On Noise-Tolerant Learning of Sparse Parities and Related Problems

Elena Grigorescu[*], Lev Reyzin[**], and Santosh Vempala[***]

School of Computer Science
Georgia Institute of Technology
266 Ferst Drive, Atlanta GA 30332
{elena,lreyzin,vempala}@cc.gatech.edu

Abstract. We consider the problem of learning sparse parities in the presence of noise. For learning parities on r out of n variables, we give an algorithm that runs in time poly $\left(\log \frac{1}{\delta}, \frac{1}{1-2\eta}\right) n^{\left(1+(2\eta)^2+o(1)\right)r/2}$ and uses only $\frac{r \log(n/\delta)\omega(1)}{(1-2\eta)^2}$ samples in the random noise setting under the uniform distribution, where η is the noise rate and δ is the confidence parameter. From previously known results this algorithm also works for adversarial noise and generalizes to arbitrary distributions. Even though efficient algorithms for learning sparse parities in the presence of noise would have major implications to learning other hypothesis classes, our work is the first to give a bound better than the brute-force $O(n^r)$. As a consequence, we obtain the first nontrivial bound for learning r-juntas in the presence of noise, and also a small improvement in the complexity of learning DNF, under the uniform distribution.

1 Introduction

Designing efficient, noise-tolerant, learning algorithms is a fundamental and important problem in part because real-world data is often corrupted, and algorithms unable to handle errors in their training data are not practically deployable. Angluin and Laird [3] formalize this notion in the "noisy PAC" setting, where the learner is required to achieve an error of ϵ with probability $1-\delta$, where the labels of the training examples are flipped at random with probability equal to the noise rate η. The statistical query (SQ) model [15] tries to capture the properties of noise-tolerant algorithms; algorithms implementable in the SQ model also work in the noisy PAC model of Angluin and Laird. Kearns's characterization has made it apparent that most algorithms that work in the noise-free setting can be adapted to work in the presence of noise.

[*] This material is based upon work supported by the National Science Foundation under Grant #1019343 to the Computing Research Association for the CIFellows Project.

[**] Supported by the Simons Postdoctoral Fellowship.

[***] Supported by National Science Foundation awards AF-0915903 and AF-0910584.

J. Kivinen et al. (Eds.): ALT 2011, LNAI 6925, pp. 413–424, 2011.

The **learning parity with noise (LPN)** problem, however, is one notable exception – techniques for learning parities in the noise-free setting cannot be extended to the noisy PAC model. In the LPN problem, the target is a parity on an unknown subset of variables, and the learning task is to discover this subset. More formally, the algorithm receives random examples $x \in \{0,1\}^n$ labeled by $\ell \in \{0,1\}$ obtained as $\ell = c^* \cdot x + e$, where $c^* \in \{0,1\}^n$ is the hidden target parity and $e \in \{0,1\}$ is a random bit set to 1 with probability η.

While the parity problem can be solved efficiently in the noise-free PAC setting using Gaussian elimination, the search of an efficient noise-tolerant algorithm has run into serious barriers. Most notably, the parity problem is unconditionally known not to be learnable in the statistical query model [5, 15], where a brute-force search is nearly optimal. The state-of-the-art algorithm, due to Blum et al. [6], runs in time $2^{O(n/\log n)}$, and it has not been improved since its publication over a decade ago. Their approach, a clever adaptation of Gaussian elimination, seems to reach its limits when trying to push these bounds further, which implies that either fundamentally new algorithms or new hardness results are needed for this problem.

The LPN problem is not only notoriously difficult, but also ubiquitous. It is closely related to the famous problem of decoding random linear codes in coding theory, and cryptosystems use the hardness of LPN and of the Learning With Errors problem (its generalization over larger rings) as a security assumption [14, 22, 23]. A variant of LPN is a candidate problem for automated tests to tell humans apart from computers [11], and as we discuss in the next section a large class of function families reduce to the LPN problem, showing its central importance to learning theory.

1.1 Sparse Parity

In this paper we focus on a variant of the LPN problem, where the target parity c^* has the additional property that it is sparse, namely only r of the n bits of the target c^* are set to 1. The sparse parity problem plays an important role in learning, specially due to its connections to r-juntas (i.e. functions depending on only r of the n variables), and DNF recently exhibited by Feldman et al. [9]. Namely, they show that an algorithm for learning noisy r-sparse parities running in time polynomial in n implies a polynomial time algorithm for learning r-juntas (Theorem 3). Similarly, learning s-term DNF can be reduced to learning noisy $(\log s)$-parities. Furthermore, they also show that learning r-parities corrupted by an η fraction of random noise is at least as hard as the intuitively more difficult task of learning an r-parity corrupted by adversarial noise of rate η. Until this work, however, no algorithm better than the brute force $O(n^r)$ has been exhibited for the sparse parity problem.

