Analysis of Boolean Functions	**(CMU 18-859S, Spring 2007)**

Lecture 26: Influences and Decision Trees

Apr. 24, 2007

Lecturer: Ryan O'Donnell — *Scribe: Ryan O'Donnell*

1 Main Theorem

In this lecture, we will show an inequality relating *decision tree complexity* and *influences*. We work in the setting of the p-biased product distribution on $\{-1, 1\}$. Recall:

Fact 1.1 *Let $f : \{-1, 1\}^n_{(p)} \to \{-1, 1\}$. Then*

$$\mathbf{Var}[f] = \mathbf{E}[f^2] - \mathbf{E}[f]^2 = 4\mathbf{Pr}[f = -1]\mathbf{Pr}[f = 1] = 2 \mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y} \text{ indep.}}[f(\boldsymbol{x}) \neq f(\boldsymbol{y})],$$

which is 1 *if* f *is "balanced" under the* p*-biased distribution. Also,*

$$\mathrm{Inf}_i(f) = \mathop{\mathbf{E}}_{\boldsymbol{x}}[\mathop{\mathbf{Var}}_{\boldsymbol{x}_i}[f]] = 2 \mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{x}^{(\sim i)}}[f(\boldsymbol{x}) \neq f(\boldsymbol{x}^{(\sim i)})],$$

where $\boldsymbol{x}^{(\sim i)}$ denotes $\boldsymbol{x}$ with the ith coordinate rerandomized *(according to the p-biased distribution).*

Since we've been considering random inputs throughout the course, let's see what this means for decisions trees.

Observation 1.2 *("The Decision Tree Observation") Let T be a deterministic decision tree (henceforth DDT). The following method constructs a random input $\boldsymbol{x}$ distributed according to $\{-1, 1\}^n_{(p)}$:*

1. Start at the root of T; say it queries coordinate i_1. Choose $\boldsymbol{x}_{i_1} \in \{-1, 1\}_{(p)}$, and follow the branch according to this choice.

2. Suppose one is now at a node labeled i_2. Choose $\boldsymbol{x}_{i_2} \in \{-1, 1\}_{(p)}$, and follow the branch according to this choice.

3. Repeat, until one comes to a leaf. At this point, some $\boldsymbol{x}_{i_1}, \dots, \boldsymbol{x}_{i_t}$ have been fixed. Now choose values independently and randomly from $\{-1, 1\}_{(p)}$ for all unfixed coordinates.

NB: As always, we assume that DDTs never query the same coordinate more than once on any path.

Definition 1.3 *Let T be a DDT. We define:*

$$\delta_i^{(p)}(T) = \mathop{\mathbf{Pr}}_{p\text{-biased}}[T \text{ queries } i\text{th coord.}],$$

$$\Delta^{(p)}(T) = \sum_{i=1}^{n} \delta_i = \mathop{\mathbf{E}}_{p\text{-biased}}[\#\text{ of coords queried}] = \mathop{\mathbf{E}}_{p\text{-biased}}[\text{depth of path } T \text{ follows}].$$

The following is a nice exercise:

Proposition 1.4 $\Delta^{(p)}(T) \leq (\log_2 \text{size}(T))/H(p)$*, where* $H(p)$ *denotes the binary entropy of* p *(which is* 1 *if* $p = 1/2$*).*

Definition 1.5 *The* p-biased average-case DT complexity of f *is*

$$\Delta^{(p)}(f) = \min\{\Delta^{(p)}(T) : T \textit{ is a DDT computing } f\}.$$

Note that

$$\Delta^{(p)}(f) \leq R(f) \leq D(f),$$

where

$$D(f) = \min\{\textit{depth(T)} : T \textit{ is a DDT computing } f\},$$

$$R(f) = \min\{\textit{cost}(\mathcal{T}) : T \textit{ is an RDT computing } f\}.$$

