58th Annual IEEE Symposium on Foundations of Computer Science

Active classification with comparison queries

Daniel M. Kane
Department of Computer Science and Engineering
Department of Mathematics
University of California, San Diego
La Jolla, United States
dakane@ucsd.edu

Shachar Lovett
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, United States
slovett@cs.ucsd.edu

Shay Moran
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, United States
shaymoran1@gmail.com

Jiapeng Zhang
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, United States
jpeng.zhang@gmail.com

***Abstract*—We study an extension of active learning in which the learning algorithm may ask the annotator to compare the distances of two examples from the boundary of their label-class. For example, in a recommendation system application (say for restaurants), the annotator may be asked whether she liked or disliked a specific restaurant (a label query); or which one of two restaurants did she like more (a comparison query).**

We focus on the class of half spaces, and show that under natural assumptions, such as large margin or bounded bit-description of the input examples, it is possible to reveal all the labels of a sample of size n using approximately $O(\log n)$ queries. This implies an exponential improvement over classical active learning, where only label queries are allowed. We complement these results by showing that if any of these assumptions is removed then, in the worst case, $\Omega(n)$ queries are required.

Our results follow from a new general framework of active learning with additional queries. We identify a combinatorial dimension, called the *inference dimension*, that captures the query complexity when each additional query is determined by $O(1)$ examples (such as comparison queries, each of which is determined by the two compared examples). Our results for half spaces follow by bounding the inference dimension in the cases discussed above.

I. Introduction

A central goal of *interactive learning* is understanding what type of interaction between a learner and a domain expert enhances the learning process, compared to the classical *passive learning* from labeled examples.

A basic model that was studied in this context is *pool-based active learning* [1]. Here, the algorithm has an access to a large pool of *unlabeled examples* from which it can pick examples and query their labels. The goal is to make as few queries as possible while achieving generalization-guarantees which are comparable with these of a passive algorithm with an access to all of the labels.

A canonical example that demonstrates an advantage of active learning is the class of threshold functions[1] over the real line. Indeed, let c denote the learned threshold function, and let $x_1 < x_2 < \ldots < x_n$ in $\mathbb{R}$ be the given pool of unlabeled examples. It is possible to infer the labels of all n points by making at most $\log n+2$ queries: query the labels of the extreme points $c(x_1), c(x_n)$; if $c(x_1) = c(x_n)$ then the remaining points must be labeled the same; otherwise, continue in a binary search fashion, by labeling the middle point of the interval whose extreme points have opposite labels. After at most $\log n$ such queries, the labels of all n points are revealed.

Unfortunately, this exponential improvement in the query complexity breaks for more general concept classes. In fact, even for the class of 2 dimensional threshold functions[2], namely the class of half-planes, the (worst-case) query complexity of active learning equals that of passive learning (see e.g. [2]). Consequently, much of the literature was dedicated to developing theory that takes into consideration further properties of the unknown underlying distribution or the target concept [3]–[18].

We consider another approach by allowing the learning algorithm to further interact with the domain expert by asking additional queries. This poses a question:

Which additional queries can the algorithm use?

Allowing arbitrary queries will result in a very strong

[1] These are "$\mathbb{R} \to \{\pm 1\}$" functions of the form $c(x) = \text{sign}(a \cdot x - b)$, where $a, b \in \mathbb{R}$.

[2] These are "$\mathbb{R}^2 \to \{\pm 1\}$" functions of the form $c(x) = \text{sign}(\langle a, x\rangle - b)$, where $a \in \mathbb{R}^2, b \in \mathbb{R}$.

0272-5428/17 $31.00 © 2017 IEEE
DOI 10.1109/FOCS.2017.40

IEEE computer society

learner (indeed, by halving the set of potential hypothesis in every query, the number of queries can be made logarithmic). However, arbitrary queries are useless in practice: as experimental work by [19] shows, algorithms that use set of queries that are too rich may result in a poor practical performance. This is not surprising if we keep in mind that the human annotator who answers the queries is restricted (computationally and in other ways). Thus, a crucial factor in choosing the additional queries is compatibility with the annotator who answers them. A popular type of queries, which is used in applications involving human annotators, is *relative queries*. These queries poll relative information between two or more data points.

This work focuses on a basic kind of relative queries — *comparison queries*. Using such queries is sensible in settings in which there is some natural ordering of the instances with respect to the learned concept. As a toy example, consider the goal of classifying films according to whether a certain individual is likely to enjoy them or not (e.g. for recommending new films for this person). In this context, the input sample consists of films watched by the individual, a label query asks whether the person liked a film, and a comparison query asks which of two given films did the individual prefer. As we will see, comparison queries may significantly help.

Another aspect implied by the restricted nature of the human annotator is that the learned concept resides in the class of concepts that can be computed by the annotator, which presumably has low capacity. Thus, realizability assumptions about the generating data distributions are plausible in this context.

A. Active learning with additional comparison queries

Consider a learned concept of the form $c(x) = \text{sign}\big(f(x)\big)$, where f is a real valued function (e.g. half spaces, neural nets), and consider two instances x_1, x_2 such that, say $f(x_1) = 10$, and $f(x_2) = 1000$. Both $c(x_1)$ and $c(x_2)$ equal $+1$, however that $f(x_2) >> f(x_1)$ suggests that x_2 is a "more positive instance" than x_1. In the setting of film classification this is naturally interpreted as that the person likes the film x_2 more than the film x_1. We call the query "$f(x_2) \geq f(x_1)$?" *a comparison query.*

1) Example: learning half-planes with comparison and label queries: To be concrete, consider the class of half-planes in $\mathbb{R}^2$. Here, a comparison query is equivalent to asking which of two sample points x_1, x_2 lies closer to the boundary line. Do such queries improve the query complexity over standard active learning? It is known that without such queries, in the worst-case, the learner essentially has to query all labels [2].

The following algorithm demonstrates an exponential improvement when comparison queries are allowed. Later, we will present more general results showing that this can be generalized to higher dimensions under some natural restrictions, and that such restrictions are indeed necessary.

Interactive learning algorithm with comparison queries for half planes in $\mathbb{R}^2$

(see Figure 1 for a graphical illustration)

Setting: an unlabeled input sample of n points $x_1, \ldots, x_n \in \mathbb{R}^2$ with hidden labels according to a half-plane $c(x) = \text{sign}\big(f(x)\big)$, where $f : \mathbb{R}^2 \to \mathbb{R}$ is affine.

Repeat until all points are labeled:

1) Sample uniformly a subsample S of 30 points.
2) Query the labels of the points in S, and denote by:
   $\mathcal{P}$ — points in S labeled by $+1$,
   $\mathcal{N}$ — points in S labeled by -1.
3) Use comparison queries to find:
   (i) q – the closest point in $\mathcal{P}$ to the boundary line,
   (ii) v – the closest point in $\mathcal{N}$ to the boundary line.
4) Denote by:
   $[p, q], [q, r]$ — the two edges of the convex hull of $\mathcal{P}$ that are incident to q,
   $[u, v], [v, w]$ — the two edges of the convex hull of $\mathcal{N}$ that are incident to v.
5) Infer that:
   all points inside the cone $\angle pqr$ are labeled $+1$,
   all points inside the cone $\angle uvw$ are labeled -1.
6) Repeat on the remaining unlabeled points.

