STA380 - Exercise

Leyang Xu/Liyao Wang/Xiaohan Sun/Yue Cui

Date: August 16, 2021

Group: Exercise Group 10

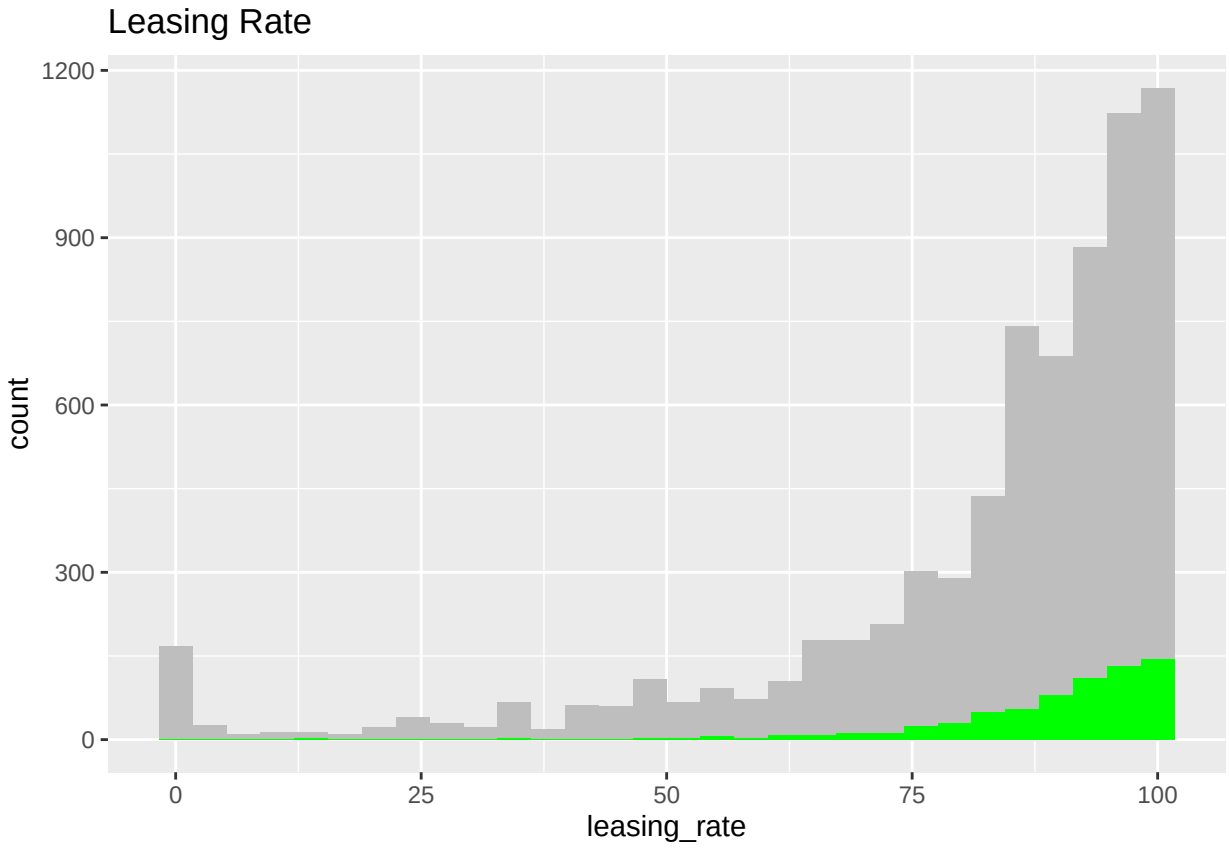
Github Link: https://github.com/EmmaS1116/STA380-GroupExercises

Question 1: Green Buildings

In this part of assignment, our goal is to provide insights and suggestion to the developer on whether he should invest this project of green building based on our explanoatory analysis of the dataset.

Let's look at the analysis from the on-staff stats guru first.

According the on-staff stats guru, he cleaned the data by removing those buildings that had very low occupacy rate. Let's look at the occupacy rate of buildings in our data.



