6.842 Randomness and Computation March 26, 2014

Lecture 13

Lecturer: Ronitt Rubinfeld *Scribe: Suthee Ruangwises*

1 Lecture Overview

This lecture covers learning via Fourier coefficients. First, we will discuss examples of some functions and their Fourier representations. Then, we will introduce the low degree algorithm and its applications.

2 Examples of Functions and Fourier Representations

Example 1: Consider the $\overline{AND}$ function on input $x = (x_1, ..., x_k) \in \{\pm 1\}^k$:

$$\overline{AND}(x) = \begin{cases} 1, \text{ if } \forall i \in T = [k], x_i = 1 \\ -1, \text{ otherwise.} \end{cases}$$

First, we "booleanize" the output (but not the input) of the AND function by defining

$$f(x) = \begin{cases} 1, \text{ if } \forall i \in T, x_i = -1 \\ 0, \text{ otherwise.} \end{cases}$$

We have

$$\begin{aligned} f(x) &= \prod_{i \in T} \frac{1 - x_i}{2} \\ &= \sum_{S \subseteq T} \frac{(-1)^{|S|}}{2^k} \chi_S \end{aligned}$$

and

$$\begin{aligned} \overline{AND}(x) &= 2f(x) - 1 \\ &= -1 + \frac{2}{2^k} + \sum_{S \subseteq T, |S| > 0} \frac{(-1)^{|S|}}{2^{k-1}} \chi_S. \end{aligned}$$

Example 2: Consider the decision tree model of computation.

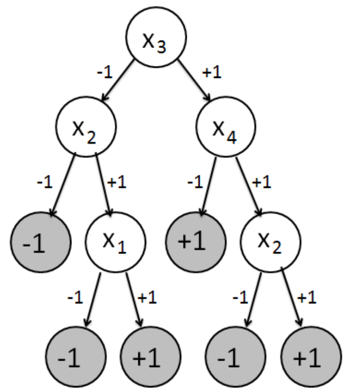

Figure 1: An example of a decision tree. Note that the left branch is always -1 and the right branch is always $+1$.

We define a path function

$$\begin{aligned} f_l(x) &= \prod_{x_i \in V_l} \frac{1 \pm x_i}{2} \\ &= \frac{1}{2^{|V_l|}} \sum_{S \subseteq V_l} (-1)^{\#\text{ left turns taken in } S} \chi_S \end{aligned}$$

when V_l is the set of variables visited on the path to leaf l. Note that the $\pm$ sign will be $-$ if we visit the left branch and $+$ if we visit the right branch of that node.

The value of each $f_l(x)$ is:

$$f_l(x) = \begin{cases} 1, & \text{if } x \text{ takes the path to } l \\ 0, & \text{otherwise.} \end{cases}$$

Note that all but one of $f_l(x)$ will be zero. Therefore, we can write $f(x)$ as

$$f(x) = \sum_{l \in \text{leaves}} f_l(x) val(l).$$

3 Fourier Concentration

Definition 1 *For $0 < \epsilon < 1$, a function $f : \{\pm 1\}^n \to \mathbb{R}$ has $\alpha(\epsilon, n)$-Fourier concentration if*

$$\sum_{S \subseteq [n], |S| > \alpha(\epsilon, n)} \hat{f}(S)^2 \leq \epsilon.$$

Example 1: If a function f depends on at most k variables, then

$$\sum_{|S|>k} \hat{f}(S)^2 = 0.$$

Example 2: $f = AND$ on $T \subseteq [n]$ has $\log(\frac{4}{\epsilon})$-Fourier concentration. Therefore,

- If $|T| \leq \log(\frac{4}{\epsilon})$, then $\sum_{|S| \geq \log(\frac{4}{\epsilon})} \hat{f}(S)^2 = 0$.
- If $|T| > \log(\frac{4}{\epsilon})$, then $\hat{f}(\phi)^2 = (1 - 2Pr[f(x) \neq \chi_\phi(x)])^2 = (1 - \frac{2}{2^{|T|}})^2 > 1 - \epsilon$. So, $\sum_{S \neq \phi} \hat{f}(s)^2 \leq \epsilon$. Therefore, f has 0-Fourier concentration.

