List Learning with Attribute Noise

Mahdi Cheraghchi
University of Michigan–Ann Arbor
mahdich@umich.edu

Elena Grigorescu
Purdue University
elena-g@purdue.edu

Brendan Juba
Washington University in St. Louis
bjuba@wustl.edu

Karl Wimmer
Duquesne University
wimmerk@duq.edu

Ning Xie
Florida International University
nxie@cis.fiu.edu

Abstract

We introduce and study the model of list learning with attribute noise. Learning with attribute noise was introduced by Shackelford and Volper (COLT, 1988) as a variant of PAC learning, in which the algorithm has access to noisy examples and uncorrupted labels, and the goal is to recover an accurate hypothesis. Sloan (COLT, 1988) and Goldman and Sloan (Algorithmica, 1995) discovered information-theoretic limits to learning in this model, which have impeded further progress. In this article we extend the model to that of list learning, drawing inspiration from the list-decoding model in coding theory, and its recent variant studied in the context of learning. On the positive side, we show that sparse conjunctions can be efficiently list learned under some assumptions on the underlying ground-truth distribution. On the negative side, our results show that even in the list-learning model, efficient learning of parities and majorities is not possible, regardless of the representation used.

1 INTRODUCTION

We study the attribute-noise PAC learning model, introduced by Shackelford and Volper (1988), in which learning must be achieved despite the presence of errors that corrupt the *attributes* of the data (instead of the *labels* of the data that are more commonly used in the learning with error setting). The inherent difficulty in learning with attribute-noise has been formalized by Sloan (1988) and Goldman and Sloan (1995) by showing information-theoretic barriers: in the presence of attribute-noise, regardless of how much data is used, it is impossible to identify which representations are accurate. Historically, similar issues of identifiability were bypassed in coding theory by relaxing the notion of a solution to that of *list decoding* (Elias, 1957; Wozencraft, 1958); more recently, a similar notion of *list-learning* has been proposed to provide solutions in other learning settings where a correct solution simply cannot be identified from the given data (Balcan et al., 2008; Charikar et al., 2017; Diakonikolas et al., 2018; Karmalkar et al., 2019; Raghavendra and Yau, 2020). We further discuss this previous work in Section 2.2. Here we ask when and to what extent it is possible to overcome the non-identifiability barrier posed by attribute-noise by relaxing the solution to outputting a list of representations of Boolean functions.

In the attribute-noise model, the task is to learn a labeling function given labeled examples, where the examples may have corrupted entries. More formally, the algorithm has access to pairs $(\tilde{x}, c(x))$, where $x = (x_1, x_2, \ldots, x_n) \in X$ is drawn independently from an unknown underlying distribution $\mathcal{D}$ over X, $c \in \mathcal{C}$ is an unknown labeling function from a concept class $\mathcal{C}$ over domain X, and $\tilde{x}$ is obtained from x by applying a noise vector $\rho = (\rho_1, \rho_2, \ldots, \rho_n)$ from a noise distribution that affects the coordinates (a.k.a. attributes) of x. The goal is to output, with probability at least $1-\delta$, a hypothesis c' that is $(1-\epsilon)$-accurate with respect to c over $\mathcal{D}$, namely $\Pr_{x\in\mathcal{D}}[c(x) = c'(x)] > 1 - \epsilon$. Hence, while in the standard PAC-learning model of Valiant (1984) the algorithm has access to $\tilde{x} = x$ (namely, actual examples from the input distribution), in the

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

A preliminary full version of this work is available as arXiv:2006.06850 (Cheraghchi et al., 2020).

attribute-noise version, the algorithm only has access to a noisy version of x, making the task of learning the labeling function significantly more difficult.

The attribute-noise model captures a setting in which one seeks an accurate model of dependencies in the "ground truth" process captured by $\mathcal{D}$ and c, in spite of errors in the recording of the data. For example, this formulation is appropriate for the task of formulating models in data-driven science; a small list of candidate functions in such a setting then corresponds to a list of possible hypotheses for further investigation. It stands in contrast to the (much easier) *label noise* model, which captures the task of making accurate predictions from the observed data while the observed data is generated from an unknown concept which may not match c. Indeed, if one is only interested in forecasting or building a device that works directly with the noisy data $\tilde{x}$ produced by given real-world sensors, such a setting may be captured by a suitable label-noise model. We stress that since accuracy in the attribute-noise model is assessed with respect to $\mathcal{D}$, which is never observed directly, the attribute-noise model is *not* captured by the label noise model, and is indeed much more challenging than the label noise model. We also note that the attribute-noise model can capture the classical *group testing* problem, see Section 2.1.

All previous work studies concept classes over Boolean attributes $x_i \in \{0,1\}$ for all $i \in [n]$, and Boolean labeling functions $c : \{0,1\}^n \to \{0,1\}$. Specifically, Shackelford and Volper (1988) show that under *uniform random attribute-noise*, where the noise flips each coordinate independently with probability $p \in [0,1]$, it is possible to learn k-DNF expressions and conjunctions efficiently, if the noise rate p is known to the algorithm. However, the knowledge of p is not necessary for efficient learning, as proved by Goldman and Sloan (1995). They further consider *product random attribute-noise* on conjunctions, where coordinates are affected independently by noise of possibly different rates p_i, and prove that if these rates are unknown, and if $p_i > 2\epsilon$ in each coordinate, then it is information-theoretically impossible to recover any $(1-\epsilon)$-accurate hypothesis. Hence, regardless of the running time of the algorithm, and the number of samples received, the algorithm is unable to output a good answer. On the other hand, if the noise rates are known, Decatur and Gennaro (1995) provide efficient algorithms for PAC-learning conjunctions and k-DNF formulas. More recently, Bshouty et al. (2003) studied the problem of learning in which noise distributions are unconstrained or unknown, but the examples are drawn from the uniform distribution.

