Exercise VIII, Computational Complexity 2025

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. Solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

Decision trees

1 Define SINK: $\{0,1\}^{\binom{n}{2}} \to \{0,1\}$ by interpreting the input $x \in \{0,1\}^{\binom{n}{2}}$ as a labelling of the edges of the complete graph on n vertices such that the bit $x_e \in \{0,1\}$ for edge $e = \{u,v\}$ defines an *orientation* of e, either (u,v) or (v,u). Thus, an input x defines a directed graph $G_x = (V, E_x)$. We define $\text{SINK}(x) = 1$ iff the graph G_x has a *sink*, that is, some node $v^* \in V$ such that $(u, v^*) \in E_x$ for all $u \neq v^*$. Show that $\text{D}(\text{SINK}) = \Theta(n)$.

Solution: We first show that $\text{D}(\text{SINK}) \leq O(n)$ by describing an algorithm solving SINK using at most $2n-3$ queries. The algorithm starts by looking at the pair of vertices 1 and 2. At least one is not a source and can be ruled out as a solution. Supposing 2 points to 1, we continue with the pair 1,3 and again, at least one is not a sink. Continuing this process using $n-1$ queries, one reach the last vertex that could potentially be a source. Finally, one can decide if this particular vertex is a source or not using $n-2$ further queries for a total of $2n-3$ queries. To argue that $\text{D}(\text{SINK}) \geq \Omega(n)$, we exhibit an adversary strategy that delays the resolution until at least $k = n/2$ queries are made. The adversary answers arbitrarily. Observe that before the k-th query, the function value is still open: one need n queries to certify a sink and at least one vertex was not queried at all, which could be a sink.

2 Suppose $f\colon \{0,1\}^n \to \{0,1\}$ is *monotone*, meaning that if $x \leq y$ (i.e., $x_i \leq y_i$ for all $i \in [n]$), then $f(x) \leq f(y)$. Show that $\text{s}(f) = \text{C}(f)$.

Solution: The direction $\text{C}(f) \leq \text{s}(f)$ is unconditional and was seen in class. For the other direction, let $x \in \{0,1\}^n$ be an input that maximizes $C(f,x)$ and suppose that $f(x) = 1$ (the other case is analogous). Let ρ be a smallest 1-certificate for x. Observe first that ρ cannot have a 0-entry - indeed, suppose toward contradiction that $\rho_i = 0$ for some $i \in [n]$. We argue that ρ' which is a copy of ρ but with $\rho'_i = *$ is also 1-certificate for x (since it is smaller, it is a contradiction with the optimality of ρ). Because x is consistent with ρ, x is consistent with ρ' too. Now, pick any $y \in \{0,1\}^n$ consistent with ρ'. If $y_i = 0$, then y is consistent with ρ and $f(y) = 1$. If $y_i = 1$ then $y \geq y^{(i)}$ and by monotonicity of f it holds that $f(y) = 1$ too (because $y^{(i)}$ is consistent with ρ). Now, let $z \in \{0,1\}^n$ be an input formed as follows:

$$z_i = \begin{cases} 0 & \text{if } \rho_i = * \\ 1 & \text{if } \rho_i = 1 \end{cases}$$

Note that $f(z) = 1$ and we show that z has sensitivity $\text{C}(f)$ by showing that flipping any of its 1-entry makes the function change value - further implying that $\text{s}(f) \geq \text{C}(f)$. Suppose toward contradiction that $z_i = 1$ and $f(z^{(i)}) = 1$. Let ρ' be a copy of ρ where $\rho'_i = \star$. Note that x matches ρ' and that for any y which is matched by ρ', $z^{(i)} \leq y$ so that $f(y) = 1$. This shows that ρ' is a 1-certificate for x of smaller size.

3 In the lecture we saw that if there exists an Adversary strategy for answering $k-1$ many queries to the input variables of a function f such that the value of f remains undetermined, then $\mathrm{D}(f) \geq k$. Prove the converse: if $\mathrm{D}(f) \geq k$, then there exists an Adversary strategy fooling any $k-1$ query algorithm.

Solution: The idea is to construct the adversary recursively. We prove that for any $f : \{0,1\}^n \to \{0,1\}$, there exists an adversary that fools any decision tree with depth $D(f)-1$ by induction on $n \geq 1$. For $n = 1$, there are only 4 different function (identity, negation, constant 0 and constant 1) and the claim holds for each of those. Now, fix any $f : \{0,1\}^n \to \{0,1\}$ and let t be a decision tree t which solves f. If f is constant, then the claim is vacuously true. Else, we can assume that $\mathrm{depth}(t) \geq 1$ and suppose that t first queries x_1. Let $f_0, f_1 : \{0,1\}^{n-1} \to \{0,1\}$ be the function f where x_1 is fixed to be 0 and 1 respectively. By the induction hypothesis, there exists an adversary A_0 that fools any tree that tries to compute f_0 with less than $D(f_0)$ queries and an adversary A_1 that does the same for f_1. Our adversary simply responds to the query x_1 with answer b:

$$b := \operatorname*{argmax}_{b \in \{0,1\}} \mathrm{D}(f_b)$$

and then uses A_b to handle further queries. Now, suppose that $\mathrm{depth}(t) \leq \mathrm{D}(f)-1$. This means that $\mathrm{depth}(t_b) \leq \mathrm{D}(f)-2 = \mathrm{D}(f_b)-1$. But since A_b can fool any tree that tries to compute f_b with $< \mathrm{D}(f_b)$ queries, it must be that A_b fools t_b so that t is incorrect.