The algorithm proceeds by iterations: it repeats steps 1-5 until all points are labeled. At each iteration at most 60 queries are performed: 30 label queries in step 2 and at most 30 comparison queries in step 3, for finding the points q, v of minimal distance. In each iteration, the algorithm infers the labels of all points in the union of the angles $\angle pqr, \angle uvw$. We will refer to this region as *"the confident region"*.

We claim that after an expected number of $O(\log n)$ iterations (and therefore using only $O(\log n)$ queries) the algorithm infers the labels of all input points. Establishing this statement boils down, via a boosting

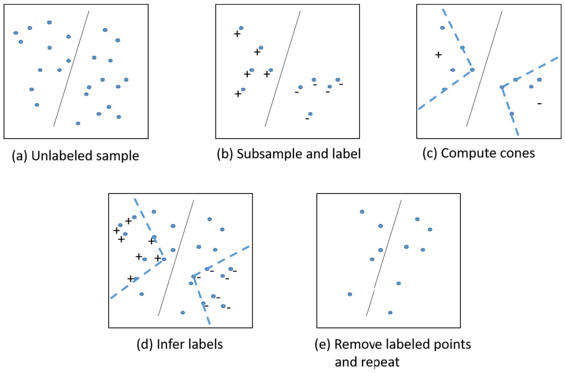

Figure 1. An illustration of a single iteration.

argument (Theorem III.2) to the following two important properties, which are easy to verify:

- **Confidence:** every point which is labeled by the algorithm is labeled correctly.
- **Inference from a small subsample:** the confident region is determined by a subsample of 6 labeled points of S (i.e. p, q, r and u, v, w).

These properties imply that at each iteration, with probability at least $1/2$, half of the remaining unlabeled points are labeled correctly (see Lemma III.3). Thus, the expected number of queries is $O(\log n)$.

Paper organization: In Section I-B we present and discuss our results, later, in Section I-C we survey previous related works. Sections II, III, and IV contain the definitions and results. This extended abstract does not include many of the proofs, due to lack of space. The full version of this paper is available on arXiv [20].

B. Results

We next present and discuss our main results.

1) Subsection I-B1 is dedicated to our results concerning half spaces,
2) Subsection I-B2 is dedicated to the inference dimension, and how it captures the query complexity of active learning with additional comparison queries, and
3) Subsection I-B3 focuses on the general framework of active learning with additional queries.

1) Interactive learning of half spaces with comparison queries: We start by discussing our results for interactive learning of half spaces in $\mathbb{R}^d$ when both label queries and comparison queries are allowed. We show that a general algorithm, as the one we described for $\mathbb{R}^2$, cannot exist for $d \geq 3$. However, we identify two useful properties that allow for such a learning algorithm: bounded bit complexity and margin.

Exact recovery of labels: We first describe our results in the context of exact recovery. Here, the labels of all n sample points need to be revealed, using as few queries as possible.

We first show that in the worst case, in $\mathbb{R}^d$ this requires $\Omega(n)$ queries for any $d \geq 3$ (recall that $O(\log n)$ queries suffice in $\mathbb{R}^2$).

Theorem I.1 (Theorem IV.6, informal version)**.** *There are n points in $\mathbb{R}^3$ that require $\Omega(n)$ label and comparison queries for revealing all labels.*

Our first positive result shows that efficient exact recovery of labels is possible if the points have low bit complexity.

Theorem I.2 (Theorem IV.1, informal version)**.** *Consider an arbitrary realizable sample of n points in $\mathbb{R}^d$ whose individual bit complexity is B. The labels of all sample points can be learned using $\tilde{O}(B \log n)$ label and comparison queries.*

As an example, consider the sample consisting of the 2^N point in the boolean hypercube $\{0,1\}^N$. The bit complexity of every point is N. The above thoerem implies that given an unknown threshold function on $\{0,1\}^N$, it is possible to reveal all 2^N labels using $\tilde{O}(N^2)$ comparison and label queries. This should be compared to the situation where only label queries are allowed, where all 2^N queries are necessary.

Our second positive result shows that a similar algorithm exists if the margin is large.

Theorem I.3 (Theorem IV.4, informal version)**.** *Assume a sample of n points in $\mathbb{R}^d$ with maximal ℓ_2 norm ρ and margin at least γ with respect to the learned half-space. The labels of all points can be recovered using $\tilde{O}\big(d \log(\rho/\gamma) \log n\big)$ label and comparison queries.*

Our bound is in fact stronger: it is

$$\tilde{O}\big(d \log(1/\eta) \log n\big),$$

where η is the *minimal-ratio* of the input sample, defined by $\frac{|\min_i f(x_i)|}{|\max_i f(x_i)|}$, where $x_1, \ldots, x_n$ are the sample points and $\text{sign}\big(f(x)\big)$ is the learned concept. In Section IV-A2 we show that the minimal ratio is lower bounded by the margin (and therefore yields a stronger statement).

Note that the above bound depends on ρ/γ logarithmically, and therefore applies even in settings when the margin is exponentially small. Similar dependence of on ρ/γ is obtained by the Ellipsoid method [21] and Vaidya Cutting Plane method [22], that use $O(d^2 \log(\rho/\gamma))$ and $O(d \log(d\rho/\gamma))$ iterations respectively, when used to find a linear classifier that is consistent with a realizable sample with margin ρ/γ in $\mathbb{R}^d$.

The upper bound in the above theorem depends on the dimension, which is often avoided in bounds that depend on the margin. It is therefore natural to ask whether there is a bound that depends only on the margin. We show that it is impossible:

Theorem I.4 (Theorem IV.11, informal version)**.** *There is a sample of n unit vectors in $\mathbb{R}^{n+1}$ with $\Omega(1)$ margin that require $\Omega(n)$ label and comparison queries to recover all the labels.*

Statistical learning: Using standard arguments, we translate the results above to the statistical setting and get bounds on the sample and query complexity: the algorithmic results above directly give a learning algorithm for realizable distributions with sample complexity $n(\epsilon, \delta) = O\left(\frac{d+\log(1/\delta)}{\epsilon}\right)$ and query complexity approximately logarithmic in $n(\epsilon, \delta)$, where ϵ is the error and $1-\delta$ is the confidence of the algorithm. (See Section IV-A1 for the bit complexity and Section IV-A2 for the margin.) Our lower bounds translate analogously to realizable distributions that require $\Omega(1/\epsilon)$ queries for achieving error ϵ and constant confidence (say $1-\delta = 5/6$). See Section IV-B for details.

2) Inference dimension: Our results for learning half spaces rely on a common combinatorial property that we describe next. Let X be a set and H a concept class where each concept is of the form $\text{sign}(f(x))$ for $f : X \to \mathbb{R}$. For example, X may be a finite set $X = \{x_1, \ldots, x_n\} \subseteq \mathbb{R}^d$ and H the class of all half spaces with margin at least $1/100$ with respect to X; or $X = \{0,1\}^d$ and H is all half spaces; or $X = \mathbb{R}^d$ and H a class of (signs of) low degree polynomials; etcetera.

Let $S \subseteq X$ be an unlabeled sample. An *$S-$query* is either a label query regarding some $x \in S$, or a comparison query regarding $x_1, x_2 \in S$. Namely, the allowed queries are "$f(x) \geq 0$?" (label query) or "$f(x_1) \geq f(x_2)$" (comparison query). For $x \in X$ and $c \in H$, let

$$S \underset{f}{\Longrightarrow} x$$

denote the statement that the comparison and label queries on S determine the label of x, when the learned concept is $c = \text{sign}\big(f(x)\big)$.

