

Limitations of Membership Queries in Testable Learning

Jane Lange  

Massachusetts Institute of Technology, Cambridge, MA, USA

Mingda Qiao  

University of Massachusetts Amherst, MA, USA

Abstract

Membership queries (MQ) often yield speedups for learning tasks, particularly in the distribution-specific setting. We show that in the *testable learning* model of Rubinfeld and Vasilyan [21], membership queries cannot decrease the time complexity of testable learning algorithms beyond the complexity of sample-only distribution-specific learning. In the testable learning model, the learner must output a hypothesis whenever the data distribution satisfies a desired property, and if it outputs a hypothesis, the hypothesis must be near-optimal.

We give a general reduction from sample-based *refutation* of boolean concept classes, as presented in [23, 17], to testable learning with queries (TL-Q). This yields lower bounds for TL-Q via the reduction from learning to refutation given in [17]. The result is that, relative to a concept class and a distribution family, no m -sample TL-Q algorithm can be super-polynomially more time-efficient than the best m -sample PAC learner.

Finally, we define a class of “statistical” MQ algorithms that encompasses many known distribution-specific MQ learners, such as those based on influence estimation or subcube-conditional statistical queries. We show that TL-Q algorithms in this class imply efficient statistical-query refutation and learning algorithms. Thus, combined with known SQ dimension lower bounds, our results imply that these efficient membership query learners cannot be made testable.

2012 ACM Subject Classification Theory of computation → Query learning

Keywords and phrases Testable learning, PAC learning

Digital Object Identifier 10.4230/LIPIcs.ITCS.2026.91

Related Version *Full Version*: <https://arxiv.org/abs/2512.02279>

Funding *Jane Lange*: Supported in part by NSF Awards CCF-2006664, DMS-2022448, CCF-2310818, and Big George Fellowship

1 Introduction

In distribution-specific PAC learning, a learning algorithm is only required to output a competitive hypothesis when the data distribution satisfies some property. Distribution-specific PAC often allows for much more efficient learning than distribution-free PAC, but with the following shortcoming: if the distribution does not satisfy the property, then the behavior of the learner is completely undefined.

A *testable agnostic learning* algorithm [21] alleviates this shortcoming by combining a distribution-specific learner with a tester for the desired property. It may output a hypothesis or it may reject the distribution and output \perp . The testable learner is run on i.i.d. samples from an unknown distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$, and has the following behavior:

- **Soundness:** If the learner outputs a hypothesis h , then with high probability

$$\Pr_{(x,y) \sim \mathcal{D}} [h(x) \neq y] \leq \text{opt} + \varepsilon,$$

where opt is the error of the best concept in the concept class. A *semi-agnostic* variant of this condition, with $\text{opt} + \varepsilon$ replaced by $O(\text{opt}) + \varepsilon$, has also been considered.



© Jane Lange and Mingda Qiao;

licensed under Creative Commons License CC-BY 4.0

17th Innovations in Theoretical Computer Science Conference (ITCS 2026).

Editor: Shubhangi Saraf; Article No. 91; pp. 91:1–91:23

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- **Completeness:** If the distribution \mathcal{D} has the desired property, then the learner outputs a hypothesis (instead of \perp) with high probability.

There is a significant body of work studying the sample and time complexities of testable learning for various concept classes and distribution properties in both the agnostic ($\text{opt} + \varepsilon$) and semi-agnostic ($O(\text{opt}) + \varepsilon$) settings [21, 13, 8, 12, 22]. There are efficient algorithms for testable learning in cases where distribution-free learning cannot be done efficiently.

1.1 The Power of Membership Queries in Agnostic Learning

One might hope to testably learn more efficiently by strengthening the learner’s access to the data distribution. In the *membership query* (MQ) model, we think of the data as being drawn from a distribution \mathcal{D}_x over \mathcal{X} and labeled by some unknown function $f : \mathcal{X} \rightarrow \{0, 1\}$. The learner gets i.i.d. samples from \mathcal{D}_x and may also query any point $x \in \mathcal{X}$ and receive its label $f(x)$.

The work of [10] shows that under the standard cryptographic assumption of one-way functions, membership queries can speed up agnostic learning in the distribution-specific setting, but not in the distribution-free setting. In the distribution-free setting, every concept class that can be agnostically learned with MQs can be agnostically learned with just random examples. In contrast, in the uniform-distribution-specific setting, there exists a concept class that can be learned strictly more efficiently with MQs.

Separations exist for more “natural” concept classes under the stronger assumption that learning sparse parities with noise (LSPN) is hard. For example, over the uniform distribution on $\{0, 1\}^n$, k -juntas can be learned in $\text{poly}(n) \cdot 2^{O(k)}$ time with membership queries [3, 20]. On the other hand, there is a statistical query lower bound of $n^{\Omega(k)}$ [4], and if LSPN is hard then one cannot hope to do better than this bound with random examples. Similarly, polynomial-size decision trees can be learned improperly in polynomial time [18, 14] and properly in $n^{O(\log \log n)}$ time [1] with membership queries, while there is an SQ lower bound of $n^{\Omega(\log n)}$ [2], and LSPN implies one cannot do better than this bound either.

1.2 Limitations of Membership Queries in Testable Learning

If membership queries do help in the distribution-specific setting but do not help in the distribution-free setting, one may then naturally wonder whether they ought to help in the testable setting as well. A *testable learner with queries* (TL-Q) has both sample access to the unknown data distribution and membership query access to the unknown function, and must satisfy the soundness and completeness guarantees of ordinary testable learning.

- **Question 1.** *How much can membership queries speed up the task of testable learning?*

Our results show that membership queries are quite weak in the TL-Q setting. Particularly, whenever agnostic learning with random examples is hard – as is believed to be the case for juntas and decision trees – testable learning is hard as well, even with queries.

- **Theorem 2** (Corollary 17, informal). *If a concept class \mathcal{C} is agnostically testably learnable with queries in time t over a distribution \mathcal{D} , then it is agnostically learnable with random examples in time $\text{poly}(t)$ over \mathcal{D} as well.*

- **Corollary 3.** *If LSPN is hard, then no concept class containing k -parities as a subset can be agnostically testably learned in $n^{o(k)}$ time over the uniform distribution, even with membership queries.*

Furthermore, we show that SQ lower bounds rule out a large class of natural query-based learning algorithms. We define a class of “statistical” membership query (MQ-SQ) algorithms – those that use membership queries only to sample from particular distributions over the input domain. For example, algorithms that use MQs only to estimate influences or to make SQs over large subsets of $\{0, 1\}^n$ are MQ-SQ algorithms (this includes the aforementioned uniform-distribution algorithms of [18, 14, 1]). We show that such algorithms cannot be “made testable” with respect to the uniform distribution without introducing non-statistical use of membership queries, due to the SQ lower bounds.

► **Theorem 4** (Theorem 29, informal). *If a concept class \mathcal{C} is testably learnable in time t over a distribution \mathcal{D} by an MQ-SQ algorithm, then the SQ dimension of \mathcal{C} with respect to \mathcal{D} is at most $\text{poly}(t)$.*

1.3 Technical Overview

Our reductions are through the intermediate task of *refutation*. Refutation, as presented in [23, 17], is the problem of distinguishing examples correlated with some function in the concept class from examples labeled uniformly at random. The work of [23] shows that in the distribution-free setting, refutation and realizable learning are polynomially equivalent, and the work of [17] shows an analogous statement in the distribution-specific agnostic setting. By giving an efficient reduction from distribution-specific refutation (without queries) to testable learning (with queries), we show that distribution-specific agnostic learning reduces to TL-Q as well.

As a warm-up, consider a special case of refutation where the labels are promised to either be completely random or exactly match some function in the class \mathcal{C} . Let the distribution be uniform over $\{0, 1\}^n$. Assume for simplicity that all functions in \mathcal{C} are balanced, i.e., $\mathbb{E}_{x \sim \{0, 1\}^n} [f(x)] = 1/2$.

► **Definition 5** (Exact refutation over the uniform distribution, informal). *An exact refutation algorithm for a concept class \mathcal{C} takes an m -tuple $\{(x_1, y_1), \dots, (x_m, y_m)\}$ of examples where the x 's are drawn uniformly at random from $\{0, 1\}^n$. It outputs either noise or structure with the following guarantees:*

- **Completeness:** *If the examples are consistent with some $g \in \mathcal{C}$, then*

$$\Pr[\mathcal{A} \text{ outputs structure}] \geq 2/3.$$

- **Soundness:** *If the y_i 's are drawn i.i.d. from Bernoulli(1/2), then*

$$\Pr[\mathcal{A} \text{ outputs noise}] \geq 2/3.$$

Suppose we want to implement exact refutation using a TL-Q algorithm. If we had any agnostic learning algorithm that did not require queries, the task would be trivially easy: split $\{(x_1, y_1), \dots, (x_m, y_m)\}$ into training and test sets, run the learner on the training set, estimate the error of the returned hypothesis on the test set, and output **structure** if the test error is, say, $\leq 1/10$. If we are in the **structure** case, the error will be $\leq \epsilon$, and if we are in the **noise** case, with high probability the error will be close to $1/2$.

Instead we have to answer queries, so we will answer them randomly. Specifically, we will draw a random function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, and whenever the TL-Q algorithm wants to make a query, we will answer according to the f we chose. We will filter both the training and test sets to just those points where $y = f(x)$. This means that we essentially sample from a domain that is uniform over the portion of $\{0, 1\}^n$ where $y(x)$ agrees with $f(x)$.

As before, if the TL-Q algorithm produces a hypothesis, we will output `structure` if the error is $\leq 1/10$ and `noise` if the error is greater. But since TL-Q can also reject the instance and output \perp , if it does so, we will output `structure`.

Notice that in the noise case, each x_i is filtered out independently with probability $1/2$; therefore the distribution of samples is uniform. By completeness of the TL-Q algorithm, it must then output a hypothesis, and with high probability the error will be close to $1/2$. In the structure case, however, the distribution of x_i 's may be far from uniform, in which case the TL-Q algorithm may output \perp . It may also output a hypothesis – but by soundness, the hypothesis must have error $\leq \varepsilon$, since the samples come from a distribution such that $y = f(x)$ for every x in its support.

Our reduction from refutation to TL-Q is basically a generalization of this strategy, adapted to handle unbalanced functions and accept functions that are close to, but not exactly, in \mathcal{C} .

