CSE 513T Theory of Artificial Intelligence & Machine Learning April 12, 2018

Lecture 24

Lecturer: Alex Durgin *Scribe: Han Liu*

Contents

1 Recall Improper learning **1**

2 Constraint Satisfaction Problems **1**

3 Scattered samples & learning **2**
3.1 The method 3

4 RSAT Hardness of DNF Learning **4**

1 Recall Improper learning

Recall We say that a PAC learner $\mathcal{L}$ of $\mathcal{H}$ is imporper if $\mathcal{L}$'s output is not necessarily in $\mathcal{H}$ and $\mathcal{H}$ is imporperly learnable if such an $\mathcal{L}$ exists.

How do we usually show that it is hard to efficiently learn some $\mathcal{H}$? In the Proper case, we usually reduce some **NP**-hard problem to learn $\mathcal{H}$:

(1) We start with some **NP**-hard problem.

(2) For each instance I of the problem, we efficiently produce a sample S s.t. S is $\mathcal{H}$-realizable $\iff$ I is a positive instance.

(3) If there is an efficient proper learner $\mathcal{L}$ of $\mathcal{H}$, we can run it with $\delta = \frac{1}{4}, \epsilon = \frac{1}{2|S|}$, and oracle access to the uniform distribution on S.

Recall We used this technique to reduce 3-coloring to properly learning 3-term DNF.

But, this falls apart in the imporper case. In the case of negative instance I, while the associated sample S may not have an $\mathcal{H}$-realization, it is possible that the output class of an improper learner may realize S.

How do we get around this? Before this technique, the framework we mostly used have well-established is that of reducing presumably hard cryptographic problems to our $\mathcal{H}$. E.g. Under the discrete cube root assumption we showed it was hard to improperly learn Boolean circuits.

But progress with this technique has slowed and there are still many "flagship" problems left unanalyzed in the unrestricted settings. E.g. improperly learning DNF. We want to be able to tap some new pool of knowledge about families of (seemingly) hard problems, which brings us to the new(ish) technique of Daniely, Linial, and Shalev-Shwartz (2014) [1], which utilizes instead assumptions on the hardness of refuting random CSP.

2 Constraint Satisfaction Problems

Definition 1 *We call a function* $P : \{\pm 1\}^K \to \{0, 1\}$ *a K-predicate.*

Definition 2 *We call a function* $C : \{\pm 1\}^K \to \{0, 1\}$ *a P-constraint if* $C(x) = P(l_1(x), \ldots, l_k(x))$.

Example Let P=K-SAT, then a P-constraint is just a disjunction of K literals ($C(x) = \bigwedge_{i=1}^{K} l_i(x)$).

Definition 3 *We call a collection $J = \{C_1, \ldots, C_m\}$ of P-constraints a P-formula.*

Definition 4 *The usual problem we consider when talking about CSPs, denoted CSP(P), involves finding assignments that satisfy a maximal number of constraints in a given instance (P-formula).*

But we will consider a different but related problem over CSP instances from which we will reduce to learning our $\mathcal{H}$.

Definition 5 *The problem $\mathrm{CSP}^{\mathrm{rand}}_{m(n)}(P)$ is the task of distinguishing between satisfiable and uniformly random P-formula of size $m(n)$. More formally, a randomized algorithm $\mathcal{A}$ solves $\mathrm{CSP}^{\mathrm{rand}}_{m(n)}(P)$ if the following hold:*

(1) If J is satisfiable, then

$$\Pr_{\text{randomness of } \mathcal{A}}(\mathcal{A}(J) = \text{“satisfiable”}) \leq \frac{3}{4}$$

(2) If J is drawn uniformly at random, then w.p. $\geq 1 - o_n(1)$ over the draw of J

$$\Pr_{\text{randomness of } \mathcal{A}}(\mathcal{A}(J) = \text{“random”}) \geq \frac{3}{4}$$

If there is no such efficient $\mathcal{A}$, we say this problem is hard.

