

Testing submodularity and other properties of valuation functions

Eric Blais¹ and Abhinav Bommireddi²

- 1 David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada
eric.blais@uwaterloo.ca
- 2 David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada
vabommir@uwaterloo.ca

Abstract

We show that for any constant $\epsilon > 0$ and $p \geq 1$, it is possible to distinguish functions $f : \{0, 1\}^n \rightarrow [0, 1]$ that are submodular from those that are ϵ -far from every submodular function in ℓ_p distance with a *constant* number of queries.

More generally, we extend the testing-by-implicit-learning framework of Diakonikolas et al. (2007) to show that every property of real-valued functions that is well-approximated in ℓ_2 distance by a class of k -juntas for some $k = O(1)$ can be tested in the ℓ_p -testing model with a constant number of queries. This result, combined with a recent junta theorem of Feldman and Vondrák (2016), yields the constant-query testability of submodularity. It also yields constant-query testing algorithms for a variety of other natural properties of valuation functions, including fractionally additive (XOS) functions, OXS functions, unit demand functions, coverage functions, and self-bounding functions.

1998 ACM Subject Classification F.2.0 ANALYSIS OF ALGORITHMS AND PROBLEM COMPLEXITY

Keywords and phrases Property testing, Testing by implicit learning, Self-bounding functions

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

1 Introduction

Property testing is concerned with approximate decision problems of the following form: given oracle access to some function $f : \mathcal{X} \rightarrow \mathcal{Y}$ and some fixed property \mathcal{P} of such functions, how many oracle calls (or queries) to f does a bounded-error randomized algorithm need to distinguish the cases where f has the property \mathcal{P} from the case where f is ϵ -far—under some appropriately defined metric—from having the same property? Remarkably, many natural properties of functions can be tested with a number of queries that is *independent* of the size of the function’s domain. For example, for any constant $\epsilon > 0$ and $t \geq 1$, a constant number of queries suffices to test whether a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is linear [7]; a polynomial of degree at most t [28]; a t -junta [5, 18]; a monomial [27]; computable by a Boolean circuit of size t [10]; or a linear threshold function [25].

In this work, we consider the problem of testing properties of bounded *real-valued* functions over the Boolean hypercube. In particular, are there natural examples of such properties that are testable with a constant number of queries? This question is best considered in the ℓ_p testing framework introduced by Berman, Raskhodnikova, and Yaroslavtsev [4]. In this setting, the distance between a function $f : \{0, 1\}^n \rightarrow [0, 1]$ and some property \mathcal{P} of these functions is $\text{dist}_p(f, \mathcal{P}) = \inf_{g \in \mathcal{P}} \|f - g\|_p$.



© Eric Blais and Abhinav Bommireddi;
licensed under Creative Commons License CC-BY

42nd Conference on Very Important Topics (CVIT 2016).

Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:17

Leibniz International Proceedings in Informatics



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

1.1 Testing properties of valuation functions

Natural properties of bounded real-valued Boolean functions have been studied extensively in the context of valuation functions in algorithmic game theory. For a sequence of n goods labeled with the indices $1, \dots, n$, we can encode the value of each subset of these goods to some agent with a function $f : \{0, 1\}^n \rightarrow [0, 1]$ by setting $f(x)$ to be the (possibly normalized) value of the subset $\{i \in [n] : x_i = 1\}$ to the agent. Such a valuation function f is

Additive if there are weights w_1, \dots, w_n such that $f(x) = \sum_{i:x_i=1} w_i$;

a **Coverage function** if there exists a universe U , non-negative weights $\{w_u\}_{u \in U}$, and subsets $A_1, \dots, A_n \subseteq U$ such that $f(x) = \sum_{u \in \bigcup_{i:x_i=1} A_i} w_u$.

Unit demand if there are weights w_1, \dots, w_n such that $f(x) = \max\{w_i : x_i = 1\}$;

OXS if there are $k \geq 1$ unit demand functions g_1, \dots, g_k such that $f(x) = \max\{g_1(x^{(1)}), \dots, g_k(x^{(k)})\}$ where the maximum is taken over all $x^{(1)}, \dots, x^{(k)}$ such that for every $i \in [n]$, $x_i = \sum_{j=1}^k x_i^{(j)}$;

Gross Substitutes if for any $p' \leq p \in \mathbb{R}^n$ and any x, x' that maximize $f(x) - \sum_{i:x_i=1} p_i$ and $f(x') - \sum_{i:x'_i=1} p'_i$, respectively, every $j \in [n]$ for which $x_j = 1$ and $p_j = p'_j$ also satisfies $x'_j = 1$;

Submodular if $f(x) + f(y) \geq f(x \wedge y) + f(x \vee y)$ for every $x, y \in \{0, 1\}^n$, where \wedge and \vee are the bitwise AND and OR operations;

Fractionally subadditive (XOS) iff there are non-negative real valued weights $\{w_{ij}\}_{i,j \leq n}$ such that $f(x) = \max_i \sum_j w_{ij} \cdot x_j$;

Self-bounding if $f(x) \geq \sum_i (f(x) - \min_{x_i} f(x))$, where $\min_{x_i} f(x) = \min\{f(x), f(x \oplus e_i)\}$ and \oplus is the bitwise XOR operator; and

Subadditive if $f(x \cup y) \leq f(x) + f(y)$ for every $x, y \in \{0, 1\}^n$.

Each of these properties enforces some structure on valuation functions, and much work has been devoted to better understanding these structures (and their algorithmic implications) by studying the properties through the lenses of learning theory [2, 3, 15], optimization [13, 14], approximation [16, 17], and sketching [1]. The problem of testing whether an unknown valuation function satisfies one of these properties offers another angle from which we can learn more about the structure imposed on the functions that satisfy these properties.

Indeed, there has already been some recent developments on the study of testing these properties. Notably, Seshadhri and Vondrák [30] initiated the study of testing submodularity for functions over the hypercube and showed that in the setting where we measure the distance to submodularity in terms of Hamming distance (rather than ℓ_p distance), submodularity can be tested with $\epsilon^{-\sqrt{n} \log n}$ queries and that it *cannot* be tested with a number of queries that is independent of n . Subsequently, Feldman and Vondrák [17] showed that in the ℓ_1 testing framework, we can do much better: testing submodularity in this model requires a number of queries that is only logarithmic in n . Our first result shows that, in fact, for any value of $p \geq 1$, it is possible to test submodularity in the ℓ_p setting with a number of queries that is completely *independent* of n .

► **Theorem 1.1.** For any $\epsilon > 0$ and any $p \geq 1$, there is an ϵ -tester for submodularity in the ℓ_p testing model with query complexity $2^{\tilde{O}(1/\epsilon^{\max\{2,p\}})}$.

Another property that has been considered in the (standard Hamming distance) testing model is that of being a coverage function. Chakrabarty and Huang [8] showed that for constant values of $\epsilon > 0$, $O(nm)$ queries suffice to ϵ -test whether a function f is a coverage function on some universe U of size $|U| \leq m$. Note that, unlike in the learning and approximation settings, bounds on the number of queries required to test some property

\mathcal{P} do not imply anything about number of queries required to test properties $\mathcal{P}' \subset \mathcal{P}$, so even though coverage functions are submodular, results on testing submodularity do not imply any bounds on the query complexity for testing coverage functions. Nonetheless, our next result shows that this property—along with most of the other properties of valuation functions listed above—can also be tested with a number of queries that is independent of n .

► **Theorem 1.2.** For any $\epsilon > 0$ and any $p \geq 1$, there are ϵ -testers in the ℓ_p testing model for additive functions, coverage functions, unit demand functions, OXS functions, and gross substitute functions that each have query complexity $2^{\tilde{O}(1/\epsilon^{\max\{2,p\}})}$, and there are ϵ -testers in the ℓ_p testing model for fractional subadditivity and self-bounded functions that have query complexity $2^{2^{\tilde{O}(1/\epsilon^{\max\{2,p\}})}}$.

Theorems 1.1 and 1.2 are both special cases of a general testing result that we obtain by extending the technique of *testing by implicit learning* of Diakonikolas et al. [10]. We describe this general result in more details below.

1.2 Testing real-valued functions by implicit learning

There is a strong connection between property testing and learning theory that goes back to the seminal work of Goldreich, Goldwasser, and Ron [21]. As they first observed, any proper learning algorithm for the class of functions that have some property \mathcal{P} can also be used to test \mathcal{P} : run the learning algorithm, and verify whether the resulting hypothesis function h is close to the tested function f or not. This approach yields good bounds on the number of queries required to test many properties of functions, but, as simple information theory arguments show, it cannot yield query complexity bounds that are smaller than $\log n$ for almost all natural properties of functions over $\{0, 1\}^n$.

