Tuango RFM HW

Rainy Chen

2/21/2022

Q1.What percentage of customers responded (i.e. bought anything) after the push message?

```
CrossTable(TG$buyer, digits=4)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  13939
##
##
##            |         0 |         1 |
##            |-----------|-----------|
##            |     13507 |       432 |
##            |    0.9690 |    0.0310 |
##            |-----------|-----------|
##
##
##
##
```

3.1% of customers responded after the push message.

Q2.Of those who bought, what was the average spending?

```
mean(TG$ordersize[TG$buyer==1])
```

```
## [1] 202.3565
```

Average spending is about 202 RMB.

Q3.Create quintile variables for recency, frequency and monetary.

```
TG_Copy <- TG %>%
  mutate(rec_quin=xtile(recency, 5),
         freq_quin=xtile(frequency, 5),
         mv_quin=xtile(monetary, 5))
```

```
head(TG_Copy %>% select(userid, buyer, recency, frequency,
                    monetary, rec_quin, freq_quin, mv_quin))
```

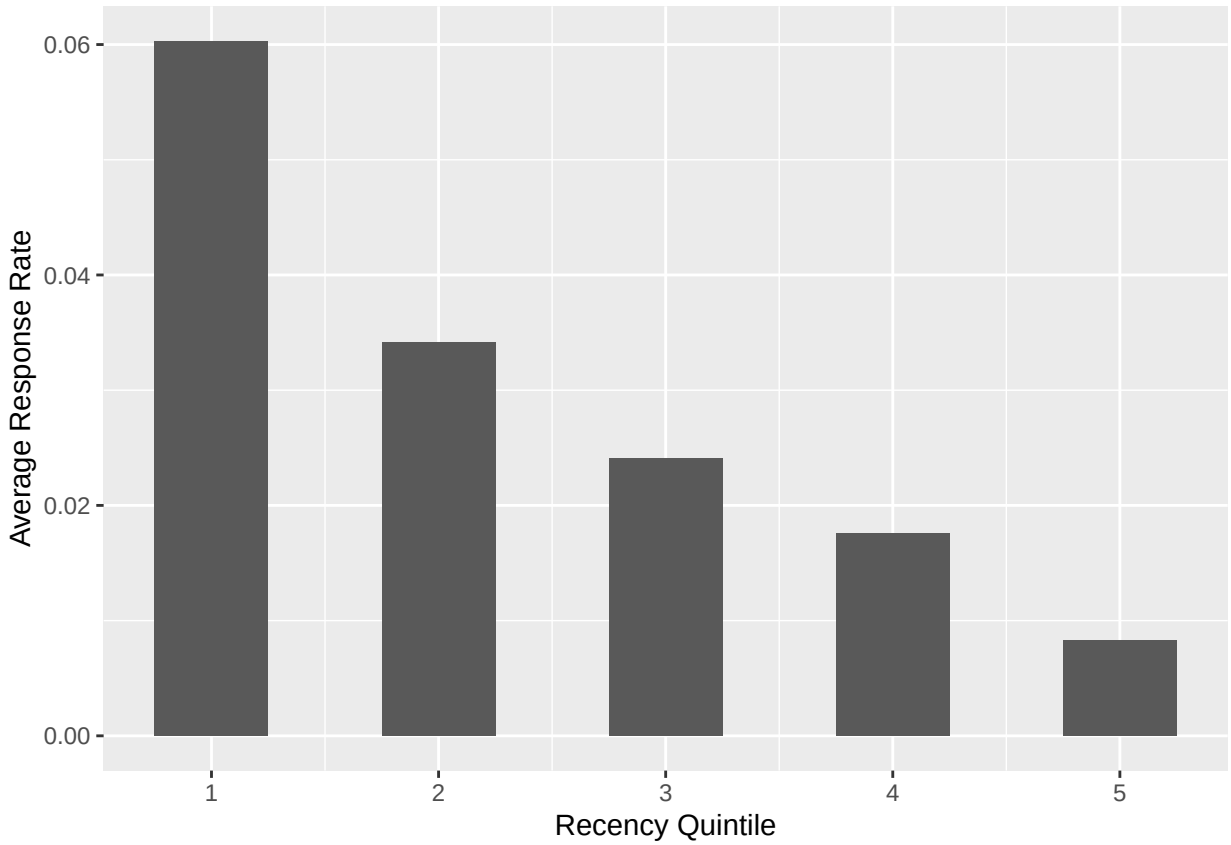
```
##     userid buyer recency frequency monetary rec_quin freq_quin mv_quin
## 1 63775658     0     309         7     39.8        5         5       3
## 2 64880613     0     297         8     39.8        5         5       3
## 3 65051746     0     295         1     72.9        5         1       4
## 4 66689882     0     277         1     40.0        5         1       3
## 5 68839217     0     259         1     21.0        5         1       2
## 6 70630920     0     243         1     19.9        5         1       2
```

