Quiz2

Haiming Wang

June 10, 2019

Problem Math1. What do you know about χ^2 test?

Solution he chi-squared test is any statistical hypothesis test where the sampling distribution of the test statistic is a chi-squared distribution when the null hypothesis is true. The chi-squared test is used to determine whether there is a significant difference between the expected frequencies and the observed frequencies in one or more categories.

Problem Math2. Given a 2 by 3 grid (which has 6 blocks and 17 edges), shortest route to visit all edges (assuming edge length is 1).

Solution e can solve this by the Euler path. If a path is a Euler path, it will use every edge in the path only once. A path that contains at most two odd nodes will be a Euler path. In this problem, we have 6 odd paths. We can add paths to make them become even paths.
Denote the vertices as following, in 1st row: ABCD, 2nd row: EFGH, 3rd row: IJKL
Then the shortest path is EABFEIJFGKJKLHGCBCDH.

Problem Math 3. X, Y are iid $N(0,1)$, calculate $P(X|X+Y>0)$, try not use density function of joint distribution.

Solution

$$\begin{aligned} P(X|X+Y>0) &= \frac{P(X+Y>0|X)f_X(x)}{P(X+Y>0)} \\ &= \frac{1}{2\pi}e^{-\frac{x^2}{2}}\int_{-x}^{\infty} e^{-\frac{t^2}{2}}\,dt \end{aligned}$$

Problem Math 4. You have a six sided dice, you can keep rolling the dice and you get the dollars equal to the mount of the sum. However, if at some point, the sum is a square number, you must stop and will get zero dollars. (1) If at some point, your sum is 35, should you stop or keep rolling? (2) in (1), if you choose to continue and this is your strategy: you will keep rolling until you exceed 43, what is the most probable amount of dollar you win when you stop? (3) Is there a best strategy for this game, any number that you should stop ?

Solution 1. If we do not continue rolling, we can get 35 instead. If we continue rolling, the expected gain is $\frac{5}{6}(35+4+3.5) = 35.4$. If we do not roll a one, then we will not hit a square number for at least next two roll. The expected value of next roll is 4 (excluding 1). The expected value of next and next roll is 3.5. As a result, we should keep rolling.

2. If we start from 35, for the next roll, we can get 37-41. We can get 38-47 after the second roll. As a result, we have $\frac{1}{36}$ probability to get 43, 44, 45, 46, 47. Also, we can get 43, 44, 45, 46, 47, 48 if we roll the dice three times. The most probable amount of dollar I will win is 43.

Problem Math 5. Given a stick, if randomly cut into 3 pieces, what's the average size of the smallest, of the middle-sized, and of the largest pieces?

Solution The smallest
Let's denote the position of the cut of the stick are made as X_1, X_2. Let the length of each segment be $V_1 = X_1 - X_0$, $V_2 = X_2 - X_1$ and $V_3 = X_3 - X_2$, where $X_0 = 0$, $X_3 = 1$ We can use the order statistics to solve this problem. Let V_1 denote the shortest piece of the stick. Then for $x \leq \frac{1}{3}$, we have

$$P(V_1 > x) = P(V_1 > x, V_2 > x, V_3 > x) = (1-3x)^2$$

As a result,

$$E[V_1] = \int_0^\infty P(V_1 > x)dx = \frac{1}{9}$$

The largest and the middle
The general case of the expectation of any order segment using order statistics can be gotten as

$$E[V_r] = \frac{1}{n}\sum_{j=1}^{r}\frac{1}{n-j+1},$$

where n is the number of segments we have. As a result, the average length of the longest segment is $\frac{11}{18}$; the average length of the middle segment is $\frac{5}{18}$.

Problem Math 6. At a party, N people throw their hats (all hats are different) into the center of room. The hats are mixed up and each people randomly selects one. Let Y be the number of people who select their own hats. Now ask (1) what is the expectation of Y ? (2) what is the variance of Y ? Now, the picking hats game rule is extended. For each hats pick round, the people choosing their own hats quit the game, while others (those picked wrong hats) put their selected hats back in the center of room, mix them up, and then reselect. Also, suppose that this game continues until each individual has his own hat. Suppose N individuals initially join the game, let R(N) be the number of rounds that are run and S(N) be the total number of selections made by the these N individuals, $(N > 1)$. (3) Find the expectation of R(N). (4) Find the expectation of S(N). (5) Find the expected number of false selections made by one of the N people.

