28, 29

[Hoe94] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The collected works of Wassily Hoeffding*. Springer, 1994. 63

[Jan04] Svante Janson. Large Deviations for Sums of Partly Dependent Random Variables. *Random Structures & Algorithms*, 2004. 63

[Mat99] J. Matoušek. *Geometric Discrepancy: An Illustrated Guide*. Algorithms and Combinatorics. Springer, Berlin, New York, 1999. 14

[Mat02] Jirí Matousek. *Lectures on Discrete Geometry*, volume 212 of *Graduate texts in mathematics*. Springer, 2002. 14

[MPC20] Kuldeep S. Meel, Yash Pote, and Sourav Chakraborty. On testing of samplers. In *NeurIPS*, 2020. 3

[PA95] J. Pach and P. K. Agarwal. *Combinatorial Geometry*. John Wiley & Sons, New York, NY, 1995. 14

[Pan08] Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 2008. 3

[PM21] Yash Pote and Kuldeep S. Meel. Testing probabilistic circuits. In *NeurIPS*, 2021. 3

[Ron08] Dana Ron. Property testing: A learning theory perspective. *Found. Trends Mach. Learn.*, 2008. 3

[Ron09] Dana Ron. Algorithmic and analysis techniques in property testing. *Found. Trends Theor. Comput. Sci.*, 2009. 3

[RS11] Ronitt Rubinfeld and Asaf Shapira. Sublinear time algorithms. *SIAM J. Discret. Math.*, 2011. 3

[RS15] Dana Ron and Rocco A Servedio. Exponentially improved algorithms and lower bounds for testing signed majorities. *Algorithmica*, 2015. 6

[Ser10] Rocco A Servedio. Testing by implicit learning: a brief survey. *Property Testing*, pages 197–210, 2010. 5

[Val11] Paul Valiant. Testing Symmetric Properties of Distributions. *SIAM Journal on Computing*, 2011. 3

[VC15] Vladimir N Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer, 2015. 14

[VN53] John Von Neumann. A certain zero-sum two-person game equivalent to the optimal assignment problem. *Contributions to the Theory of Games*, 1953. 32

[VV10] Gregory Valiant and Paul Valiant. A clt and tight lower bounds for estimating entropy. In *Electron. Colloquium Comput. Complex.*, 2010. 10, 46

[VV17] Gregory Valiant and Paul Valiant. Estimating the unseen: Improved estimators for entropy and other properties. *Journal of the ACM (JACM)*, 2017. 3

[Yao77] Andrew Chi-Chin Yao. Probabilistic computations: Toward a unified measure of complexity. In *FOCS*, 1977. 15

A Some probability results

Lemma A.1 (**Multiplicative Chernoff bound** ([DP09]))**.** *Let $X_1, \ldots, X_n$ be independent random variables such that $X_i \in [0,1]$. For $X = \sum_{i=1}^{n} X_i$ and $\mu = \mathbb{E}[X]$, the following holds for any $0 \leq \delta \leq 1$.*

$$\Pr(|X - \mu| \geq \delta\mu) \leq 2\exp\left(-\mu\delta^2/3\right).$$

Lemma A.2 (**Additive Chernoff bound** ([DP09]))**.** *Let $X_1, \ldots, X_n$ be independent random variables such that $X_i \in [0,1]$. For $X = \sum_{i=1}^{n} X_i$ and $\mu_l \leq \mathbb{E}[X] \leq \mu_h$, the following hold for any $\delta > 0$.*

(i) $\Pr\left(X \geq \mu_h + \delta\right) \leq \exp\left(-2\delta^2/n\right)$.

(ii) $\Pr\left(X \leq \mu_l - \delta\right) \leq \exp\left(-2\delta^2/n\right)$.

Lemma A.3 (**Hoeffding's Inequality** ([DP09]))**.** *Let $X_1, \ldots, X_n$ be independent random variables such that $a_i \leq X_i \leq b_i$ and $X = \sum_{i=1}^{n} X_i$. Then, for all $\delta > 0$,*

$$\Pr\left(|X - \mathbb{E}[X]| \geq \delta\right) \leq 2\exp\left(-2\delta^2 / \sum_{i=1}^{n} (b_i - a_i)^2\right).$$

Lemma A.4 (**Hoeffding's Inequality for sampling without replacement** ([Hoe94])). *Let n and m be two integers such that $1 \leq n \leq m$, and $x_1, \ldots, x_m$ be real numbers, with $a \leq x_i \leq b$ for every $i \in [m]$. Suppose that I is a set that is drawn uniformly from all subsets of $[m]$ of size n, and let $X = \sum_{i \in I} x_i$. Then, for all $\delta > 0$,*

$$\Pr\left(|X - \mathbb{E}[X]| \geq \delta\right) \leq 2\exp\left(-2\delta^2/n \cdot (b-a)^2\right).$$

Now let us consider the following observation which states that if the normalized Hamming distance between two vectors $\mathbf{X}$ and $\mathbf{Y}$ are small, the same also holds with high probability when $\mathbf{X}$ and $\mathbf{Y}$ are projected on a set of random indices K. A similar result also holds when the distance is large between the two vectors $\mathbf{X}$ and $\mathbf{Y}$.