Q4.Create a bar chart showing the response rate (i.e., the proportion of customers who bought something) to this deal by recency quintile.

```
avg_resp_rate_rec <- TG_Copy %>% group_by(rec_quin) %>%
  summarise(avg_resp_rate=mean(buyer), .groups="drop")
avg_resp_rate_rec
```

```
## # A tibble: 5 x 2
##   rec_quin avg_resp_rate
##      <int>         <dbl>
## 1        1       0.0603
## 2        2       0.0342
## 3        3       0.0241
## 4        4       0.0176
## 5        5       0.00827
```

```
bar_avg_resp_rate_rec <-
  ggplot(data=avg_resp_rate_rec,
         aes(x = rec_quin, y = avg_resp_rate)) +
  labs(x="Recency Quintile", y="Average Response Rate") +
  geom_bar(stat="identity", width=0.5)
bar_avg_resp_rate_rec
```

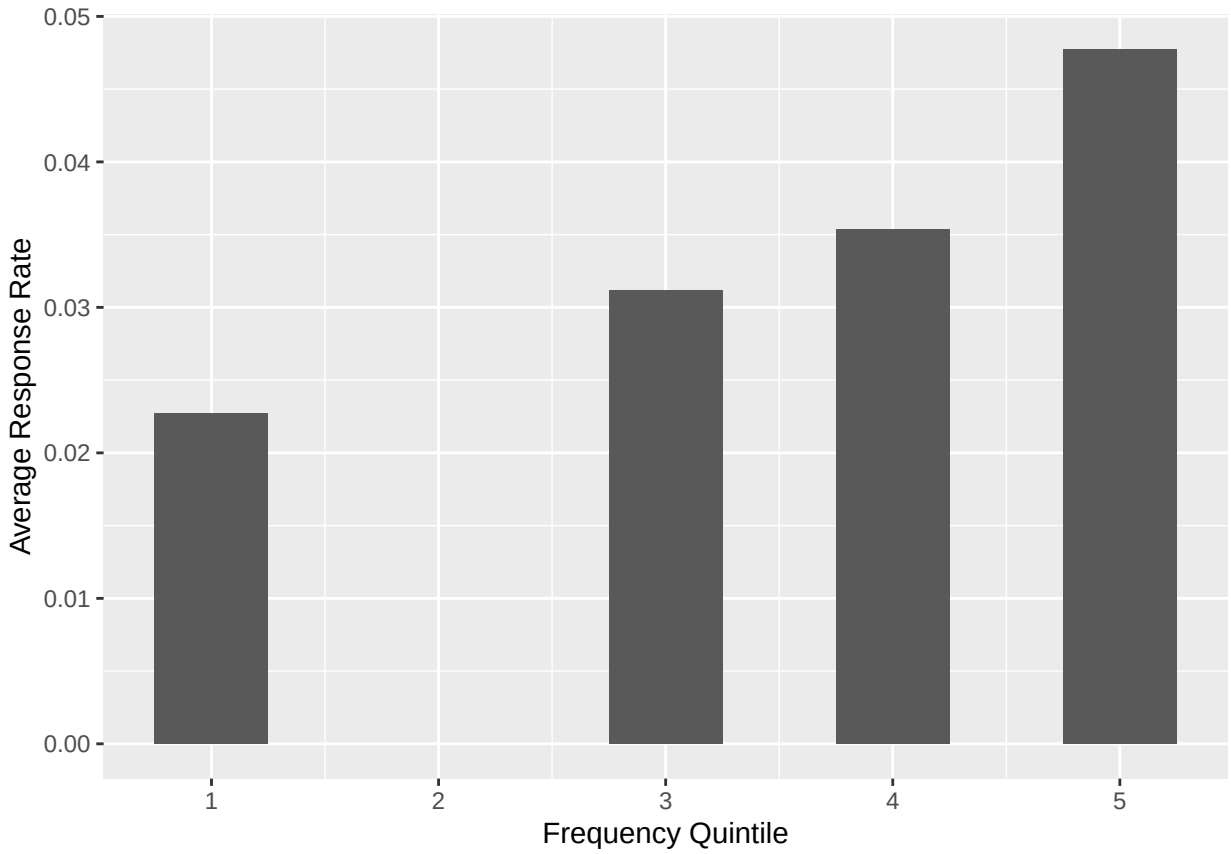


Q5.Show the response rate to this deal by frequency quintile.

```
avg_resp_rate_freq <- TG_Copy %>%
  group_by(freq_quin) %>%
  summarise(avg_resp_rate=mean(buyer), .groups="drop")
avg_resp_rate_freq
```

```
## # A tibble: 4 x 2
##   freq_quin avg_resp_rate
##       <int>         <dbl>
## 1         1        0.0227
## 2         3        0.0311
## 3         4        0.0354
## 4         5        0.0478
```

```
bar_avg_resp_rate_freq <-
  ggplot(data=avg_resp_rate_freq,
         aes(x = freq_quin, y = avg_resp_rate)) +
  labs(x="Frequency Quintile", y="Average Response Rate") +
  geom_bar(stat="identity", width=0.5)
bar_avg_resp_rate_freq
```