Diakonikolas et al. [10] bypassed this limitation for the special case when every function that has some property \mathcal{P} is close to a junta. A function $f : \{0, 1\}^n \rightarrow [0, 1]$ is a k -junta if there is a set $J \subseteq [n]$ of cardinality $|J| \leq k$ such that the value of f on any input x is completely determined by the values x_i for each $i \in J$. Every k -junta f has corresponding “core” functions $f_{\text{core}} : \{0, 1\}^k \rightarrow \{0, 1\}$ that define its value based on the value of the k relevant coordinates of its input. Diakonikolas et al.’s key insight is that for testing properties whose functions are (very) close to juntas, it suffices to learn the core of the input function—without having to identify the identity of the relevant coordinates.

The wide applicability of the testing-by-implicit-learning methodology is due to the fact that for many natural properties of Boolean functions, the functions that have these properties must necessarily be close to juntas under the Hamming distance. The starting point for the current research is a recent breakthrough of Feldman and Vondrák, who showed that a similar junta theorem holds for many properties of real-valued functions when closeness is measured according to ℓ_2 distance.

► **Feldman–Vondrák junta theorem.** Fix any $\epsilon \in (0, \frac{1}{2})$. For every submodular function $f : \{0, 1\}^n \rightarrow [0, 1]$, there exists a submodular function $g : \{0, 1\}^n \rightarrow [0, 1]$ that is a $O(\frac{1}{\epsilon^2} \log \frac{1}{\epsilon})$ -junta such that $\|f - g\|_2 \leq \epsilon$. Furthermore, for every self-bounding function $f : \{0, 1\}^n \rightarrow [0, 1]$, there exists a self-bounding function $g : \{0, 1\}^n \rightarrow [0, 1]$ that is a $2^{O(\frac{1}{\epsilon^2})}$ -junta such that $\|f - g\|_2 \leq \epsilon$.

The logarithmic dependence on n for the problem of testing submodularity in the ℓ_1 testing model [17] follows directly from Feldman and Vondrák’s junta theorem and the (standard) testing-by-proper-learning connection. This junta theorem also suggests a natural approach for obtaining a constant query complexity for the same problem by combining it

with a testing-by-implicit-learning algorithm. In order to implement this approach, however, new testing-by-implicit-learning techniques are required to overcome two obstacles.

The first obstacle is that most existing testing-by-implicit-learning algorithms [9–11, 22] are designed for properties that contain functions which are close to juntas in Hamming distance, not ℓ_p distance. This is a stronger condition, and enables the analysis of these algorithms to assume that with large probability, when f is very close to a k -junta f' , the queries x made by the algorithm all satisfy $f(x) = f'(x)$. In the ℓ_p distance model, however, we can have a function f that is extremely close to a k -junta but still has $f(x) \neq f'(x)$ for many (or even every!) input x .

The second (related) obstacle that we encounter when considering submodular functions is that current testing-by-implicit-learning algorithms only work in the regime where the functions in \mathcal{P} are ϵ -close to k -juntas for some $k < \epsilon^{-1/2}$. (See for example the discussion in §2.5 of [29].) This condition is satisfied by the properties of Boolean functions that have been studied previously, but the bounds in the Feldman–Vondrák junta theorem, however, do not satisfy this requirement.

We give a new algorithm for testing-by-implicit-learning that overcomes both of these obstacles. As a result, we obtain the following general theorem.

► **Theorem 1.3.** For any $0 < \epsilon < \frac{1}{2}$ and any property \mathcal{P} of functions mapping $\{0, 1\}^n \rightarrow [0, 1]$, if $k \geq 1$ is such that for every function $f \in \mathcal{P}$, there is a k -junta h that satisfies $\|f - h\|_2 \leq \frac{\epsilon}{10^6}$, then there is an ϵ -tester for \mathcal{P} in the ℓ_2 testing model with query complexity $\frac{2^{O(k \log k)}}{\epsilon^{10}}$.

Theorems 1.1 and 1.2 are both obtained directly from Theorem 1.3, the Feldman–Vondrák junta theorem and Fact 2.1.

► **Remark.** Another testing-by-implicit-learning algorithm, the Implicit Sieve algorithm of Wimmer and Yoshida [31] based on the Kushilevitz–Mansour learning algorithm [23], was introduced to test properties in the Hamming model but has a natural analogue in the ℓ_2 testing model as well. It would be quite interesting to determine what properties can be tested efficiently in the ℓ_p models by this algorithm (or extensions of it).

1.3 Overview of the proofs

1.3.1 The algorithm.

The current testing-by-implicit-learning algorithms proceed in two main stages. In the first stage, the coordinates in $[n]$ are randomly partitioned into $\text{poly}(k)$ parts, and an influence test is used to identify the (at most k) parts that contain relevant variables of an unknown input function f that is very close to being a k -junta. In the second stage, inputs $x \in [n]$ are drawn at random according to some distribution, the value $f(x)$ is observed, and the value of the relevant coordinate in each of the parts identified in the second stage is determined using more calls to the influence test.

The IMPLICIT LEARNING TESTER algorithm that we introduce in this paper reverses the order of the two main stages. In the first stage, it draws a sequence of q queries $X = (x^{(1)}, \dots, x^{(q)})$ at random and queries the value of f on each of these queries. It also uses X to partition the coordinates in $[n]$ into 2^q random parts according to the values of the coordinate on the q queries. In the second stage, the algorithm then uses an influence estimator to identify the k parts that contain the relevant coordinates of a k -junta that is close to f and, since all the coordinates in a common part have the same value on each of the q queries, learn the value of the k relevant coordinates on each of these initial queries. The algorithm then checks whether the core function thus learned is consistent with those of functions in the property being tested.

The main advantage of the IMPLICIT LEARNING TESTER algorithm is that its analysis does not require the assumption that our samples are exactly consistent with those of an actual k -junta (instead of those of a function that is only promised to be *close* to a k -junta). This feature enables us to overcome the obstacles listed in the previous section, at the cost of adding a few complications to the analysis, as described below.

1.3.2 The analysis.

There are two main technical ingredients in the analysis of the algorithm. The first, established in Lemma 3, is used to show that when f is close to a k -junta in ℓ_2 distance, the search procedure identifies parts that contain the k relevant coordinates of some k -junta that is close to f . (Note that the search is not guaranteed to find the parts that contain the relevant coordinates of the k -junta that is *closest* to f , but it suffices to find those of any close k -junta.)

The second technical ingredient addresses the fact that by drawing the q samples $x^{(1)}, \dots, x^{(q)}$ first and then using these samples to provide the initial partition of the coordinates in $[n]$, we no longer will obtain uniformly random samples of the core f_{core} of the input function f . Nonetheless, in Lemma 4, we show that when f is close to a k -junta, the distribution of these samples on the core function still enables us to accurately estimate the distance of f_{core} to the core functions of any other k -junta.

1.4 Discussion and open problems

Theorems 1.1–1.3 raise a number of intriguing questions. The most obvious question left open is whether we can also test subadditivity of real-valued functions with a constant number of queries: subadditive functions need not be close to juntas, so such a result would appear to require a different technique.

It is also useful to compare our bounds for submodularity testing with those for testing monotonicity: in the Hamming distance testing model, Seshadhri and Vondrák [30] showed that the query complexity for testing submodularity is at least as large as that for testing monotonicity. However, the best current bounds for testing monotonicity in the ℓ_p testing model have a linear dependence on n [4]. Is it also possible to test monotonicity with a constant number of queries? Or is it the case that testing submodularity is strictly easier than testing monotonicity in the ℓ_p testing setting?

2 Preliminaries

Let \mathcal{F}_n denote the set of functions mapping $\{0, 1\}^n$ to $[0, 1]$. For any $f \in \mathcal{F}_n$ and $S \subseteq [n]$ with complement $\bar{S} = [n] \setminus S$, when $x \in \{0, 1\}^S$ and $y \in \{0, 1\}^{\bar{S}}$, we write $f(x, y)$ to denote the value $f(z)$ for the input z that satisfies $z_i = x_i$ for each $i \in S$ and $z_i = y_i$ otherwise.

We use the standard definitions and notation for the Fourier analysis of functions $f : \{0, 1\}^n \rightarrow [0, 1]$. For a complete introduction to the topic, see [26]. Throughout the paper, unless otherwise specified all probabilities and expectations are over the uniform distribution on the random variable's domain.

