Computational Learning Theory - Hilary Term 2018
9 : Agnostic Learning

Lecturer: Varun Kanade

So far in all the learning frameworks we've studied, we've made an assumption that there is some "ground truth" target function that we attempt to learn. Our goal has been to identify a hypothesis that is close to this target, with respect to the target distribution. Learning algorithms are given access to the target function in the form of labelled observations, which in some cases may be noisy. In this lecture, we'll drop the assumption of a ground-truth target completely; it is for this reason that the framework is called *agnostic* learning. As there is no longer a well-defined notion of target, our goal will be to identify a hypothesis that is competitive with respect to the best concept from a particular class.

1 Agnostic Learning

We'll focus our attention on boolean functions, although real-valued functions and more general loss functions can also be considered in the agnostic learning framework. We will (annoyingly) switch between $\{0,1\}$ and $\{-1,1\}$ as the range of boolean functions (and also bit-values) from time to time, and use the one that is mathematically convenient for the particular calculation at hand. Let us formally define the framework and then making a few observations.

Definition 1 (Agnostic Learning)**.** *Let C be a concept class and H a hypothesis class. We say that C is agnostically learnable using H, if there exists a learning algorithm L that for all $n \geq 1$, for any D over $X_n \times \{0,1\}$, for all $0 < \delta < 1$ and for all $0 < \epsilon < \frac{1}{2}$, with access to a sample of size $m(\epsilon, \delta, n, \mathrm{size}(c))$ drawn independently from the distribution D, and inputs $\epsilon, \delta, \mathrm{size}(c)$, outputs $h \in H_n$ that with probability at least $1-\delta$, satisfies,*

$$\mathrm{err}(h; D) \leq \mathsf{opt} + \epsilon,$$

where $\mathrm{err}(h; D) = \mathbb{P}_{(x,y)\sim D}\left[h(x) \neq y\right]$ *and* $\mathsf{opt} = \min_{c \in C_n} \mathrm{err}(c; D)$.

Furthermore, we say that C is efficiently agnostically learnable *if there exists a polynomially evaluatable H, such that C is agnostically learnable using H and the corresponding algorithm L runs in time polynomial in n, $\frac{1}{\epsilon}$, $\frac{1}{\delta}$ and* $\mathrm{size}(c)$.

Before we investigate *learnability* in the agnostic framework, let us make a few brief observations.

- We no longer assume that the observed data is perfectly labelled according to some concept $c \in C$. In fact, we make no assumption at all about a target concept (clean or corrupted). Instead, we just assume that we are getting labelled data from some process. We'd like our algorithm to guarantee that if the data is well-explained by some concept from a class, then our algorithm will find a hypothesis that's almost as good. This is much closer in spirit to machine learning methods in practice, where of course we have no knowledge of the functional form of the true target. Nevertheless, we attempt to find linear separators, decision trees, boosted random forests, kernel SVMs, neural networks, *etc.* that fit the data "well".

- If we restrict the joint distribution, D, over $X_n \times \{0,1\}$ to be such that there exists $c \in C_n$, such that the support of D is contained in the set $\{(x, c(x)) \mid x \in X_n\}$, then we recover the probably approximately correct (PAC) framework. In this case, clearly we have $\mathsf{opt} = 0$, thus we require that $\mathrm{err}(h; D) \leq \epsilon$.

- The PAC framework with random classification noise can also be viewed as a restriction of the agnostic learning framework, where conditioned on x, $(x, c(x))$ has probability mass $1-\eta$ and $(x, 1-c(x))$ has probability mass η.[1] In this setting, any function $f : X_n \to \{0,1\}$ must have $\mathrm{err}(f; D) \geq \eta$, thus $\mathsf{opt} = \eta$ and is achieved by the target, c. If we run an agnostic learning algorithm with the accuracy parameter set to $(1-2\eta)\epsilon$, we obtain a hypothesis h with the desired guarantee, *i.e.*, $\mathbb{P}_{x \sim D_X}\left[h(x) \neq c(x)\right] \leq \epsilon$, where D_X is the marginal distribution over X_n.

What classes are agnostically learnable? The above observations suggest that agnostic learning is challenging—at the very least, algorithms for agnostic learning imply algorithms for PAC learning in the presence of random classification noise. If we restrict our attention to *efficient sample complexity* and ignore the running time, then agnostic learning is characterised very well by the VC-dimension. Let $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subseteq X_n \times \{0,1\}$ be a sample of size m. For any $h : X_n \to \{-1, 1\}$, define

$$\widehat{\mathrm{err}}(h; S) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}(h(x_i) \neq y_i)$$

The following theorem relates the empirical error on a sample to the expected error with respect to the underlying distribution. We will not give a proof here, but the proof is similar to that of other results of this type that we've seen earlier.

