Marketing Assignment 1

2024-02-28

```r
#Loading the dataframe
library(readr)
Data<-read_csv('TuscanDataForRFMAnalysis.csv')
```

```
## New names:
## Rows: 96551 Columns: 7
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): buyer dbl (6): ...1, numords, totdol, last, buyerdummy, dollars
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```r
df<-Data
head(df)
```

```
## # A tibble: 6 x 7
##    ...1 numords totdol  last buyer buyerdummy dollars
##   <dbl>   <dbl>  <dbl> <dbl> <chr>      <dbl>   <dbl>
## 1     1       7    493   207 no             0       0
## 2     2       4    423   625 no             0       0
## 3     3       4    246    28 no             0       0
## 4     4       3    271   778 no             0       0
## 5     5       2    148   396 no             0       0
## 6     6      10    937     6 no             0       0
```

*Q*1.

```r
library(tidyverse)
library(dplyr)
```

```r
#Filtering the dataframe for customers buying from test catalog
test_cat<-filter(df, buyerdummy==1)
non_test_cat<-filter(df,buyerdummy==0)
#Percent of total customers who bought from test catalog
percentage_buyers_catalog<-(dim(test_cat)[1]/dim(df)[1])*100
percentage_buyers_catalog
```

```
## [1] 2.455697
```

It was observed that there are about 2.456% of customers who purchase from the test catalog, implying that 2.456% of customers from all the customers mailed to will purchase from the test catalog.

```r
#Summary Statistics of total dollar value of past purchases for customers buying
#in test catalog
summary_tot_spent<-summary(test_cat$totdol)
summary_tot_spent
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     8.0   125.5   276.0   518.1   593.0  9858.0
```

```
sd_tot_spent<-sd(test_cat$totdol)
sd_tot_spent
```

```
## [1] 757.9626
```

```
mean_tot_spent<-mean(test_cat$totdol)
mean_tot_spent
```

```
## [1] 518.1084
```

```
#Summary statistics of spending by customers buying from test catalog using
#the test catalog
summary_spent_test_cat<-summary(test_cat$dollars)
summary_spent_test_cat
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     5.0    40.0    75.0   104.2   130.0  6249.0
```

```
sd_spent_test_cat<-sd(test_cat$dollars)
sd_spent_test_cat
```

```
## [1] 157.0009
```

```
mean_spent_test_cat<-mean(test_cat$dollars)
mean_spent_test_cat
```

```
## [1] 104.2429
```

```
#Summary Statistics of total dollar value of past purchases for customers not
#buying in test catalog
summary_tot_spent_non<-summary(non_test_cat$totdol)
summary_tot_spent_non
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       2      89     183     332     373   21316
```

```
sd_tot_spent_non<-sd(non_test_cat$totdol)
sd_tot_spent_non
```

```
## [1] 523.2939
```

```
mean_tot_spent_non<-mean(non_test_cat$totdol)
mean_tot_spent_non
```

```
## [1] 331.9912
```

```
#Summary statistics of spending by customers not buying from test catalog
#using the test catalog
summary_spent_test_cat_non<-summary(non_test_cat$dollars)
summary_spent_test_cat_non
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       0       0       0       0       0
```

```
sd_spent_test_cat_non<-sd(non_test_cat$dollars)
sd_spent_test_cat_non
```

```
## [1] 0
```

```r
mean_spent_test_cat_non<-mean(non_test_cat$dollars)
mean_spent_test_cat_non
```

```
## [1] 0
```

```r
#Summary Statistics of total dollar value of past purchases for customers
summary_tot_spent_all<-summary(df$totdol)
summary_tot_spent_all
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     2.0    89.0   184.0   336.6   377.0 21316.0
```

```r
sd_tot_spent_all<-sd(df$totdol)
sd_tot_spent_all
```

```
## [1] 531.0781
```

```r
mean_tot_spent_all<-mean(df$totdol)
mean_tot_spent_all
```

```
## [1] 336.5616
```

```r
#Summary statistics of spending by customers
summary_spent_test_cat_all<-summary(df$dollars)
summary_spent_test_cat_all
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00    0.00    2.56    0.00 6249.00
```

```r
sd_spent_test_cat_all<-sd(df$dollars)
sd_spent_test_cat_all
```

```
## [1] 29.41705
```

```r
mean_spent_test_cat_all<-mean(df$dollars)
mean_spent_test_cat_all
```

```
## [1] 2.559891
```

Q2.

```r
#Creating R, F, and M quintiles
df$r_quin <-.bincode(df$last, quantile(df$last, probs = seq(0, 1, 0.2)),
                     right = TRUE,include.lowest = TRUE)
df$f_quin<-.bincode(df$numords, quantile(df$numords, probs = seq(0, 1, 0.2)),
                    right = TRUE,include.lowest = TRUE)
df$m_quin<-.bincode(df$totdol, quantile(df$totdol, probs = seq(0, 1, 0.2)),
                    right = TRUE,include.lowest = TRUE)
#Selecting the necessary columns
df_selected<-select(df,last,r_quin,numords,f_quin,totdol,m_quin)
#Showing the 5 selected observations information about R, F, M
head(df_selected,5)
```