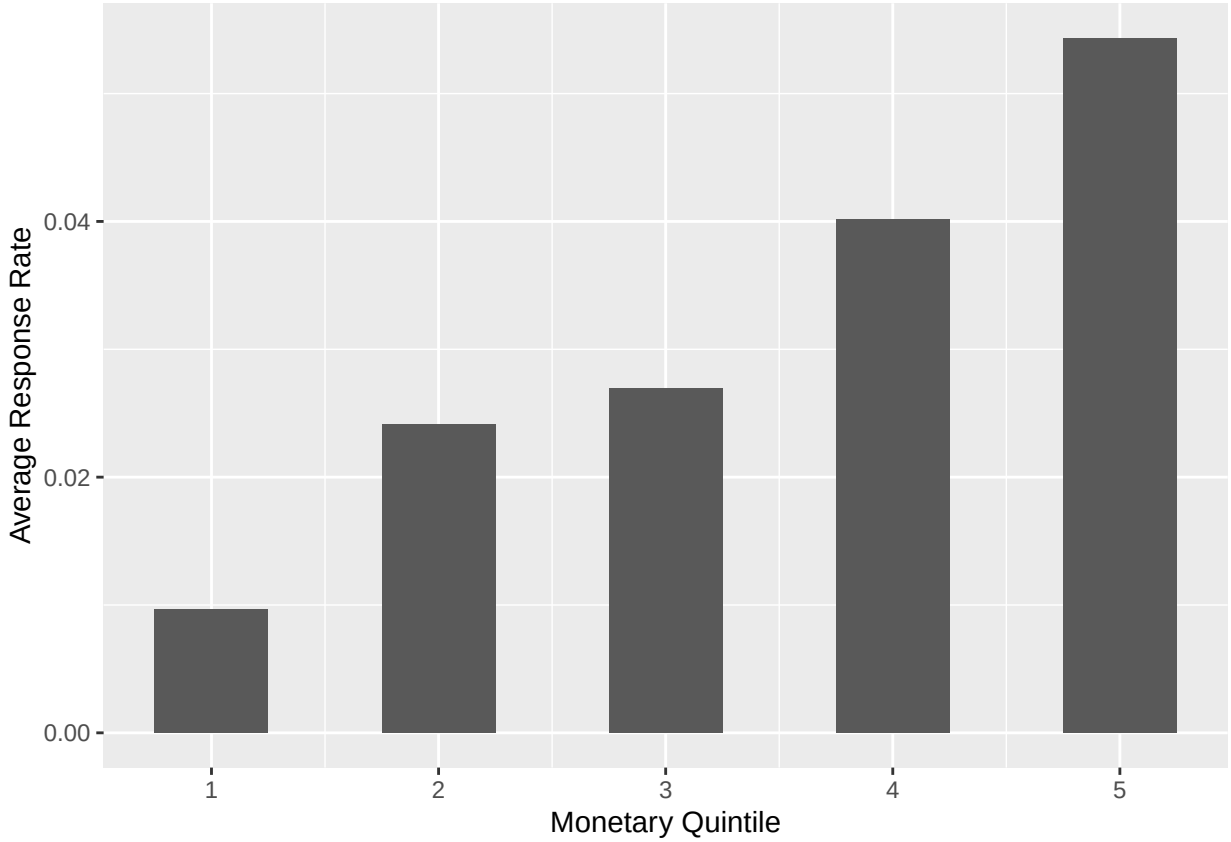


Q6.Show the response rate to this deal by monetary quintile.

```
avg_resp_rate_mv <- TG_Copy %>% group_by(mv_quin) %>%
  summarise(avg_resp_rate=mean(buyer), .groups="drop")
avg_resp_rate_mv
```

```
## # A tibble: 5 x 2
##   mv_quin avg_resp_rate
##     <int>         <dbl>
## 1       1       0.00965
## 2       2       0.0242
## 3       3       0.0269
## 4       4       0.0401
## 5       5       0.0543
```

```
# The bar chart
bar_avg_resp_rate_mv <-
  ggplot(data=avg_resp_rate_mv,
         aes(x = mv_quin, y = avg_resp_rate)) +
  labs(x="Monetary Quintile", y="Average Response Rate") +
  geom_bar(stat="identity", width=0.5)
bar_avg_resp_rate_mv
```

