QUIZ 2

JINGGUO LAI

1. Math/Stat

Question 1. According to Wikipedia, a χ^2 test is any statistical hypothesis test in which the sampling distribution of the test statistic is a χ^2 distribution when the null hypothesis is true. χ^2 tests are often constructed from a sum of squared errors, or through the sample variance. The χ^2 tests that I am familiar with are *goodness of fit test* and χ^2 *test for independence*.

(i) Goodness of fit test
Let X be a discrete random variable with values $1, 2, ..., k$. Denote $\mathbb{P}(X = i) = p_i, i = 1, 2, ..., k$. We want to test H_0 against all alternatives, where

$$H_0 : p_i = p_{i0}, i = 1, 2, ..., k.$$

We sample the random variable X n times to get a frequency that $X = i$ happens n_i times, i.e. $n_1 + n_2 + ... + n_k = n$. If the null hypothesis H_0 is true, the test statistic

$$Q_{k-1} = \sum_{i=1}^{k} \frac{(n_i - np_{i0})}{np_{i0}}$$

has an approximate χ^2 distribution with $k - 1$ degrees of freedom.

(ii) χ^2 test for independence
Let X and Y be two discrete random variables. Denote $\mathbb{P}(X = i, Y = j) = p_{ij}$, $\mathbb{P}(X = i) = p_{i\cdot}$, and $\mathbb{P}(Y = j) = p_{\cdot j}$. We want to test H_0 against all alternatives, where

$$H_0 : p_{ij} = p_{i\cdot}p_{\cdot j}, i = 1, 2, ..., I,\ j = 1, 2, ..., J.$$

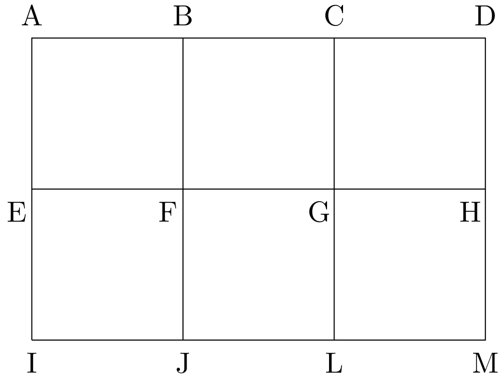
We sample the random variable X n times and the random variable Y m times to get a frequency that $X = i, Y = j$ happens n_{ij} times, i.e. $\sum_i \sum_j n_{ij} = n$. If the numll hypothesis H_0 is true, the test statistic

$$\sum_{i}^{I} \sum_{j}^{J} \frac{[n_{ij} - n(n_{i\cdot}/n)(n_{\cdot j}/n)]^2}{n(n_{i\cdot}/n)(n_{\cdot j}/n)}$$

has an approximate χ^2 distribution with $(I - 1)(J - 1)$ degree of freedom, where

$$n_{i\cdot} = \sum_{j}^{J} n_{ij},\ n_{\cdot j} = \sum_{i}^{I} n_{ij}.$$

Question 2. Given a 2×3 grid with 6 blocks and 17 edges



Assuming edge length is 1, we want to find the shortest route to visit all edges. This is a variant of the famous *Seven Bridges of Königsberg Problem.* In the original paper of Euler, he proved that

Theorem (Euler). (i) *A finite graph G contains an Euler circuit if and only if G is connected and contains no vertices of odd degree.*

(ii) *A finite graph G contains an Euler path if and only if G is connected and contains at most two vertices of odd degree.*

In this problem, we can see that nodes B, C, E, H, J, L are vertices of odd degrees. So we need to connect nodes B, C and nodes J, L to guarentee the existence of an Euler path. Once we have these done, we know the shortest route to visit all edges should be with length $17 + 2 = 19$. A possible shortest route would be

$$E - A - B - C - D - H - G - C - B - F - E - I - J - L - G - F - J - L - M - H$$

Question 3. X and Y are i.i.d. $\mathcal{N}(0, 1)$.

$$\begin{aligned}\mathbb{P}(X = x | X + Y > 0) &= 2\mathbb{P}(X = x, Y > -x) \\ &= 2\mathbb{P}(X = x)\mathbb{P}(Y > -x) \\ &= 2 \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \cdot \int_{-x}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}} du.\end{aligned}$$

Question 4.