As we can see see from the histogram above, there are more outlier on leasing rate in non-green buildings than green building.Overall, we see that green building are more concentrated on higher leasing rate, and non green building are more spread on leasing rate.

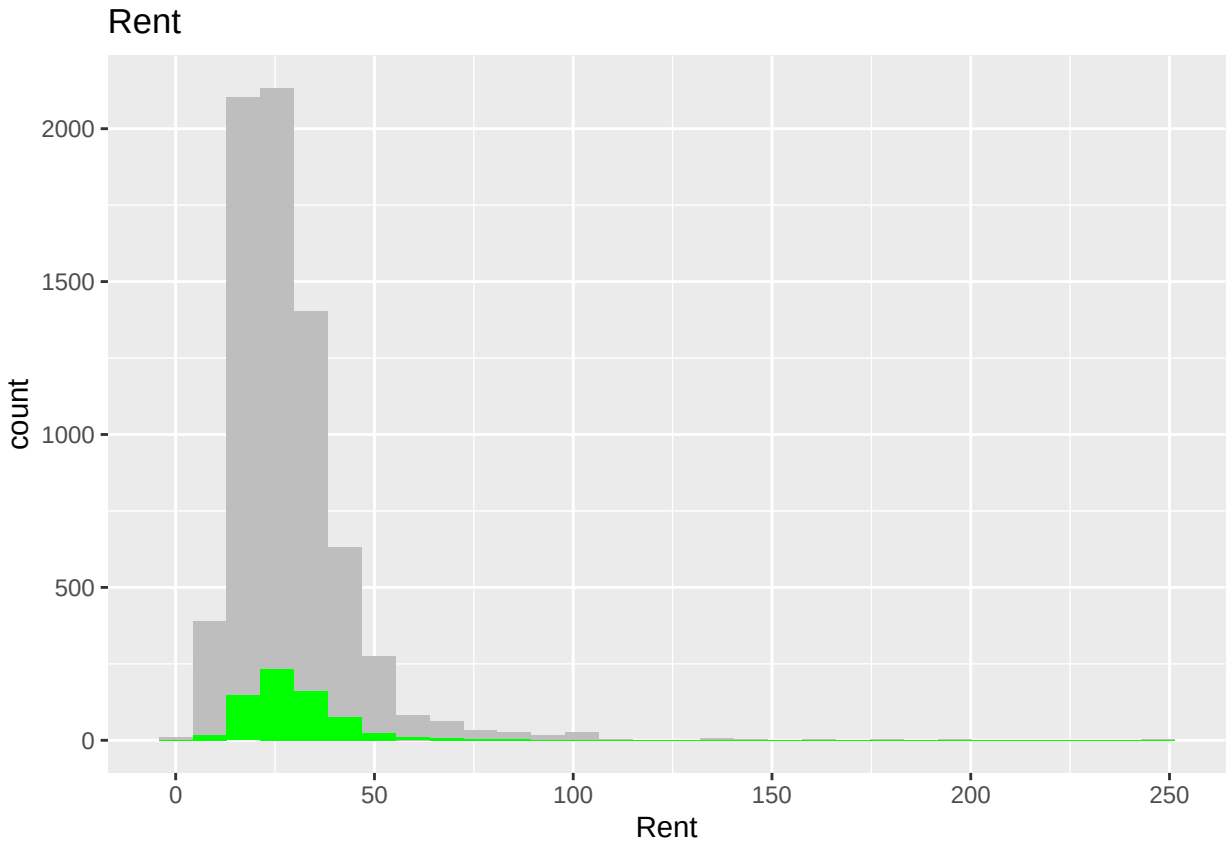
For non-green buildings, there are significant amount of buildings that have leasing rent less than 10%. Let's check how many buildings have leasing rate <=10.

```
## [1] 215
```

It seems not reasonable for the on-staff stats guru to clean this part of outlier. There are considerable number of these buildings and they may represent an important part of our data.Removing them may cause some bias on the calculation of rent.

Then, the on-staff stats guru just simply separated the green and non-green buildings and subtract their median rent per square foot per yer.

Let's look at the distribution of the rent based on green and non green building.



Clearly, there are many outliers points that are over $100, so he is right on using median instead of mean since median is more robust to outliers.

