PROBABİLİSTİC ANALYSİS

335 Fall 2022 – Recitation 4

DICE

1. Expected value of the outcome of rolling a dice
2. Expected value of the summations of outcome of rolling pair of dice
3. Expected value of the distinct summations of outcome of rolling pair of dice
4. Expected value of the summations of distinct outcome of rolling pair of dice

DICE2 \ DICE1	1	2	3	4	5	6	AVG
1	2	3	4	5	6	7	4.5
2	3	4	5	6	7	8	5.5
3	4	5	6	7	8	9	6.5
4	5	6	7	8	9	10	7.5
5	6	7	8	9	10	11	8.5
6	7	8	9	10	11	12	9.5
AVG	4.5	5.5	6.5	7.5	8.5	9.5	7

For this exercise, we know/assume that dices are fair (equally likely).

Therefore, the question becomes 'which values should we take average?'.

1. In Q1, We have one dice, thus our sample space is {1,2,3,4,5,6}. We must consider all sides => (1+2+3+4+5+6)/6 = 3.5.
2. In Q2, we have a pair of dice. The question asks for the summations of all outcomes. Therefore, the sample space is all combinations of dice {(1,1),(1,2),...(3,4),....(5,6),(6,6),}. The expected value is already calculated as 7 (from the previous table). 7 is also the sum of expected values of both dice (3.5+3.5=7).
3. In Q3, the question asks for distinct summations. This is every unique value in summations in the previous slide. The sample space is {2,3,…,11,12}. The expected value is (12*13/2 0 1)/11 = 77/11 = 7.
4. In Q4, the question asks for distinct pairs/combinations where (1,5) is equivalent to (5,1). In this case, the sample space is the upper/lower triangle of the previous table 21 elements). The expected value is 147/21 = 7.

Why it is still 7 even though we changed the sample space?

HAT-CHECK

Problem: A hat0checker losses track of which of N hats belong to which owners and hands them back at random.

1. Probability of the first customer get the correct hat
2. Expected number of customers that gets the correct hat

1. As the ordering of the hats is random, we have 1/n probability.
2. For the second part, the answer is below. $P(X_i)$ is the probability of i^{th} person getting the correct hat. Like above, each person has 1/n probability.

$$X_i = \begin{cases} 0, \, if \;\; costumer \; gets \; another \; hat \\ 1, \, if \; costumer \; gets \; her/his \; hat \end{cases}$$

$$E[X] = E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[X_i]$$

$$E[X_i] = \sum_{i=0}^{n} 0 * \frac{n-1}{n} + 1 * \frac{1}{n} = \frac{1}{n}$$

$$E[X] = \sum_{i=1}^{n} E[X_i] = \sum_{i=1}^{n} \frac{1}{n} = 1$$

BIRTHDAY

1. How many (other) people must there be in a room before the probability that someone has the same birthday as you do is at least 0.5
2. How many (other) people must there be in a room before the probability that at least two people have the same birthday as you is greater than 0.5

1. We asked to calculate P(X≥1) = 1 – P(X=0). Therefore, we need to find the expression of no one has the same birthday as yours. Which is $(364/365)^n$. The remaining as fallows;

$$P(X \geq 1) \geq \frac{1}{2} \qquad log(\frac{1}{2}) \geq log(\frac{364}{365}^n)$$

$$1 - \frac{364}{365}^n \geq \frac{1}{2} \qquad -log(2) \geq n * log(\frac{364}{365})$$

$$\frac{1}{2} \geq \frac{364}{365}^n \qquad n \geq 253$$

2. Now we asked to calculate P(X≥2) = 1 – P(X=0)– P(X=1). As the #P(X=k) terms increase, it's better to use the general formula 0> Binomial.

$$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k} \qquad p = \frac{1}{365}$$

$$\frac{1}{2} \geq \left(\frac{364}{365}\right)^{n-1} \times \frac{364+n}{365} \qquad n \geq 612.257$$

HIRING

```
HIRE-ASSISTANT(n)
  best <- 0
  for i <- 1 to n do
    if candidate[i] is better than candidate[best]
      best <- i
      hire candidate i
```

1. Probability of hiring exactly one time
2. Probability of hiring exactly N times
3. Probability of hiring exactly two times

   … Probability of hiring exactly three times

Total number of possible orders of n candidates is n!

1. We want to hire one time so the best candidate must be interviewed first and do not care the rest. Therefore, the probability of best candidate will come first is (n-1)!/n! = 1/n.
2. If we want to hire every candidate, they must be ordered starting from worst candidate to best. Here we have a specific order from all possible orders. Therefore, the probability becomes 1/n!
3. In Q3, we need just one condition; the first hired must be better than all candidates up to the best candidate. We will have c_{first},[$c_2 \ldots c_k$], c_{best}, [$c_{k+1} \ldots c_n$]. The inner order is not important for both subarrays. We need to calculate for each possible k value.

$$\frac{\sum_{k=1}^{n-1}\binom{n-1}{k}.(k-1)!\,.(n-k-1)!}{n!} = \frac{\sum_{k=1}^{n-1}\frac{(n-1)!}{(n-1-k)!\,.\,k!}\,.(k-1)!\,.(n-k-1)!}{n!} = \frac{(n-1)!\,\sum_{k=1}^{n-1}\frac{1}{k}}{n!} = \frac{1}{n}\sum_{k=1}^{n-1}\frac{1}{k}$$

INVERSION

Problem: Let A[1.. n] be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an inversion (or swap) of A (they are "out of order" with respect to each other).

- Expected value of the inversions

{i,j}	1	2	3	4	5	…
1	0	1/2	1/2	1/2	1/2	…
2	0	0	1/2	1/2	1/2	…
3	0	0	0	1/2	1/2	…
4	0	0	0	0	1/2	…
…	0	0	0	0	0	…

The probabilities of inversion for each {i,j] location is can be represented in the left table. As you can see, to find the expected number of inversions, we need to find how many coordinates are there (which is the upper triangle without the diagonal) and multiply with p=½ (inversions is uniformly likely). Since we do not have any {i,i} coordinates (i<j is given), the edges of the triangle are n and (n-1). The #of coordinates is n*(n-1)/2. Then we need to add p=½ to the equation, then the answer becomes n*(n-1)/4.

QUICK0SORT

```
QUICKSORT(A, p, r)
if p < r
    q = PARTITION(A, p, r)
    QUICKSORT(A, p, q - 1)
    QUICKSORT(A, q + 1, r)
```

- Worst Running Time
- Best Running Time
- Average Running Time

```
PARTITION(A, p, r)
x = A[r]
i = p - 1
for j = p to r - 1
    if A[j] ≤ x
        i = i + 1
        exchange A[i] with A[j]
exchange A[i + 1] with A[r]
return i + 1
```