

October 20,2011

ASSIGNMENT 3

Problem 1. Random Points in an interval ?

Here we shall use a the following approach.

- (a) Consider that the points are first randomly, uniformly and independently permuted, and then added on to the line one point at a time.
- (b) The shortest segment (or any other segment) is defined by the two points (or the start, end).
- (c) We shall find out the probability for any point to be part of the shortest segment using
 - i. Forward Analysis
 - ii. Backward Analysis
- (d) Since we are ultimately finding the probability of the same event we shall equate them, to obtain both upper and lower bounds for the length of the shortest segment.

Let us consider a stage when i points have been already inserted into the line segment. Let P_{i+1} be the $i + 1^{th}$ point to be inserted. Now let ϵ_{i+1} be the event that P_{i+1} is the part of the shortest segment at the end of adding $i + 1$ points to the line segment.

(a) **Backward Analysis**

There are $i + 1$ points in which two points can be a part of the shortest segment, since the points are added uniformly, and independently at random we can see that.

$$Pr[\epsilon_{i+1}] = \frac{2}{i + 1} \quad (1-1)$$

There are exceptions of both the end points, but we can avoid that by transforming the line to a circle, or it can also be seen that since we are only interested in the asymptotic limits, we can see the number of points increase the significance of the end points reduce.

(b) **Forward Analysis**

Let the first i points be already placed on the line, and let X_i be the length of the shortest segment after the placing of i points. Now to find the probability that the $i + 1^{th}$ point P_{i+1} is part of the shortest segment. We can see that to be part of the shortest segment it has to be at a length shorter than X_i from one of the points. We shall now both underestimate and overestimate the probability and use these to arrive at an interval which the value can take.

Consider that from each point we take X_i distance to the left and to the right, we know that if the point P_{i+1} has to be part of the shortest line segment then it has to be in the above said interval.

$$Pr[\epsilon_{i+1}] \leq 2iX_i \quad (1-2)$$

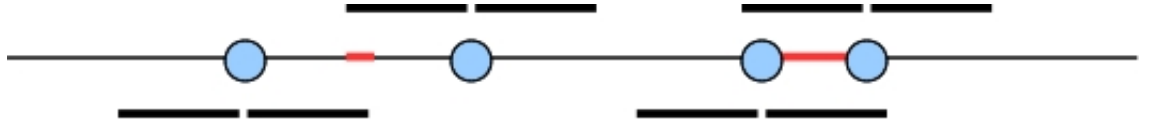


Figure 1: Overestimating the Probability - Overlapping regions shown in Red



Figure 2: Underestimating the Probability - Possible regions not considered shown in Red

The less than factor comes since the lengths can overlap and hence the actual possible length can be lower than the one calculated. This can be seen from Fig:1

Now we consider regions with separation of length $\frac{X_i}{2}$ we can easily see that since $\frac{X_i}{2} < X_i$ if the point is within this interval from the other points it will surely be part of the shortest segment. Since we are considering exactly half the distance there will be NO overlap, but we can see that there are regions which could actually accomodate the point but is not considered by the algorithm, hence this is actually underestimating the probability. Substituting this in the above equation we get.

$$Pr[\epsilon_{i+1}] \geq 2i \frac{X_i}{2} = iX_i \quad (1-3)$$

Figure 2 will make things clear Combining the two equation we get a range

$$iX_i \leq Pr[\epsilon_{i+1}] \leq 2iX_i \quad (1-4)$$

Substituting for $Pr[\epsilon_{i+1}]$ in the equation we get

$$iX_i \leq \frac{2}{i+1} \leq 2iX_i \quad (1-5)$$

Since all numbers are positive hence taking multiplying throughout by $\frac{i}{X_i}$ and taking the inverse we get.

$$i^2 \leq \frac{2i}{(i+1)X_i} \leq 2i^2 \quad (1-6)$$

Taking the inverse we have

$$\frac{1}{2i^2} \leq \frac{(i+1)X_i}{2i} \leq \frac{1}{i^2} \quad (1-7)$$

Since we need to find the asymptotic value we shall consider $i = n$, and we know $\frac{n+1}{n} \rightarrow 1$.

$$\frac{1}{n^2} \leq X_n \leq \frac{2}{n^2} \quad (1-8)$$

Applying expectation we get.

$$\frac{1}{n^2} \leq E[X_n] \leq \frac{2}{n^2} \quad (1-9)$$

Since the other two are already free of any random variables.

Hence not only are we able to prove that the expected length of the shortest segment $= O(n^2)$ but are able to prove a stronger result which says that the expected length of the shortest segment $= \Omega(n^2)$.

→ Answer

Problem 2. Smallest Enclosing Circle

We will select a uniformly random sample $\frac{n}{2}$ out of the n points given, let the set be A and find the smallest possible circle C enclosing these points. Let us have set S be the set of $n/2$ points that were not sampled.

