6.842 Randomness and Computation December 6, 2017

Lecture 25

Lecturer: Ronitt Rubinfeld *Scribe: Jennifer Tang*

1 Amplifying Hardness: Yao's XOR Lemma

Goal: To "amplify hardness" by taking any slightly hard function (worst case hard function) f and turn it into a new actually hard function (average case hard function) f^*.

How will we do this? By showing that if a function is not hard in the average case, we can solve it in the worst case.

1.1 Yao's XOR lemma

Here's an example to understand the intuition behind Yao's XOR lemma: Suppose you have a δ-biased coin where the probability of heads is $1 - \delta$ (suppose $\delta \leq \frac{1}{2}$).

- We can correctly predict the result of one coin flip with probability $1 - \delta$ (by guessing heads)
- We can correctly predict the result of k coin flips with probability $(1 - \delta)^k$ (by guessing all heads)
- If we were asked to guess the parity of k coin flips (odd parity if there is an odd number of heads), we can correctly predict the parity with probability $\approx \frac{1}{2} + (1 - 2\delta)^k$. This approaches $\frac{1}{2}$ as k goes to infinity.

What we want to do is apply this not to coin flips, but to the function f. This is not so straightforward.

1.2 Plan

Note that this topic deals with functions on circuits as opposed to functions on Turing machines. Fix a class of circuits. The overall plan is to show:

Function f is wrong on some fraction δ of inputs for any circuit

$\Downarrow$ (using boosting)

There exists a measure where f is wrong on almost $\frac{1}{2}$ fraction of inputs with any circuit

$\Downarrow$ (using probabilistic argument)

There exists a subset of inputs such that f is wrong on $\frac{1}{2}$ the inputs with any circuit

$\Downarrow$ (using Yao's XOR lemma)

There exists a function f^* which is wrong on almost $\frac{1}{2}$ of all inputs with any circuit

We have amplified hardness by significantly increasing the proportion of inputs any circuit is wrong on.

1.3 Several Definitions

Definition 1 *Function* $f : \{\pm 1\}^n \to \{\pm 1\}$ *is* δ**-hard** *on distribution* $\mathcal{D}$ *for size* g *if for any Boolean circuit* C *with less than* g *gates*

$$Pr_{\mathcal{D}}[C(x) = f(x)] \leq 1 - \delta\,.$$

In other words, f is δ-hard if there is always an error on at least δ fraction of inputs given by distribution $\mathcal{D}$. For example, if $\mathcal{D}$ is uniform on binary n bit inputs, a function f is δ-hard for $\delta = 2^{-n}$ if more than one input always gives the wrong answer for any circuit. If f is δ-hard for $\delta = \frac{1}{2}$, then no circuit does better than randomly guessing the function. In such a case, we can always set $C(\cdot) = 1$ or $C(\cdot) = -1$.

Our goal is to find a function and distribution pair, $(f, \mathcal{D})$, which is δ-hard on approximately $\frac{1}{2}$ of the inputs under distribution $\mathcal{D}$.

For the next definition, recall

$$Adv_x(M) = \sum_x R_c(x)M(x)$$

where

$$R_c(x) = \begin{cases} +1 \text{ if } C(x) = f(x) \\ -1 \text{ if } C(x) \neq f(x) \end{cases}$$

Definition 2 *If* $\frac{Adv_c(M)}{\sum_x M(x)} \leq \varepsilon$ *for every circuit* C *with less than* g *gates, then* f *is* ε**-hardcore** *on* M *for size* g.

Note that the condition $\frac{Adv_c(M)}{\sum_x M(x)} \leq \varepsilon$ is equivalent to $Pr_M[C(x) = f(x)] \leq \frac{1}{2} + \frac{\varepsilon}{2}$ (where the probability is taken over the measure given by M).

Definition 3 *Let* $S \subseteq \{\pm 1\}^n$*, then* f *is* ε**-hardcore** *on* S *for size* g *if for every circuit* C *of size at most* g *is such that* $Pr[C(x) = f(x)] \leq \frac{1}{2} + \frac{\varepsilon}{2}$ *where the probability measure is uniform on the elements in set* S.

We have defined these terms so that we can show for every hard function f, there is a hardcore function on the set $S \subseteq \{\pm 1\}^n$.

1.4 Several Theorems: Hard functions have hardcore measure

Theorem 4 *Suppose* f *is a* δ*-hard function for the uniform distribution for size* g*. Let* $0 < \varepsilon < 1$*. Then there exists a measure* M *such that* $\mu(M) = \frac{\sum_x M(x)}{\#x} \geq \delta$ *such that* f *is* ε*-hardcore on* M *for size* $g' = \frac{1}{4}\varepsilon^2\delta^2 g$.

The proof of this theorem is given by boosting. Notice how the size of the circuit grows by a similar constant used in boosting.

Proof Given f, suppose there is no measure M that meets the condition of the theorem. Then, for every M such that $\mu(M) \geq \delta$, there is a circuit of size g' with $Adv_c(M) \geq \varepsilon$. Let this circuit be the "weak learner" in the boosting argument.

