

Sublinear-Query Relative-Error Testing of Halfspaces

Xi Chen ✉️🏠 

Columbia University, New York, NY, USA

Anindya De ✉️🏠 

University of Pennsylvania, Philadelphia, PA, USA

Yizhi Huang ✉️ 

Columbia University, New York, NY, USA

Shivam Nadimpalli ✉️🏠 

MIT, Cambridge, MA, USA

Rocco A. Servedio ✉️🏠 

Columbia University, New York, NY, USA

Tianqi Yang ✉️🏠 

Columbia University, New York, NY, USA

Abstract

The *relative-error* property testing model was introduced in [11] to facilitate the study of property testing for “sparse” Boolean-valued functions, i.e. ones for which only a small fraction of all input assignments satisfy the function. In this framework, the distance from the unknown target function f that is being tested to a function g is defined as $\text{Vol}(f \triangle g)/\text{Vol}(f)$, where the numerator is the fraction of inputs on which f and g disagree and the denominator is the fraction of inputs that satisfy f .

Recent work [12] has shown that over the Boolean domain $\{0, 1\}^n$, any relative-error testing algorithm for the fundamental class of halfspaces (i.e. linear threshold functions) must make $\Omega(\log n)$ oracle calls. In this paper we complement the [12] lower bound by showing that halfspaces can be relative-error tested over $\mathbb{R}^n$ under the standard $N(0, I_n)$ Gaussian distribution using a *sublinear* number of oracle calls – in particular, substantially fewer than would be required for learning. Our results use a wide range of tools including Hermite analysis, Gaussian isoperimetric inequalities, and geometric results on noise sensitivity and surface area.

2012 ACM Subject Classification Theory of computation → Streaming, sublinear and near linear time algorithms

Keywords and phrases Property testing, relative-error testing, halfspaces, Gaussian space

Digital Object Identifier 10.4230/LIPIcs.ICALP.2026.61

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version:* <https://arxiv.org/abs/2604.01557>

Funding This work was awarded a grant by the AI Security Institute (AIS) via the Alignment Project (“Discovering rare harmful behaviors exhibited by high-dimensional AI systems”). This work is also funded by OpenAI. T.Y. and Y.H. are also supported by an Amazon Research Award and a Google CyberNYC award.

Xi Chen: NSF grants CCF-2106429 and CCF-2107187.

Anindya De: NSF grant CCF 2045128.

Yizhi Huang: NSF grants CCF-2211238, CCF-2312242, CCF-2106429, and CCF-2238221.

Rocco A. Servedio: NSF grants CCF-2211238 and CCF-2106429.

Tianqi Yang: NSF grants CCF-2211238, CCF-2106429, CCF-2312242, and AF-Medium 2212136.



© Xi Chen, Anindya De, Yizhi Huang, Shivam Naidimpalli, Rocco A. Servedio, and Tianqi Yang;

licensed under Creative Commons License CC-BY 4.0

53rd International Colloquium on Automata, Languages, and Programming (ICALP 2026).

Editors: Sayan Bhattacharya, Danupon Nanongkai, Michael Benedikt, and Gabriele Puppis;

Article No. 61; pp. 61:1–61:19



Leibniz International Proceedings in Informatics

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



1 Introduction

Over the past three decades Boolean function property testing has become a well-established research topic in theoretical computer science, see e.g. the books and monographs [45, 29, 3]. In this paper we consider an extension of the standard model of property testing, known as *relative-error* testing, which was recently proposed in [11].

To describe the relative-error model, we first recall that in the standard model of Boolean function property testing the goal is to distinguish between the two cases that (i) $f \in \mathcal{C}$, where $\mathcal{C}$ is the class of functions that have the property that is being tested, versus (ii) f is ε -far (under the uniform distribution) from every function in $\mathcal{C}$, meaning that

$$\text{dist}(f, \mathcal{C}) \geq \varepsilon, \quad \text{where } \text{dist}(f, \mathcal{C}) := \min_{g \in \mathcal{C}} \text{dist}(f, g) \quad \text{and} \quad \text{dist}(f, g) := \Pr_{\mathbf{x} \sim \{0,1\}^n} [f(\mathbf{x}) \neq g(\mathbf{x})].$$

A testing algorithm in the standard model can obtain information about the unknown function $f : \{0,1\}^n \rightarrow \{0,1\}$ that is being tested by making black-box oracle calls to f .

The impetus for the study of relative-error testing is that the standard model described above is poorly suited for testing *sparse* functions, i.e. functions which have few satisfying assignments. This is because any such function has very small uniform-distribution distance to the constant-0 function, and so the tester can simply ignore the input function f and answer according to the constant-0 function. Indeed, for sparse functions it is arguably more natural to ask whether f is “close” to having the property of interest (i.e. close to belonging to $\mathcal{C}$) for a notion of closeness that is *relative* to the small size of $f^{-1}(1)$ rather than to the “absolute” scale of all 2^n possible inputs.

With this motivation, the relative-error property testing model which was defined in [11] changes the standard model in the following ways:

- The distance between the target function f and a function g is now measured using *relative distance*, which is the uniform-distribution distance defined above scaled by the sparsity of f :

$$\text{rel-dist}(f, g) := \frac{\Pr_{\mathbf{x} \sim \{0,1\}^n} [f(\mathbf{x}) \neq g(\mathbf{x})]}{\Pr_{\mathbf{x} \sim \{0,1\}^n} [f(\mathbf{x}) = 1]}, \quad \text{i.e.} \quad \text{rel-dist}(f, g) = \frac{\text{dist}(f, g)}{\Pr_{\mathbf{x} \sim \{0,1\}^n} [f(\mathbf{x}) = 1]}.$$

Relative distance thus captures the distance between f and g at the scale of f , and continues to be meaningful even if f is very sparse.¹

- In addition to the usual black-box oracle for f , a relative-error testing algorithm can also access a *random sample oracle* $\text{SAMP}(f)$, which takes no input and returns a uniform random satisfying assignment $\mathbf{x} \sim f^{-1}(1)$. (Note that without a $\text{SAMP}(f)$ oracle, many black-box queries to f could be required even to find a single satisfying assignment of f .)

After the introduction of the relative-error testing model in [11], the relative-error testability of a number of well-studied properties (or classes of functions) was investigated in a number of papers [14, 15, 12, 9, 10]. Before turning to the specific results in those papers, we first give a quick overview of the general relationship between standard-model testability and relative-error testability.

¹ It is easy to verify, as observed in [11], that while rel-dist is not perfectly symmetric, if $\text{rel-dist}(f, g) = \varepsilon \leq 1/2$ then $\text{rel-dist}(g, f)$ is also $\Theta(\varepsilon)$, so rel-dist is symmetric up to constant factors in the setting we are interested in.

Standard-model versus relative-error testability. As shown already in [11], standard-model testing is never more difficult than relative-error testing: for any class $\mathcal{C}$, if $\mathcal{C}$ is relative-error testable to error ε using q oracle calls then $\mathcal{C}$ is also standard-model testable to error ε using $O(q/\varepsilon)$ oracle calls. (We remark that [14] gave a slight sharpening of this result under mild assumptions about the class $\mathcal{C}$.) On the other hand, [11] showed, by considering a contrived class of functions, that relative-error testing can sometimes be much more difficult (require many more queries) than standard-model testing. This artificial class of functions leaves open the following natural question: for “natural” classes of functions that are commonly studied in computational learning theory and concrete complexity, how does the query complexity of testing in the standard model compare to the relative-error model?

Relative-error testability of well-studied properties. The above question has been the subject of a significant amount of recent research [11, 14, 15, 12, 9, 10]. Most of these results suggest that – unlike the contrived example of [11] alluded to above – relative-error testing often turns out to be essentially no more (or at least not much more) difficult than standard-model testing for “natural” classes of functions $\mathcal{C}$. In particular, the papers [11, 14, 15, 9, 10] show that for the classes of monotone Boolean functions; unate Boolean functions; conjunctions; decision lists; k -juntas; subclasses of k -juntas such as size- k decision trees and size- k branching programs; and s -term DNF formulas, the query complexity of relative-error testing is at most some fixed polynomial in the query complexity of standard-model testing.

However, an intriguing exception was discovered in the very recent work of [12] on testing *halfspaces*, also known as linear threshold functions or LTFs.² Halfspaces are one of the most intensively studied classes in computational learning theory, dating from the introduction of the Perceptron algorithm more than sixty years ago [5, 43] down through to the present day [33, 17, 24, 23, 8]. They have also been studied intensively both in property testing, see e.g. [28, 37, 36, 20, 21, 32, 13], and in other fields including probability theory and Boolean function analysis, see e.g. [38, 41, 39, 7, 18, 25]. In [37] halfspaces were shown to be testable in the standard model with $\text{poly}(1/\varepsilon)$ queries independent of the ambient dimension n ; in contrast, [12] gave an $\tilde{\Omega}(\log n)$ lower bound on the number of queries that are required in the relative error model.