(i) If our sum now is 35 and we choose to keep roling, then the next step we would end up with 36, 37, 38, 39, 40, or 41 with equal probability. When we have a sum of 36, we are out. Otherwise, we definitely want to keep playing since $41 + 6 = 47 < 49$. This results in an expectation of

$$\frac{1}{6} \times 0 + \frac{1}{6} \times (37 + 3.5) + ... + \frac{1}{6} \times (41 + 3.5) = 35.4167 > 35.$$

Hence, we should keep playing.

(ii) Now that our sum is 35 and our strategy is keep rolling until exceeding 43, we could only stop the game and receive 0, 44, 45, 46, 47, 48, or 49. Run the following Python script, we can compute the probabilities to stop the game according to this strategy

(36, 0.16666666666666666),
(44, 0.23113556908245694), (45, 0.19872816167504956),
(46, 0.16091951969974091), (47, 0.11758104233348576),

(48, 0.06869129610577657), (49, 0.03698762098003353).
So, the most probable amount of dollar we win would be 44.

```
results = []
# initialize the first 6 cases
results.append(1.0/6)
results.append(1.0/6)
results.append(1.0/6 + (1.0/6)**2)
results.append(1.0/6 + 2*(1.0/6)**2 + (1.0/6)**3)
results.append(1.0/6 + 3*(1.0/6)**2 + 2*(1.0/6)**3 +
        (1.0/6)**4)
results.append(1.0/6 + 4*(1.0/6)**2 + 3*(1.0/6)**3 +
        2*(1.0/6)**4 + (1.0/6)**5)

# P(n) = 1/6\sum P(n-i) whenever n-i is not a squared number
for i in xrange(42, 50):
        curr = 0
        for j in xrange(6):
                if ((i - j - 1) != 36) & (i - j - 1 <= 43):
                        curr += results[i - j - 1 - 36]
        curr /= 6
        results.append(curr)
```

(iii) This is really complicated.. Maybe this post helps?

The only time we might consider stopping is when k is nearly a perfect square. Suppose $k \approx n^2$ (more specifically, take $n^2 - 6 \leq k < n^2$).

If we stop, our score is always at least $n^2 - 6$. If we try to continue past n^2, and then stop when we nearly reach $(n+1)^2$, the probability that we actually make it past n^2 is at most 5/6, so the expected score is at best $5/6((n+1)^2 - 1) = 5/6(n^2 + 2n)$. So stopping now is better than going through $(n+1)^2$ whenever $n^2 - 6 > 5/6(n^2 + 2n)$, and possibly sooner. This inequality holds whenever $n \geq 13$.

Question 5. Let x and y be the positions of the two breaking points, x the leftmost one, i.e. $0 < x < y < 1$. So, the 3 pieces are with length x, $y - x$, and $1 - y$.

If x is the smallest piece, then $x < y - x$ and $x < 1 - y$, i.e.

$$\{(x, y) : 0 < 2x < y < 1, 0 < x + y < 1\}.$$

This region has area 1/9. So the average size of the smallest piece is 1/9.

If x is the largest piece, then $x > y - x$ and $x > 1 - y$, i.e.

$$\{(x, y) : 0 < x < y < 2x < 1, x + y > 1\}.$$

This region has area 11/18. So the average size of the largest piece is 11/18. Eventually, the averge of the middle-sized piece is 5/18.

Question 6. N people wair N hats.

(i) Let Y be the number of people who select their own hats. To compute $\mathbb{E}[Y]$, we introduce i.i.d. random variables

$$Y_i = \begin{cases} 1 & \text{the i-th person select his own hat} \\ 0 & \text{otherwise} \end{cases}$$

So, we can compute

$$\mathbb{E}[Y] = \sum_{i=1}^{N} \mathbb{E}[Y_i] = N \cdot \frac{1}{N} = 1.$$

(ii) To compute $var(Y)$, again we have

$$\begin{aligned} var(Y) = var\left(\sum_{i=1}^{N} Y_i\right) &= \sum_{i=1}^{N} var(Y_i) + \sum_{i \neq j} cov(Y_i, Y_j) \\ &= N \cdot \frac{N-1}{N^2} + N(N-1) \cdot \frac{1}{N^2(N-1)} = 1. \end{aligned}$$

(iii) Let $R(N)$ be the number of rounds that are run.
Let $S(N)$ be the total number of selections made by these N individuals,
Let $F(N)$ be the number of false selections made by these N individuals.