However, he cannot jump to the conclusion that green buildings have higher rent by just comparing the median rent between these two groups. this is very very shortsighted because there are many other factor that also influence the rent, and we need further analysis on the possibility of confounding variables for the relationship between rent and green status.

First, Let's run a quick linear regression on rent and other variables, and look at the correlation and potential interactions.

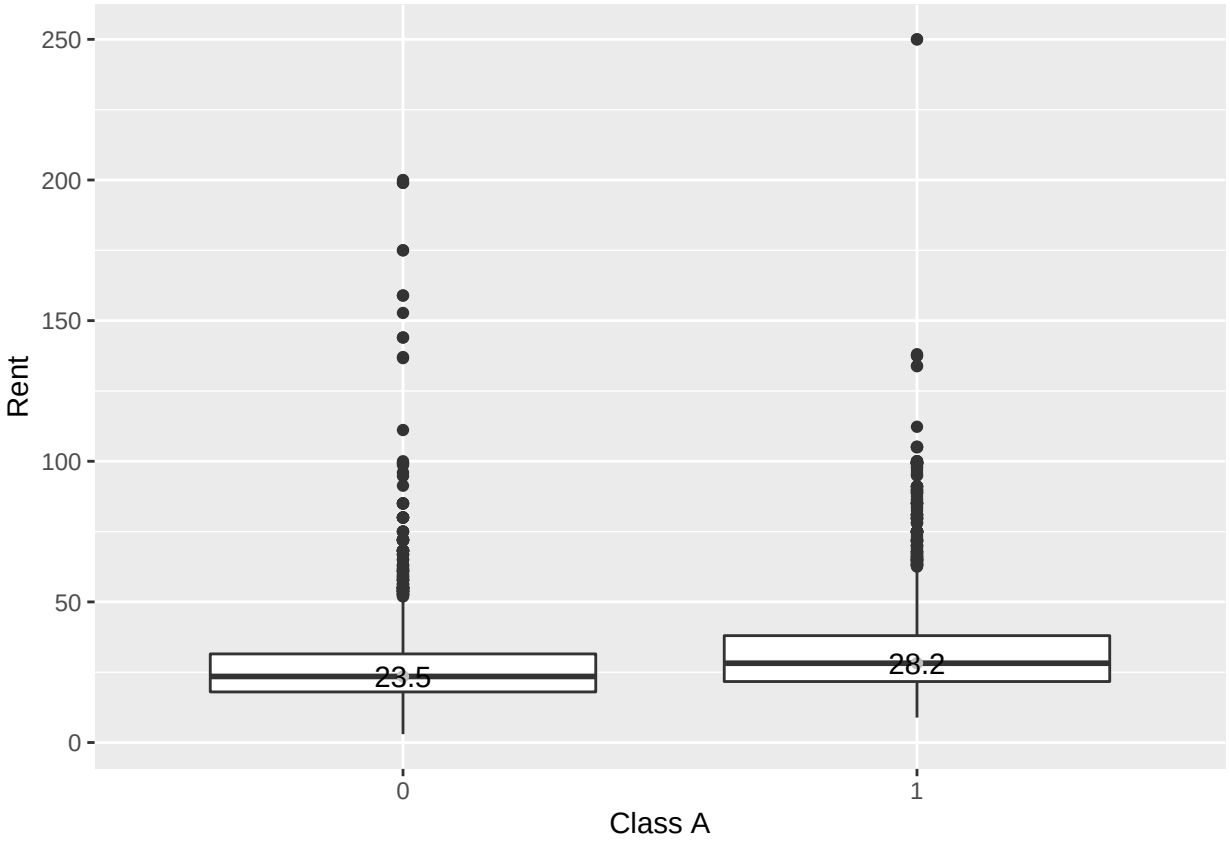
```
##
## Call:
## lm(formula = Rent ~ ., data = green_buildings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -53.753  -3.581  -0.526   2.491 173.916
##
## Coefficients: (1 not defined because of singularities)
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -8.315e+00  1.018e+00  -8.167 3.67e-16 ***
## CS_PropertyID      2.959e-07  1.574e-07   1.879 0.060241 .
## cluster            7.532e-04  2.840e-04   2.653 0.008006 **
## size               6.741e-06  6.561e-07  10.276  < 2e-16 ***
## empl_gr            6.450e-02  1.700e-02   3.794 0.000149 ***
## leasing_rate       9.454e-03  5.332e-03   1.773 0.076247 .
```

```
## stories           -3.472e-02  1.617e-02  -2.147 0.031823 *
## age               -1.249e-02  4.717e-03  -2.649 0.008096 **
## renovated         -1.425e-01  2.586e-01  -0.551 0.581681
## class_a            2.872e+00  4.377e-01   6.563 5.63e-11 ***
## class_b            1.186e+00  3.427e-01   3.462 0.000539 ***
## LEED               1.877e+00  3.582e+00   0.524 0.600318
## Energystar        -2.127e-01  3.818e+00  -0.056 0.955572
## green_rating       6.969e-01  3.839e+00   0.182 0.855929
## net               -2.559e+00  5.929e-01  -4.316 1.61e-05 ***
## amenities          6.703e-01  2.519e-01   2.661 0.007802 **
## cd_total_07       -1.248e-04  1.464e-04  -0.852 0.394005
## hd_total07         5.354e-04  8.972e-05   5.967 2.52e-09 ***
## total_dd_07               NA         NA      NA       NA
## Precipitation      4.830e-02  1.611e-02   2.997 0.002735 **
## Gas_Costs         -3.559e+02  7.842e+01  -4.538 5.76e-06 ***
## Electricity_Costs  1.886e+02  2.493e+01   7.563 4.38e-14 ***
## cluster_rent       1.008e+00  1.421e-02  70.949  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.413 on 7798 degrees of freedom
##   (74 observations deleted due to missingness)
## Multiple R-squared:  0.6126, Adjusted R-squared:  0.6116
## F-statistic: 587.2 on 21 and 7798 DF,  p-value: < 2.2e-16
```