We emphasize that the attribute-noise model is not captured by noisy-PAC. Indeed, the celebrated results of Angluin and Laird (1987) show that learning in the noisy PAC model is information theoretically possible for any noise rate $\rho < 1/2$, and in fact k-CNF and k-DNFs can be learned efficiently in this high-noise regime. Again, this is in contrast with the attribute-noise setting where identifiability is not possible for unknown noise rate $\rho > 2\epsilon$ per coordinate (Goldman and Sloan, 1995). One can also view attribute-noise as an intermediate between noisy PAC and malicious noise, where the latter model assumes that $1-\rho$ fraction of the output is correct, and the remaining ρ fraction may be completely irrelevant. Kearns and Li (1993) show that under this model in order to identify an ϵ-accurate hypothesis one must have $\rho < \epsilon/(1+\epsilon)$.

Motivated by its applications in certain real-world machine learning scenarios, as well as its apparent difficulty, we revisit the learning with attribute-noise model and study it under product random attribute-noise, in which the noise rates are *not* known. We overcome the information-theoretic impossibility result of Sloan (1988); Goldman and Sloan (1995) by allowing the algorithm output a small *list* of labeling functions that contains one which is accurate. Thus, even if it is impossible to identify a single accurate function, we can hope to produce a small list of candidate hypotheses that contains an accurate one. Indeed, the proof of Sloan (1988); Goldman and Sloan (1995) follows from an explicit construction of two pairs $(\mathcal{D}_1, c_1, \mathcal{R}_1)$ and $(\mathcal{D}_2, c_2, \mathcal{R}_2)$ of distributions, distinct dictators as labeling functions, and product noise distributions, respectively. The two pairs of tuples lead to exactly the same observed distribution over the $n+1$ bits received $(\tilde{x}, c(x))$, when $\nu > 2\epsilon$, where ν is an upper bound on the noise amount per attribute. In the list-learning model the algorithm is allowed to output both solutions. In fact, as in PAC learning, any $(1-\epsilon)$-accurate hypothesis with respect to the input distribution $\mathcal{D}_i$ is a valid solution to the learning problem, hence it is enough to outputs a small *net* of hypotheses that covers all the valid inputs, in the sense that for any valid input that could have resulted in the observed distribution, the list contains a hypothesis that is $(1-\epsilon)$-accurate with respect to that input. Our results provide some sufficient conditions under which efficient list learning is still possible despite the previous barriers. We also show strong lower bounds for most natural classes of Boolean functions.

1.1 The Model: List Learning with Attribute Noise

We denote an instance of the attribute-noise learning problem by a tuple $(\mathcal{D}, c, \mathcal{R})$, where $\mathcal{D}$ is the unknown distribution from which the algorithm receives noisy

samples, c is the labeling function, and $\mathcal{R}$ is the noise distribution. We will denote by $\tilde{\mathcal{D}}$ the observed distribution of $(\tilde{x}, c(x))$, where $\tilde{x} = x + \rho$, $x \leftarrow \mathcal{D}$ and $\rho \leftarrow \mathcal{R}$. We will often abuse notation and denote the marginal distribution on $\tilde{x}$ by $\tilde{\mathcal{D}}$ as well.

For an observed distribution $\tilde{\mathcal{D}}$, a *net* $\mathcal{H}$ (specifically, an ϵ-*net*) is a set of hypotheses such that for any tuple $(\mathcal{D}, c, \mathcal{R})$ that could have resulted in the observed distribution $\tilde{\mathcal{D}}$, there exists at least one $h \in \mathcal{H}$ that is a $(1-\epsilon)$-accurate solution with respect to c and $\mathcal{D}$.

Inspired by the list-decoding model in coding theory, we seek answers to the following general questions:

1. (Combinatorial): Does there exist a small net $\mathcal{H}$ for the attribute-noise learning problem with observed distribution $\tilde{\mathcal{D}}$?
2. (Algorithmic): Can a net for the attribute-noise learning problem with observed distribution $\tilde{\mathcal{D}}$ be computed efficiently?

We formalize these notions below, in the attribute-noise PAC-learning model, with product random noise.

Definition 1.1. *(List learning with random product attribute-noise) Let $\mathcal{C}$ be a concept class containing Boolean functions $c : \{0,1\}^n \to \{0,1\}$, $\mathcal{D}$ a distribution over $\{0,1\}^n$, let $\nu, \epsilon \in (0,1)$, and $0 \le p_1, \ldots, p_n \le \nu$. Let $\mathcal{R}$ be noise distribution defined as the product of n independent Bernoulli distribution with parameters p_i, $i \in [n]$.*

1. *(Combinatorial) $\mathcal{C}$ is said to be* list-learnable *with list size $\ell = \ell(\nu, \epsilon)$ if there exists a net $\mathcal{H}$ for the solutions of the attribute-noise learning problem with input distribution $\mathcal{D}$, such that $|\mathcal{H}| \le \ell$.*
2. *(Algorithmic) $\mathcal{C}$ is said to be* algorithmically list learnable *if there exists a randomized algorithm outputting all $h \in \mathcal{H}$ with probability $1-\delta$ in time proportional to ℓ.*

1.2 Our Results

First, we show that the classes of parities and majorities are not amenable to efficient list learning, as every net for them has exponential size, regardless of the representation used for the net. More generally, we obtain our lower bound for any symmetric family of functions with sufficiently high *noise sensitivity*. (Recall that the noise sensitivity under ρ noise, $\mathbb{NS}_\rho(f)$, is the probability the value of f changes when its inputs are corrupted by product noise of rate ρ.)