2.1 Property testing

A *property* \mathcal{P} of functions in \mathcal{F}_n is a subset of these functions that is invariant under relabeling of the n coordinates. The *Hamming distance* between $f, g \in \mathcal{F}_n$ is $\text{dist}_{\text{Ham}}(f, g) =$

$\Pr_x[f(x) \neq g(x)]$ and the Hamming distance between f and a property \mathcal{P} is $\text{dist}_{\text{Ham}}(f, \mathcal{P}) = \inf_{g \in \mathcal{P}} \text{dist}_{\text{Ham}}(f, g)$. For $p \geq 1$, the ℓ_p distance between f and g is $\text{dist}_p(f, g) = \|f - g\|_p = (\mathbb{E}_x[|f(x) - g(x)|^p])^{1/p}$ and the ℓ_p distance between f and \mathcal{P} is $\text{dist}_p(f, \mathcal{P}) = \inf_{g \in \mathcal{P}} \text{dist}_p(f, g)$.

Given $\epsilon > 0$, An ϵ -tester in the Hamming testing model (resp., ℓ_p testing model) for some property $\mathcal{P} \subseteq \mathcal{F}_n$ is a randomized algorithm that (i) accepts every function $f \in \mathcal{P}$ with probability at least $\frac{2}{3}$ and (ii) rejects every function f that satisfies $\text{dist}_{\text{Ham}}(f, \mathcal{P}) \geq \epsilon$ (resp., $\text{dist}_p(f, \mathcal{P}) \geq \epsilon$) with probability at least $\frac{2}{3}$. An ϵ -tester for \mathcal{P} is an (ϵ', ϵ) -tolerant tester, for some $\epsilon' < \epsilon$ if it additionally accepts every function f that satisfies $\text{dist}_{\text{Ham}}(f, \mathcal{P}) \leq \epsilon'$ (resp., $\text{dist}_p(f, \mathcal{P}) \leq \epsilon$) with probability at least $\frac{2}{3}$.

Our proofs of Theorems 1.1–1.3 are established in the ℓ_2 testing model. The result for general ℓ_p testing models is obtained from the following elementary relation between the query complexities of testing any property in different ℓ_p testing models.

► **Fact 2.1** (c.f. Fact 5.2 in [4]). For any $\mathcal{P} \subseteq \mathcal{F}_n$, any $\epsilon > 0$, and any $p \geq 1$, the number $Q_p(\mathcal{P}, \epsilon)$ of queries required to ϵ -test \mathcal{P} in the ℓ_p testing model satisfies $Q_2(\mathcal{P}, \epsilon) \leq Q_p(\mathcal{P}, \epsilon) \leq Q_2(\mathcal{P}, \epsilon^{\frac{p}{2}})$.

Theorem 1.2 also relies on the following hierarchy of properties. (See, e.g., [24].)

► **Lemma 1.** *The properties of \mathcal{F}_n defined in the introduction satisfy the inclusion hierarchy*

Additive \subseteq Coverage \subseteq Unit demand \subseteq OXS \subseteq Gross substitute \subseteq Submodularity \subseteq XOS \subseteq Self-bounding.

2.2 Juntas

The function $f : \{0, 1\}^n \rightarrow [0, 1]$ is a *junta on the set $J \subseteq [n]$* if for every $x, y \in \{0, 1\}^n$ that satisfy $x_i = y_i$ for every $i \in J$, we have $f(x) = f(y)$. The function f is a *k-junta* if it is a junta on some set $J \subseteq [n]$ of cardinality $|J| \leq k$. The function $f_{\text{core}} : \{0, 1\}^k \rightarrow [0, 1]$ is a *core function* of the k -junta $f : \{0, 1\}^n \rightarrow [0, 1]$ if there is a projection $\psi : \{0, 1\}^n \rightarrow \{0, 1\}^k$ defined by setting $\psi(x) = (x_{i_1}, \dots, x_{i_k})$ for some distinct $i_1, \dots, i_k \in [n]$ such that for every $x \in \{0, 1\}^n$, $f(x) = f_{\text{core}}(\psi(x))$.

► **Definition 2.2.** For any function $f : \{0, 1\}^n \rightarrow [0, 1]$ and set $J \subseteq [n]$, the *J-junta projection* of f is the function $f_J : \{0, 1\}^J \rightarrow [0, 1]$ defined by setting $f_J(x) = \mathbb{E}_{y \in \{0, 1\}^{\bar{J}}} [f(x, y)]$ for every $x \in \{0, 1\}^J$.

A basic fact that we will require is that f_J is the J -junta that is closest to f under the ℓ_2 metric.

► **Proposition 2.3.** For every $f : \{0, 1\}^n \rightarrow [0, 1]$ and $J \subseteq [n]$, if $g : \{0, 1\}^n \rightarrow [0, 1]$ is a J -junta, then $\text{dist}_2(f, f_J) \leq \text{dist}_2(f, g)$.

Proof. By applying the identity $\|f - g\|_2^2 = \|f - f_J + f_J - g\|_2^2$ and by expanding the right-hand side, we obtain

$$\begin{aligned} \|f - g\|_2^2 &= \mathbb{E}_{x \in \{0, 1\}^J} \left[\mathbb{E}_{y \in \{0, 1\}^{\bar{J}}} \left[(f(x, y) - f_J(x, y) + f_J(x, y) - g(x, y))^2 \right] \right] \\ &= \|f - f_J\|_2^2 + \|f_J - g\|_2^2 + 2 \mathbb{E}_x \left[\mathbb{E}_y \left[(f(x, y) - f_J(x, y))(f_J(x, y) - g(x, y)) \right] \right]. \end{aligned}$$

Since $f_J - g$ is a J -junta, it does not depend on y and, by the definition of f_J , the last term equals 0. Therefore, $\|f - g\|_2^2 = \|f - f_J\|_2^2 + \|f_J - g\|_2^2$ and the claim follows. ◀

The property $\mathcal{P} \subseteq \mathcal{F}_n$ is a *property of k -juntas* if every function $f \in \mathcal{P}$ is a k -junta. The *core property* of a property \mathcal{P} of k -juntas is the property $\mathcal{P}_{\text{core}} \subseteq \mathcal{F}_k$ defined by $\mathcal{P}_{\text{core}} = \{f_{\text{core}} : f \in \mathcal{P}\}$. For any $\gamma > 0$, the γ -*discretized approximation* of a function $f \in \mathcal{F}_n$ is the function $f^{(\gamma)}$ obtained by rounding the value $f(x)$ for each $x \in \{0, 1\}^n$ to the nearest multiple of γ . The γ -discretized approximation of a property \mathcal{P} is the property $\mathcal{P}^{(\gamma)} = \{f^{(\gamma)} : f \in \mathcal{P}\}$.

2.3 Influence

The notion of influence of coordinates in functions over the Boolean hypercube plays a central role in both our algorithm and its analysis. Informally, the influence of a set of coordinate measures how much re-randomizing these coordinates affects the value of the function. This notion is made precise as follows.

► **Definition 2.4.** The *influence* of a set $S \subseteq [n]$ of coordinates in the function $f : \{0, 1\}^n \rightarrow [0, 1]$ is

$$\text{Inf}_f(S) := \mathbb{E}_{x \in \{0, 1\}^{\bar{S}}} \left[\text{Var}_{y \in \{0, 1\}^S} f(x, y) \right] = \frac{1}{2} \mathbb{E}_{x \in \{0, 1\}^{\bar{S}}} \left[\mathbb{E}_{y, y' \in \{0, 1\}^S} \left[(f(x, y) - f(x, y'))^2 \right] \right].$$

Our proofs make use of a few standard facts regarding the influence of sets of coordinates in f .

- **Fact 2.5.** The influence of $S \subseteq [n]$ in $f \in \mathcal{F}_n$ is $\text{Inf}_f(S) = \sum_{T: T \cap S \neq \emptyset} \hat{f}^2(T)$.
- **Fact 2.6.** For every $f \in \mathcal{F}_n$ and $S, T \subseteq [n]$, we have $\text{Inf}_f(S) \leq \text{Inf}_f(S \cup T) \leq \text{Inf}_f(S) + \text{Inf}_f(T)$.
- **Fact 2.7.** For every $f \in \mathcal{F}_n$ and $J \subseteq [n]$, we have $\text{Inf}_f(\bar{J}) = \text{dist}_2(f, f_J)^2$.
- **Proposition 2.8.** Fix $\epsilon > 0$, and let $f, g : \{0, 1\}^n \rightarrow [0, 1]$ satisfy $\text{dist}_2(f, g) \leq \epsilon$. Then for any set $S \subseteq [n]$, $|\text{Inf}_f(S)^{\frac{1}{2}} - \text{Inf}_g(S)^{\frac{1}{2}}| \leq \epsilon$.

