Reverse Engineering Algorithmic Mechanism Behind WeChat Red Envelope

Qifan Zhang, 47422183, zhangqf@shanghaitech.edu.cn
School of Information Science and Technology
ShanghaiTech University

***Abstract*—*WeChat Red Envelope* is now becoming a fashion around. We always ask, is it fair for every one participating in this *trying-luck* game? This article will look into this problem taking the case that 5 people pick 10 *yuan* as an example.**

I. Introduction

I give out 10 *yuan* for 5 people each time. Priorly my classmates and I conduct 80 trials to get the prior distribution of money every one gets by sequence of picking red envelopes. Then I guess and build up our model based on the 80 prior trials. Finally, I will test my model by 100 posterior trials.
All cell phones used in our experiments are conducted on WeChat 6.6.1 on iOS.
The main results and contributions of this report are summarized as follows:

- **Modeling.**
- **Simulation Verification.**
- **Theoretical Analysis**.

II. Model and Algorithms

From 80 prior cases, we notice that the first participant **will never get more than $\frac{2}{5}$ of the total money (i.e. 4 *yuan*)**. The distribution and potential PMF curve are showing below:
Define money gotten by the jth participant is X_j.

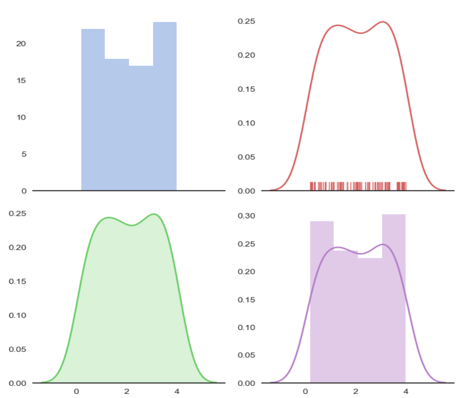

Fig. 1. Distribution of X_1

We can see it clearly that no red envelop for the first participant is larger than 4 *yuan*, and from 0 to 4 *yuan*, it distributes nearly uniformly. I also found that in a 100-*yuan* red envelop, the first participant could get at most 40 *yuan* and in a 1-*yuan* red envelop, the first participant could get at most 0.4 *yuan*.
According to facts above, I model that:

$$X_1 \sim Unif(0, \frac{2}{5} \times 10)$$

Then, we guess that every one except the last is following this rule, i.e.

$$X_j | X_1, X_2, ..., X_{j-1} \sim Unif(0, \frac{2}{5} \times (10 - \sum_{i=1}^{j-1} X_i)) \ (1 \leq j \leq 4)$$

$X_5 = 10 - \sum_{i=1}^{4} X_i$
It does not fit the following prior distributions, but the shape is quite similar:

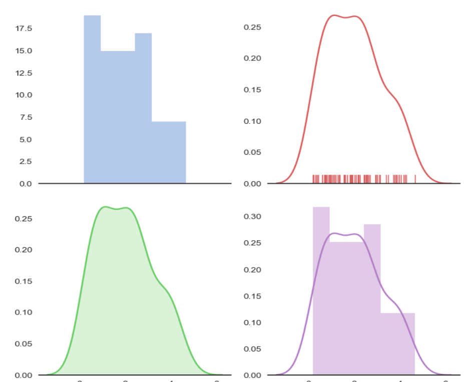

Fig. 2. Distribution of X_2

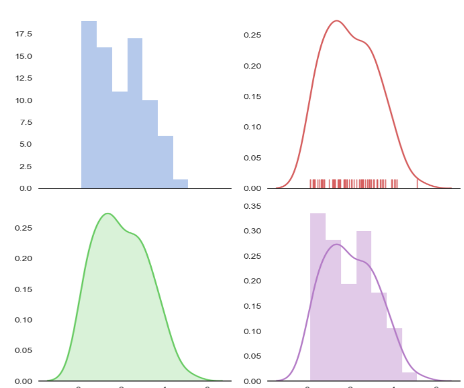

Fig. 3. Distribution of X_3

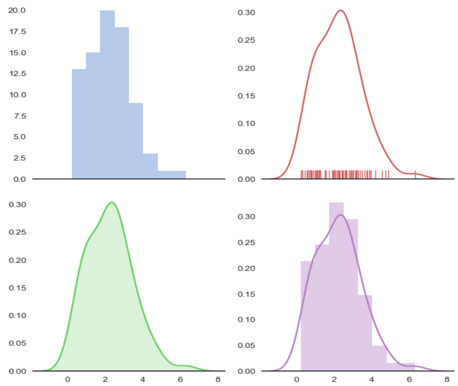

Fig. 4. Distribution of X_4

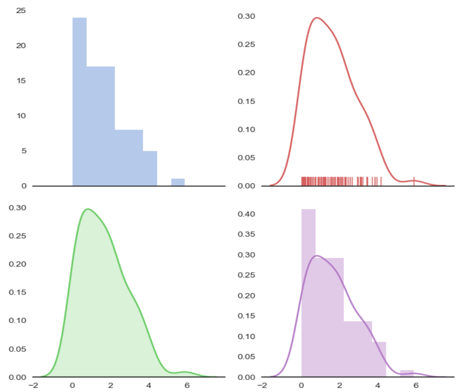

Fig. 5. Distribution of X_5

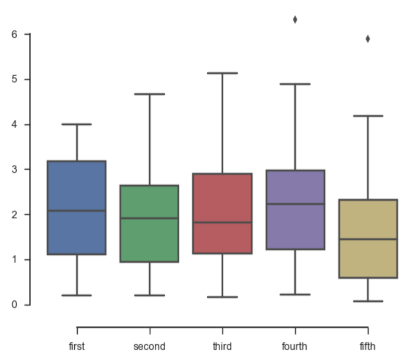

Fig. 6. Overall Distribution in one red envelop

Therefore, we guess that it follows:

$$X_j|X_1, X_2, ..., X_{j-1} \sim Unif(0, \frac{2}{6-i} \times (10 - \sum_{i=1}^{j-1} X_i)) \ (1 \leq j \leq 4)$$

$X_5 = 10 - \sum_{i=1}^{4} X_i$
It works well.
Then we generalize the distribution, for a n-*yuan* red envelop for m participators:

$$X_1 \sim Unif(0, \frac{2n}{m})$$
$$X_j|X_1, X_2, ..., X_{j-1} \sim Unif(0, \frac{2(n - \sum_{i=1}^{j-1} X_i)}{m+1-j})$$
$$2 \leq j \leq m-1$$
$$X_m = n - \sum_{i=1}^{m} X_i$$

III. Simulation Results

I use Python programming language for simulation:

```python
import random
import sys
f = open("output.csv","w")
def Rp(value,num):
  value_left = value
  result = []
  for i in range(num-1):
    now = random.uniform(0, (2/(num-i))*value_left)
    now = float('%.2f' %now)
    result.append(now)
    value_left = value_left - now
    value_left = float('%.2f' %value_left)
  result.append(float('%.2f' %value_left))
  result_str = ''
  for i in result:
    result_str = result_str + '%.2f,'%i
  result_str = result_str.rstrip(',')
  print(result_str, file = f)
```

I generate 100 cases for a 10-*yuan* red envelop for 5 participators. Results are below:

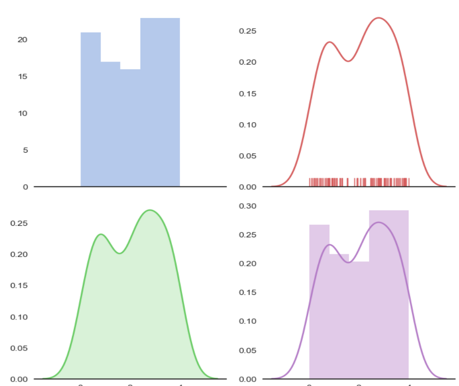

Fig. 7. Distribution of X_1

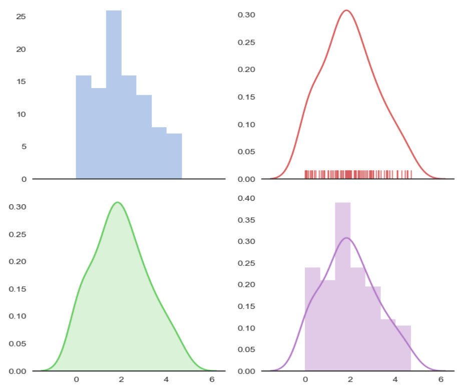

Fig. 8. Distribution of X_2

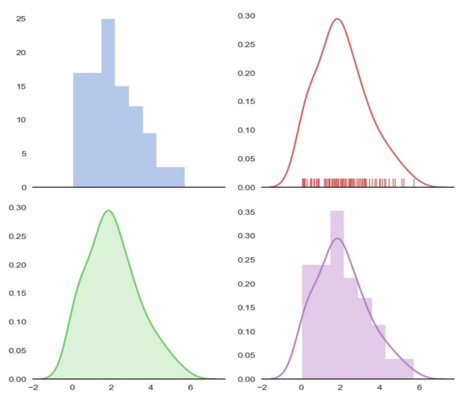

Fig. 9. Distribution of X_3

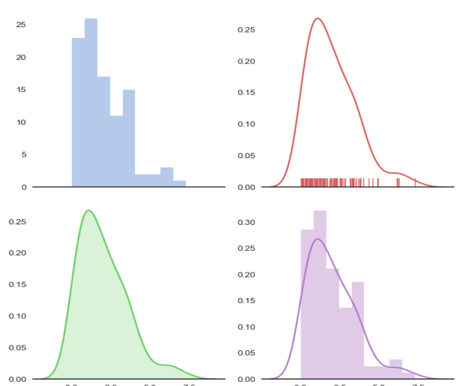

Fig. 10. Distribution of X_4

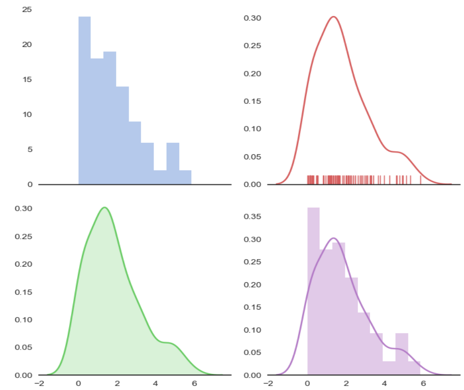

Fig. 11. Distribution of X_5

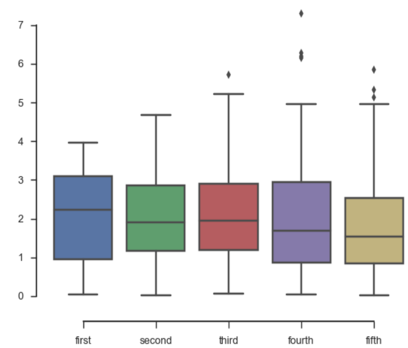

Fig. 12. Overall Distribution in one red envelop

IV. Result Analysis

In order to compare with my stimulation results with real ones, I do 20 more real trials combined with previous 80 trials as posterior verification.
The results are below:

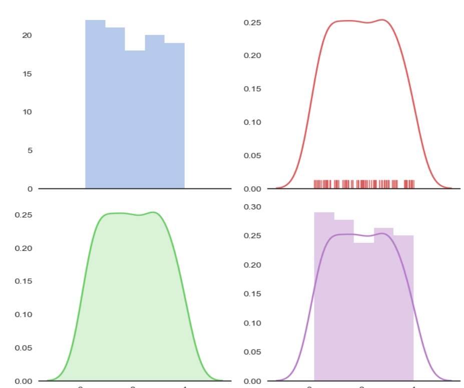

Fig. 13. Distribution of X_1

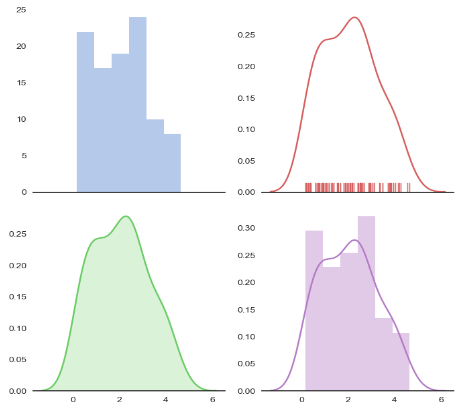

Fig. 14. Distribution of X_2

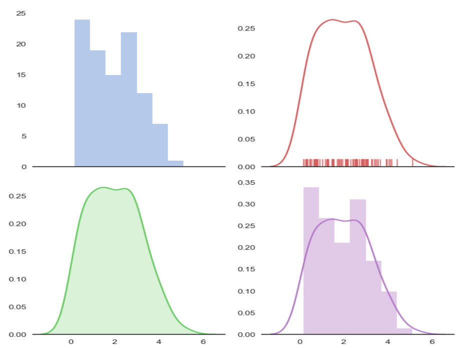

Fig. 15. Distribution of X_3

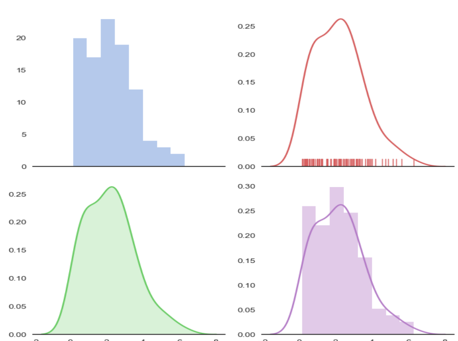

Fig. 16. Distribution of X_4

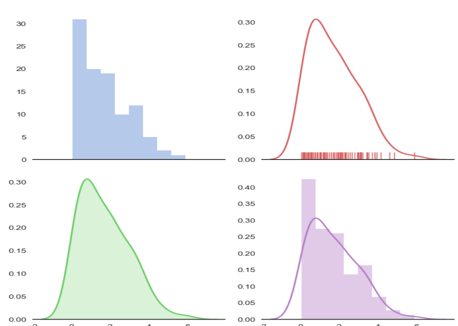

Fig. 17. Distribution of X_5

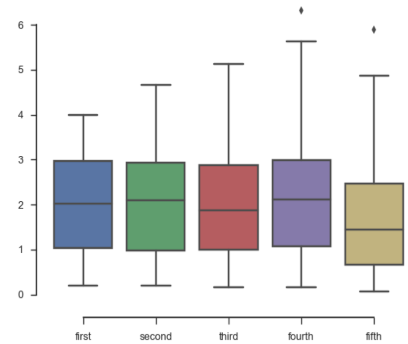

Fig. 18. Overall Distribution in one red envelop

Intuitively, my model satisfies real trials very well.
Assuming that my model is right, I conduct 1 million times stimulation trials. Since 1 million is large, we see it as the whole of the distribution. **Fig.19** below shows its overall mean and variance.

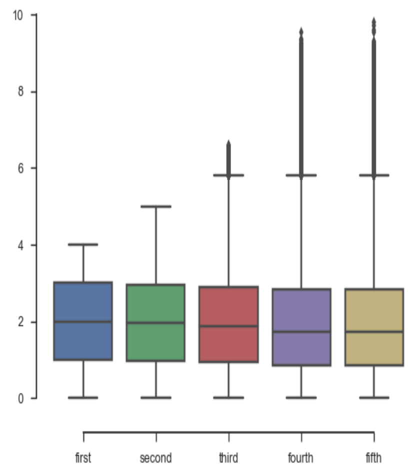

Fig. 19. Distribution of X_5

It could be seen that the mean of money every one gets is 2 with variance becoming larger and larger.
From *Standard Deviation of Probability Distribution*:

$$\sigma_X = \sqrt{Var(X)}$$

$$= \lim_{n\to\infty} \sqrt{\frac{1}{N}\sum_{i=1}^{n}(X_i - \mu)}$$

$$S = \sqrt{\frac{1}{N-1}\sum_{i=1}^{n}(X_i - \mu)}$$

Compare the 1 million stimulation model with real trials, we could see that the third, fourth and fifth participants suffer more unstable distributions, which meets our expectation from the theory *Standard Deviation of Probability Distribution*. We can also see that the sharps of the distributions are quite similar with the real ones. In all,it is a good model for red envelope distribution.

V. Conclusions

Just like my model, define X_j as the money the jth participant gets for a n-*yuan* red envelop for m participators. The distribution is:

$$X_1 \sim Unif(0, \frac{2n}{m})$$

$$X_j|X_1, X_2, ..., X_{j-1} \sim Unif(0, \frac{2(n - \sum_{i=1}^{j-1} X_i)}{m+1-j}$$

$$2 \leq j \leq m-1$$

$$X_m = n - \sum_{i=1}^{m} X_i$$

Furthermore, from our model, we can learn that mean $E(X_j)$ for each participator is the same. But variance $Var(X_j)$ gets larger with serial numbers.

Acknowledgment

During this project, I collaborated and discussed with my classmates Cheng'an Wang, Huifan Zhang, and Letong Wang. Data of real trials are collected by Cheng'an Wang, Huifan Zhang and me. I got inspiration of *Uniform Distribution* from Zhihu[1] and Zybuluo.com[2]. I think the idea *Uniform Distribution* is quite excellent, but the author does not choose a right domain for it. All the resources of this report have been pushed to my Github[3].

Reference

[1] https://www.zhihu.com/question/22625187/answer/85530416
[2] "Brief Introduction to Framework Design of WeChat Red Envelope" https://www.zybuluo.com/yulin718/note/93148
[3] https://github.com/KevinZhang199803/SI140_Final_Project.git