The *inference-dimension* of (X, H) is the smallest number k such that for every $c = \text{sign}(f(x)) \in H$, and every $S \subset X$ of size at least k, there exists $x \in S$ such that

$$S \setminus \{x\} \underset{f}{\Longrightarrow} x.$$

In other words, if the inference dimension is k then in every sample of size k or more, there is a point whose label can be inferred from the label and comparison queries on the remaining points.

For example the inference dimension of (X, H), where $X = \mathbb{R}$, and H is the class of threshold functions is 3: indeed, to see it is at most 3, note that if all 3 points have the same label, then the label of the midpoint can be inferred from the other two labels, and if not all points have the same label then the midpoint and, say the right point have the same label, and so the label of the right point can be inferred from the other 2 labels. In this example comparison queries are not required. Another example, which requires comparison queries, is where $X = \mathbb{R}^2$ and H is the class of half-planes. Here the inference dimension[3] is at most 7: indeed, in any sample of at least 7 points there are 4 points with the same label, and the label of one of these points can be inferred from the other 3 (see Figure 1 and Section I-A1).

The next Theorem shows that inference dimension captures the query-complexity in active learning with comparison queries. It is worth noting that in the classical setting of active learning, when only label queries are allowed, the inference dimension specializes to the the *star dimension* [23], which similarly captures the (worst-case) query complexity in this setting.

Theorem I.5. *Let k denote the inference dimension of (X, H). Then:*

1) *There is an algorithm that reveals the labels of any realizable sample of size n using at most $O(k \log k \log n)$ queries.*
2) *Any algorithm that reveals the labels of any realizable sample of size k must use $\Omega(k)$ queries in the worst-case.*

The upper bound (the first item) is a corollary of Theorem III.2, and the lower bound is a corollary of Theorem III.5. Both Theorems are discussed in the next subsection. While the lower bound is relatively straight forward, our derivation of the upper bound requires several steps, which we summarize next.

(i) **Low inference dimension $\Longrightarrow$ weak confident learner**: we first show that if the inference dimension is at most k then there is a *weak confident learner* for (X, H) with query complexity $O(k \log k)$ (see Lemma III.3). A confident learner is a learning algorithm that may abstain from predicting on some points $x \in X$, but must be correct on every point where it does not abstain. A weak confident learner is a confident learner that with constant probability does not abstain on a constant fraction of X (see Section III-A for a formal definition).

(ii) **Boosting the weak confident learner**: once a weak confident learner is derived, we transform it

[3] One can show the inference dimension here is 5.

into the desired learning algorithm using a simple boosting argument.

While the boosting part is rather standard, showing that low inference dimension implies a weak confident learner relies on a symmetrization argument. This symmetrization argument can be replaced by a more standard sample compression argument, however this would result in a suboptimal query complexity bound.

We get the following corollary of Theorem I.5 in the statistical setting:

Corollary I.6. *Let k denote the inference dimension of (X, H).*

1) *Let $n(\epsilon, \delta)$ denote the passive sample complexity of learning (X, H) with error ϵ and confidence $1 - \delta$. There is an algorithm that learns (X, H) with sample complexity $n(\epsilon, \delta)$ and query complexity $O\big(k \log k \log(n(\epsilon, \delta))\big)$.*
2) *Any algorithm that learns (X, H) with error at most $\epsilon = {}^{1}\!/_{k}$ and confidence at least ${}^{5}\!/_{6}$ must use at least $\Omega\big({}^{1}\!/_{\epsilon}\big)$ queries for some realizable distribution.*

This Corollary follows from Corollaries I.8 and III.6.

A dichotomy: The passive sample complexity of (X, H) is $n(\epsilon, \delta) = \Theta(\frac{d + \log(1/\delta)}{\epsilon})$, where d is the VC-dimension of H [24]. Thus the above corollary implies a dichotomy: if the inference dimension is finite then the query complexity is logarithmic in $1/\epsilon$, and if it is infinite then the query complexity is $\Omega(1/\epsilon)$.

The results presented in the previous section regarding learning half spaces with comparison and label queries are derived by analyzing the inference-dimension of the relevant classes, which we sketch next.

Sketch of upper bounding the inference dimension of half spaces: Our upper bounds in terms of margin and bit-complexity follow a similar outline. We next give a rough sketch of the arguments in order to convey their flavor.

Consider the case where every instance $x \in X$ has bounded bit-complexity, say $X \subset [N]^d$ for some bounded $N \in \mathbb{N}$. We wish to show that for a sufficiently large $Y \subseteq X$, and every half space $c = \text{sign}(f)$ there is some $x \in Y$ such that

$$Y \setminus \{x\} \underset{f}{\Longrightarrow} x.$$

By removing at most half of the elements of Y, we may assume that c is constant on Y, without loss of generality assume that $c(x) = +1$, for every $x \in Y$. Let $x_1, x_2, \ldots$ be an ordering of the elements of Y such that

$$f(x_1) \le f(x_2) \le \ldots$$

The first observation is that it suffices to show that there exists i_0 such that $x_{i_0} - x_1$ is a nonnegative linear combination of the $x_i - x_j$, where $i > j$.

The existence of such an i_0 is achieved by a pigeon hole argument showing that if Y is sufficiently large then there are two distinct linear combinations of the $x_{i+1} - x_i$'s with coefficients from $\{0, 1\}$ that yield the same vector:

$$\sum_{i=1}^{k/2} \beta_i (x_{i+1} - x_i) = \sum_{i=1}^{k/2} \gamma_i (x_{i+1} - x_i),$$

Then a short calculation yields that the maximal j such that $\beta_j \neq \gamma_j$ serves as the desired i_0.

The upper bound in terms of margin is technically more involved. Instead of nonnegative linear combinations, one can consider linear combinations such that every coefficient is at least $-\gamma$, where γ is the margin, and the pigeon hole argument is replaced by a volume argument.

Sketch of lower bounding the inference dimension of half spaces: Our lower bounds are based on embedding the class $\big\{\emptyset, \{i\} : 1 \le i \le n\big\}$ as half spaces in a way that the $n+1$ half spaces induce the same ordering on the n points in terms of distance from the boundary. Comparison queries are useless for such an embedding, which implies that the inference dimension is at least n, Indeed, consider the half space $c_\emptyset$ that corresponds to the empty-set; then for any subset Y of at most $n-1$ points $c_\emptyset$ is indistinguishable from $c_{\{i\}}$, the half space corresponding to $\{i\}$, where $i \notin Y$.

3) General framework: We next describe how the notion of inference-dimension, as well as Theorem I.5 extend to settings where the additional queries are not necessarily comparison queries.

Consider an interactive model, where the learning algorithm is allowed to use additional queries from a prescribed set of queries $\mathcal{Q}$. More formally, let S be the unlabeled input sample. In pool-based active learning, the algorithm may query the label of any point in S. Here, the algorithm is allowed to use additional queries from a set $\mathcal{Q} = \mathcal{Q}(S)$ (we stress that the allowable queries depend on the input sample). For example, in the setting discussed in the previous section, $\mathcal{Q}(S)$ contains all comparison queries among points in S. Another example, which is used by crowd-sourcing algorithms [25] involve 3-wise queries of the form "Is x_2 more similar to x_1 than to x_3?".