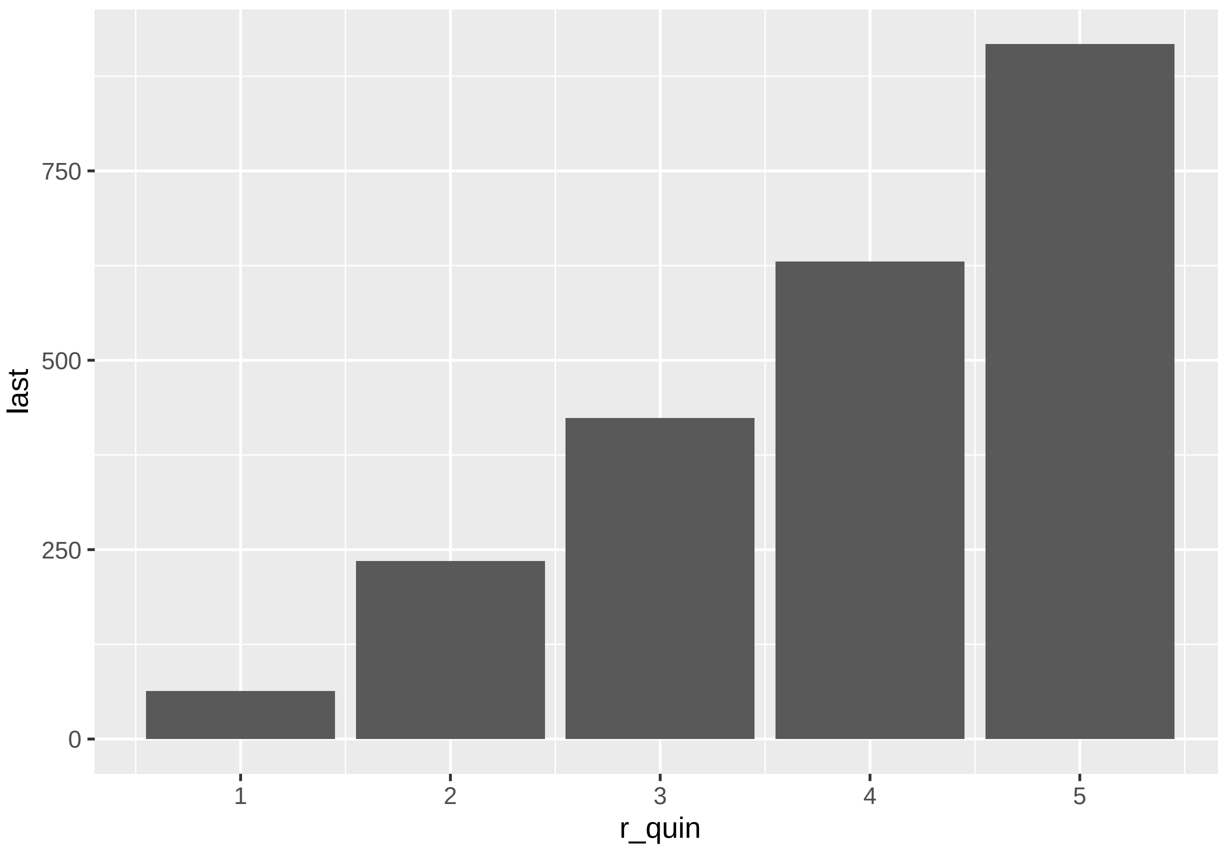
```
## # A tibble: 5 x 6
##    last r_quin numords f_quin totdol m_quin
##   <dbl>  <int>   <dbl>  <int>  <dbl>  <int>
## 1   207      2       7      5    493      5
## 2   625      4       4      4    423      4
```

```
## 3    28       1        4       4     246       4
## 4   778       5        3       3     271       4
## 5   396       3        2       2     148       3
```

*Q*3.

```r
library(ggplot2)
#Average of R by Quintile
ggplot(data=df,aes(x=r_quin,y=last)) +
  geom_bar(stat = "summary", fun.y = "mean")
```

```
## No summary function supplied, defaulting to `mean_se()`
```

