Test of a new feature launched by Riipen

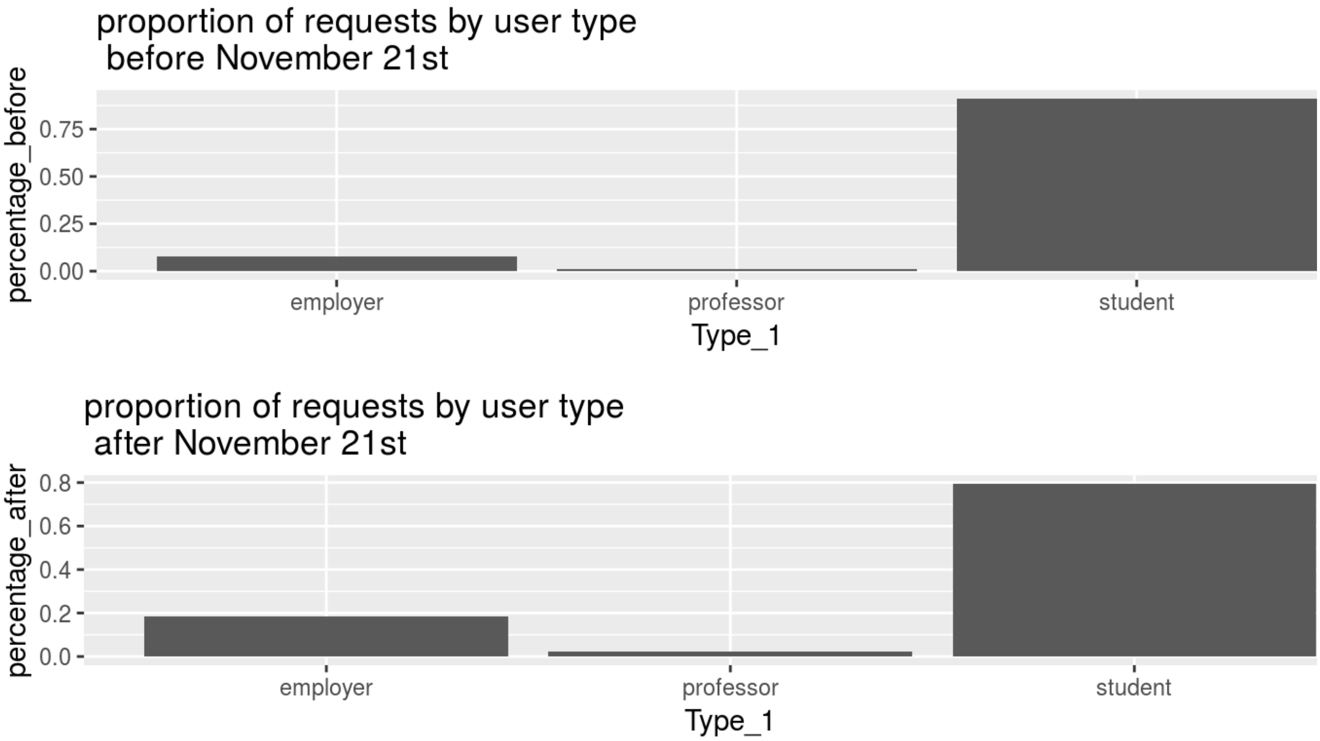
How the Launch of Request Expire Day Changes Users' Behavior

Yining Chen, Meng Sun, Xingyu Yu, Ruolan Zhang, T0105-3

Introduction

Riipen is an company which provides project-based learning opportunities for connect schools to employers. On November 21, 2018, Riipen launched a “request expiry” feature which requires users to respond to requests within 14 days, it will expire otherwise. This project is meant to test whether this new expiry feature would have any change on people’s behavior and response time.

