MATHS 7027 Mathematical Foundations of Data Science

Trimester 2，2024

Dongju Ma

Assignment 2 - Question 1

1(a)

To arrange the labourers into 5 groups of 3, we should pick 3 random guys from the origin 15 people, then 3 random guys from the 12 people left, etc. So the expression of this process should be:

$$\frac{\binom{15}{3} \times \binom{12}{3} \times \binom{9}{3} \times \binom{6}{3} \times \binom{3}{3}}{5!}$$

which equals 1401400.
And the each team should have different task with others, we should multiple the result before with all the permutions the assign ways have, which is

$$1401400 \times 5! \times 4! \times 3! \times 2! \times 1!$$

which equals 16816800

1(b)

Now they would be performing the same tasks, we should just divide the labourers to 5 diffrent groups. The permution in each group doesn't count and the permution of groups also doesn't count, so we can just use combination of 15 taken 3 to times tne combination of 15-3=12 taken 3 and so on till the combination of 3 taken 3. Which is important is because of the permution of th3 combinations also doesn't count so we would divide by 5!
The expression is:

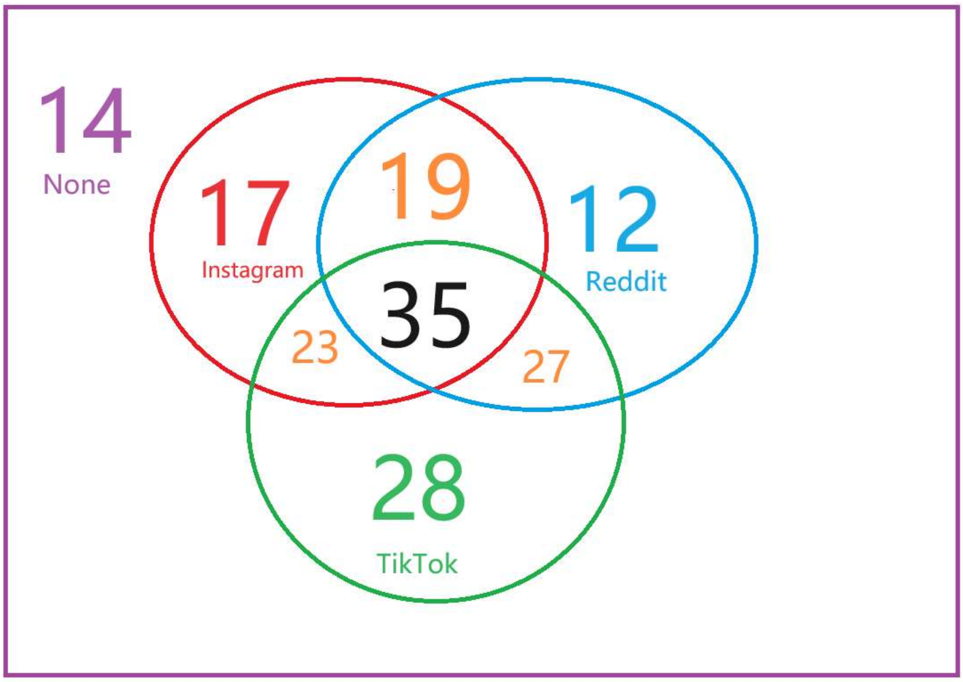
$$\frac{\binom{15}{3} \times \binom{12}{3} \times \binom{9}{3} \times \binom{6}{3} \times \binom{3}{3}}{5!}$$

which equals 1401400.