Theorem 2. *Let H be a hypothesis class with VC dimension d and let D be any distribution over $X \times \{0,1\}$. Then there exists $c_0 > 0$ such that for all $\epsilon, \delta > 0$, if S is a sampe of size m drawn independently from D, where $m \geq \frac{c_0}{\epsilon^2}\left(d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)$, then with probability at least $1-\delta$, it holds for every $h \in H$ that $|\widehat{\mathrm{err}}(h; S) - \mathrm{err}(h; D)| \leq \epsilon$.*

Let us see how this result may be applied to yield learning algorithms. Let C be a concept class for which we are interested in designing an agnostic learning algorithm. Let H be some hypothesis class such that $C \subseteq H$; we consider a more general H as the problem of minimising empirical error using H might be easier than if using C (as in the case of 3-TERM-DNF vs 3-CNF). We choose S be a sample of size m that is large enough so that the above theorem may be applied with confidence parameter δ and accuracy parameter $\epsilon/2$. Let $h \in H$ be the hypothesis that minimises $\widehat{\mathrm{err}}(h; S)$ and let $c^* \in C$ be the concept that minimises $\mathrm{err}(c; D)$, so that $\mathrm{err}(c^*; D) = \min_{c \in C} \mathrm{err}(c, D) = \mathsf{opt}$. Thus, we have:

$$\begin{aligned} \mathrm{err}(h; D) &\leq \widehat{\mathrm{err}}(h; S) + \frac{\epsilon}{2} && \text{Using Theorem 2} \\ &\leq \widehat{\mathrm{err}}(c^*; S) + \frac{\epsilon}{2} && \text{As } c^* \in H \text{ and } h \in H \text{ was chosen to minimise } \widehat{\mathrm{err}}(h; S) \\ &\leq \mathrm{err}(c^*; D) + \epsilon && \text{Using Theorem 2} \end{aligned}$$

The difficulty is, of course, in finding $h \in H$ that minimises $\widehat{\mathrm{err}}(h; S)$. In Section 2, we'll see a result showing that *proper* agnostic learning linear halfspaces is hard unless $\mathsf{NP} = \mathsf{RP}$. In Section 3, we'll see that even for *improper* agnostic learning, it would be highly surprising if efficient agnostic learning algorithms could be obtained for classes such as CONJUNCTIONS and linear halfspaces. If one is not concerned with computational complexity, then the results obtained using Theorem 2 are essentially tight; the following theorem shows this.

Theorem 3. *Any algorithm for agnostically learning a concept class C with VC dimension d that achieves an error of at most $\mathsf{opt}+\epsilon$ with probability at least $1-\delta$ requires $\Omega\left(\frac{1}{\epsilon}^2\left(d + \log \frac{1}{\delta}\right)\right)$ examples.*

[1] Here we are assuming X_n is finite; in any case, we'll ignore issues related to measurability in this course.

We will not give a proof of this theorem, but let us understand why the linear dependence on $\frac{1}{\epsilon^2}$ is necessary in the case of agnostic learning. (The correponding dependence for PAC learning is linear in $\frac{1}{\epsilon}$.) Let us consider a very simple hypothesis class, consisting of only two hypotheses, the constant functions which take the values 0 and 1. Suppose the data is generated using a coin with bias for HEADS either $\frac{1}{2}+\epsilon$ or $\frac{1}{2}-\epsilon$, *i.e.,* a coin is tossed and if HEADS appears the label returned is 1 and otherwise the label is 0. In the former case, the algorithm must output the constant hypothesis 1, and in the latter case the constant hypothesis that always predicts 0. By computing the variance of the random variable that is the number of HEADS, it can be seen that unless at least $\frac{1}{\epsilon^2}$ observations are made, it is impossible to distinguish between the two cases. (In the PAC setting, we are interested in separating a coin with bias 0 from those with bias at least ϵ; in this case, if we toss $\frac{1}{\epsilon}$ times we expect to see at least one HEADS if the bias is not 0.)

Remark: Proofs of Theorems 2 and 3 can be found in the textbook by Mohri et al. (2012, Chap. 3).

2 Hardness of Proper Agnostic Learning

In this section, we'll focus on *proper* agnostic learning, *i.e.,* when the hypothesis class, H, from which the algorithm chooses the output hypothesis is the same as the concept class, C. We will show that *proper* agnostical learning of linear halfspaces is hard unless $\mathsf{NP} = \mathsf{RP}$. The proof is similar to other such results that we've seen in the context of PAC-learning.

Theorem 4. *The class of linear halfspaces,* $\mathsf{LTF} = \bigcup_{n\geq 1} \mathsf{LTF}_n$*, where,*

$$\mathsf{LTF}_n = \{x \mapsto \mathbb{1}_{\geq 0}(w \cdot x + w_0) \mid w \in \mathbb{R}^n, \|w\|_2 = 1, w_0 \in \mathbb{R}\},$$

is not efficiently *agnostically learnable using* LTF *unless* $\mathsf{NP} = \mathsf{RP}$.

Proof. We will reduce from $\mathsf{MAX\text{-}INDSET}$, the problem of finding the largest independent set in a graph, which is known to be NP-hard. Let $G = (V, E)$ be a graph; a set $I \subseteq V$ is an independent set if there does not exist $(i, j) \in E$, such that $i \in I$ and $j \in I$, *i.e.,* an independent set is a set of vertices with no edges among them.

Given a graph $G = (V, E)$, we will construct $S \subseteq \mathbb{R}^n \times \{0, 1\}$. For vertex i, let $v_i \in \mathbb{R}^n$ denote the point that is 0 in all co-ordinates except i and let the i^{th} co-ordinate be 1. For every edge, (i, j), let $e_{i,j} \in \mathbb{R}^n$ denote the point that has 0 in all co-ordinates except i and j, and both the i^{th} and the j^{th} co-ordinate are 1. Notice that $e_{i,j} = v_i + v_j$, an observation that we will use later on. Let $\mathbf{0}$ be the origin in $\mathbb{R}^n$. Consider the set S of labelled points,

$$S = \{(v_i, 1) \mid i \in V\} \cup \{(e_{i,j}, 0) \mid (i, j) \in E\} \cup \{(\mathbf{0}, 0)\}$$

We will count weighted error—each point in S is given some non-negative weight and the total weight is 1. In particular, we'll set the weight of $(\mathbf{0}, 0)$ to be $\frac{2}{3}$ and the remaining $\frac{1}{3}$ weight is distributed equally among all the other points in S.