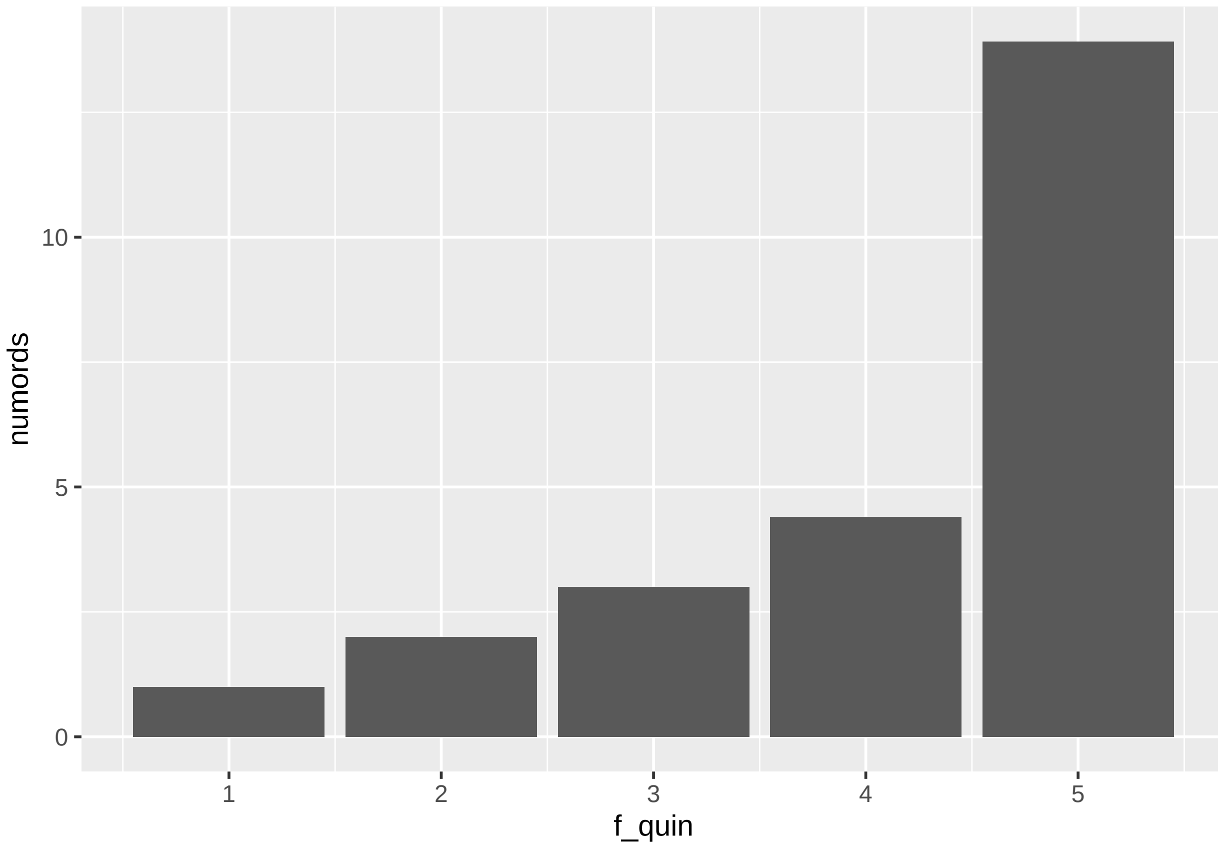


The graph depicts each recency quintile's average days before their last purchase. R-quin represents 5 quintiles, where 1 is the group with the most recent purchase with an average of nearly 50 days from last purchase and 5 being the group with an average purchase dating back nearly 950 days. Similarly, we have a look at frequency, where group 5 has the most frequency and group 1 has the least, which needs to be reversed to make it comparable with recency. Group 1 has an average of 1 order however group 5 is close to 14 orders.

```r
#Average of F by Quintile
ggplot(data=df,aes(x=f_quin,y=numords)) +
  geom_bar(stat="summary",fun.y="mean")
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`
```

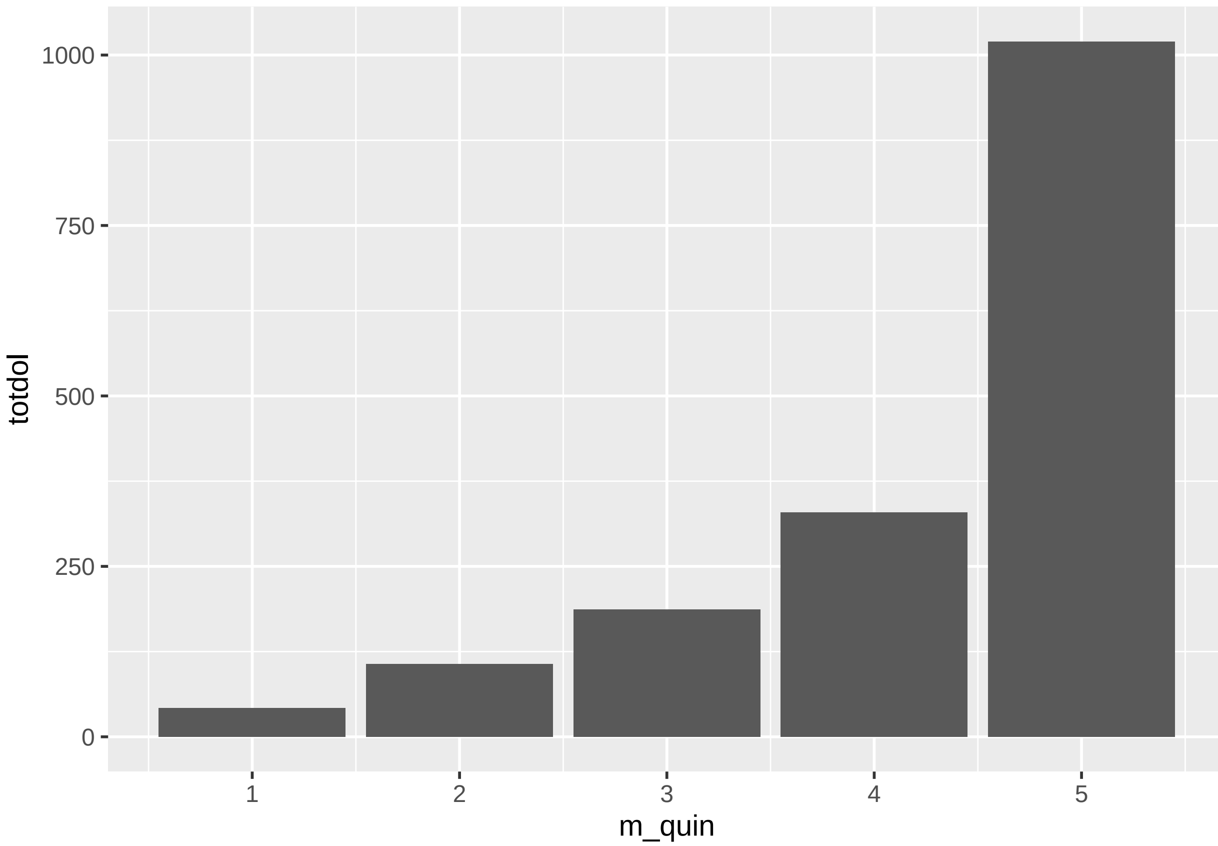
```
## No summary function supplied, defaulting to `mean_se()`
```




```r
#Average of M by Quintile
ggplot(data=df,aes(x=m_quin,y=totdol)) +
  geom_bar(stat="summary",fun.y="mean")
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`
```

```
## No summary function supplied, defaulting to `mean_se()`
```