Observation A.5 (**Approximating-string-distances).** For $\mathbf{U}, \mathbf{V} \in \{0,1\}^n$ and assume that $K \subseteq [n]$ is a set of indices chosen uniformly at random without replacement. Then the following holds with probability at least $1 - e^{-\mathcal{O}(\delta^2|K|)}$:

$$|d_H(\mathbf{U}, \mathbf{V}) - d_H(\mathbf{U}\mid_K, \mathbf{V}\mid_K)| \leq \delta.$$

Proof. Follows from the fact that sampling without replacement is as good as sampling with replacement (Lemma A.4). □

Lemma A.6 (**Chernoff bound for bounded dependency** ([Jan04])). *Let $X_1, \ldots, X_n$ be random variables such that $a_i \leq X_i \leq b_i$ and $X = \sum_{i=1}^{n} X_i$. Let $\mathcal{D}$ be the (directed)* dependency *graph, where $V(\mathcal{D}) = \{X_1, \ldots, X_n\}$ and X_i is completely independent of all variables X_j for which (X_i, X_j) is not a directed edge. Then for any $\delta > 0$,*

$$\Pr(|X - \mathbb{E}[X]| \geq \delta) \leq 2e^{-2\delta^2/\chi^*(\mathcal{D}) \sum_{i=1}^{n} (b_i - a_i)^2}.$$

where $\chi^*(\mathcal{D})$ denotes the fractional chromatic number *of $\mathcal{D}$.*

Corollary A.7 (**Corollary of Lemma A.6).** *Let $X_1, \ldots, X_n$ be indicator random variables such that the dependency graph is a disjoint union of n/k many k size cliques. For $X = \sum_{i=1}^{n} X_i$ and $\mu_l \leq \mathbb{E}[X] \leq \mu_h$, the followings hold for any $\delta > 0$:*

(i) $\Pr\left(X \geq \mu_h + \delta\right) \leq \exp\left(\frac{-2\delta^2}{kn}\right)$,

(ii) $\Pr\left(X \leq \mu_\ell - \delta\right) \leq \exp\left(\frac{-2\delta^2}{kn}\right)$.

Proof. Follows from the fact that the dependency graph has chromatic number k, and the fractional chromatic number of a graph is at most the chromatic number of any graph. □

Algorithm 1: FIND-PERMUTATION

Input: Sample and Query access to a distribution D over $\{0,1\}^N$.
Output: Either a permutation $\pi : [N] \to [N]$, or FAIL.

(i) First take a multi-set $\mathcal{X}$ of $\mathcal{O}(\log^2 N/\alpha)$ samples from D, and query all the entries of the sampled vectors of $\mathcal{X}$ to know the vectors of $\mathcal{X}$ completely.

(ii) Find the set of distinct vectors in $\mathcal{X}$ that have exactly one 1. If no such vector exists or there is more than one such vector, FAIL. Otherwise, denote by $\mathbf{U}'$ the vector that has exactly one 1, and denote the corresponding index by i^*. Set $\pi(i^*) = 1$, and proceed to the next step.

(iii) Find the set of distinct vectors $\mathcal{S}' \subseteq \mathcal{X} \setminus \{\mathbf{U}'\}$ such that every vector in $\mathcal{S}'$ has 1 at the index i^* and has at least another 1 among other indices. If no such vector exists, or $|\mathcal{S}'| \neq b$, FAIL. Otherwise, if the vectors of $\mathcal{S}'$ form a chain $\mathbf{V}'_1, \ldots, \mathbf{V}'_b$, where $\mathbf{V}'_j$ has exactly $j+1$ many 1, then set $\pi(i_j) = j+1$, where i_j is the index where $\mathbf{V}'_j$ has 1, but $\mathbf{V}'_{j-1}$ has 0 there, for every $j \in [b]$ (denoting $\mathbf{V}'_0 = \mathbf{U}'$ for the purpose here). Also, set $B' = (i_1, \ldots, i_b)$. If $\mathcal{S}'$ does not form a chain $\mathbf{V}'_1, \ldots, \mathbf{V}'_b$ as mentioned above, FAIL.