Theorem 1.2. *(Theorem 3.3, informal) Let f be a symmetric function $f : \{0,1\}^{n/2} \to \{0,1\}$. Let $\mathcal{F}_f$ be the family of functions on n bits containing all functions f_S obtained by instantiating f on the set $S \subset [n]$ with $|S| = n/2$. Let $\rho > 0$. Suppose $\epsilon \le (\frac{1}{2} - o(1))\mathbb{NS}_{\rho/15}(f)$. Then if for every $f_S \in \mathcal{F}_f$ and distribution $\mathcal{D}$ on $\boldsymbol{x}$ there is an $h \in \mathcal{H}$ satisfying $\Pr_{\boldsymbol{x}\sim\mathcal{D}}[f_S(x) \ne h(x)] < \epsilon$, then $|\mathcal{H}| > 2^{\Omega(n)}$.*

Two immediate corollaries follow:

Corollary 1.3. *Taking $f(x_1, x_2, \ldots, x_{n/2}) = \sum_{i=1}^{n/2} x_i$, namely $f = \mathrm{PARITY}_{n/2}$, in Theorem 1.2, the lower bound holds for any $\rho > 0$ and $\epsilon < \frac{1}{4} - o(1)$.*

Corollary 1.4. *Taking $f(x_1, x_2, \ldots, x_{n/2}) = \mathrm{MAJORITY}(x_1, x_2, \ldots, x_{n/2})$, namely $f = \mathrm{MAJORITY}_{n/2}$, in Theorem 1.2, the lower bound holds for any $\rho > 0$ and $\epsilon < \Omega(\sqrt{\rho})$.*

We stress that since these lower bounds hold regardless of the representation used in the list, they give lower bounds for richer function classes that contain parities or majorities (respectively) as special cases, such as general linear threshold functions and so on. Of course, such a distinction between "proper" (representation-specific) and "improper" (representation-independent) solutions does not arise in coding theory, but is a common feature in learning theory. Improper learning is the main subject of interest in learning theory, but lower bounds against improper learning algorithms are usually much more challenging. The same holds here: it is generally much easier to argue that an exponential lower bound holds if the function is forced to be a parity function or a conjunction (see below), for example.

Our main results focus on conjunctions, for which we give a general lower bound, and an upper bound for a specific restriction on the input distribution on examples.

Theorem 1.5. *(Theorem A.2, informal) Let $k > 0$ be an integer, $\epsilon > 0$, and let $\mathcal{C}_k$ be the set of all conjunctions over k bits out of n bits $f : \{0,1\}^n \to \{0,1\}$. If the attribute-noise is $\rho = \frac{1}{k} > 8\epsilon$, then there is an input distribution $\mathcal{D}$ such that list learning $\mathcal{C}_k$ under $\mathcal{D}$ with accuracy ϵ would require a list of size $|\mathcal{H}| > 2^{\Omega(k)}$.*

Again, since this theorem is representation-independent, we obtain the same lower bound for any family of functions that can express the conjunctions on k out of n bits. Thus, even with $k = \Omega(n)$, we obtain lower bounds for decision trees, DNFs, s-CNFs, and so on. (By standard reductions, i.e., swapping 0 and 1, one can also obtain the same lower bound for s-DNFs.) Between Theorem 1.2 and the above, we have lower bounds for essentially all of the natural families of functions studied in learning theory, provided that the function depends

on $\omega(\log n)$ coordinates. (When $k = O(\log n)$, the problems are all open, see Section 4.)

Our main result is a sufficient assumption on the input distribution on examples that allows efficient list learning of sparse conjunctions under arbitrary probabilities of flipping individual attributes.

Theorem 1.6. *(Theorem B.1, informal) For any positive integer k, k', and any real number $0 < \epsilon, \delta < 1$, $0 < \gamma \leq 1/2$, there exists a randomized algorithm which, with probability at least $1 - \delta$, list learns k-conjunctions with accuracy $1 - \epsilon$, with sample complexity* $\mathrm{poly}(k, \frac{1}{\epsilon}, \frac{1}{\gamma}, \log \frac{1}{\delta})$ *and time complexity* $\mathrm{poly}(n, \frac{1}{\epsilon}, \log \frac{1}{\delta}, \frac{1}{\gamma}, (\frac{k}{\epsilon\gamma})^k)$ *in the attribute-noise model with bit noise rate $0 \leq \nu_i < \frac{1}{2} - \gamma$ for every $1 \leq i \leq n$, under the assumption that the ground-truth distribution is k'-wise independent.*

We note that the trivial PAC learning algorithm that tries all monotone conjunctions of size at most k and outputs the best candidate works only for noise rate $\nu \leq \frac{\epsilon}{2k}$; we include a proof for completeness in the Appendix[1] C.

Theorem 1.6 shows that, even under high noise rates, list learning conjunctions with attribute noise is *fixed-parameter tractable*; meaning that the algorithm runs in polynomial time in n (with a universal constant exponent) for any fixed value of the parameter k. The result only requires a mild restriction of pairwise independence on the ground-truth distribution. This a much milder assumption than being a product distribution (in turn, much milder than being uniform) which is commonplace in computational learning theory.

2 FURTHER RELATED WORK

We will briefly discuss the history and context of the models we use in this work, as well as a further application to the *group testing* problem.

2.1 Connection with Group Testing

As a further motivation for the study of conjunctions (or, equivalently, disjunctions) as a concept class, here we briefly discuss an application for the classical combinatorial group testing problem; cf. Du and Hwang (2000).