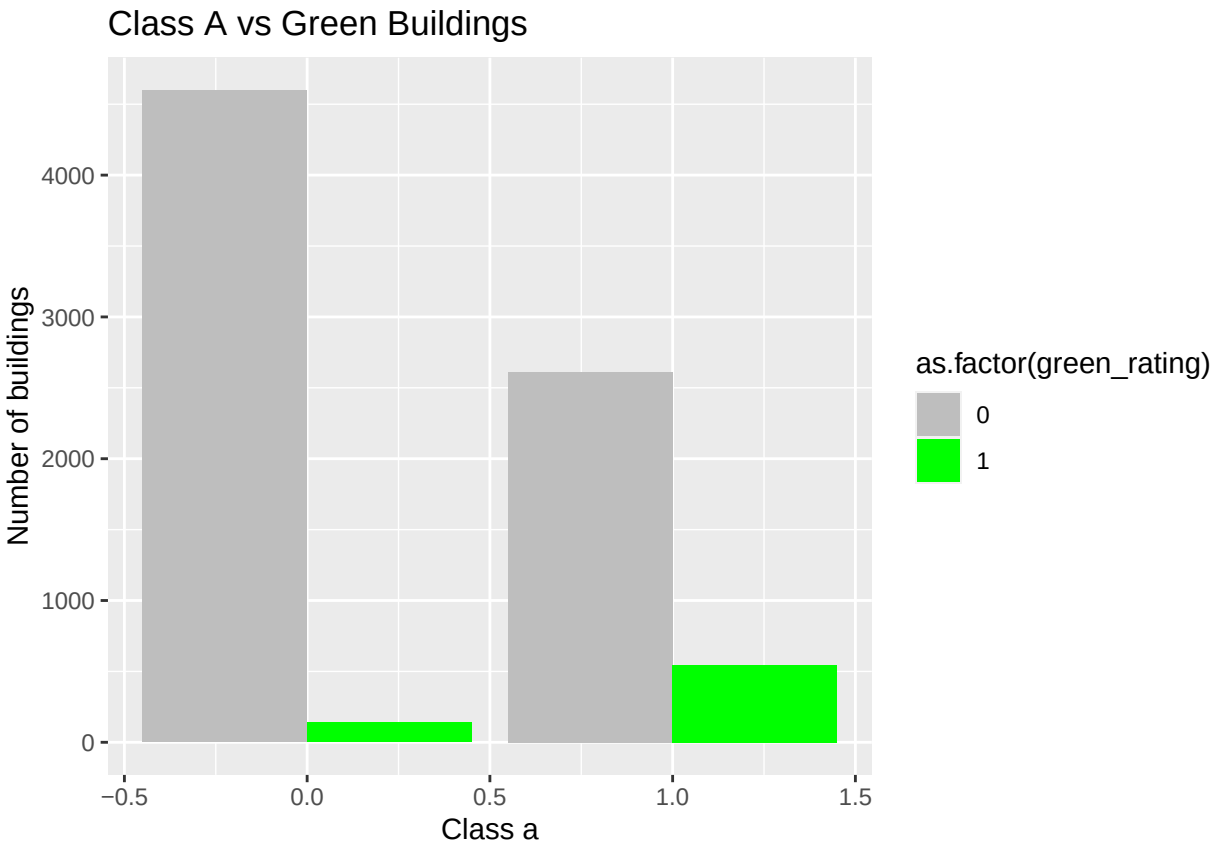
It appears that the green rating is not that significant on determining the rent.The summary also provides us the target for some potential compounding variables, like class a,cluster, age and size