Let $I \subseteq V$ be the largest independent set in G, and let $\bar{I} = V \setminus I$. Consider the following linear halfspace, $f_I(x)$ given by

$$f_I(x) = \mathbb{1}_{\geq 0}\left(\sum_{i\in I} x_i - \sum_{i\in \bar{I}} x_i - \frac{1}{2}\right)$$

Let us compute the weighted error made by f_I on S. The point $(\mathbf{0}, 0)$ is classified correctly. All the points $(e_{i,j}, 0)$ are also classified correctly, since each edge (i, j) has at least one end-point that is not in I. Finally, all the vertices in I are classified correctly, while those in $\bar{I}$ are classified incorrectly. Thus, the weighted error made by f_I on S is given by $\frac{|V|-|I|}{3(|V|+|E|)}$.

The argument that a proper agnostic learning algorithm gives, with probability at least $1-\delta$, a linear halfspace that minimises the weighted error on S is essentially the same as in other such proofs. We set $\epsilon = \frac{1}{10(|V|+|E|)}$, and D to be the distribution given by the weights of points in S and that has support only over S. Thus, any halfspace that has weighted error at most $\mathsf{opt} + \epsilon$, where opt is the least weighted error made by any linear halfspace on S, essentially has to produce such a halfspace (as any additional mistake increases the weighted error by at least $\frac{1}{3(|V|+|E|)}$). Thus, using the proper agnostic learning algorithm, we get a randomised algorithm, that with probability at least $1-\delta$, finds a linear halfspace that minimises the weighted error on S.

The last part that remains to be shown is that if we can find a linear halfspace that minimises the weighted error on S, then we can find a maximum independent set. Let $g_{(w,w_0)}(x) = \mathbb{1}_{\geq 0}(w \cdot x + w_0)$ be such a linear halfspace. As there exists a linear halfspace with weighted error at most $\frac{1}{3}$, namely one that classifies the point $(\mathbf{0}, 0)$ correctly, we know that the error of $g_{(w,w_0)}$ is at most $\frac{1}{3}$. Thus in particular, $(\mathbf{0}, 0)$ is classified correctly by the halfspace $g_{(w,w_0)}$, which implies that $w_0 < 0$. Thus, it suffices to compare $g_{(w,w_0)}$ and f_I in terms of the mistakes made on the set $S' := S \setminus \{(\mathbf{0}, 0)\}$, as both of these linear halfspaces classify the point $(\mathbf{0}, 0)$ correctly.

Let $\bar{J} = \{i \mid w \cdot v_i + w_0 < 0\}$ and let $F = \{(i,j) \mid w \cdot e_{i,j} + w_0 \geq 0\}$. Clearly, the number of mistakes made by the $g_{(w,w_0)}$ on S' is $|\bar{J}| + |F|$. The number of mistakes made by f_I on S' is $|\bar{I}|$. Let $J = V \setminus \bar{J}$. As $g_{(w,w_0)}$ was chosen to have the least weighted error on S and both $g_{(w,w_0)}$ and f_I clasify $(\mathbf{0}, 0)$ correctly, it must be the case that $|\bar{J}| + |F| \leq |\bar{I}|$—alternatively $|J| \geq |I| + |F|$. Note that J may not be an independent set, but we'll describe how to obtain an independent set from J that is at least as large as I. Suppose there is an edge (i,j) such that $i \in J$ and $j \in J$. We claim that it must be the case that $(i,j) \in F$. This follows by using the facts that $w \cdot v_i + w_0 \geq 0$, $w \cdot v_j + w_0 \geq 0$, $e_{i,j} = v_i + v_j$ and that $w_0 < 0$. In particular, this means

$$w \cdot e_{i,j} + w_0 \geq w \cdot (v_i + v_j) + 2w_0 \geq 0$$

Thus, for any edge (i,j) with both endpoints in J, we can remove one of the points from J. As $|J| \geq |I| + |F|$ and every such edge must be from F, in the end we are left with a set that is at least as large as I.

Thus, the existence of a proper agnostic learning algorithm for LTF implies $\mathsf{RP} = \mathsf{NP}$. □

The above theorem shows that in the agnostic setting, proper learning linear halfspaces, a problem that was easy in the PAC framework, is already hard. On the problem sheets you have been asked to prove that even learning monotone conjunctions is hard. This provides some indication that agnostic learning is significantly more challenging than PAC learning. Of course, one may hope that allowing the learning algorithm to output hypotheses from larger classes would overcome some of these difficulties. In the next section, we show that it is unlikely to be the case.

3 Hardness Results for Improper Agnostic Learning

In this section we'll show that agnostic learning continues to be challenging, even when learning algorithms are allowed to output any polynomial-time evalutable hypothesis. Obviously, we are not able to show these results unconditionally, or even on relatively standard assumptions such as $\mathsf{RP} \neq \mathsf{NP}$. Instead, we'll reduce other believed-to-be-hard learning problems to agnostic learning certain concept classes. This also gives us the opportunity to explicity state two well-known challenging learning problems in computational learning theory.

3.1 Learning Parities with Noise

Recall that the class $\mathsf{PARITIES}_n$ consists of functions of the form $f_S(x) = \bigoplus_{i \in S} x_i$ for $x \in \{0,1\}^n$. Let us list some key results about about this problem.