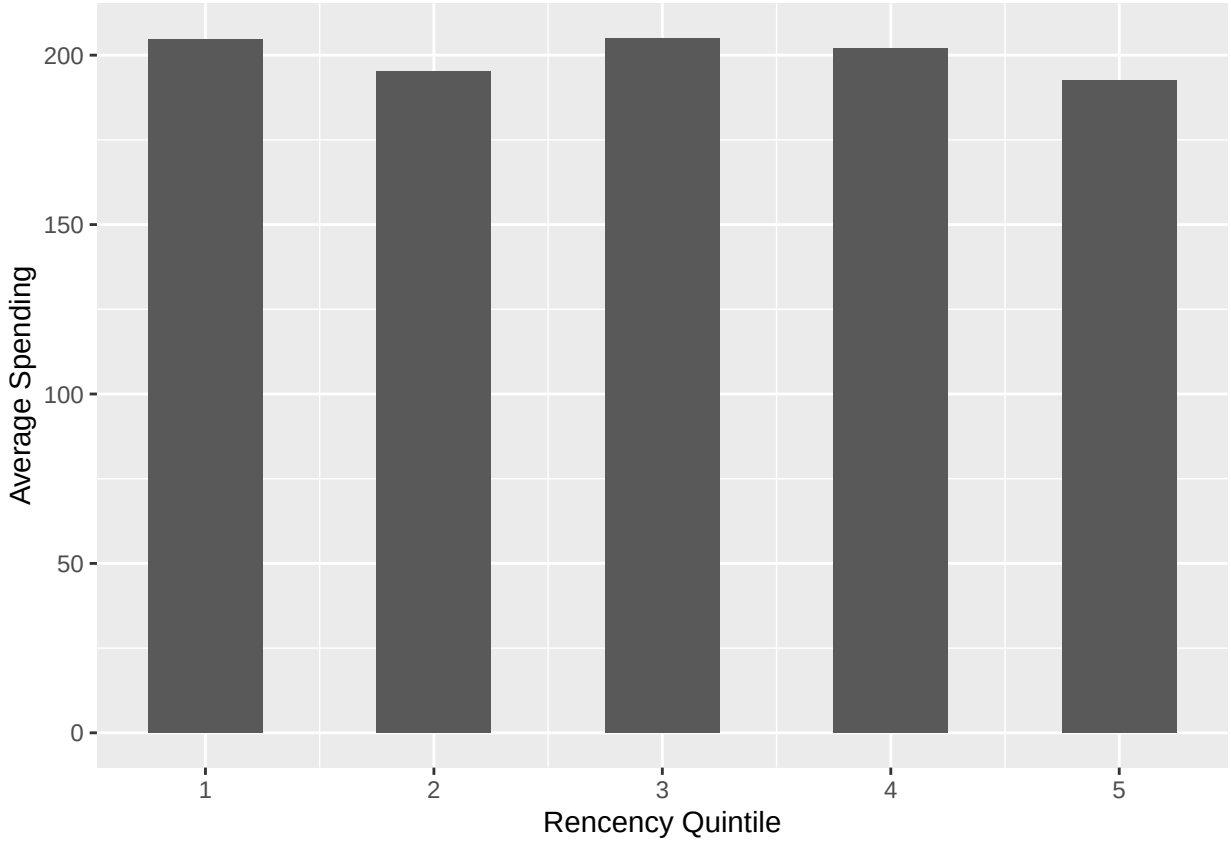


Q7.Repeat questions 4-6 using only those customers who placed an order after the push message, i.e. create bar charts showing the average spending (in RMB) spent by recency, frequency and monetary quintile.

```
#recency
avg_spend_rec <- TG_Copy %>% group_by(rec_quin) %>%
  summarise(avg_spend=mean(ordersize[buyer == 1]), .groups="drop")
avg_spend_rec
```

```
## # A tibble: 5 x 2
##   rec_quin avg_spend
##      <int>     <dbl>
## 1        1      205.
## 2        2      195.
## 3        3      205.
## 4        4      202.
## 5        5      192.
```