Given the [12] lower bound, a natural goal – which is the focus of the current paper – is to gain a better understanding of the query complexity of testing halfspaces in the relative-error model. In particular, the main question driving our research is the following: can n -dimensional halfspaces be relative-error tested with a number of queries that is *sublinear* in n ? As motivation for this specific question, we remark that (as is well known) in many learning models the problem of *learning* an unknown halfspace is known to require $\Theta(n)$ samples or queries. Since the complexity of learning is an upper bound on the complexity of testing (see [30]), it is natural to aim for $o(n)$ -query testing algorithms, since this would show that testing is easier than learning, which is one of the central motifs and motivations for property testing of Boolean functions. We further remark that the sample and query complexity of testing halfspaces in the *distribution-free* model is known to be $\tilde{\Theta}(n)$ [4, 13], which lends additional impetus to the question of understanding whether it is possible to achieve sublinear complexity in the relative-error testing model.

² Recall that an n -variable halfspace is a function of the form $\text{sign}(w \cdot x - \theta)$, with some $w \in \mathbb{R}^n$ and $\theta \in \mathbb{R}$.

1.1 Our contributions: Sublinear relative-error testing of halfspaces

Before describing our main results, we give some more context on halfspace testing in the standard model. Halfspace testing has primarily been studied under two distributions, which correspond to discrete and continuous domains. These are the usual uniform distribution over $\{0, 1\}^n$ [37, 36] and the standard $N(0, I_n)$ Gaussian distribution over $\mathbb{R}^n$ [37, 41, 20, 21]. For $N(0, I_n)$, the “standard-model” distance between $f, g : \mathbb{R}^n \rightarrow \{\pm 1\}$ is simply

$$\text{dist}(f, g) := \Pr_{\mathbf{x} \sim N(0, I_n)}[f(\mathbf{x}) \neq g(\mathbf{x})].$$

In the standard model [37] showed that halfspaces can be tested using only $\text{poly}(1/\varepsilon)$ queries under both the uniform distribution over $\{0, 1\}^n$ and the Gaussian $N(0, I_n)$ distribution over $\mathbb{R}^n$. The [37] result for $\{0, 1\}^n$ essentially uses their result over the Gaussian space as a subroutine.³ Thus, developing relative-error halfspace testing algorithms for the Gaussian distribution is a natural first step towards obtaining improved results for the uniform distribution over $\{0, 1\}^n$.

Towards this end, as the main results of this paper we provide three sublinear-query testing algorithms for halfspaces, all over the Gaussian space, where the relative distance between f and another function g , where $f, g : \mathbb{R}^n \rightarrow \{0, 1\}$, is defined (analogous to the Boolean setting) as

$$\text{rel-dist}(f, g) := \frac{\Pr_{\mathbf{x} \sim N(0, I_n)}[f(\mathbf{x}) \neq g(\mathbf{x})]}{\Pr_{\mathbf{x} \sim N(0, I_n)}[f(\mathbf{x}) = 1]}, \quad \text{i.e.} \quad \text{rel-dist}(f, g) = \frac{\text{dist}(f, g)}{\Pr_{\mathbf{x} \sim N(0, I_n)}[f(\mathbf{x}) = 1]},$$

and in this setting a call to the random sample oracle $\text{SAMP}(f)$ returns a draw from the Gaussian distribution conditioned on $f^{-1}(1)$. (See Section 3 for a more detailed definition of the model.)

► **Remark 1.** Before detailing our algorithmic results, we remark that by combining the $\text{poly}(1/\varepsilon)$ -query testing algorithms of [37, 41] for halfspaces under the Gaussian $N(0, I_n)$ distribution in the standard model and Fact 9 of [11], we easily obtain relative-error testing algorithms with query complexity $\text{poly}(1/(\varepsilon p))$ when we are promised that the “Gaussian volume” $\Pr_{\mathbf{x} \sim N(0, I_n)}[f(\mathbf{x}) = 1]$ of the function f being tested is at least p . (Intuitively, this is because achieving relative error ε for a “ p -volume” function is the same as achieving standard error εp .) However, this is a very poor bound in the (most interesting) case when p is very small; in the results below, we achieve an exponential improvement by giving algorithms whose dependence on n is sublinear and whose dependence on p is only inverse *poly-logarithmic*.

Our results. For our first two results, the algorithm is assumed to be given (an accurate estimate of) the “Gaussian volume” $p := \Pr_{\mathbf{x} \sim N(0, I_n)}[f(\mathbf{x}) = 1]$ of the unknown target function f , which we sometimes refer to as the *volume* of f and denote by $\text{Vol}(f)$. In the third result, the algorithm is only given a lower bound $p_{\min}$ on $\text{Vol}(f)$ rather than an accurate estimate. We first give informal statements of our three algorithms’ guarantees and provide detailed theorem statements later.

1. The first algorithm (see Theorem 2) uses $\text{poly}(\log(1/p), 1/\varepsilon)$ samples and queries to do ε -relative-error testing. At a high level, similar to [41], the algorithm relies on the robust extremal isoperimetric properties of halfspaces over the Gaussian space. See Section 2.1 for a more detailed description of the ideas underlying this algorithm.

³ We remark that a very different tester for the Gaussian space than the [37] tester was given by Mossel and Neeman in [41]; these results will be discussed in more detail later.

2. The second algorithm (see Theorem 3) uses $\tilde{O}(\sqrt{n}) \cdot \text{poly}(\log(1/p), 1/\varepsilon)$ samples to do ε -relative-error testing. In contrast with the first one, this algorithm does not use any black-box queries to f , only random samples drawn from $\text{SAMP}(f)$. The high-level idea behind this tester is that for halfspaces over the Gaussian space, the level-1 Hermite weight depends only on the bias of the function, and this relationship is robust enough to serve as the basis for a testing algorithm (Section 2.2 gives a more complete overview). We note that similar ideas were used in the original work of [37] for testing under the Gaussian distribution, as well as a number of other related works including [32] and [22].
3. The third algorithm (see Theorem 4) addresses the problem when p is unknown; it only requires a lower bound $p_{\min}$ on the value of p rather than an accurate estimate. It uses $O(\sqrt{n}) \cdot \text{poly}(\log(1/p_{\min}), 1/\varepsilon)$ samples and queries to do ε -relative-error testing. The high level idea is to use the estimator from the proof of Theorem 3 to obtain an *upper* bound on p , in conjunction with a variant of Theorem 2 which only errs if it is given an estimate of p which is *too low*. See Section 2.3 for a more detailed overview of the main ideas underlying this algorithm.

Here are detailed statements of our algorithmic results:

► **Theorem 2** (Gaussian halfspace testing for known p). *There is an algorithm GSA-TEST with the following property: If GSA-TEST is given sample access $\text{SAMP}(f)$ and black-box access $\text{MQ}(f)$ to a measurable function $f : \mathbb{R}^n \rightarrow \{0, 1\}$, a parameter ε , and an estimate $\hat{p}$ of $p := \text{Vol}(f)$ satisfying*

$$\frac{p}{1 + \zeta} \leq \hat{p} \leq (1 + \zeta)p, \quad \text{where} \quad \zeta = \frac{c_1 \varepsilon^2}{\log^2(1/p)}$$

for a suitably small absolute constant $c_1 > 0$, then it makes $O(\varepsilon^{-14} \log^{13}(1/p))$ calls to the oracles to test whether f is a halfspace or ε -far from any halfspace in relative distance.

► **Theorem 3** (Sample-based Gaussian halfspace testing for known p). *There is a sample-based algorithm HERMITE-TEST with the following property: If it is given sample access $\text{SAMP}(f)$ to a measurable function $f : \mathbb{R}^n \rightarrow \{0, 1\}$, a parameter ε , and an estimate $\hat{p}$ of $p := \text{Vol}(f)$ satisfying*

$$\frac{p}{1 + \eta} \leq \hat{p} \leq (1 + \eta)p, \quad \text{where} \quad \eta = \frac{c_2 \varepsilon^2}{\log(1/p)}$$

for some suitably small absolute constant $c_2 > 0$, then HERMITE-TEST makes

$$\max \left\{ \Theta \left(\frac{\sqrt{n}}{\varepsilon^2} + \frac{\log^2(1/p)}{\varepsilon^4} \right), \tilde{\Theta} \left(\frac{\sqrt{n}}{\varepsilon^7} \right) \right\}$$

calls to $\text{SAMP}(f)$ to test whether f is a halfspace or ε -far from any halfspace in relative distance.