In combinatorial group testing, there is a population of n specimens among which up to k are defective, and the goal is to learn the subset S of the defectives via as few disjunctive tests as possible. Each disjunctive test picks a group of the items and returns positive if and only if there is a defective among the picked group. Define the Boolean function $f(x_1, \ldots, x_n) := \vee_{i \in S} x_i$, so the goal is now reduced to properly learning the disjunction f.

When a subset $T \subseteq [n]$ of the specimens are randomly pooled and the pooled sample is tested, the test result is positive if and only if at least one specimen in T is defective. Observe that in terms of f, the test outcome is $f(t_1, \ldots, t_n)$ where $t_i = 1$ iff $i \in T$. Thus, effectively, learning disjunctions captures group testing (the distribution of attributes corresponds to the pooling design, for which it is known that product distributions lead to an essentially optimal number of tests (Du and Hwang, 2000, Chapter 4)).

Now, attribute noise captures the realistic consideration that each sample participating in each pool may register incorrectly with some probability due to such effects as dilution (it is worth noting that our algorithm in Theorem 1.6 actually outputs a *unique* superset of S of size not much larger than k).

As another related example, suppose a small group of people (call them *agents*) have interacted in a social gathering with a population of size n, among which k have a contagious disease. The agents are later tested for the infection, and from the results it is desired to identify the k infected individuals using contact tracing data. This problem has been considered by Cheraghchi et al. (2011). Again, we have a group testing instance, where each agent defines a pool of the individuals, with whom they have interacted in the event. One can assume that each agent has interacted with a random i.i.d. set of individuals (product distribution), and in each interaction the infection (if present) will be caught by the agent with some fixed but unknown probability (attributed noise). Again, the problem reduces to learning disjunctions with attribute noise.

2.2 Further Discussion of Related Work

The information theoretic lower bounds of Sloan (1988); Goldman and Sloan (1995) are analogous to the classical scenario in coding theory, in which, upon receiving a word corrupted by a high amount of noise, decoding becomes ambiguous. (We remark briefly that several authors, including Rubin (1976), Schuurmans and Greiner (1994), and Michael (2010) furthermore have introduced models of learning from data with omitted attributes, which is analogous to decoding from erasures, and is much easier.) As a result, Elias (1957) and Wozencraft (1958) extended the classical notion of unique decoding to that of *list-decoding*, where the algorithm is required to output a list of all possible messages that could have resulted in the re-

[1] The appendices are available as supplementary material.

ceived one. A similar motivation prompted Balcan et al. (2008) to introduce the notion of *list-decodable learning* in the context of clustering, where their algorithm is required to output a small list that includes a "good" clustering, with high probability. Follow-up results by Charikar et al. (2017) use this framework in the context of learning from untrusted data when there is a minority fraction of "inliers" and so identifiability cannot hold. In the same vein, Diakonikolas et al. (2018) obtain algorithms for robust mean estimation, and learning mixtures of Gaussians. More recently, Karmalkar et al. (2019) and Raghavendra and Yau (2020) independently gave list-decodable linear regression algorithms for this minority-inlier setting. In all of these works, the difference is that there is guaranteed to be a fixed fraction of uncorrupted examples (whereas the corruption of the remaining examples is arbitrary). By contrast, in the attribute-noise model we study, with high probability *every* example has a non-negligible fraction of corrupted attributes, though conversely, the corruptions are stochastic and independent. Nevertheless, in spite of ours being a stochastic-noise model, we will see that the lack of clean examples still poses serious challenges, even for a list learner.

3 LOWER BOUNDS AND A CONSTRUCTIVE ALGORITHM

We now give a technical overview of our results.

3.1 The Lower Bounds, Theorems 1.2 and 1.5

The high-level idea of the lower-bound proofs is to explicitly construct a large set of labeling functions $c \in \mathcal{C}$ and initial input and noise distributions such that any function in the net can only be $(1-\epsilon)$-accurate for a small number of possible initial solutions $(\mathcal{D}, c, \mathcal{R})$, regardless of the representations used for the functions in the net. Hence, to cover an exponential number of such potential solutions a net has to have large size. The construction of the initial distributions exploits the idea that bits (x_{2i}, x_{2i+1}) that are ρ-correlated (meaning that x_{2i+1} takes the same value as x_{2i} w.p. $1-\rho$, and takes the flipped value with probability ρ) appear identical to an observer when adding Bernoulli random noise ρ to one copy and no noise to the other copy.

For the proof, define the **noise operator at ρ on S**, denoted by $N_{S,\rho}(x)$, to be a random string such that $N_{S,\rho}(x)_i$ is a uniform random bit ρ-correlated with x_i if $i \in S$, and $N_{S,\rho}(x)_i = x_i$ with probability 1 for $i \notin S$. We further define the **noise sensitivity at ρ on S** to be $\mathbb{NS}_{S,\rho}(f) = \Pr_{\boldsymbol{y}\sim\mathcal{U}_{n/2}}[f(\boldsymbol{y}) \neq f(N_{S,\rho}(\boldsymbol{y}))]$. These are related to the standard noise sensitivity constructions via $N_\rho(x) = N_{[n],\rho}(x)$, and $\mathbb{NS}_\rho(f) = \Pr_{\boldsymbol{y}\sim\mathcal{U}_{n/2}}[f(\boldsymbol{y}) \neq f(N_\rho(\boldsymbol{y}))]$:

Claim 3.1. *Let $S \subseteq [n]$ be a set such that $|S| = n/14$. For every symmetric Boolean function f on $n/2$ variables such that $\mathbb{NS}_{S,\rho}(f) = 2^{-o(n)}$ for all S, $\mathbb{NS}_{S,\rho}(f) \geq (1-o(1))\mathbb{NS}_{\rho/15}(f)$.*

We defer the proof of Claim 3.1 to Appendix A.