Solution 1. The expectation of Y Denote

$$I_i = \begin{cases} 1, & \textit{if the ith people get its hat} \\ 0, & \textit{else} \end{cases}$$

Then the expectation of Y is

$$E[Y] = E[\sum_i = 1^N I_i] = \sum_1^N \frac{1}{N} = 1$$

2. The variance of Y

$$Var[Y] = Var[\sum_i = 1^N I_i] = \sum_1^N Var[I_i] + \sum_{i \neq j} Cov[I_i, I_j] = 1$$

3. Using induction For $N = 1$, we have E[R(1)]=1; Guess when $N = k$, we have E[R(k)]=k. Consider $E[R(n)]$

$$\begin{aligned} E[Y] &= \sum_{i=0}^{n} E[R(n)|X = i]P(X = i) \\ &= \sum_{i=0}^{n}(1 + E[R(n - i)])P(X = i) \\ &= E[R(n)]P(X = 0) + n(1 - P(X = 0)) \end{aligned}$$

As a result, $E[R(n)] = \frac{n(1-P(X=0))}{1-P(X=0)} = n$

Problem Math 7. Consider linear regression of Y on features X1, X2: Model1-(Y,X1), R2 = 0.1; Model2- (Y,X2), R2 = 0.2; Model3-(Y,X1, X2), calculate the range of R2 of Model3.

Solution R^2 under single linear regression is the square of the correlation. Under two variable linear regression, we have the relation

$$R^2(X1, X2) = \frac{corr^2(X1, Y) + corr^2(X2, Y) - 2\, corr(X1, Y)corr(X2, Y)corr(X1, X2)}{1 - corr^2(X1, X2)}.$$

This can be proved by using the cosine relationship btwn the vectors and angles in the Euclidean space. Here, we have $corr^2(X1, Y) = R^2(X1) = 0.1$ and $corr^2(X2, Y) = R^2(X2) = 0.2$. The range of $corr(X1, X2)$ can be calculated as the cosine of the angle between X1 and X2. Denoted this angle as θ. We know that $\theta_1 - \theta_2 \leq \theta \leq \theta_1 + \theta_2$. As a result,

$$cos(\theta_1)cos(\theta_2) - sin(\theta_1)sin(\theta_2) \leq corr(X1, X2) = cos(\theta) \leq cos(\theta_1)cos(\theta_2) + sin(\theta_1)sin(\theta_2).$$

Thus, the range is between 0.2 to 1.0

Problem Math 8. Given a function for a fair coin, write a function for a biased coin that returns heads with probability 1. n (n is a param).

Solution ssuming we have a function that gives us fair coin toss as following

int fairCoinToss();

Then, we can get function of a biased coin toss that gives us a head with $\frac{1}{n}$ probability.

```
int biasedCoinToss(int n) {
  if (n == 1) {
    return 1; // always heads
  } else if (n == 2) {
    return fairCoinToss();
  }
  int r = random_number(n);
  return r == 0 ? 1 : 0;
}
```

where the random_number(n) function can uniformly gives us a random number from n to n-1.

Problem Math 9. 10 islands with 9 bridges. The bridges are either strong or weak (half half). Weak bridge falls if stepped on and the man is drifted to the 1st island, then all the bridges are miraculously fixed. To arrive the 10th island, how many bridges on average the man has to cross?

Solution enote

$$X_i = \begin{cases} 1, & \textit{if we take the weak bridge in order to get from island i to i+1} \\ 0, & \textit{else} \end{cases}$$

If we indeed take the weak bridge from island i to i+1, we need to get back to 1st island. As a result, we need to take extra iX_i.