We remark that the requirement in Theorem 3 that HERMITE-TEST is given an estimate of p is inherent to any sample-based algorithm rather than a limitation of our arguments. In the full version we observe that any algorithm which uses only samples from $\text{SAMP}(f)$ and is *not* given an estimate of p must draw $\Omega(n)$ samples, even if we are given a promise that p is either $1/2$ or 1 .

► **Theorem 4** (Gaussian halfspace testing for unknown p). *There is an algorithm COMBINED-TEST with the following property: If it is given sample access $\text{SAMP}(f)$ and black-box access $\text{MQ}(f)$ to an unknown measurable function $f : \mathbb{R}^n \rightarrow \{0, 1\}$, which is guaranteed to satisfy $p := \text{Vol}(f) \geq p_{\min}$, and a parameter ε , then it makes*

$$\Theta\left(\frac{\sqrt{n \cdot \log(1/p_{\min})}}{\varepsilon^2} + \frac{\log^6(1/p_{\min})}{\varepsilon^{12}}\right)$$

calls to the oracles to test whether f is a halfspace or ε -far from any halfspace in relative distance.

2 Technical overview

2.1 Our algorithm for halfspace testing using queries and samples: Overview of Theorem 2

To explain the main idea of Theorem 2, we start by recalling two fundamental notions in Gaussian analysis. For any set $A \subseteq \mathbb{R}^n$, its *Gaussian volume*, denoted by $\text{Vol}(A)$, is $\Pr_{\mathbf{x} \sim N(0, I_n)}[\mathbf{x} \in A]$. Furthermore, if A is sufficiently smooth, we can also define its *Gaussian surface area*, denoted $\text{surf}(A)$, as $\int_{x \in \partial A} \varphi_n(x) d\sigma(x)$, where $\varphi_n(\cdot)$ denotes the standard n -dimensional Gaussian density; in other words, we integrate the standard Gaussian density over the surface of A . Given these notions of volume and surface area over the Gaussian space, one is naturally led to the “isoperimetric problem:” *for a given (Gaussian) volume, what set minimizes the (Gaussian) surface area?*

This question was answered by Borell [6] and Sudakov-Tsirelson [46], who showed that for any $0 < p < 1$, for any measurable set $A \subseteq \mathbb{R}^n$ with $\text{Vol}(A) = p$, it always holds that $\text{surf}(A) \geq \text{surf}(H_p)$ where H_p is a halfspace with $\text{Vol}(H_p) = p$. In other words, in the Gaussian space, for any given volume p , the halfspace with volume p has minimal surface area. Subsequently, Ehrhard [26] showed that halfspaces are the *unique* minimizers of the Gaussian surface area. In other words, if a set $A \subseteq \mathbb{R}^n$ has $\text{Vol}(A) = p$ and $\text{surf}(A) = \text{surf}(H_p)$, then the set A is a halfspace.

One is now naturally led to the *robustness* question: Suppose a set A has $\text{Vol}(A) = p$ and $\text{surf}(A)$ is δ -close to $\text{surf}(H_p)$. Then is it the case that A is δ' -close to a halfspace, where $\delta' \rightarrow 0$ as $\delta \rightarrow 0$? Cianchi *et al.* [16] were the first to affirmatively answer this question. Soon thereafter, in a significant breakthrough, Mossel and Neeman [40] obtained a “dimension free robust isoperimetry result”, i.e. one in which the dependence between δ' and δ is independent of the ambient dimension n .

This result was quantitatively improved by Eldan [27] and then by Barchiesi *et al.* [2] (the latter result is the underlying technical ingredient we use in Theorem 2). A natural question at this point is “what is the connection between robust isoperimetry and testing halfspaces?”

The connection arises from a 2014 result of Neeman [42], who gave an algorithm for testing surface area over Gaussian space. The precise technical guarantee is somewhat cumbersome to state, but roughly speaking, the algorithm makes $\text{poly}(S, 1/\varepsilon)$ -queries to an unknown set A and (i) accepts with high probability if $\text{surf}(A) \leq S$, but (ii) rejects with high probability if $\text{surf}(B) > (1 + \varepsilon)S$ for every set B which is ε -close to A^4 . Now, observe

⁴ This perturbation of A is necessary because one can increase the surface area of a set arbitrarily by modifying on a measure zero set.

that if an unknown set A of some given volume p is a halfspace, then its surface area is exactly given by $\varphi(\Phi^{-1}(p))$ where Φ is the cdf and φ is the pdf of the standard Gaussian. So to test whether A is a halfspace, we can use Neeman’s algorithm to test whether its surface area is essentially that of a halfspace of volume p , and accept if and only if that is the case. Correctness of this simple algorithm is established using the dimension-free robust isoperimetry result of Barchiesi *et al.* [2] that was mentioned above.

2.2 Our algorithm for sample-based halfspace testing: Overview of Theorem 3

Our starting point for Theorem 3 is the work of Matulef, O’Donnell, Rubinfeld, and Servedio [37] which gives an algorithm (the “MORS algorithm”) that tests halfspaces over $\mathbb{R}^n$ using $O_\varepsilon(1)$ queries. The MORS algorithm relies on two structural ingredients:

- First, there is a function $U : [0, 1] \rightarrow \mathbb{R}$ (see the full version) which, given as input the volume $\text{Vol}(f)$ of a function $f : \mathbb{R}^n \rightarrow \{0, 1\}$, tells us exactly what the *level-1 Hermite weight*

$$\mathbf{W}^1[f] := \sum_{i=1}^n \mathbf{E}_{\mathbf{x} \sim N(0, I_n)} [f(\mathbf{x}) \cdot \mathbf{x}_i]^2$$

would be if the function f were the indicator function of a halfspace.

- Second, if $U(\text{Vol}(f)) \approx \mathbf{W}^1[f]$, then the set f is close to some halfspace.

The above facts suggest a natural algorithm: estimate $\text{Vol}(f)$ and $\mathbf{W}^1[f]$, and then compare $U(\text{Vol}(f))$ to $\mathbf{W}^1[f]$. (We remark that the MORS algorithm uses queries to do this.)

The algorithm HERMITE-TEST (see the full version) that yields Theorem 3 can be viewed as a relative-error, sample-based variant of the MORS algorithm. Our main technical lemma (see the full version) shows that if $U(\text{Vol}(f))$ is sufficiently close to $\mathbf{W}^1[f]$ in a suitable sense, then f is close *under relative distance* to some halfspace. In this sense, this lemma can be viewed as a relative-error strengthening of the second structural ingredient of the MORS algorithm above. With this lemma in hand, we apply a test which is similar in spirit to the MORS algorithm: estimate $\mathbf{W}^1[f]$ and see if it is (sufficiently) close to $U(\text{Vol}(f))$.

This leads to a second technical challenge: in order to get a sample-based algorithm we must accurately estimate $\mathbf{W}^1[f]$ using access to samples from $\text{SAMP}(f)$ alone. (Note that the MORS algorithm relies on query access to the function f , and there are sample-based testers [1, 32] that rely on *labeled* samples from f as opposed to just positive samples from $\text{SAMP}(f)$.) To accomplish this, we employ a recent estimator used in a different context by [22, 19] (for the problem of detecting distribution truncation) which is as follows: given a draw of $2m$ i.i.d. samples $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}, \mathbf{y}^{(1)}, \dots, \mathbf{y}^{(m)} \leftarrow \text{SAMP}(f)$, we compute the statistic

$$\mathbf{T} := \frac{1}{m^2} \sum_{i \neq j} \mathbf{x}^{(i)} \cdot \mathbf{y}^{(j)}.$$

It is readily verified that $\mathbf{E}[\mathbf{T}] = \text{Vol}(f)^{-2} \cdot \mathbf{W}^1[f]$; since $\text{Vol}(f)$ is known, we can estimate $\mathbf{W}^1[f]$ if we can estimate $\mathbf{E}[\mathbf{T}]$. To estimate $\mathbf{E}[\mathbf{T}]$, we must establish a bound on the variance of $\mathbf{T}$; to do this, we employ (consequences of) *hypercontractivity of the Gaussian measure* (see Chapters 9 and 11 of [44]), drawing inspiration from recent work [19].