Class

```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```



The median rent of classA building is 28.2 dollar, which is near five dollar higher than those buildings that not belongs to classA, with median rent of 23.5 dollar.



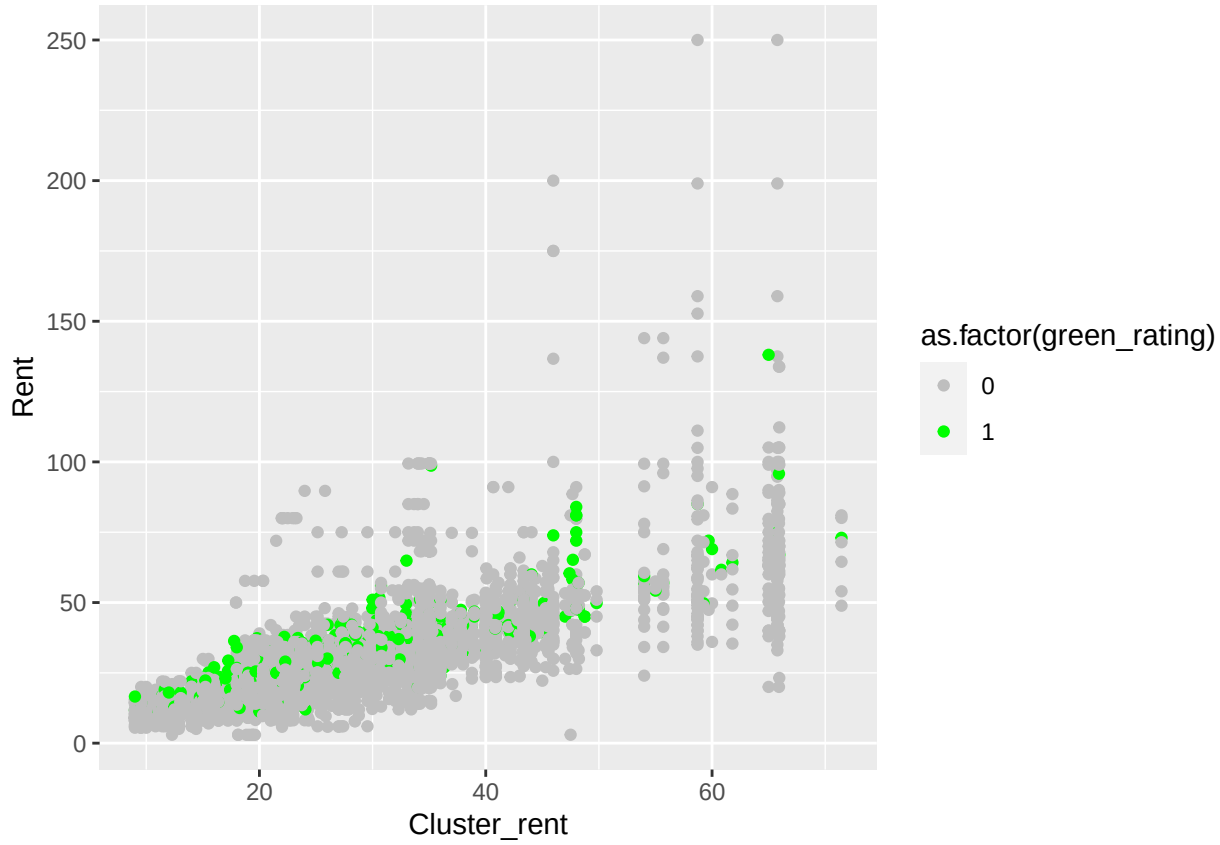
Also, more green buildings belongs to ClassA as we see from the graph.