Now for a single point p_i in S consider the set of points. $A \cup p_i$.

We can see that if we find the probability of the point p_i not being in the circle C , then we can easily expand it to the complete set S as follows.

Let us define the event ϵ_i as the even that the point p_i lies outside the circle. We also define the following random variable X_i as follows

$$X_i = \begin{cases} 1 & \text{if } p_i \text{ lies outside the circle} \\ 0 & \text{otherwise} \end{cases} \quad (2-1)$$

Now consider the enclosing circle for the set $A \cup p_i$, we can see that if p_i lies outside the circle C then p_i will be one of the points that define the new circle. We can now can preform a backward analysis on this point ($[\frac{n}{2} + 1]^{th}$). The analysis is as follows:

The point p_i becomes part of the smallest enclosing circle C when it is one of the three points to form a circle. Clearly the remaining two point must be from points within the C to extend C to include the new point.

Just like A can be thought of as any random uniform set we can consider $A \cup p_i$ also as a random uniform set, since we will be considering all the p_i . Each combination fo $A \cup p_i$ is independent of the other p_i 's the C .

Now consider that there exists a enclosing circle for the set $A \cup p_i$. We can see that

$$Pr[p_i \text{ outside the circle}] = Pr[p_i \text{ is one of the defining points of the circle}] \quad (2-2)$$

Since each circle can have atmost 3 defining point if we do a backward analysis we can see that

$$Pr[p_i \text{ is one of the defining points of the circle}] \leq \frac{3}{\frac{n}{2} + 1} \quad (2-3)$$

The \leq comes from the fact that they circle may also have been defined with just two points. We shall now use this to find $E[X_i]$

$$E[X_i] = 1 \cdot Pr[p_i \text{ outside the circle}] + 0 \cdot (1 - Pr[p_i \text{ outside the circle}]) \quad (2-4)$$

$$E[X_i] \leq \frac{3}{\frac{n}{2} + 1} \quad (2-5)$$

We can see that $E[X]$ gives the expected number of points that will lie outside the circle. By linearity of expectation .

$$X_i = \sum_{i=0}^{\frac{n}{2}} X_i \quad (2-6)$$

$$E[X_i] = \sum_{i=0}^{\frac{n}{2}} E[X_i] \quad (2-7)$$

$$\leq \sum_{i=0}^{\frac{n}{2}} \frac{3}{\frac{n}{2} + 1} \quad (2-8)$$

$$\leq 3 \cdot \frac{\frac{n}{2}}{\frac{n}{2} + 1} \quad (2-9)$$

$$\leq 3 \quad (2-10)$$

The answer is surprisingly a constant number, but in hindsight makes perfect sense.

The expected number of unsampled points lie outside circle $C < 3$

→ Answer

Problem 3. Convex Hull problem

(a) Backward analysis

According to the algorithm discussed in class, each point is in some cone of the data structure. Whenever a new point is taken for inserting into the convex hull the cone gets split and thus the number of cone change of point p increases by one. It can be observed that the expected running time of the algorithm is equivalent to the expected number of cone changes caused by insertion of new point $p \in S$ to the Hull.

We know that the expectation of running time of the algorithm is $n \log n$. Let X denote the total number of cone changes during the entire execution of the algorithm. Also X_i denotes the number of cone changes for a point $p \in S$. So we need to prove that

$$Pr[Running Time > cn \log n] \leq \sum_{p \in S} Pr(X_i > cE[X_i]) \quad (3-1)$$

It will be easier to take a microscopic view of the algorithm execution to find the expected number of cone changes. Consider a point $p \in S$ which is to be inserted into hull. Let Y be a random variable that counts the number of cone changes for the point p . Let Y_i be a 0-1 random variable which indicates the contribution of i^{th} iteration of algorithm to Y

$$E[Y_i] = Pr(\text{There is a cone change for } p). \quad (3-2)$$

We will use backward analysis to find out the RHS of the above equation. The point p could have changed the cone if it was part of the cone created by the convex hull points k and $K + 1$ during $(i - 1)^{th}$ iteration and k occurred during i^{th} iteration. Thus the probability that p had changed cone in i^{th} iteration is equal to $\frac{2}{i}$. Thus

$$E[Y_i] = \frac{2}{i}$$

Which gives

$$E[Y] = \sum_{1 \leq i \leq n} E[Y_i]$$

$$= \sum_{1 \leq i \leq n} \frac{2}{i}$$

$$E[Y] = H_n$$

$$E[Y] = \ln n, \text{ since } H_n = \ln n$$

$$E[X_i] = \ln n$$

Now the expected number of cone changes can be obtained as follows.