2.3 Our algorithm for halfspace testing when the volume is unknown: Overview of Theorem 4

We obtain the algorithm of Theorem 4, which does not need to have (an estimate of) the value of p , by combining the ideas underlying Theorem 2 and Theorem 3, as well as bringing in new ideas. In particular, a crucial structural property of LTFs which we use for Theorem 4 is the following: not only does the level-1 Hermite weight of an LTF f depend only on the bias of the function $\mathbf{E}[f]$ as mentioned earlier, but under the Gaussian distribution LTFs are in fact *robustly extremal*, among all $\{0, 1\}$ -valued functions, in terms of how their level-1 Hermite weight depends on the bias of the function. In more detail, writing h_p to denote a halfspace with Gaussian volume p , it is the case that *any* function $f : \mathbb{R}^n \rightarrow \{0, 1\}$ with Gaussian volume p must satisfy

$$\mathbf{W}^1[f] \leq \mathbf{W}^1[h_p]. \quad (1)$$

(As alluded to in the last paragraph of Section 2.2, the LHS of the inequality above is precisely $\text{Vol}(f)^2 \cdot \mathbf{E}_{\mathbf{x}, \mathbf{y} \sim f^{-1}(1)}[\mathbf{x} \cdot \mathbf{y}]$.) Moreover the inequality can only be an equality if f is a halfspace, and it can only be close to an equality if f is close to a halfspace. Versions of this structural property are established, in varying degrees of explicitness, in Theorem 7 of [34], Theorem 26 of [37], and Theorem 3.3.4 of [31]; our result (see the full version) gives a version of this which is – crucially, for our purposes – quantitatively stronger than those earlier results in the small- p regime.

With this perspective, we can reinterpret the testing algorithm of Theorem 3 in the following light: if the algorithm were *not* given p , it could compute an estimate τ of $\mathbf{E}_{\mathbf{x}, \mathbf{y} \sim f^{-1}(1)}[\mathbf{x} \cdot \mathbf{y}]$ using draws from $\text{SAMP}(f)$ and solve the equation

$$\tau = \frac{\mathbf{W}^1[h_{p_2}]}{(p_2)^2}$$

for the value p_2 . If f were an LTF and the value of τ were precisely the value of $\mathbf{E}_{\mathbf{x}, \mathbf{y} \sim f^{-1}(1)}[\mathbf{x} \cdot \mathbf{y}]$, then this would give us the exact correct value $p_2 = p$. On the other hand, because of the extremal property (1) mentioned above, if f is far from every LTF then (up to a small factor due to estimation error) the value of p_2 will be *significantly larger than* the true value $p = \text{Vol}(f)$.

Now let us return to Theorem 2; the key for our analysis is that this result relies on a robust extremal property of LTFs that *goes in the other direction*. Recall that the Theorem 2 algorithm essentially works by estimating

$$\Pr_{\mathbf{x} \sim f^{-1}(1), \mathbf{y} \sim N_\lambda(\mathbf{x})} [f(\mathbf{x}) \neq f(\mathbf{y})], \quad \text{i.e.} \quad \Pr_{\mathbf{x} \sim f^{-1}(1), \mathbf{y} \sim N_\lambda(\mathbf{x})} [f(\mathbf{y}) = 0]$$

(where “ $\mathbf{y} \sim N_\lambda(\mathbf{x})$ ” means that $\mathbf{y}$ is a λ -correlated random perturbation of $\mathbf{x}$; see Section 4.2 for a formal definition), and checking whether it takes the “right value” that it should take for an 0/1-valued LTF with expectation p , which is $\frac{1}{2p} \text{Sens}_\lambda(LTF_p) = C\sqrt{\lambda \ln(1/p)}$. In the Theorem 2 algorithm we use a value of λ which depends on p , but for a *fixed* value of λ , the function $\frac{1}{2p} \text{Sens}_\lambda(LTF_p)$ is a *decreasing* function of p ; as we will see, this will be crucial for us.

Now, known isoperimetric properties of halfspaces can be shown to imply that if f is any $\{0, 1\}$ -valued function f with $\text{Vol}(f) = p$, then f must (essentially) satisfy

$$\Pr_{\mathbf{x} \sim f^{-1}(1), \mathbf{y} \sim N_\lambda(\mathbf{x})} [f(\mathbf{x}) \neq f(\mathbf{y})] \geq \frac{1}{2p} \text{Sens}_\lambda(h_p);$$

i.e. that halfspaces are *minimizers* of the quantity $\Pr_{\mathbf{x} \sim f^{-1}(1), \mathbf{y} \sim N_\lambda(\mathbf{x})}[f(\mathbf{x}) \neq f(\mathbf{y})]$, which the algorithm of Theorem 2 estimates from samples and queries. This means that if we run the algorithm of Theorem 2 with the value p_2 obtained from the modified algorithm of Theorem 3 as described earlier, there are two possibilities:

- If f is a halfspace, then p_2 will be (approximately) equal to the true value of p , and since we are running the algorithm of Theorem 2 with (essentially) the right value of p , it will accept.
- On the other hand, if f is far from every halfspace, then as explained earlier p_2 will be significantly *larger* than the true value of p . But since $\frac{1}{2p} \text{Sens}_\lambda(LTF_p)$ is a decreasing function of p , the algorithm of Theorem 2 can only accept if it is run on a “guessed” value of $\text{Vol}(f)$ which is either *equal to or smaller than* the true value of $\text{Vol}(f)$. So in this case, the algorithm will reject.

This concludes the overview of the proof of Theorem 4. One point of technical interest is that the actual proof of Theorem 4 does not use Theorem 2 as a black-box. In fact, unlike Theorem 2 which crucially relies on the robust isoperimetry results of [2], Theorem 4 only relies on the classical result of Borell [7] which states that halfspaces are the minimizers of noise sensitivity. The result of Borell has many proofs by now, including some quite elementary (see [18] and references therein). So, arguably, the proof of Theorem 4 is significantly simpler than that of Theorem 2 (though conceptually, Theorem 4 is inspired by Theorem 2).

3 Preliminaries for relative-error testing over $N(0, 1)^n$

We use boldfaced letters such as $\mathbf{x}, \mathbf{f}, \mathbf{A}$, etc. to denote random variables (which may be real-valued, vector-valued, function-valued, or set-valued; the intended type of the random variable will be clear from the context). We write $\mathbf{x} \sim \mathcal{D}$ to indicate that the random variable $\mathbf{x}$ is distributed according to probability distribution $\mathcal{D}$.

We will write $(e_i)_{i=1}^n$ for the collection of standard basis vectors in $\mathbb{R}^n$. Given two sets A and B , we use $A \triangle B$ to denote their symmetric difference, i.e. $A \triangle B = (A \setminus B) \cup (B \setminus A)$. As in [44] we write $\mathbb{N}$ for the set $\mathbb{N} = \{0, 1, 2, \dots\}$.

The Gaussian distribution and Gaussian relative-error testing. We will write $N(0, I_n)$ to denote the n -dimensional standard Gaussian distribution, and denote its density function by φ_n , i.e., for $x \in \mathbb{R}^n$,

$$\varphi_n(x) = \frac{1}{\sqrt{(2\pi)^n}} \cdot e^{-\|x\|^2/2}.$$

We frequently write $\varphi \equiv \varphi_1$ to denote the one-dimensional standard Gaussian density. We write $\text{Vol}(K)$ to denote the Gaussian measure of a (Lebesgue measurable) set $K \subseteq \mathbb{R}^n$, that is

$$\text{Vol}(K) := \Pr_{\mathbf{x} \sim N(0, I_n)}[\mathbf{x} \in K].$$

Given a 0/1-valued function $f : \mathbb{R}^n \rightarrow \{0, 1\}$, we write $\text{Vol}(f)$ to denote $\text{Vol}(f^{-1}(1))$.

Finally, we will write $\Phi : \mathbb{R} \rightarrow (0, 1)$ for the c.d.f. of $N(0, 1)$, i.e.

$$\Phi(r) = \Pr_{\mathbf{g} \sim N(0, 1)}[\mathbf{g} \leq r] = \int_{-\infty}^r \varphi(x) dx.$$

We will frequently use the following standard tail bound on (univariate) Gaussian random variables:

► **Proposition 5** (Proposition 2.1.2 of [47] or Exercise 2.2 of [48]). *Suppose $g \sim N(0, 1)$ is a one-dimensional Gaussian random variable. Then for all $r > 0$, we have*

$$\left(\frac{1}{r} - \frac{1}{r^3}\right) \cdot \varphi(r) \leq \Phi(-r) = \Pr_{g \sim N(0,1)}[g \geq r] \leq \left(\frac{1}{r} - \frac{1}{r^3} + \frac{3}{r^5}\right) \cdot \varphi(r).$$

From Proposition 5 we get the following estimate on $\Phi(\cdot)$ that will be convenient:

► **Fact 6.** *Let r be any positive value bounded away from 0 by an absolute constant, e.g. $r \geq 0.01$. Then we have $1 - \Phi(r) = \Phi(-r) = \Theta(e^{-r^2/2}/r)$.*

