Agnostic Learning and Noisy Parity Problem
(Short Notes)

Based in part on a paper by Kalai, Mansour and Verbin [KMV08]
Presented by Alexandr Andoni[*]

March 11, 2008

1 Introduction and definitions

Suppose we want to learn a function $f : \{0,1\}^n \to \{0,1\}$ from a number m of samples $(x_1, f(x_1)), \ldots (x_m, f(x_m))$ More precisely, we are given $(x_1, f(x_1)), \ldots (x_m, f(x_m))$ where $x_1, \ldots x_m$ are drawn from some fixed distribution D, and we are to produce a *hypothesis* $h : \{0,1\}^n \to \{0,1\}$ such that the *error* is small:

$$\mathrm{err}_D(h) = \Pr_{x \in D}[h(x) \neq f(x)] \leq \epsilon$$

for any $\epsilon > 0$ given in advance.

In general this is not possible with small m (consider random f). We can then consider the case when f is in some class of functions $\mathcal{C} \subseteq \{\phi : \{0,1\}^n \to \{0,1\}\}$ (such as decision trees) – this problem is called PAC learning class $\mathcal{C}$.

In agnostic learning, we do not restrict the class of functions f but instead weaken the error guarantee. Given a class of functions $\mathcal{C}$, we want to do as good as the best function from $\mathcal{C}$ does to learn f.

Definition 1.1 (Agnostic learning). *Consider any $n > 0$, a class of functions $\mathcal{C} \subseteq \{\phi : \{0,1\}^n \to \{0,1\}\}$, and input distribution D on $\{0,1\}^n$.*

An algorithm $\mathcal{A}$ agnostically learns *$\mathcal{C}$ under distribution D if, for any a target function $f : \{0,1\}^n \to \{0,1\}$, and for any $\epsilon > 0$, there exists $m = \mathrm{poly}(n, 1/\epsilon)$ such that given m samples $x_1, x_2, \ldots x_m$ and the values $f(x_1), \ldots f(x_m)$, the algorithm $\mathcal{A}$ produces a circuit $h : \{0,1\}^n \to \{0,1\}$ of size* $\mathrm{poly}(n, 1/\epsilon)$ *satisfying:*

$$\mathrm{err}_D(h) \leq \min_{\phi \in \mathcal{C}} \mathrm{err}_D(\phi) + \epsilon,$$

with probability of success $\geq 2/3$.

We can view agnostic learning also as follows: take some $f^* \in \mathcal{C}$ and flip $\eta < 1/2$ fraction of outputs to obtain the function f (more formally, the set of x's where f and f^* has η measure in D). The goal is to learn f with error at most $\eta + \epsilon$. Note that the flips are adversarial.

A third model is *classification noise* which is as above but the η flips are random. Formally, it's easiest to state for D being uniform over $\{0,1\}^n$. Then f is obtained from some $f^* \in \mathcal{C}$ by flipping $f(x)$, for every x, with probability $\eta < 1/2$.

[*] All inaccuracies in this note are due to the presenter.

Observation 1.2. *Suppose D is uniform. If we can learn f in the classification noise model, then the resulting hypothesis $h = \mathcal{A}((x_1, f(x_1)), \ldots (x_m, f(x_m)))$ is close to f^*, with probability $\geq 4/7$, that is*

$$\Pr_{x \in D}[h(x) \neq f^*(x)] \leq \frac{\epsilon}{1 - 2\eta}$$

holds with probability $\geq 4/7$ (over the choice of f).

If we are only concerned with the number of samples, then the three models are asymptotically equivalent (for fixed D). But not such if we consider the runtime of the learning algorithm $\mathcal{A}$.

1.1 Variants and Related Models

[Note: We will not include any references for this section because we do not know the (accepted) most representative sources, and it is well beyond this note to list all the nice results in these areas.]

In *proper learning* we require that the hypothesis h is also from the class $\mathcal{C}$. This requirement often makes the problems intractable (e.g., NP-hard).

In *query model*, the algorithm $\mathcal{A}$ is allowed to pick the samples $x_1, \ldots x_m$ as it wishes (not drawn from D).