Our paper is the first to improve on this brute-force bound and gives a poly $\left(\log \frac{1}{\delta}, \frac{1}{1-2\eta}\right) \cdot n^{\left(1+(2\eta)^2+o(1)\right)r/2}$ algorithm for the uniform distribution (Corollary 1) and holds even for adversarial noise. This bound also generalizes to learning r-parities with random classification noise for arbitrary distributions

(Theorem 5). Moreover, our sample complexity of $\frac{r\log(n/\delta)\omega(1)}{(1-2\eta)^2}$ almost meets the information theoretic lower bound. These bounds are comparable to the results obtained by Klivans and Servedio [16] in the "attribute efficient" noiseless model, which have been subsequently improved by Buhrman et al. [7]. Interestingly, the same running time of $\tilde{O}(n^{r/2})$ is obtained in [7] for learning parities in the related "mistake-bound" model, but this is again in the noiseless case.

1.2 Implications to Related Problems

Given the state of the art for r-parities, clearly no algorithm better than $O(n^r)$ has been found for r-juntas. However, due to connections between parities and juntas discovered by Feldman et al. [9], our results have immediate implications for learning r-juntas. Namely, we obtain an algorithm for learning noisy r-juntas that runs in time poly $\left(\log\frac{1}{\delta}, \frac{1}{1-2\eta}, 2^r\right) n^{\left(1-2^{1-r}(1-2\eta)+2^{1-2r}(1-2\eta)^2+o(1)\right)r}$ (Corollary 2). For the uniform distribution without noise, Mossel et al. [19] show that juntas can be learned in time $O(n^{\frac{\omega r}{\omega+1}})$, where ω is the matrix multiplication exponent. Our bound, however, is the first nontrivial result for noisy juntas.

Our sparse parity algorithm also has implications to learning DNF under the uniform distribution, again using the reduction of Feldman et al. [9] from DNF to noisy parities. While there has been significant recent progress on the problem, including that random DNF are learnable under the uniform distribution [12, 24, 25], virtually nothing is known about their learnability in the worst case, even in the classical noiseless PAC model. In fact, there are serious impediments to learning DNF, including hardness results for their proper learnability [1]. The previous best algorithm for learning s-term DNF from random examples is due to Verbeurgt [27] and runs in quasipolynomial time $O(n^{\log\frac{s}{\epsilon}})$, where ϵ is the error rate. Our algorithm leads to an improved bound of poly $\left(\log\frac{1}{\delta}, \frac{1}{\epsilon}, s\right) n^{\left(1-\tilde{O}(\epsilon/s)+o(1)\right)\log\frac{s}{\epsilon}}$ for this problem in the uniform setting (Corollary 4).

1.3 Our Approach

Our algorithm works roughly as follows. We consider all $\frac{r}{2}$-parities and evaluate them on m examples. For each parity we can view these evaluations as coordinates for points on the m-dimensional Hamming cube, creating the set $\mathcal{H}_1$. Similarly, we consider all these high dimensional points XOR-ed coordinate-wise with the respective labels, creating the set $\mathcal{H}_2$. This gives us a set $\mathcal{H}$ of $2\binom{n}{r/2}$ points. Notice that in the noiseless case, two $\frac{r}{2}$-parities comprising the target c^* have the property that the corresponding point in $\mathcal{H}_1$ of one of them is at Hamming distance 0 from the corresponding point in $\mathcal{H}_2$ of the other. Moreover, two points not comprising the target are far apart – this is a key point employed in our exact learning algorithm for the uniform distribution. In the presence of noise these distances are perturbed, and to actually find two nearby points corresponding to the good parities we rely upon the recent approximate nearest neighbor / closest pair algorithm of Andoni and Indyk [2]. A careful analysis of this approach when generalizing this observation to arbitrary distributions yields a poly $\left(\log\frac{1}{\delta}, \frac{1}{\epsilon}, \frac{1}{1-2\eta}\right) n^{(1+(\frac{\eta}{\epsilon+\eta-2\epsilon\eta})^2+o(1))r/2}$ running time for proper learning (Theorem 5). Furthermore, we can use the above approach to design a learner that gives improved running time of poly $\left(n, \log\frac{1}{\delta}, \frac{1}{\epsilon-\eta}, \frac{1}{1-2\eta}\right) n^{\left(1+(2\eta)^2+o(1)\right)r/2}$ (Theorem 6) for arbitrary distributions, but it only works for error rates up to the noise rate.

Our approach is inspired by Hopper and Blum [11], who, in giving an overview of candidate hard problems to use for secure human identification, informally suggested a similar idea for an $O(n^{r/2})$ algorithm for a closely related problem. Their suggestion indeed works for the noiseless setting, but runs into unforeseen difficulty when noise is introduced. We remark that Buhrman et al. [7] also invoke the Hopper and Blum [11] approach for their results on learning sparse parities in the mistake bound model, and they also note that the idea of using $\frac{r}{2}$-parities appears in results of Klivans and Servedio [16] (who in turn cite a personal communication with Spielman). All these references, however, only treat the noise-free model under the uniform distribution.