- In the PAC framework, with access to the oracle $\mathsf{EX}(f_S, D)$, there is a simple learning algorithm for $\mathsf{PARITIES}$. The algorithm essentially performs Gaussian elimination, treating parities as linear functions from $\mathbf{GF}(2)^n \to \mathbf{GF}(2)$.
- In the statistical query framework, any algorithm that makes queries to $\mathsf{STAT}(f_S, \mathcal{U})$, where $\mathcal{U}$ is the uniform distribution on $\{0,1\}^n$ with tolerance at least τ, must make $\Omega(\tau^2 2^n)$ queries. As we've seen earlier, this result is unconditional. Thus, the class $\mathsf{PARITIES}$ cannot be *efficiently* learnt in the statistical query framework.
- In the PAC framework with random classification noise, *i.e.,* with access to the oracle $\mathsf{EX}^\eta(f_S, D)$, this problem is widely believed to be hard. One of the hard cases is believed to be the case where $D = \mathcal{U}$, the uniform distribution over $\{0,1\}^n$. As a statistical query algorithm for $\mathsf{PARITIES}$ does not exist, we cannot apply the general reduction to obtain an algorithm for learning $\mathsf{PARITIES}$ in the presence of random classification noise. The current best algorithm due to Blum et al. (2003) has running time $2^{O(n/\log n)}$. Cryptosystems have been designed based on the assumption that learning $\mathsf{PARITIES}$ with noise is hard (and more often, on the harndness of learning noisy linear functions on larger finite fields; see (Regev, 2009) for further details).

We will show how an agnostic learning algorithm for linear halfspaces yields an algorithm for learning $\mathsf{PARITIES}$ with noise. Rather than give a formal proof, we will describe the key ideas in sufficient detail for the interested student to complete the formal proof. We will consider the concept class of $\mathsf{MAJORITIES}$. As in the case of parities, these are functions defined over some subset of the variables, the difference being that the function evaluates to 1 if at least half the bits in the subset are 1 and 0 otherwise. Formally, $\mathsf{MAJORITIES}_n$, contains functions g_S defined as,

$$g_S(x) = \mathbb{1}_{\geq 0}\left(\sum_{i \in S} x_i - \frac{|S|}{2}\right)$$

for all $S \subseteq [n]$.

To make calculations slightly simpler, we'll use the transformation $0 \mapsto 1$ and $1 \mapsto -1$ both for inputs and outputs. This means that a parity on subset S is simply defined as $f_S(x) = \prod_{i \in S} x_i$ and the majority is defined as $g_S(x) = \mathrm{sign}\left(\sum_{i \in S} x_S - \frac{1}{2}\right)$. Furthermore, we'll assume that the target parity is defined on some set S, such that $|S|$ is even.[2] Then, we show that $\left|\mathop{\mathbb{E}}_{x \sim \mathcal{U}}\left[g_S(x) f_S(x)\right]\right| = \Omega\left(\frac{1}{\sqrt{|S|}}\right)$. For any input $x \in \{-1,1\}^n$, let $\bar{x}_S$ denote the input obtained by flipping every bit of x in the set S. Both parity and majority are symmetric functions on the set S, *i.e.,* they only depend on the number of bits that are -1, but not on specific bits. If x is such that k bits in S are -1 for $k \in \{0, 1, \ldots, |S|/2 - 1\}$, then $g_S(x) = -g_S(\bar{x}_S)$ and $f_S(x) = f_S(\bar{x}_S)$. Now for $k = 0, \ldots, |S|/2 - 1$, the probability distribution (under the uniform distribution over $\{-1,1\}^n$) of having k bits be -1 is exactly the same as having $|S| - k$ bits be -1. Thus, the only contribution to $\mathop{\mathbb{E}}_{x \sim \mathcal{U}}\left[g_S(x) f_S(x)\right]$ comes from points $x \in \{-1,1\}^n$, such that exactly $|S|/2$ out of the $|S|$ bits in S are -1. Thus,

$$\left|\mathop{\mathbb{E}}_{x \sim \mathcal{U}}\left[g_S(x) f_S(x)\right]\right| = \frac{\binom{|S|}{|S|/2}}{2^{|S|}} = \Omega\left(\frac{1}{\sqrt{|S|}}\right)$$

[2] Otherwise, it is possible to guess one of the bits in the target parity, say by trying all n bits, and flip the sign of the labels. In at least one case, we've reduced the problem to learning a parity on an even set.

Now suppose that the data was noisy, then we get that the correlation of the function g_S, with the observed data (labels treated as being in $\{-1, 1\}$), is at least $\Omega\left(\frac{1-2\eta}{\sqrt{|S|}}\right)$. Essentially, this is $\underset{x\sim\mathcal{U},Z}{\mathbb{E}}\left[f_S(x)g_S(x)Z\right]$, where Z is a random variable that takes value 1 with probability $1-\eta$ and -1 with probability η. Thus, an agnostic learning algorithm for learning halfspaces would give us some $h : \{-1, 1\}^n \to \{-1, 1\}$, such that $\underset{x\sim\mathcal{U}}{\mathbb{E}}\left[f_S(x)h(x)\right] = \Omega\left(\frac{1-2\eta}{\sqrt{|S|}}\right)$, with probability at least $1-\delta$.

