Decision tree Topics-

- Randomised Decision tree
- RDT complexity
- Distributional complexity
- Yao’s Min Max Lemma

Randomised Decision tree

What?

A decision tree in which branch taken is determined by two factors:

1. Query value
2. Random choice(coin flip)

OR

We use some probability distribution over deterministic decision trees

Why?

Let us suppose T be a distribution out of which a tree t is chosen for computing f

P(t) = Prob that tree t is chosen

For some input x,

$$c(P,x) = \sum P(t).cost(P,x)$$

Where c(P,x) = expected no of queries a tree chosen from T will make on input x

RDT complexity

- Randomised decision tree complexity R(f) of f is defined as follows :

$$R(f) = \min(c(P,x))$$

Where x is worst case input

- Thus , R(f) expresses how well best probability distribution of decision trees will do against worst possible input.

R(f) >= C(f) .... C : certificate complexity

Why?

- R(f) is an expected value averaged over all values .
- C(f) is the minimum value of complexity
- R(f)>=C(f) because average will never be less than the minimum value.

Example : Majority function

X1	x2	x3
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

F = Maj(x1 , x2 , x3)   i .e. Largest value

To check which is the largest number

We calculate that two need 2 queries with prob ⅓
And   3 queries with prob ⅔

Hence R(f) by definition is = 2*⅓ + 3*⅔ = 8/3

Distributional decision tree complexity

Distributional decision tree complexity is defined by

$$\Delta(f) = \min_D (\max_A d(A,D))$$

Where A - deterministic algorithm A runs over set of decision trees S that are deciders for f

- It measures the expected efficiency of most efficient decision tree algorithm works ,given the worst case distribution of inputs

Yao’s Min Max Lemma

Why?

- Used in variety of settings to prove lowerbounds on randomised algorithms.

Let

X : a finite set of inputs

A : finite set of Algorithms to solve computational problem on these inputs

cost(a,x) : computational cost for algorithm a in A on x in X

Randomised Algorithm:

Lets assume a probability distribution R on A, then cost of R on input x , denoted by cost(R,x) is E[cost(A,x)].

Randomised complexity of the problem is

$$\mathrm{Min}_R \max_x \mathrm{cost}(R,x)$$

Let D be distribution on inputs. For algorithm A , the cost incurred at D,

$$\mathrm{Max}_D \min_A \mathrm{cost}(A,D)$$

By Yao's Lemma ,the above two complexities are same.

- One defines a suitable distribution D on inputs
- One propose that 'Every deterministic algorithm incurs high cost say C on this distribution
- According to Yao's Lemma, randomised complexity then is atleast C.

Thank you

Questions

Assignment questions:

Proof of Yao's lemma.

Solution:

http://ce.sharif.edu/courses/98-99/1/ce685-1/resources/root/Yao%20minmax.pdf

Deep quiz questions : Find R(f) for Maj(x1,x2,x3,x4)

Ans : 3 * 4/24 + 4* 20/24 = 92/24 = 51/12

Light quiz question:

Where in the following cases can yao’s lemma used?

a) to establish a lower bound on the performance of randomized algorithms
b) In two player zero sum game
c) to find a distribution D for which any algorithm in A has high cost.
d) All of the above

Resources

- https://www.cs.purdue.edu/homes/egrigore/Fall12/lect20.pdf
- Arora barak - Decision trees