Following up on this idea, we are able to handle large noise rates and arbitrary distributions.

1.4 Past Work

In their seminal work on the SQ model for noise-tolerant learning, Kearns [15] and Blum et al. [5] prove unconditional lower bounds for both the sparse and unrestricted versions of LPN of $\Omega(n^{r/c})$ and $\Omega(2^{n/c})$ (for constants $c > 1$), respectively. In the SQ model, the learner can ask the oracle statistical properties of the target, and this proves insufficient to efficiently learn parities. Then, in a breakthrough result, Blum et al. [6] show that in the noisy PAC model (see Section 2 for definition), one can circumvent the SQ lower bound of $2^{n/c}$ and give a Gaussian elimination type algorithm that runs in time $2^{O(n/\log n)}$. By considering parities on the first $\log n \log\log n$ bits of an example, they separate the classes SQ and noisy PAC. For the sparse r-parity problem, no similar breakthrough has occurred, and no algorithm better than the brute force $O(n^r)$ is known (before this work) for the noisy PAC model. On the other hand, if membership queries are allowed, the parity problem is solvable in polynomial time [10, 17].

Another interesting direction in the literature is that of establishing nontrivial tradeoffs between the sample and time complexity of the LPN problem, both in the noisy and noise-free versions. In the *noiseless* model for r-parities, one can trivially obtain an algorithm with $O(r \log n)$ sample complexity that runs in time $O(n^r)$, and this is improved to $O(n^{r/2})$ by Klivans and Servedio [16]. Additionally, Buhrman et al. [7] give a $o(n)$ sample complexity for a running time of $2^{O(r)+\log n}$, and Klivans and Servedio [16] and Buhrman et al. [7] also show polynomial time algorithms with a sample complexity of $\tilde{O}(n^{1-\frac{1}{r}})$.

In the *noisy* PAC model, the Blum et al. [6] result (for parities of unrestricted size) requires as many as $2^{O(n/\log n)}$ samples, but it works for any noise rate $\eta <$

$\frac{1}{2} - \exp(-n^\delta)$. This sample complexity has been improved by Lyubashevsky [18] to $O(n^{1+\epsilon})$ with higher running time and noise rate, and it remains open whether this can be further improved to a sample complexity of $O(n)$.

2 Notation and Preliminaries

In this paper, we are concerned with the model of **PAC learning under random classification noise** [3]. Let $c^* \in C : X \to \{0,1\}$ be the target concept, and D a distribution over X. In this model the learner has access to the oracle $\mathrm{EX}_\eta(c^*, D)$, which chooses $x \sim D$ and returns $(x, \ell(x))$, which is $(x, c^*(x))$ with probability $1 - \eta$ and $(x, 1 - c^*(x))$ with probability η.

Definition 1. *Algorithm L is said to PAC learn class $C : X \to \{0,1\}$ by class H in the presence of noise if, $\forall$ $c^* \in C$, distribution D over X, error rate $0 < \epsilon < \frac{1}{2}$, failure rate $0 < \delta < \frac{1}{2}$, noise rate $0 \leq \eta < \frac{1}{2}$, if L is given inputs ϵ, δ, η and access to $EX_\eta(c^*, D)$, then it will output hypothesis $h \in H$ s.t. with probability $1 - \delta$*

$$\boldsymbol{Pr}_{x \sim D}[h(x) \neq c^*(x)] < \epsilon.$$

If $H = C$ then we say the algorithm learns **properly**, otherwise it learns **improperly**. If the algorithm can recover c^*, we say it learns **exactly**. When we consider learning boolean functions under the **uniform distribution**, we restrict our attention to the distribution D over $X = \{0,1\}^n$ that assigns a probability of $\frac{1}{2^n}$ to each length n vector. When we consider a **noiseless** setting, we mean $\eta = 0$; this is the classical model of PAC learning [26]. Finally, if we relax the learning requirement to ask the algorithm to achieve only an error $< 1/2 - \gamma$, then we say the algorithm is a γ-**weak learner** for C.

Now we define the problem of learning r-parities. We note that operations on parities are performed modulo 2.

Definition 2. *In the (sparse) r-**parity problem**: the example domain is $X = \{0,1\}^n$, the target class C consists of vectors c in $\{0,1\}^n$ s.t. $||c||_1 = r$, and the target $c^* \in C$ labels examples $x \in X$ by $c^*(x) = \sum_i^n x_i c_i^* \in \{0,1\}$.*

Next we state some results from the literature that will be useful in our proofs. The first is an approximate closest pair algorithm that our algorithm in Section 3 relies upon.