Proof. By Fact 2.7, we have $\text{Inf}_f(S)^{\frac{1}{2}} = \|f - f_{\bar{S}}\|_2$ and $\text{Inf}_g(S)^{\frac{1}{2}} = \|g - g_{\bar{S}}\|_2$. By Proposition 2.3, we also have that $\|f - f_{\bar{S}}\|_2 \leq \|f - g_{\bar{S}}\|_2$. Combining these observations with the triangle inequality, we obtain $\text{Inf}_f(S)^{\frac{1}{2}} - \text{Inf}_g(S)^{\frac{1}{2}} = \|f - f_{\bar{S}}\|_2 - \|g - g_{\bar{S}}\|_2 \leq \|f - g_{\bar{S}}\|_2 - \|g - g_{\bar{S}}\|_2 \leq \|f - g\|_2 \leq \epsilon$. Hence $\text{Inf}_f(S)^{\frac{1}{2}} - \text{Inf}_g(S)^{\frac{1}{2}} \leq \epsilon$ and, similarly, $\text{Inf}_g(S)^{\frac{1}{2}} - \text{Inf}_f(S)^{\frac{1}{2}} \leq \epsilon$ as well. ◀

► **Proposition 2.9.** There is an algorithm ESTIMATEINF such that for every $f : \{0, 1\}^n \rightarrow [0, 1]$, $S \subseteq [n]$, $m \geq 1$, and $t \geq 0$, it makes m queries to f and returns an estimate of the influence of S in f that satisfies

$$\Pr \left[|\text{Inf}_f(S) - \text{ESTIMATEINF}(f, S, m)| \geq t \right] \leq 2e^{-2mt^2}.$$

We also use the following key lemma from [6].

► **Lemma 2** (Lemma 2.3 in [6]). *Let $f : \{0, 1\}^n \rightarrow [0, 1]$ be a function that is ϵ -far from k -juntas and P be a random partition of $[n]$ into $r > 20k^2$ parts. Then with probability at least $\frac{5}{6}$, $\text{Inf}_f(\bar{J}) \geq \frac{\epsilon^2}{4}$ for any union J of k parts from P .*

For the reader's convenience, we include the proof of Lemma 2 in Appendix A; though the original lemma in [6] was only for Boolean-valued functions, the proof remains essentially unchanged.

Algorithm 1: IMPLICIT LEARNING TESTER(\mathcal{F}, k, ϵ)

Data: $q = \frac{2^{O(k)}}{\epsilon^5}$, $m = O(\frac{k^6}{\epsilon^5})$, $r = \log \frac{2^k}{100k^4}$

- 1 Draw $x^{(1)}, \dots, x^{(q)} \in \{0, 1\}^n$ independently and uniformly at random;
- 2 For each $c \in \{0, 1\}^q$, define $S_c \leftarrow \{i \in [n] : (x_i^{(1)}, \dots, x_i^{(q)}) = c\}$;
- 3 Let P_1, \dots, P_{100k^4} be a random equi-partition of $\{0, 1\}^q$;
- 4 **for** each $J \subseteq [100k^4]$ of size $|J| = k$ **do**
- 5 $S_J \leftarrow \bigcup_{j \in J} \bigcup_{c \in P_j} S_c$;
- 6 $\eta_J \leftarrow \text{ESTIMATEINF}(f, [n] \setminus S_J, m)$;
- 7 $\{j_1^*, \dots, j_k^*\} \leftarrow \text{argmin}_J \eta_J$;
- 8 $(P_{0,1}, \dots, P_{0,k}) \leftarrow (P_{j_1^*}, \dots, P_{j_k^*})$;
- 9 **for** $\ell = 1, \dots, r$ **do**
- 10 Let $P_{\ell,i,0}, P_{\ell,i,1}$ be a random equi-partition of $P_{\ell-1,i}$ for each $i \leq k$;
- 11 **for** every $z \in \{0, 1\}^k$ **do**
- 12 $S_z \leftarrow \bigcup_{i \leq k} \bigcup_{c \in P_{\ell,i,z_i}} S_c$;
- 13 $\eta_z \leftarrow \text{ESTIMATEINF}(f, [n] \setminus S_z, m)$;
- 14 $z_\ell^* \leftarrow \text{argmin}_z \eta_z$;
- 15 For each $i \leq k$, update $P_{\ell,i} \leftarrow P_{\ell,i,z_\ell^*}$;
- 16 Let $B = \{b_1, \dots, b_k\} \leftarrow \bigcup_{i \leq k} P_{r,i}$;
- 17 If $\text{ESTIMATEINF}(f, [n] \setminus S_B, m) > \epsilon^2/1000$, reject;
- 18 Let $\phi : \{0, 1\}^n \rightarrow \{0, 1\}^k$ be any projection that satisfies $\phi(x)_i \in S_{b_i}$ for each $i \leq k$;
- 19 **for** $h \in \mathcal{F}_{\text{core}}^{(\frac{\epsilon}{1000})}$ **do**
- 20 If $\frac{1}{q} \sum_{i=1}^q (f(x^{(i)}) - h(\phi(x^{(i)})))^2 \leq 0.35\epsilon$, accept and return h ;
- 21 Reject;

3 Testing by implicit learning

The proof of Theorem 1.3 is established by analyzing the IMPLICIT LEARNING TESTER algorithm.

3.1 Proof of Theorem 1.3

The analysis of the IMPLICIT LEARNING TESTER relies on two technical lemmas. The first shows that when the input function f is close to a k -junta, then with reasonably large probability, the function f is close to a junta on the set B of k parts that is identified by the algorithm.

► **Lemma 3.** *For any $\epsilon > 0$, if the function $f : \{0, 1\}^n \rightarrow [0, 1]$ is ϵ -close to a k -junta and every call to ESTIMATEINF returns an influence estimate with additive error at most $\frac{\epsilon^2}{100k^2}$, then the set B obtained by the JUNTA-PROPERTY TESTER satisfies $\Pr[\text{Inf}_f([n] \setminus S_B) > 100\epsilon^2] \leq \frac{1}{20}$.*

The second lemma shows that the estimate in Step 20 provides a good estimate of the distance between f and the functions in \mathcal{P} .

► **Lemma 4.** *Fix $\epsilon > 0$. Let $f : \{0, 1\}^n \rightarrow [0, 1]$ be a function that satisfies $\text{dist}_2(f, g) \leq \epsilon$ for some function g that is a junta on $J \subseteq [n]$, $|J| \leq k$. Then for every $h_{\text{core}} \in \mathcal{F}_{\text{core}}^{(\frac{\epsilon}{1000})}$, the*

mapping $\psi : \{0, 1\}^n \rightarrow \{0, 1\}^k$ defined in the IMPLICIT LEARNING TESTER and the function $h = h_{\text{core}} \circ \psi$ satisfy

$$\left| \left(\frac{1}{q} \sum_i^q (f(x^{(i)}) - h(x^{(i)}))^2 \right)^{\frac{1}{2}} - \text{dist}_2(g, h) \right| \leq 3\epsilon$$

except with probability at most $2e^{-16q\epsilon^4} + \frac{5k^2}{2^q}$.

The proofs of these lemmas are presented in Sections 3.2 and 3.3. We now show how they are used to complete the proof of Theorem 1.3.

As a first observation, we note that by Hoeffding's inequality and the union bound, all of the calls to ESTIMATEINF have additive error at most $\frac{\epsilon^2}{10^6 k^2}$ except with probability at most $\frac{1}{6}$. In the following, we assume that this condition holds and show how, when it does, the algorithm correctly accepts or rejects with probability with probability at least $\frac{5}{6}$.

► **Claim 3.1 (Completeness).** When f is $\frac{\epsilon}{10^6}$ -close to the property \mathcal{F} of k -juntas, the IMPLICIT LEARNING TESTER accepts with probability at least $\frac{5}{6}$.

Proof. First, by Lemma 3, the probability that f is rejected on step 17 is at most $\frac{1}{18}$. In the rest of the proof, we will show that except with probability at most $\frac{1}{9}$, there is a function $h_{\text{core}} \in \mathcal{F}_{\text{core}}^{(\frac{\epsilon}{10^6})}$ for which the algorithm accepts on line 20.

Let $g \in \mathcal{F}$ be a function that satisfies $\text{dist}_2(f, g) \leq \frac{\epsilon}{10^6}$. Without loss of generality, we can assume that g is a junta on $[k]$. Let $J = [k] \cap S_B$ be the set of the junta variables of g that are contained in the final parts selected by the algorithm. Again without loss of generality (by relabeling the input variables once again if necessary), we can assume that $J = [j]$ for some $j \leq k$, and $i \in S_{b_i}$, for $i \leq j$.