Once we have obtained h, we are able to evaluate h on any point in the set $\{-1, 1\}^n$. It is well-known that there is an algorithm, that given black-box access to h, and in time polynomial in $\frac{1}{\delta}$ and $\frac{1}{\tau}$, outputs all subsets T, such that $\left|\underset{x\sim\mathcal{U}}{\mathbb{E}}\left[h(x)f_T(x)\right]\right| \geq \tau$. Essentially, the set of all parity functions forms an orthogonal basis for functions defined from $\{-1, 1\} \to \mathbb{R}$, with respect to the uniform distribution over $\{-1, 1\}^n$. This algorithm finds all the large "Fourier" coefficients of any $h : \{-1, 1\}^n \to \mathbb{R}$. The interested student is referred to the excellent survey by Mansour (1994).

3.2 PAC-Learning DNF

Let $\mathsf{CONJUNCTIONS}_n$ denote the class of conjunctions over $\{0, 1\}^n$. Let $s(\cdot)$ be a polynomial. Define the class,

$$\mathsf{DNF}_{n,s(n)} = \{c_1 \vee \cdots \vee c_k \mid c_i \in \mathsf{CONJUNCTIONS}_n, k \leq s(n)\}$$

and $\mathsf{DNF}_s = \bigcup_{n\geq 1} \mathsf{DNF}_{n,s(n)}$, to be the class of boolean functions over n variables that can be expressed as a DNF formula with at most s terms. Thus, an efficient algorithm for learning DNF_s runs in time polynomial in n, $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$ (the parameter size(c) is not necessary as it is always polynomial in n). Let us recall some known results regarding learning DNF_s.

- Learning DNF_s was posed an open problem by Valiant (1984). It has remained open despite much effort for the last three decades.
- Any statistical query algorithm with tolerance τ, such that τ^{-1} is bounded by a polynomial in n, has to make $n^{\Omega(\log n)}$ queries to the $\mathsf{STAT}(c, D)$ oracle. The key idea here is that parities over $\log n$ bits can be expressed as a DNF formula of polynomial size; there are $n^{\Omega(\log n)}$ such parity functions. Details appear in the paper by Blum et al. (1994).
- The current best known algorithm for PAC learning DNF_s is due to Klivans and Servedio (2001) and has running time $2^{\widetilde{O}(n^{1/3})}$, where the notation $\widetilde{O}(\cdot)$ hides poly-logarithmic factors.
- Recent work by Daniely and Shalev-Shwartz (2014) has shown that under plausible, but as yet unproven complexity-theoretic assumptions, learning DNF_s is hard for some polynomial $s(\cdot)$.

Thus, the problem of PAC learning DNF_s is widely considered to be hard. We will show that a *computationally efficient* algorithm for agnostically learning conjunctions would imply a *computationally efficient* algorithm for PAC learning DNF_s, thus providing evidence, that agnostically learning conjunctions, even improperly, is hard.

As in the previous section, we will focus on the key ideas rather than provide a formal proof. Let $\varphi \in \mathsf{DNF}_s$ and let D be some distribution over $\{0, 1\}^n$. Let

$$\mathsf{opt} = \min_{c\in\mathsf{CONJUNCTIONS}_n} \mathrm{err}(c; \varphi, D),$$

i.e., opt is the least error made by a conjunction when the data is labelled according to a DNF formual φ. The claim is that $\mathsf{opt} \le \frac{1}{2} - \frac{1}{10s(n)}$. In fact, you have already proved this on one of the problem sheets. You were asked to show that some function from the set $H = \{0, 1, x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n\}$ is a good candidate weak hypothesis for learning conjunctions; the same applies to disjunctions. A DNF formula is essentially a disjunction on an expanded input space, which includes the evaluation of all possible conjunctions on the original n-bit input. This means that the output h of an agnostic learning algorithm, when run with $\epsilon = \frac{1}{20s(n)}$ and with access to examples drawn from $\mathsf{EX}(\varphi, D)$, satisfies, $\mathrm{err}(h; \varphi, D) \le \frac{1}{2} - \frac{1}{20s(n)}$. Such an algorithm serves as a $\frac{1}{20s(n)}$-weak learner and hence using a boosting algorithm, such as Adaboost, yields a PAC-learning algorithm for learning DNF_s. Thus, it is unlikely that there is a computationally efficient agnostic learning algorithm for CONJUNCTIONS.

Remark 5. *In fact, the problem of agnostically learning conjunctions may be significantly harder than PAC learning polynomial-size DNF formulas. The current best known algorithm for agnostically learning conjunctions is due to Kalai et al. (2008) and runs in time* $2^{\tilde{O}(\sqrt{n})}$.

4 Distribution-Specific Agnostic Learning

As we've seen by now, agnostic learning is highly challenging, and if one is to obtain positive results, further relaxations will need to be made in terms of what is asked of the learning algorithm. One such assumption that is common in the literature, and which we'll study, is the requirement that the learning algorithm work no matter what the distribution over the data. We'll relax this requirement by making assumptions regarding the marginal distribution D_X over X. In keeping with the spirit of agnostic learning, we'll make no assumptions whatsoever about how the data is labelled. Formally, for a joint distribution D over $X \times \{-1, 1\}$, let D_X denote the marginal distribution over X. We will only require the learning algorithm to succeed provided the marginal distribution, D_X, comes from a relatively benign class of distributions. Our focus will be on the setting where D_X is the spherical normal distribution in $\mathbb{R}^n$, $\mathcal{N}(0, \sigma^2 I)$.

Let us first formally define distribution-specific agnostic learning.