Before defining the functions in $\mathcal{F}$ for Theorem 1.2, we will make some notational conventions. For the sake of presentation we assume n is even.

For a string $x \in \{0,1\}^n$, we may view it as the concatenation of pairs (x_{2i+1}, x_{2i+2}), for $i = 0, 1, ..., n/2-1$, and define two strings $x^0, x^1 \in \{0,1\}^{n/2}$, by selecting the odd, respectively the even, indices of these pairs in order, namely $x^0 = x_1, x_3, \ldots, x_{n-1}$ and $x^1 = x_2, x_4, \ldots, x_n$. For $x \in \{0,1\}^n$ and a string $z \in \{0,1\}^{n/2}$, we define the hybrid string $x^z \in \{0,1\}^{n/2}$ to be the string that for each $0 \leq i \leq n/2-1$ selects either x_{2i+1} if $z_i = 0$, or x_{2i+2} if $z_i = 1$, denoted by $x^z = (x_1^{z_1}, x_2^{z_2}, \ldots, x_{n/2}^{z_{n/2}})$, where $x_i^{z_i} = x_{2i+1}$ if $z_i = 0$, and $x_i^{z_i} = x_{2i+2}$ if $z_i = 1$.

We now define the set of functions $\mathcal{F}$. For a symmetric function $f : \{0,1\}^{n/2} \to \{0,1\}$, such as *parity* or *majority*, and a string $z \in \{0,1\}^{n/2}$, let $f^z : \{0,1\}^n \to \{0,1\}$ be the function $f^z(x) = f(x^z) = f(x_1^{z_1}, x_2^{z_2}, \ldots, x_{n/2}^{z_{n/2}})$. Let $\mathcal{F} = \mathcal{F}(f) = \{f^z\}_{z\in\{0,1\}^{n/2}}$.

Further, for $z \in \{0,1\}^{n/2}$ let $\mathcal{D}^z$ be the distribution[2] on $\{0,1\}^n$ defined by the following probability experiment:

- The coordinates in $\boldsymbol{x}^z$ are drawn independently and uniformly at random. That is, $\boldsymbol{x}^z \sim \mathcal{U}_{n/2}$, where $\mathcal{U}_{n/2}$ represents the uniform distribution on $\{0,1\}^{n/2}$.
- The coordinates in $\boldsymbol{x}^{\overline{z}}$ are ρ-noisy copies of $\boldsymbol{x}^z$—each bit $x_i^{\overline{z_i}}$ is a ρ-noisy copy of $x_i^{z_i}$.

We will show that if z is unknown, and we see labeled examples according to f^z under $\mathcal{D}^z$ with ρ-bounded attribute-noise, then list-learning to small accuracy requires an exponential size list. That is, for every set of functions $\mathcal{H}$ (our proposed net), the quantity $\max_{z\in\{0,1\}^{n/2}} \min_{h\in\mathcal{H}} \Pr_{\boldsymbol{x}\sim\mathcal{D}^z}[f^z(x) \neq h(x)]$ is "large" if $|\mathcal{H}|$ is sub-exponential.

For f^z with respect to $\mathcal{D}^z$, given x, the attribute-noise $N_\rho^z(x)$ is as follows: we apply ρ-noise to each $x_i^{z_i}$, and no noise to $x_i^{\overline{z_i}}$. It follows that for *every* $\mathcal{D}^z$, the resulting distribution over the *labeled* examples is the same. We define $\mathcal{D}$ to be distribution[3] on $\{0,1\}^n$ such that, for each i, x_i^0 and x_i^1 are ρ-correlated uniformly

[2] Actually, $\mathcal{D}^z$ is the same distribution no matter what z is.

[3] Actually, this is the same as $\mathcal{D}^z$.

random bits, and the $n/2$ pairs (x_i^0, x_i^1) are chosen independently.

We now exploit the fact that totally symmetric functions with high "noise sensitivity" are often far apart.

Lemma 3.2. *Let $z, z' \in \{0,1\}^{n/2}$ be strings such that $|z - z'| \geq n/14$. Then $\Pr_{\boldsymbol{x}\sim\mathcal{D}^z}[f^z(\boldsymbol{x}) \neq f^{z'}(\boldsymbol{x})] \geq (1 - o(1))\mathbb{NS}_{\rho/15}(f)$.*

Proof. Define S to be the set of strings where z and z' differ.

$$\begin{aligned}
\Pr_{\boldsymbol{x}\sim\mathcal{D}^z}[f^z(\boldsymbol{x}) \neq f^{z'}(\boldsymbol{x})] &= \Pr_{\boldsymbol{x}\sim\mathcal{D}^z}[f(\boldsymbol{x}^z) \neq f(\boldsymbol{x}^{z'})] \\
&= \Pr_{\boldsymbol{x}\sim\mathcal{D}^z}[f(\boldsymbol{x}^z) \neq f(N_{S,\rho}(\boldsymbol{x}^z))] \\
&= \Pr_{\boldsymbol{y}\sim\mathcal{U}_{n/2}}[f(\boldsymbol{y}) \neq f(N_{S,\rho}(\boldsymbol{y}))] \\
&= \mathbb{NS}_{S,\rho}(f) \\
&\geq (1 - o(1))\mathbb{NS}_{\rho/15}(f),
\end{aligned}$$

where the last inequality follows by Claim 3.1. □

Thus, any single member of the net can only be accurate for at most one of these far pairs, and so we must have a large net. We thus obtain a more specific version of Theorem 1.2.