$$E[X] = \sum_{p \in S} E[X_i]$$

$$E[X] = n * \ln n$$

Now we need to calculate $Pr[(X_i > cE[X_i])]$ which is same as $Pr(Y > cE[Y])$. As Y is sum of independent random variable we can apply Chernoff bound with $c = 1 + \delta$.

$$\begin{aligned} Pr(X_i > cE[X_i]) &\leq \left(\frac{e^{c-1}}{c^c} \right)^{E[X_i]} \\ &\leq \left(\frac{e^{c-1}}{e^{c \ln c}} \right)^{E[X_i]} \\ &\leq \left(e^{-(c \ln c - c + 1)} \right)^{E[X_i]} \\ &\leq e^{-(c \ln c - c + 1) \ln n} \\ &\leq n^{-(c \ln c - c + 1)} \end{aligned}$$

From equation (2-1)

$$\begin{aligned}
Pr[Running\ Time > cn\log n] &\leq \sum_{p \in S} Pr(X_i > cE[X_i]) \\
&\leq n * Pr(X_i > cE[X_i]) \\
&\leq n * n^{-(c \ln c - c + 1)} \\
&\leq n^{-(c \ln c - c)}
\end{aligned}$$

→ Answer

(b) RQsort analysis method

→ Answer

Problem 4. Analysing the duration of a randomized experiment by partitioning it into stages

We shall look at a slight modification of the algorithm, and prove bounds on that can be carried forward to the current algorithm, and then prove the bounds.

Modified Algorithm

- (a) Initialise and set E as all $\binom{n}{2}$ edges, and $V = \{1, \dots, n\}$.
- (b) Select the first edge randomly and add them to the set of connected components.
- (c) Till graph is connected do the following. Select a edge uniformly randomly from the set E without replacement, let it be (u, v) .
 - i. If it creates a cycle add the edge
 - ii. Else discard the edge. (Point of difference with original algorithm)

Cycles in no way will help the connectedness of the graph hence keeping them will be of no use.

Let X_a be the running time of the actual algorithm and X_m be the running time of the modified algorithm.

We can also see that since all $\binom{n}{2}$ edges exists, this method will ultimately reach a solution, and hence completion is also assured.

Now we shall partition the random experiment into different stages, let the i^{th} stage be the stage where $|A| = i$ we can see that the first two stages are reached in time 1. Let us define the random variable X_{mi} as the time spent in the i^{th} step in the modified algorithm. Since once we leave each step we are not coming back to that (since we are only adding elements to A) we can see that. See that if we add $n - 1$ edges and we have no cycles then we have to have a connected component.

$$X_m = X_{m1} + X_{m2} + \dots + X_{mn} \quad (4-1)$$

What we need to find is the expected value of \mathbf{X}_m , which we will denote by $E[\mathbf{X}_m]$ Applying linearity of expectation we arrive at

$$E[\mathbf{X}_m] = E[\mathbf{X}_{m1}] + E[\mathbf{X}_{m2}] + \dots + E[\mathbf{X}_{mn}] \quad (4-2)$$

We already know the initial conditions that

$$E[\mathbf{X}_{m1}] = 0 \quad (4-3)$$

$$E[\mathbf{X}_{m2}] = 1 \quad (4-4)$$

Now consider the i^{th} stage, we shall now consider the probability that it will move in a single step, let us look at the set of edges, out of the total $\binom{n}{2}$ edge the edges between the same components are not useful, considering that we have i edges already added, let the size of a component be denoted by C_i

$$i = \sum_{\forall \text{components } j} C_j \quad (4-5)$$

Considering this the total number of useless components U in the set can be seen as

$$U = \frac{1}{2} \cdot \sum_{\forall \text{components } (i,j)} (C_j)(C_i) \quad (4-6)$$

To maximise this we can set the number of component to be 1, and $C_1 = i$. We are maximising the bad conditions since we need the upperbound, in actual case it will be much less . We shall now look at the probability that an edge selected is useful .

This is nothing but seeing the probability of selecting a useful edge .

$$Pr[Useful] = \frac{\binom{n}{2} - U}{\binom{n}{2} - \text{Edges selected till now}} \quad (4-7)$$

$$Pr[Useful] \geq \frac{\binom{n}{2} - \binom{i}{2}}{\binom{n}{2}} \quad (4-8)$$

We can see that once we get even a single good event we will stop, hence we have been given a random experiment with **Success** probability p and we need to determine the expected number of trials until one success. This is given by $\frac{1}{p}$

$$E[\mathbf{X}_{i+1}] \leq \frac{\binom{n}{2}}{\binom{n}{2} - \binom{i}{2}} \quad (4-9)$$

$$\leq \frac{n^2}{n^2 - i^2} \quad (4-10)$$

Summing this as shown in the formula we have.

