CS5160: Topics in Computing
Problem Set 1

Taha Adeel Mohammed

CS20BTECH11052

(Crediting the course)

(1) Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function, and k be the largest natural number such that $|f^{-1}(1)|$ is divisible by 2^k. Show that $D(f) \geq n-k$, where $D(f)$ is the decision tree complexity of f.

Proof by contradiction:

Assume that $D(f) < n-k$, i.e. $D(f) \leq n-k-1$. Then, there exists a decision tree T of depth atmost $n-k-1$ that computes f.
Let P be a path from the root to a leaf node with value 1 in T. Then the path P would query atmost $n-k-1$ bits. This implies atleast $k+1$ bits in the input can be varied while still following the same path. Hence, there exists 2^{k+1+t} inputs, where $t \geq 0$, that follow the same path P.
To get $|f^{-1}(1)|$, we sum the number of inputs for all paths P that lead to a leaf node with value 1. Hence,

$$\begin{aligned}
|f^{-1}(1)| &= \sum_P 2^{k+1+t} \\
&= 2^{k+1} \sum_P 2^t \\
\implies |f^{-1}(1)| &= 2^{k+1} \cdot q, \text{ where } q \in \mathbb{N}
\end{aligned}$$

This implies that $|f^{-1}(1)|$ is divisible by 2^{k+1}, which contradicts the fact that k is the largest natural number such that $|f^{-1}(1)|$ is divisible by 2^k. Hence, our assumption that $D(f) < n-k$ is false, and hence

$$\boxed{D(f) \geq n-k}$$

. $\square$

(2) For every k, define $\mathrm{f_k}$ to be the following function taking inputs of length $\mathrm{n = 2^k}$:

$$f_k(x_1, \ldots, x_n) = \begin{cases} f_{k-1}(x_1, \ldots, x_{2^{k-1}}) \wedge f_{k-1}(x_{2^{k-1}+1}, \ldots, x_n) & \textbf{if } k > 0 \textbf{ is even} \\ f_{k-1}(x_1, \ldots, x_{2^{k-1}}) \vee f_{k-1}(x_{2^{k-1}+1}, \ldots, x_n) & \textbf{if } k \textbf{ is odd} \\ x_i & \textbf{if } k = 0 \end{cases}$$

Show that $D(f_k) = 2^k$.

Proof by induction:

Induction Hypothesis: For $k \geq 0$, $\exists$ distinct inputs x, y s.t for $2^k - 1$ bits, $x_i = y_i$, and $f_k(x) = 0$, $f_k(y) = 1$ and $D(f, x) = D(f, y) = 2^k$. This would also imply that $D(f_k) \geq 2^k$, and since $D(f_k) \leq n = 2^k$, we have $D(f_k) = 2^k$.

Base case: For $k = 0$, we have $x = 0$ and $y = 1$ s.t $f_0(x) = 0$, $f_0(y) = 1$, and $D(f_0, x) = D(f_0, y) = 1$. Hence, the base case holds.

Induction step: Assume that the induction hypothesis holds for $k = m$. i.e. there exists x_m and y_m s.t for $2^m - 1$ bits, $x_{mi} = y_{mi}$, and $f_m(x_m) = 0$, $f_m(y_m) = 1$ and $D(f_m, x_m) = D(f_m, y_m) = 2^m$. We need to show that it holds for $k = m + 1$.

Case 1: $m + 1$ is even

This implies that $f_m(x_1, \ldots, x_n) = f_m(x_1, \ldots, x_{n/2}) \wedge f_m(x_{n/2+1}, \ldots, x_n)$. Here we would have

$$\begin{aligned} x_{m+1} &= y_m + x_m && (f_{m+1}(x_{m+1}) = f_m(x_m) \wedge f_m(y_m) = 1 \wedge 0 = 0) \\ y_{m+1} &= y_m + y_m && (f_{m+1}(y_{m+1}) = f_m(y_m) \wedge f_m(y_m) = 1 \wedge 1 = 1) \end{aligned}$$

Clearly $x_{m+1} \neq y_{m+1}$, and for $2^{m+1}-1$ bits, $x_{(m+1)i} = y_{(m+1)i}$. Also, we have $f_{m+1}(x_{m+1}) = 0$, $f_{m+1}(y_{m+1}) = 1$, and $D(f_{m+1}, x_{m+1}) = D(f_{m+1}, y_{m+1}) = 2^{m+1}$. Hence, the induction step holds for this case.

Case 2: $m + 1$ is odd

This implies that $f_m(x_1, \ldots, x_n) = f_m(x_1, \ldots, x_{n/2}) \vee f_m(x_{n/2+1}, \ldots, x_n)$. Here we would have

$$\begin{aligned} x_{m+1} &= x_m + x_m && (f_{m+1}(x_{m+1}) = f_m(x_m) \vee f_m(y_m) = 0 \vee 0 = 0) \\ y_{m+1} &= x_m + y_m && (f_{m+1}(y_{m+1}) = f_m(x_m) \vee f_m(y_m) = 0 \vee 1 = 1) \end{aligned}$$

Clearly $x_{m+1} \neq y_{m+1}$, and for $2^{m+1}-1$ bits, $x_{(m+1)i} = y_{(m+1)i}$. Also, we have $f_{m+1}(x_{m+1}) = 0$, $f_{m+1}(y_{m+1}) = 1$, and $D(f_{m+1}, x_{m+1}) = D(f_{m+1}, y_{m+1}) = 2^{m+1}$. Hence, the induction step also holds for this case.