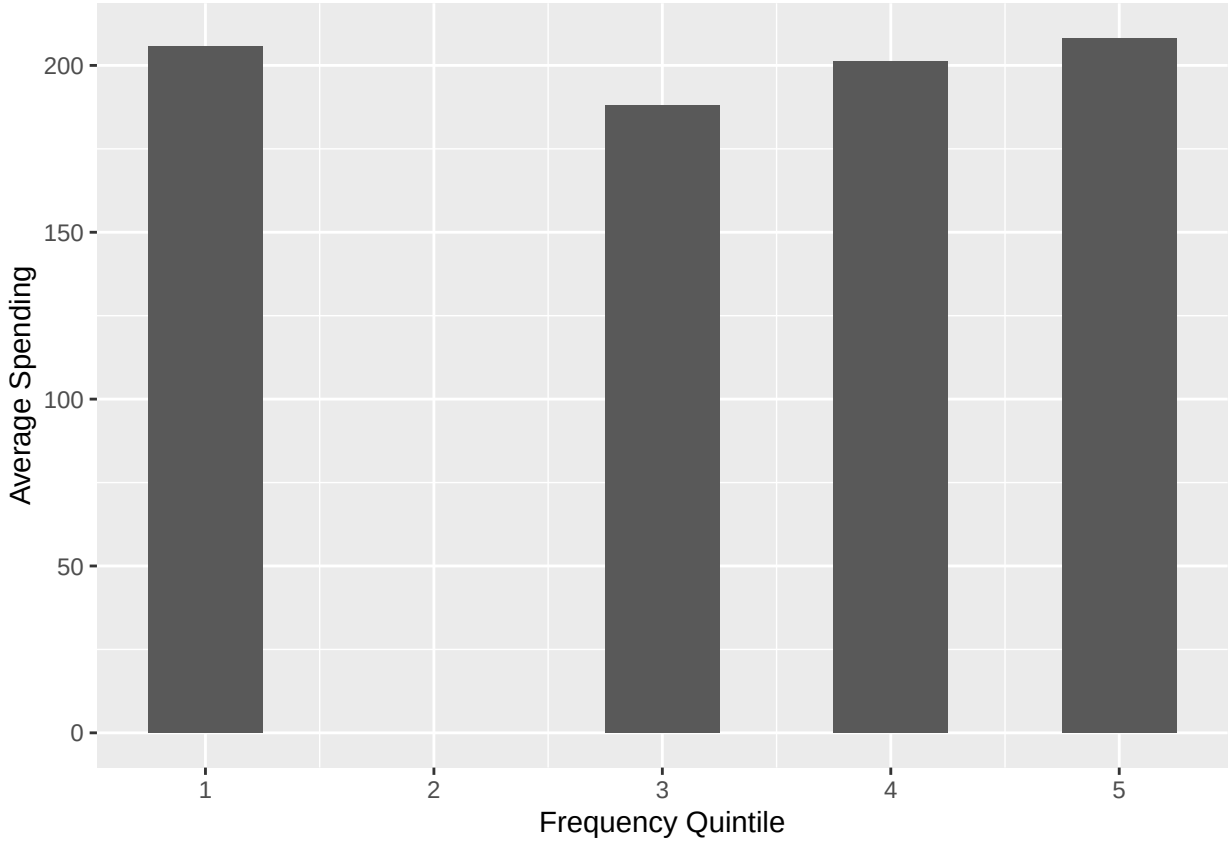
```
#bar chart
bar_avg_spend_rec <-
  ggplot(data=avg_spend_rec,
         aes(x = rec_quin, y = avg_spend)) +
  labs(x="Rencency Quintile", y="Average Spending") +
  geom_bar(stat="identity", width=0.5)
bar_avg_spend_rec
```




```
#frequency
avg_spend_freq <- TG_Copy %>%
  group_by(freq_quin) %>%
  summarise(avg_spend=mean(ordersize[buyer == 1]), .groups="drop")
avg_spend_freq
```

```
## # A tibble: 4 x 2
##   freq_quin avg_spend
##       <int>     <dbl>
## 1         1      206.
## 2         3      188.
## 3         4      201.
## 4         5      208.
```