The total number of bridges is $9 + \sum_{i=1}^{9} iX_i$. Thus, the expectation is $9 + \sum_{i=1}^{9} iE(X_i) = 9 + \frac{1}{2}\sum_{i=1}^{9} i = 31.5$

Problem Programming 10. Explain the following code:

```
const int* const fun(const int* const& p) const;
```

Solution ??

Problem Programming 11. How do you implement delete operation in a single-linked list?

Solution The following is the psuedo code of the "delete" operation

```
def delete(target, head) {
    p = head
    while p.next != target:
        p = p.next
    p.next = target.next
    del target
}
```

Problem Programming 12. Implement the interface for matrix class in C++.

Solution maybe need add, inverse and multiply

Problem Programming 13. Can the constructor of a class be virtual? How to realize a similar function as a virtual constructor?

Solution The purpose of creating a virtual constructor is to create a copy of an object or a new object without knowing its concrete type.
A virtual call is a mechanism to allow us to call a function with knowing only an interfaces/-parent class and not the exact type of the object. To create an object you need complete information. In particular, you need to know the exact type of what you want to create. Consequently, a "call to a constructor" cannot be virtual.

Problem Programming 14. Is it okay for a non-virtual function of the base class to call a virtual function?

Solution Yes. It is allowed.

Problem Programming 15. Given a string, return the longest palindrome subsequence.

Solution Leetcode problem 5

```
class Solution(object):
    def longestPalindrome(self, s):

        n = len(s)
        start, par_len = 0, 1
        for i in range(1,n):

            if i >= par_len + 1:
                x = s[i - par_len - 1: i + 1]
                if x == x[::-1]:
                    start, par_len = i - par_len - 1, par_le
                    continue
            y = s[i - par_len: i + 1]
            if y == y[::-1]:
                start, par_len = i - par_len, par_len + 1

        return s[start: start+par_len]


        """
        :type s: str
        :rtype: str
        """

```

Problem Programming 16. How to inverse a string of sentence (without reverse the word) ?

Solution Leetcode question 151

```python
class Solution(object):
    def reverseWords(self, s):
        s = s.lstrip(' ').rstrip(' ')
        x = [ ea for ea in s.split(' ') if ea ]
        return ' '.join( x[::-1] )
        """
        :type s: str
        :rtype: str
        """
```

Problem Programming 17. Say you have an array for which the i-th element is the price of a given stock on day i. Design an algorithm to find the maximum profit. You may complete at most two transactions. Note: You may not engage in multiple transactions at the same time (i.e., you must sell the stock before you buy again).

Solution Leetcode question 123.

```
# -*- coding: utf-8 -*-
"""
Created on Sat Jun  8 18:23:34 2019

@author: haimingwd
"""

def maxProfit(k, prices):
    n = len(prices)
    if n <= 1:
        return 0
    ret = [ prices[i+1] - prices[i] for i in range(n-1) ]
    if k >= n/2:
        return sum([max(ea,0) for ea in ret])
    def helper(k):
        if k == 0:
            return [ 0 for _ in range(n) ]
        res1 = helper(k-1)
        res3 = [ 0 for _ in range(n) ]
        res2, res3[1] = ret[0], max(ret[0], 0)
        for i in range(1, n-1):
            res2 = max( res2 + ret[i], res1[i-1] + ret[i] )
            res3[i+1] = max(res3[i], res2)
        return res3
    return helper(k)[-1]

maxProfit(2, [1,3,4,6,2,9,2,4,6])
```

Problem Programming 18. The book problem: There is a group of N ($2 \leq N \leq 1000$) people which are numbered 1 through N, and everyone of them has not less than $\frac{N+1}{2}$ friends. A man with number 1 has the book, which others want to read. Write the program which finds a way of transferring the book so that it will visit every man only once, passing from the friend to the friend, and, at last, has come back to the owner. Note: if A is a friend of B then B is a friend of A. INPUT: First line of input contains number N. Next N lines contain information about friendships. (i+1)-th line of input contains a list of friends of i-th man. OUTPUT: If there is no solution then your program must output ?No solution?. Else your program must output exactly N + 1 number: this sequence should begin and should come to end by number 1, any two neighbors in it should be friends, and any two elements in it, except for the first and last, should not repeat.

Solution ????