Hence the induction step holds, and hence the induction hypothesis holds for all $k \geq 0$. Therefore we have

$$\boxed{D(f_k) = 2^k}$$

□

(3) Show that for a non-constant symmetric function $f : \{0, 1\}^n \to \{0, 1\}$,

$$\boldsymbol{s(f) \geq \left\lceil \frac{n+1}{2} \right\rceil,}$$

where $s(f)$ is the sensitivity of f. Also, give an example where this bound is tight. Recall, we say a function is *symmetric* if its value depends only on the number of 1s in the input. That is, it depends on the hamming weight of the input.

Since the function is symmetric, we have that $f(x)$ only depends on $|x|$, the hamming weight of x. Also, since the function is non-constant, there exist values $k \in [1, n]$ such that for inputs x, y, with $|x| = k - 1$ and $|y| = k$, $f(x) \neq f(y)$. Here, we would have

$$s(f, x) \geq n - (k - 1) \qquad \text{(converting } x \text{ to } y \text{ by flipping the 0s)} \tag{1}$$

$$s(f, y) \geq k \qquad \text{(converting } y \text{ to } x \text{ by flipping the 1s)} \tag{2}$$

Therefore we have,

$$\begin{aligned} s(f) &\geq \max\{s(f,x), s(f,y)\} \\ &\geq \max\{n-(k-1), k\}, \text{ for some } k \in [1,n] \\ \implies & \boxed{s(f) \geq \left\lceil \frac{n+1}{2} \right\rceil} \end{aligned}$$

□

Example: Consider the function $f : \{0,1\}^n \to \{0,1\}$ defined as

$$f(x) = \begin{cases} 1 & \text{if } |x| \geq \left\lceil \frac{n+1}{2} \right\rceil = k \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Then for x s.t $|x| < k-1$ or $|x| > k$, we would have

$$s(f,x) = 0.$$

For x s.t $|x| = k-1$, we would have

$$\begin{aligned} s(f,x) &= n-(k-1) \\ &= n - \left\lceil \frac{n+1}{2} \right\rceil + 1 \\ &= \left\lceil \frac{n+1}{2} \right\rceil \end{aligned}$$

Similarly for x s.t $|x| = k$, we would have

$$s(f,x) = k = \left\lceil \frac{n+1}{2} \right\rceil$$

Hence, we have that $s(f) = \left\lceil \frac{n+1}{2} \right\rceil$. Therefore, the bound is tight.

□

(4) Show that $s(f) = bs(f) = C(f)$ for every monotone Boolean function f, where $bs(f)$ and $C(f)$ respectively denote the block sensitivity and certificate complexity of f. Recall we say a function is monotone if for any x, y such that $x \leq y$, that is, $x_i \leq y_i$ for all $i \in [n]$, $f(x) \leq f(y)$.

In class we have seen that for all $x \in \{0,1\}^n$,

$$s(f,x) \leq bs(f,x) \leq C(f,x) \leq D(f,x) \tag{4}$$

This implies that

$$s(f) \leq bs(f) \leq C(f)$$

Hence, it is sufficient to show that $s(f) = C(f)$.

(5) Prove that when f is a monotone, $s(f) \leq \deg(f)$.

(6) Define *average sensitivity*, $as(f)$, of a Boolean function f to be the expected sensitivity of f on a random input: $as(f) = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} s(f,x)$. Let T be a decision tree and let ℓ_x be the length of the unique path in T consistent with x. Define *average depth* of T to be $\frac{1}{2^n} \sum_{x \in \{0,1\}^n} \ell_x$. Show that the average depth of any decision tree for f is at least $as(f)$.

In class, we have seen the following results for all $x \in \{0,1\}^n$:

$$C(f,x) \leq D(f,x), \tag{5}$$
$$s(f,x) \leq bs(f,x) \leq C(f,x) \leq D(f), \tag{6}$$

where C is the certificate complexity, D is the decision tree complexity, s is the sensitivity, and bs is the block sensitivity.

Combining these results, we have

$$s(f,x) \leq D(f,x) = l_x, \ \forall \ x \in \{0,1\}^n \tag{7}$$

Hence, we have

$$\begin{aligned} \sum_{x \in \{0,1\}^n} s(f,x) &\leq \sum_{x \in \{0,1\}^n} l_x \\ \implies \frac{1}{2^n} \cdot as(f) &\leq \frac{1}{2^n} \sum_{x \in \{0,1\}^n} l_x \\ \implies \qquad as(f) &\leq \textit{average depth of } T \end{aligned}$$

Therefore the average depth of any decision tree for f is at least $as(f)$.

□