$$E[X] = \sum_{i=1}^n n-1 E[X_i] \quad (4-11)$$

$$\leq \sum_{i=1}^n n-1 \frac{n^2}{n^2 - i^2} \quad (4-12)$$

$$\leq n^2 \sum_{i=1}^n \frac{1}{(n+1)(n-1)} \quad (4-13)$$

$$\leq n^2 \sum_{i=1}^n \frac{1}{2n} \left(\frac{1}{n+i} + \frac{1}{n-i} \right) \quad (4-14)$$

$$\leq \frac{n^2}{2n} \sum_{i=1}^n \frac{1}{2n} \left(\frac{1}{n+i} + \frac{1}{n-i} \right) \quad (4-15)$$

$$\leq \frac{n^2}{2n} \left(\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n-1} + \frac{1}{n} + \frac{1}{n+1} + \dots + \frac{1}{2n-1} \right) \quad (4-16)$$

$$\leq \frac{n}{2} H_{2n-1} \quad (4-17)$$

$$\leq \frac{n \log 2n}{2} \quad (4-18)$$

This is the best possible bound that we are able to obtain.

→ Answer

Problem 5. Online Sampling

We shall use the following algorithm to sample, let S be the set of elements we are maintaining, and U be the set of elements we are sampling from.

Algorithm 1 *OnlineSample*(U)

```

for  $i$  from  $1 \dots n$  do
   $S[i] \leftarrow U[i]$ 
end for
Permute  $S[i]$ .
for  $i$  from  $n+1 \dots m$  do
   $RandVar \leftarrow rand(1, i+1)$ 
  if  $RandVar > n$  then
    DiscardApple
  else
    for  $j$  from  $n \dots RandVar$  do
       $S[j] \leftarrow S[j-1]$ 
    end for
     $S[RandVar] \leftarrow U[i]$ 
  end if
end for
Return  $S$ 

```

Explanation

The algorithm does the following, it selects all the first n elements. now for each new element coming after it consider the following situation.

Consider the apples as an ordering and then if we need to insert it, there are exactly $(n + 1)$ position, to do it, and hence we insert it in one of the position and output the first n apples. For each apple we throw away we maintain a "ghost element" in our mind to find out the number of positions the next apple can be inserted into. , and do the same thing.

Since at each point of time, no apple moves backward, an apple discarded is permanent, and hence we need to maintain only the ghost of the apples discarded.



Figure 3: With $n=3$ inserting the 6th apple, the ghost apples which are already discarded are shown in white

Proof of Correctness We give the following proofs for the correctness of the algorithm. The first thing we can see is if we consider all the permutation of the elements seen till then, then the first n elements form exactly what we desire, but the problem being it requires m space now we will prove that the above algorithm achieves the same result with n space.

(a) Informal proof using Backward Analysis

Since this is an online sampling experiment which cannot be backtracked, we can see that at each point of time we should have a selection that is the result of uniformly, randomly, independently selecting n apples till then. T

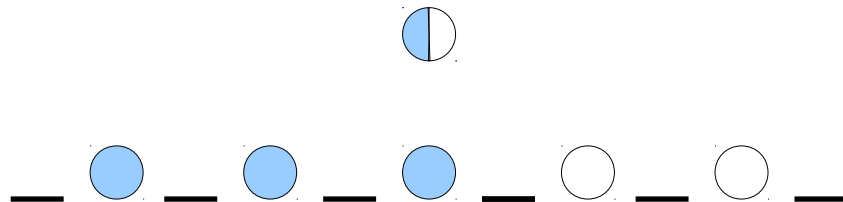


Figure 4: Given that there were 6 apples, and the 6 apple was removed where could it have been removed from

Let the i^{th} step be completed and the i^{th} apple be either discarded (kept as ghost elements), or kept in the ordering, now consider the condition just before it is placed, given that it was placed in some location, there are exactly i locations in which it can be placed, and since it is a uniform random distribution we can see that it should have been placed in any of the places with equal probability. Hence, our method of taking a random integer from 1 to $i + 1$ works, since we need to only consider if it is placed, we will only check if the number is $\leq n$. Now once we have placed the element it has to move the other elements appropriately.

(b) Formal Proof using Induction

- i. It works for the trivial case n apples since they are explicitly permuted.

- ii. Now assume that it works for the case where we have sampled k apples. We have ordered and stored n apples, since for the discarded apples order doesn't matter we just keep a count of the number.
- iii. Now consider the $k + 1^{th}$ apple, we can see that if we already know that with the k apples we already have the permutation, out of which the first n apples are chosen, since we will never choose the rest of the apples as said, we need to maintain only their count, and think as if they exist, then placing the apples as seen in the last image, will give a totally uniform distribution. Placing of the apples in the possible $(k + 1)$ location is nothing but uniformly randomly placing them in the total ordering and checking if it comes in the first n elements or not.

→ Answer

Submitted by Abhimanyu M A (11111002) , Sumesh T A (11111065) on October 20, 2011.