1.3.1 An SQ-Preserving Reduction

We observe that some membership query algorithms, such as those of [18, 14, 1], use membership queries only to estimate statistical properties of the unknown function. We roughly categorize MQ-SQ queries as follows (formalized in Definition 20):

- Standard SQs: queries of the form $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)]$ or $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)]$ for a test function ϕ . These don't require membership queries to implement.
- Pair SQs: queries of the form $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)f(\pi(x))]$, where π is a permutation of the domain without fixed points. This generalizes influence estimation.
- Customized distribution SQs: Any of the above queries, where the expectation is taken over a specific (and sufficiently spread-out) distribution \mathcal{D}^* instead of the unknown distribution \mathcal{D} . This generalizes making SQs over restrictions of $\{0, 1\}^n$.

Our goal is to simulate an MQ-SQ testable learner by making only SQs to the unknown distribution $\mathcal{D}^{\text{refut.}}$ (over $\mathcal{X} \times \{0, 1\}$) in the refutation instance. As in the non-SQ setting, we choose a random function to be the target function and answer queries according to that random function.

For example, the customized-distribution query $\mathbb{E}_{x \sim \mathcal{D}^*} [f(x)\phi(x)]$ is easy to simulate with just one SQ to $\mathcal{D}^{\text{refut.}}$: simply estimate the mean $p := \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y]$, and answer the MQ-SQ with $p \cdot \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)]$ (sampling from \mathcal{D}^* requires neither samples nor membership queries, since it's the customized distribution). The value of this MQ-SQ concentrates around this estimate as each $f(x)$ is an independent random variable with mean p .

Pair SQs are handled similarly, though in this case the random variables $f(x)f(\pi(x))$ are not independent. However, since the dependence graph of these variables decomposes into cycles, we can partition the graph into large independent sets and prove concentration of the variables within the independent sets. Thus, for example, the MQ-SQ $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)f(\pi(x))]$ can be answered with $p \cdot \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x)y]$, which is an SQ to $\mathcal{D}^{\text{refut.}}$.

1.4 Related Work and Discussion

The power of membership queries

It is well known that in the realizable setting, PAC learners with membership queries are strictly stronger than PAC learners without them under standard cryptographic assumptions [9, 11]. The work of [10] establishes an equivalence between PAC with random examples and PAC with membership queries in the distribution-free agnostic setting, and a separation in the distribution-specific agnostic setting.

Testable learning and friends

There are many papers that address the computational and sample complexities of testably learning various natural concept classes; these works are in the standard agnostic testable learning model. Some examples, but certainly not all, are the works of [21, 13, 8, 12, 22]. Some works addressing related problems include [16, 15, 19].

The work of [13] characterizes the sample complexity of testable learning by the Rademacher complexity, which is especially relevant to this work in light of the result of [17], which establishes refutation complexity as an analogue of Rademacher complexity for the computationally bounded setting.

Learning and refutation

A connection between learning and refutation was first introduced in [6] as a means of proving computational lower bounds for learning based on the assumption that refuting random CSPs is hard, and other works including those of [7, 5] use this method to give conditional lower bounds for various learning problems. Of particular relevance to this work are [23, 17], which give polynomial equivalences between PAC learning and refutation.

1.4.1 Directions for Future Work

While our work reduces ordinary sample-based PAC learning to query-based testable learning, we do not resolve the strongest, most natural question on the power of membership queries: whether sample-based *testable* learning reduces to query-based testable learning. This would be a strictly stronger lower bound for TL-Q than anything obtainable through refutation, as there are function classes for which refutation is known to be easier than testable learning with samples – for example, the class of monotone functions. It is proven in [21] that testably learning monotone functions on the uniform distribution requires $2^{\Omega(n)}$ samples. On the other hand, agnostic learning (and therefore refutation) can be done in $2^{O(\sqrt{n})}$ time and samples.

We leave this possible stronger lower bound as an open question for future work.

► **Conjecture 6.** *If a concept class \mathcal{C} is agnostically testably learnable with queries in time t over a distribution \mathcal{D} , then it is agnostically testably learnable with samples in time $\text{poly}(t)$ over \mathcal{D} as well.*

We also remark that in the semi-agnostic setting, our method relates semi-agnostic TL-Q to *weak* agnostic learning. It is an open question for future work to resolve the connection between semi-agnostic TL-Q and semi-agnostic learning as well.

2 Preliminaries

2.1 Distances and Errors

► **Definition 7** (Distance of functions and distance to a concept class). *Relative to a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ with \mathcal{X} -marginal \mathcal{D}_x , we denote*

$$\text{dist}_{\mathcal{D}_x}(f, g) = \Pr_{x \sim \mathcal{D}_x} [f(x) \neq g(x)]$$

$$\text{err}_{\mathcal{D}}(f) = \Pr_{(x,y) \sim \mathcal{D}} [f(x) \neq y].$$

We also use the following notation to denote the classification error of the most accurate concept in a class, which we often refer to as opt :

$$\text{dist}_{\mathcal{D}_x}(f, \mathcal{C}) = \inf_{g \in \mathcal{C}} \text{dist}_{\mathcal{D}_x}(f, g).$$

$$\text{err}_{\mathcal{D}}(\mathcal{C}) = \inf_{g \in \mathcal{C}} \text{err}_{\mathcal{D}}(g).$$

2.2 Refutation and Learning

We state a definition of refutation similar to the definition presented in [17]. We have modified it to use classification error rather than correlation, for ease of use in our $\{0, 1\}$ -labeled setting ([17] uses $\{-1, 1\}$ labels).

► **Definition 8** (η -refutation). *Let $\mathcal{C} \subseteq \{f : \mathcal{X} \rightarrow \{0, 1\}\}$ be a concept class over a finite input domain \mathcal{X} , and let \mathcal{F} be a family of distributions on \mathcal{X} . An η -refutation algorithm \mathcal{A} for \mathcal{C} on \mathcal{F} with m samples is an algorithm that takes an m -tuple of labeled examples $\{(x_1, y_1), \dots, (x_m, y_m)\}$ and outputs either noise or structure. If the examples are i.i.d. from a distribution \mathcal{D} over $\mathcal{X} \times \{0, 1\}$ such that the marginal on \mathcal{X} is some $\mathcal{D}_x \in \mathcal{F}$, then the following guarantees hold:*

- **Completeness:** *If there exists $g \in \mathcal{C}$ such that $\text{err}_{\mathcal{D}}(g) \leq \eta$, then*

$$\Pr_{\substack{\{(x_i, y_i)\} \sim \mathcal{D} \\ \text{internal randomness of } \mathcal{A}}} [\mathcal{A} \text{ outputs structure}] \geq 2/3.$$

- **Soundness:** *If the y_i 's are drawn i.i.d. from $\text{Bernoulli}(1/2)$, then*

$$\Pr_{\substack{\{(x_i, y_i)\} \sim \mathcal{D} \\ \text{internal randomness of } \mathcal{A}}} [\mathcal{A} \text{ outputs noise}] \geq 2/3.$$

We also define a similar but stronger task:

► **Definition 9** (Biased (α, η) -refutation). *A biased- (α, η) -refutation algorithm is as above except the soundness condition is the following:*

- **Soundness:** *For all $p \in [\alpha, 1 - \alpha]$, if the y_i 's are drawn i.i.d. from $\text{Bernoulli}(p)$, then*

$$\Pr_{\substack{\{(x_i, y_i)\} \sim \mathcal{D} \\ \text{internal randomness of } \mathcal{A}}} [\mathcal{A} \text{ outputs noise}] \geq 2/3.$$

Here we state definitions and facts used in [17]'s reduction from refutation to agnostic learning. Again we modify these statements to use classification error rather than correlation.

► **Definition 10** (Weak agnostic learning). *A (γ, α) -weak agnostic learner for the concept class \mathcal{C} over the distribution \mathcal{D} outputs a hypothesis h satisfying the following:*

$$\text{err}_{\mathcal{D}}(h) \leq \frac{1 + \alpha - \gamma}{2} + \gamma \text{err}_{\mathcal{D}}(\mathcal{C}).$$

► **Lemma 11** (Learning by refutation: Lemma 6 of [17]). *Suppose there is an η -refutation algorithm for the class \mathcal{C} over distribution \mathcal{D} running in $T(n)$ time with m samples. Then there is an algorithm that runs in $T(n) \cdot \frac{m^2}{\varepsilon^2}$ and uses $O(\frac{m^3}{\varepsilon^2})$ samples to agnostically learn \mathcal{C} on \mathcal{D} with excess error $1 - 2\eta + \varepsilon$.*

2.3 Testable Learning with Queries

We now define semi-agnostic *testable learning with queries*.

► **Definition 12** (Testable learning with queries, or TL-Q). *A concept class \mathcal{C} over the input domain \mathcal{X} is (c, ε, δ) -PAC-testably-learnable with q queries, m samples, and t time, on a set \mathcal{F} of distributions over \mathcal{X} , if there is a t -time algorithm \mathcal{A} that takes m samples from an unknown distribution $\mathcal{D} \in \mathcal{F}$ and q membership queries to an unknown function f^* , and outputs $h \in \mathcal{C} \cup \{\perp\}$ such that:*

■ *Soundness: If $h \neq \perp$, then*

$$\Pr[\text{dist}_{\mathcal{D}}(h, f^*) \geq c \cdot \text{opt} + \varepsilon] \leq \delta.$$

■ *Completeness: If $\mathcal{D} \in \mathcal{F}$, then*

$$\Pr[h = \perp] \leq \delta.$$

3 Refutation, Learning, and Testable Learning with Queries

3.1 A General Reduction from Refutation to TL-Q

In this section, we show that if a class is efficiently testably-learnable, then it is efficiently refutable as well, with polynomial dependence on the sample, time, and query complexity of the testable learner.

In fact, our Algorithm 1 solves the harder problem of biased refutation (Definition 9), though the distinction between the two will not be relevant until Section 3.3. A biased refutation algorithm can always be used to solve the standard unbiased refutation problem, as the soundness guarantee for biased refutation must always hold when $p = 1/2$. To avoid confusion, we let $\mathcal{D}^{\text{refut}}$ denote the distribution of labeled pairs $(x, y) \in \mathcal{X} \times \{0, 1\}$ in the refutation instance, while \mathcal{D} denotes the unknown marginal distribution in a TL-Q instance.

The main result of this section is the following:

► **Theorem 13.** *Let \mathcal{C} be $(c, \varepsilon, \frac{1}{10})$ PAC-testably-learnable with m samples, q queries, and t time, on a distribution family \mathcal{F} satisfying*

$$m + q/\varepsilon^2 \ll \frac{1}{\sup_{\mathcal{D}_x \in \mathcal{F}}(\|\mathcal{D}_x\|_2)}.$$

Then for any ε satisfying $\varepsilon^2 \geq ck \cdot \sup_{\mathcal{D}_x \in \mathcal{F}}(\|\mathcal{D}_x\|_2)$ for sufficiently large constant k , and any $\eta < \frac{1/2 - 4\varepsilon}{c}$, \mathcal{C} is $(c\eta + 4\varepsilon, \eta)$ -refutable over all members of \mathcal{F} with m' samples and t' time, where

$$m' = O\left(\frac{m + 1/\varepsilon^2}{\varepsilon} + q\right)$$

$$t' = O(m' + t).$$

Algorithm 1 essentially draws a random function f of bias p , where p is the mean of the labels in the refutation distribution $\mathcal{D}^{\text{refut}}$, and filters the samples to just pairs where $y = f(x)$. The drawing of f is “lazy:” to draw f and answer queries to it, it suffices to draw each value of $f(x)$ from $\text{Bernoulli}(p)$ the first time we need to know $f(x)$, then store $(x, f(x))$ in a table for consistency. We implement the random coin by reading a label, as the labels are distributed as $\text{Bernoulli}(p)$.