Number of requests per user, by user type



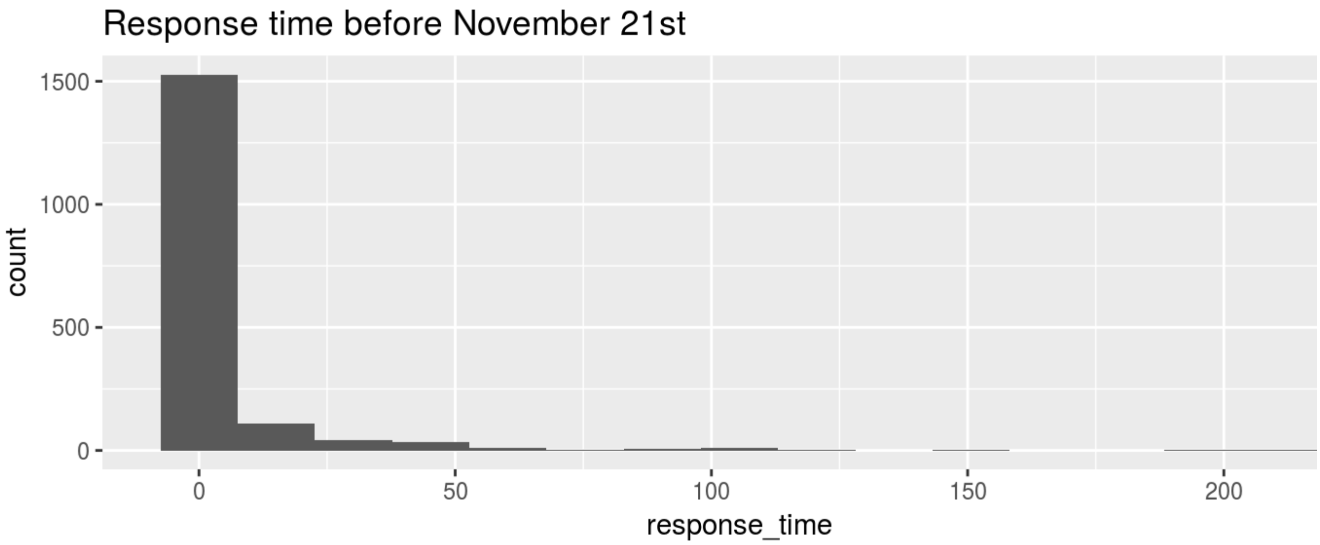
```
##             Before Nov 21st After Nov 21st
## employer          1.153846       1.379310
## professor         1.166667       1.111111
```

The mean requests per employer sent before the November 21 launch of request expiry was 1.153846. After the launch, the mean requests per employer sent has increased to 1.3793.

The mean requests per professor sent before the November 21 launch of request expiry was 1.1666667. After the launch, the mean requests per professor sent is 1.11111. The number of requests per professor drops approximately 0.05555.