Definition 6 (Distribution-specific Agnostic Learning)**.** *Let $\mathcal{D}$ be a class of distributions over X. We say that C is agnostically learnable with respect to the class of distributions $\mathcal{D}$, if there exists a polynomially evaluatable hypothesis class H and a learning algorithm L, that for all $n \ge 1$, for every D over $X_n \times \{-1, 1\}$, such that the marginal distribution D_X is in $\mathcal{D}$, for every $0 < \delta < 1$ and $0 < \epsilon < 1$, with access to $\mathsf{EX}(D)$ and with inputs ϵ, δ and* $\mathrm{size}(c)$, *outputs h that with probability at least $1 - \delta$, satisfies*

$$\mathrm{err}(h; D) \le \min_{c \in C} \mathrm{err}(c; D) + \epsilon.$$

We've dropped the notion of *efficient learning* from the definition. This is because most of the algorithms that we'll design will not be polynomial in $\frac{1}{\epsilon}$, rather we will get some dependence of the form $n^{\frac{1}{\epsilon^\kappa}}$ for some constant κ. This means that the overall running time is polynomial only if ϵ is chosen to be a constant. Even after the restriction on the marginal distribution and fixing ϵ to be a constant, designing learning algorithms is quite a challenge. We will see the main ideas that are required to design and analyse an algorithm for agnostically learning halfspaces. The full algorithm and detailed proofs can be found in the article by Kalai et al. (2008).

4.1 Agnostically Learning Halfspaces

We will restrict our attention to the case where the marginal distribution D_X is the spherical normal distribution with covariance matrix $\frac{1}{n}I$, *i.e.,* $D_X = \mathcal{N}(0, \frac{1}{n}I)$. As part of the algorithm, we will use real-valued functions and real-valued losses as intermediate steps, before producing a boolean function as an output hypothesis. Along the way, we'll use some results from approximation theory.

Definition 7 (ϵ-ℓ_1-Approximation). *Let $\Phi = \{\phi_1, \ldots, \phi_M\}$ be a collection of* basis functions, *where $\phi_i : X \to \mathbb{R}$. We say that a concept class C can be ϵ-ℓ_1-approximated by Φ with respect to a distribution D_X over X, if for every $c \in C$, there exist $a_1, \ldots, a_M \in \mathbb{R}$, such that,*

$$\mathop{\mathbb{E}}_{x \sim D_X}\left[\left|c(x) - \sum_{j=1}^{M} a_j \phi_j(x)\right|\right] \leq \epsilon$$

The following result shows that ϵ-ℓ_1-approximability of a class C is sufficient for agnostic learning. Observe the polynomial dependence on M, the size of Φ, of the sample and computational complexity of the algorithm. The challenge is to show that a concept class C can be approximated by using a basis that is not *too large*.

Theorem 8. *If C can be ϵ-ℓ_1-approximated by some collection of basis functions $\Phi = \{\phi_1, \ldots, \phi_M\}$ with respect to distribution D_X, then C is agnostically learnable with respect to D_X. Furthermore, the sample complexity and running time of the algorithm is polynomial in $M, \frac{1}{\epsilon}, \frac{1}{\delta}$.*

Proof. Let $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ be obtained from a distribution D over $X \times \{-1, 1\}$ such that the marginal distribution over X is D_X. We perform the following optimisation problem. Find $a_1, \ldots, a_M$ that minimise the following objective:

$$J(a_1, \ldots, a_M; S) = \frac{1}{m} \sum_{i=1}^{m} \left| \sum_{j=1}^{M} a_j \phi_j(x_i) - y_i \right|$$

It is left as an exercise to show that this optimisation problem can be expressed as a linear program. (See the Appendix of the article by Kalai et al. (2008) for the solution.)

Output Hypothesis: Let $h : X \to \{-1, 1\}$ be the boolean function given by

$$h(x) = \operatorname{sign}\left(\sum_{j=1}^{M} a_j \phi_j(x) - t\right),$$

where t is chosen to minimise $\widehat{\text{err}}(h; S)$ from among hypotheses of this form. Note that the values $a_1, \ldots, a_M$ were already obtained as part of the optimisation algorithm, thus only t is chosen so that the empirical error is minimised. This is just a simple one dimensional problem and can be solved easily.

We claim that $\widehat{\text{err}}(h; S) \leq \frac{1}{2} J(a_1, \ldots, a_M; S)$. To see this, imagine that $t \in [-1, 1]$ was chosen uniformly at random (denoted by $t \sim \mathcal{U}([-1, 1])$) and let h_t be the corresponding hypothesis; for $x \in \mathbb{R}^n$, let $z = \sum_{j=1}^{M} a_j \phi_j(x)$ and suppose $y \in \{-1, 1\}$. Clearly, if $z < -1$ or $z > 1$, for every $t \in [-1, 1]$.

$$\mathbb{1}(\operatorname{sign}(z - t) \neq y) = \frac{|\operatorname{sign}(z) - y|}{2} \leq \frac{|z - y|}{2}.$$

On the other hand, if $z \in [-1, 1]$, then for $t \in [z, 1]$ if $y = 1$ and for $t \in [-1, z]$ if $y = 1$, it is the case that $\mathbb{1}(\operatorname{sign}(z - t) \neq y) = 1$, hence,

$$\mathop{\mathbb{E}}_{t \sim \mathcal{U}([-1,1])}\left[\mathbb{1}(\operatorname{sign}(z - t) \neq y)\right] = \frac{|z - y|}{2}$$