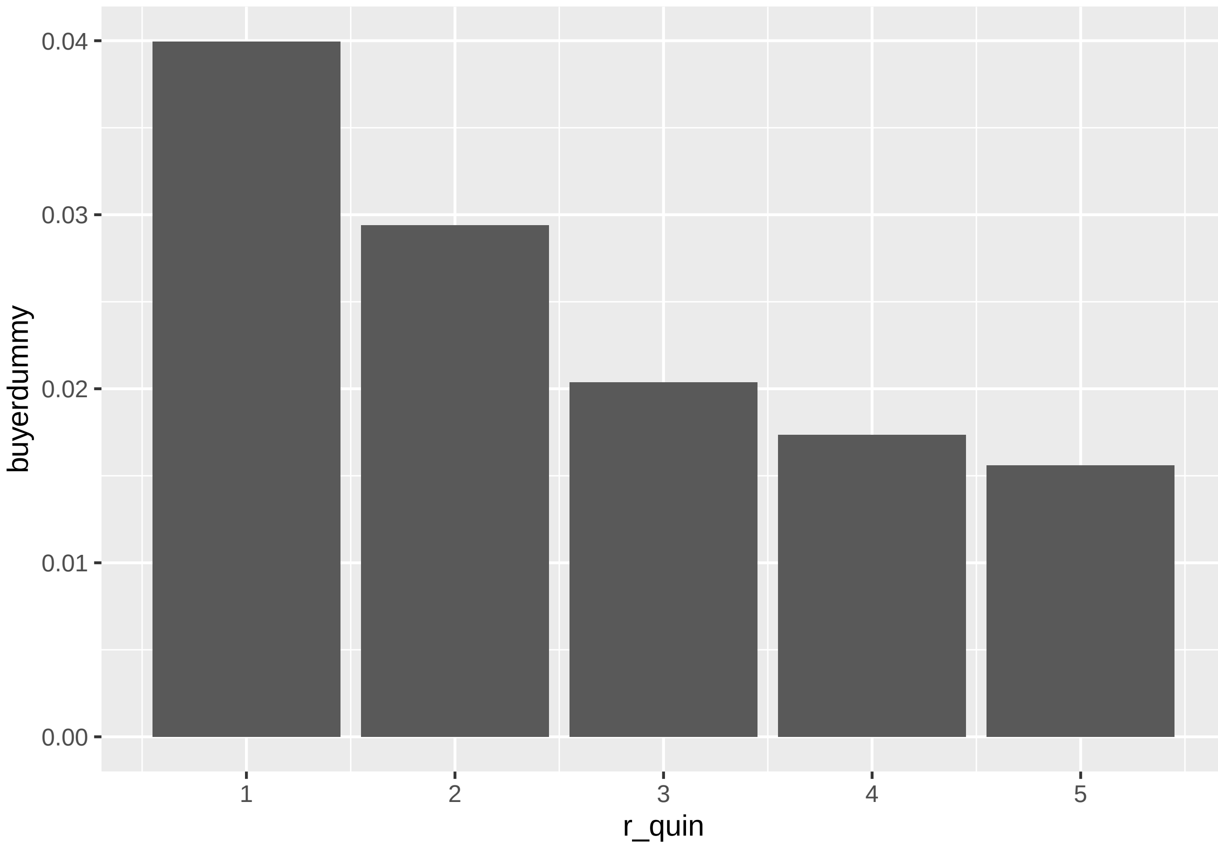


We move on to the third pillar of the RFM analysis with monetary value. Here we see a similar trend as the frequency where group 1 spends on average nearly 30 dollars per order. However, group 5 being the most spending customer with an average order value of 1000 dollars.

*Q*4.

```
#Predictive Nature of Recency for buying probability
ggplot(data=df,aes(x=r_quin,y=buyerdummy)) +
  geom_bar(stat = "summary", fun.y = "mean")
```

```
## No summary function supplied, defaulting to `mean_se()`
```