Upper bound: The next "boosting result" generalizes the upper bound from Theorem I.5 and shows that if there are not too many queries in $\mathcal{Q}(S)$, or alternatively if there is an algorithm that infers the answers to the queries in $\mathcal{Q}(S)$ using a few queries, then it is possible to reveal all labels using a logarithmic number of queries:

Theorem I.7 (Restatement of Theorem III.2)**.** *Let k denote the inference dimension of (X, H). Assume that there is an algorithm that, given a realizable sample S of size n as input, uses at most $q(n)$ queries and infers the answers to all queries in $\mathcal{Q}(S)$ and all labels in S. Then there is a randomized algorithm that infers all the labels in S using at most*

$$2q(4k)\log n$$

queries in expectation.

For example, in the setting of comparison queries, $q(n) = n + n \log n$ queries suffice to infer all comparison queries and label queries in $\mathcal{Q}(S)$, and thus, the upper bound in Theorem I.5 follows from the above theorem.

An interactive algorithm that infers all labels using a few queries can be combined with any passive learner by first using the interactive learner to reveal all labels of the input sample, and then apply the passive learner on the labeled sample. Thus, we get:

Corollary I.8. *Let (X, H) be as in Theorem III.2, and let $n(\epsilon, \delta)$ denote the (passive) sample complexity of learning (X, H) with error ϵ and confidence $1 - \delta$. Then there exists an algorithm that learns H with sample complexity $n(\epsilon, \delta)$, and query complexity $O(q(4k)\log n(\epsilon, \delta))$.*

Lower bound: The next lower bound on the query complexity generalizes the one from Theorem I.5. It demonstrates a property of comparison and label queries which suffices for the lower bound in Theorem I.5 to hold. Call an additional query *t-local*, if its answer is determined by $f(x_1), \ldots, f(x_t)$ for some $x_1, \ldots, x_t \in X$. For example, every label-query is 1-local and every comparison query is 2-local. The following Theorem extends Theorem I.5 to this setting:

Theorem I.9 (Restatement of Theorem III.5)**.** *Assume that the inference dimension of (X, H) is $> k$. Assume that every additional query is t-local, and that for every sample S the set of allowable queries $\mathcal{Q}(S)$ is the set of all queries that are determined by subsets (of size at most t) of S. Then any algorithm that reveals the labels of any realizable sample of size k must use $\Omega(k/t)$ queries in the worst-case.*

In the statistical setting, we get the following Corollary:

Corollary I.10 (Restatement of Corollary III.6)**.** *Set $\epsilon = \frac{1}{k}$, $\delta = \frac{1}{6}$. Then any algorithm that learns (X, H) with error ϵ and confidence $1 - \delta$ must use $\Omega(1/t\epsilon)$ queries.*

C. Related work

Studying statistical learning where the learner has access to additional queries was considered by various works. A partial list includes: [19], [26]–[36]. Many of these focus on the case where the additional queries are *membership queries*.

In the context of active learning, which is considered in this paper, [10] considered additional queries of two types: *Class conditional queries* and *Mistake queries*, in the first type the learner provides the annotator with a list of examples, and a label and asks her to point out an example in this list with the given label. In the second type, the learner gives the oracle a list of examples with proposed labels and she replies whether it is correct or points out a mistake. Note that these queries may have more than two answers, which is different than binary queries, which are considered in this paper.

[37] give an active learning algorithm for clustering using pairwise similarity queries, and [38] gives a clustering algorithm that uses queries that ask whether two elements belong to the same cluster. [39] consider clustering in an interactive setting where the algorithm may present the annotator a clustering of a $O(1)$ size subset of the domain, and the annotator replies whether the target-clustering agrees with this partition, and points out a difference in case it does not.

[40] give an active ranking algorithm from pairwise comparisons. They consider a setting that bears resemblance with ours: their goal is to find the ranking among a sample of points in $\mathbb{R}^d$, where the ranking is determined according to the euclidean distance from a fixed, unknown reference point. They present an algorithm that use an expected number of $O(d \log n)$ comparisons when the ranking is chosen uniformly at random.

Recently, [41] considered a similar setting to ours. They also study active classification with additional comparison queries, but they focus on minimizing the total number of label queries, while the number of comparison queries can be large (more than $1/\epsilon$, where ϵ is the error). In contrast, we study when it is possible to achieve total number of queries which is logarithmic in $1/\epsilon$.

II. Preliminaries

Basic definitions.: A hypothesis class is a pair (X, H), where X is a set, and H is a class of functions $h : X \to \{\pm 1\}$. Each function $h : X \to \{\pm 1\}$ is called *a hypothesis*, or *a concept*. In this paper we study classes $H = H_{\mathcal{F}}$ of the form

$$H = \{\text{sign}(f) : f \in \mathcal{F}\},$$

where $\mathcal{F} = \{f : X \to \mathbb{R}\}$ is a class of real-valued functions, and $\text{sign}(f)(x) = \text{sign}\big(f(x)\big) \in \{\pm 1\}$ is equal $+1$ if and only if $f(x) \geq 0$. For example, when $\mathcal{F}$ is the class of $\mathbb{R}^d \to \mathbb{R}$ affine functions then $H_{\mathcal{F}}$ is the class of d-dimensional half spaces. Other examples include neural-nets, low degree polynomials, and more. The reason we "remember" the underlying class $\mathcal{F}$ is because we will use it to define comparison queries.

An *example* is a pair $(x, y) \in X \times \{\pm 1\}$. A *labeled sample* $\bar{S}$ is a finite sequence of examples. We denote by S the *unlabeled sample* underlying $\bar{S}$ that is obtained by removing the labels from the examples in $\bar{S}$. We will sometimes abuse notation and treat S as a subset of X (formally it is a sequence). Given a distribution D on $X \times \{\pm 1\}$, the (expected) *loss* of a hypothesis h is defined by:

$$L_D(h) \triangleq \mathop{\mathbb{E}}_{(x,y)\sim D}\big[1_{h(x)\neq y}\big].$$

Given a labeled sample $\bar{S}$, the *empirical loss* of h is defined by:

$$L_{\bar{S}}(h) \triangleq \frac{1}{|\bar{S}|}\sum_{(x,y)\in\bar{S}} 1_{h(x)\neq y}.$$

A distribution D on $X \times \{\pm 1\}$ is *realizable* by H if there exists $c \in H$ with $L_D(c) = 0$. We will sometimes refer to such a c as the "learned concept". A labeled sample $\bar{S}$ is *realizable* by H if there exists $c \in H$ with $L_{\bar{S}}(c) = 0$.

Passive learning (PAC learning).: A learning algorithm is an (efficiently computable) mapping that gets a sample as an input and outputs a hypothesis. An algorithm A learns H if there exists a sample complexity bound $n(\epsilon, \delta)$, such that for every realizable distribution D, given a labeled sample $\bar{S}$ of size $n \geq n(\epsilon, \delta)$, A outputs $h = A(\bar{S})$ such that:

$$\Pr_{\bar{S}\sim D^n}\big[L_D(h) > \epsilon\big] \leq \delta.$$

The parameter ϵ is called the error of the algorithm, and $1 - \delta$ is called the confidence of the algorithm We will assume throughout that a learning algorithm for H also receives ϵ, δ as part of the input and can compute $n(\epsilon, \delta)$.

A. Active learning

It is helpful to recall the framework of active learning before extending it by allowing additional queries. A (pool-based) active learning algorithm has an access to the unlabeled sample S underlying the input labeled sample $\bar{S}$. It queries the labels of a subsample of it, and outputs a hypothesis. The choice of which subsample A queries may be adaptive. Each active learning algorithm is associated with two complexity measures: (i) the sample-complexity $n(\epsilon, \delta)$, is the number of examples required to achieve error at most ϵ with confidence at least $1-\delta$ (like in the passive setting), and (ii) the query-complexity (also called label-complexity) $q(\epsilon, \delta)$, is the number of queries it makes.