(iv) Let $\mathcal{T}' \subseteq \mathcal{X}$ be the set of distinct vectors such that every vector in $\mathcal{T}'$ has 0 at the index i^*, and does not have 1 in all indices of B'. If no such vector exists, FAIL. For every j, denote by $\mathbf{W}'_j$ the vector in $\mathcal{T}'$ for which $\mathbf{W}'_j \mid_{B'} = b(j)$, where $b(j)$ denotes the binary representation of j. For every $j \in \{0\} \cup [\lceil \log kn \rceil - 1]$, if either there are no vectors $\mathbf{W}'_j \in \mathcal{T}'$ or there is more than one distinct vector with $\mathbf{W}'_j \mid_{B'} = b(j)$, FAIL. Also, if there is any vector in $\mathbf{W}'_j \in \mathcal{T}'$ such that $\mathbf{W}'_j \mid_{B'} = b(j)$ for $\log kn \leq j < 2^b - 1$, FAIL.

(v) For any $i \in [N] \setminus (\{i^*\} \cup B')$, let l_i be the integer with binary representation $(\mathbf{W}'_0)_i, \ldots, (\mathbf{W}'_{\lceil \log kn \rceil - 1})_i$. Set $\pi(i) = b + 2 + l_i$ for each $i \in [N] \setminus (\{i^*\} \cup B')$. If π is not a permutation of $[N]$, FAIL.

(vi) Take another multi-set $\mathcal{X}'$ of $\mathcal{O}(\log^2 N/\alpha)$ samples from D, and query all the entries of the sampled vectors of $\mathcal{X}'$ to know the vectors of $\mathcal{X}'$ completely. Let $\mathcal{Y}$ be the set of vectors in $\mathcal{X}'$ such that $\mathcal{Y} = \{\mathbf{Z} \in \mathcal{X}' : \mathbf{Z} \mid_{\{i^*\} \cup B'} \neq \mathbf{01}^b\}$. If $|\mathcal{Y}| / |\mathcal{X}'| > 4\alpha$, FAIL. Otherwise, output the permutation π.

Algorithm 2: ALG-ADAPTIVE

Input: Sample and Query access to a distribution D over $\{0,1\}^N$, and a parameter $\varepsilon \in (0,1)$.

Output: Either ACCEPT or REJECT.

(i) Call FIND-PERMUTATION. If FIND-PERMUTATION returns FAIL, REJECT. Otherwise, let π be the permutation returned by FIND-PERMUTATION. Denote for convenience $i^* = \pi^{-1}(1), B' = \pi^{-1}(B)$, and $C'_j = \pi^{-1}(C_j)$ for every $j \in [n]$.

(ii) Take a multi-set $\mathcal{X}$ of $\mathcal{O}(1/\varepsilon)$ samples from D, and query all the entries of the sampled vectors of $\mathcal{X}$ to know the vectors of $\mathcal{X}$ completely. If there is any vector $\mathbf{X}$ for which $\mathbf{X}\,|_{\{i^*\}\cup B'} = \mathbf{01}^b$ (according to the permutation π obtained from Step (i)) for which $\mathbf{X}_\pi$ is not in the image of FE (i.e. it is not a valid encoding of any vector in $\{0,1\}^n$), REJECT. Otherwise, proceed to the next step.

(iii) Take a sequence of samples $\mathcal{Y}$ such that $|\mathcal{Y}| = \mathcal{O}(n/\varepsilon)$ from D. Now construct the sequence of vectors $\mathcal{Y}'$ such that $\mathcal{Y}' = \{\mathbf{Y} \in \mathcal{Y} : \mathbf{Y}\,|_{\{i^*\}\cup B'} = \mathbf{01}^b\}$ by querying the indices corresponding to $\{i^*\} \cup B'$.

(iv) Call SUPP-EST$(\mathcal{Y}', \varepsilon/3)$ (from Lemma 8.8), where a query to an index j is simulated by querying the indices of C'_j and decoding the obtained vector with respect to to SE (that is, checking whether the restriction of the queried vector to C'_j is equal to SE$(i,0)$ for some i, or equal to SE$(i,1)$ for some i). REJECT if any of the following conditions hold:

(a) $|\mathcal{Y}'| \,/\, |\mathcal{Y}| \leq 1/2$ (due to the absence of sufficiently many samples in $\mathcal{Y}'$ to apply SUPP-EST).

(b) SUPP-EST$(\mathcal{Y}', \varepsilon/3)$ queries an index j from some $\mathbf{Y}_i$ corresponding to an invalid encoding of $\mathbf{Y}_i\,|_{C'_j}$ (that is, when $\mathbf{Y}_i\,|_{C'_j}$ is not in the image of SE).

(c) SUPP-EST$(\mathcal{Y}', \varepsilon/3)$ outputs REJECT.

Otherwise, ACCEPT.