Intuitively, we can answer $\mathbb{E}[R(N)]$, $\mathbb{E}[S(N)]$, and $\mathbb{E}[F(N)]$ pretty easily. Since by (i), on average we know each round there should be 1 person selects his/her hat. Hence, the game should last for N rounds, i.e. $\mathbb{E}[R(N)] = N$.
Since on average, every round there should be 1 person quits the game, the total number of selections made should be $N + (N-1) + ... + 1 = (N+1)N/2$, i.e. $\mathbb{E}[S(N)] = (N+1)N/2$.
Similarly, the number of false selections made should be $(N-1) + (N-2) + ... + 1 = N(N-1)/2$, i.e. $\mathbb{E}[F(N)] = N(N-1)/2$. In terms of the expected number of false selections made by 1 person, we have by symmetry the answer equals $(N-1)/2$.

However, to make the calculation rigorously, we need the following facts

$$\begin{aligned} \sum_{n=0}^{N} \mathbb{P}(Y = n) &= 1, \\ \mathbb{E}[Y] = \sum_{n=0}^{N} n\mathbb{P}(Y = n) &= 1, \\ \mathbb{E}[Y^2] = \sum_{n=0}^{N} n^2\mathbb{P}(Y = n) &= 1. \end{aligned}$$

Let us do the mathematically rigorous calculation.

(iv) Prove by induction that $\mathbb{E}[R(N)] = N$. Trivially, $\mathbb{E}[R(0)] = 0$. Assume that $\mathbb{E}[R(n)] = n$ for all $0 \leq n < N$, we have

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

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

Hence, using the fact that $\mathbb{P}(Y=0)>0$, we can solve exactly $\mathbb{E}[R(N)] = N$.

(v) Prove by induction that $\mathbb{E}[S(N)] = (N+1)N/2$. Trivially, $\mathbb{E}[S(0)] = 0$. Assume that $\mathbb{E}[S(n)] = (n+1)n/2$ for all $0 \le n < N$, we have

$$
\begin{aligned}
\mathbb{E}[S(N)] &= \sum_{n=0}^{N} \mathbb{E}[S(N)|Y=n]\mathbb{P}(Y=n) = \sum_{n=0}^{N} \left(\mathbb{E}[S(N-n)] + N\right)\mathbb{P}(Y=n) \\
&= N + \mathbb{E}[S(N)]\mathbb{P}(Y=0) + \sum_{n=0}^{N} \frac{(N-n+1)(N-n)}{2}\mathbb{P}(Y=n) - \frac{(N+1)N}{2}\mathbb{P}(Y=0) \\
&= N + \mathbb{E}[S(N)]\mathbb{P}(Y=0) + \frac{(N+1)N}{2}\sum_{n=0}^{N} \mathbb{P}(Y=n) - N\sum_{n=0}^{N} n\mathbb{P}(Y=n) \\
&+ \frac{1}{2}\sum_{n=0}^{N} n^2\mathbb{P}(Y=n) - \frac{1}{2}\sum_{n=0}^{N} n\mathbb{P}(Y=n) \\
&= \mathbb{E}[S(N)]\mathbb{P}(Y=0) + \frac{(N+1)N}{2}(1-\mathbb{P}(Y=0))
\end{aligned}
$$

Hence, using the fact that $\mathbb{P}(Y=0)>0$, we can solve exactly $\mathbb{E}[R(N)] = (N+1)N/2$.

(vi) Again prove by induction that $\mathbb{E}[F(N)] = N(N-1)/2$. Trivially, $\mathbb{E}[F(0)] = 0$. Assume that $\mathbb{E}[F(n)] = n(n-1)/2$ for all $0 \le n < N$, we have

$$
\begin{aligned}
\mathbb{E}[F(N)] &= \sum_{n=0}^{N} \mathbb{E}[F(N)|Y=n]\mathbb{P}(Y=n) = \sum_{n=0}^{N} \left(\mathbb{E}[F(N-n)] + N - n\right)\mathbb{P}(Y=n) \\
&= \mathbb{E}(F(N))\mathbb{P}(Y=n) + \sum_{n=0}^{N} \frac{(N-n+1)(N-n)}{2}\mathbb{P}(Y=n) - \frac{N(N-1)}{2}\mathbb{P}(Y=0) \\
&= \mathbb{E}(F(N))\mathbb{P}(Y=n) + \frac{N(N-1)}{2}(1-\mathbb{P}(Y=0))
\end{aligned}
$$

Hence, using the fact that $\mathbb{P}(Y=0)>0$, we can solve exactly $\mathbb{E}[F(N)] = N(N-1)/2$. In terms of the expected number of false selections made by 1 person, we have by symmetry the answer equals $(N-1)/2$.