In the process of active learning, it is natural to distinguish between points whose label can be inferred and points for which there is uncertainty concerning their label. It is therefore convenient to consider partial hypotheses. A *partial hypothesis* is a partially labeled hypothesis $h : X \to (Y \cup \{?\})$, where if $h(x) =$ "?" it means that h abstains from labeling x. We extend the $0/1$ loss function to "?", such that abstaining is always treated as a mistake. The *coverage* of h with respect to a distribution D is defined as

$$C_D(h) \triangleq \Pr_{x\sim D}\big[h(x) \neq ?\big],$$

and the *empirical-coverage* of h with respect to a sample S is defined as

$$C_S(h) \triangleq \frac{\big|\{x \in S_X : h(x) \neq ?\}\big|}{|S|}.$$

Since abstaining counts as error it follows that $C_D(h) \leq 1 - L_D(h)$ for every partial hypothesis h, and every distribution D.

Confident algorithms.: A learning algorithm is *confident* if it outputs a partial hypothesis that is correct on all points where it does not abstain:

Definition II.1 (Confident learning algorithm)**.** *A learning algorithm A is* confident *with respect to a class (X, H) if it satisfies the following additional requirement. For every realizable labeled sample $\bar{S}$ that is consistent with a learned concept $c \in H$, the output hypothesis $h \triangleq A(\bar{S})$ satisfies that whenever $h(x) \neq ?$ then $h(x) = c(x)$.*

Thus, if A is confident then $C_D(h) = 1 - L_D(h)$ for every distribution D, where $h = A(\bar{S})$. Therefore, in the context of confident learners we will only discuss their coverage, and omit explicit reference to their error. For example a class (X, H) is learned by a confident learner A if there exists a sample complexity $n(\epsilon, \delta)$ such that for every realizable distribution D, given a labeled sample $\bar{S}$ of size $n \geq n(\epsilon, \delta)$, A outputs $h = A(\bar{S})$ such that

$$\Pr_{\bar{S}\sim D^n}\big[C_D(h) < 1 - \epsilon\big] \leq \delta.$$

B. Interactive learning with additional queries

Consider an extension of the active learning setting, by allowing the learning algorithm to use additional queries from a prescribed set of queries $\mathcal{Q}$. An additional query is modeled as a boolean function $q : \mathcal{F} \to \{True, False\}$. We stress that the query may depend on the function f that underlies the learned concept $c = \text{sign}(f)$ (e.g. comparison queries, see below). In this setting, given an input sample $\bar{S}$, the algorithm is given access to S, the unlabeled sample underlying $\bar{S}$, and is allowed to:

- query the label of any of point in S,
- query an additional query q from a prescribed set of queries $\mathcal{Q}(S)$.

We stress that the set of allowable queries $\mathcal{Q}(S)$ depend on the input sample $\bar{S}$. For example, a comparison query on x_1, x_2 is the query "$f(x_1) \leq f(x_2)$?", where $c = \text{sign}(f)$ is the learned concept. An *answered query* in a pair (q, b), where q is a query, and b is a possible answer to q. For example, $\big($"$f(x_1) \leq f(x_2)$?"$, True\big)$ is an answered comparison query, and $\big($"$c(x_1) = $?"$, -1\big)$ is an answered label query. The standard notions of *version space* and *agreement set* are naturally extended to this context: let

$$\bar{Q} = \big((q_1, b_1), (q_2, b_2), \dots (q_m, b_m)\big)$$

be a sequence of answered queries. Define the *version space*, denoted by $V(\bar{Q})$, as the set of hypotheses in H that are consistent with $\bar{Q}$:

$$V(\bar{Q}) \triangleq \big\{h \in H : q_i(h) = b_i,\ i = 1, \dots, m\big\},$$

and the confidence region, $\text{Conf}(\bar{Q})$, as the agreement set of $V(\bar{Q})$:

$$\text{Conf}(\bar{Q}) \triangleq \big\{x \in X : \text{hypotheses in } V(\bar{Q}) \text{ agree on } x\big\}.$$

III. Inference Dimension

Let (X, H) be a hypothesis class, where $H = H_{\mathcal{F}}$ for some class of real functions $\mathcal{F}$, and let $\mathcal{Q}$ be a set of additional queries. For $S \subseteq X$, $x \in X$, $f \in \mathcal{F}$ and $c = \text{sign}(f) \in H$, let

$$S \underset{f}{\Longrightarrow} x$$

denote the statement that there exists a sequence $\bar{Q}$ of answered label queries of S and/or additional queries from $\mathcal{Q}(S)$ that determine the label of x, when the learned concept is c. Namely, that $x \in \text{Conf}(\bar{Q})$.

Definition III.1 (Inference dimension)**.** *The inference dimension of (X, H) is the minimal number k such that for every $S \subseteq X$ of size k, and every $c \in H$ there exists $x \in S$ such that*

$$S \setminus \{x\} \underset{f}{\Longrightarrow} x.$$

If no such k exists then the inference dimension of (X, H) is defined as ∞.

A. Upper bound

Theorem III.2 (Boosting)**.** *Let k denote the inference dimension of (X, H). Assume that there is an algorithm that, given a realizable sample $\bar{S}$ of size n as input, uses at most $q(n)$ queries and infers the answers to all queries in $\mathcal{Q}(S)$ and all labels in $\bar{S}$. Then there is a randomized algorithm that infers all the labels in $\bar{S}$ using at most*

$$2q(4k)\log n$$

queries in expectation.

We prove Theorem III.2 in two steps: (i) First, Lemma III.3 shows that (X, H) has a weak confident learner A that, given a realizable input sample of size $4k$, uses at most $q(4k)$ queries, and outputs a partial hypothesis h with coverage $1/2$. (ii) Then, we show that the labels of a given sample $\bar{S}$ of size n are revealed after applying A on roughly $\log n$ random subsamples of $\bar{S}$.

Lemma III.3 (Weak confident-learning)**.** *Let k denote the inference dimension of (X, H). Then there exists a confident learner for (X, H) that is defined on input samples of length $4k$, makes at most $q(4k)$ queries, and has coverage $\geq 1/2$ with probability $\geq 1/2$. That is, for any distribution D over X,*

$$\Pr_{S \sim D^{4k}}\big[C_D(h) \geq 1/2\big] \geq 1/2,$$

where h is the output hypothesis of the algorithm.

Proof: The learner is defined as follows. Given a realizable input sample $S = \big((x_i, c(x_i))\big)_{i=1}^{4k}$, from D^{4k}, the algorithm infers the answers to all queries in $\mathcal{Q}(S)$ and all labels in $\bar{S}$ (by assumption, this can be done with $q(4k)$ queries). It outputs the partial hypothesis h, which labels any x whose label can be inferred from the queries. Namely:

$$h(x) \triangleq \begin{cases} c(x) & S \underset{f}{\Longrightarrow} x \\ ? & \text{otherwise} \end{cases}$$

We next claim that the expected coverage of h, $C_D(h)$, is at least $3/4$. This implies that $\Pr_S[C_D(h) \geq 1/2] \geq 1/2$, which shows that the learning algorithm is a weak confident learner.

To this end we use the following observation.