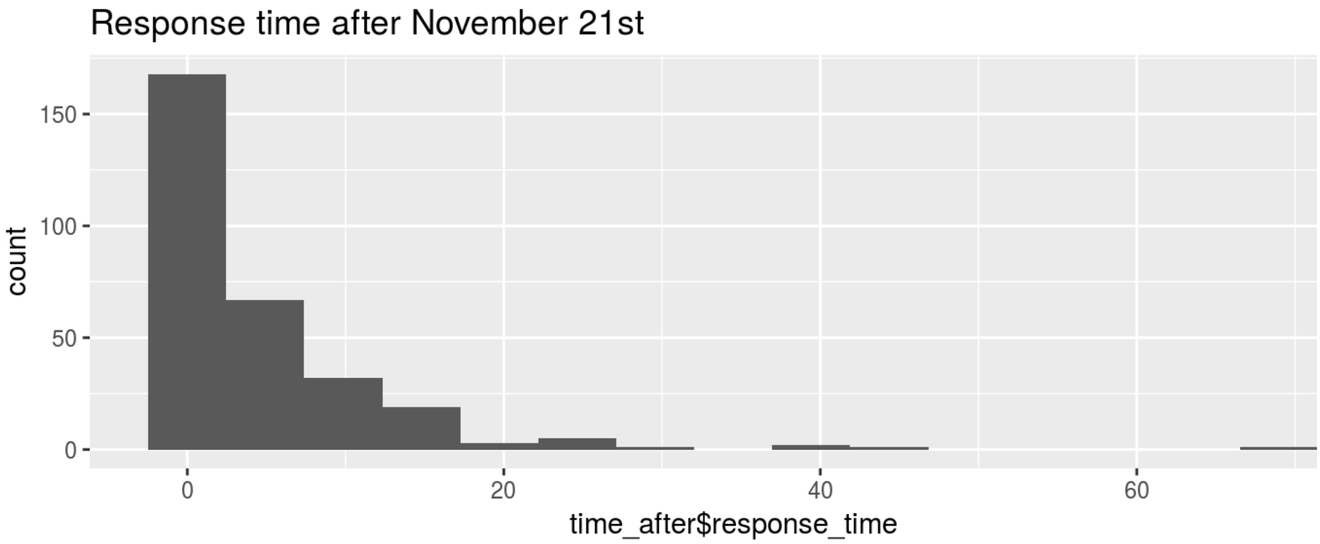
##request response time

#Before 11-21



This histogram collects data of response time before November 21st. We calculate response time by using update day minus created day. It centred at 0, which is right_skewed.

#After 11-21



This histogram collects data of response time after November 21st.It also centres at 0 and it is right-skewed. What can be observed is, there are less data which is larger than 0 comparing to last plot.

6/22

Data for request response time

```
## # A tibble: 1 x 5
##   mean          med        sd min    max
##   <time>        <time> <dbl> <time> <time>
## 1 4.976055 days 0 days  17.6 0 days 211 days
```

```
## # A tibble: 1 x 5
##   mean          med        sd min    max
##   <time>        <time> <dbl> <time> <time>
## 1 4.561873 days 1 days  7.62 0 days 69 days
```

#compare the mean

To compare the request time of the two different time periods, we can calculate and compare the mean and median. In this case, we chose to compare mean value of the two different datasets. The median of the data before the launch of request expiry is 0 because the graph of that dataset is very right-skewed. It is not fair to compare the two datasets using the median value since there are much more 0 value than the other values in the dataset. When we compare the mean value of the two data sets, we have found that the mean response time decreased from 4.976055 to 4.561873. In conclusion, from this table, we can see that the response time decreased as Riipen expected. However, we can not generate the conclusion simply from this table.In this case, we still need to use the method of hypothesis test.

Hypothesis test

Null Hypothesis(H0):The mean of response time of the two different time periods are the same.

H0:x1(mean1)-x2(mean2)=0

Alternative Hypothesis(Ha):The mean of response time of the two different time periods are different.

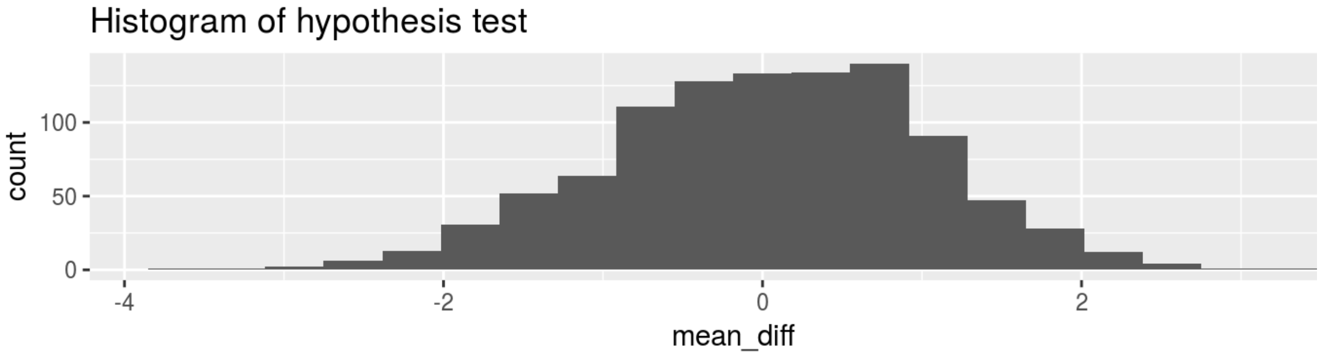
Ha:x1(mean1)-x2(mean2)≠0

parameter:The mean of response time (population = all requests)

Test Statistic: x1(mean1)- x2(mean2)=4.976055-4.561873=0.414182

#Simulate test statistic assuming H0 is true

In order to explore what we might expect to see if the mean value are not the same, we'll use simulation. We use simulation to randomly generate samples under the assumption that they have the same mean value. We'll do this many times to see what values are possible under the assumption of the same mean value.