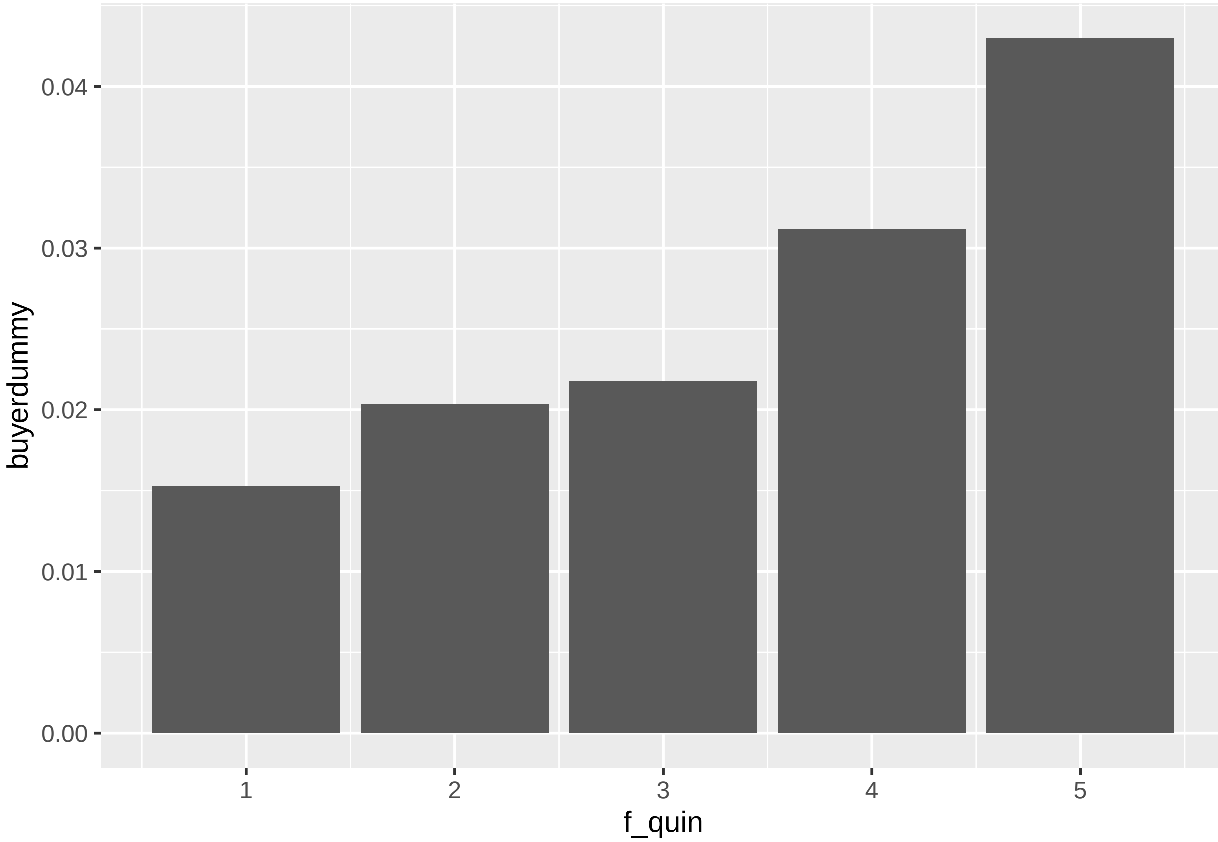


RFM analysis works with the philosophy that people who have their last purchase recently, purchase products from us frequently and have a high order value are our ideal customers. This maps out well with the response probability for each group. As we see in the graph, the most recent customers have a higher response rate verifying our claim.

```
#Predictive Nature of Frequency for buying probability
ggplot(data=df,aes(x=f_quin,y=buyerdummy)) +
  geom_bar(stat="summary",fun.y="mean")
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`
```

```
## No summary function supplied, defaulting to `mean_se()`
```

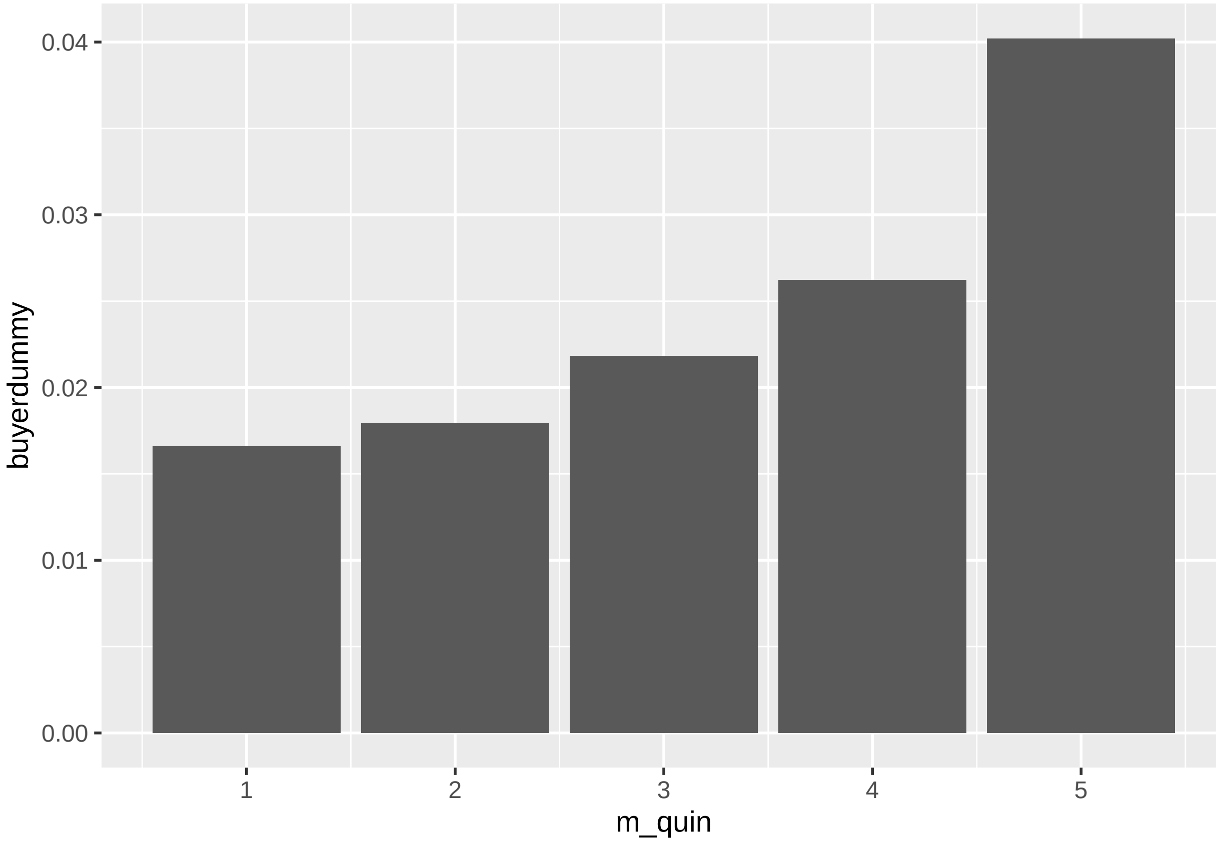


The frequency matches out as well however we need to switch around the groups to match with recency allowing group 1 to be the highest frequency group and group 5 to be the least. As we see, the group 5 with the highest frequency at the moment has the highest response to new purchases. Similarly, If we look at monetary value, we can see that group 5 which has the highest order value has the highest response rate when compared to group 1 which spends the least and has less probability of responding to the new product.

```r
#Predictive Nature of Monetary for buying probability
ggplot(data=df,aes(x=m_quin,y=buyerdummy)) +
  geom_bar(stat="summary",fun.y="mean")
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`
```

```
## No summary function supplied, defaulting to `mean_se()`
```