■ **Algorithm 1** BIASEDREFUTATION(samples, $\eta, \varepsilon, m, q, c$).

-
- 1: **Input:** sample set `samples` of size m' drawn from the refutation distribution $\mathcal{D}^{\text{refut.}}$, gap parameters η and ε , TL-Q parameters m, q, c
 - 2: **Output:** noise, structure, or an error
 - 3:
 - 4: Use the first $C_1 \cdot 1/\varepsilon^2$ samples to estimate $p := \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y]$. If the estimated value $\hat{p} < 2\varepsilon$ or $\hat{p} > 1 - 2\varepsilon$, **return structure**.
 - 5: Reserve the next $C_2 \cdot \left(\frac{m+1/\varepsilon^2}{\varepsilon} + q\right)$ samples to be used only for their labels (whenever a draw from $\text{Bernoulli}(p)$ is required, use a new label from this set).
 - 6: Initialize an empty set S and an empty truth table f .
 - 7: **for** each remaining example (x_i, y_i) **do**
 - 8: Draw $b \sim \text{Bernoulli}(p)$ and store $f(x_i) = b$.
 - 9: If $y_i = f(x_i) = 0$, add x_i to S with probability p .
 - 10: If $y_i = f(x_i) = 1$, add x_i to S with probability $1 - p$.
 - 11: If $y_i \neq f(x_i)$, continue without adding x_i to S .
 - 12: If $|S| < m + C_3/\varepsilon^2$, **return** an error. Let the first m examples be S_{train} and the remaining be S_{test} .
 - 13: Call the TL-Q algorithm \mathcal{A} with the first m members of S . Whenever \mathcal{A} makes a query to some point x , if $f(x)$ is in the table, answer with $f(x)$; otherwise draw $b \sim \text{Bernoulli}(p)$, store $f(x) = b$, then answer $f(x)$.
 - 14: If \mathcal{A} returned \perp , **return structure**.
 - 15: If S_{test} contains any duplicate elements, elements in S_{train} , or elements queried by \mathcal{A} , **return** an error.
 - 16: If \mathcal{A} returned a hypothesis h , evaluate

$$\widehat{\text{err}}(h) = \Pr_{x \sim S_{\text{test}}} [h(x) \neq f(x)].$$

- 17: If $\widehat{\text{err}}(h) > c\eta + 3\varepsilon$, **return noise**. Otherwise **return structure**.
-

3.1.1 Properties of the Filtered Sample Distribution

Before proving the theorem, it will be useful to have the following supporting claims about the distribution that the sets S_{train} and S_{test} are drawn from. This distribution \mathcal{D} – which is the unknown distribution that the TL-Q instance is running on – depends on the random function f . Formally, the PMF of \mathcal{D} is the following, and one may observe that the construction of S in lines 7-11 of Algorithm 1 produces samples from this distribution:

► **Definition 14** (Filtered sample distribution). *Let $\mathcal{D}^{\text{refut.}}$ be a distribution over $\mathcal{X} \times \{0, 1\}$ with \mathcal{X} -marginal \mathcal{D}_x and let $f : \mathcal{X} \rightarrow \{0, 1\}$ be a p -biased random function. Let the function $y(x)$ be defined as $\mathbb{E}_{(x', y') \sim \mathcal{D}^{\text{refut.}}} [y' \mid x' = x]$. Then the filtered sample distribution \mathcal{D} is defined by the following PMF:*

$$\mathcal{D}(x) = \frac{\mathcal{D}_x(x)}{Z} \cdot (p(1 - y(x))(1 - f(x)) + (1 - p)y(x)f(x)),$$

where Z is the normalization factor

$$Z = \mathbb{E}_{x \sim \mathcal{D}_x} [(1 - p)y(x)f(x) + p(1 - y(x))(1 - f(x))].$$

We state the properties below; the proofs are deferred to the full version of this paper.

► **Lemma 15.** For every $\delta \geq 0$, it holds with probability at least $1 - 2 \exp\left(-\Omega\left(\frac{\delta^2 \cdot p^2 (1-p)^2}{\|\mathcal{D}_x\|_2^2}\right)\right)$ over the randomness of f that

$$|Z - p(1-p)| \leq \delta \cdot p(1-p).$$

▷ **Claim 16.** Let $g : \mathcal{X} \rightarrow \{0, 1\}$ be an arbitrary function. For any δ , with probability at least $1 - \exp\left(-\Omega\left(\frac{(\delta p)^2 (1-p)^2}{\|\mathcal{D}_x\|_2^2}\right)\right)$ over the randomness of f , we have

$$\Pr_{x \sim \mathcal{D}} [g(x) \neq f(x)] \leq \Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [g(x) \neq y] + \delta.$$

3.1.2 Proof of Theorem 13

With the above properties in hand, we will now prove the main theorem of this section.

Proof of Theorem 13. Let \mathcal{A} be the $(c, \varepsilon, 1/10)$ -testable learner for \mathcal{C} . We will show that Algorithm 1 is a $(c\eta + 4\varepsilon, \eta)$ refutation algorithm over any $\mathcal{D}^{\text{refut.}}$ with \mathcal{X} -marginal \mathcal{D}_x such that $\mathcal{D}_x \in \mathcal{F}$. Since the samples come from a refutation instance, one of the following must hold:

- Structure: $\Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [g(x) \neq y] \leq \eta$ for some $g \in \mathcal{C}$, or
- Noise: $y \sim \text{Bernoulli}(p)$ for some $p \in [c\eta + 4\varepsilon, 1 - c\eta - 4\varepsilon]$.

We will refer to $\mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y]$ as p regardless of whether we are in the structure case or the noise case. We consider the following possibilities:

- **Structure:** In this case, there is some function $g \in \mathcal{C}$ such that $\Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [g(x) \neq y] \leq \eta$. By setting the constant C_1 large enough, we have by Hoeffding's inequality that $\Pr[|\hat{p} - p| > \varepsilon] \leq 1/100$. With the remaining probability, if $p < \varepsilon$ or $p > 1 - \varepsilon$ we would output structure after line 4, so we will assume from here that $\varepsilon \leq p \leq 1 - \varepsilon$.

We will denote by \mathcal{D} the distribution over \mathcal{X} from which our sample set S is drawn, as discussed in Section 3.1.1. Since $\varepsilon \leq p \leq 1 - \varepsilon$, by our assumption that $\varepsilon^2 \geq ck \cdot \sup_{\mathcal{D}_x \in \mathcal{F}} (\|\mathcal{D}_x\|_2)$ we have

$$\frac{(\varepsilon/c)^2 \cdot p^2 (1-p)^2}{\|\mathcal{D}_x\|_2^2} \geq \Omega\left(\frac{\varepsilon^4}{c^2 \|\mathcal{D}_x\|_2^2}\right) \geq \Omega(k^2).$$

By Claim 16, setting the constant k to be large enough, we then have

$$\text{dist}_{\mathcal{D}}(f, \mathcal{C}) \leq \Pr_{x \sim \mathcal{D}} [f(x) \neq g(x)] + \varepsilon/c \leq \eta + \varepsilon/c$$

with probability at least $1 - \exp(-\Omega(k^2)) \geq 99/100$ over the randomness of f . When this happens, \mathcal{A} has two possible sound behaviors: either output \perp , or output h satisfying

$$\text{dist}_{\mathcal{D}}(h, f) \leq c(\eta + \varepsilon/c) + \varepsilon \leq c\eta + 2\varepsilon.$$

By the TL-Q guarantee, it produces one of these sound behaviors with probability at least $9/10$. By setting the constant C_3 large enough, by Hoeffding's inequality if \mathcal{A} outputs a hypothesis $h \neq \perp$, it satisfies $\widehat{\text{err}}(h) \leq \text{dist}_{\mathcal{D}}(h, f) + \varepsilon$ with probability at least

$$1 - \exp(-2\varepsilon^2|T|) \geq 99/100.$$

When this happens, we have $\widehat{\text{err}}(h) \leq c\eta + 3\varepsilon$, so we successfully output structure.

91:10 Limitations of Membership Queries in Testable Learning

- **Noise:** In this case, the labels are drawn from $\text{Bernoulli}(p)$, and the elements are drawn from \mathcal{D} . Since $y(x) = p$ for all x , we have

$$\mathcal{D}(x) = \frac{\mathcal{D}_x(x) \cdot p(1-p)}{Z} \quad \text{and} \quad Z = \sum_{z \in \mathcal{X}} \mathcal{D}_x(x) \cdot p(1-p).$$

Thus, $\mathcal{D} = \mathcal{D}_x$, so by completeness of \mathcal{A} , it must output \perp with probability at most $1/10$. With the remaining probability, \mathcal{A} outputs a hypothesis h ; we will argue that with high probability its error on S_{test} is at least $\min(p, 1-p) - \varepsilon > c\eta + 3\varepsilon$. We return an error if any element in the test set appears anywhere else, so assuming this does not happen, each label in the test set is a new independent draw from $\text{Bernoulli}(p)$. Thus we have:

$$\begin{aligned} \Pr_{x \sim T} [h(x) \neq f(x)] &= \frac{1}{|T|} \sum_{x \in T: h(x)=0} \text{Bernoulli}(p) + \frac{1}{|T|} \sum_{x \in T: h(x)=1} \text{Bernoulli}(1-p) \\ &\geq \frac{1}{|T|} \text{Binomial}(\min(p, 1-p), |T|). \end{aligned}$$

By a Hoeffding bound it follows that $\widehat{\text{err}}(h) \geq \min(p, 1-p) - \varepsilon$ with probability at least

$$1 - 2 \exp(-2\varepsilon^2|T|) \geq 99/100.$$

Thus, we successfully output noise with high probability.

In both cases the refutation algorithm succeeds with probability $\geq 4/5$ conditioned on not returning an error due to either insufficient samples, duplicate samples, or overlap between the query set and the test set.