#Histogram of hypothesis test

```
## [1] 0.693
```

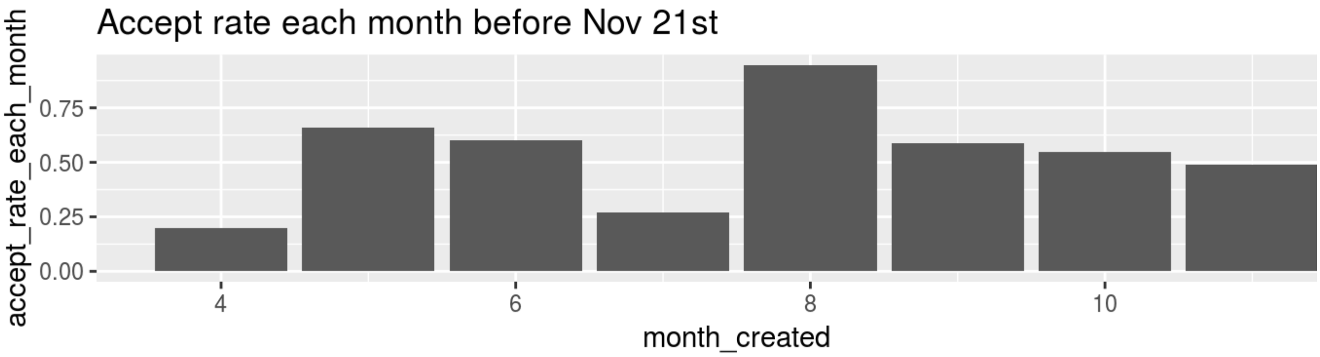
Then we assess the P-value: The p - value is the probability of observing data that are at least as unusual (or at least as extreme) as the sample data under the assumption that the null hypothesis is true. We have a P-value of 0.693, which is greater than 0.1, so there is no evidence against H0.

Question Conclusion

Since we have no evidence against H0, we can make the conclusion that mean of response time before November 21st and the mean of response time after November 21st is the same. Although there are some small differences in their mean value in our calculation, by using hypothesis test, we finally get the conclusion that the two response time are almost the same. Consequently, the feature of expiry is useless.

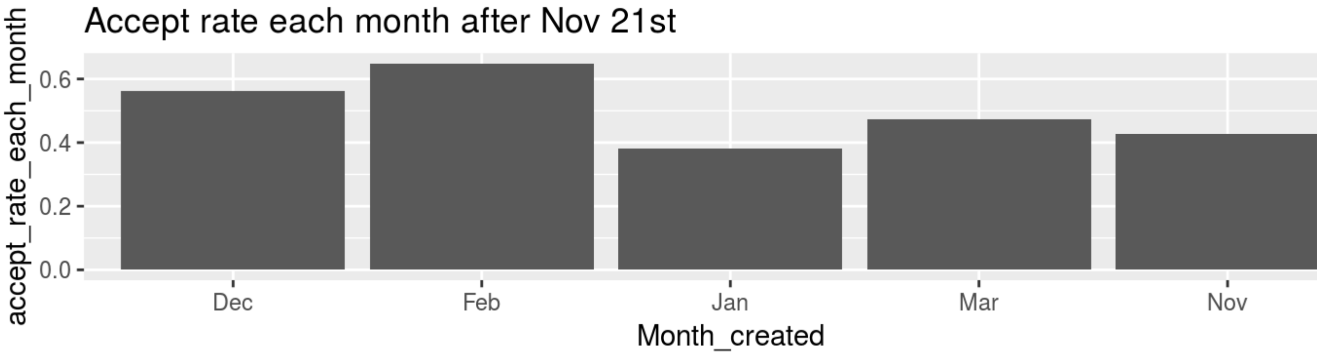
##Request acceptance rate

#Before Nov 21st



The graph indicates the acceptance rate each month before Nov 21st. The x-axis is the month of when the request was created. The y-axis is the acceptance rate of each month. In general, the acceptance rate fluctuate over time. In July, the acceptance rate was the lowest, which was approximately 0.25. And the highest acceptance rate occurred in August, which was approximately 0.95. The lowest and highest rate has a large difference which is 0.7 approximately, even in just one month.