Gaussian relative-error testing. The model of relative-error testing under the Gaussian distribution is a natural variant of the model of relative-error testing over $\{0, 1\}^n$ that was described earlier. Now a call to $\text{SAMP}(f)$ returns a draw from $N(0, 1)^n|_{f^{-1}(1)}$, the standard Gaussian distribution conditioned on $f^{-1}(1)$. Similar to before, a relative-error testing algorithm for $\mathcal{C}$ must output “yes” with high probability (say at least 9/10; this success probability can be easily amplified) if $f \in \mathcal{C}$, and must output “no” with high probability (again, say at least 9/10) if $\text{rel-dist}(f, \mathcal{C}) \geq \varepsilon$, where $\text{rel-dist}(f, \mathcal{C}) = \min_{g \in \mathcal{C}} \text{rel-dist}(f, g)$ and the relative distance between f and g is defined as

$$\text{rel-dist}(f, g) = \frac{\text{Vol}(f^{-1}(1) \triangle g^{-1}(1))}{\text{Vol}(f)}.$$

Testing algorithms in the standard model. For handling certain edge cases, we will rely on known testing algorithms in the standard (not relative-error) property testing model. One such result that we will use is the following theorem due to [37]:

► **Theorem 7** (Theorem 26 of [37]). *There is an algorithm MORS-TESTER that, when given error parameter $\varepsilon > 0$ and oracle access to $f : \mathbb{R}^n \rightarrow \{0, 1\}$, makes $O(1/\varepsilon^{12})$ queries to f and has the following guarantee:*

1. *If f is a halfspace, the algorithm outputs **accept** with probability at least 0.99.*
2. *If $\text{Vol}(f^{-1}(1) \triangle h^{-1}(1)) \geq \varepsilon$ for every halfspace h , then the algorithm outputs **reject** with probability at least 0.99.*

For sample-based testing, we will use the following result due to Harms [32]:

► **Theorem 8** (Theorem 1.1 of [32]). *There is an algorithm HARMS-TESTER that, when given error parameter $\varepsilon > 0$ and access to random labeled examples $(\mathbf{x}, f(\mathbf{x}))$ with each $\mathbf{x} \sim N(0, I_n)$, uses $\tilde{O}(\sqrt{n}/\varepsilon^7)$ samples and has the following guarantee:*

1. *If f is a halfspace, the algorithm outputs **accept** with probability at least 0.99.*
2. *If $\text{Vol}(f^{-1}(1) \triangle h^{-1}(1)) \geq \varepsilon$ for every halfspace h , then the algorithm outputs **reject** with probability at least 0.99.*

4 Testing with known volume via Gaussian surface area: Proof of Theorem 2

In this section we prove Theorem 2. Before we start, note that we can make a simplifying assumption, which is that $p := \text{Vol}(f)$ is at most 0.1. This is because if $p > 0.1$, we can just run the testing algorithm from [37] (Theorem 7) with error parameter $\varepsilon/10$; as $p > 0.1$, standard-model $\varepsilon/10$ -testing implies ε -relative-error testing. So for the rest of this section, we assume that $p \leq 0.1$.

4.1 Gaussian surface area

We begin by recalling the notion of surface area in the Gaussian space.

► **Definition 9.** Let $A \subseteq \mathbb{R}^n$. The surface area of A (under the standard Gaussian measure φ_n) is given by

$$\text{surf}(A) := \lim_{\delta \rightarrow 0^+} \frac{\text{Vol}(A_\delta) - \text{Vol}(A)}{\delta},$$

where $A_\delta := \{x : d(x, A) \leq \delta\}$. For sets A with a smooth boundary, $\text{surf}(A)$ is equivalent to

$$\int_{x \in \partial A} \varphi_n(x) d\sigma(x),$$

where $d\sigma(x)$ is the standard surface area element in $\mathbb{R}^n$ and ∂A denotes the boundary of the set A .

There is a close connection between the notion of surface area and noise sensitivity, as shown by Ledoux [35]:

► **Theorem 10.** For any $t \geq 0$ and any set $A \subseteq \mathbb{R}^n$ with $\mathcal{C}^1$ boundary, we have

$$\text{NS}_t(A) \leq \frac{2\sqrt{t}}{\sqrt{\pi}} \cdot \text{surf}(A).$$

A near-converse of this result was shown by Neeman [42]: every set A as above is close to a set B whose surface area can be bounded in terms of the noise sensitivity of A .

► **Theorem 11** (Theorem 1.2, Theorem 2.1 of [42]). Let $A \subseteq \mathbb{R}^n$ be a set with $\mathcal{C}^1$ boundary and let $t, \xi > 0$. Then, there exists a set $B \subseteq \mathbb{R}^n$ such that

1. $\text{Vol}(A \triangle B) \leq \frac{\text{NS}_t(A)}{\xi}$; and
2. $\text{surf}(B) \leq \sqrt{\frac{\pi}{2}} \cdot \left(1 + O\left(\frac{\xi}{\sqrt{\log(1/\xi)}}\right)\right) \cdot \frac{1}{\sqrt{e^{2t} - 1}} \cdot \text{NS}_t(A).$

We recall (see e.g. Chapter 5 of [44]) that the function

$$I(p) := \varphi(\Phi^{-1}(p)) \tag{2}$$

is sometimes known as the *Gaussian isoperimetric function* (cf. Fact 14 below). This function will be used in various ways throughout our arguments, including in the proof of the following simple fact:

► **Fact 12.** Let $0 < q \leq p \leq 1/2$ and $p/q \leq (1 + \varepsilon)$. Then,

$$\varphi(\Phi^{-1}(p)) \geq \varphi(\Phi^{-1}(q)) \geq \frac{1}{(1 + \varepsilon)} \varphi(\Phi^{-1}(p)).$$

Proof. The function $I(\cdot)$ is easily seen to be monotonically non-decreasing in the interval $(0, 1/2]$, which gives the first inequality. Moreover, it is known that the function $I(p)$ is concave (see Exercise 5.24 [44]). This implies that

$$I(q) \geq \frac{q}{p} I(p) \geq \frac{1}{1 + \varepsilon} I(p). \quad \blacktriangleleft$$

We will also need some basic facts about the function $\varphi(\Phi^{-1}(p))/p$ which we state here.

61:12 Sublinear-Query Relative-Error Testing of Halfspaces

▷ **Claim 13.** The function $R(p) := \frac{\varphi(\Phi^{-1}(p))}{p}$ is decreasing in the interval $(0, 1/2)$.

Proof. Let $s \in (-\infty, 0)$ be chosen so that $\int_{-\infty}^s \varphi(t)dt = p$. Then, note that

$$R(p) = \frac{\varphi(s)}{\int_{-\infty}^s \varphi(t)dt}. \quad \triangleleft$$

We also need the following simple fact characterizing the surface area of any halfspace in the Gaussian space in terms of its volume; the proof of the final equality is a calculation using standard bounds on the Gaussian pdf and cdf, in particular Proposition 5.

► **Fact 14.** Let $f : \mathbb{R}^n \rightarrow \{0, 1\}$ be a halfspace and let $\text{Vol}(f^{-1}(1)) = p$. Then

$$\text{surf}(f^{-1}(1)) = \varphi(\Phi^{-1}(p)) = \Theta\left(p\sqrt{\ln(1/\min\{p, 1-p\})}\right).$$

4.2 Other basic algorithmic and structural ingredients

Gaussian noise sensitivity. For any $x \in \mathbb{R}^n$ and any $t \geq 0$, we let $N_t(x)$ denote the distribution over $\mathbb{R}^n$ defined by $e^{-t}x + \sqrt{1 - e^{-2t}}\mathbf{y}$ where $\mathbf{y} \sim N(0, I_n)$. Further, for any function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $t \geq 0$, we let $P_t f(x) = \mathbf{E}_{\mathbf{y} \sim N_t(x)}[f(\mathbf{y})]$.

► **Definition 15.** For any Boolean function $f : \mathbb{R}^n \rightarrow \{0, 1\}$ and $t \geq 0$, we define $\mathbf{NS}_t(f)$ (read as the “noise sensitivity” of f) as

$$\mathbf{NS}_t(f) = \Pr_{\mathbf{x} \sim N(0, I_n), \mathbf{y} \sim N_t(\mathbf{x})} [f(\mathbf{x}) \neq f(\mathbf{y})].$$

For a set $A \subseteq \mathbb{R}^n$, we define $\mathbf{NS}_t(A)$ to be $\mathbf{NS}_t(\mathbf{1}_A)$ where $\mathbf{1}_A(\cdot)$ is the indicator function of A .

We observe that using calls to $\text{SAMP}(f)$ and $\text{MQ}(f)$, we can efficiently estimate the (normalized) noise sensitivity of f :

► **Lemma 16** (Estimating the normalized noise sensitivity). For noise parameter $t > 0$ and error parameter $\kappa > 0$, there is an algorithm *EST-SENSE* which for any $f : \mathbb{R}^n \rightarrow \{0, 1\}$, given query and sample access to f , outputs a $\pm\kappa$ accurate additive estimate to $\mathbf{NS}_t(f)/\text{Vol}[f]$ with confidence 0.99 using $\Theta(\kappa^{-2})$ samples from $\text{SAMP}(f)$ and the same number of queries to $\text{MQ}(f)$.