Assignment 2 - Question 2

2(a)

The Venn diagram is below:



2(b)

- i. The probability of who uses Reddit is

$$\frac{12+27+35+19}{14+17+19+35+23+28+27+12} = \frac{93}{175} \approx 0.554$$

- ii The probability of who uses at least one platform is

$$\frac{175-14}{175} = \frac{23}{25} = 0.92$$

- iii Suppose the probability of who doesn't uses Instagram is $P(A)$, the probability of who uses TikTok is $P(B)$. So who does not use Instagram given that they use TikTok is $P(A|B)$

So we can learn

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{28+27}{28+27+35+23} \approx 0.487$$

Assignment 2 - Question 3

3(a) The probability of five are all diamonds is

$$\frac{C(13,5)}{C(52,5)} \approx 0.000495$$

3(b) The probability of five are the same suit is

$$\frac{C(13,5)\times C(4,1)}{C(52,5)}\approx 0.00198$$

3(c) There are only 4 circumstances of Royal Flush happening, which is the cards with 4 different suits. So, the probability would be:

$$\frac{4}{C(52,5)}\approx 1.54\times 10^{-6}$$

MATHS 7027 Mathematical Foundations of Data Science

Assignment 2 - Question 4

In this question, we will explore the use of `for` loops to calculate sums.

The notebook below will teach you how to use these. You should modify this notebook to solve the problems in the assignment, then attach your completed notebook (as a PDF) to your solutions to the other assignment questions.

Let's get started.

We can calculate sums by adding each new term to the existing sum. This suggests an efficient way to compute sums - by using a **loop**!

`for` loops

Here's an example of a `for` loop, which we use to calculate the sum of the first $n=10$ natural numbers:

In [ ]:
```
n = 10
sum_n = 0

for i in range(1,n+1):
    sum_n = sum_n + i

print("Sum of the first 10 numbers:",sum_n)
```

You can check that this is correct using the relevant formula from the course material.

Let's go through the code line-by-line.

First, we define the number of terms to sum (`n = 10`).
Next, we initialise our sum at 0 (`sum_n`).
Then, we loop over the values in `range(1,n+1)` , adding each new value (`+ i`) to `sum_n` . The operation that gets repeated inside the loop needs to be *indented* in Python - the language makes you write nice-looking code, but nice-looking code is generally easy to read, so this is not a bad thing.

The final line, which is not indented, is outside of our `for` loop, so our result is only printed after the loop is finished running, and only prints once.

So, what are the values in `range(1,n+1)` ? Type `list(range(1,n+1))` in the cell below to find out.

In [ ]:
```
list(range(1,n+1))
```

That might be a bit surprising: `range(a,b)` creates a list of integers, starting at `a` and ending at `b-1` . You might expect the list to end at `b` , but this isn't the case. The `range` command can also take a single argument, `range(n)` , which is equivalent to `range(0,n)` .

To give us a better understanding of what's going on you can try the following (but you **don't** have to submit this):

- put `print(i)` inside the loop above to see how the loop is incremented
- put `print(sum_n)` inside the loop to see how the sum is recursively calculated

4(a)

Use a loop to calculate:

$$\sum_{n=3}^{9} n^3 - 6n^2$$

In [ ]:
```
n = 3
sum_n =0

for i in range(n,10):
    sum_n += i**3 - 6*i**2

print(sum_n)
```

```
336
```

4(b)

Now, let's try to approximate an infinite series. Consider the following series:

$$\sum_{i=0}^{\infty} \frac{1}{i!}$$

Use loops to calculate the partial sums S_4, S_9, and S_{20}.

Based on your partial sums, say if you think the infinite series converges, and what value it converges to.

Make sure you show the values you calculate for S_4, S_9, and S_{20}.

```
In [ ]: def series_sum(n):
            sum_n = 0
            factorial_n = 1
            for i in range(0,n+1):
                if i == 0:
                    factorial_n = 1
                else:
                    factorial_n *= i

                sum_n += 1/factorial_n

            return sum_n

        s4 = series_sum(4)
        s9 = series_sum(9)
        s20 = series_sum(20)
        print(s4,s9,s20)
```

```
2.708333333333333 2.7182815255731922 2.7182818284590455
```

I think the series does converges and it converges to e.