Here an RDT (randomized decision tree) $\mathcal{T}$ *computing* f *is a probability distribution over DDTs computing* f *(i.e., a "zero-error" randomized DT), and*

$$\textit{cost}(\mathcal{T}) = \max_{x \in \{-1,1\}^n} \operatorname*{avg}_{\mathcal{T}\textit{'s randomness}} [\textit{\# coords queried}].$$

The main theorem for this lecture is:

Theorem 1.6 *Let* $f : \{-1,1\}^n_{(p)} \to \{-1,1\}$ *and let* T *be a DDT computing* f*. Then*

$$\mathbf{Var}[f] \leq \sum_{i=1}^{n} \delta_i^{(p)}(T) \cdot \text{Inf}_i(f). \tag{1}$$

In words: "The expected sum of influences experienced along a random path is at least the variance."

2 Interpretations

1. Functions with efficient decision trees have influential variables. We have

$$\sum_{i=1}^{n} \delta_i^{(p)}(T) \cdot \text{Inf}_i(f) \leq \left(\max_i \text{Inf}_i(f)\right) \cdot \sum_{i=1}^{n} \delta_i^{(p)}(T) = \left(\max_i \text{Inf}_i(f)\right) \cdot \Delta^{(p)}(T).$$

Hence:

Corollary 2.1

$$\exists i \textit{ s.t. } \text{Inf}_i(f) \geq \mathbf{Var}[f]/\Delta^{(p)}(f) \quad (\geq \mathbf{Var}[f]/R(f) \geq \mathbf{Var}[f]/D(f)). \tag{2}$$

E.g., if f *is balanced and has a DDT of depth* d*, then there exists* i *with* $\text{Inf}_i(f) \geq 1/d$*.*

In particular, (2) is better than KKL for any function f with average-case DT complexity $o(\frac{n}{\log n})$.

This interpretation may be of interest for learning theory. Many popular, practical machine learning algorithms ("CART", "C4.5") try to build a DT hypothesis as follows: (a) Identify a "very relevant" or "very influential" variable. (b) Put this at the root of a DDT. (c) Recurse on the two possible restrictions. There isn't a lot of theoretical justification for this, and indeed most PAC-style learning algorithms for DTs don't do this. This result at least shows that the idea is not completely broken: *If* there is, say, a depth-d DDT computing the function f, then there will at least *exist* some variable with influence at least $\mathbf{Var}[f]/d$.

This interpretation is also of interest for the study of threshold phenomena:

Corollary 2.2 *Let $f : \{-1,1\}^n \to \{-1,1\}$ be any nonconstant transitive (weakly symmetric) monotone function (e.g., a monotone graph property). Let p_c be the critical probability for f. Then:*

$$\mathbb{I}^{(p_c)}(f) \geq n/\Delta^{(p_c)}(f);$$

hence f has a sharp threshold if its p_c-biased average DT complexity is $o(n)$.

Proof: By definition, $\mathbf{Var}[f] = 1$ at the critical probability; also, since f is transitive all its influences are the same, $\mathbb{I}^{(p_c)}(f)/n$. $\square$

2. Functions with all influence small require complex decision trees. There is a lot of work in complexity theory on proving lower bounds for randomized decision trees. We will talk about this later in Section 5.

3 Proof of Theorem 1.6

Actually, the proof requires no Fourier analysis! It only requires probabilistic reasoning.

Let $f : \{-1,1\}^n_{(p)} \to \{-1,1\}$, and let T be a DDT for f.

Let $\boldsymbol{x}$, $\boldsymbol{y}$ be independent random inputs. Think of $\boldsymbol{x}$ as being chosen via The Decision Tree Observation, but think of $\boldsymbol{y}$ as just a bank of random p-biased bits.

Let $\boldsymbol{i}_1, \ldots, \boldsymbol{i_d}$ be the coordinates T queries on $\boldsymbol{x}$, in order. Note that here $\boldsymbol{d}$ is also a random variable. For all $j > \boldsymbol{d}$, define $\boldsymbol{i}_j = \perp$.