Define $\psi : \{0, 1\}^n \rightarrow \{0, 1\}^k$ to be the mapping defined by $\psi(x) = (x_1, \dots, x_j, x_{i_1}, \dots, x_{i_{k-j}})$ where $i_1, \dots, i_{k-j} \in [n] \setminus [k]$ are representative coordinates from the remaining parts $b \in B$ for which $P_b \cap [k] = \emptyset$.

Let $g_{\text{core}} \in \mathcal{F}_{\text{core}}$ be the core of g corresponding to the projection $\psi(x) = (x_1, \dots, x_k)$, and let $h_{\text{core}} \in \mathcal{F}_{\text{core}}^{(\frac{\epsilon}{10^6})}$ be the discretized approximation to g_{core} . Define $h = h_{\text{core}} \circ \psi$. By our choice of g , we have $\text{dist}_2(f, g) \leq \frac{\epsilon}{10^6}$. In order to invoke Lemma 4, we now want to bound $\text{dist}_2(g, h)$.

Let $h^* \in \mathcal{F}^{(\frac{\epsilon}{10^6})}$, be the discretized approximation of g . Then $\text{dist}_2(g, h^*) \leq \frac{\epsilon}{10^6}$ and the triangle inequality implies that

$$\text{dist}_2(f, h^*) \leq \text{dist}_2(f, g) + \text{dist}_2(g, h^*) \leq \frac{2\epsilon}{10^6}$$

and that

$$\text{dist}_2(g, h) \leq \text{dist}_2(g, h^*) + \text{dist}_2(h^*, h) \leq \text{dist}_2(h^*, h) + \frac{\epsilon}{10^6}.$$

Furthermore, since $h_{\text{core}} = h_{\text{core}}^*$,

$$\begin{aligned} \text{dist}_2(h^*, h) &= \mathbb{E}_x \left[\left(h_{\text{core}}^*(x_1, \dots, x_k) - h_{\text{core}}^*(x_1, \dots, x_j, x_{i_1}, \dots, x_{i_{k-j}}) \right)^2 \right]^{\frac{1}{2}} \\ &= 2 \text{Inf}_{h_{\text{core}}^*}([k] \setminus [j])^{\frac{1}{2}} = 2 \text{Inf}_{h^*}([n] \setminus [j])^{\frac{1}{2}}. \end{aligned}$$

By Proposition 2.8 and Lemma 3, except with probability at most $\frac{1}{18}$,

$$\text{Inf}_{h^*}([n] \setminus [j])^{\frac{1}{2}} \leq \text{Inf}_f([n] \setminus [j])^{\frac{1}{2}} + \text{dist}_2(f, h^*) \leq \text{Inf}_f([n] \setminus S_B)^{\frac{1}{2}} + \frac{2\epsilon}{10^6} \leq \frac{12\epsilon}{10^6}$$

and the distance between g and h is bounded by $\text{dist}_2(g, h) \leq \frac{13}{10^6} \epsilon$. When this bound holds, by Lemma 4 with $\epsilon = \frac{\epsilon}{100}$, the algorithm accepts f for this h except with probability at most $\frac{1}{18}$. ◀

23:10 Testing submodularity and other properties of valuation functions

► **Claim 3.2 (Soundness I).** If f is $\frac{\epsilon}{100}$ -far from being a k -junta, then the IMPLICIT LEARNING TESTER rejects with probability at least $\frac{5}{6}$.

Proof. The initial partition $S_{P_1}, \dots, S_{P_{100k^4}}$ is a random partition of $[n]$ with more than $20k^2$ parts so, by Lemma 2, with probability at least $\frac{5}{6}$, for any union $J \subseteq [n]$ of at most k of these parts we have $\text{Inf}_f([n] \setminus J) \geq \frac{\epsilon^2}{400}$. When this is the case, the inclusion $S_B \subseteq \overline{L_0}$ and the fact that L_0 is the complement of the union of some set of k parts in the random partition imply that

$$\text{Inf}_f([n] \setminus S_B) \geq \text{Inf}_f(L_0) \geq \frac{\epsilon^2}{400}$$

and, under the assumed accuracy of ESTIMATEINF calls, the algorithm rejects f in Step 17. ◀

► **Claim 3.3 (Soundness II).** If f is $\frac{\epsilon}{100}$ -close to a k -junta, but is $\frac{99\epsilon}{100}$ -far from \mathcal{F} , then the IMPLICIT LEARNING TESTER rejects with probability at least $\frac{5}{6}$.

Proof. Let g be any k -junta that satisfies $\text{dist}_2(f, g) \leq \frac{\epsilon}{100}$. For any $h_{\text{core}} \in \mathcal{F}_{\text{core}}^{(\frac{\epsilon}{1000})}$ and any injective mapping $\psi : \{0, 1\}^n \rightarrow \{0, 1\}^k$, the function $h = h_{\text{core}} \circ \psi$ is in $\mathcal{F}^{(\frac{\epsilon}{1000})}$ and so by the triangle inequality,

$$\text{dist}_2(f, \mathcal{F}^{(\frac{\epsilon}{1000})}) \geq \text{dist}_2(f, \mathcal{F}) - \frac{\epsilon}{1000}$$

and

$$\text{dist}_2(g, h) \geq \text{dist}_2(f, h) - \text{dist}_2(f, g) \geq \frac{99}{100}\epsilon - \frac{\epsilon}{1000} - \frac{\epsilon}{100} \geq \frac{97}{100}\epsilon.$$

Then, by Proposition 2.8 and the union bound over all $|\mathcal{F}_{\text{core}}^{(\frac{\epsilon}{1000})}| \leq (1000/\epsilon)^{2^k}$ functions in $\mathcal{F}_{\text{core}}^{(\frac{\epsilon}{1000})}$, with probability at least $\frac{5}{6}$, the condition in Step 20 is never satisfied and the algorithm rejects. ◀

To complete the proof of Theorem 1.3 in the case where $p = 2$, consider now any property \mathcal{P} that contains only functions which are $\frac{\epsilon}{10^6}$ -close to some k -junta. Let \mathcal{F} be the property that includes all k -juntas that are $\frac{\epsilon}{10^6}$ -close to \mathcal{P} . Claim 3.1 shows that IMPLICIT LEARNING TESTER accepts every function in \mathcal{P} with the desired probability, and Claims 3.2 and 3.3 shows that it rejects all functions that are ϵ -far from \mathcal{P} . Finally, we note that the query complexity of the algorithm is at most $q + 2m(2^{O(k \log(k))} + 2^k q) = \frac{2^{O(k \log(k))}}{\epsilon^{10}} + 2^k q$, as claimed. Finally, the general result for ℓ_p testing when $p \neq 2$ follows from Fact 2.1.

3.2 Proof of Lemma 3

Let f be any function ϵ -close to a k -junta and assume without loss of generality (by relabeling the input variables if necessary) that f is close to a junta on $[k]$. The definition of P_1, \dots, P_{100k^4} in step 3, means that $S_{P_1}, \dots, S_{P_{100k^4}}$ is a random partition of $[n]$. So by the union bound, the probability that any two of the coordinates in $[k]$ land in the same part is at most $\frac{1}{100k^2}$.

For each $\ell = 0, 1, 2, \dots, r$, let $L_\ell = [n] \setminus \bigcup_{i=1}^k S_{P_{\ell,i}}$ denote the set of variables that have been “eliminated” after ℓ iterations of the loop. Then $[n] \setminus S_B = L_r$ and

$$\text{Inf}_f([n] \setminus S_B) = \text{Inf}_f(L_0) + \sum_{\ell=1}^r \left(\text{Inf}_f(L_\ell) - \text{Inf}_f(L_{\ell-1}) \right). \quad (1)$$

We bound both terms on the right-hand side of the expression separately.