Observation III.4. *For any set Y of size $4k+1$, there are $x_{i_1}, \dots, x_{i_{3k+1}} \in Y$ such that for all $1 \leq j \leq 3k+1$ it holds that*

$$Y \setminus \{x_{i_j}\} \underset{f}{\Longrightarrow} x_{i_j}.$$

Proof: This follows since the inference dimension of (X, H) is k. Assume we already constructed $x_{i_1}, \dots, x_{i_{j-1}}$ for $j \leq 3k+1$. Let $Y' = Y \setminus$

$\{x_{i_1}, \ldots, x_{i_{j-1}}\}$. As $|Y'| \geq k$, there exists $x_{i_j} \in Y'$ such that $Y' \setminus \{x_{i_j}\} \underset{f}{\Longrightarrow} x_{i_j}$. But then also $Y \setminus \{x_{i_j}\} \underset{f}{\Longrightarrow} x_{i_j}$. ■

Next, we show that this observation implies that $\mathbb{E}_S[C_D(h)] \geq 3/4$. Clearly, we have that

$$\begin{aligned}&\mathop{\mathbb{E}}_{S}\big[C_D(h)\big]\\ =&\Pr_{(x_1,x_2,\ldots,x_{4k+1})\sim D^{4k+1}}\big[\{x_1,\ldots,x_{4k}\} \underset{f}{\Longrightarrow} x_{4k+1}\big].\end{aligned}$$

Letting $T = \{x_1, x_2, \ldots, x_{4k+1}\}$, this is the probability that $T\backslash\{x_{4k+1}\} \underset{f}{\Longrightarrow} x_{4k+1}$. However, by symmetry, this is the same as the probability that $T\backslash\{x_i\} \underset{f}{\Longrightarrow} x_i$ for any $1 \leq i \leq 4k+1$. Taking the average over i, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{S}\big[C_D(h)\big] =& \mathop{\mathbb{E}}_{T}\left[\frac{1}{4k+1}\big|\{i : T\backslash\{x_i\} \underset{f}{\Longrightarrow} x_i\}\big|\right]\\ \geq& \frac{3k+1}{4k+1} \geq \frac{3}{4}.\end{aligned}$$

■

Proof of Theorem III.2:

Let A denote the weak confident learner from Lemma III.3. Let $\bar{S}$ be a realizable sample, corresponding to an unknown concept $c \in H$. Our goal is to fully recover the labels of $\bar{S}$.

The algorithm proceed in iterations $t = 1, 2, \ldots$, At each iteration it applies A on a subsample of size $4k$ of $\bar{S}$. Let h_t denote the output hypothesis of A on iteration t, and let

$$DIS_t = \{x : h_s(x) = ? \text{ for all } s \leq t\}.$$

Since A is confident it follows for any point $x \notin DIS_t$, the label $c(x)$ is equal to $h_s(x)$ for any $h_s(x)$ such that $h_s(x) \neq ?$. As long as $DIS_t \cap S \neq \emptyset$, perform the following update step.

```
Update step at time $t$:
  (1) Let $D_t$ be the uniform distribution over $DIS_t \cap S$. Sample $\bar{R}_t \sim (D_t)^{4k}$.
  (2) Apply $A$ to $R_t$, the unlabeled sample underlying $\bar{R}_t$.
  (3) Let $h_t = A(\bar{R}_t)$ be the confident partial hypothesis that $A$ outputs on $\bar{R}_t$.
  (4) Compute $e_t = C_{D_t}[h_t] = \Pr_{x\sim D_t}[h_t(x) \neq ?]$.
  (5) If $e_t < 1/2$ then go back to step (1).
      Otherwise set $t \leftarrow t + 1$ and continue.
```

Since A is confident, it follows that once $DIS_t \cap S = \emptyset$ then all the labels of $\bar{S}$ are revealed.

Query-complexity.: In order to analyze the query-complexity of the algorithm, first observe that since $\Pr[e_t \geq 1/2] \geq 1/2$ then in expectation, we proceed to the next iteration after at most two samples of $\bar{R}_t$. Next, if $e_t \geq 1/2$ then by definition $|DIS_{t+1} \cap S_X| \leq |DIS_t \cap S_X|/2$. Thus, we only apply the update step at most $t_{max} \leq 2\log n$ many times. It follows that the expected query-complexity is at most $2q(4k)\log n$. ■

Computational complexity.: The algorithm derived in Theorem III.2 has expected running time of $O(T_{\text{update}} \log n)$, where T_{update} is the running time of the update step. In every update step the algorithm makes $q(4k)$ queries and determines e_t, by checking for each unlabeled point, whether its label can be inferred by the queries performed in this step. Assume that testing this for each point takes take T_{infer}. So,

$$T_{\text{update}} = O\left(q(4k) + n \cdot T_{\text{infer}}\right)$$

and the total running time is

$$T_{\text{total}} = O\left(\left(q(4k) + n \cdot T_{\text{infer}}\right)\log n\right).$$

For example, when the hypothesis class is half spaces in $\mathbb{R}^d$, and the the set of additional queries is comparisons, the total running time is polynomial in n. This is since $q(4k) = O(k \log k)$ (by sorting), and since checking if the label of a point is inferred by a set of label and comparison queries can be phrased as a linear program and solved in polynomial time.

B. Lower bound

Next, we show that if the inference dimension is large then many queries are needed to infer all the labels. We further assume that every query is t-local, in the sense that it depends on $f(x_1), \ldots, f(x_t)$ for some $x_1, \ldots, x_t \in X$. We set of allowable queries $\mathcal{Q}(S)$ to be all queries that are associated with subsets of S of size t.

Let (X, H) be a hypothesis class with inference dimension $> k$, for some $k \geq 3$. This means that there exists $Z \subseteq X$ of size k and a concept $c \in H$ such that for every $z \in Z$ there is $c_z \in C$ with $c_z(z) \neq c(z)$, but $c_z(x) = c(x)$ for all $x \in Z \setminus \{z\}$, and moreover, $q(c) = q(c_z)$ for every query $q \in \mathcal{Q}\big(Z \setminus \{z\}\big)$.

Theorem III.5. *Any algorithm that reveals the labels of any realizable sample of size k must use at least k/t queries in the worst-case.*

We omit the proof of this theorem here, the proof is available on full version [20].

Corollary III.6. *Let $\epsilon = \frac{1}{k}, \delta = \frac{1}{6}$, and let D be the uniform distribution over Z. Then any learning algorithm that makes less than $\frac{1}{t\epsilon}$ queries suffers a loss of ϵ, with probability at least δ.*

IV. Interactive learning of half spaces with comparison-queries

In this section we restrict our attention to the class $H_d = \{\text{sign}(f) : f : \mathbb{R}^d \to \mathbb{R}\}$ of half spaces in $\mathbb{R}^d$, where for simplicity of exposition we consider linear functions f (these correspond to homogeneous half spaces). Our results extend to the non-homogeneous case, as non-homogeneous half spaces in dimension d can be embedded as homogeneous half spaces in dimension $d+1$. The additional queries allowed are comparison queries. That is, a label query returns the answer to $\text{sign}(f(x))$ and a comparison query returns the answer to $f(x_1) \geq f(x_2)$.

In Subsection IV-A we present our upper bounds on the query complexity, under two natural conditions: small bit complexity, or large margin. In Subsection IV-B we present lower bounds showing that these conditions are indeed necessary for obtaining query complexity sub-linear in the sample complexity.