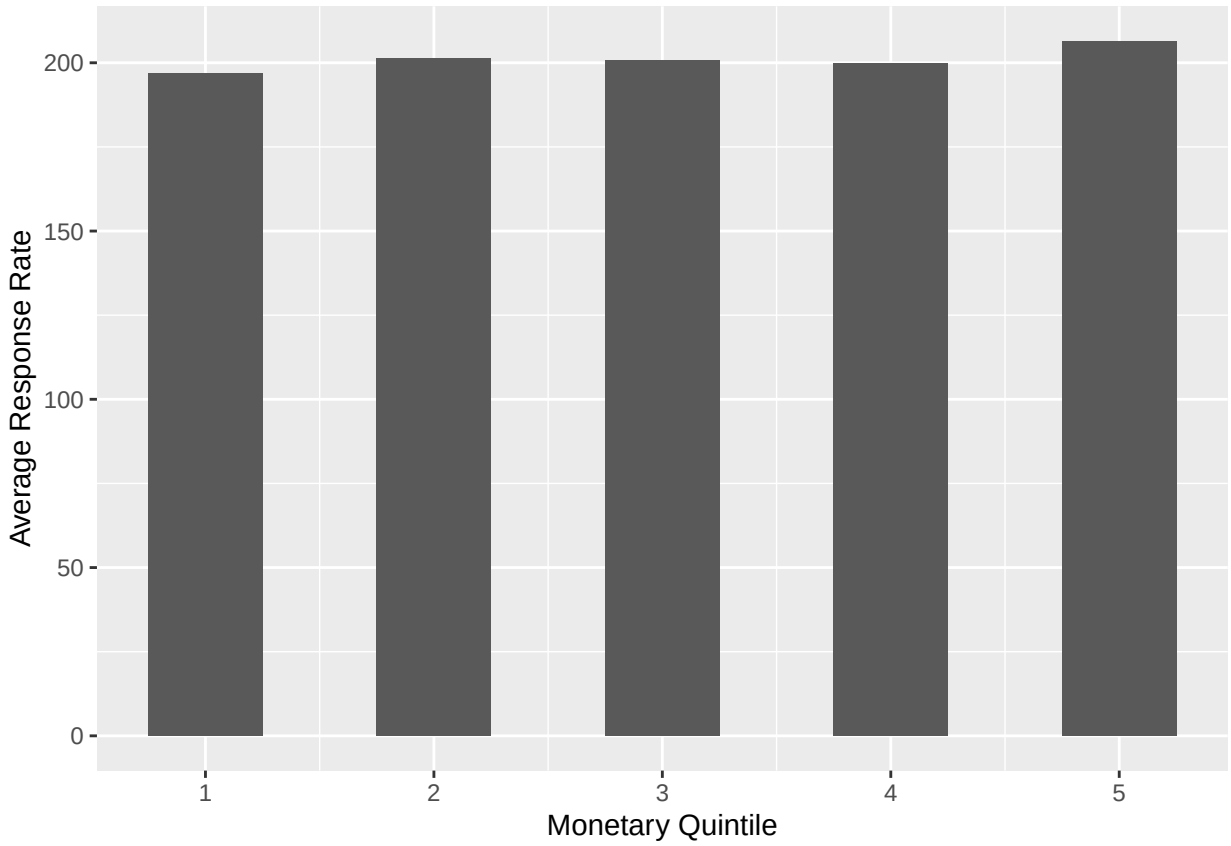
```
#bar chart
bar_avg_spend_freq <-
  ggplot(data=avg_spend_freq,
         aes(x = freq_quin, y = avg_spend)) +
  labs(x="Frequency Quintile", y="Average Spending") +
  geom_bar(stat="identity", width=0.5)
bar_avg_spend_freq
```




```
#monetary
avg_spend_mv <- TG_Copy %>% group_by(mv_quin) %>%
  summarise(avg_resp_rate=mean(ordersize[buyer == 1]), .groups="drop")
avg_spend_mv
```

```
## # A tibble: 5 x 2
##   mv_quin avg_resp_rate
##     <int>         <dbl>
## 1       1          197.
## 2       2          201.
## 3       3          201.
## 4       4          200.
## 5       5          206.
```

```
#bar chart
bar_avg_spend_mv <-
  ggplot(data=avg_spend_mv,
         aes(x = mv_quin, y = avg_resp_rate)) +
  labs(x="Monetary Quintile", y="Average Response Rate") +
  geom_bar(stat="identity", width=0.5)
bar_avg_spend_mv
```



Q8. What do the above bar charts reveal about the likelihood of response and the size of the order across the different recency, frequency, and monetary quintiles?

Overall, the more recent, frequent purchases, and more average spending per deal a customer has, the more it responds. For customers who bought the deal, there is no clear trend on RFM.