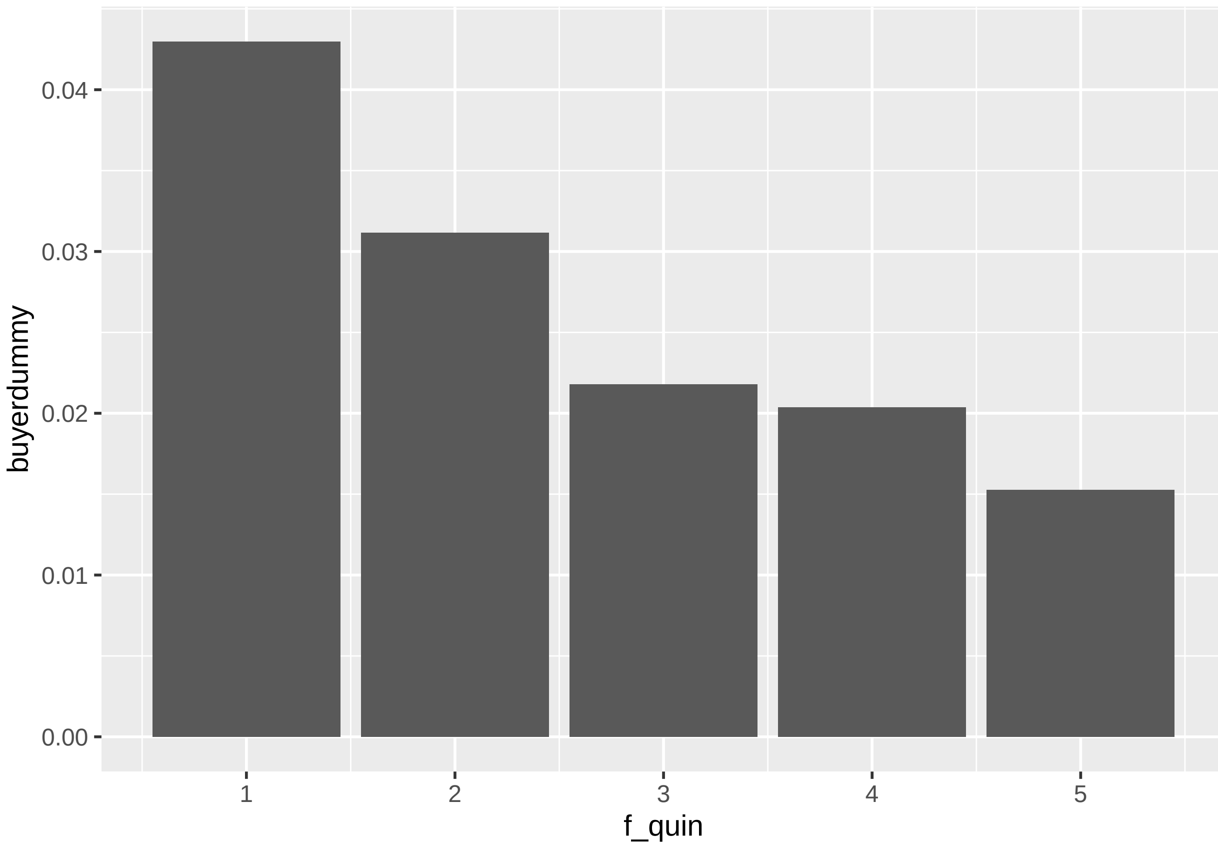


From this graph, it is observed that the first quintile isn't occupied by people with the highest spending.

```
#Reordering f_quin so group 1 is most likely to buy
df$f_quin<-6-df$f_quin
ggplot(data=df,aes(x=f_quin,y=buyerdummy)) +
  geom_bar(stat="summary",fun.y="mean")
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`
```

```
## No summary function supplied, defaulting to `mean_se()`
```