A. Upper bounds

1) Bit-complexity: Here we show that if the examples can be represented using a bounded number of bits then comparison-queries can reduce the query-complexity. We formalize bounded bit-complexity by assuming that $X = [N]^d$, where $[N] = \{0, \ldots, N\}$. Note that each example can be represented by $B = d \log N$ bits. We provide a bound on the query-complexity that depends efficiently on$d$ and $\log N$. Variants of the arguments we use apply to other standard ways of quantifying bounded bit-complexity.

Theorem IV.1. *Consider the class $\big([N]^d, H_d\big)$. There exists an algorithm that reveals the labels of any realizable input sample of size n using at most $O(k \log k \log n)$ label/comparison-queries in expectation, where $k = O\big(d \log(Nd)\big)$.*

As a consequence it follows that the hypothesis class $\big([N]^d, H_d\big)$ is learnable with

$$\text{sample complexity } \tilde{O}\big({}^{d}\!/\!_{\epsilon}\big) \text{ and query-complexity } \tilde{O}\big(d \log(N) \log({}^{1}\!/\!_{\epsilon})\big),$$

where the $\tilde{O}$ notation suppresses lower order terms and the usual $\log(1/\delta)$ dependence.

In order to prove the above theorem, we use Theorem I.5 that reduces it to the following lemma. We sketch a proof outline here, a complete proof is availabe on full version [20].

Lemma IV.2. *Let k such that $2^{k/2} > 2(kN+1)^d$. Then the inference dimension of the class $\big([N]^d, H_d\big)$ is at most k. In particular, it is at most $16d \log(4Nd)$.*

2) Minimal-ratio and margin: Let $X \subseteq \mathbb{R}^d$ and let $c = \text{sign}(f) \in H_d$. The *minimal-ratio* of X with respect to c is defined by

$$\eta = \eta(c, X) \triangleq \frac{\min_{x \in X} |f(x)|}{\max_{x \in X} |f(x)|}.$$

Here we show that it is possible to reveal all labels using at most $\tilde{O}\Big(d \log({}^{1}\!/\!_{\eta}) \log n\Big)$, where the minimal-ratio is η. Note that the minimal-ratio is invariant under scaling and that it is upper bounded by the margin:

Claim IV.3. *Let η be the minimal-ratio of X with respect to c, let $\rho = \max_{x \in X} ||x||_2$, and let γ be the margin of X with respect to c. Then*

$$\frac{\gamma}{\rho} \leq \eta.$$

Thus, the upper bound in Theorem IV.4 below applies when the minimal-ratio is replaced by the standard margin parameter ${}^{\gamma}\!/\!_{\rho}$.

Note that there are cases where $\eta >> {}^{\gamma}\!/\!_{\rho}$. For example, assume $X = \{e_1, \ldots, e_d\}$ is the standard basis and $c_w \in H_d$ is determined by the normal $w = \frac{1}{\sqrt{d}}(+1, -1, +1, -1, \ldots)$. In this case, ${}^{\gamma}\!/\!_{\rho} = {}^{1}\!/\!_{\sqrt{d}} << 1 = \eta$.

We next state and prove the upper bound. Let $X \subseteq \mathbb{R}^d$, and let $H_{d,\eta} \subseteq H_d$ be the set of all half spaces with minimal-ratio at least η with respect to X.

Theorem IV.4. *Consider the class $(X, H_{d,\eta})$. There exists an algorithm that reveals the labels of any realizable input sample of size n using at most $O\big(k \log k \log n\big)$ label/comparison-queries in expectation, where $k = O\big(d \log(d) \log({}^{1}\!/\!_{\eta})\big)$.*

As a consequence it follows that the hypothesis class $(X, H_{d,\eta})$ is learnable with

$$\text{sample complexity } \tilde{O}\big({}^{d}\!/\!_{\epsilon}\big) \text{ and query-complexity } \tilde{O}\big(d \log({}^{1}\!/\!_{\eta}) \log({}^{1}\!/\!_{\epsilon})\big),$$

As before, the $\tilde{O}$ notation suppresses lower order terms and the usual $\log(1/\delta)$ dependence.

The above theorem is a corollary of Theorem I.5 via the following lemma, which upper bounds the inference dimension of the class $(X, H_{d,\eta})$.

Lemma IV.5. *Let k such that $(k/2+1)^d < 2^{k/2}(\eta/6)^d$. Then the inference dimension of the class $\big(X, H_{d,\eta}\big)$ is at most k. In particular, it is at most $10d \log(d+1) \log({}^{2}\!/\!_{\eta})$.*

B. Lower bounds

In this Section we show that (in the worst-case) comparison-queries yield no advantage when the bit complexity is large in dimension $d \geq 3$, or when the dimension is large even if the margin is large.

1) Dimension $d \geq 3$: We show that (in the worst-case) comparison-queries do not yield a significant saving in query complexity for learning half spaces, already in $\mathbb{R}^3$. This is tight since, as discussed in the introduction, in $\mathbb{R}^2$ comparison-queries yield an exponential saving.

Theorem IV.6. *Consider the class $(\mathbb{R}^3, H_3)$ of half spaces in $\mathbb{R}^3$, Any algorithm that reveals the labels of any realizable sample of size n must use $\Omega(n)$ comparison/label queries in the worst-case.*

In the statistical setting, we get that

Corollary IV.7. *Let $\epsilon > 0$. Then any algorithm that learns $(\mathbb{R}^3, H_3)$ with error ϵ and confidence at least $5/6$ must use $\Omega(1/\epsilon)$ comparison/label queries on some realizable distributions.*

We derive these statements by showing that the inference dimension of $(\mathbb{R}^3, H_3)$ is ∞. Then, Theorem IV.6 and Corollary IV.7 follow by plugging $t = 2$ in Theorem III.5 and Corollary III.6 respectively. (Note that comparison queries are 2-local, and thus $t = 2$). We skip the proof here, it appears in full version [20].

Theorem IV.8. *The inference dimension of $(\mathbb{R}^3, H_3)$ is ∞.*

2) Margin: We show here that, in the worst-case, comparison-queries do not yield a significant saving in query complexity for learning half spaces, even if it is guaranteed that the margin is large, say at least $1/8$.

Theorem IV.9. *For every n there is a class (X, H), where $X \subseteq \mathbb{R}^{n+1}$, and $H \subseteq H_{n+1}$ contains all the half spaces with margin at least $1/8$ such that the following holds: any algorithm that reveals the labels of any realizable sample of size n must use $\Omega(n)$ comparison/label queries in the worst-case.*

In the statistical setting, we get that

Corollary IV.10. *For every $\epsilon > 0$, there is n and a class (X, H), where $X \subseteq \mathbb{R}^{n+1}$, and $H \subseteq H_{n+1}$ contains all the half spaces with margin at least $1/8$ such that the following holds: any algorithm that learns (X, H) with error ϵ and confidence at least $5/6$ must use $\Omega(1/\epsilon)$ comparison/label queries on some realizable distributions.*

We derive these statements by establishing the existence of classes with large margin and large infrence dimension. Then, Theorem IV.9 and Corollary IV.10 follow by plugging $t = 2$ in Theorem III.5 and Corollary III.6 respectively.

Theorem IV.11. *For every n, there is a set of n unit vectors $X = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^{n+1}$ such that the class (X, H) has inference dimension at least n, where H contains all half spaces with margin at least $1/6$ with respect to X.*

Acknowledgements

This work has benefited from various discussions during the special program on *Foundations of Machine Learning* that took place at the *Simons Institute for the Theory of Computing*, in Berkeley. In particular, the authors would like to thank Sanjoy Dasgupta for inspiring the focus on comparison queries.