Q9.What is the breakeven response rate?

```
#marginal cost/average sales revenue = average break-even response rate
breakeven=1.6/(0.5*mean(TG$ordersize[TG$buyer==1]))
breakeven
```

```
## [1] 0.01581368
```

Average break-even response rate is 1.6%.

Q10.What is the projected (a) profit in RMB, (b) return on marketing expenditures if you offer the deal to all remaining 264,841 customers.

```
#(a) profit=[(total customer*average response rate)*average spending]*50% of sales revenue as fee - mar
profit=264841*0.031*202.3565*0.5-1.6*264841
profit
```

```
## [1] 406935
```

```
#(b) return on marketing cost=profit/(marginal cost*total customer)
return=profit/(1.6*264841)
```

```
return
```

```
## [1] 0.9603286
```

Q11. Consider offering the deal only to those of the 264,841 customers in RFM cells (using the sequential n-tiles approach, coded as the rfm1 variable) with a response rate that is equal to or greater than the breakeven response rate.

```
#number of customers belonging to profitable cells
TG_Copy <- TG_Copy %>%
  group_by(rfm1) %>%
  mutate(avg_resp_rate_rfm=mean(buyer)) %>% ungroup()
TG_Copy <- TG_Copy %>%  mutate(pushto = avg_resp_rate_rfm > breakeven)

rfm1_group_size <- TG_Copy[TG_Copy$pushto,] %>% group_by(rfm1) %>%
  summarise(group_size_rfm1 = n(), .groups="drop")

#number of buyers belonging to profitable cells
rfm1_buyer_size <- TG_Copy[TG_Copy$pushto&TG_Copy$buyer==1,] %>% group_by(rfm1) %>%
  summarise(buyer_size_rfm1 = n(), avg_revenue_rfm1 = mean(ordersize))
```

```
# profit
rfm1_profit <- left_join(rfm1_group_size, rfm1_buyer_size)
```

```
## Joining, by = "rfm1"
```

```
# Calculated using average revenue in each rfm group
rfm1_profit$profit_rfm1 = (rfm1_profit$avg_revenue_rfm1 * rfm1_profit$buyer_size_rfm1 * 0.5 - 1.6 * rfm
rfm1_profit$return_rfm1 = rfm1_profit$profit_rfm1 / (1.6 * rfm1_profit$group_size_rfm1)

rfm1_profit[,c("rfm1", "profit_rfm1", "return_rfm1")]
```

```
## # A tibble: 59 x 3
##     rfm1 profit_rfm1 return_rfm1
##    <int>       <dbl>       <dbl>
##  1   111      42393.       199.
##  2   112      19882.        91.4
##  3   113      18968.        89.8
##  4   114       9570.        45.0
##  5   115       4942.        22.2
##  6   121      24520.       106.
##  7   122      24307.        99.9
##  8   123      15363.        62.4
##  9   124      16650.        68.5
## 10   125       7452.        30.4
## # ... with 49 more rows
```

```
sum(rfm1_profit$profit_rfm1) - profit
```

```
## [1] 116883.6
```

Q12.What do you notice about the rfm1 and rfm2 values? That is – do the two approaches generally yield the same RFM index for any given customer? What do you see as the pros and cons of the two approaches

```
TG_Copy <- TG_Copy %>% mutate(same_rfm = rfm1==rfm2)
CrossTable(TG_Copy$same_rfm)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  13939
##
##
##            |     FALSE |      TRUE |
##            |-----------|-----------|
##            |      1779 |     12160 |
##            |     0.128 |     0.872 |
##            |-----------|-----------|
##
##
##
##
```

There are about 87.2% of same RFM index. Sequential RFM pros is that it provides a more even distribution RFM group; cons is that the index is hard to interpret since frequency and monetary rank dependent on recency rank. Independent RFM pros is that it is easy to interpret, but may provide empty bin because of uneven distribution.