Theorem 3.3. *Let $f : \{0,1\}^{n/2} \to \{0,1\}$ be a symmetric function, and $\rho > 0$. If $\epsilon \leq (\frac{1}{2} - o(1))\mathbb{NS}_{\rho/15}(f)$ then, for family $\mathcal{F} = \{f^z\}_{z\in\{0,1\}^{n/2}}$ of Boolean functions on n bits where the oracle produces examples with attribute-noise rate ρ, we have that any net $\mathcal{H}$ satisfying $\max_{z\in\{0,1\}^{n/2}} \min_{h\in\mathcal{H}} \Pr_{\boldsymbol{x}\sim\mathcal{D}^z}[f^z(x) \neq h(x)] < \epsilon$ must have $|\mathcal{H}| > 2^{\Omega(n)}$.*

Proof. By the triangle inequality, no function in the net can approximate both f^z and $f^{z'}$ for two strings z, z' where $|z - z'| \geq n/14$ (with respect to $\mathcal{D} = \mathcal{D}^z = \mathcal{D}^{z'}$) to within $(\frac{1}{2} - o(1))\mathbb{NS}_{\rho/15}(f))$. Thus, any function in the net can cover at most $\binom{n}{n/14}$ such functions f^z with respect to $\mathcal{D}^z$. It follows that any net requires $2^{n/2}/\binom{n}{n/14} \geq 2^{n/14}$ functions (here we used that $\binom{n}{k} < (ne/k)^k$, with $k = n/14$). □

The proof of Theorem 1.5 uses similar ideas, and appears in Appendix A.2. In this case we use distributions $\mathcal{D}^z$ over $\{0,1\}^{2k}$ such that

- The coordinates in $\boldsymbol{x}^z$ are drawn independently at random with bias $1/k$. That is, $\boldsymbol{x}^z \sim \mu_{k,1/k}$, where $\mu_{n,p}$ denotes the p-biased distribution over $\{0,1\}^n$.
- The coordinates in $\boldsymbol{x}^{\overline{z}}$ are ρ-noisy copies of $\boldsymbol{x}^z$—each bit $\boldsymbol{x}_i^{\overline{z_i}}$ is a ρ-noisy copy of $\boldsymbol{x}_i^{z_i}$.

For f^z with respect to $\mathcal{D}^z$, given x, the attribute-noise $N_\rho^z(x)$ is as follows: we apply ρ-noise to each $x_i^{z_i}$, and no noise to $x_i^{\overline{z_i}}$. It follows again that for *every* $\mathcal{D}^z$, the resulting distribution over the *labeled* examples is the same.

We show a function in the net covers the most conjunctions by taking f to be 1 on $199k/100$ of these strings and 0 on the other $k/100$ since for every conjunction, a false 0 is roughly k times as costly as a false 1: To make the error less than $(1 - 1/k)^{k-1}(1/k)(1 - \rho)^{k-1}\rho \cdot (99k/100)$, there must be a function in the net that has no false 0's and at most $99k/100$ false 1's on these strings. (A function is covered if its bits correspond to those with ones.) There are $2^{99k/100}$ conjunctions covered, but 2^k conjunctions in total, so any net must have $2^{k/100}$ functions in it to achieve error below $(1 - 1/k)^{k-1}(1/k)(1 - \rho)^{k-1}\rho \cdot (99k/100)$. Taking $\rho = 1/k$, the error is at least $\rho/8$, so we need $\rho < 8\epsilon$ for a sub-exponential net.

3.2 The Conjunction Algorithm, Theorem 1.6

The essential difficulty in learning conjunctions under the attribute-noise model is that on the one hand, conjunctions are in general very sensitive to the attributes that appear in them; missing even one significant attribute incurs a large error. But, on the other hand, as illustrated in the lower bound, it is in general impossible to distinguish bits of the conjunction corrupted by noise in our examples from bits that would thus incur a serious error if they were included in the conjunction. Thus, we seek to find a small set of candidate coordinates and output all small subsets of these. Both the size of the set of candidates and the size of the conjunctions must be small to obtain a polynomial-size list. Proving that the algorithm does output a net for the solution space is the most difficult part of our arguments, the difficulty emerging from the fact that the accuracy of the solution is measured against the original unknown distribution rather than the observed distribution itself. The algorithm can only perform tests and optimize quantities using the corrupted examples, and we must then bound the distances from the unknown distribution.

We use the following notation: $\tilde{D}$: the observed distribution; D: the original distribution before applying the attribute-noise; $c = \wedge_{i\in c}\ell_i$: a conjunction[4] of size at most k, where $c \subset [n]$, $|c| \leq k$ and ℓ_i is either x_i or $1-x_i$; D_b (resp. $\tilde{D}_b$): the original (resp. observed) distribution conditioned on label c being b, for $b \in \{0,1\}$; ν_i: the attribute-noise rate of bit i.

[4]We abuse notation here to let c denote both the conjunction and the set of variables in the conjunction. Furthermore, the conjunction over the empty set is understood to be $\mathbf{1}$.

Algorithm 1: Learning-Conjunction $(\tilde{\mathrm{EX}}, k, \epsilon, \delta)$

input : Noisy example oracle $\tilde{\mathrm{EX}}(c, D)$, integer k, error parameter ϵ, and confidence parameter δ
output: A list of conjunctions

$m := 32k^2/(\epsilon^5\gamma^2)$
$M := O(k^4 \log n \log(1/\delta)/(\epsilon^9\gamma^4))$
$\mathcal{M} \leftarrow M$ random labeled examples drawn from the noisy example oracle $\tilde{\mathrm{EX}}(c, D)$
$S \leftarrow$ Pairwise-Independence-Test $(\mathcal{M}, \epsilon, \delta)$
for $i \leftarrow 1$ **to** n **do**
  Use $\mathcal{M}$ to estimate label sensitivity at the i^{th} bit $\widehat{\mathrm{LS}}_i$
  if $\widehat{\mathrm{LS}}_i < \epsilon\gamma/k$ **then**
    remove i from S
if $|S| < m$ **then**
  Output the list of conjunctions $\mathbf{0} \cup \{\wedge_{i \in c'} x_i\}_{c' \in \binom{S}{\leq k}}$
else
  Output $\mathbf{0}$