For $0 \leq t \leq d$, define the hybrid input $\boldsymbol{z}_t$ to be the input that is mostly $\boldsymbol{y}$, except that coordinates $\boldsymbol{i}_{t+1}, \ldots, \boldsymbol{i_d}$ have $\boldsymbol{x}$'s values substituted in.

We have that $\boldsymbol{z}_0$ is the string that agrees with $\boldsymbol{x}$ on the bits in the path T follows on $\boldsymbol{x}$, but agrees with $\boldsymbol{y}$ on the remaining bits "chosen after T completes its path on $\boldsymbol{x}$". We have that $f(\boldsymbol{z}_0) = f(\boldsymbol{x})$, since T computes f.

Also, we have $\boldsymbol{z_d} = \boldsymbol{y}$, and hence $f(\boldsymbol{z_d}) = f(\boldsymbol{y})$.

Thus:

$$\begin{aligned}
\mathbf{Var}[f] = 2\Pr_{\boldsymbol{x},\boldsymbol{y}}[f(\boldsymbol{x}) \neq f(\boldsymbol{y})] &= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}}[|f(\boldsymbol{x}) - f(\boldsymbol{y})|] \\
&= \mathbf{E}[|f(\boldsymbol{z}_0) - f(\boldsymbol{z}_{\boldsymbol{d}})|] \\
&\leq \mathbf{E}\Big[\sum_{t\geq 1} |f(\boldsymbol{z}_{t-1}) - f(\boldsymbol{z}_t)|\Big] \quad \text{(for } t \geq \boldsymbol{d}\text{, the summand is 0)} \\
&= \sum_{t\geq 1} \mathbf{E}[|f(\boldsymbol{z}_{t-1}) - f(\boldsymbol{z}_t)|\Big].
\end{aligned}$$

For each t, we condition on the value of $\boldsymbol{i}_t$. This can be one of $n+1$ values: $1, 2, \ldots, n, \perp$. However,

$$\boldsymbol{i}_t = \perp \quad \Rightarrow \quad t > \boldsymbol{d} \quad \Rightarrow \quad \boldsymbol{z}_{t-1} = \boldsymbol{z}_t = \boldsymbol{y} \quad \Rightarrow \quad |f(\boldsymbol{z}_{t-1}) - f(\boldsymbol{z}_t)| = 0.$$

Thus we may disregard the $\boldsymbol{i}_t = \perp$ possibility and write

$$\sum_{t\geq 1} \mathbf{E}[|f(\boldsymbol{z}_{t-1}) - f(\boldsymbol{z}_t)|\Big] = \sum_{t\geq 1}\sum_{j=1}^{n} \Pr[\boldsymbol{i}_t = j]\, \mathbf{E}[|f(\boldsymbol{z}_{t-1}) - f(\boldsymbol{z}_t)| \;\Big|\; \boldsymbol{i}_t = j].$$

We now come to the only subtle point in the proof:

Claim 3.1 *Fix $t \geq 1$ and $j \in [n]$. Conditioned on $\boldsymbol{i}_t = j$, the distribution $(\boldsymbol{z}_{t-1}, \boldsymbol{z}_t)$ is the same as the distribution $(\boldsymbol{w}, \boldsymbol{w}^{(\sim i)})$, where $\boldsymbol{w}$ is random and $\boldsymbol{w}^{(\sim i)}$ is $\boldsymbol{w}$ with the jth coordinate rerandomized.*

Proof: Certainly conditioning $\boldsymbol{i}_t = j$ imposes constraints on $\boldsymbol{x}_{\boldsymbol{i}_1}, \ldots, \boldsymbol{x}_{\boldsymbol{i}_{t-1}}$. But all values $\boldsymbol{x}_{\boldsymbol{i}_t}, \ldots, \boldsymbol{x}_{\boldsymbol{i}_{\boldsymbol{d}}}$ are independent of these. And in $\boldsymbol{z}_{t-1}$, we have completely independent random bits for the non-$\boldsymbol{i}$ variables, and we also have completely independent random bits for coordinates $\boldsymbol{i}_1, \ldots \boldsymbol{i}_{t-1}$. Hence $\boldsymbol{z}_{t-1}$ is just distributed like a totally random string $\boldsymbol{w}$. And then $\boldsymbol{z}_t$ is formed just by rerandomizing the $\boldsymbol{i}_t$ coordinate; i.e., the j coordinate. $\square$