#After Nov 21st



The graph indicates the acceptance rate each month after Nov 21st. The x-axis is the month of when the request was created. The y-axis is the acceptance rate of each month. In general, the acceptance rate fluctuate as well, but not as large as the rate before Nov.21. In January, the acceptance rate was the lowest, which was approximately 0.39. And the highest acceptance rate occurred in February, which was approximately 0.63. The difference of the lowest and highest rate is approximately 0.24. It also happens within one month, but the difference is not as big as the difference of acceptance rate before Nov.21.

#overall acceptance rate

```
## Observations: 1
## Variables: 1
## $ n <dbl> 0.7651367
```

```
## Observations: 1
## Variables: 1
## $ n <dbl> 0.4895397
```

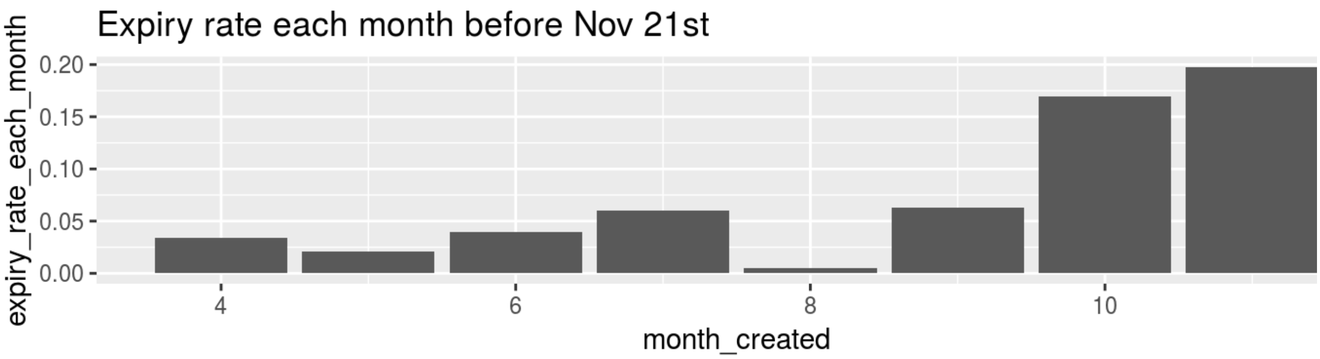
Before acceptance rate:0.7651367

After acceptance rate:0.4895397

The acceptance rate decreases, less users make response within available days. Therefore people do not increase their response rate.