Theorem 1 (Corollary of Andoni and Indyk [2]). *Given N points on the d-dimensional Hamming cube, finding a pair of points whose distance is within a factor $\rho > 1$ from the distance of the closest pair[1] can be done in time*

$$O(dN^{1+1/\rho^2 + O(\log\log N/\log^{1/3} N)}) = O(dN^{1+1/\rho^2+o(1)}).$$

[1] This is effectively done by running an approximate nearest neighbor algorithm on each point in the data structure.

We use the next theorem in Section 4 for an improved algorithm for improper learning of parities for arbitrary distributions.

Theorem 2 (Kalai and Servedio [13]). *For any* $0 < \eta < \epsilon < \frac{1}{2}$, $0 < \delta < \frac{1}{2}$, *there exists a boosting algorithm which, given access to a noise tolerant* γ*-weak learner and an example oracle* $EX_\eta(D, c^*)$*, runs in time poly*$(\log \frac{1}{\delta}, \frac{1}{\epsilon-\eta}, \frac{1}{\gamma}, \frac{1}{1-2\eta})$*, and with probability* δ *outputs a hypothesis* h *such that* $\Pr_{x\sim D}[h(x) \neq c^*(x)] < \epsilon$.

The following theorems are used in Section 5 to give improved bounds for learning juntas and DNF.

Theorem 3 (Feldman et al. [9]). *Let* A *be an algorithm that learns parities of* r *variables on* $\{0,1\}^n$*, under the uniform distribution, for noise rate* $\eta' \leq \frac{1}{2}$ *in time* $T(n, r, \eta')$*. Then there exists an algorithm that exactly learns* r*-juntas under the uniform distribution with noise rate* η *in time*

$$O(r2^{2r}T(n, r, 1/2 - 2^{-r}(1 - 2\eta))).$$

Theorem 4 (Feldman et al. [9]). *Let* A *be an algorithm that learns parities of* r *variables on* $\{0,1\}^n$*, under the uniform distribution, for noise rate* $\eta \leq \frac{1}{2}$ *in time* $T(n, r, \eta)$ *and sample complexity* $S(n, r, \eta)$*. Then there exists an algorithm that PAC learns* s*-term DNF under the uniform distribution in time*

$$\tilde{O}\left(\frac{s^4}{\epsilon^2}T\left(n, \log(\tilde{O}(s/\epsilon)), 1/2 - \tilde{O}(\epsilon/s)\right) \cdot S^2\left(n, \log(\tilde{O}(s/\epsilon)), 1/2 - \tilde{O}(\epsilon/s)\right)\right).$$

3 Learning Sparse Parities

We begin by presenting our algorithm for r-parities and afterwards prove its correctness and running time. As discussed in Section 1.3, our algorithm tries to find two "nearby" $\frac{r}{2}$-parities that compose to form the correct parity. We do this by evaluating all the parities on a sufficiently large set of examples and finding an approximate closest pair according to the evaluations and the evaluations XORed with the labels. The exact procedure appears in Algorithm 1.

Algorithm 1. Learn r-Parities $(r, n, \epsilon, \delta, \eta)$

```
Obtain a set $\hat{X} = \{x_1, \ldots, x_m\}$ of examples drawn from the oracle $\mathrm{EX}_\eta(c^*, D)$, where $m = \frac{r \log(n/\delta)\omega(1)}{(\epsilon' - \eta)^2}$ and $\epsilon' = \epsilon + \eta - 2\epsilon\eta$.
For each $\frac{r}{2}$-parity $c$, evaluate it on $\hat{X}$ to obtain the corresponding $\langle c\cdot x_1, c\cdot x_2, \ldots, c\cdot x_m\rangle \in \{0,1\}^m$ and $\langle c \cdot x_1 + \ell(x_1), c \cdot x_2 + \ell(x_2), \ldots, c \cdot x_m + \ell(x_m)\rangle \in \{0,1\}^m$. Let $\mathcal{H}$ be the set of all these $2 \cdot \binom{n}{r/2}$ points on the Hamming cube.
Run the Approximate Closest Pair algorithm from Theorem 1 on $\mathcal{H}$ with the approximation parameter $\rho = \epsilon'/\eta$, to obtain the closest pair of points in $\{0,1\}^m$ with corresponding $\frac{r}{2}$-parities $c_1$ and $c_2$, respectively.
Return $c_1 + c_2$.
```

Now we state and prove our main theorem.