We now conclude:

$$\begin{aligned}
& \sum_{t\geq 1}\sum_{j=1}^{n}\mathbf{Pr}[\boldsymbol{i}_t=j]\,\mathbf{E}\Big[|f(\boldsymbol{z}_{t-1})-f(\boldsymbol{z}_t)|\;\Big|\;\boldsymbol{i}_t=j\Big] \\
= & \sum_{t\geq 1}\sum_{j=1}^{n}\mathbf{Pr}[\boldsymbol{i}_t=j]\,\mathbf{E}[|f(\boldsymbol{w})-f(\boldsymbol{w}^{(\sim i)})|] \\
= & \sum_{t\geq 1}\sum_{j=1}^{n}\mathbf{Pr}[\boldsymbol{i}_t=j]\cdot 2\mathbf{Pr}[f(\boldsymbol{w})\neq f(\boldsymbol{w}^{(\sim i)})] \\
= & \sum_{j=1}^{n}\sum_{t\geq 1}\mathbf{Pr}[\boldsymbol{i}_t=j]\cdot \mathrm{Inf}_j(f) \\
= & \sum_{j=1}^{n}\mathrm{Inf}_j(f)\cdot\sum_{t\geq 1}\mathbf{Pr}[\boldsymbol{i}_t=j] \\
= & \sum_{j=1}^{n}\mathrm{Inf}_j(f)\delta_j^{(p)}(f) \qquad \square
\end{aligned}$$

4 Tightness

The inequality can often be tight. To see some cases, note first that the entire proof has equalities, except at one point:

$$|f(\boldsymbol{z}_0)-f(\boldsymbol{z}_{\boldsymbol{d}})|\leq\sum_{t\geq 1}|f(\boldsymbol{z}_{t-1})-f(\boldsymbol{z}_t)|.$$

One case in which this inequality is tight is if the tree T is **read-once**. This means that every coordinate in every node in T is different. In this case, consider the smallest t for which $f(\boldsymbol{z}_{t-1})\neq f(\boldsymbol{z}_t)$. First, this means that $\boldsymbol{y}_{\boldsymbol{i}_t}\neq\boldsymbol{x}_{\boldsymbol{i}_t}$. But now since T is read-once, further changing the values on coordinates $\boldsymbol{i}_{t+1},\dots,\boldsymbol{i}_{\boldsymbol{d}}$ won't change the value of f, because these coordinates are not queried on the newly followed path.

Examples of read-once T's include the natural DDTs for AND, OR, and

$$\mathrm{SEL}(x_1,x_2,x_3)=\begin{cases}x_2 & \text{if } \boldsymbol{x}_1=-1,\\ x_3 & \text{if } x_1=1.\end{cases}$$

E.g., for SEL the main inequality reads $1\leq 1\cdot\frac{1}{2}+\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot\frac{1}{2}$.

Once can also check that the inequality becomes tight for **recursively read-once DDTs**. Without making a formal definition, suppose that f and g have read-once DDT. Then there is a natural DDT for $f\otimes g$, which is *not* (in general) read-once, but which we call recursively read-once. Then the inequality becomes tight for $f\otimes g$ with that tree.

For example, the equality is tight for the function Tribes, with any one of the "natural" DDTs computing it.