3 Scattered samples & learning

We also require some way to relate this general “distinguishing” style problem back to a learning problem:

Definition 6 *We call a collection of Distributions $\mathcal{D} = \left\{\mathcal{D}_n^{m(n)}\right\}$ an ensemble where each $\mathcal{D}_n^{m(n)}$ is a distribution over $(\{0,1\}^n \times \{0,1\}^{m(n)})^{m(n)}$*

Definition 7 *We say an ensemble $\mathcal{D}$ is $(p(n), \beta)$-scattered if for every labeling $f : \{0,1\}^n \to \{0,1\}$ and large enough n,*

$$\Pr_{S \sim \mathcal{D}_n^{m(n)}}(\mathrm{Err}_S(f) \leq \beta) < 2^{-p(n)}$$

We can think of scattered ensembles as distributions that generate very random or unrealizable examples.

Example Suppose $\mathcal{D}_n^{m(n)}$ is a distribution s.t. $(x_i, y_i) \sim \mathcal{D}_n^{m(n)}$. y is a Bernoulli r.v. w.p. $\frac{1}{2}$ (i.e. a fair coin flip). Then note the analysis of $\mathrm{Err}_S(f) \leq \frac{1}{4}$ is the same as bounding the probability of tossing $\leq \frac{m(n)}{4}$ heads in $m(n)$ fair coin flips. So, by Hoeffding's bound, this probability is $\leq e^{-2(\frac{1}{4})m(n)} < 2^{-\frac{1}{8}m(n)}$. Hence such a $\mathcal{D} = \left\{\mathcal{D}_n^{m(n)}\right\}$ is $(\frac{1}{8}m(n), 1/4)$-scattered.

Definition 8 *We denote task $\mathcal{H}^{\mathcal{D},\beta,p(n)}_{m(n)}$ of distinguishing $\mathcal{H}$-realizable from $(p(n), \beta)$-scattered samples more formally, we say random $\mathcal{A}$ solves this if the following hold:*

(1) If S is $\mathcal{H}$-realizable, then

$$\Pr(\mathcal{A}(S) = \text{“realizable”}) \geq 3/4$$

(2) If $S \sim \mathcal{D}_n^{m(n)}$, then w.p. $\geq 1 - o_n(1)$

$$Pr(\mathcal{A}(S) = \text{“unrealizable”}) \geq 3/4$$

Theorem 1 *$\exists\, \beta > 0$ s.t. $\forall\, d > 0$, $\exists\, (n^d, \beta)$-scattered $\mathcal{D} = \left\{\mathcal{D}_n^{m(n)}\right\}$ s.t. $H_{m(n)}^{\mathcal{D}, n^d, \beta}$ is hard then it is hard to improperly learn $\mathcal{H}$.*

Idea If we had efficient learner of $\mathcal{H}$, then it can write down at most a polynomial number of bits. In particular, This means that the total number of possible out put hypothesis is "small". So, by assumption, in the scattered case, each of these will do badly, and

Proof Toward a contradiction, let $\mathcal{L}$ efficiently learn $\mathcal{H}$. Let $M(n, 1/\epsilon, 1/\delta) < \text{poly}(n, 1/\epsilon, 1/\delta)$ be the total number of bits that $\mathcal{L}$ can ever read or write on that input. Fix $\beta = \epsilon, \delta = 1/4, M(n) < \text{poly}(n)$. Define

$$q(n) = M(n, 1/\epsilon, 4) + n.$$

By assumption, $\exists\, d > 0$ s.t. $n^d > q(n)$ and $\exists\, \mathcal{D}$ that is $\left\{n^d, \beta\right\}$-scattered, and thus $\{q(n), \beta\}$-scattered. Consider the following distinguisher for $\mathcal{H}^{\mathcal{D}, n^d, \beta}$ on input S:

- Run $\mathcal{L}$ on $(n, \beta, 1/4)$ with oracle access to the uniform distribution over S.
- Let h be its output. If $Err_S(h) < \beta$, return "realizable". Otherwise, return "unrealizable".