We will set the sample complexity so that the probability of insufficient samples is small. Let B be the number of samples reserved for drawing from $\text{Bernoulli}(p)$ or estimating \hat{p} . Each of the remaining $m' - B$ samples is included in S with probability $p(1-p)$. Thus we will set $m' - B \geq \frac{100}{(c\eta+4\varepsilon)(1-c\eta-4\varepsilon)} \cdot (m + C_3/\varepsilon^2)$. Then we have by a Chernoff bound,

$$\Pr \left[|S| < m + \frac{C_3}{\varepsilon^2} \right] \leq \exp \left(-\frac{(99/100)^2}{2} \cdot p(1-p)(m' - B) \right) \leq 1/100.$$

To set B , we need 2 draws for each of the samples and 1 draw for each membership query; by setting C_2 large enough this requirement is satisfied. Thus the total sample complexity is

$$m' := O \left(\frac{m + 1/\varepsilon^2}{\varepsilon} + q \right)$$

and the total time complexity is $O(m' + t)$.

Finally, we will bound the probability of duplicate samples and overlap with the query set. By the assumption that $m + 1/\varepsilon^2 \ll 1/\|\mathcal{D}_x\|_2$ and the fact that each pair of samples collides with probability $\|\mathcal{D}_x\|_2^2$, it follows from a union bound over the pairs of samples in S that w.h.p. there are no duplicates in $S_{\text{train}} \cup S_{\text{test}}$. Furthermore, by the assumption that $q/\varepsilon^2 \ll 1/\|\mathcal{D}_x\|_2 \leq 1/\|\mathcal{D}_x\|_\infty$, it follows that every set of size q has distributional mass $\ll \varepsilon^2$. Thus, with high probability none of the C_3/ε^2 elements in the test set appear in the query set.

Union bounding over all the failure probabilities in each case, the total success probability remains at least $2/3$. \blacktriangleleft

3.2 TL-Q Implies Sample-Based Learnability

A corollary of Theorem 13 is the fact that TL-Q implies efficient sample-based, distribution-specific agnostic learning.

► **Corollary 17** (Testable learning with queries implies learning with samples). *Let \mathcal{C} be $(c, \varepsilon, \frac{1}{10})$ PAC-testably-learnable with m samples, q queries, and t time, on a distribution family \mathcal{F} satisfying*

$$m + q/\varepsilon^2 \ll \frac{1}{\sup_{\mathcal{D}_x \in \mathcal{F}} (\|\mathcal{D}_x\|_2)}.$$

Then for any ε satisfying $\varepsilon^2 \geq ck \cdot \sup_{\mathcal{D}_x \in \mathcal{F}} (\|\mathcal{D}_x\|_2)$ for sufficiently large constant k , there is an agnostic learner for \mathcal{C} over \mathcal{F} with excess error $1 - 1/c + O(\varepsilon)$. In particular, when $c = 1$, i.e. \mathcal{C} is fully agnostically learnable in TL-Q, \mathcal{C} is fully agnostically learnable with samples.

The sample complexity of the learner is

$$O((m')^3/\varepsilon^2) \quad \text{where} \quad m' = O\left(\frac{m + 1/\varepsilon^2}{\varepsilon} + q\right)$$

and the time complexity is

$$O\left(\frac{(m')^2(m' + t)}{\varepsilon^2}\right).$$

Proof. By Theorem 13, there is a $(c\eta + 4\varepsilon, \eta)$ -refutation algorithm for any $\eta < \frac{1/2-4\varepsilon}{c}$. Observe that biased (α, η) -refutation is at least as strong as η -refutation: the (α, η) -refutation algorithm can be used to solve η -refutation, as $p = 1/2$ is always in the range $[\alpha, 1 - \alpha]$. By Lemma 11, this gives an agnostic learner with excess error $1 - 2 \cdot \frac{1/2-4\varepsilon}{c} + \varepsilon = 1 - 1/c + O(\varepsilon)$. The time and sample bounds are obtained by combining the bounds in Lemma 11 with those in Theorem 13. ◀

3.3 Realizably Learning Juntas via Exact Refutation

In the above subsections, we gave a general reduction from TL-Q to agnostic learning, citing as a black box the learning-by-refutation lemma of [17], Lemma 11. This lemma yields learners whose excess error depends on the refutation gap parameter η , which in our case must necessarily be smaller than $1/2c$, as one cannot hope to distinguish a function that is $(1/2c)$ -close to the concept class from a random function using a c -semi-agnostic learner. Thus the performance of this learner quickly degrades with c , becoming trivial when $c = 2$.

For the class of sparse juntas over the uniform distribution on $\{0, 1\}^n$, we give a realizable learner from an algorithm that solves the easier task of *exact* refutation ($\eta = 0$). Exact refutation reduces to c -semi-agnostic TL-Q for any value of c , via Theorem 13. Thus we show that for juntas, even for large values of c , semi-agnostic TL-Q is as hard as learning with samples. We state the result below; the proof is deferred to the full version of this paper.

► **Lemma 18.** *Let $\varepsilon \gg 2^{-n/4}$ and $m + q/\varepsilon^2 \ll 2^{n/2}$. For any constant c , if the class of k -juntas is $(c, \varepsilon, \frac{1}{10})$ -PAC-testably-learnable with q queries, m samples, and t time over the uniform distribution on $\{0, 1\}^n$, then the class of k -juntas is agnostically learnable over the uniform distribution with excess error $O(\varepsilon)$ and confidence $1 - \delta$, with sample complexity*

$$m' = \left(\frac{m + 1/\varepsilon^2}{\varepsilon} + q\right) \cdot 2^{O(k)} \cdot n \log^2(n/\delta)$$

and time complexity

$$t' = \left(\frac{m + 1/\varepsilon^2}{\varepsilon} + q + t \right) \cdot 2^{O(k)} \cdot n \log(n/\delta).$$

Since k -sparse parities are a subclass of k -juntas, we conclude that even semi-agnostic TL-Q cannot be done in $n^{o(k)}$ time, under the assumption that LSPN is hard.

► **Corollary 19.** *If LSPN requires $n^{\Omega(k)}$ time, then for any constant c , c -semi-agnostic TL-Q for k -juntas over the uniform distribution requires $n^{\Omega(k)}$ time.*

4 MQ-SQ Lower Bounds

We introduce a class of “MQ-SQ” (*membership-query-statistical-query*) algorithms that capture many existing learning algorithms that use membership queries. Then, we prove an SQ-analogue of the reduction in Section 3.1: an MQ-SQ testable learner for a class \mathcal{C} implies an SQ algorithm for refutation (Definition 9). This result, together with a reduction from SQ weak learning to SQ refutation, allows us to prove lower bounds against MQ-SQ algorithms for testably learning several fundamental concept classes, including parity functions, k -juntas, and decision trees.

4.1 Five Types of MQ-SQs

Let \mathcal{X} denote the instance space and $f : \mathcal{X} \rightarrow \{0, 1\}$ denote the target function in the TL-Q instance. Let $\mathcal{D} \in \Delta(\mathcal{X})$ be the unknown marginal distribution over \mathcal{X} . An MQ-SQ oracle for (f, \mathcal{D}) answers the following five types of queries up to a small error.

► **Definition 20 (MQ-SQ Oracle).** *An MQ-SQ oracle with tolerance $\tau \geq 0$ answers the following five types of queries within an additive error of τ , given any test function $\phi : \mathcal{X} \rightarrow [0, 1]$, any distribution $\mathcal{D}^* \in \Delta(\mathcal{X})$, and any permutation $\pi : \mathcal{X} \rightarrow \mathcal{X}$ without fixed points:*

- *Type I:* $\mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)f(x)]$.
- *Type II:* $\mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)f(x)f(\pi(x))]$.
- *Type III:* $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)]$.
- *Type IV:* $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)]$.
- *Type V:* $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)f(\pi(x))]$.

For concreteness, consider the case that $\mathcal{X} = \{0, 1\}^n$ is the hypercube. By setting \mathcal{D}^* to the uniform distribution over $\{0, 1\}^n$ (denoted by \mathcal{U}) in Type I queries, we recover the usual statistical query model for learning over \mathcal{U} . Queries of Types II and V allow us to estimate the correlation between f and a permuted version of f (weighted by ϕ) over both a generic customized distribution \mathcal{D}^* and the unknown marginal \mathcal{D} , respectively. For instance, setting $\pi : x \mapsto x^{\oplus i}$ and $\mathcal{D}^* = \mathcal{U}$ allows us to estimate the influence of variable x_i with respect to f .

When the distribution \mathcal{D}^* in a Type I MQ-SQ is degenerate (i.e., with all its probability mass on a single point $x_0 \in \mathcal{X}$), the MQ-SQ reduces to a usual membership query at x_0 . Our reduction for MQ-SQ algorithms only works when no such queries are made. More concretely, the reduction requires that the squared 2-norm of \mathcal{D}^* , $\|\mathcal{D}^*\|_2^2 := \sum_{x \in \mathcal{X}} [\mathcal{D}^*(x)]^2$, is *sufficiently small* in all queries that the algorithm makes. As we will see later, when many existing query-based learning algorithms are implemented using MQ-SQs, \mathcal{D}^* is usually uniform over a size- $2^{\Omega(n)}$ subset of $\{0, 1\}^n$. This implies $\|\mathcal{D}^*\|_2^2 \leq 2^{-\Omega(n)}$, so our reduction applies to most of the interesting parameter regimes.

4.2 Implementing Query-Based Learning Algorithms Using MQ-SQs

We note that many existing MQ-based learning algorithms (or components thereof) for the uniform distribution \mathcal{U} over the hypercube $\mathcal{X} = \{0, 1\}^n$ can be implemented using MQ-SQs. Examples of well-known MQ algorithms and their MQ-SQ implementations are given in the full version of this paper.

4.3 MQ-SQ Testable Learning Implies SQ Refutation

We will show that, if there is an efficient MQ-SQ algorithm that testably learns class \mathcal{C} , the same class can be refuted by an efficient SQ algorithm. To this end, we first recall the definition of a (usual) SQ-based algorithm in the context of refutation (Definition 9). As in Section 3.1, we let $\mathcal{D}^{\text{refut.}}$ denote the distribution of labeled pairs in the refutation instance and \mathcal{D} denote the unknown marginal distribution in a TL-Q instance.

► **Definition 21** (SQ oracle for refutation). *An SQ oracle for refutation with tolerance $\tau \geq 0$ answers queries of form $\mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x,y)]$ within an additive error of τ given a test function $\phi : \mathcal{X} \times \{0, 1\} \rightarrow [0, 1]$.*

Recap: Reduce refutation to TL-Q

We start by recalling the reduction from Section 3.1. Let $p := \Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y = 1]$ be the fraction of positive labels in the refutation instance. We consider the TL-Q instance on the same concept class \mathcal{C} , where the target function $f : \mathcal{X} \rightarrow \{0, 1\}$ is chosen as a *random p -biased function*, i.e., each function value $f(x)$ is sampled from $\text{Bernoulli}(p)$ independently. The marginal distribution $\mathcal{D} \in \Delta(\mathcal{X})$ is the distribution of the output $x \in \mathcal{X}$ produced by the following procedure:

- Sample $(x, y) \sim \mathcal{D}^{\text{refut.}}$.
- If $y = f(x) = 1$, output x with probability $1 - p$.
- If $y = f(x) = 0$, output x with probability p .
- If no output is produced, return to the first step.