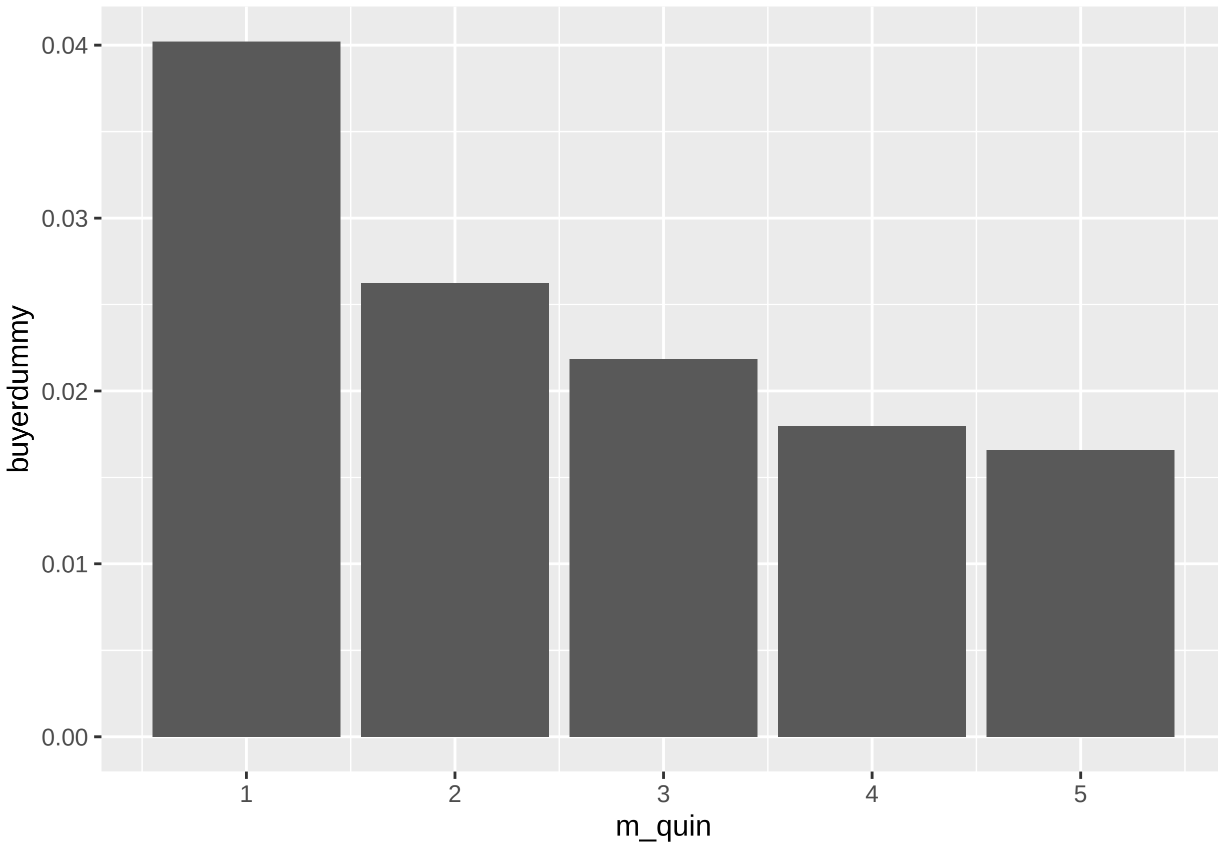


After the reordering, we see that group 1 is now the highest frequency group and group 5 being the lowest.

```
#Reordering m_quin so group 1 is most likely to buy
df$m_quin<-6-df$m_quin
ggplot(data=df,aes(x=m_quin,y=buyerdummy)) +
  geom_bar(stat="summary",fun.y="mean")
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`

## No summary function supplied, defaulting to `mean_se()`
```



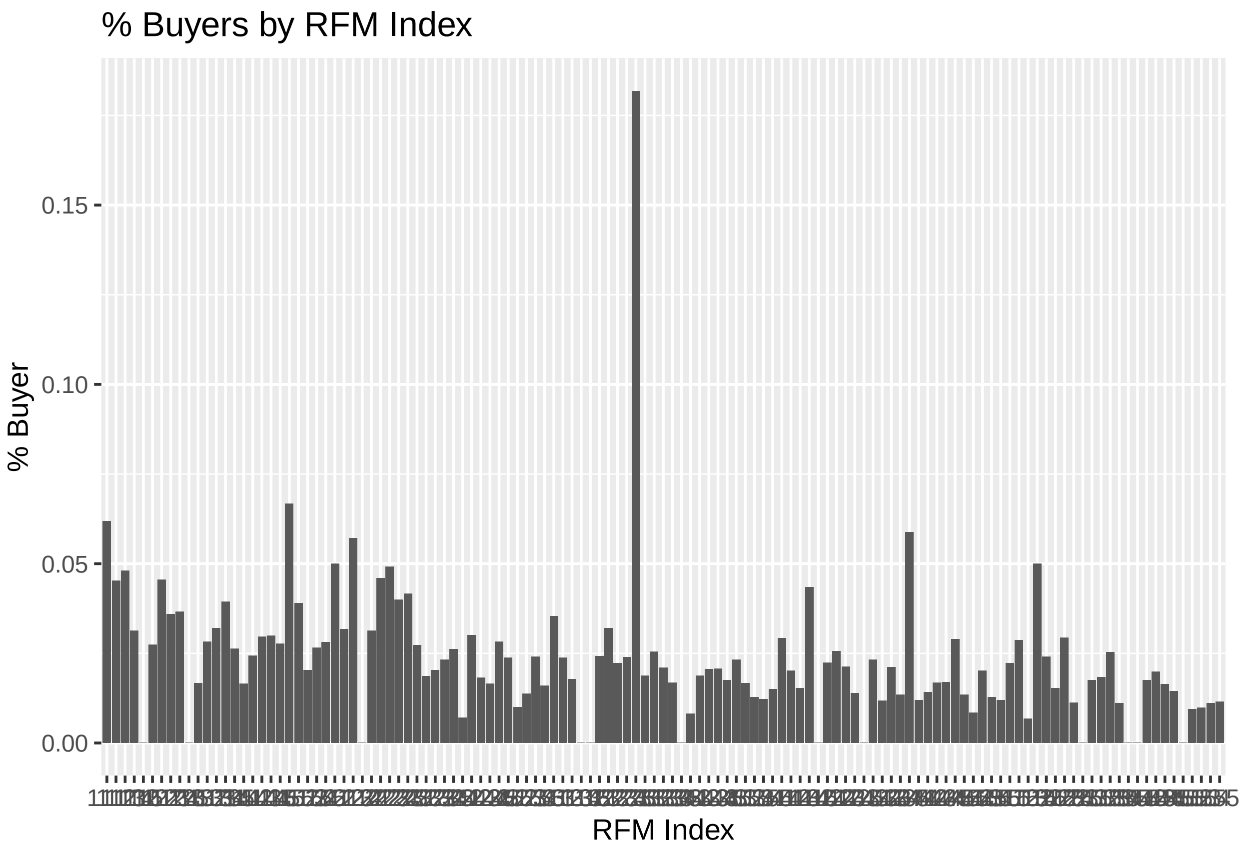
Similarily, after reordering for the M quintiles group 1

```
#Creating RFM Composite Index
df$rfm_ind<-(100*df$r_quin)+(10*df$f_quin)+(df$m_quin)
#Average response rate for each composite segment
df$rfm_response<-ave(df$buyerdummy,df$rfm_ind)
head(df)
```

```
## # A tibble: 6 x 12
##    ...1 numords totdol  last buyer buyerdummy dollars r_quin f_quin m_quin
##   <dbl>   <dbl>  <dbl> <dbl> <chr>      <dbl>   <dbl>  <int>  <dbl>  <dbl>
## 1     1       7    493   207 no             0       0      2      1      1
## 2     2       4    423   625 no             0       0      4      2      2
## 3     3       4    246    28 no             0       0      1      2      2
## 4     4       3    271   778 no             0       0      5      3      2
## 5     5       2    148   396 no             0       0      3      4      3
## 6     6      10    937     6 no             0       0      1      1      1
## # i 2 more variables: rfm_ind <dbl>, rfm_response <dbl>
```

```
ggplot(data = df, aes(x = as.factor(rfm_ind), y = buyerdummy)) +
  geom_bar(stat = "summary", fun.y = "mean")+
  labs(title="% Buyers by RFM Index", x="RFM Index", y="% Buyer")
```

```
## No summary function supplied, defaulting to `mean_se()`
```