Another variant is when f is "randomized" function, specifically for any x, $f(x)$ is a distribution (over $\{0, 1\}$). Result presented here actually holds for this variant, but we'll ignore this aspect.

We can also consider the *testing* problem, where, given samples (or the query model), one is to decide whether $f \in \mathcal{C}$ or $\min_{\phi \in C} \Pr_{x \in D}[\phi(x) \neq f(x)] \geq \epsilon$.

1.2 Noisy parity problem

For $S \subseteq [n]$, a parity function is a function $\chi_S : \{0, 1\}^n \to \{0, 1\}$ with $\chi_S(x) = \oplus_{i \in s} x_i$. From now on the class $\mathcal{C}$ will be the set of parities χ_S.

The *noisy parity problem* is to learn the parity class under the classification noise model. Similarly, *agnostic parity problem* is to learn the parity class under the agnostic learning model.

PAC-learning of parity with $m = O(n)$ samples is trivial: just do Gaussian elimination. Otherwise, the following are the known:

- [BKW03]: when D is uniform, noisy parity problem can be solved with $2^{O(n/\log n)}$ samples and time complexity.
- [Lyu05]: reduced the query complexity of [BKW03] to $n^{1+\delta}$ at the expense of increasing the time to $2^{O(n/\log \log n)}$, for any small $\delta > 0$.
- [FGKP06]: when D is uniform, agnostic noisy parity problem can be reduced to noisy parity problem, with only polynomial blowup in query and time complexity.

All the above results recover a hypothesis h that is an actual parity function.

Here, we present the following result of Kalai-Mansour-Verbin [KMV08]. It works for any distribution D, and it requires $m = 2^{O(n/\log n)}$ queries and time, matching the performance of [BKW03] and [FGKP06]. The hypothesis is generally not a parity function.

Theorem 1.3 ([KMV08], Main Theorem)**.** *There exists an algorithm $\mathcal{A}$ such that, for any distribution D and any function $f : \{0,1\}^n \to \{0,1\}$, given $m = 2^{O(n/\log n)}$ samples $((x_1, f(x_1)), \dots (x_m, f(x_m)))$, outputs a circuit $h : \{0,1\}^n \to \{0,1\}$ such that, with probability at least* 0.99*,*

$$\mathrm{err}_D(h) \leq \min_{\phi \in \mathcal{C}} \mathrm{err}_D(\phi) + 2^{-n^{0.99}}.$$

The runtime of $\mathcal{A}$ is $2^{O(n/\log n)}$.

The noisy parity problem has connections to coding theory and cryptography.

First, note that noisy parity problem is roughly the same problem with decoding a random linear code. Specifically consider a code of length m with 2^n codewords described by the generator matrix X formed by concatenating vectors $x_1, x_2, \dots x_m$. Then, the algorithm $\mathcal{A}$ gets the matrix X and "message" $(f(x_1), \dots f(x_m))$ which is some codeword corrupted by η fraction of errors.

Second, noisy parity problem, under larger alphabets, has been used as the "hard problem" in crypto applications [Reg05]. In fact, [Reg05] also showed that better algorithms for noisy parity problem would give a better quantum algorithms for SVP and SIVP for mild approximation range (depends on the alphabet size).

2 Proof Outline

We'll have three steps in our presentation.

2.1 Step 1: Noisy Parity in $2^{O(n/\log n)}$ time

We'll go over the original [BKW03] algorithm that learns *noisy* parity under *uniform* distribution. Hereon, we let

$$\eta = \min_{\phi \in \mathcal{C}} \mathrm{err}_D(\phi).$$

Theorem 2.1 ([BKW03])**.** *Let a, b be such that $ab = n$. Then we can solve the noisy parity problem under uniform distribution in* $\mathrm{poly}((1-2\eta)^{-2^a}, 2^b)$ *time and sample complexity.*

We use the theorem with $a = \frac{\log n}{1000}$ and $b = 1000\frac{n}{\log n}$, and suppose $\eta < 1/2$, then get $m = 2^{O(n/\log n)}$.

2.2 Step 2: Weak learner for agnostic parity

It turns out we can twick the algorithm from above to work for agnostic parity learning under any distribution D, but we get a weaker guarantee on the error.