Formally, the probability mass function of \mathcal{D} is given by

$$\mathcal{D}(x) = \frac{1}{Z} \cdot \mathcal{D}_x(x) [(1 - p) \cdot y(x) \cdot f(x) + p \cdot (1 - y(x)) \cdot (1 - f(x))], \quad (1)$$

where $\mathcal{D}_x \in \Delta(\mathcal{X})$ is the \mathcal{X} -marginal of $\mathcal{D}^{\text{refut.}}$,

$$y(x) := \mathbb{E}_{(x', y') \sim \mathcal{D}^{\text{refut.}}} [y' \mid x' = x]$$

is the conditional expectation of $y \mid x$ over $\mathcal{D}^{\text{refut.}}$, and

$$Z := \sum_{x \in \mathcal{X}} \mathcal{D}_x(x) [(1 - p) \cdot y(x) \cdot f(x) + p \cdot (1 - y(x)) \cdot (1 - f(x))] \quad (2)$$

is a normalization factor.

Simulation of oracle queries

We state the following lemmas, whose proofs are deferred to the full version of this paper.

► **Lemma 22** (Types I and II). *The following holds for every $\varepsilon \geq 0$, $\phi : \mathcal{X} \rightarrow [0, 1]$, $\mathcal{D}^* \in \Delta(\mathcal{X})$ and permutation $\pi : \mathcal{X} \rightarrow \mathcal{X}$ without fixed points:*

91:14 Limitations of Membership Queries in Testable Learning

- With probability at least $1 - 2e^{-\Omega(\varepsilon^2/\|\mathcal{D}^*\|_2^2)}$ over the randomness of f ,

$$\left| \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)f(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)] \right| \leq \varepsilon.$$

- With probability at least $1 - 6e^{-\Omega(\varepsilon^2/\|\mathcal{D}^*\|_2^2)}$ over the randomness of f ,

$$\left| \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)f(x)f(\pi(x))] - p^2 \cdot \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)] \right| \leq \varepsilon.$$

► **Lemma 23** (Types III, IV and V). *The following holds for every $\varepsilon \geq 0$, $\phi : \mathcal{X} \rightarrow [0, 1]$ and permutation $\pi : \mathcal{X} \rightarrow \mathcal{X}$ without fixed points:*

- With probability at least $1 - 4e^{-\Omega(\varepsilon^2 \cdot p^2(1-p)^2/\|\mathcal{D}_x\|_2^2)}$ over the randomness of f ,

$$\left| \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] - \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x)] \right| \leq \varepsilon.$$

- With probability at least $1 - 4e^{-\Omega(\varepsilon^2 \cdot p^2(1-p)^2/\|\mathcal{D}_x\|_2^2)}$ over the randomness of f ,

$$\left| \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)] - \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x) \cdot y] \right| \leq \varepsilon.$$

- With probability at least $1 - 8e^{-\Omega(\varepsilon^2 \cdot p^2(1-p)^2/\|\mathcal{D}_x\|_2^2)}$ over the randomness of f ,

$$\left| \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)f(\pi(x))] - p \cdot \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x) \cdot y] \right| \leq \varepsilon.$$

Now, we put everything together and prove our main result on MQ-SQ algorithms for testable learning.

► **Proposition 24.** *Let \mathcal{C} be a concept class of boolean functions over instance space \mathcal{X} . Suppose that there is a (c, ε, δ) -PAC MQ-SQ algorithm that testably learns \mathcal{C} over distribution family $\mathcal{F} \subseteq \Delta(\mathcal{X})$ using at most q queries to an MQ-SQ oracle with tolerance $\tau > 0$. Then, there is an algorithm that solves biased- (α, η) -refutation on \mathcal{C} over the same distribution family \mathcal{F} by making at most $q' = q + O(1)$ queries to an SQ oracle with tolerance $\tau' = \tau/4$ and has a failure probability of at most $\delta' = \delta + O(q) \cdot e^{-\Omega(\tau^2 B)}$, assuming the following:*

1. $\alpha > c\eta + \varepsilon + (c + 4)\tau + 6\tau'$;
2. $B \leq p^2(1-p)^2/\|\mathcal{D}_x\|_2^2$ for every $\mathcal{D}_x \in \mathcal{F}$, where $p = \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y]$ is the average label in the refutation instance;
3. $B \leq 1/\|\mathcal{D}^*\|_2^2$ holds for all MQ-SQs of Types I and II that \mathcal{A} makes.

Proof. Let \mathcal{A} denote the hypothetical MQ-SQ algorithm that testably learns \mathcal{C} . We construct a new algorithm, denoted by \mathcal{A}' , that refutes \mathcal{C} using an SQ oracle by simulating the execution of \mathcal{A} .

Recall that we defined $p = \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y]$. As the first step of the algorithm, \mathcal{A}' queries the SQ oracle (Definition 21) with $\phi(x, y) = y$ to obtain an estimate $\hat{p} \in [p - \tau', p + \tau']$ for p .

Handling queries. Whenever the simulated copy of \mathcal{A} makes an MQ-SQ, \mathcal{A}' answers the query as follows:

- When \mathcal{A} makes a Type I query on $\mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)f(x)]$, \mathcal{A}' returns $\hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)]$.
- When \mathcal{A} makes a Type II query on $\mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)f(x)f(\pi(x))]$, \mathcal{A}' returns $\hat{p}^2 \cdot \mathbb{E}_{x \sim \mathcal{D}^*} [\phi(x)]$.
- When \mathcal{A} makes a Type III query on $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)]$, \mathcal{A}' queries the SQ oracle on the value of $\mu = \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x)]$ and then forwards the answer $\hat{\mu} \in [\mu - \tau', \mu + \tau']$ to \mathcal{A} .
- When \mathcal{A} makes a Type IV query on $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)]$, \mathcal{A}' queries the SQ oracle on the value of $\mu = \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x) \cdot y]$ and then forwards the answer $\hat{\mu} \in [\mu - \tau', \mu + \tau']$ to \mathcal{A} .

- When \mathcal{A} makes a Type V query on $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f(x)f(\pi(x))]$, \mathcal{A}' queries the SQ oracle on the value of $\mu = \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\phi(x) \cdot y]$ and then forwards the answer $\hat{\mu} \in [\mu - \tau', \mu + \tau']$ multiplied by \hat{p} to \mathcal{A} .

In the first two cases, \mathcal{A}' can exactly compute the expectations since it has full knowledge of $\phi : \mathcal{X} \rightarrow [0, 1]$ and $\mathcal{D}^* \in \Delta(\mathcal{X})$.

Decision rule. When \mathcal{A} terminates, \mathcal{A}' decides on the refutation instance as follows:

- If \mathcal{A} rejects the TL-Q instance, \mathcal{A}' returns **structure**, indicating that some $f^* \in \mathcal{C}$ has an error $\leq \eta$ on $\mathcal{D}^{\text{refut.}}$.
- If \mathcal{A} accepts and returns a function $\hat{f} : \mathcal{X} \rightarrow \{0, 1\}$, \mathcal{A}' makes the following three additional MQ-SQs on behalf of \mathcal{A} and answer them as described above:

$$\mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)], \quad \mathbb{E}_{x \sim \mathcal{D}} [f(x)], \quad \text{and} \quad \mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)f(x)].$$

Let $\hat{\mu}_1, \hat{\mu}_2, \hat{\mu}_3$ denote the answers to the three queries, and let

$$\hat{\mu} = \hat{\mu}_1 + \hat{\mu}_2 - 2\hat{\mu}_3$$

be a weighted sum of them. Note that $\hat{\mu}$ is intended to be an estimate of

$$\mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x) + f(x) - 2\hat{f}(x)f(x)] = \Pr_{x \sim \mathcal{D}} [\hat{f}(x) \neq f(x)].$$

Finally, \mathcal{A}' outputs **noise** (indicating that the labels are random) if $\hat{\mu} \geq \min\{\hat{p}, 1 - \hat{p}\} - 5\tau'$, and outputs **structure** otherwise.

Overview of analysis. We first upper bound the number of SQs that \mathcal{A}' makes. By construction, \mathcal{A}' queries the SQ oracle at most once for every MQ-SQ made by \mathcal{A} . In addition, \mathcal{A}' makes one query at the beginning and at most three queries at the end. Thus, \mathcal{A}' makes at most $q' = q + O(1)$ queries in total.

To analyze the correctness of \mathcal{A}' , let $f : \mathcal{X} \rightarrow \{0, 1\}$ be a random p -biased function obtained by independently drawing the function value $f(x)$ from $\text{Bernoulli}(p)$ for each $x \in \mathcal{X}$. Note that f is only for the analysis; it is never used in algorithm \mathcal{A}' . Also, let \mathcal{D} denote the distribution over \mathcal{X} induced by $\mathcal{D}^{\text{refut.}}$ and f (see Equation (1)). We will first argue that, with high probability, the simulated copy of \mathcal{A} effectively runs on an instance of testable learning with target function f and marginal distribution \mathcal{D} . We will then show that the decision made by \mathcal{A}' is correct due to the intended behavior of \mathcal{A} on such an instance.

A good event. Let $\mathcal{E}^{\text{good}}$ be the “good event” that the three conditions below hold simultaneously:

- The simulated execution of \mathcal{A} coincides with its execution on the testable learning instance (f, \mathcal{D}) using an MQ-SQ oracle with tolerance τ . In other words, every MQ-SQ made by \mathcal{A} is answered up to an additive error of τ .
- If the first condition holds, the output of \mathcal{A} is valid with respect to the testable learning instance.
- If there exists $f^* \in \mathcal{C}$ that satisfies $\Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [f^*(x) \neq y] \leq \eta$ (i.e., we are in the **structure** case), it holds that $\Pr_{x \sim \mathcal{D}} [f^*(x) \neq f(x)] \leq \eta + \tau$.