```
## the propotion of green building that belongs to classA = 0.7970803
```

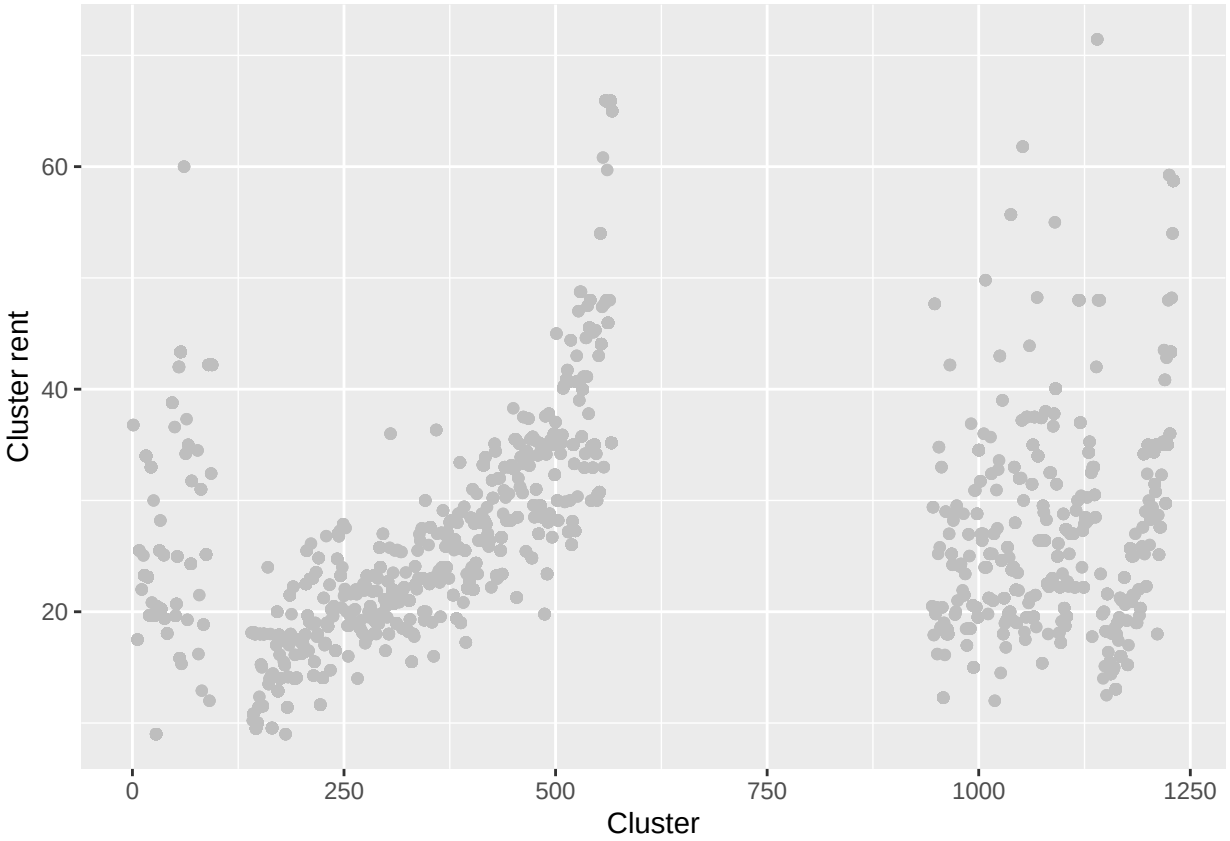
```
## the propotion of not green building that belongs to classA = 0.3621862
```

The class_a is a compounding variable that both affect rent and green or not.

Clusters