Theorem 5. *For any $0 < \epsilon < \frac{1}{2}$, $0 < \delta < \frac{1}{2}$, $0 \le \eta < \frac{1}{2}$, and distribution D over $\{0,1\}^n$, the class of r-parities can be properly PAC learned with random classification noise using $\frac{r \log(n/\delta)\omega(1)}{\epsilon^2(1-2\eta)^2}$ samples in time*

$$\frac{\log(1/\delta) n^{\left(1+\left(\frac{\eta}{\epsilon+\eta-2\epsilon\eta}\right)^2+o(1)\right)r/2}}{\epsilon^2(1-2\eta)^2}.$$

Proof. For convenience, in this proof, we define the quantity ϵ' to be the error we need to achieve on the *noisy* examples, drawn from $\mathrm{EX}_\eta(c^*, D)$. There is a simple relationship between ϵ' and the quantities ϵ and η:

$$\begin{aligned} \epsilon' &= 1 - ((1-\epsilon)(1-\eta) + \epsilon\eta) \\ &= \epsilon + \eta - 2\epsilon\eta. \end{aligned}$$

Note that $\epsilon' > \eta$. To analyze Algorithm 1, we first define the *empirical agreement* of a parity c on a sample $\hat{X} = \{x_1, x_2, \dots, x_m\}$ as

$$\hat{\mathrm{agr}}_{\hat{X}}(c) = \sum_{x \in \hat{X}} c \cdot x \ .$$

We define the set $\mathcal{B}$ of bad parities c' as those whose error according to the examples chosen from the noisy oracle is $\ge \epsilon'$, as in $c' \in \mathcal{B}$ iff

$$\mathbf{Pr}_{x \sim \mathrm{EX}_\eta(c^*,D)}[c'(x) \neq \ell(x)] \ge \epsilon'.$$

If we are able to find a parity not in the bad set, we will succeed in learning.

The empirical agreement of an r-parity $c' \in \mathcal{B}$ can be bounded by Hoeffding's inequality as follows:

$$\mathbf{Pr}_{\hat{X} \sim \mathrm{EX}_\eta(c^*,D)} \left[\hat{\mathrm{agr}}_{\hat{X}}(c') - \mathbf{E}_{\hat{X} \sim \mathrm{EX}_\eta(c^*,D)}[\hat{\mathrm{agr}}_{\hat{X}}(c')] > t\right] < e^{-t^2/m}.$$

By the union bound we have that $\forall c_i, c_j$ s.t. $c_i + c_j = c' \in \mathcal{B}$,

$$\mathbf{Pr}_{\hat{X} \sim \mathrm{EX}_\eta(c^*,D)} \left[\hat{\mathrm{agr}}_{\hat{X}}(c') - \mathbf{E}_{\hat{X} \sim \mathrm{EX}_\eta(c^*,D)}[\hat{\mathrm{agr}}_{\hat{X}}(c')] > t\right] < n^r e^{-t^2/m}.$$

Hence $t = \sqrt{mr \log(n/\delta)}$ suffices to bound by $(1-\delta)$ the probability that *all* pairs of $n^{r/2}$ agree on no more than $t + \mathbf{E}[\hat{\mathrm{agr}}_{\hat{X}}(c')]$ positions. We can now, with probability $1-\delta$, bound the maximum agreement between two parities comprising a parity in $\mathcal{B}$ by

$$\begin{aligned} \max_{c' \in \mathcal{B}} \left(\hat{\mathrm{agr}}_{\hat{X}}(c')\right) &\le \max_{c' \in \mathcal{B}} \left(\mathbf{E}[\hat{\mathrm{agr}}_{\hat{X}}(c')]\right) + \sqrt{mr \log(n/\delta)} \\ &\le (1-\epsilon')m + \sqrt{mr \log(n/\delta)}. \end{aligned} \tag{1}$$

Furthermore, we know that $\mathbf{E}[\hat{\mathrm{agr}}_{\hat{X}}(c^*)] = (1-\eta)m$ and can similarly bound from below the empirical agreement $\hat{\mathrm{agr}}_{\hat{X}}(c^*)$ to get with probability $1-\delta$,

$$\hat{\mathrm{agr}}_{\hat{X}}(c^*) \ge (1-\eta)m - \sqrt{m \log(1/\delta)}. \tag{2}$$

We now rely on the following observation. If two $\frac{r}{2}$-parities c_1, c_2 comprise the target c^*, i.e. $c_1 + c_2 = c^*$, then their corresponding points $\langle c_1 \cdot x_1, c_1 \cdot x_2, \ldots, c_1 \cdot x_m \rangle$ and $\langle c_2 \cdot x_1 + \ell(x_1), c_2 \cdot x_2 + \ell(x_2), \ldots, c_2 \cdot x_m + \ell(x_m) \rangle$ in $\mathcal{H}$ are, by Equation 1, w.h.p. within Hamming distance $m - (1 - \epsilon')m - \sqrt{mr \log(n/\delta)}$, whereas if $c_1 + c_2 \in \mathcal{B}$, then by Equation 2, these points are at distance at least $m - (1 - \eta)m + \sqrt{m \log(1/\delta)}$. Hence by finding an approximate closest pair, with parameters properly set, we can find a pair c_1, c_2 such that $c_1 + c_2 \notin \mathcal{B}$, which suffices for learning. To do this, we appeal to Theorem 1, where we can set $N = O(n^{r/2})$ (the number of half-parities) and $d = m$ (the sample complexity).