Proof. Sample $\mathbf{x} \sim N(0, I_n)$ and $\mathbf{y} \sim N_t(\mathbf{x})$. Then, note that

$$\mathbf{NS}_t(f) = \Pr[f(\mathbf{x}) \neq f(\mathbf{y})] = \Pr[\mathbf{x} \in f^{-1}(1) \wedge \mathbf{y} \notin f^{-1}(1)] + \Pr[\mathbf{y} \in f^{-1}(1) \wedge \mathbf{x} \notin f^{-1}(1)].$$

As the distribution of $(\mathbf{x}, \mathbf{y})$ is exchangeable, it follows that

$$\mathbf{NS}_t(f) = 2 \Pr[\mathbf{x} \in f^{-1}(1) \wedge \mathbf{y} \notin f^{-1}(1)] = 2\text{Vol}(f) \cdot \Pr[f(\mathbf{y}) = 0 \mid f(\mathbf{x}) = 1].$$

Standard sampling shows that the RHS probability can be estimated to error $\pm\kappa/2$ with confidence 0.99 using $\Theta(\kappa^{-2})$ calls to $\text{SAMP}(f)$ and the same number of queries to f . The lemma follows. ◀

Another key ingredient we require is the following deep result of Barchiesi et al. [2] which obtains a sharp stability version of the Gaussian isoperimetric inequality. In particular, they establish the following:

► **Theorem 17** (Main theorem of [2] and subsequent discussion: sharp stability version of Gaussian isoperimetric inequality). *Let E be a measurable subset of $\mathbb{R}^n$ and let $s \in \mathbb{R}$ satisfy $\text{Vol}(E) = \Phi(s)$, i.e., the Gaussian volume of E is the same as that of the one-dimensional halfspace $\{x : x \leq s\}$. Define the Gaussian isoperimetric deficit (namely, the gap between the two sides of the Gaussian isoperimetric inequality) to be*

$$D(E) := \text{surf}(E) - \varphi(\Phi^{-1}(\text{Vol}(E))) = \text{surf}(E) - \frac{1}{\sqrt{2\pi}} e^{-s^2/2}.$$

Then, there is a halfspace $H = \{x : x \cdot \omega \leq s\}$ (for some unit vector ω) such that

$$\text{Vol}(E \triangle H)^2 \leq C'(1 + s^2) e^{-s^2/2} \cdot D(E),$$

where $C' > 0$ is a universal constant (which can be taken to be at most $320\sqrt{2\pi^2}$).

► **Remark 18.** The main theorem in [2] defines the isoperimetric deficit $D(E) = \text{surf}(E) - e^{-s^2/2}$ – i.e. vis-a-vis their definition, we have an extra factor of $1/\sqrt{2\pi}$. The reason is just that while our definition of the surface area of a set A is $\int_{x \in \partial A} \varphi_n(x) d\sigma(x)$, the definition in [2] is equivalent to defining it as $\int_{x \in \partial A} \sqrt{2\pi} \varphi_n(x) d\sigma(x)$.

An easy corollary of this is the following:

► **Corollary 19.** *Let E be a measurable subset of $\mathbb{R}^n$ such that $\text{Vol}(E) = p$ with $p \leq 0.1$. Suppose*

$$D(E) \leq \frac{\beta^2 p}{\log^{3/2}(1/p)}$$

for some β . Then there is a halfspace H that satisfies $\text{Vol}(H) = p$ and $\text{Vol}(E \triangle H) \leq C\beta p$ for some absolute constant $C > 0$. (Recalling $\varphi(\Phi^{-1}(p)) = \Theta(p\sqrt{\log(1/p)})$, an equivalent statement is if

$$D(E) \leq \frac{\beta^2}{\log^2(1/p)} \cdot \varphi(\Phi^{-1}(p)),$$

then there is a halfspace H that satisfies $\text{Vol}(H) = p$ and $\text{Vol}(E \triangle H) \leq C\beta p$.)

Proof. Let $H = \{x : x_1 \leq s\}$ be a one-dimensional halfspace such that $\text{Vol}(H) = p$. Note that

$$\varphi(\Phi^{-1}(p)) = \frac{1}{\sqrt{2\pi}} e^{-s^2/2}$$

so $e^{-s^2/2} = \Theta(\varphi(\Phi^{-1}(p)))$ and using $\varphi(\Phi^{-1}(p)) = \Theta(p\sqrt{\log(1/p)})$, we have $s^2 = \Theta(\log(1/p))$.

Now, applying Theorem 17, we have that there is a halfspace H with $\text{Vol}(H) = p$ and

$$\text{Vol}(E \triangle H)^2 \leq O\left(\log(1/p) \cdot \varphi(\Phi^{-1}(p)) \cdot \frac{\beta^2}{\log^2(1/p)} \cdot \varphi(\Phi^{-1}(p))\right).$$

This finishes the proof by plugging in $\varphi(\Phi^{-1}(p)) = \Theta(p\sqrt{\log(1/p)})$. ◀

4.3 Proof of Theorem 2

We now have all the necessary ingredients required to prove Theorem 2. The algorithm alluded to in Theorem 2, called GSA-TEST, is given in Algorithm 1.

■ **Algorithm 1** A relative-error LTF tester over Gaussian space using samples and queries.

Input: $\text{SAMP}(f)$, $\text{MQ}(f)$, $\varepsilon \in (0, 1]$, $\hat{p} \in [(1 + \zeta)^{-1}p, (1 + \zeta)p]$ where $p = \text{Vol}(f) \leq 0.1$ and $\zeta = c_1 \varepsilon^2 / \log^2(1/p)$ for a suitably absolute constant $c_1 > 0$.

Output: “Accept” or “reject”

GSA-TEST($\text{SAMP}(f)$, $\text{MQ}(f)$, $\hat{p}$, ε):

1. Set $p_{\text{lb}} := \hat{p}/(1 + \zeta)$ (as in Equation (5)) and κ, t (as in Equations (4) and (15)) as:

$$\kappa := \frac{\zeta \sqrt{t} \cdot \varphi(\Phi^{-1}(p_{\text{lb}}))}{2\sqrt{\pi} \cdot p_{\text{lb}}},$$

$$t := \frac{c_2 \varepsilon^{10}}{\log^{10}(1/p_{\text{lb}})}.$$

2. Run the algorithm EST-SENSE to compute a value α , which is a $\pm\kappa$ -accurate estimate of $\text{NS}_t(f)/\text{Vol}[f]$ with confidence 0.99.

3. If

$$\alpha \leq \frac{2\sqrt{t} \cdot \varphi(\Phi^{-1}(p_{\text{lb}}))}{\sqrt{\pi} \cdot p_{\text{lb}}} + \kappa$$

then output “accept,” otherwise output “reject.”

Proof of Theorem 2. The algorithm is given a value $\hat{p}$ such that $p := \text{Vol}(f)$ satisfies

$$\frac{p}{1 + \zeta} \leq \hat{p} \leq (1 + \zeta)p, \quad (3)$$

where $\zeta = c_1 \varepsilon^2 / \log^2(1/p)$ for a suitable absolute constant $c_1 > 0$.

The algorithm and its analysis use three main parameters, which are κ , t , and ξ . The precise values of t and ξ (in terms of p and ε) are set later (see (15)), but looking ahead, importantly, both ξ and t are set to be $\text{poly}(\log(1/p), 1/\varepsilon)$. The parameter κ is defined as

$$\kappa := \frac{\zeta \sqrt{t} \cdot \varphi(\Phi^{-1}(p_{\text{lb}}))}{2\sqrt{\pi} \cdot p_{\text{lb}}} \leq \frac{\zeta \sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{2\sqrt{\pi} \cdot p}. \quad (4)$$

using Fact 12 and $p_{\text{lb}} \leq p$. We also define the parameter p_{lb} as

$$p_{\text{lb}} := \hat{p}/(1 + \zeta). \quad (5)$$

Correctness analysis. There are two parts of the analysis establishing correctness. We first start with the easy part, which is completeness.

▷ **Claim 20 (Completeness).** If f is a halfspace with $\text{Vol}(f) = p$ and the input parameter $\hat{p}$ satisfies (3), then f passes the test with probability at least 0.99.