Lemma 2.2 ([KMV08])**.** *Let a, b be such that $ab = n$ and $2^a = o(b)$. Then there exists a learning algorithm $\mathcal{A}$ that, for any distribution D, and function f, given $m =$* $\mathrm{poly}((1-2\eta)^{-2^a}, 2^b, 1/\epsilon)$ *samples, outputs a hypothesis circuit $h : \{0,1\}^n \to n$ such that*

$$\mathrm{err}_D(h) \leq \frac{1}{2} - \frac{(1-2\eta)^{2^a}}{2} + 2^{-b}.$$

Time complexity is $\mathrm{poly}(m)$*.*

As before, we use $a = \frac{\log n}{1000}$ and $b = 1000\frac{n}{\log n}$, and suppose $\eta < 1/2$. Then

$$\mathrm{err}_D(h) \leq \frac{1}{2} - \frac{(1-2\eta)^{n^{0.001}}}{2} + 2^{-2^{\sqrt{n}}} \leq \frac{1}{2} - \frac{(1-2\eta)^{n^{0.001}}}{3}.$$

In other words, h does a bit better than random guessing. We can exploit this with an *agnostic booster.*

2.3 Step 3: Agnostic Boosting

Let $0 < \gamma \leq \alpha \leq 1/2$, and let m denote the number of samples.

Definition 2.3 ((α, γ, m)-weak learner)**.** *A learning algorithm is* (α, γ, m)-weak learner *if it satisfies the following. For any* $\epsilon > 0$*, distribution* D*, function* f*, given* m *labelled samples from* D*, the algorithm* $\mathcal{A}$ *outputs hypothesis (circuit)* $h : \{0,1\}^n \to \{0,1\}$ *such that, with probability* $\geq 2/3$*, we have*

$$\left(\min_{\phi\in\mathcal{C}} \mathrm{err}_D(\phi) \leq \frac{1}{2} - \alpha\right) \implies \left(\mathrm{err}_D(h) \leq \frac{1}{2} - \gamma\right).$$

In words, if the best function from $\mathcal{C}$ can do α-better than random guessing, then the weak learner should be able to do a bit better than random guessing. In particular the algorithm from the the 2nd step is a weak learner with $\alpha = n^{-0.99}$ (in fact even $1/2 - \eta$, but we need a smaller α) and $\gamma = \frac{(1-2\eta)^{n^{0.001}}}{3}$.

Using a weak learner, Kalai-Mansour-Verbin design the following agnostic boosting algorithm.

Lemma 2.4 ([KMV08])**.** *There is an agnostic learning algorithm that, given any (α, γ, m)-weak learning algorithm $\mathcal{A}$, and an $\epsilon > 0$, outputs a hypothesis circuit $h : \{0,1\}^n \to \{0,1\}$ such that, with probability $\geq 2/3$:*

$$\mathrm{err}_D(h) \leq \min_D(\phi) + \alpha + \epsilon.$$

The sample and time complexity is $\mathrm{poly}(m, 1/\gamma, 1/\epsilon)$*. The number of calls to the weak learner is* $\mathrm{poly}(1/\gamma, 1/\epsilon)$*.*

Applying this boosting algorithm to the weak learner from Step 2 with $\alpha = n{-}0.99$ and $\gamma = \frac{(1-2\eta)^{n^{0.001}}}{3}$, we obtain a circuit hypothesis h satisfying:

$$\mathrm{err}_D(h) \leq \eta + n^{-0.99} + \epsilon$$

proving our main theorem.

References

[BKW03] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, 2003.

[FGKP06] Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. New results for learning noisy parities and halfspaces. *Proceedings of the Symposium on Foundations of Computer Science*, 0:563–574, 2006.

[KMV08] Adam Kalai, Yishay Mansour, and Elad Verbin. On agnostic boosting and parity learning. *Proceedings of the Symposium on Theory of Computing*, 2008. To appear.

[Lyu05] Vadim Lyubashevsky. The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem. *RANDOM*, 2005.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC '05: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 84–93, New York, NY, USA, 2005. ACM.