ASSIGNMENT 3 – COMP 252

Alexandre St-Aubin and Dylan Telio

February 7, 2024

1. DECISION TREE LOWER BOUND.

   (i) Give a simple algorithm for this problem and an upper bound on the complexity. For n = 5, your answer should be 7.

   Solution:

Algorithm 1: An algorithm to classify the *n* monkeys.

Input: An array Monkeys of monkeys.
Output: Three arrays separating the monkey species.

1 Species_1 ← new Array;
2 Species_2 ← new Array;
3 Species_3 ← new Array;
4 $i \leftarrow 1$;
5 Species_1.append(Monkeys[0]);
6 **while** Monkeys[0] = Monkeys[i] **do**
7 | Species_1.append(Monkeys[i]);
8 | $i{+}{+}$;

   `// loop above exits only if a second monkey species was found`
9 Species_2.append(Monkeys[i]);

   `// for each remaining monkey, compare to the 2 known species`
10 **for** *j in range* $[i, n]$ **do**
11 | **if** Monkeys[j] = Species_1[0] **then**
12 | | Species_1.append (Monkeys [j]);
13 | **else if** Monkeys[j] = Species_2[0] **then**
14 | | Species_2.append (Monkeys [j]);
15 | **else**
16 | | Species_3.append (Monkeys [j]);

Remark. The worst case for the above algorithm is when the two first monkeys are different, and each remaining monkey is of the third or second species, hence must be compared twice (once to each of the first two species), in order to be classified. This yields an upper bound complexity of $1 + 2(n - 2)$. When $n = 5$, this is indeed equal to 7.

(ii) Show that all leaves in the decision tree must be different (i.e., correspond to different outputs).

Proof. Fix a sequence of monkeys. The decision tree representing the algorithm described in part (i) is binary, where each decision takes as input two monkeys, and outputs whether they are of the same species or not. Every monkey must be compared at most twice to be classified. We can represent each comparison by a bit, 1 being a match, and 0 otherwise. The output for a given monkey will therefore be:

$$\begin{cases} 10 & \text{if match with Species\_1} \\ 01 & \text{if match with Species\_2} \\ 00 & \text{if match with Species\_3} \end{cases}$$

We note that in reality, if there is a match with Species_1, the monkey will not be compared with the other species, as it would be useless. So, the notation 10 is not entirely representative of what truly happens in the algorithm, it should rather be 1, as only one comparison is made. However, for sake of notation, we keep it as 10.

   Hence, one easily sees that each leaf in the decision tree can be represented as a binary sequence of comparisons, where each 2 consecutive bits represents the classification of a monkey. In order to relax notation, let $10 = 1$, $01 = 2$, $00 = 3$, then each leaf is given as a sequence of the form

$$1233322123123121$$

We show by induction that each leaf represents a unique monkey sequence. The base case is clear when $n = 2$. Now, assume that for $n - 1$ monkeys, each leaf is represented by a unique sequence of the numbers $\{1, 2, 3\}$ of length $n - 1$. Then, if we add a monkey to the end of the array of monkeys, for each existing sequence $\{a_i\}_{i=1}^{n-1}$, 3 sequences of length n will be created:

$$a_{i,1} := \{a_i\}_{i=1}^{n-1} \cup \{a_n = 1\}$$
$$a_{i,2} := \{a_i\}_{i=1}^{n-1} \cup \{a_n = 2\}$$
$$a_{i,3} := \{a_i\}_{i=1}^{n-1} \cup \{a_n = 3\}$$

Clearly the 3 new sequences are unique for each i, and since the $n - 1$ first elements of every sequence are unique by the induction hypothesis, it follows that all of the sequences representing the decision tree for n monkeys are unique. Hence, the leaves represent unique outputs.

$\square$

(iii) Using a decision tree-based method, show that the lower bound for this problem is 6 when $n = 5$.

Proof. The algorithm outputs three sets of monkeys of identical species. For each monkey, there are three possibilities of species, hence 3^5 total permutations. Yet, this accounts for the order of the species, which we don't care about. That is, it shouldn't matter whether the *popa langurs* are grouped in Species_1, Species_2, or Species_3. To remove this unwanted notion of order, we divide by 3!. It follows that the total number of possible outputs of the algorithm is given by

$$\frac{3^n}{3!}$$

By part (ii), each leaf corresponds to a different output, so we have a tree with $\frac{3^n}{3!}$ leaves. Thus, by the method of decision trees, the lower bound is

$$\left\lceil \log_2 \left(\frac{3^5}{3!} \right) \right\rceil = 6$$

$\square$

(iv) For general n, show that the decision tree lower bound is at least $an - b$ and at most $an + b$ for $n \geq n_0$, for some positive numbers a, b, n_0. Determine suitable values for these numbers. Compare this with your answer in *(i)* and conclude that the decision tree bound for this problem is rather weak.

2. ADVERSARIAL LOWER BOUND. Using the method of adversaries, show that the lower bound for this problem is 7 when $n = 5$.

Solution:
ADVERSARY STRATEGY: The adversary always answers NO, unless it has to answer YES.

Claim: This strategy always yields at least 7 comparisons.

Proof. Denote the set of monkeys by $\Omega := \{a, b, c, d, e\}$. There are two cases where the adversary must answer "yes" to a comparison between two monkeys. Without loss of generality, let $a \in$ Species_1 and $x \in \Omega$, then the adversary will be forced to answer yes if,

(i) x was already compared to Species_2 and Species_3, and the adversary answered "no" to both.

(ii) x was already compared to Species_1 and the adversary answered "yes".

Notice that in either case, the algorithm doesn't gain any new information. We may therefore assume that the adversary will never be forced to answer "yes", as any good algorithm would not waste comparisons to obtain information it already knows. $\square$

3. Lower bounds by the method of adversaries, and an algorithm.