Proof. As f is a halfspace, by Theorem 10 and Fact 14 we have

$$\frac{\text{NS}_t(f)}{\text{Vol}(f)} \leq \frac{2\sqrt{t}}{\sqrt{\pi}} \cdot \frac{\varphi(\Phi^{-1}(p))}{p}.$$

Note that $p_{\text{lb}} \leq p$ and the function $\Lambda(p) := \varphi(\Phi^{-1}(p))/p$ is decreasing in the interval $(0, 1/2)$. Thus,

$$\frac{\text{NS}_t(f)}{\text{Vol}(f)} \leq \frac{2\sqrt{t}}{\sqrt{\pi}} \cdot \frac{\varphi(\Phi^{-1}(p_{\text{lb}}))}{p_{\text{lb}}}.$$

As $\alpha \leq \text{NS}_t(f)/\text{Vol}(f) + \kappa$ with probability 0.99 by Lemma 16, this finishes the proof. ◁

The next claim analyzes the soundness of the test.

▷ **Claim 21 (Soundness).** Suppose $f : \mathbb{R}^n \rightarrow \{0, 1\}$ has $\text{Vol}(f) = p$, and suppose that p and $\hat{p}$ satisfy (3). If $\text{rel-dist}(f, g) \geq \varepsilon$ for every halfspace g , then f fails the test with probability at least 0.99.

Proof. We prove the contrapositive. Suppose f passes the test with probability greater than 0.01. By Line 2 of the algorithm and Lemma 16, it must be the case that

$$\frac{\mathbf{NS}_t(f)}{\text{Vol}(f)} \leq \frac{2\sqrt{t}}{\sqrt{\pi}} \cdot \frac{\varphi(\Phi^{-1}(p_{\text{lb}}))}{p_{\text{lb}}} + 2\kappa.$$

Equivalently (recalling that $\text{Vol}(f) = p$), we get that

$$\mathbf{NS}_t(f) \leq \frac{2\sqrt{t}p}{\sqrt{\pi}} \cdot \frac{\varphi(\Phi^{-1}(p_{\text{lb}}))}{p_{\text{lb}}} + 2\kappa \cdot p.$$

Now, using Fact 12 and $p_{\text{lb}} \leq p \leq (1 + \zeta)p_{\text{lb}}$, we get that

$$\mathbf{NS}_t(f) \leq \frac{2\sqrt{t}p}{\sqrt{\pi}} \cdot \frac{\varphi(\Phi^{-1}(p))}{p} \cdot (1 + \zeta) + 2\kappa \cdot p = \frac{2(1 + \zeta)\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi}} + 2\kappa \cdot p.$$

Plugging in the value of κ from (4) in the above estimate, we get

$$\mathbf{NS}_t(f) \leq \frac{2\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi}} + \frac{3\zeta\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi}} \leq \frac{3\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi}}, \quad (6)$$

where we use the fact that $\zeta \leq 1/3$ when c_1 is sufficiently small. For any parameter $\xi > 0$, we can now apply Theorem 11 to get that there is another function $f_{\text{sm}} : \mathbb{R}^n \rightarrow \{0, 1\}$ satisfying the two bounds Equation (7) and Equation (10) given below: The first bound, using Equation (6), is

$$\text{Vol}(f \triangle f_{\text{sm}}) \leq \frac{\mathbf{NS}_t(f)}{\xi} \leq \frac{3\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi} \cdot \xi}. \quad (7)$$

For the second bound, let us now define τ as

$$\tau := \frac{3\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi} \cdot \xi \cdot p} = \Theta\left(\frac{\sqrt{t \log(1/p)}}{\xi}\right), \quad (8)$$

where we use Fact 14 in the last inequality. (Looking ahead to Equation (15), note that our choice of parameters will ensure that $\tau \leq 1/10$.) By Equation (7), we have that

$$(1 + \tau)p \geq p_{\text{sm}} := \text{Vol}(f_{\text{sm}}) \geq (1 - \tau)p. \quad (9)$$

The second bound given by Theorem 11 is

$$\begin{aligned} \text{surf}(f_{\text{sm}}) &\leq \sqrt{\frac{\pi}{2}} \left(1 + O\left(\frac{\xi}{\sqrt{\log(1/\xi)}}\right)\right) \cdot \frac{1}{\sqrt{e^{2t} - 1}} \cdot \mathbf{NS}_t(f) \\ &\leq \sqrt{\frac{\pi}{2}} \cdot \frac{1}{\sqrt{e^{2t} - 1}} \cdot \left(\frac{2\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi}} + \frac{3\zeta\sqrt{t} \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\pi}}\right) \\ &\quad + \frac{\mathbf{NS}_t(f)}{\sqrt{e^{2t} - 1}} \cdot O\left(\frac{\xi}{\sqrt{\log(1/\xi)}}\right), \end{aligned} \quad (10)$$

61:16 Sublinear-Query Relative-Error Testing of Halfspaces

where we used the first estimate of (6). Plugging in $e^{2t} - 1 \geq 2t$ and the second bound of (6),

$$\text{surf}(f_{\text{sm}}) \leq (1 + 3\zeta) \cdot \varphi(\Phi^{-1}(p)) + \frac{O(\xi) \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\log(1/\xi)}}. \quad (11)$$

Thus, defining the set $E := f_{\text{sm}}^{-1}(1)$, the isoperimetric deficit $D(E)$ for this set E satisfies

$$\begin{aligned} D(E) &= \text{surf}(f_{\text{sm}}) - \varphi(\Phi^{-1}(p_{\text{sm}})) \\ &\leq (1 + 3\zeta) \cdot \varphi(\Phi^{-1}(p)) + \frac{O(\xi) \cdot \varphi(\Phi^{-1}(p))}{\sqrt{\log(1/\xi)}} - \varphi(\Phi^{-1}(p_{\text{sm}})), \end{aligned}$$

where we use (11) to get the inequality. Now, recall that by Equation (9) we have

$$p/p_{\text{sm}} \leq 1/(1 - \tau) \leq 1 + 2\tau$$

using $\tau \leq 1/10$. Combining this with Fact 12, it follows that $\varphi(\Phi^{-1}(p)) \leq (1 + 2\tau)\varphi(\Phi^{-1}(p_{\text{sm}}))$ and

$$D(E) \leq \left(2\tau + 3\zeta(1 + 2\tau) + O\left(\frac{\xi}{\sqrt{\log(1/\xi)}}\right) \right) \cdot \varphi(\Phi^{-1}(p_{\text{sm}})). \quad (12)$$

Now using Corollary 19 with this estimate (along with the fact $\text{Vol}(E) = p_{\text{sm}}$ and $\tau \leq 1/10$), there is a halfspace H such that

$$\text{Vol}(H \triangle E) = O(\beta p_{\text{sm}}) = O(\beta p) \quad (13)$$

where

$$\beta = \log(1/p) \cdot O\left(\sqrt{\tau} + \sqrt{\zeta} + \frac{\sqrt{\xi}}{\log^{1/4}(1/\xi)}\right).$$

Using (8), it follows that

$$\beta = \log(1/p) \cdot O\left(\frac{(t \log(1/p))^{1/4}}{\sqrt{\xi}} + \sqrt{\zeta} + \frac{\sqrt{\xi}}{\log^{1/4}(1/\xi)}\right). \quad (14)$$

We can now use the fact that $\zeta = c_1 \varepsilon^2 / \log^2(1/p)$ for a sufficiently small $c_1 > 0$ and finally set the parameters t and ξ as follows

$$t = \frac{c_2 \varepsilon^{10}}{\log^{10}(1/p_{\text{lb}})} \quad \text{and} \quad \xi = \frac{c_3 \varepsilon^{2.1}}{\log^{2.1}(1/p_{\text{lb}})}, \quad (15)$$

for sufficiently small constants $c_2 \ll c_3 \ll 1$, to get from Equations (13) and (14) that

$$\text{Vol}(H \triangle E) \leq \frac{\varepsilon}{2} \cdot p. \quad (16)$$

Combining Equation (7) with Equation (16), and observing that by Equation (15) we have that (7) is bounded from above by $\varepsilon p/2$ as well (with room to spare), by the triangle inequality we get that $\text{Vol}(f^{-1}(1) \triangle H) \leq \varepsilon p$, i.e. f has relative distance at most ε from the halfspace H . This establishes the contrapositive and concludes the proof of Claim 21. $\triangleleft$

Complexity analysis. We finally note that κ defined in (4) is $\Theta(\varepsilon^7 / \log^{6.5}(1/p))$. As the sample complexity of Lemma 16 scales as $O(\kappa^{-2})$, our final sample complexity is $O(\varepsilon^{-14} \cdot \log^{13}(1/p))$. This concludes the proof of Theorem 2. $\blacktriangleleft$