5 Randomized Decision Tree lower bounds

On Homework #4 (Problem #6) we saw an upper bound on the sum of degree-1 Fourier coefficients in terms of decision tree complexity:

Theorem 5.1 *Let $f : \{-1,1\}^n \to \{-1,1\}$ be computed by a depth-d DDT. Then $\sum_{i=1}^n \hat{f}(i) \leq (\sqrt{\frac{2}{\pi}} + o(1))\sqrt{d}$.*

A nice exercise is to improve this to depend on $\Delta(f)$ rather than $D(f)$ (hint: use Cauchy-Schwarz on $\mathbf{E}_{\boldsymbol{P}}[f(\boldsymbol{P}) \cdot (\sum_{i \in \boldsymbol{I}} x_i^{\boldsymbol{I}})]$):

Theorem 5.2 *Let $f : \{-1,1\}^n \to \{-1,1\}$. Then $\sum_{i=1}^n \tilde{f}(i) \leq \sqrt{\Delta(f)}$.*

This is easily generalized to the p-biased case:

Theorem 5.3 *Let $f : \{-1,1\}^n_{(p)} \to \{-1,1\}$. Then $\sum_{i=1}^n \tilde{f}(i) \leq \sqrt{\Delta^{(p)}(f)}$.*

In particular, if f is monotone, then we know that $\tilde{f}(i) = \mathrm{Inf}_i(f)/(2\sqrt{pq})$. Hence:

Corollary 5.4 *Let $f : \{-1,1\}^n_{(p)} \to \{-1,1\}$ be a monotone function. Then $\mathbb{I}^{(p)}(f) \leq 2\sqrt{pq}\sqrt{\Delta^{(p)}(f)}$.*

But we can now combine this with Corollary 2.2:

$$n/\Delta^{(p_c)}(f) \leq \mathbb{I}^{(p_c)}(f) \leq 2\sqrt{pq}\sqrt{\Delta^{(p_c)}(f)}$$

and we get

Theorem 5.5 *Let $f : \{-1,1\}^n \to \{-1,1\}$ be any nonconstant transitive (weakly symmetric) monotone function. Let p_c be the critical probability for f. Then:*

$$\Delta^{(p_c)}(f) \geq \frac{n^{2/3}}{(4pq)^{1/3}}.$$

This is known to be essentially best possible:

Theorem 5.6 (Benjamini-Schramm-Wilson '05) *There is a $\frac{1}{2}$-critical monotone transitive f with $\Delta(f) \leq O(n^{2/3} \log n)$.*

When f is a monotone graph property on v vertices, the situation is very interesting. First:

Conjecture 5.7 *(Aanderaa-Karp-Rosenberg Conjecture '73) If f is a monotone graph property on v vertices, then $D(f) = \binom{v}{2}$.*

Results:
$\geq v^2/16$, by Rivest-Vuillemin-'75.
$\geq v^2/9$, by Kleitman-Kwiatowski-'80.
$\geq \binom{v}{2}/2$ and $= \binom{v}{2}$ if v is a prime power, by Kahn-Saks-Sturtevant-'84 (uses topology and group theory!)
$= n$ in the bipartite case, by Yao-'88.

Conjecture 5.8 *(Yao Conjecture '77) If f is a monotone graph property on v vertices, then $R(f) \geq \Omega(v^2)$.*

Results:

$\geq \Omega(v)$, by Yao-'77.

$\geq \Omega(v \log^{1/12} v)$, by Yao-'87 using "graph packing".

$\geq \Omega(v^{5/4})$, by King-'88 using more elaborate graph packing.

$\geq \Omega(v^{4/3}) = \Omega(n^{2/3})$, by Hajnal-'91 using more elaborate graph packing.

$\geq \Omega(v^{4/3} \log^{1/3} v)$, by Chakrabarti-Khot-'01 using more elaborate graph packing.

$\geq \min\{\Omega(v/pq), \Omega(v^2/\log v)\}$ by Friedgut-Wigderson-'02 using less elaborate graph packing and more probabilistic reasoning.

$\geq \Omega(v^{4/3}/(pq)^{1/3})$ by our results from today, using *no* graph packing!

The last three results are all incomparable.

It is very strange that all of the graph packing arguments get stuck at roughly the same point: $n^{2/3}$ — the very point that you *cannot* beat if you only have a transitive function and not a graph property.