We choose[2]

$$m = \frac{r \log(n/\delta) \omega(1)}{(\epsilon' - \eta)^2} \tag{3}$$

and

$$\begin{aligned}\rho &= \frac{m - (1 - \epsilon')m - \sqrt{mr \log(n/\delta)}}{m - (1 - \eta)m + \sqrt{m \log(1/\delta)}} \\ &= \frac{\epsilon' - (\epsilon' - \eta)/\sqrt{\omega(1)}}{\eta + (\epsilon' - \eta)\sqrt{\log(1/\delta)}/\sqrt{r \log(n/\delta)\omega(1)}} \\ &= \frac{\epsilon'}{\eta} - o(1).\end{aligned}$$

This ensures that the method in Theorem 1 will return two half parities composing a parity of error $< \epsilon'$ with probability $\geq 1 - 2\delta$.

All that is left is to analyze the running time of the method above, which, using Theorem 1 gives

$$O(dN^{1+1/\rho^2+o(1)}) = O\left(mn^{\left(1+\left(\frac{\eta}{\epsilon'}\right)^2+o(1)\right)r/2}\right).$$

Substituting in m from Equation 3 and substituting $\epsilon' = \epsilon + \eta - 2\epsilon\eta$ gives the statement of the theorem. □

For all settings of ϵ and η this beats the brute-force $O(n^r)$ bound.

Corollary 1. *For all $0 < \delta < \frac{1}{2}$ and $0 \leq \eta < \frac{1}{2}$, the class of r-sparse parities can be learned exactly under the uniform distribution using $m = \frac{r \log(n/\delta)\omega(1)}{(1-2\eta)^2}$ samples and a running time of*

$$\frac{\log(1/\delta) n^{\left(1+(2\eta)^2+o(1)\right)r/2}}{(1 - 2\eta)^2}.$$

This bound holds even for adversarial noise.

Proof. We set $\epsilon = 1/2$ in Theorem 5 and note that because every wrong parity has error $1/2$, if a parity has true error rate below $1/2$, it must be correct, and the target is therefore learned exactly. Furthermore, Feldman et al. [9] show that for the uniform distribution, an algorithm for learning r-parities for random noise works for adversarial noise, without a blow-up in the running time. □

[2] Note that $\omega(1) = \omega_n(1)$.

4 Improved Bounds for Improper Learning of r-Parities

In this section we present an algorithm which gives up on proper learning, but can learn parities under noise without a dependence on the error rate ϵ in the exponent. The following theorem holds for $\epsilon > \eta$ and uses the noise-tolerant boosting algorithm of Kalai and Servedio [13].

Theorem 6. *For any $0 < \epsilon < \frac{1}{2}$, $0 < \delta < \frac{1}{2}$, $0 \leq \eta < \frac{1}{2}$, and distribution D over $\{0,1\}^n$, the class of r-parities can be learned improperly in time*

$$\text{poly}\left(n, \log\frac{1}{\delta}, \frac{1}{\epsilon-\eta}, \frac{1}{1-2\eta}\right) n^{\left(1+(2\eta)^2+o(1)\right)r/2}.$$

Proof. Our idea here is to use the argument in Theorem 5 in order to obtain a parity c' such that

$$\mathbf{Pr}_{x\sim \mathrm{EX}_\eta(c^*,D)}[c'(x) \neq \ell(x)] < 1/2 - 1/n.$$

In order to do this we use the approximation factor ρ in the nearest neighbor search set to

$$\rho \ = \ \frac{1}{\eta}(1/2 - 1/n) \ = \ \frac{1}{2\eta} - o(1).$$

This gives us a noise-tolerant $\frac{1}{n}$-weak learner in time

$$\text{poly}(\log\frac{1}{\delta}, \frac{1}{\epsilon-\eta}, \frac{1}{1-2\eta}) n^{\left(1+(2\eta)^2+o(1)\right)r/2},$$

which can be further used together with Theorem 2 to give us the final improved result. This multiplies our running time and sample complexity by a factor of poly(n), which goes into the $o(1)$ in the exponent in our bound. □

5 Application to Learning Juntas and DNF

Using the results of the previous section and that learning juntas with noise reduces to learning parities with noise under the uniform distribution [9], we get an algorithm for learning sparse juntas with noise better than by brute-force.

Theorem 3 implies the following Corollary.