Thus, in all cases, we obtain $\mathop{\mathbb{E}}_{t \sim \mathcal{U}([-1,1])}\left[\widehat{\text{err}}(h_t; S)\right] \leq \frac{1}{2} J(a_1, \ldots, a_M; S)$. This in particular implies that there exists $t \in [-1, 1]$ such that $\text{err}(h_t; S) \leq \frac{1}{2} J(a_1, \ldots, a_M; S)$, and as the output hypothesis h was obtained by choosing t that minimises the number of mistakes on the sample S, it must be the case that $\widehat{\mathrm{err}}(h; S) \leq \frac{1}{2} J(a_1, \ldots, a_M; S)$. We will choose m to be large enough so that $|\widehat{\mathrm{err}}(h; S) - \mathrm{err}(h; S)| \leq \epsilon/4$ for every h that is a linear halfspace over the basis functions $\phi_1, \ldots, \phi_M$ with probability at least $1 - \frac{\epsilon}{4}$. Note that this is possible for some m that is still polynomially bounded in M, $\frac{1}{\epsilon}$, and $\frac{1}{\delta}$. Note that h is still a function of the sample S and hence is considered as a random variable. We thus obtain the following (after some calculations of conditional expectations):

$$\mathop{\mathbb{E}}_{S \sim D^m} \left[\mathrm{err}(h; D)\right] \leq \frac{1}{2} \mathop{\mathbb{E}}_{S \sim D^m} \left[J(a_1, \ldots, a_M; S)\right] + \frac{\epsilon}{4}$$

Let $c^* \in C$ be such that,

$$\mathrm{err}(c^*; D) = \mathsf{opt} = \min_{c \in C} \mathrm{err}(c; D).$$

Let $a_1^*, \ldots, a_M^*$ be such that $\mathop{\mathbb{E}}_{x \sim D_X} \left[\left|\sum_{j=1}^M a_j^* \phi_j(x) - c^*(x)\right|\right] \leq \epsilon$. Using the fact that $a_1, \ldots, a_M$ were chosen to optimise $J(a_1, \ldots, a_M; S)$ and $a_1^*, \ldots, a_M^*$ constitute a feasible solution, we get

$$\begin{aligned}
\mathop{\mathbb{E}}_{S \sim D^m} \left[\mathrm{err}(h; D)\right] &\leq \frac{1}{2} \mathop{\mathbb{E}}_{S \sim D^m} \left[J(a_1^*, \ldots, a_M^*; S)\right] + \frac{\epsilon}{4} \\
&\leq \frac{1}{2} \mathop{\mathbb{E}}_{S \sim D^m} \left[\frac{1}{m} \sum_{i=1}^m |y_i - c^*(x_i)|\right] + \frac{1}{2} \mathop{\mathbb{E}}_{S \sim D^m} \left[\frac{1}{m} \sum_{i=1}^m \left|c^*(x_i) - \sum_{j=1}^M a_j^* \phi_j(x_i)\right|\right] + \frac{\epsilon}{4} \\
&\leq \frac{1}{2} \mathop{\mathbb{E}}_{(x,y) \sim D} \left[|y - c^*(x)|\right] + \frac{1}{2} \mathop{\mathbb{E}}_{x \sim D_X} \left[\left|c^*(x) - \sum_{j=1}^M a_j^* \phi_j(x)\right|\right] + \frac{\epsilon}{4}
\end{aligned}$$

As $c^*(x) \in \{-1, 1\}$ and $y \in \{-1, 1\}$, $\mathop{\mathbb{E}}_{(x,y) \sim D} \left[|y - c^*(x)|\right] = 2\mathrm{err}(c^*; D) = 2\mathsf{opt}$. Hence, we have,

$$\mathop{\mathbb{E}}_{S \sim D^m} \left[\mathrm{err}(h; D)\right] \leq \mathsf{opt} + \frac{3\epsilon}{4}$$

Finally, we apply Markov's inequality to obtain:

$$\mathbb{P}_{S \sim D^m} \left[\mathrm{err}(h; D) \geq \mathsf{opt} + \frac{7\epsilon}{8}\right] \leq \frac{\mathop{\mathbb{E}}_{S \sim D^m} \left[\mathrm{err}(h; D)\right]}{\mathsf{opt} + \frac{7\epsilon}{8}} \leq \frac{\mathsf{opt} + \frac{3\epsilon}{4}}{\mathsf{opt} + \frac{7\epsilon}{8}} \leq 1 - \frac{\epsilon}{16}$$

Repeating the entire algorithm $N = \Theta\left(\frac{1}{\epsilon} \log \frac{1}{\delta}\right)$ times and choosing the hypothesis with the best performance among the N obtained hypothesis on a freshly drawn sample of size $O\left(\frac{1}{\epsilon^2} \log \frac{N}{\delta}\right)$ yields a hypothesis that with probability at least $1 - \delta$, satisfies $\mathrm{err}(h; D) \leq \mathsf{opt} + \epsilon$. □

Approximating Linear Halfspaces using Polynomials

What is left to prove is that there exists a suitable family of basis functions that can ϵ-ℓ_1-approximate linear halfspaces when the marginal distribution D_X is $\mathcal{N}(0, \frac{1}{n} I)$. The next theorem shows that monomials of degree $O\left(\frac{1}{\epsilon^4}\right)$ provide such a family of basis functions, or alternatively, degree $O\left(\frac{1}{\epsilon^4}\right)$-multivariate polynomials provide ϵ-ℓ_1-approximations to linear halfspaces for the distribution $\mathcal{N}(0, \frac{1}{n} I)$.