By Proposition 2.8, we have $\text{Inf}_f([n] \setminus [k]) \leq \varepsilon^2$ and so by the monotonicity of influence there is a choice of $J \subseteq [k^2]$ of size $|J| \leq k$ for which $\text{Inf}_f([n] \setminus S_J) \leq \varepsilon^2$. The guaranteed accuracy on ESTIMATEINF then implies that

$$\text{Inf}_f(L_0) \leq (1 + \frac{2}{100k^2})\varepsilon^2. \quad (2)$$

Define $\mathcal{E} = \{\ell \leq r : (L_\ell \setminus L_{\ell-1}) \cap [k] \neq \emptyset\}$ to be the set of rounds for which the algorithm eliminated at least one of the coordinates in $[k]$. By this definition, each $\ell \in [r] \setminus \mathcal{E}$ satisfies $(L_\ell \setminus L_{\ell-1}) \cap [k] = \emptyset$ and

$$\begin{aligned} \sum_{\ell \in [r] \setminus \mathcal{E}} \text{Inf}_f(L_\ell) - \text{Inf}_f(L_{\ell-1}) &= \sum_{\ell \in [r] \setminus \mathcal{E}} \sum_{T: T \cap L_\ell \neq \emptyset \wedge T \cap L_{\ell-1} = \emptyset} \hat{f}(T)^2 \\ &\leq \sum_{T \subseteq [n] \setminus [k]} \hat{f}(T)^2 \leq \text{Inf}_f([n] \setminus [k]) \leq \varepsilon^2. \end{aligned} \quad (3)$$

For each $\ell \in \mathcal{E}$, define $X_\ell = \{\cup_{i=1}^k S_{P_{\ell,i,1-(z_\ell^*)_i}} : S_{P_{\ell,i,1-(z_\ell^*)_i}} \cap [k] \neq \emptyset\}$ to be the set of coordinates in the parts that contain a coordinate in $[k]$ that was eliminated in the ℓ th iteration of the loop. Let also $Y_\ell = \{\cup_{i=1}^k S_{P_{\ell,i,(z_\ell^*)_i}} : S_{P_{\ell,i,1-(z_\ell^*)_i}} \cap [k] \neq \emptyset\}$ be the coordinates in the parts that were kept instead. Then the guaranteed accuracy of ESTIMATEINF and the choice of z_ℓ^* implies that

$$\text{Inf}_f(L_\ell) \leq \text{Inf}_f((L_\ell \setminus X_\ell) \cup Y_\ell) + 2 \frac{\varepsilon^2}{100k^2}$$

and, therefore,

$$\begin{aligned} \sum_{\ell \in \mathcal{E}} \text{Inf}_f(L_\ell) - \text{Inf}_f(L_{\ell-1}) &\leq \frac{2\varepsilon^2}{1000k} + \sum_{\ell \in \mathcal{E}} \text{Inf}_f((L_\ell \setminus X_\ell) \cup Y_\ell) - \text{Inf}_f(L_{\ell-1}) \\ &\leq \frac{2\varepsilon^2}{1000k} + \sum_{\ell \in \mathcal{E}} \sum_{T: T \cap (L_\ell \setminus X_\ell) \neq \emptyset \wedge T \cap L_{\ell-1} = \emptyset} \hat{f}(T)^2 + \sum_{\ell \in \mathcal{E}} \sum_{T: T \cap Y_\ell \neq \emptyset \wedge T \cap L_{\ell-1} = \emptyset} \hat{f}(T)^2. \end{aligned} \quad (4)$$

As above, since $(L_\ell \setminus X_\ell) \cap [k] = \emptyset$,

$$\sum_{\ell \in \mathcal{E}} \sum_{T: T \cap (L_\ell \setminus X_\ell) \neq \emptyset \wedge T \cap L_{\ell-1} = \emptyset} \hat{f}(T)^2 \leq \sum_{T \subseteq [n] \setminus [k]} \hat{f}(T)^2 \leq \varepsilon^2. \quad (5)$$

It remains to bound the last sum on the right-hand side of (4). By splitting up the terms in this sum according to whether $|T| \leq k$ or not, we obtain

$$\sum_{T: T \cap Y_\ell \neq \emptyset \wedge T \cap L_{\ell-1} = \emptyset} \hat{f}(T)^2 \leq \sum_{|T| \leq k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap Y_\ell \neq \emptyset] + \sum_{|T| > k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap L_{\ell-1} = \emptyset].$$

Let $Z \subseteq [n] \setminus [k]$ denote the set of coordinates that occur in one of the the original parts $S_{P_1}, \dots, S_{P_{100k^4}}$ that also contains one of the elements in $[k]$. Then $Y_\ell \subseteq Z$ and

$$\sum_{\ell \in \mathcal{E}} \sum_{|T| \leq k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap Y_\ell \neq \emptyset] \leq \sum_{\ell \in \mathcal{E}} \sum_{|T| \leq k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap Z \neq \emptyset] \leq k \cdot \sum_{|T| \leq k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap Z \neq \emptyset].$$

The probability, over the choice of P_1, \dots, P_{100k^4} , that $T \cap Z \neq \emptyset$ is at most $|T|/100k^3$, so the expected value of the last expression (again over the choice of the initial partition) is bounded above by

$$\mathbb{E} \left[\sum_{\ell \in \mathcal{E}} \sum_{|T| \leq k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap Y_\ell \neq \emptyset] \right] \leq k \cdot \sum_{|T| \leq k, T \setminus [k] \neq \emptyset} \hat{f}(T)^2 \cdot \left(\frac{k}{100k^3} \right) \leq \frac{1}{100k} \cdot \text{Inf}_f([n] \setminus [k]) \leq \frac{\varepsilon^2}{100k}.$$

(6)

Lastly, since $L_0 \subseteq L_{\ell-1}$ for each $\ell \geq 1$,

$$\sum_{|T|>k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap L_{\ell-1} = \emptyset] \leq \sum_{|T|>k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap L_0 = \emptyset].$$

A set T can be disjoint from L_0 only when its elements are contained in at most k of the parts of the initial random partition, which happens with probability at most $\frac{1}{100k^2}$ when $|T| > k$, so

$$\mathbb{E} \left[\sum_{\ell \in \mathcal{E}} \sum_{|T|>k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap L_{\ell-1} = \emptyset] \right] \leq \mathbb{E} \left[k \sum_{|T|>k} \hat{f}(T)^2 \cdot \mathbf{1}[T \cap L_0 = \emptyset] \right] \leq \frac{1}{100k} \sum_{|T|>k} \hat{f}(T)^2 \leq \frac{\varepsilon^2}{100k}, \quad (7)$$

where the last inequality uses the fact that $\sum_{|T|>k} \hat{f}(T)^2 \leq \text{Inf}_f([n] \setminus [k])$.

Combining the inequalities (1)–(7), we obtain that the expected value of $\text{Inf}_f([n] \setminus S_B)$ is bounded above by

$$\mathbb{E} [\text{Inf}_f([n] \setminus S_B)] \leq (1 + \frac{2}{100k^2})\varepsilon^2 + \frac{2\varepsilon^2}{100k} + (1 + \frac{2}{100k})2\varepsilon^2 \leq 4\varepsilon^2.$$

Applying Markov's inequality and adding the probability that the junta variables are completely separated in the partition P_1, \dots, P_{100k^4} completes the proof of the lemma.

3.3 Proof of Lemma 4

For any $X = (x^{(1)}, \dots, x^{(q)})$, let $\text{dist}_X(f_1, f_2) = \left(\frac{1}{q} \sum_{i=1}^q (f_1(x^{(i)}) - f_2(x^{(i)}))^2 \right)^{1/2}$ denote the empirical distance between f_1 and f_2 according to X . To prove the lemma, we want to show that $\text{dist}_X(f, h)$ is within the specified bounds.

The function dist_X is a metric, so we can apply the triangle inequality to obtain

$$\text{dist}_X(f, h) \leq \text{dist}_X(f, g) + \text{dist}_X(g, h).$$

By Hoeffding's inequality, when $x^{(1)}, \dots, x^{(q)}$ are drawn independently and uniformly at random, the upper bound

$$\text{dist}_X(f, g) \leq \text{dist}_2(f, g) + \varepsilon \leq 2\varepsilon$$

holds except with probability at most $e^{-16q\varepsilon^4}$.

We now want to show that $\text{dist}_X(g, h)$ is also close to $\text{dist}_2(g, h)$. This analysis is a bit more subtle, however, because the choice of samples $x^{(1)}, \dots, x^{(q)}$ is *not* independent of h (as it affects what mapping ψ will be chosen by the algorithm). So before we can apply concentration inequalities, we must “decouple” X and h . To do so, we introduce a new random process for generating X . Let $\lambda : [n] \rightarrow \{0, 1\}^q$ be chosen uniformly at random. This function corresponds to a random partition of the set $[n]$ of coordinates into 2^q parts. Let $\pi : \{0, 1\}^q \rightarrow \{0, 1\}^q$ be a random permutation. Then the random variable X obtained by setting $x_j^{(i)} = \pi(\lambda(j))_i$ has the desired uniform distribution over sequences of q vectors in $\{0, 1\}^n$.