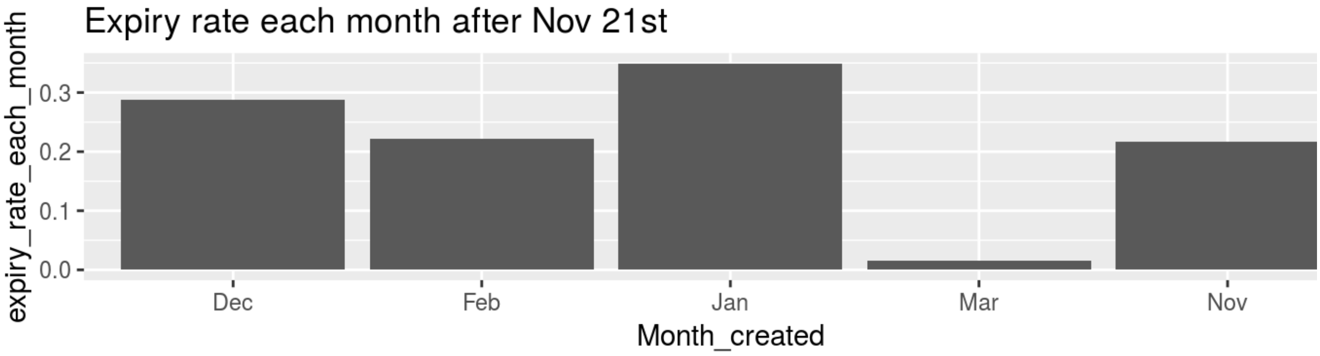
##request expiry rate

#Before Nov 21st



The graph indicates the expiry rate each month before Nov 21st. The x-axis is the month of when the request was made. The y-axis is the expiry rate of each month. In general, the expiry rate was increasing over time, because users had less and less time to respond to the requests. In August, the expiry rate was the lowest. And the highest expiry rate occurred in November, which was approximately 0.20.

#After Nov 21st



The graph indicates the expiry rate each month after Nov 21st. The x-axis is the month of when the request was made. The y-axis is the expiry rate of each month. In general, the expiry rate first decreased and then increased. The lowest expiry rate occurred in January, and the highest expiry rate was still in November, particularly after Nov 21st, approximately 0.35. However, the expiry rate was generally higher than that of each month before the launch of request expiry. That makes sense, because users had at most 8 months to give a response before the launch, but only had 14 days to respond after the launch.

#overall rate

```
## Observations: 1
## Variables: 1
## $ n <dbl> 0.02783203
```

```
## Observations: 1
## Variables: 1
## $ n <dbl> 0.2426778
```

Before expiry rate: 0.02783203

After expiry rate: 0.2426778

The expiry rate increases, less users make response within available days. Therefore people do not increase their response rate.

Statistical Methods

We used histogram, bar chart, table and hypothesis test to analyze our data.

To find out number of requests per user by user type, we used bar charts to analyze. We draw two different bar charts. The first bar chart has the number of requests by user type before Nov.21 as y-axis and three groups of people such as employer, professor and student on the x-axis. The second bar chart has the number of requests by user type after Nov.21 as y-axis and three groups of people such as employer, professor and student on the x-axis. We also made a table to show the mean requests per employer and professor sent before and after Nov.21.

We use histogram to count the number of response time before and after Nov.21, the histogram has response time as x-axis, and number of count as y-axis.

#Statistical Methods

We also use hypothesis test to help us draw the conclusion if the mean values are the same or not. We use histograms to show the result of simulations.

We use bar chart to analyze the question of the request acceptance rate. The bar chart has the months as x-axis, and the acceptance rate as y-axis. Similarly, we use bar chart to analyze the question of the request expire rate. The bar chart has the months as x-axis, and the expire rate as y-axis.

Results

The hypothesis we used in question 3 can reflect the main result of the project. After hypothesis testing, we calculated a p-value under the assumption that users will not respond more quickly or send more requests after the launch of request expiry. Since the estimated p-value is 0.693, we conclude that we have no evidence against the null hypothesis. The data provided convincing evidence that the Nov 21 launch of request expiry does not have efficient effect on the user's behavior to respond more quickly or send more requests. However, there is a potential of making a Type II Error since we rejected the null hypothesis, but null hypothesis may still not be actually true.

Conclusion

In conclusion, there is an observable increase in number of requests sent per employer before and after Nov.21, but a small decrease in number of requests sent per professor before and after Nov.21. Acceptance rate decreases, and the expire rate increases. In this case, less users response within available time period, therefore they do not increase their response rate. There is not much difference of response time before and after Nov.21 according to our histograms and hypothesis test. Furthermore, we can draw the conclusion the method does not have an efficient effect. There is a limitation in our project, that is we do not filter out two outliers in question 3 when calculating p-value. By using codes, we find that 2018-4-12 and 2018-8-30 are two outliers in requests dataframe. More than hundards of requests are received from people. Since we do not remove these two data, it may have some influence on the p-value we calculated. Another concern is, from question 1, after applying the expiry feature, in average each professor send less requests than before.