Algorithm 2: Pairwise-Independence-Test $(\mathcal{M}, \epsilon, \delta)$

input : M random labeled examples $\mathcal{M}$, error parameter ϵ, and confidence parameter δ
output: A subset $S \subset [n]$ of nearly pairwise independent bits under $\tilde{D}_1$

$S \leftarrow [n]$
for $i \leftarrow 1$ **to** n **do**
  Use positive examples in $\mathcal{M}$ to empirically estimate $\widehat{\mathrm{E}_{\tilde{D}_1}[x_i]}$
for $i \leftarrow 1$ **to** $n-1$ **do**
  for $j \leftarrow i+1$ **to** n **do**
    if $i \notin S$ *or* $j \notin S$ **then**
      continue
    if $\widehat{\mathrm{E}_{\tilde{D}_1}[x_i]} \leq 1/(8\epsilon m)$ *or* $\widehat{\mathrm{E}_{\tilde{D}_1}[x_j]} \leq 1/(8\epsilon m)$ **then**
      continue
    Use sampled examples to empirically estimate $\widehat{\mathrm{E}_{\tilde{D}_1}[x_i \cdot x_j]}$
    if $\left|\widehat{\mathrm{E}_{\tilde{D}_1}[x_i]} \cdot \widehat{\mathrm{E}_{\tilde{D}_1}[x_j]} - \widehat{\mathrm{E}_{\tilde{D}_1}[x_i \cdot x_j]}\right| > 1/(8\epsilon m)$ **then**
      Remove both i and j from S
Output S

We call a bit $i \in [n]$ a *conjunction bit* if $i \in c$ and *non-conjunction bit* otherwise. Note that without loss of generality, we may assume that every candidate conjunction bit in S is biased towards 1, i.e. $\mathrm{E}_{\tilde{D}}[x_i] \geq 1/2$ for every $i \in S$, as otherwise we simply replace x_i with $1 - x_i$ in our arguments.

The algorithm for list learning conjunctions under random product attribute-noise operates under the assumption that the attributes in the initial distribution on examples are pairwise independent.

Definition 3.4 (Non-uniform k-wise independence)**.** *Let $P : \{0,1\}^n \to \mathbb{R}^{\geq 0}$ be a distribution and k be a positive integer. P is said to be* (non-uniform) k-wise independent *if for any subset of k indices $\{i_1, \ldots, i_k\} \subset [n]$ and for any $z_1 \ldots z_k \in \{0,1\}^k$,*

$$\Pr_P[X_{i_1} \cdots X_{i_k} = z_1 \cdots z_k] = \Pr_P[X_{i_1} = z_1] \times \cdots \times \Pr_P[X_{i_k} = z_k].$$

Claim 3.5. *For any positive integer k and any distribution $D : \{0,1\}^n \to \mathbb{R}^{\geq 0}$, D is k-wise independent if and only $\tilde{D}$ is k-wise independent. In other words, attribute-noise does not change the k-wise independence of the underlying distribution.*

We defer the proof of this claim to Appendix D.

We first observe that since the bits of the actual conjunction must all take value 1 on label 1, and the noise is a product distribution, the bits of the actual conjunction in the noisy examples are fully independent when conditioned on label 1.

Learning conjunctions is easy when there is no attribute-noise because, if x_i is in the conjunction, then conditioned on label being 1, $\Pr[X_i = 1] = 1$ and this probability should be lower without the conditioning — unless variable x_i is almost surely being 1 under the distribution D. In other words, the expectation of a (relevant) conjunction bit should be *sensitive* to label change. This is also true under attribute-noise, although with lower sensitivity in general.

Definition 3.6. *The (observed)* label sensitivity *at bit i is defined by $\mathrm{LS}_i = \mathrm{E}_{\tilde{D}_1}[X_i] - \mathrm{E}_{\tilde{D}_0}[X_i]$; that is, LS_i is the difference between expectation of x_i conditioned on label being 1 and the expectation of x_i conditioned on label being 0.*

The algorithm thus first identifies the subset of variables that are (at least) pairwise independent on label 1, and then eliminates from this surviving set the variables that are not too sensitive to the label. These eliminated variables could not have been significant bits of the conjunction: if there is no attribute-noise, the variables in the conjunction would be very sensitive to the label, since they would always take value 1 on label 1, and they would take value 0 on label 0 sig-

nificantly often. Indeed, we note the following simple fact: since attribute-noise does not change the labels of examples, the total mass of positive or negative examples are the same for D and $\tilde{D}$.

Fact 3.7. *For any underlying distribution D of the example oracle and any attribute-noise vector ν, $\Pr_D[c(x) = 1] = \Pr_{\tilde{D}}[c(x) = 1]$ and $\Pr_D[c(x) = 0] = \Pr_{\tilde{D}}[c(x) = 0]$.*

Now, either the function is nearly constant and so a constant function predicts the label sufficiently well, or else there is a bounded statistical distance between the distribution conditioned on label 1 and the original distribution, which is a mixture of the label 1 and label 0 distributions. We show that when the function is far from constant, there cannot be too many coordinates surviving. Intuitively, otherwise, the weight would allow us to distinguish the label 1 distribution from the original distribution beyond the statistical distance, due to Chebyshev's inequality: the total weight would concentrate if there were many coordinates left. Thus we can afford to enumerate all small subsets of the surviving coordinates in this case.