By Lemmas 22 and 23, for each MQ-SQ made by \mathcal{A} , the first condition gets violated with probability at most $8e^{-\Omega(\tau^2 B)}$, where B is the minimum between the value of $1/\|\mathcal{D}^*\|_2^2$ (among all queries of Types I and II) and $p^2(1-p)^2/\|\mathcal{D}_x\|_2^2$. Applying the union bound to

91:16 Limitations of Membership Queries in Testable Learning

the $\leq q + 4$ queries shows that the first condition gets violated with probability at most $O(q) \cdot e^{-\Omega(\tau^2 B)}$. Since \mathcal{A} is assumed to be (c, ε, δ) -PAC, the second condition gets violated with probability at most δ . Applying Claim 16 with $\delta = \tau$ shows that the third condition gets violated with probability at most $e^{-\Omega(\tau^2 p^2 (1-p)^2 / \|\mathcal{D}_x\|_2^2)} \leq e^{-\Omega(\tau^2 B)}$. Applying the union bound again gives

$$\Pr[\mathcal{E}^{\text{good}}] \geq 1 - \delta - O(q) \cdot e^{-\Omega(\tau^2 B)}.$$

In the rest of the proof, we show that event $\mathcal{E}^{\text{good}}$ implies that \mathcal{A}' decides correctly.

Proof of completeness. Suppose that some $f^* \in \mathcal{C}$ satisfies $\Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [f^*(x) \neq y] \leq \eta$, where η is the parameter of the refutation instance (Definition 9). The third condition of the good event $\mathcal{E}^{\text{good}}$ implies that the same f^* has with an error $\leq 2\eta$ over \mathcal{D} . Then, the testable learner \mathcal{A} may output either \perp or a function $\hat{f} : \mathcal{X} \rightarrow \{0, 1\}$ that satisfies

$$\Pr_{x \sim \mathcal{D}} [\hat{f}(x) \neq f(x)] \leq c \cdot \Pr_{x \sim \mathcal{D}} [f^*(x) \neq f(x)] + \varepsilon \leq c(\eta + \tau) + \varepsilon.$$

In the former case, \mathcal{A}' would correctly output **structure**. For the latter case, applying the identity $\mathbb{1}[b_1 \neq b_2] = b_1 + b_2 - 2b_1b_2$ for $b_1, b_2 \in \{0, 1\}$ gives

$$\Pr_{x \sim \mathcal{D}} [\hat{f}(x) \neq f(x)] = \mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)] + \mathbb{E}_{x \sim \mathcal{D}} [f(x)] - 2 \mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)f(x)].$$

Assuming event $\mathcal{E}^{\text{good}}$, \mathcal{A}' obtains an estimate of each of the three expectations on the right-hand side above up to an additive error of τ . It then follows that the value of $\hat{\mu}$ computed at the end satisfies

$$\hat{\mu} \leq \Pr_{x \sim \mathcal{D}} [\hat{f}(x) \neq f(x)] + 4\tau \leq c\eta + \varepsilon + (c + 4)\tau.$$

Since $\min\{p, 1 - p\} \geq \alpha > c\eta + \varepsilon + (c + 4)\tau + 6\tau'$, we have

$$\hat{\mu} < \min\{p, 1 - p\} - 6\tau' \leq \min\{\hat{p}, 1 - \hat{p}\} - 5\tau'$$

in this case, and \mathcal{A}' would correctly output **structure**.

Proof of soundness. Suppose that the distribution $\mathcal{D}^{\text{refut.}}$ in the refutation instance is the product distribution of some $\mathcal{D}_x \in \mathcal{F}$ and $\text{Bernoulli}(p)$. Then, the resulting marginal distribution \mathcal{D} in the testable learning instance is exactly $\mathcal{D}_x \in \mathcal{F}$. Thus, assuming that \mathcal{A} is correct, \mathcal{A} would accept and output a function $\hat{f} : \mathcal{X} \rightarrow \{0, 1\}$. Then, at the end of \mathcal{A}' , we compute

$$\hat{\mu} \approx \mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)] + \mathbb{E}_{x \sim \mathcal{D}} [f(x)] - 2 \mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)f(x)].$$

By the way in which \mathcal{A}' handles the MQ-SQs, the three terms

$$\mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)], \mathbb{E}_{x \sim \mathcal{D}} [f(x)], \text{ and } \mathbb{E}_{x \sim \mathcal{D}} [\hat{f}(x)f(x)]$$

are approximated with

$$\mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\hat{f}(x)], \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [y], \text{ and } \mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} [\hat{f}(x) \cdot y],$$

respectively. All the three values are obtained from querying the SQ oracle. Since the SQ oracle has a tolerance of τ' , the value of $\widehat{\mu}$ is within an additive error of $4\tau'$ to

$$\mathbb{E}_{(x,y) \sim \mathcal{D}^{\text{refut.}}} \left[\widehat{f}(x) + y - 2\widehat{f}(x) \cdot y \right] = \Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} \left[\widehat{f}(x) \neq y \right],$$

which is exactly the error of \widehat{f} on distribution $\mathcal{D}^{\text{refut.}}$. Since $\mathcal{D}^{\text{refut.}}$ is the product of \mathcal{D}_x and $\text{Bernoulli}(p)$, regardless of the choice of \widehat{f} , $\Pr_{(x,y) \sim \mathcal{D}^{\text{refut.}}} \left[\widehat{f}(x) \neq y \right]$ is at least $\min\{p, 1-p\}$. Together with the fact that $|p - \widehat{p}| \leq \tau'$, this further implies

$$\widehat{\mu} \geq \min\{p, 1-p\} - 4\tau' \geq \min\{\widehat{p}, 1-\widehat{p}\} - 5\tau'.$$

Thus, \mathcal{A}' would correctly output noise. ◀

4.4 SQ Refutation Implies SQ Weak Learning

We show that an SQ algorithm for refutation implies an SQ algorithm for weakly learning the same concept class. Later, using the equivalence between SQ dimension and weak learning [2], we can lift SQ-dimension lower bounds to lower bounds against SQ-based refutation. By Proposition 24, this further leads to lower bounds against MQ-SQ algorithms for testable learning with queries.

We first recall the definition of SQ-based weak learning.

► **Definition 25** (SQ Weak learning). *Let $\mathcal{C} \subseteq \{f : \mathcal{X} \rightarrow \{0,1\}\}$ be a concept class over a finite instance space \mathcal{X} . Let \mathcal{D} be a given distribution over \mathcal{X} and $f^* \in \mathcal{C}$ be an unknown target function. An SQ oracle for (f^*, \mathcal{D}) with tolerance $\tau \geq 0$ answers queries of form $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x) \cdot f^*(x)]$ up to an additive error of τ , where $\phi : \mathcal{X} \rightarrow [0,1]$ is any given test function. An SQ algorithm ε -weakly learns \mathcal{C} if it, by making queries to an SQ oracle, with probability $\geq 2/3$ outputs a classifier $\widehat{f} : \mathcal{X} \rightarrow \{0,1\}$ that satisfies*

$$\Pr_{x \sim \mathcal{D}} \left[\widehat{f}(x) \neq f^*(x) \right] \leq \frac{1}{2} - \varepsilon.$$

Intuitively, to solve refutation (Definition 9) using an SQ algorithm, in the “structure” case that some $f^* \in \mathcal{C}$ has a low error, the algorithm must query the SQ oracle using a test function that has a non-trivial correlation with f^* . Such a test function would then allow us to learn the unknown function up to an error that is better than random guessing.

► **Proposition 26.** *Suppose that an SQ algorithm solves biased- (α, η) -refutation for concept class \mathcal{C} on distribution \mathcal{D} by making at most q queries with tolerance $\tau \geq 0$. Then, assuming that $\alpha \leq 1/2 - \Omega(\tau)$, there is an SQ algorithm that $\Omega(\tau)$ -weakly learns \mathcal{C} on \mathcal{D} by making at most $q' = O(q + 1/\tau)$ queries to an SQ oracle with tolerance $\tau' = \Omega(\tau)$.*

To prove the proposition, we will use the following simple fact: If a $[0,1]$ -valued function has a non-trivial correlation with a sufficiently balanced boolean function, it can be rounded into a random binary classifier with a non-trivial accuracy in expectation. (If the boolean function is far from balanced, it can be easily learned by a constant function.)

► **Lemma 27.** *The following holds for every $\delta \geq 0$, distribution \mathcal{D} over \mathcal{X} , and binary function $f^* : \mathcal{X} \rightarrow \{0,1\}$ with mean $p := \mathbb{E}_{x \sim \mathcal{D}} [f^*(x)] \in [1/2 - \gamma, 1/2 + \gamma]$: Suppose that function $\phi : \mathcal{X} \rightarrow [0,1]$ satisfies $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] \geq \delta$. Then, for the random*

91:18 Limitations of Membership Queries in Testable Learning

function $\tilde{\phi} : \mathcal{X} \rightarrow \{0, 1\}$ obtained from sampling each $\tilde{\phi}(x)$ from $\text{Bernoulli}(\phi(x))$ independently, we have

$$\mathbb{E}_{\tilde{\phi}} \left[\Pr_{x \sim \mathcal{D}} \left[\tilde{\phi}(x) \neq f^*(x) \right] \right] \leq \frac{1}{2} - (2\delta - 3\gamma).$$

Similarly, if $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] \leq -\delta$, we have

$$\mathbb{E}_{\tilde{\phi}} \left[\Pr_{x \sim \mathcal{D}} \left[1 - \tilde{\phi}(x) \neq f^*(x) \right] \right] \leq \frac{1}{2} - (2\delta - 3\gamma).$$

Proof. It suffices to prove the first part; the second part follows by symmetry. By the identity $\mathbb{1}[b_1 \neq b_2] = b_1 + b_2 - 2b_1b_2$ for $b_1, b_2 \in \{0, 1\}$, it holds for every possible realization of $\tilde{\phi} : \mathcal{X} \rightarrow \{0, 1\}$ that

$$\Pr_{x \sim \mathcal{D}} \left[\tilde{\phi}(x) \neq f^*(x) \right] = \mathbb{E}_{x \sim \mathcal{D}} \left[\tilde{\phi}(x) + f^*(x) - 2\tilde{\phi}(x)f^*(x) \right].$$

Taking an expectation over the randomness of $\tilde{\phi}$ shows that

$$\begin{aligned} \mathbb{E}_{\tilde{\phi}} \left[\Pr_{x \sim \mathcal{D}} \left[\tilde{\phi}(x) \neq f^*(x) \right] \right] &= \mathbb{E}_{\tilde{\phi}} \left[\mathbb{E}_{x \sim \mathcal{D}} \left[\tilde{\phi}(x) + f^*(x) - 2\tilde{\phi}(x)f^*(x) \right] \right] \\ &= \mathbb{E}_{x \sim \mathcal{D}} \left[\phi(x) + f^*(x) - 2\phi(x)f^*(x) \right] \\ &= \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] + p - 2 \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f^*(x)] \\ &\leq p + \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] - 2 \left(p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] + \delta \right) \\ &= p + (1 - 2p) \cdot \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] - 2\delta, \end{aligned}$$

where the fourth step applies the assumption that $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] \geq \delta$. Since $1 - 2p \in [-2\gamma, 2\gamma]$ and $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x)] \in [0, 1]$, the second term above is at most 2γ . It follows that the expected error of $\tilde{\phi}$ is at most $p + 2\gamma - 2\delta \leq (\frac{1}{2} + \gamma) + 2\gamma - 2\delta = \frac{1}{2} - (2\delta - 3\gamma)$. \blacktriangleleft