Based on the RFM Index calculated using the n-tile quintile method we got groups individually for R, F and M. Now we put together the 3 indices to form a graph as above which represents the buyer probability to buy the new product.

Q5a.

```
rem_cust<-1834469
cost_mailing<-1
rev_cust<-mean_spent_test_cat
buyer_expect<-round((percentage_buyers_catalog*rem_cust)/100,0)
buyer_expect
```

```
## [1] 45049
```

We would expect 45049 buyers.

Q5b.

```
#Net Profit from revenue and cost of the marketing technique adopted
tot_rev<-rev_cust*buyer_expect
cost_gs<-0.5*tot_rev
tot_mail_cost<-cost_mailing*rem_cust
tot_cost<-cost_gs+tot_mail_cost
net_profit<-tot_rev-tot_cost
net_profit
```

```
## [1] 513551
```

The net profit is $513551.

Q5c.

```
#The return on marketing expenditure as a percent
return_market_exp<-(net_profit/tot_mail_cost)*100
return_market_exp
```

```
## [1] 27.99453
```

The return of market expenditure is approximately 27.99%.

Q6.

```
#In order to calculate the break-even response rate there is a need to use
#average revenue per customer, cost of mailing an offer, and
#cost of goods & services
c_gs_one<-0.5*rev_cust
break_even_rate<-(cost_mailing/(rev_cust-(c_gs_one)))*100
break_even_rate
```

```
## [1] 1.918595
```

The break-even response rate is approximately 1.92.

Q7a.

```
break_even_rate1<-break_even_rate/100
df$targeted[df$rfm_response>break_even_rate1]<-1
```

```
## Warning: Unknown or uninitialised column: `targeted`.
```

```
df$targeted[df$rfm_response<=break_even_rate1]<-0
head(df)
```

```
## # A tibble: 6 x 13
##    ...1 numords totdol  last buyer buyerdummy dollars r_quin f_quin m_quin
##   <dbl>   <dbl>  <dbl> <dbl> <chr>      <dbl>   <dbl>  <int>  <dbl>  <dbl>
## 1     1       7    493   207 no             0       0      2      1      1
## 2     2       4    423   625 no             0       0      4      2      2
## 3     3       4    246    28 no             0       0      1      2      2
## 4     4       3    271   778 no             0       0      5      3      2
## 5     5       2    148   396 no             0       0      3      4      3
## 6     6      10    937     6 no             0       0      1      1      1
## # i 3 more variables: rfm_ind <dbl>, rfm_response <dbl>, targeted <dbl>
```

```
tabled<-table(as.integer(df$targeted))/nrow(df)
optimum_market_fraction<-tabled[2]
optimum_market_fraction
```

```
##         1
## 0.5455355
```

The optimum fraction of the market to target is approximately 0.55.

Q7b.

```r
library(psych)
```

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
summed_stats_by_test_cat<-describeBy(df$buyerdummy,df$targeted)
summed_stats_by_test_cat
```

```
##
##  Descriptive statistics by group
## group: 0
##    vars     n mean   sd median trimmed mad min max range skew kurtosis se
## X1    1 43879 0.01 0.12      0       0   0   0   1     1 8.35    67.66  0
## ------------------------------------------------------------
## group: 1
##    vars     n mean   sd median trimmed mad min max range skew kurtosis se
## X1    1 52672 0.03 0.18      0       0   0   0   1     1 5.18    24.84  0
```

```r
avg_response_rate<-0.03
avg_response_rate
```

```
## [1] 0.03
```

The expected response rate is 0.03.

Q7c.

```r
#Number of mails sent under the targeted policy
sent_mail_num<-rem_cust*mean(df$targeted)
sent_mail_num
```

```
## [1] 1000768
```

```r
#Expected number of responses under the targeted policy
exp_responses_num<-sent_mail_num*mean(df[df$targeted==1,]$buyerdummy)
exp_responses_num
```

```
## [1] 33573
```

```r
#Net Profit achieved from targeted policy
mail_cost_targeted<-cost_mailing*sent_mail_num
net_profit_targeted<-((rev_cust-c_gs_one)*exp_responses_num)-(mail_cost_targeted)
net_profit_targeted
```

```
## [1] 749106
```

The net profit is $749106.

Q7d.

```r
roi_targeted<-(net_profit_targeted/mail_cost_targeted)*100
roi_targeted
```

```
## [1] 74.85312
```

The ROI of marketing expenditure is approximately 74.85.

Q7e.

The RFM approach targets customers based on recency, frequency, and monetary value of past purchases which are strong predictors of future purchasing behaviour of the customers. By focusing on the recent, frequent, high-value buyers the RFM approach manages to allocate marketing resources more appropriately as compared to a mass marketing strategy as observed above. The RFM approach yields higher response rates, better conversion, and higher net profit and ROI as it manages to- 1. Reduce Waste by not targeting those customers who are unlikely to respond 2. Increase efficiency by focusing on those with higher propensity to buy 3. Enhances customer experience by tailoring offers to those interested in them resulting in higher customer loyalty and lifetime value