Corollary 2. *For all $0 < \delta < \frac{1}{2}$, $0 \leq \eta < \frac{1}{2}$, r-juntas on n variables can be learned exactly in time*

$$\text{poly}\left(\log\frac{1}{\delta}, \frac{1}{1-2\eta}, 2^r\right) n^{\left(1-2^{1-r}(1-2\eta)+2^{1-2r}(1-2\eta)^2+o(1)\right)r}$$

under the uniform distribution.

Proof. We combine Corollary 1 and Theorem 3 to get that r-juntas can be learned in time

$$r2^{2r}\frac{\log(1/\delta)n^{\left(1+(2\eta')^2+o(1)\right)r/2}}{(1-2\eta')^2},$$

where $\eta' = 1/2 - 2^{-r}(1-2\eta)$. Replacing η' completes the proof. □

We can now specialize Corollary 2 for the noiseless case.

Corollary 3. *For all $0 < \delta < \frac{1}{2}$, r-juntas with on n variables can be learned exactly in time*

$$\text{poly}\left(\log\frac{1}{\delta}, 2^r\right) n^{\left(1-2^{1-r}+2^{1-2r}+o(1)\right)r}$$

in the noise-free setting, under the uniform distribution.

We end this section by stating the implication to DNF of our Corollary 1 and Theorem 4 of Feldman et al. [9].

Corollary 4. *For all $0 < \epsilon < \frac{1}{2}$, $0 < \delta < \frac{1}{2}$, the class s-term DNF can be learned under the uniform distribution in time*

$$\text{poly}\left(\log\frac{1}{\delta}, \frac{1}{\epsilon}, s\right) n^{\left(1-\tilde{O}(\epsilon/s)+o(1)\right)\log\frac{s}{\epsilon}}.$$

We note that the log in the exponent is base 2. We further recall that, from Theorem 1, the $o(1)$ term in the exponent is $\log\log N/\log^{1/3} N$, where $N = n^{\log(\tilde{O}(s/\epsilon))}$. Therefore, the bound above is an improvement over Verbeurgt's bound [27] of $O(n^{\log\frac{s}{\epsilon}})$ when $s/\epsilon = o\left(\frac{\log^{1/3} n}{\log\log n}\right)$.

6 Discussion

In this paper, we give an algorithm for learning r-parities in time essentially $n^{r/2}$ and show implications of this result to related problems. Our results draw attention to a nice set of open problems related to the sparse version of LPN. We give a proper algorithm running in time $\text{poly}\left(\log\frac{1}{\delta}, \frac{1}{1-2\eta}\right) n^{(1+(2\eta)^2+o(1))r/2}$ for the uniform distribution and $\text{poly}\left(\log\frac{1}{\delta}, \frac{1}{\epsilon}, \frac{1}{1-2\eta}\right) n^{(1+\left(\frac{\eta}{\epsilon+\eta-2\epsilon\eta}\right)^2+o(1))r/2}$ for arbitrary distributions, for the r-sparse LPN problem.

For improper learning, we give an $\text{poly}\left(n, \log\frac{1}{\delta}, \frac{1}{\epsilon-\eta}, \frac{1}{1-2\eta}\right) n^{(1+(2\eta)^2+o(1))r/2}$ algorithm, which requires $\epsilon > \eta$ and uses $\text{poly}(n)$ samples. Obtaining an algorithm without the restriction of $\epsilon > \eta$, yet without the reliance on ϵ in the exponent, would be an interesting direction. One observation is that an improper learning algorithm achieving arbitrary low error can be converted to a proper learning algorithm for the LPN problem by drawing n examples from D, labeling them with the low-error algorithm (with the error parameter set $< \epsilon/n$), and running Gaussian elimination to find the correct parity. We note that a similar technique is used in Blum et al. [4] to obtain a proper learning algorithm for linear threshold functions. Another interesting direction would be to remove the dependence on η in the exponent.

We note that it is tempting to try to push the approach taken in the proof of Theorem 5 further by considering, say, $\frac{r}{3}$-parities. To improve our $n^{r/2}$ bound asymptotically, we would need an algorithm that, given a set of N points in the Hamming cube, finds 3 of them that ‘approximately sum’ up to 0, and runs in time substantially better than N^2. This problem is somewhat related to the famous 3-SUM question in computational geometry which asks if among N elements of a set of integers there exist 3 that sum to 0. Erickson [8] presents the N^2 as a barrier and shows it is intrinsic in many difficult problems, giving some weak evidence that extending our approach in this way is unlikely. We also point out that the nearest neighbor algorithm of Andoni and Indyk [2] runs in time essentially N^{1+1/ρ^2} and almost matches the lower bounds for data-structures [20, 21], suggesting again that obtaining even a $n^{\frac{r}{3}}$ algorithm for the r-parity problem may require fundamentally new techniques. It remains open whether a polynomial time algorithm exists for learning $\omega(1)$-parities.