Proof of Proposition 26. Let \mathcal{A} denote the hypothetical SQ algorithm that solves biased- (α, η) -refutation for \mathcal{C} . Let $\varepsilon, \tau' = \Theta(\tau)$ be sufficiently small such that: (1) $\alpha \leq 1/2 - (\varepsilon + 2\tau')$; (2) $\tau \geq 4\varepsilon + 22\tau'$. We construct an SQ algorithm \mathcal{A}' that weakly learns \mathcal{C} by simulating \mathcal{A} on the distribution of $(x, f^*(x))$ where $x \sim \mathcal{D}$ and $f^* \in \mathcal{C}$ is the unknown target function in the weak learning instance:

- **Step 1:** Query the SQ oracle (for the weak learning instance) to estimate the value of $p := \mathbb{E}_{x \sim \mathcal{D}} [f^*(x)]$ using the constant function $\phi(x) \equiv 1$. Let $\hat{p} \in [p - \tau', p + \tau']$ be the output of the oracle. If $\hat{p} + \tau' \leq 1/2 - \varepsilon$, output the constant function 0 and terminate. If $\hat{p} - \tau' \geq 1/2 + \varepsilon$, output the constant function 1 and terminate.
- **Step 2:** Simulate the refutation algorithm \mathcal{A} . Whenever \mathcal{A} tries to query the SQ oracle (for the refutation instance) with test function $\phi : \mathcal{X} \times \{0, 1\} \rightarrow [0, 1]$, consider the function $\Delta : \mathcal{X} \rightarrow [-1, 1]$ defined as $\Delta(x) := \phi(x, 1) - \phi(x, 0)$ and $\Delta' : \mathcal{X} \rightarrow [0, 1]$ defined as $\Delta'(x) := \frac{\Delta(x) + 1}{2}$.
- **Step 3:** Query the SQ oracle (for weak learning) with test function Δ' to estimate $\mu := \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)]$. Let $\hat{\mu} \in [\mu - \tau', \mu + \tau']$ denote the output of the SQ oracle. Check whether it holds that

$$\left| \hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| \leq \frac{\tau}{2} - 2\tau'.$$

If so, we compute $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x, 0) + p \cdot \Delta(x)]$, return the result to the refutation algorithm \mathcal{A} , and continue the simulation by going back to Step 2. Otherwise, go to Step 4.

- **Step 4:** We apply Lemma 27 to Δ' and obtain a randomized boolean function $\tilde{\phi}$ from either Δ' or $1 - \Delta'$. We query the SQ oracle to obtain an estimate $\hat{\varepsilon}$ of

$$\Pr_{x \sim \mathcal{D}} [\tilde{\phi}(x) \neq f^*(x)] = \mathbb{E}_{x \sim \mathcal{D}} [\tilde{\phi}(x)] + \mathbb{E}_{x \sim \mathcal{D}} [f^*(x)] - 2 \mathbb{E}_{x \sim \mathcal{D}} [\tilde{\phi}(x) \cdot f^*(x)].$$

If $\hat{\varepsilon} \leq 1/2 - \varepsilon - 3\tau'$, we return the function $\tilde{\phi}$. Otherwise, repeat this step.

If \mathcal{A}' outputs a constant classifier in the first step, the output clearly has an error $\leq 1/2 - \varepsilon$. Thus, we may focus on the case that $|\hat{p} - 1/2| \leq \varepsilon + \tau'$. Since $|\hat{p} - p| \leq \tau'$, we must have $|p - 1/2| \leq \gamma := \varepsilon + 2\tau'$ in this case. The rest of the proof proceeds in the following three steps:

- If Δ' has a low correlation with f^* (i.e., $|\hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)]| \leq \tau - 4\tau'$ holds in Step 3 of \mathcal{A}'), the answer $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x, 0) + p \cdot \Delta(x)]$ that we return to \mathcal{A} is a valid answer for an SQ oracle with tolerance τ .
- If Δ' has a high correlation with f^* (i.e., $|\hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)]| > \tau - 4\tau'$), we will find a good $\tilde{\phi}$ without repeating Step 4 too many times.
- If Δ' never has a high correlation with f^* , the execution of \mathcal{A} will be indistinguishable from that in the “noise” case of the refutation instance. Therefore, a high-correlation Δ' must be found with a good probability.

Low correlation gives accurate answers. Suppose that, for some test function $\phi : \mathcal{X} \times \{0, 1\} \rightarrow [0, 1]$ chosen by \mathcal{A} and the corresponding Δ' , it holds in Step 3 that

$$\left| \hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| \leq \frac{\tau}{2} - 2\tau'.$$

Recall that $\hat{\mu}$ is within an additive error of τ' to $\mu = \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)]$ and \hat{p} is within error τ' to $p = \mathbb{E}_{x \sim \mathcal{D}} [f^*(x)]$. We have

$$\left| \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| \leq \left| \hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| + 2\tau' \leq \frac{\tau}{2}.$$

Then, the difference between the correct answer,

$$\mathbb{E}_{x \sim \mathcal{D}} [\phi(x, f^*(x))] = \mathbb{E}_{x \sim \mathcal{D}} [\phi(x, 0) + (\phi(x, 1) - \phi(x, 0)) \cdot f^*(x)] = \mathbb{E}_{x \sim \mathcal{D}} [\phi(x, 0) + \Delta(x) \cdot f^*(x)],$$

and the answer $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x, 0) + p \cdot \Delta(x)]$ returned by \mathcal{A}' is exactly

$$\left| \mathbb{E}_{x \sim \mathcal{D}} [\Delta(x) \cdot f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta(x)] \right| = 2 \cdot \left| \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| \leq \tau.$$

In other words, \mathcal{A}' simulates a valid SQ oracle with tolerance τ when the correlation is low.

High correlation gives good $\tilde{\phi}$. Now, suppose that $|\hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)]| > \frac{\tau}{2} - 2\tau'$ holds in Step 3. Again, since $\hat{\mu} \approx \mu = \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)]$ and $\hat{p} \approx p = \mathbb{E}_{x \sim \mathcal{D}} [f^*(x)]$ hold up to error τ' , we have

$$\left| \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| \geq \left| \hat{\mu} - \hat{p} \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)] \right| - 2\tau' > \frac{\tau}{2} - 4\tau'.$$

By the assumption that $\tau \geq 4\varepsilon + 22\tau'$, the above implies $|\mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x) \cdot f^*(x)] - p \cdot \mathbb{E}_{x \sim \mathcal{D}} [\Delta'(x)]| \geq \delta := 2\varepsilon + 7\tau'$. Recall that we assumed $p \in [1/2 - \gamma, 1/2 + \gamma]$ for $\gamma = \varepsilon + 2\tau'$. Applying Lemma 27 to Δ' shows that Δ' can be rounded to a random function $\tilde{\phi} : \mathcal{X} \rightarrow \{0, 1\}$ with an expected error of at most

$$\frac{1}{2} - (2\delta - 3\gamma) = \frac{1}{2} - (\varepsilon + 8\tau').$$

91:20 Limitations of Membership Queries in Testable Learning

By Markov's inequality, the probability that $\tilde{\phi}$ has an error $\leq 1/2 - (\varepsilon + 6\tau')$ is at least

$$1 - \frac{1/2 - (\varepsilon + 8\tau')}{1/2 - (\varepsilon + 6\tau')} = \frac{2\tau'}{1/2 - (\varepsilon + 6\tau')} = \Omega(\tau).$$

Note that in Step 4, $\hat{\varepsilon}$ is within an additive error of $3\tau'$ to the actual error of $\tilde{\phi}$. Then, if $\tilde{\phi}$ has an error $\leq 1/2 - (\varepsilon + 6\tau')$, we would have

$$\hat{\varepsilon} \leq \Pr_{x \sim \mathcal{D}} [\tilde{\phi}(x) \neq f^*(x)] + 3\tau' \leq \frac{1}{2} - \varepsilon - 3\tau',$$

and algorithm \mathcal{A}' would terminate. Therefore, whenever Step 4 is entered, at most $O(1/\tau)$ repetitions are needed in expectation. This shows that \mathcal{A}' makes at most $O(q + 1/\tau)$ SQs in expectation.

The probability of making high-correlation queries. Now, we argue that the hypothetical refutation algorithm \mathcal{A} must make the aforementioned high-correlation query. To this end, we couple the execution of \mathcal{A} simulated by our weak learner \mathcal{A}' (the *simulated copy*) with a slight variant of it (the *imaginary copy*): In the imaginary copy, we never check the correlation or go to Step 4; we always return $\mathbb{E}_{x \sim \mathcal{D}} [\phi(x, 0) + p \cdot \Delta(x)]$ for every query $\phi : \mathcal{X} \times \{0, 1\} \rightarrow [0, 1]$ that \mathcal{A} makes.

Note that the imaginary copy of \mathcal{A} exactly runs on a refutation instance in which the distribution is the product distribution of \mathcal{D} and $\text{Bernoulli}(p)$, i.e., the label y is a p -biased coin flip regardless of x . Since we assumed that $|p - 1/2| \leq \gamma = \varepsilon + 2\tau' \leq 1/2 - \alpha$, we have $p \in [\alpha, 1 - \alpha]$. Then, by the soundness guarantee of \mathcal{A} , the imaginary copy \mathcal{A} must output noise with probability at least $2/3$.

In contrast, the simulated copy of \mathcal{A} runs on an instance in which the labels are consistent with $f^* \in \mathcal{C}$. Then, the completeness of \mathcal{A} ensures that the simulated copy outputs structure with probability $\geq 2/3$. Therefore, in the coupling between the simulated and imaginary copies, they must diverge with probability at least $1/3$. The only way for the two copies to disagree is that, during the execution of the simulated copy, the algorithm makes an SQ with a high-correlation test function. Therefore, algorithm \mathcal{A}' outputs a classifier with error $\leq 1/2 - \varepsilon$ with probability at least $1/3$. Since \mathcal{A}' never returns an incorrect answer (i.e., a classifier with error $> 1/2 - \varepsilon$), repeating \mathcal{A}' a constant number of times would boost the success probability to $2/3$, thereby giving an ε -weak learner for class \mathcal{C} on distribution \mathcal{D} . ◀

4.5 Put Everything Together

So far, we have established a reduction from SQ weak learning to SQ refutation (Proposition 26), and one from SQ refutation to MQ-SQ testable learning (Proposition 24). We now combine them and prove a lower bound for MQ-SQ testable learning in terms of the *statistical query dimension* (SQ dimension) of the concept class introduced by Blum, Furst, Jackson, Kearns, Mansour, and Rudich [2].