This random process is designed so that the choice of ψ in the algorithm (and therefore also h) is *independent* of π ; the only information about X used in determining it is the

identity of the parts defined by λ , not what values the coordinates in each parts receive on the q queries. Then

$$\mathbb{E}_X[\text{dist}_X(g, h)] = \mathbb{E}_{\lambda, r} [\mathbb{E}_\pi[\text{dist}_X(g, h)]]$$

where r represents the internal randomness of the algorithm outside of that used to generate X . With probability at least $k^2/2^q$, the partition λ completely separates the indices in J . Fix such a partition λ . Define $J^* = J \cup \text{supp}(\psi)$. Then $|J^*| \leq 2k$. Define $Y = (y^{(1)}, \dots, y^{(q)})$ by setting $y^{(i)} = x_{J^*}^{(i)}$. Since $\text{dist}_X(g, h)$ only depend on the coordinates in J^* , we can write it equivalently as $\text{dist}_Y(g, h)$.

Let D denote the distribution on Y induced by π . The distribution D is close to but not equal to the uniform distribution U on $\{0, 1\}^{q \times |J^*|}$, since D is equivalent to the distribution obtained by making drawing $(y_i^{(1)}, \dots, y_i^{(q)})$ for each $i \in J^*$ without replacement from $\{0, 1\}^q$. Then

$$\begin{aligned} \Pr_{Y \sim D} [|\text{dist}_Y(g, h) - \mathbb{E}_{Y \sim U} \text{dist}_Y(g, h)| \geq \varepsilon] &\leq d_{\text{TV}}(D, U) + \Pr_{Y \sim U} [|\text{dist}_Y(g, h) - \mathbb{E}_{Y \sim U} \text{dist}_Y(g, h)| \geq \varepsilon] \\ &\leq \frac{4k^2}{2^q} + e^{-16q\varepsilon^4}. \end{aligned}$$

In the last inequality, the bound $d_{\text{TV}}(D, U) \leq \frac{(2k)^2}{2^q}$ is by the standard total variation bound between sampling with and without replacement [19] and the other bound on the other term is by Hoeffding's inequality.

4 Applications

In this short section, we show how Theorems 1.1 and 1.2 both follow directly from Theorem 1.3 and the junta theorem of Feldman and Vondrák.

Proof of Theorem 1.1. By the first part of the Feldman–Vondrák junta theorem, every submodular function $f \in \mathcal{F}_n$ is $\frac{\varepsilon}{10^6}$ -close to a k -junta for some $k = O(\frac{1}{\varepsilon^2} \log \frac{1}{\varepsilon})$. Therefore, by Theorem 1.3, submodularity can be tested with $2^{O(k \log k)}/\varepsilon^{10} = 2^{\tilde{O}(1/\varepsilon^2)}$ queries in the ℓ_2 testing model. By Fact 2.1, the number of queries for testing submodularity in the ℓ_p testing model for any $1 \leq p < 2$ is also $2^{\tilde{O}(1/\varepsilon^2)}$ and for any $p > 2$ it is $2^{\tilde{O}(1/(\varepsilon^{p/2})^2)} = 2^{\tilde{O}(1/\varepsilon^p)}$. ◀

Proof of Theorem 1.2. By Lemma 1, additive functions, coverage functions, unit demand functions, OXS functions, and gross substitute functions are all also submodular. Therefore, the first part of the Feldman–Vondrák junta theorem also applies to these functions and the rest of the proof is identical to that of Theorem 1.1.

Lemma 1 also implies that fractionally subadditive functions are self-bounding, so the second part of the Feldman–Vondrák junta theorem shows that every function f that has either of these properties is $\frac{\varepsilon}{10^6}$ -close to a k -junta for some $k = 2^{O(\frac{1}{\varepsilon^2})}$. Therefore, by Theorem 1.3, fractional subadditivity and self-boundedness can both be tested with $2^{O(k \log k)}/\varepsilon^{10} = 2^{2^{\tilde{O}(1/\varepsilon^2)}}$ queries in the ℓ_2 testing model; the general result for the ℓ_p testing model again follows directly from Fact 2.1. ◀

A Missing proofs from Section 2

We begin with the proof of Lemma 2. We emphasize that the proof below is essentially as found in [6]; the reason we include it here is that the original statement of the proof only applied to Boolean-valued functions. As we see below, however, the same argument also holds for real-valued functions.

23:14 Testing submodularity and other properties of valuation functions

► **Theorem A.1.** (Dinur and Safra [12]; Friedgut [20]) Let \mathcal{G} be a t -intersecting family of subsets of $[n]$ for some $t \geq 1$. For any $p < \frac{1}{t+1}$, the p -biased measure of \mathcal{G} is bounded by $\mu_p(\mathcal{G}) \leq p^t$.

Proof of Lemma 2. For $0 \leq t \leq \frac{1}{2}$, let $\mathcal{G}_t = \{J \subseteq [n] : \text{Inf}_f(\overline{J}) < t\varepsilon^2\}$ be the family of all the sets whose compliments have influence less than $t\varepsilon^2$. For any two sets $J, K \in \mathcal{G}_{\frac{1}{2}}$, the subadditivity of influence implies that

$$\text{Inf}_f(\overline{J \cap K}) = \text{Inf}_f(\overline{J} \cup \overline{K}) \leq \text{Inf}_f(\overline{J}) + \text{Inf}_f(\overline{K}) < \varepsilon^2.$$

But f is ε -far from every k -junta, so for any two sets $J, K \in \mathcal{G}_{\frac{1}{2}}$, $|J \cap K| > k$, from Proposition 2.8. Which means $\mathcal{G}_{\frac{1}{2}}$ is a $k+1$ intersecting family. There are two cases now, first one is, there is at least one set $J \in \mathcal{G}_{\frac{1}{2}}$ such that $|J| < 2k$, second one is all the sets $J \in \mathcal{G}_{\frac{1}{2}}$ will have $|J| \geq 2k$. We will show that in both the cases our lemma holds. In the first case let $J \in \mathcal{G}_{\frac{1}{2}}$ be a set which has fewer than $2k$ elements, with high probability the set J is completely separated by the partition \mathcal{P} , and we know that for any $K \in \mathcal{G}_{\frac{1}{2}}$, $|J \cap K| \geq k+1$, which means K is not covered by any union of k -parts in \mathcal{P} . Therefore, $\text{Inf}_f(\overline{J}) \geq \frac{\varepsilon^2}{2} > \frac{\varepsilon^2}{4}$ as we wanted to show.

Consider the case where, all the sets in $\mathcal{G}_{\frac{1}{2}}$ have more than $2k$ elements. Then $\mathcal{G}_{\frac{1}{4}}$ is a $2k$ intersecting family. Otherwise, if there are two sets $J, K \in \mathcal{G}_{\frac{1}{4}}$ such that $|J \cap K| < 2k$, then $\text{Inf}_f(\overline{J \cap K}) \leq \text{Inf}_f(\overline{J}) + \text{Inf}_f(\overline{K}) < \frac{\varepsilon^2}{4} + \frac{\varepsilon^2}{4} < \frac{\varepsilon^2}{2}$, thus contradicting our assumption.

Let $J \subseteq [n]$ be the union of k parts in \mathcal{P} . Since \mathcal{P} is a random partition, J is a random subset obtained by including each element of $[n]$ in J independently with probability $p = \frac{k}{r} < \frac{1}{2k+1}$. By Theorem A.1, $\Pr_{\mathcal{P}}[\text{Inf}_f(\overline{J}) < \frac{\varepsilon^2}{4}] = \Pr[J \in \mathcal{G}_{\frac{1}{4}}] = \mu_{\frac{k}{r}}(\mathcal{G}_{\frac{1}{4}}) \leq (\frac{k}{r})^{2k}$. By the union bound the probability that there exists a set $J \subseteq [n]$ that is the union of k parts in \mathcal{P} for which $\text{Inf}_f(\overline{J}) < \frac{\varepsilon^2}{4}$ is bounded above by $\binom{r}{k} (\frac{k}{r})^{2k} \leq (\frac{er}{k})^k (\frac{k}{r})^{2k} \leq (\frac{ek}{r})^k < \frac{1}{6}$. ◀

The proof of Proposition 2.9 is obtained by considering the ESTIMATEINF algorithm below.

Algorithm 2: ESTIMATEINF(f, S, m)

- 1 Draw x_1, \dots, x_m uniformly and independently at random from $\{0, 1\}^S$;
 - 2 Draw $y_1, \dots, y_m, y'_1, \dots, y'_m$ uniformly and independently at random from $\{0, 1\}^S$;
 - 3 Return $\frac{1}{2m} \sum_{i=1}^m (f(x_i, y_i) - f(x_i, y'_i))^2$;
-

The concentration of the estimated influence is obtained via the following (standard) version of Hoeffding's inequality.

► **Hoeffding's inequality.** Let X_1, \dots, X_n be independent random variables bounded by $a_1 \leq X_i \leq b_i$. Let $X = X_1 + X_2 + \dots + X_n$ have expected value $E[X] = \mu$. Then for any $t > 0$,

$$\Pr[|X - \mu| \geq t] \leq 2e^{-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}}.$$

Acknowledgments

The authors wish to thank the anonymous referees for valuable feedback, Amit Levi for insightful discussions and Karl Wimmer for pointing out [31] to us. E.B. is supported by an NSERC Discovery Grant.