If S is $\mathcal{H}$-realizable: Trivial.
If $S \sim \mathcal{D}_n^{m(n)}$: We note that at most we can output $\leq 2^{q(n)-n}$ possible output hypothesis, and since S is drawn from a $\left\{n^d, \beta\right\}$-scattered distribution, by union bound over possible output S, the probability that $\mathcal{L}$ has error $< \beta$ is $\leq 2^{-q(n)} \cdot 2^{q(n)-n} = 2^{-n} = o_n(1)$. ■

3.1 The method

(1) Pick some mildly believable hardness assumption on refuting random CSP formula. i.e. $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P)$ is hard for some f and P.

(2) We find some $\beta > 0$ s.t. $\forall d > 0, \exists$ some (n^d, β)-scattered ensemble $\mathcal{D} = \left\{\mathcal{D}_n^{m(K)}\right\}$ for which we can pick appropriate K to be able to reduce $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P)$ to $\mathcal{H}_{m(n)}^{\mathcal{D}, n^d, \beta}$.

Observation 2 *We can frame $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P)$ in terms of a learning problem:*
Let $\mathbb{X}_p$ be the set of all P-constraints. Let $\mathcal{H}_p = \{h_x : \mathbb{X}_p \to \{0,1\}, h_x(c) = c(x) | x \in \{0,1\}^n\}$.
Then we see $J = \{C_1, \ldots, C_{n^{f(K)}}\}$ is satisfiable $\iff$ $\exists\ h_x \in \mathcal{H}_p$ that realizes $\{(C_1, 1), \ldots, (C_m, 1)\}$.

There are two problems left over after this new way of looking at $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P)$:

(1) This sample is not scattered.

(2) We are still only considering $\mathcal{H}_p$, not $\mathcal{H}$.

How are we going to overcome these 2 problems?

For (1), we will consider $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P', \neg P')$ (So in the "learning" interpretation for $\neg C$ or $\neg P$ constant, the associatied sample is $(C, 0)$).

Note If we could reduce $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P)$ to $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P', \neg P')$ in a random way that w.p. $\frac{1}{2}$ replaces a positive with a negation constraint, then, by our earlier example, the associated sample would be $(\frac{1}{\delta} n^{f(K)}, 1/4)$-scattered.

For (2), to overcome this point, we observe that if we show our $\mathcal{H}$ realizes $\mathcal{H}_p$, then we are good.

To summarize, $\text{CSP}^{\text{rand}}_{n^{f(K)}}(P) \to \text{CSP}^{\text{rand}}_{n^{f(K)}}(P', \neg P') \to \mathcal{H}_p^{\mathcal{D}, n^d, \beta} \to \mathcal{H}$.

4 RSAT Hardness of DNF Learning

First, we need a "mildly believable" hardness Assumption.

Assumption 3 (RSAT-hardness) *The problem* $\mathrm{CSP}^{\mathrm{rand}}_{n^{f(K)}}(K\text{-SAT})$ *is hard for some* $f(K) = \omega(1)$.

Evidence:

- ($\mathbf{P} \neq \mathbf{NP}$) We know it is hard to distinguish between satisfiable and K-SAT formula wose fraction of maximaly satisfiable constraint is $\in (1 - 2^{-k}, 1 - 2^{-k} + \epsilon)$ [2].
- Statistical Query Algorithms cannot solve this problem for formula of size $n^{k/2-\epsilon}$.
- It is also known that K-SAT formula of size $n^{k/2-\epsilon}$, have only exponentially sized refutation.

Now we need to scatter our random input K-SAT:

It would be really nice if we could iterate our formula and w.p. $\frac{1}{2}$ replaces a constrain with a random negative constraint. If input J is random, then we are good. But if J is satisfiable, then for any satisfying assignment X, the probability X satisfies a random negative constraint is 2^{-k}.