4 Low Degree Algorithm

Given degree d, accuracy τ, and confidence δ, we do the following steps:

- Take $m = O(\frac{n^d}{\tau} \log(\frac{n^d}{\delta}))$ samples.
- Set $c_s \leftarrow$ estimate of $\hat{f}(x)$.
- Output $h(x) = \sum_{|S| \leq d} c_s \chi_S(x)$.

We use $sign(h(x))$ as hypothesis for $f(x) = \sum \hat{f}(S)\chi_S(x)$. We will prove that this estimation works.

Theorem 2 *If f has $d = \alpha(\epsilon, n)$-Fourier concentration, then h satisfies $E_x[(h(x) - f(x))^2] \leq \epsilon + \tau$ with probability at least $1 - \delta$.*

Note that for a boolean function f, this theorem implies $\sum_{|S| \leq \alpha(\epsilon, n)} \hat{f}(S)^2 \geq 1 - \epsilon$ by Parseval's theorem.

Claim 3 *For any set S such that $|S| \leq d$, we have $|c_S - \hat{f}(S)| \leq \gamma$ for $\gamma \leftarrow \sqrt{\frac{\tau}{n^d}}$ with probability at least $1 - \delta$.*

Proof of Theorem 2:

Assume that our claim holds. (δ probability of error did not happen.)

Define $g(x) \equiv f(x) - h(x)$.

Since Fourier transform is linear, we have $\forall S, \hat{g}(S) = \hat{f}(S) - \hat{h}(S)$.

- If $|S| > d$, then $\hat{h}(s) = 0$, so $\hat{g}(s) = \hat{f}(s)$.
- If $|S| \leq d$, then $\hat{f}(S) = c_S$. So $\hat{g}(s) = \hat{f}(S) - c_S$, and $\hat{g}(s)^2 \leq \gamma^2$.

Therefore,

$$\begin{aligned}
E[(f(x) - h(x))^2] &= E[g(x)^2] \\
&= \sum_S \hat{g}(S)^2 (\text{by Parseval's theorem}) \\
&= \sum_{|S| \le d} \hat{g}(S)^2 + \sum_{|S|>d} \hat{g}(S)^2 \\
&\le n^d \gamma^2 + \epsilon \\
&\le \tau + \epsilon.
\end{aligned}$$

■

Theorem 4 *For a function* $f : \{\pm 1\}^n \to \{\pm 1\}$ *and* $h : \{\pm 1\}^n \to \mathbb{R}$*, we have* $Pr[f(x) \neq sign(h(x))] \le E[(f(x) - h(x))^2]$.

Proof

Observe that

$$E[(f(x) - h(x))^2] = \frac{1}{2^n} \sum_x (f(x) - h(x))^2$$

and

$$Pr[f(x) \neq sign(h(x))] = \frac{1}{2^n} \sum_x 1_{f(x) \neq sign(h(x))}.$$

Consider each term in the summation. We know that for each x,

- If $f(x) = sign(h(x))$, then $(f(x) - h(x))^2 \ge 0 = 1_{f(x) \neq sign(h(x))}$.
- If $f(x) \neq sign(h(x))$, then $f(x)$ and $h(x)$ differs by at least 1, so $(f(x) - h(x))^2 \ge 1 = 1_{f(x) \neq sign(h(x))}$.

This completes our proof.

■

Therefore, we can run with $\tau = \epsilon$ and get h such that $E[(f(x) - h(x))^2] \le \epsilon + \epsilon = 2\epsilon$.

5 Applications

Application 1: Consider the bounded depth decision tree. By linearity, any $\hat{f}_l(S)$ is 0 for all S such that $|S| >$ depth.

Application 2: We can compute any n-bit function in constant depth circuit. However, we cannot compute parity of n bits.

Application 3: Sample query algorithm:

Theorem 5 *For any function f computable via size s and depth d circuits:*

$$\sum_{|S|>t} \hat{f}(S)^2 \leq \alpha,$$

for $t = O(\log \frac{2s}{\alpha})^{d-1}$.

We take $s = poly(n), d =$ constant, and $\alpha = O(\epsilon)$, which gives $n^{O(\log^d(\frac{n}{\epsilon}))}$ sample query algorithm.