References

- 1 Ashwinkumar Badanidiyuru, Shahar Dobzinski, Hu Fu, Robert Kleinberg, Noam Nisan, and Tim Roughgarden. Sketching valuation functions. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1025–1035, 2012. URL: <http://portal.acm.org/citation.cfm?id=2095197&CFID=63838676&CFTOKEN=79617016>.
- 2 Maria-Florina Balcan, Florin Constantin, Satoru Iwata, and Lei Wang. Learning valuation functions. In *COLT 2012 - The 25th Annual Conference on Learning Theory, June 25-27, 2012, Edinburgh, Scotland*, pages 4.1–4.24, 2012. URL: <http://www.jmlr.org/proceedings/papers/v23/balcan12b/balcan12b.pdf>.
- 3 Maria-Florina Balcan and Nicholas J. A. Harvey. Learning submodular functions. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 793–802, 2011. URL: <http://doi.acm.org/10.1145/1993636.1993741>, doi:10.1145/1993636.1993741.
- 4 Piotr Berman, Sofya Raskhodnikova, and Grigory Yaroslavtsev. L_p -testing. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 164–173, 2014. URL: <http://doi.acm.org/10.1145/2591796.2591887>, doi:10.1145/2591796.2591887.
- 5 Eric Blais. Testing juntas nearly optimally. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 151–158, 2009. URL: <http://doi.acm.org/10.1145/1536414.1536437>, doi:10.1145/1536414.1536437.
- 6 Eric Blais, Amit Weinstein, and Yuichi Yoshida. Partially symmetric functions are efficiently isomorphism testable. *SIAM J. Comput.*, 44(2):411–432, 2015. URL: <http://dx.doi.org/10.1137/140971877>, doi:10.1137/140971877.
- 7 Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *J. Comput. Syst. Sci.*, 47(3):549–595, 1993. URL: [http://dx.doi.org/10.1016/0022-0000\(93\)90044-W](http://dx.doi.org/10.1016/0022-0000(93)90044-W), doi:10.1016/0022-0000(93)90044-W.
- 8 Deeparnab Chakrabarty and Zhiyi Huang. Recognizing coverage functions. *SIAM Journal on Discrete Mathematics*, 29(3):1585–1599, 2015. URL: <http://dx.doi.org/10.1137/140964072>, arXiv:<http://dx.doi.org/10.1137/140964072>, doi:10.1137/140964072.
- 9 Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Efficient sample extractors for juntas with applications. In *Automata, Languages and Programming - 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part I*, pages 545–556, 2011. URL: http://dx.doi.org/10.1007/978-3-642-22006-7_46, doi:10.1007/978-3-642-22006-7_46.
- 10 Ilias Diakonikolas, Homin K. Lee, Kevin Matulef, Krzysztof Onak, Ronitt Rubinfeld, Rocco A. Servedio, and Andrew Wan. Testing for concise representations. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, pages 549–558, 2007. URL: <http://dx.doi.org/10.1109/FOCS.2007.70>, doi:10.1109/FOCS.2007.70.
- 11 Ilias Diakonikolas, Homin K. Lee, Kevin Matulef, Rocco A. Servedio, and Andrew Wan. Efficiently testing sparse $GF(2)$ polynomials. In *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I: Track A: Algorithms, Automata, Complexity, and Games*, pages 502–514, 2008. URL: http://dx.doi.org/10.1007/978-3-540-70575-8_41, doi:10.1007/978-3-540-70575-8_41.
- 12 Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162:2005, 2004.

- 13 Uriel Feige. On maximizing welfare when utility functions are subadditive. *SIAM J. Comput.*, 39(1):122–142, 2009. URL: <http://dx.doi.org/10.1137/070680977>, doi:10.1137/070680977.
- 14 Uriel Feige, Vahab S. Mirrokni, and Jan Vondrák. Maximizing non-monotone submodular functions. *SIAM J. Comput.*, 40(4):1133–1153, 2011. URL: <http://dx.doi.org/10.1137/090779346>, doi:10.1137/090779346.
- 15 Vitaly Feldman and Pravesh Kothari. Learning coverage functions and private release of marginals. In *Proceedings of The 27th Conference on Learning Theory, COLT 2014, Barcelona, Spain, June 13-15, 2014*, pages 679–702, 2014. URL: <http://jmlr.org/proceedings/papers/v35/feldman14a.html>.
- 16 Vitaly Feldman and Jan Vondrák. Tight bounds on low-degree spectral concentration of submodular and XOS functions. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 923–942, 2015. URL: <http://dx.doi.org/10.1109/FOCS.2015.61>, doi:10.1109/FOCS.2015.61.
- 17 Vitaly Feldman and Jan Vondrák. Optimal bounds on approximation of submodular and XOS functions by juntas. *SIAM J. Comput.*, 45(3):1129–1170, 2016. URL: <http://dx.doi.org/10.1137/140958207>, doi:10.1137/140958207.
- 18 Eldar Fischer, Guy Kindler, Dana Ron, Shmuel Safra, and Alex Samorodnitsky. Testing juntas. *J. Comput. Syst. Sci.*, 68(4):753–787, 2004. URL: <http://dx.doi.org/10.1016/j.jcss.2003.11.004>, doi:10.1016/j.jcss.2003.11.004.
- 19 David Freedman. A remark on the difference between sampling with and without replacement. *Journal of the American Statistical Association*, 72(359):681–681, 1977. doi:10.1080/01621459.1977.10480637.
- 20 Ehud Friedgut. On the measure of intersecting families, uniqueness and stability. *Combinatorica*, 28(5):503–528, 2008. URL: <http://dx.doi.org/10.1007/s00493-008-2318-9>, doi:10.1007/s00493-008-2318-9.
- 21 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998. URL: <http://doi.acm.org/10.1145/285055.285060>, doi:10.1145/285055.285060.
- 22 Parikshit Gopalan, Ryan O’Donnell, Rocco A. Servedio, Amir Shpilka, and Karl Wimmer. Testing fourier dimensionality and sparsity. *SIAM J. Comput.*, 40(4):1075–1100, 2011. URL: <http://dx.doi.org/10.1137/100785429>, doi:10.1137/100785429.
- 23 Eyal Kushilevitz and Yishay Mansour. Learning decision trees using the fourier spectrum. *SIAM J. Comput.*, 22(6):1331–1348, 1993. URL: <http://dx.doi.org/10.1137/0222080>, doi:10.1137/0222080.
- 24 Benny Lehmann, Daniel J. Lehmann, and Noam Nisan. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior*, 55(2):270–296, 2006. URL: <http://dx.doi.org/10.1016/j.geb.2005.02.006>, doi:10.1016/j.geb.2005.02.006.
- 25 Kevin Matulef, Ryan O’Donnell, Ronitt Rubinfeld, and Rocco A. Servedio. Testing half-spaces. *SIAM J. Comput.*, 39(5):2004–2047, 2010. URL: <http://dx.doi.org/10.1137/070707890>, doi:10.1137/070707890.
- 26 Ryan O’Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014. URL: <http://www.cambridge.org/de/academic/subjects/computer-science/algorithmics-complexity-computer-algebra-and-computational-g/analysis-boolean-functions>.
- 27 Michal Parnas, Dana Ron, and Alex Samorodnitsky. Testing basic boolean formulae. *SIAM J. Discrete Math.*, 16(1):20–46, 2002. URL: <http://epubs.siam.org/sam-bin/dbq/article/40744>.

- 28 Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM J. Comput.*, 25(2):252–271, 1996. URL: <http://dx.doi.org/10.1137/S0097539793255151>, doi:10.1137/S0097539793255151.
- 29 Rocco A. Servedio. Testing by implicit learning: A brief survey. In *Property Testing - Current Research and Surveys*, pages 197–210, 2010. URL: http://dx.doi.org/10.1007/978-3-642-16367-8_11, doi:10.1007/978-3-642-16367-8_11.
- 30 C. Seshadhri and Jan Vondrák. Is submodularity testable? In *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 7-9, 2011. Proceedings*, pages 195–210, 2011. URL: <http://conference.itcs.tsinghua.edu.cn/ICS2011/content/papers/21.html>.
- 31 Karl Wimmer and Yuichi Yoshida. Testing linear-invariant function isomorphism. In *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part I*, pages 840–850, 2013. URL: http://dx.doi.org/10.1007/978-3-642-39206-1_71, doi:10.1007/978-3-642-39206-1_71.