In summary, our list-learning algorithm is described in Algorithm 1, in which we call Pairwise-Independence-Test (Algorithm 2, see Appendix B) as a subroutine to select the pairwise independent variables under distribution $\tilde{D}_1$. Essentially, for any pair of variables that each take both values with probability at least $1/(8\epsilon m)$, if the distance from independence is at least $1/(8\epsilon m)$, we filter out both members of the pair. Our analysis of Algorithm 1, sketched above, establishes Theorem 1.6; see Appendix B for the full details.

4 OPEN PROBLEMS

In our work we seek to develop the natural, yet difficult-to-analyze model of learning under attribute noise. While we prove several impossibility results and a sufficient condition for learning sparse conjunctions, our work leaves open a plethora of intriguing possibilities. We describe below a few important ones.

The first, most natural question is whether or not the pairwise-independence assumption is really needed for our algorithm:

Open Question 4.1. *Is the set of sparse conjunctions list-learnable under arbitrary product distributions of the attribute-noise?*

But, moreover, we note that our lower bounds do not rule out the possibility of obtaining polynomial-size lists for $O(\log n)$-sparse functions in general. So it is still open whether or not natural function families with small numbers of relevant coordinates have efficient list-learning algorithms, e.g.:

Open Question 4.2. *Is the set of sparse Boolean threshold functions list-learnable under arbitrary product distributions of the attribute-noise?*

Thus, in contrast to the usual theory of supervised learning, we do not have a characterization of which families of functions are (information-theoretically) learnable in terms of some parameter like the VC-dimension or Rademacher complexity in the attribute-noise list-learning setting:

Open Question 4.3. *What are necessary and sufficient conditions for families of Boolean functions to be list-learnable under the product distribution of the attribute-noise?*

Or, more generally:

Open Question 4.4. *What families of Boolean functions are list-learnable under general (not-necessarily independent product) noise distributions?*

Of course, one can ask both computational/algorithmic and statistical/combinatorial variants of these questions. But again, a central difficulty here is that the usual statistical techniques for estimating losses from data cannot be used directly to estimate losses from our corrupted data. Thus it seems that new tools may need to be developed to address these questions.

Acknowledgements

Mahdi Cheraghchi's research was partially supported by the National Science Foundation under Grant No. CCF-2006455. Elena Grigorescu was supported in part by NSF awards CCF-1910659 and NSF CCF-1910411. Brendan Juba was supported by NSF awards CCF-1718380, IIS-1908287, and IIS-1939677, and was hosted by the Simons Institute for Theory of Computing during portions of this work. Ning Xie's research was partially supported by grant ARO W911NF1910362.

References

D. Angluin and P. D. Laird. Learning from noisy examples. *Mach. Learn.*, 2(4):343–370, 1987.

M. Balcan, A. Blum, and S. Vempala. A discriminative framework for clustering via similarity functions. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 671–680, 2008.

N. H. Bshouty, J. C. Jackson, and C. Tamon. Uniform-distribution attribute noise learnability. *Inf. Comput.*, 187(2):277–290, 2003.

M. Charikar, J. Steinhardt, and G. Valiant. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 47–60, 2017.

M. Cheraghchi, A. Hormati, A. Karbasi, and M. Vetterli. Group testing with probabilistic tests: Theory, design and application. *IEEE Transactions on Information Theory*, 57(10):7057–7067, 2011.

M. Cheraghchi, E. Grigorescu, B. Juba, K. Wimmer, and N. Xie. List learning with attribute noise. *arXiv preprint arXiv:2006.06850*, 2020.

S. E. Decatur and R. Gennaro. On learning from noisy and incomplete examples. In *Proceedings of the Eigth Annual Conference on Computational Learning Theory, COLT 1995, Santa Cruz, California, USA, July 5-8, 1995*, pages 353–360, 1995.

I. Diakonikolas, D. M. Kane, and A. Stewart. List-decodable robust mean estimation and learning mixtures of spherical gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 1047–1060, 2018.

D.-Z. Du and F. Hwang. *Combinatorial Group Testing and its Applications*. World Scientific, second edition, 2000.

P. Elias. List decoding for noisy channels. *Technical Report 335, Research Laboratory of Electronics, MIT*, 1957.

S. A. Goldman and R. H. Sloan. Can PAC learning algorithms tolerate random attribute noise? *Algorithmica*, 14(1):70–84, 1995.

S. Karmalkar, P. Kothari, and A. Klivans. List-decodable linear regression. In *Advances in Neural Information Processing Systems 32*, pages 7423–7432, 2019.

M. J. Kearns and M. Li. Learning in the presence of malicious errors. *SIAM J. Comput.*, 22(4):807–837, 1993.

L. Michael. Partial observability and learnability. *Artificial Intelligence*, 174(11):639–669, 2010.

R. O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, USA, 2014. ISBN 1107038324.

P. Raghavendra and M. Yau. List decodable learning via sum of squares. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, pages 161–180, 2020.

D. B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

D. Schuurmans and R. Greiner. Learning default concepts. In *Proceedings of the Tenth Canadian Conference on Artificial Intelligence (AI'94)*, pages 99–106, 1994.

G. Shackelford and D. Volper. Learning k-DNF with noise in the attributes. In *Proceedings of the First Annual Workshop on Computational Learning Theory, COLT '88, Cambridge, MA, USA, August 3-5, 1988.*, pages 97–103, 1988.

R. H. Sloan. Types of noise in data for concept learning. In *Proceedings of the First Annual Workshop on Computational Learning Theory, COLT '88, Cambridge, MA, USA, August 3-5, 1988.*, pages 91–96, 1988.

L. G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984.

J. M. Wozencraft. List Decoding. *Quarterly Progress Report, Research Laboratory of Electronics, MIT*, 48:90–95, 1958.