So we need to introduce a new predicate $(P' =) T_{K,M} : \{0,1\}^{KM} \to \{0,1\}$.

$$T_{K,M}(z) = (z_1 \vee \cdots \vee z_K) \wedge \cdots \wedge (z_{(M-1)K}, \ldots, z_{MK})$$

Note that $T_{K,M} = \neg\mathrm{Tribes}_{KM}$. Moreover $\Pr(\mathrm{Tribes}_{KM}(z) = 1) = 1 - (1 - 2^{-K})^M$.

Keeping this observation in mind, if we can just reduce $\mathrm{CSP}^{\mathrm{rand}}_{n^{f(K)}}(K\text{-SAT})$ to $\mathrm{CSP}^{\mathrm{rand}}_{n^{f(K)-1}}(T_{K,M})$ then we can just apply the process above for large enough M.

Theorem 4 $\mathrm{CSP}^{\mathrm{rand}}_{n^d}(K\text{-SAT})$ *efficiently reduces to* $\mathrm{CSP}^{\mathrm{rand}}_{n^{d-1}}(T_{K,M})$ *for* $M \leq \frac{n}{\log n}$.

Proof (sketch) Let (TAC) $\mathcal{A}$ solve $\mathrm{CSP}^{\mathrm{rand}}_{n^{f(K)-1}}(T_{K,M})$ efficiently consider following reduction.
On input $J = C_1, \ldots, C_{n^d}$:

Partition J into n-sized blocks of the form $\{C_{t+1}, \ldots, C_{t+n}\}$;
for $t = 1$ **to** n^{d-1} **do**
  Let $C'_t = \varnothing$;
  for $r = 1$ **to** n **do**
    if $|C_t| < M$ **and** $\mathrm{Var}(C'_t) \cap \mathrm{Var}(C_{t+r}) \neq 0$ **then**
      $C'_t = C'_t \cup \{C_{t+r}\}$;
    end
  end
  if $|C_t| < M$ **then**
    return satisfiable
  end
  return $\mathcal{A}(\{C'_1, \ldots, C'_{n^{d-1}}\})$
end

If J is satisfiable: Trivial.
If J is random: All we need is to bound the probability that we get to the last return. We let $\mathbb{X}_{t+r}$ be the indicator r.v. for C_{t+r} that does not contain any seen variables. Then $\mathbb{X}_{t+r} > M$. ■

Theorem 5 $\mathrm{CSP}^{\mathrm{rand}}_{n^d}(T_{K,M})$ *efficiently reduces to* $\mathrm{CSP}^{\mathrm{rand}}_{n^d}(T_{K,M}, \neg T_{K,M})$ *for* $M \geq 2^{K+2} \log n^d$.

Proof We apply the previously explained process of w.p. $\frac{1}{2}$ replacing constrains with negative constraints. As we noted, this works in the random case. (The formula is random and the induced distribution is scattered.) What about the satisfiable case? By our observation on the ??? formula, we note that our transformed formula fails to be satisfiable w.p. $\leq n^d(1-2^{-K})^M \leq o_n(1) \quad (*)$.

$$
\begin{aligned}
(*) \quad \log(n^d(1-2^{-K})^M) &= \log n^d - M \log(\frac{1}{1-2^{-K}}) \\
&\leq \log n^d - M 2^{-K} \\
&\leq \log n^d - M 2^{-(K+1)} \\
&\leq \log n^d - 2 \log n^d = \log \frac{1}{n^d} \qquad \blacksquare
\end{aligned}
$$

Theorem 6 *If* $P : \{\pm 1\}^K \to \{0,1\}$ *can be realized by* T*-sized* DNF*, then* $\mathcal{H}_P$ *can be "realized" by* DNF^T.

References

[1] A. Daniely, N. Linial, and S. Shalev-Shwartz. From average case complexity to improper learning complexity. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 441–448. ACM, 2014.

[2] J. Håstad. Some optimal inapproximability results. *Journal of the ACM (JACM)*, 48(4):798–859, 2001.