References

- 1 Maria-Florina Balcan, Eric Blais, Avrim Blum, and Liu Yang. Active Property Testing. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pages 21–30, 2012. doi:10.1109/FOCS.2012.64.
- 2 Marco Barchiesi, Alessio Brancolini, and Vesa Julin. Sharp dimension free quantitative estimates for the Gaussian isoperimetric inequality. *The Annals of Probability*, pages 668–697, 2017.
- 3 Arnab Bhattacharyya and Yuichi Yoshida. *Property Testing - Problems and Techniques*. Springer, 2022. doi:10.1007/978-981-16-8622-1.
- 4 Eric Blais, Renato Ferreira Pinto Jr, and Nathaniel Harms. VC dimension and distribution-free sample-based testing. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 504–517, 2021. doi:10.1145/3406325.3451104.
- 5 H. Block. The Perceptron: a model for brain functioning. *Reviews of Modern Physics*, 34:123–135, 1962.
- 6 C. Borell. The Brunn-Minkowski inequality in Gauss space. *Invent. Math.*, 30:207–216, 1975.
- 7 C. Borell. Geometric bounds on the Ornstein-Uhlenbeck velocity process. *Probability Theory and Related Fields*, 70:1–13, 1985.
- 8 Gautam Chandrasekaran, Adam Klivans, Vasilis Kontonis, Raghu Meka, and Konstantinos Stavropoulos. Smoothed Analysis for Learning Concepts with Low Intrinsic Dimension. In *The Thirty Seventh Annual Conference on Learning Theory (COLT)*, pages 876–922, 2024. URL: <https://proceedings.mlr.press/v247/chandrasekaran24a.html>.
- 9 X. Chen, D. Palit, K. Peshawaria, W. Pires, R. Servedio, and Y. Zhang. Relative-error unateness testing. Available at <https://arxiv.org/abs/2510.21589>, 2025. doi:10.48550/arXiv.2510.21589.
- 10 X. Chen, W. Pires, T. Pitassi, and R. Servedio. DNF formulas are efficiently testable with relative error. Available at <https://arxiv.org/pdf/2601.16076>, 2026.
- 11 Xi Chen, Anindya De, Yizhi Huang, Yuhao Li, Shivam Nadimpalli, Rocco A. Servedio, and Tianqi Yang. Relative-error monotonicity testing. Available at <https://arxiv.org/abs/2410.09235>, to appear in SODA 2025, 2024. doi:10.48550/arXiv.2410.09235.
- 12 Xi Chen, Anindya De, Yizhi Huang, Shivam Nadimpalli, Rocco A. Servedio, and Tianqi Yang. Halfspaces are hard to test with relative error. SODA 2026, to appear, 2026.
- 13 Xi Chen and Shyamal Patel. Distribution-free testing for halfspaces (almost) requires PAC learning. In *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1715–1743. SIAM, 2022. doi:10.1137/1.9781611977073.70.
- 14 Xi Chen, William Pires, Toniann Pitassi, and Rocco A. Servedio. Relative-error testing of conjunctions and decision lists. In *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025*, volume 334 of *LIPIcs*, pages 52:1–52:18, 2025. doi:10.4230/LIPIcs.ICALP.2025.52.
- 15 Xi Chen, William Pires, Toniann Pitassi, and Rocco A. Servedio. Testing juntas and junta subclasses with relative error. In *38th Annual Conference on Learning Theory (COLT)*, 2025.
- 16 Andrea Cianchi, Nicola Fusco, Francesco Maggi, and Aldo Pratelli. On the isoperimetric deficit in Gauss space. *American Journal of Mathematics*, pages 131–186, 2011.
- 17 A. Daniely. Complexity theoretic limitations on learning halfspaces. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016*, pages 105–117, 2016.
- 18 A. De, E. Mossel, and J. Neeman. Majority is stablest: discrete and SoS. In *Proc. 45th Annual ACM Symposium on Theory of Computing (STOC)*, pages 477–486, 2013.
- 19 Anindya De, Huan Li, Shivam Nadimpalli, and Rocco A Servedio. Detecting low-degree truncation. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 1027–1038, 2024. doi:10.1145/3618260.3649633.

- 20 Anindya De, Elchanan Mossel, and Joe Neeman. Is your function low dimensional? In *Conference on Learning Theory*, pages 979–993. PMLR, 2019. URL: <http://proceedings.mlr.press/v99/de19a.html>.
- 21 Anindya De, Elchanan Mossel, and Joe Neeman. Robust testing of low dimensional functions. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 584–597, 2021. doi:10.1145/3406325.3451115.
- 22 Anindya De, Shivam Nadimpalli, and Rocco A Servedio. Testing convex truncation. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4050–4082. SIAM, 2023. doi:10.1137/1.9781611977554.CH155.
- 23 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, 2024.
- 24 Ilias Diakonikolas, Daniel M Kane, and Pasin Manurangsi. The complexity of adversarially robust proper learning of halfspaces with agnostic noise. *Advances in Neural Information Processing Systems*, 33:20449–20461, 2020.
- 25 Irit Dinur, Ehud Friedgut, Guy Kindler, and Ryan O’Donnell. On the Fourier tails of bounded functions over the discrete cube. In *Proc. 38th ACM Symp. on Theory of Computing*, pages 437–446, 2006. doi:10.1145/1132516.1132580.
- 26 Antoine Ehrhard. Éléments extrémaux pour les inégalités de Brunn-Minkowski Gaussiennes. In *Annales de l’IHP Probabilités et statistiques*, volume 22(2), pages 149–168, 1986.
- 27 Ronen Eldan. A two-sided estimate for the Gaussian noise stability deficit. *Inventiones Mathematicae*, 201:561–624, 2015.
- 28 Dana Glasner and Rocco A. Servedio. Distribution-free testing lower bound for basic boolean functions. *Theory of Computing*, 5(1):191–216, 2009. doi:10.4086/toc.2009.v005a010.
- 29 Oded Goldreich. *Introduction to Property Testing*. Cambridge University Press, 2017. doi:10.1017/9781108135252.
- 30 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45:653–750, 1998. doi:10.1145/285055.285060.
- 31 Nathaniel Harms. Halfway to halfspace testing. Master’s thesis, University of Waterloo, 2017.
- 32 Nathaniel Harms. Testing Halfspaces over Rotation-Invariant Distributions. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, pages 694–713. SIAM, 2019. doi:10.1137/1.9781611975482.44.
- 33 A. Kalai, A. Klivans, Y. Mansour, and R. Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008. doi:10.1137/060649057.
- 34 S. Khot, G. Kindler, E. Mossel, and R. O’Donnell. Optimal inapproximability results for Max-Cut and other 2-variable CSPs? *SIAM Journal on Computing*, 37(1):319–357, 2007. doi:10.1137/S0097539705447372.
- 35 M. Ledoux. Semigroup proofs of the isoperimetric inequality in Euclidean and Gauss space. *Bull. Sci. Math.*, 118:485–510, 1994.
- 36 Kevin Matulef, Ryan O’Donnell, Ronitt Rubinfeld, and Rocco A. Servedio. Testing ± 1 -weight halfspace. In *APPROX-RANDOM*, pages 646–657, 2009. doi:10.1007/978-3-642-03685-9_48.
- 37 Kevin Matulef, Ryan O’Donnell, Ronitt Rubinfeld, and Rocco A. Servedio. Testing halfspaces. *SIAM Journal on Computing*, 39(5):2004–2047, 2010. doi:10.1137/070707890.
- 38 E. Mossel and R. O’Donnell. On the noise sensitivity of monotone functions. *Random Structures and Algorithms*, 23(3):333–350, 2003. doi:10.1002/RS.10097.
- 39 E. Mossel, R. O’Donnell, and K. Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. *Annals of Mathematics*, 171:295–341, 2010.
- 40 Elchanan Mossel and Joe Neeman. Robust dimension free isoperimetry in Gaussian space. *Annals of Probability*, 43(3):971–991, 2015.

- 41 Elchanan Mossel and Joe Neeman. Robust optimality of gaussian noise stability. *Journal of the European Mathematical Society*, 17:433–482, 2015.
- 42 Joe Neeman. Testing surface area with arbitrary accuracy. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014*, pages 393–397. ACM, 2014. doi:10.1145/2591796.2591807.
- 43 A. Novikoff. On convergence proofs on perceptrons. In *Proceedings of the Symposium on Mathematical Theory of Automata*, volume XII, pages 615–622, 1962.
- 44 R. O’Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.
- 45 D. Ron. Algorithmic and analysis techniques in property testing. *Foundations and Trends in Theoretical Computer Science*, 5:73–205, 2010.
- 46 V. Sudakov and B. Tsirel’son. Extremal properties of half-spaces for spherically invariant measures. *J. Soviet Math.*, 9:9–18, 1978. Translated from Zap. Nauchn. Sem. Leningrad. Otdel. Math. Inst. Steklova. 41 (1974), 14–21.
- 47 Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*, volume 47. Cambridge University Press, 2018.
- 48 M. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press, 2019.