Kane is supported by NSF CAREER award 1553288. Lovett, Moran and Zhang are supported by NSF CAREER award 1350481 and NSF CCF award 1614023.

References

[1] A. McCallum, K. Nigam *et al.*, "Employing em and pool-based active learning for text classification." in *ICML*, vol. 98, 1998, pp. 350–358.

[2] S. Dasgupta, "Analysis of a greedy active learning strategy," in *Advances in neural information processing systems*, 2005, pp. 337–344.

[3] M.-F. Balcan, A. Broder, and T. Zhang, "Margin based active learning," in *Proceedings of the 20th annual conference on Learning theory*. Springer-Verlag, 2007, pp. 35–50.

[4] S. Hanneke, "A bound on the label complexity of agnostic active learning," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 353–360.

[5] S. Dasgupta, D. J. Hsu, and C. Monteleoni, "A general agnostic active learning algorithm," in *Advances in neural information processing systems*, 2008, pp. 353–360.

[6] M. Balcan, A. Beygelzimer, and J. Langford, "Agnostic active learning," *J. Comput. Syst. Sci.*, vol. 75, no. 1, pp. 78–89, 2009.

[7] A. Beygelzimer, D. J. Hsu, J. Langford, and T. Zhang, "Agnostic active learning without constraints," in *24th Annual Conference on Neural Information Processing Systems.*, 2010, pp. 199–207.

[8] M. Balcan, S. Hanneke, and J. W. Vaughan, "The true sample complexity of active learning," *Machine Learning*, vol. 80, no. 2-3, pp. 111–139, 2010.

[9] V. Koltchinskii, "Rademacher complexities and bounding the excess risk in active learning," *Journal of Machine Learning Research*, vol. 11, pp. 2457–2485, 2010.

[10] M. Balcan and S. Hanneke, "Robust interactive learning," *CoRR*, vol. abs/1111.1422, 2011.

[11] S. Hanneke and L. Yang, "Surrogate losses in passive and active learning," *CoRR*, vol. abs/1207.3772, 2012.

[12] R. El-Yaniv and Y. Wiener, “Active learning via perfect selective classification,” *Journal of Machine Learning Research*, vol. 13, pp. 255–279, 2012.

[13] M. Balcan and P. M. Long, “Active and passive learning of linear separators under log-concave distributions,” in *The 26th Annual Conference on Learning Theory, 2013*, 2013, pp. 288–316.

[14] A. Gonen, S. Sabato, and S. Shalev-Shwartz, “Efficient active learning of halfspaces: an aggressive approach,” *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2583–2615, 2013.

[15] R. Urner, S. Wulff, and S. Ben-David, “Plal: Cluster-based active learning,” in *The 26th Annual Conference on Learning Theory*, 2013, pp. 376–397.

[16] C. Zhang and K. Chaudhuri, “Beyond disagreement-based agnostic active learning,” in *Advances in Neural Information Processing Systems*, 2014, pp. 442–450.

[17] Y. Wiener, S. Hanneke, and R. El-Yaniv, “A compression technique for analyzing disagreement-based active learning,” *Journal of Machine Learning Research*, vol. 16, pp. 713–745, 2015.

[18] C. Berlind and R. Urner, “Active nearest neighbors in changing environments,” in *Proceedings of the 32nd International Conference on Machine Learning*, 2015, pp. 1870–1879.

[19] K. Lang and E. Baum, “Query learning can work poorly when a human oracle is used,” in *Proceedings of the IEEE International Joint Conference on Neural Networks*, 1992, pp. 335–340.

[20] D. M. Kane, S. Lovett, S. Moran, and J. Zhang, “Active classification with comparison queries,” *arXiv preprint arXiv:1704.03564*, 2017.

[21] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica*, vol. 4, no. 4, pp. 373–396, 1984.

[22] P. M. Vaidya, “A new algorithm for minimizing convex functions over convex sets,” *Math. Program.*, vol. 73, pp. 291–341, 1996.

[23] S. Hanneke and L. Yang, “Minimax analysis of active learning,” *Journal of Machine Learning Research*, vol. 16, pp. 3487–3602, 2015.

[24] S. Hanneke, “The optimal sample complexity of PAC learning,” *CoRR*, vol. abs/1507.00473, 2015.

[25] O. Tamuz, C. Liu, S. J. Belongie, O. Shamir, and A. Kalai, “Adaptively learning the crowd kernel,” in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 2011, pp. 673–680.

[26] D. Angluin, “Queries and concept learning,” *Machine Learning*, vol. 2, no. 4, pp. 319–342, 1987.

[27] E. B. Baum, “Neural net algorithms that learn in polynomial time from examples and queries,” *IEEE Trans. Neural Networks*, vol. 2, no. 1, pp. 5–19, 1991.

[28] G. Turán, “Lower bounds for PAC learning with queries,” in *Proceedings of the Sixth Annual Conference on Computational Learning Theory, COLT 1993*, 1993, pp. 384–391.

[29] J. C. Jackson, “An efficient membership-query algorithm for learning dnf with respect to the uniform distribution,” *Journal of Computer and System Sciences*, vol. 55, no. 3, pp. 414 – 440, 1997.

[30] S. Kwek and L. Pitt, “PAC learning intersections of halfspaces with membership queries,” *Algorithmica*, vol. 22, no. 1/2, pp. 53–75, 1998.

[31] A. Blum, P. Chalasani, S. A. Goldman, and D. K. Slonim, “Learning with unreliable boundary queries,” *J. Comput. Syst. Sci.*, vol. 56, no. 2, pp. 209–222, 1998.

[32] N. H. Bshouty, J. C. Jackson, and C. Tamon, “More efficient pac-learning of dnf with membership queries under the uniform distribution,” *J. Comput. Syst. Sci.*, vol. 68, no. 1, pp. 205–234, 2004.

[33] V. Feldman, “On the power of membership queries in agnostic learning,” *Journal of Machine Learning Research*, vol. 10, pp. 163–182, 2009.

[34] ——, “Hardness of approximate two-level logic minimization and PAC learning with membership queries,” *J. Comput. Syst. Sci.*, vol. 75, no. 1, pp. 13–26, 2009.

[35] R. D. Nowak, “The geometry of generalized binary search,” *IEEE Trans. Information Theory*, vol. 57, no. 12, pp. 7893–7906, 2011.

[36] L. Chen, S. H. Hassani, and A. Karbasi, “Dimension coupling: Optimal active learning of halfspaces via query synthesis,” *CoRR*, vol. abs/1603.03515, 2016.

[37] F. L. Wauthier, N. Jojic, and M. I. Jordan, “Active spectral clustering via iterative uncertainty reduction,” in *The 18th ACM SIGKDD*, 2012, pp. 1339–1347.

[38] H. Ashtiani, S. Kushagra, and S. Ben-David, “Clustering with same-cluster queries,” in *Annual Conference on Neural Information Processing Systems 2016*, 2016, pp. 3216–3224.

[39] S. Vikram and S. Dasgupta, “Interactive bayesian hierarchical clustering,” in *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, 2016, pp. 2081–2090.

[40] K. G. Jamieson and R. D. Nowak, “Active ranking using pairwise comparisons,” in *25th Annual Conference on Neural Information Processing Systems 2011.*, 2011, pp. 2240–2248.

[41] Y. Xu, H. Zhang, K. Miller, A. Singh, and A. Dubrawski, “Noise-tolerant interactive learning from pairwise comparisons with near-minimal label complexity,” *arXiv:1704.05820*, 2017.