We can take the majority of the $\frac{1}{\varepsilon^2\delta^2}$ circuits of size g'. The output of each of the circuits of size g' is feed into one large majority gate which produces the final answer. By boosting, this predicts f with error less than δ. The total size of the circuit is $\frac{1}{\varepsilon^2\delta^2}g' + o(\frac{1}{\varepsilon^2\delta^2}) = \frac{1}{\varepsilon^2\delta^2}\frac{1}{4}\varepsilon^2\delta^2 g + o(\frac{1}{\varepsilon^2\delta^2}) < g$. This implies that f cannot be δ-hard for circuits of size g. ■

Using a probabilistic argument, we can get that if there is an ε-hardcore measure M for size g' where $2n < g < \frac{\varepsilon^2\delta^2}{8}\frac{2^n}{n}$ then there exists a 2ε-hardcore set S for f of size g where $|S| \leq \delta 2^n$.

The following theorem (or lemma rather) shows that if there is hardcore set then there is a function (a different one) which is hard to predict on all of the domain. The new function is created by taking a combination of XOR's of the original function.

Lemma 5 (Yao's XOR lemma) *Given* f *which is* ε*-hardcore for set* H *of size greater than* $\delta 2^n$ *for size* $g+1$*, the function*

$$f^{\oplus k}(x_1, ..., x_k) = f(x_1) \oplus f(x_2) \oplus ... \oplus f(x_k)$$

is $\varepsilon + 2(1-\delta)^k$*-hardcore for size* g *on the whole domain.*

Before we present the actual proof, we will provide an idea which is key to the proof. This idea does *not* work on its own, but it helpful for understanding the proof.

Assume the theorem is not true. Then there exists a circuit C with less than g gates, which that

$$Pr_{x_1,...,x_k}[C(x_1,...,x_k) = f^{\oplus k}(x_1,...,x_k)] \geq \frac{1}{2} + \frac{\varepsilon}{2} + (1-\delta)^k\,.$$

Here is a way to use $f^{\oplus k}$ to determine f: Given an input x, let $x_1 = x$. Let the rest of the inputs into $f^{\oplus k}$ each equal the zero vector, that is $x_2 = \underline{0}, ..., x_k = \underline{0}$. Let $b = \oplus_{i=2}^{k} f(\underline{0})$, then

$$f(x) = f^{\oplus k}(x, \underline{0}, ..., \underline{0}) \oplus b\,.$$

We can create a circuit for $f(x)$ by using $C(x, \underline{0}, ..., \underline{0}) \oplus b$. If

$$Pr[f(x) = C(x, \underline{0}, ..., \underline{0}) \oplus b] > \frac{1}{2} + \frac{\varepsilon}{2}$$

then we have a contradiction. The only problem here is that while $C(x_1, ..., x_k)$ approximates $f(x_1, ..., x_k)$ well on $\frac{1}{2} + \frac{\varepsilon}{2} + (1-\delta)^k$ of the inputs, setting the inputs $x_2, ..., x_k$ to zero vectors might not be one of the instances which $C(x_1, ..., x_k)$ is correct for. The proof below uses this idea but fixes this issue by evaluating on a better choice of $x_2, ..., x_k$.

Proof Let A_m be the event that exactly m of $x_1, ..., x_k$ are in H. Note that

$$Pr_{x_1,...,x_k}[A_0] \leq (1-\delta)^k\,.$$

The event A_0 are instances which have no x_i in H, so these are not "hard" to evaluate on. If we condition on not getting A_0, we must have

$$Pr_{x_1,...,x_k}[C(x_1,...,x_k) = f^{\oplus k}(x_1,...,x_k) | \cup_{m>0} A_m] > \frac{1}{2} + \frac{\varepsilon}{2}\,.$$

By averaging, we know that there is one $i > 0$ where

$$Pr_{x_1,...,x_k}[C(x_1,...,x_k) = f^{\oplus k}(x_1,...,x_k) | A_i] > \frac{1}{2} + \frac{\varepsilon}{2}\,.$$

With this value of i, the procedure is that given any $x \in H$, compute $f(x)$ by

1. Picking $x_1, ..., x_{i-1} \in H$ randomly
2. Picking $y_{i+1}, ..., y_k \in \bar{H}$ randomly
3. Picking a random permutation π of $(x_1, ..., x_{i-1}, x, y_{i+1}, ..., y_k)$

$$Pr_{x_1,...,x_k}[C(\pi(x_1,...,x_{i-1},x,y_{i+1},...,y_k)) = f^{\oplus k}(\pi(x_1,...,x_{i-1},x,y_{i+1},...,y_k))] > \frac{1}{2} + \frac{\varepsilon}{2}\,.$$

By averaging, there exists a specific choice of $\pi, x_1, ..., x_{i-1}, y_{i+1}, ...y_k$ and a bit $b = \oplus_{j=1}^{i-1} f(x_j) \oplus_{j=i+1}^{k} f(y_j)$ so that

$$Pr_{x \in H}[f(x) = C(\pi(x_1,...,x_{i-1},x,y_{i+1},...,y_k)) \oplus b] \geq \frac{1}{2} + \frac{\varepsilon}{2}$$

.

We will create the circuit for predicting f by hardcoding C with the choices of variables $x_1, ..., x_{i-1}, y_{i+1}, ...y_k$, the permutation π, and the bit b. The size of this circuit is less than $g+1$, so f is not ε-hardcore for size $g+1$ for on the set H.

■