Question 7. Regress Y on X_1, we have $R^2 = corr(X_1, Y)^2 = 0.1$. Regress Y on X_2, we have $R^2 = corr(X_2, Y)^2 = 0.2$. Now, consider the Model3-(Y, X_1, X_2), i.e.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon.$$

We have the R^2 for this model calculated by

$$R^2 = \frac{ESS}{TSS} = \frac{\sum_i (\widehat{Y}_i - \overline{Y})^2}{\sum_i (Y_i - \overline{Y})^2} = \frac{\widehat{\beta_1}^2 var(X_1) + \widehat{\beta_2}^2 var(X_2) + 2\widehat{\beta_1}\widehat{\beta_2} cov(X_1, X_2)}{var(Y)},$$

where $\widehat{\beta_1}$ and $\widehat{\beta_2}$ are the OLS estimators of Model3-(Y, X_1, X_2). Since we have

$$cov(X_1, Y) = \widehat{\beta_1} var(X_1) + \widehat{\beta_2} cov(X_1, X_2),$$
$$cov(X_2, Y) = \widehat{\beta_1} cov(X_1, X_2) + \widehat{\beta_2} var(X_2).$$

We can simplify R^2 to

$$R^2 = \frac{\widehat{\beta_1} cov(X_1, Y) + \widehat{\beta_2} cov(X_2, Y)}{var(Y)}.$$

Moreover, we can solve for $\widehat{\beta_1}$ and $\widehat{\beta_2}$

$$\widehat{\beta_1} = \frac{cov(X_1, Y) var(X_2) - cov(X_2, Y) cov(X_1, X_2)}{var(X_1) var(X_2) - cov(X_1, X_2)^2},$$
$$\widehat{\beta_2} = \frac{cov(X_1, Y) cov(X_1, X_2) - cov(X_2, Y) var(X_1)}{cov(X_1, X_2)^2 - var(X_1) var(X_2)}.$$

Substitute these in the expression of R^2, we have

$$R^2 = \frac{0.3 - 2corr(X_1, Y) corr(X_2, Y) corr(X_1, X_2)}{1 - corr(X_1, X_2)^2} = \frac{0.3 - 2\rho_{1y}\rho_{2y}\rho_{12}}{1 - \rho_{12}^2}.$$

We know that $\rho_{1y}^2 = 0.1$, $\rho_{2y}^2 = 0.2$, and the following matrix must be positive definite

$$\begin{pmatrix} 1 & \rho_{12} & \rho_{1y} \\ \rho_{12} & 1 & \rho_{2y} \\ \rho_{1y} & \rho_{2y} & 1 \end{pmatrix}$$

Hence, we deduce that

$$\rho_{12}^2 - 2\rho_{1y}\rho_{2y}\rho_{12} - 0.7 \leq 0.$$

The first case to consider is $\rho_{1y}\rho_{2y} = \sqrt{0.1} \cdot \sqrt{0.2} = 0.1\sqrt{2} > 0$. In this case, we can solve

$$-0.5\sqrt{2} \leq \rho_{12} \leq 0.7\sqrt{2}.$$

And we know that

$$R^2 = \frac{0.3 - 0.2\sqrt{2}\rho_{12}}{1 - \rho_{12}^2},$$

so we can easily find that R^2 is decreasing on $[-0.5\sqrt{2}, \sqrt{2}-1]$ and increasing on $[\sqrt{2}-1, 0.7\sqrt{2}]$. Therfore, we conclude that

$$0.15 + 0.05\sqrt{2} \leq R^2 \leq 1.$$

Note that there is another case $\rho_{1y}\rho_{2y} = -\sqrt{0.1} \cdot \sqrt{0.2} = -0.1\sqrt{2} < 0$. Similar calculation yields again

$$0.15 + 0.05\sqrt{2} \leq R^2 \leq 1.$$

Question 8. There are at least two classical ways to solve this problem.