Theorem 9. *Let Φ be the set of all monomials over $\{x_1, \ldots, x_n\}$ of degree at most d, then for some $d = O\left(\frac{1}{\epsilon^4}\right)$, linear halfspaces in $\mathbb{R}^n$ are ϵ-ℓ_1-approximated by Φ with respect to $\mathcal{N}(0, \frac{1}{n} I)$.*

In order to prove this theorem, we will use the following lemma, whose proof we can be found in the article by Kalai et al. (2008).

Lemma 1. *For any $d \geq 1$ and for any $\theta \in \mathbb{R}$, there is a degree d univariate polynomial, $p_{d,\theta}$, such that,*

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (p_{d,\theta}(z) - \operatorname{sign}(z - \theta))^2 e^{-z^2/2} \, \mathrm{d}z = O\left(\frac{1}{\sqrt{d}}\right)$$

Proof of Theorem 9. Let $f(x) = \operatorname{sign}(w \cdot x + w_0)$ denote some linear halfspace, where $w \in \mathbb{R}^n$ with $\|w\|_2 = 1$ and $w_0 \in \mathbb{R}$. Let $\theta = -\sqrt{n} w_0$ and let P be a multivariate polynomial defined as $P(x) = p_{d,\theta}(\sqrt{n}(w \cdot x))$. Note that P is a degree d multivariate polynomial. Then, we have

$$\mathop{\mathbb{E}}_{x \sim D_X} \left[|P(x) - f(x)| \right] \leq \sqrt{\mathop{\mathbb{E}}_{x \sim D_X} \left[(P(x) - f(x))^2 \right]}$$

As w is a unit vector and $D_X = \mathcal{N}(0, \frac{1}{n} I)$, the marginal (univariate) distribution of $\sqrt{n}(w \cdot x)$ is $\mathcal{N}(0, 1)$. Substituting $z = \sqrt{n}(w \cdot x)$ we have,

$$\mathop{\mathbb{E}}_{x \sim D_X} \left[(P(x) - f(x))^2 \right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (p_{d,\theta}(z) - \operatorname{sign}(z - \theta))^2 \, \mathrm{d}z = O\left(\frac{1}{\sqrt{d}}\right)$$

Thus, for some $d = O\left(\frac{1}{\epsilon^4}\right)$ we get the required result. □

As a consequence of Theorems 8 and 9, we have the following corollary.

Corollary 1. *The class of linear halfspaces in $\mathbb{R}^n$ is agnostically learnable with respect to the spherical normal distribution with sample complexity and running time polynomial in $n^{\frac{1}{\epsilon^4}}$, $\frac{1}{\delta}$, $\frac{1}{\epsilon}$.*

5 Notes

In a sense, the agnostic learning framework pre-dates the PAC learning framework; the essential components are present in the statistical learning framework formulated by Vapnik in the 1970s. Kearns et al. (1994) formulated the framework as described in the network, bringing the study of boolean functions and computational complexity at the forefront. Without restrictions on the marginal distribution over X, positive results are known essentially only for trivial concept classes for boolean functions. For real-valued functions it it possible to have such results for some general classes for certain kinds of convex *loss* functions.

In the computational learning theory and theoretical computer science community, much effort has been expended on designing algorithms that work under restrictions on the marginal distribution. We have studied one such result in this lecture. We can consider learning using membership queries in the agnostic learning setting. A few more positive results are known in that case, including learning parities and decision trees (cf. (Gopalan et al., 2008)). However, when restrictions are not made on the marginal distribution, membership queries provided no added benefit in the case agnostic learning! The intuition behind this is that as the labels have nothing to do with any specific target function, querying the label at specific points is unlikely to be helpful; this was formalised and proved by Feldman (2009).

The techniques to design agnostic learing algorithms typically use results from approximation theory. The proof of Lemma 1 uses the fact that Hermite polynomials form an orthogonal basis for square integrable functions with respect to the standard normal distribution. For boolean functions, positive results are mainly known for the case where the marginal distribution over X is the uniform (or some other product) distribution on the boolean cube. In this case, techniques from discrete Fourier analyis can be used to obtain results in learning theory and beyond (cf. (O'Donnell, 2014)).

References

Avrim Blum, Merrick Furst, Jeffrey Jackson, Michael Kearns, Yishay Mansour, and Steven Rudich. Weakly learning dnf and characterizing statistical query learning using fourier analysis. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 253–262. ACM, 1994.

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

Amit Daniely and Shai Shalev-Shwartz. Complexity theoretic limitations on learning DNFs. *CoRR, abs/1404.3378*, 1(2.1):2–1, 2014.

Vitaly Feldman. On the power of membership queries in agnostic learning. *Journal of Machine Learning Research*, 10(Feb):163–182, 2009.

Parikshit Gopalan, Adam Tauman Kalai, and Adam R Klivans. Agnostically learning decision trees. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 527–536. ACM, 2008.

Adam Tauman Kalai, Adam R Klivans, Yishay Mansour, and Rocco A Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

Michael J Kearns, Robert E Schapire, and Linda M Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2-3):115–141, 1994.

Adam R Klivans and Rocco Servedio. Learning dnf in $2^{\widetilde{o}(n^{1/3})}$ time. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 258–265. ACM, 2001.

Yishay Mansour. Learning boolean functions via the fourier transform. In *Theoretical advances in neural computation and learning*, pages 391–424. Springer, 1994.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning.* MIT Press, 2012.

Ryan O'Donnell. *Analysis of boolean functions.* Cambridge University Press, 2014.

Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009.

Leslie Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.