► **Definition 28** (SQ dimension). *The SQ dimension of a concept class $\mathcal{C} \subseteq \{f : \mathcal{X} \rightarrow \{0, 1\}\}$ on a distribution \mathcal{D} over \mathcal{X} is the maximum number d such that there exists $f_1, f_2, \dots, f_d \in \mathcal{C}$ that satisfy*

$$\Pr_{x \sim \mathcal{D}} [f_i(x) \neq f_j(x)] \in \left[\frac{1 - 1/d^3}{2}, \frac{1 + 1/d^3}{2} \right]$$

for all $i \neq j \in [d]$.

► **Theorem 29.** *The following holds for a sufficiently small constant $\varepsilon_0 > 0$: Suppose that \mathcal{C} is a concept class with SQ dimension d on distribution \mathcal{D} and $\|\mathcal{D}\|_2^2 \leq O(1/\text{poly}(d))$. Let $c \geq 1$ and $\varepsilon, \delta \leq \varepsilon_0$. Then, no MQ-SQ algorithm can (c, ε, δ) -testably learn \mathcal{C} on distribution \mathcal{D} by making $q \leq o(\text{poly}(d))$ queries to an MQ-SQ oracle with tolerance $\tau \geq \omega(1/\text{poly}(d))$ such that $\|\mathcal{D}^*\|_2^2 \leq O(1/\text{poly}(d))$ holds for all queries of types I and II.*

For concreteness, consider the hypercube $\mathcal{X} = \{0, 1\}^n$ and the uniform distribution over it. It is well-known that the family of parity functions has an SQ dimension of 2^n . Furthermore, for every $k \leq n^{1-\Omega(1)}$, both k -juntas and depth- k decision trees have SQ dimensions of $n^{\Omega(k)}$, as they both contain all parity functions of $\leq k$ variables. Thus, Theorem 29 gives $2^{\Omega(n)}$ or $n^{\Omega(k)}$ lower bounds against MQ-SQ testable learners for these classes. Regarding the constraint on \mathcal{D}^* , if \mathcal{D}^* is the uniform distribution over a d' -dimensional subcube, we need $2^{-d'} = \|\mathcal{D}^*\|_2^2 \leq O(1/\text{poly}(d))$, so ensuring $d' = \Omega(n)$ would suffice. (Recall from Section 4.2 that this condition holds when implementing many existing query-based learners as MQ-SQ algorithms.)

Proof. Suppose towards a contradiction that such an MQ-SQ algorithm exists. Ignoring all the other parameters for now, Proposition 24 shows that there is an algorithm that refutes \mathcal{C} by making $q' = q + O(1)$ queries to an SQ oracle with tolerance $\tau' = \tau/4$. Applying Proposition 26 then gives an algorithm that $\Omega(\tau)$ -weakly learns parity functions using $O(q' + 1/\tau') = O(q + 1/\tau)$ queries to an SQ oracle with tolerance $\Theta(\tau)$. By [2, Theorem 12], to $\Omega(1/d^3)$ -weakly learn concept class \mathcal{C} using an SQ oracle with tolerance $\Omega(d^{-1/3})$, at least $\Omega(d^{1/3})$ queries are needed. Since $q, 1/\tau = o(\text{poly}(d))$, we obtain a contradiction.

Now, we set the parameters in Propositions 24 and 26 carefully. We may assume that $\tau \leq \varepsilon_0/c$ without loss of generality; a smaller tolerance makes the SQ oracle (and thus the lower bound result) stronger. Since $\varepsilon, \delta, c\tau \leq \varepsilon_0$ are sufficiently small and $\tau' = \tau/4$, we can choose $\alpha = 0.1$ and $\eta = 0$ in Proposition 24 such that the first condition $\alpha > c\eta + \varepsilon + (c+4)\tau + 6\tau'$ is satisfied. Furthermore, the condition that $\alpha \leq 1/2 - \Omega(\tau)$ in Proposition 26 would also hold. Recall that the failure probability increases from $\delta \leq \varepsilon_0$ to $\delta + O(q) \cdot e^{-\Omega(\tau^2 B)}$ in Proposition 24. Setting $B = \Theta((\log q)/\tau^2) = o(\text{poly}(d))$ suffices to control the new failure probability by $2\varepsilon_0$.

It remains to check the second and the third conditions of Proposition 24. For the third, we need the MQ-SQ testable learner to restrict the distribution \mathcal{D}^* in its queries such that $\|\mathcal{D}^*\|_2^2 \leq 1/B$. This is ensured by $\|\mathcal{D}^*\|_2^2 \leq O(1/\text{poly}(d))$ and $B \leq o(\text{poly}(d))$. Finally, to check the second condition that $B \leq p^2(1-p)^2/\|\mathcal{D}_x\|_2^2$, we note that $\|\mathcal{D}_x\|_2^2 = \|\mathcal{D}\|_2^2 \leq O(1/\text{poly}(d))$. Furthermore, by the way in which the reduction works in the proof of Proposition 26, whenever the refutation algorithm is called, the labels are nearly balanced, i.e., $p^2(1-p)^2 = \Omega(1)$. Therefore, the second condition is always satisfied by our choice of $B = o(\text{poly}(d))$. This completes the proof. ◀

References

- 1 Guy Blanc, Jane Lange, Mingda Qiao, and Li-Yang Tan. Properly learning decision trees in almost polynomial time. *Journal of the ACM*, 69(6):1–19, 2022. doi:10.1145/3561047.
- 2 Avrim Blum, Merrick Furst, Jeffrey Jackson, Michael Kearns, Yishay Mansour, and Steven Rudich. Weakly learning dnf and characterizing statistical query learning using fourier analysis. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 253–262, 1994. doi:10.1145/195058.195147.
- 3 Avrim L Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2):245–271, 1997. doi:10.1016/S0004-3702(97)00063-5.

- 4 Nader H Bshouty and Vitaly Feldman. On using extended statistical queries to avoid membership queries. *Journal of Machine Learning Research*, 2(Feb):359–395, 2002. URL: <https://jmlr.org/papers/v2/bshouty02a.html>.
- 5 Amit Daniely. Complexity theoretic limitations on learning halfspaces. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 105–117, 2016. doi:10.1145/2897518.2897520.
- 6 Amit Daniely, Nati Linial, and Shai Shalev-Shwartz. From average case complexity to improper learning complexity. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 441–448, 2014. doi:10.1145/2591796.2591820.
- 7 Amit Daniely and Shai Shalev-Shwartz. Complexity theoretic limitations on learning dnf’s. In *Conference on Learning Theory*, pages 815–830. PMLR, 2016. URL: <http://proceedings.mlr.press/v49/daniely16.html>.
- 8 Ilias Diakonikolas, Daniel Kane, Vasilis Kontonis, Sihan Liu, and Nikos Zarifis. Efficient testable learning of halfspaces with adversarial label noise. *Advances in Neural Information Processing Systems*, 36:39470–39490, 2023.
- 9 Ariel Elbaz, Homin K Lee, Rocco A Servedio, and Andrew Wan. Separating models of learning from correlated and uncorrelated data. *The Journal of Machine Learning Research*, 8:277–290, 2007. URL: <https://jmlr.org/papers/v8/elbaz07a.html>.
- 10 Vitaly Feldman. On the power of membership queries in agnostic learning. *The Journal of Machine Learning Research*, 10:163–182, 2009. doi:10.5555/1577069.1577076.
- 11 Vitaly Feldman and Shrenik Shah. Separating models of learning with faulty teachers. *Theoretical computer science*, 410(19):1903–1912, 2009. doi:10.1016/J.TCS.2009.01.017.
- 12 Aravind Gollakota, Adam Klivans, Konstantinos Stavropoulos, and Arsen Vasilyan. Tester-learners for halfspaces: Universal algorithms. *Advances in Neural Information Processing Systems*, 36:10145–10169, 2023.
- 13 Aravind Gollakota, Adam R. Klivans, and Pravesh K. Kothari. A moment-matching approach to testable learning and a new characterization of rademacher complexity. In *Symposium on Theory of Computing (STOC)*, pages 1657–1670, 2023. doi:10.1145/3564246.3585206.
- 14 Parikshit Gopalan, Adam Tauman Kalai, and Adam R Klivans. Agnostically learning decision trees. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 527–536, 2008. doi:10.1145/1374376.1374451.
- 15 Adam Klivans, Konstantinos Stavropoulos, and Arsen Vasilyan. Learning intersections of halfspaces with distribution shift: Improved algorithms and sq lower bounds. In Shipra Agrawal and Aaron Roth, editors, *Proceedings of Thirty Seventh Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, pages 2944–2978. PMLR, 30 June–03 July 2024. URL: <https://proceedings.mlr.press/v247/klivans24b.html>.
- 16 Adam Klivans, Konstantinos Stavropoulos, and Arsen Vasilyan. Testable learning with distribution shift. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 2887–2943. PMLR, 2024. URL: <https://proceedings.mlr.press/v247/klivans24a.html>.
- 17 Pravesh K. Kothari and Roi Livni. Improper Learning by Refuting. In *Innovations in Theoretical Computer Science (ITCS)*, pages 55:1–55:10, 2018. doi:10.4230/LIPIcs.ITCS.2018.55.
- 18 Eyal Kushilevitz and Yishay Mansour. Learning decision trees using the fourier spectrum. In *Symposium on Theory of Computing (STOC)*, pages 455–464, 1991.
- 19 Cassandra Marcussen, Ronitt Rubinfeld, and Madhu Sudan. Quality control in sublinear time: a case study via random graphs. *arXiv preprint arXiv:2508.16531*, 2025. doi:10.48550/arXiv.2508.16531.
- 20 Elchanan Mossel, Ryan O’Donnell, and Rocco A Servedio. Learning functions of k relevant variables. *Journal of Computer and System Sciences*, 69(3):421–434, 2004. doi:10.1016/J.JCSS.2004.04.002.

- 21 Ronitt Rubinfeld and Arsen Vasilyan. Testing distributional assumptions of learning algorithms. In *Symposium on Theory of Computing (STOC)*, pages 1643–1656, 2023. doi:10.1145/3564246.3585117.
- 22 Lucas Slot, Stefan Tiegel, and Manuel Wiedmer. Testably learning polynomial threshold functions. *Advances in Neural Information Processing Systems*, 37:3781–3831, 2024.
- 23 Salil Vadhan. On learning vs. refutation. In *Conference on Learning Theory (COLT)*, pages 1835–1848, 2017. URL: <http://proceedings.mlr.press/v65/vadhan17a.html>.