- Method I
  Denote $1/n = 0.p_1p_2...p_k... = p_1 2^{-1} + p_2 2^{-2} + ... + p_k 2^{-k} + ...$, $p_i \in \{0, 1\}$. Then, we start tossing the fair coin and count heads as 1 and tails as 0. Let $s_i \in \{0, 1\}$ be the result of the i-th toss starting from $i = 1$. After each toss, we compare p_i with s_i.
  If $s_i < p_i$, we return a head and the coin tossing stops. If $s_i > p_i$, we return a tail and the coin tossing stops. If $s_i = p_i$, we continue to toss more coins.

If the sequence happens to be finite, e.g. $1/4$, when we reach the final stage, we return a tail. This will return heads with probability $1/n$.

- Method II
The problem is equivalent to generate $rand(n)$ using $rand(2)$. We first recall a simple lemma.

Lemma. *For every $m \leq n$, the following algorithm generates $rand(m)$ from $rand(n)$*

```
while true
        x = rand(n)
        if x < m
                return x
```

Having this lemma, we first find the smallest k, such that $2^{k+1} \geq n$. Note that the expression

$$rand(2) + 2 \cdot rand(2) + ... + 2^k \cdot rand(2)$$

generates uniformly distribution on $[0, 2^{k+1})$. Using the lemma, we can generate $rand(n)$.

Question 9. We can formulate this as a finite state Markov Chain model. Let μ_i be the expected number of bridges the man crosses when he arrives at the i-th island, $i = 0, 1, ..., 9$, where $i = 9$ is an absorbing state. We have

$$\begin{aligned}
\mu_0 &= \frac{1}{2}\mu_0 + \frac{1}{2}\mu_1 + 1 \\
\mu_1 &= \frac{1}{2}\mu_0 + \frac{1}{2}\mu_2 + 1 \\
\mu_2 &= \frac{1}{2}\mu_0 + \frac{1}{2}\mu_3 + 1 \\
\mu_3 &= \frac{1}{2}\mu_0 + \frac{1}{2}\mu_4 + 1 \\
&\dots \\
\mu_8 &= \frac{1}{2}\mu_0 + \frac{1}{2}\mu_9 + 1 \\
\mu_9 &= 0.
\end{aligned}$$

It is easy to solve $\mu_0 = 2^{10} - 2 = 1022$.

2. Programming

Question 10. The code: **const int *const fun(const int *const& p) const;** means this is a const member function named fun that takes a reference to a const pointer to a const int and returns a const pointer to a const int.

Question 11. Implement delete operation in a single-linked list.

Question 12. Implement the interface for matrix class.

Question 13. No. A virtual call is a mechanism to get work done given partial information. In particular, *"virtual"* allows us to call a function knowing only any interfaces 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.

—(Bjarne Stroustup (P424 The C++ Programming Language SE))

We can get the effect of a "virtual constructor" by a virtual clone() member function, or a virtual create() member function. (https://isocpp.org/wiki/faq/virtual-functions#virtual-ctors)
Example:

```
class Shape {
public:
  virtual ~Shape() { }                    // A virtual destructor
  virtual void draw() = 0;                // A pure virtual function
  virtual void move() = 0;
  // ...
  virtual Shape* clone()  const = 0;    // Uses the copy constructor
  virtual Shape* create() const = 0;    // Uses the default constructor
};
class Circle : public Shape {
public:
  Circle* clone()  const;   // Covariant Return Types; see below
  Circle* create() const;   // Covariant Return Types; see below
  // ...
};
Circle* Circle::clone()  const { return new Circle(*this); }
Circle* Circle::create() const { return new Circle();      }
```

Question 14. Yes. Example:

```
class Polygon {
  protected:
    int width, height;
  public:
    void set_values (int a, int b)
      { width=a; height=b; }
    virtual int area () const
```

```
      { return 0; }
    int print() const
      { return this->area();}
};

class Rectangle: public Polygon {
  public:
    int area () const
      { return width * height; }
};

class Triangle: public Polygon {
  public:
    int area () const
      { return (width * height / 2); }
};
```

Question 15.

Question 16.

Question 17.

Question 18.

Department of Mathematics and Computer Science, Brown University, Providence, RI 02912
E-mail address: jglai@math.brown.edu