4 A DNF formula $F = T_1 \vee \cdots \vee T_m$ is said to be *unambiguous* if for any input x, at most one of the terms T_i evaluate to true, $T_i(x) = 1$. Define $\mathrm{UC}_1(f)$ as the least k such that f can be written as an unambiguous k-DNF. Prove that $\mathrm{D}(f) \leq \mathrm{UC}_1(f)^2$.
(*Hint: Use a similar idea as in the proof that* $\mathrm{D}(f) \leq \mathrm{C}(f)^2$.)

Solution: Let $F = T_1 \vee \cdots \vee T_m$ be an unambiguous formula representing f with the least degree $k := \mathrm{UC}_1(f)$. We build a decision tree t for f with query cost $\leq k^2$. Let $\mathcal{T}$ be the set of terms in F, where a term is represented as a set of literals. Note that any two $s_1, s_2 \in \mathcal{T}$ has at least one literal in common else F is ambiguous. The algorithm picks any $s \in \mathcal{T}$ and query all the variables of s. If this is already enough to get that the value is 1, we're done. Else, one can strip-off from $\mathcal{T}$ anything that is inconsistent with the bits that have been queried. Each $s \in \mathcal{T}$ also gets its size reduced by at least one. The resulting $\mathcal{T}$ is still such that each pair of sets has a common literal. This process is continued until the value of the function is found or $\mathcal{T}$ is empty (in which case the function value is 0). Note that at each round, the size of the terms in $\mathcal{T}$ decreases by at least one so that the number of queries is at most:

$$k + (k-1) + (k-2) + \cdots + 1 \leq O(k^2)$$

Definition 0.1 *For $x \in \{0,1\}^n$ and $S \subseteq [n]$, let x^S denote the point in $\{0,1\}^n$ obtained by flipping all of the bits x_i such that $i \in S$.*

The sensitivity of $f\colon \{0,1\}^n \to \{0,1\}$ at x, denoted $s_f(x)$, is the number of points y that differ from x in exactly one bit and satisfy $f(x) \neq f(y)$. The sensitivity $\mathrm{s}(f)$ of f is the maximum over all $x \in \{0,1\}^n$ of $\mathrm{s}(f,x)$.

The block sensitivity of $f\colon \{0,1\}^n \to \{0,1\}$ at x, denoted $\mathrm{bs}(f,x)$, is the maximum number of disjoint subsets of the coordinates $B_1, B_2, \ldots, B_k \subseteq [n]$ such that $f(x) \neq f(x^{B_j})$ for all $j \in [k]$. The block sensitivity $\mathrm{bs}(f)$ is the maximum value of $\mathrm{bs}(f,x)$ over all $x \in \{0,1\}^n$.

5 *Sensitivity vs. block-sensitivity.* Say that an n-bit string x is *paired* if there exists an $i \in [n-1]$ such that $x_i = x_{i+1} = 1$ and $x_j = 0$ for all other $j \in [n] \setminus \{i, i+1\}$. Define a boolean

function $f\colon \{0,1\}^{n^2} \to \{0,1\}$ by interpreting the input $x \in \{0,1\}^{n^2}$ as an n-by-n boolean matrix and setting $f(x) = 1$ iff there exists a row of x that is paired. Prove that f has sensitivity $\mathrm{s}(f) \leq O(n)$ but quadratically larger block-sensitivity $\mathrm{bs}(f) \geq \Omega(n^2)$.

Solution: Observe that $\mathrm{bs}(f) \geq n(n-1)$ because for the all-zero input, flipping any two contiguous bits of a row changes the value of the function from 0 to 1. We now show that $\mathrm{s}(f,x) \leq 2n$ for any $x \in \{0,1\}^{n^2}$. As a first case, assume that x has ≥ 2 paired rows. Then flipping one bit only is not enough to change the value of the function from 1 to 0. If x has exactly one paired row, then the only way to make the function change from 1 to 0 is to un-pair that row. Since there are at most n bits in a row, this leaves us with $\mathrm{s}(f,x) \leq n$. Finally, if the input has no paired row, changing the value from 0 to 1 amounts to modifying a row to make it paired. There are two kinds of row that can be changed from un-paired to paired. The first one is a row with a single one, in which case two different flips can make it paired. The second is a row with 3 ones and one must be flipped. This shows that when there are no paired row, $\mathrm{s}(f,x) \leq 3n$.