The implication of our results to learning juntas brings up immediate questions of whether one can extend the non-trivial bound from Corollary 2 to arbitrary distributions, and furthermore, whether it is possible to improve the running time to n^{cr}, for some constant $c < 1$. As before, an important open problem is whether a polynomial time algorithm exists for learning $\omega(1)$-juntas.

Acknowledgments. We thank Avrim Blum and Adam Kalai for very helpful discussions and Alex Andoni for references on nearest neighbor search. We also thank the anonymous reviewers for useful comments.

References

[1] Alekhnovich, M., Braverman, M., Feldman, V., Klivans, A.R., Pitassi, T.: The complexity of properly learning simple concept classes. J. Comput. Syst. Sci. 74(1), 16–34 (2008)

[2] Andoni, A., and Indyk, P. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In: FOCS, pp. 459–468 (2006)

[3] Angluin, D., Laird, P.D.: Learning from noisy examples. Machine Learning 2(4), 343–370 (1987)

[4] Blum, A., Frieze, A.M., Kannan, R., Vempala, S.: A polynomial-time algorithm for learning noisy linear threshold functions. Algorithmica 22(1/2), 35–52 (1998)

[5] Blum, A., Furst, M.L., Jackson, J.C., Kearns, M.J., Mansour, Y., Rudich, S.: Weakly learning dnf and characterizing statistical query learning using fourier analysis. In: STOC, pp. 253–262 (1994)

[6] Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. J. ACM 50(4), 506–519 (2003)

[7] Buhrman, H., García-Soriano, D., Matsliah, A.: Learning parities in the mistake-bound model. Inf. Process. Lett. 111(1), 16–21 (2010)

[8] Erickson, J.: Lower bounds for linear satisfiability problems. In: SODA, Philadelphia, PA, USA, pp. 388–395 (1995)

[9] Feldman, V., Gopalan, P., Khot, S., Ponnuswami, A.K.: On agnostic learning of parities, monomials, and halfspaces. SIAM J. Comput. 39(2), 606–645 (2009)

[10] Goldreich, O., Levin, L.A.: A hard-core predicate for all one-way functions. In: STOC, pp. 25–32 (1989)

[11] Hopper, N.J., Blum, M.: Secure human identification protocols. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 52–66. Springer, Heidelberg (2001)

[12] Jackson, J.C., Lee, H.K., Servedio, R.A., Wan, A.: Learning random monotone DNF. In: Goel, A., Jansen, K., Rolim, J.D.P., Rubinfeld, R. (eds.) APPROX and RANDOM 2008. LNCS, vol. 5171, pp. 483–497. Springer, Heidelberg (2008)

[13] Kalai, A.T., Servedio, R.A.: Boosting in the presence of noise. J. Comput. Syst. Sci. 71(3), 266–290 (2005)

[14] Katz, J.: Efficient cryptographic protocols based on the hardness of learning parity with noise. In: IMA Int. Conf., pp. 1–15 (2007)

[15] Kearns, M.J.: Efficient noise-tolerant learning from statistical queries. In: STOC, pp. 392–401 (1993)

[16] Klivans, A.R., Servedio, R.A.: Toward attribute efficient learning of decision lists and parities. In: Shawe-Taylor, J., Singer, Y. (eds.) COLT 2004. LNCS (LNAI), vol. 3120, pp. 224–238. Springer, Heidelberg (2004)

[17] Kushilevitz, E., Mansour, Y.: Learning decision trees using the fourier spectrum. SIAM J. Comput. 22(6), 1331–1348 (1993)

[18] Lyubashevsky, V.: The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem. In: Chekuri, C., Jansen, K., Rolim, J.D.P., Trevisan, L. (eds.) APPROX 2005 and RANDOM 2005. LNCS, vol. 3624, pp. 378–389. Springer, Heidelberg (2005)

[19] Mossel, E., O'Donnell, R., Servedio, R.A.: Learning functions of k relevant variables. J. Comput. Syst. Sci. 69(3), 421–434 (2004)

[20] Panigrahy, R., Talwar, K., Wieder, U.: A geometric approach to lower bounds for approximate near-neighbor search and partial match. In: FOCS, pp. 414–423 (2008)

[21] Panigrahy, R., Talwar, K., Wieder, U.: Lower bounds on near neighbor search via metric expansion. In: FOCS, pp. 805–814 (2010)

[22] Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: STOC, pp. 333–342 (2009)

[23] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM 56(6) (2009)

[24] Sellie, L.: Learning random monotone dnf under the uniform distribution. In: COLT, pp. 181–192 (2008)

[25] Sellie, L.: Exact learning of random dnf over the uniform distribution. In: STOC, pp. 45–54 (2009)

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

[27] Verbeurgt, K.A.: Learning dnf under the uniform distribution in quasi-polynomial time. In: COLT, pp. 314–326 (1990)