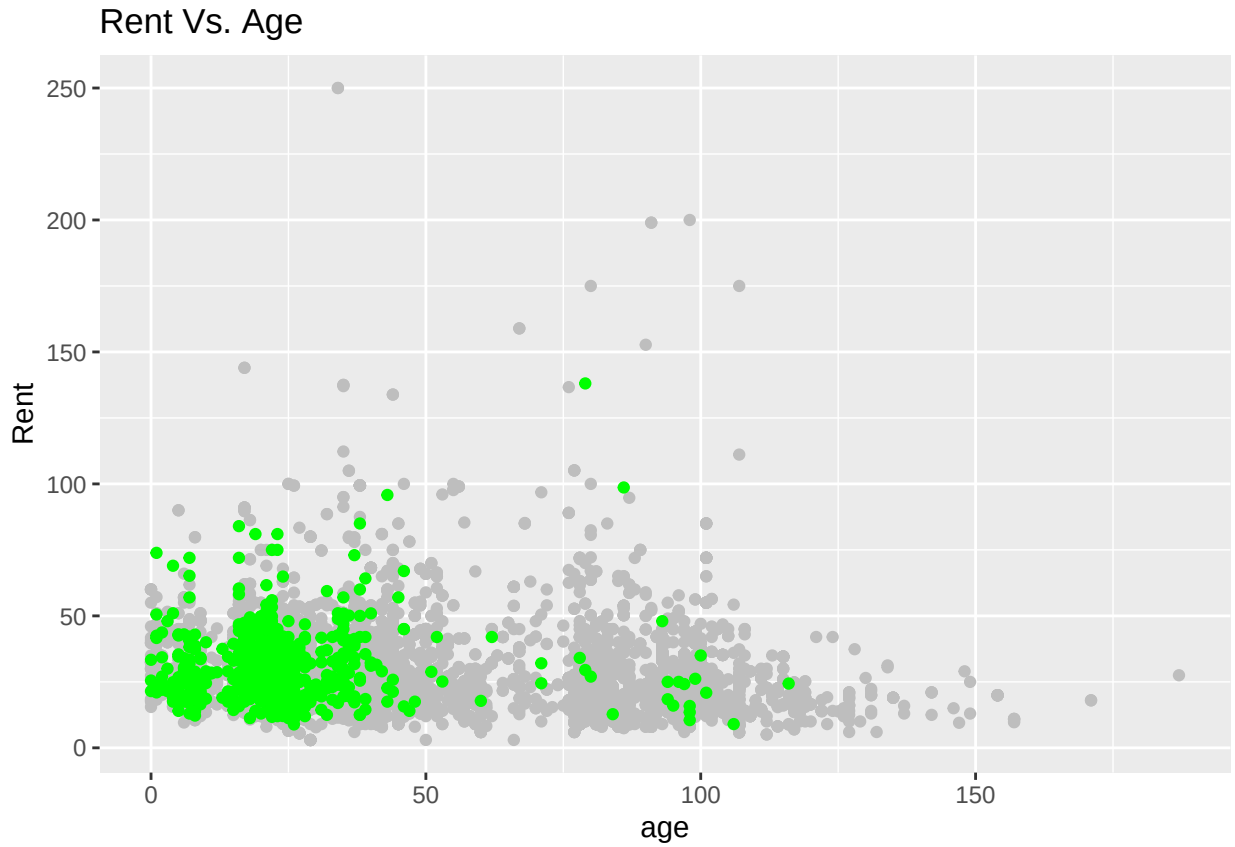
Rent is correlated with the cluster rent for both green and not green buildinga



Most of the clusters' rents are concentrated between 10-40 dollar. There are some outliers go up to 60+ too, maybe are those clusters that located in very great location.

So, which cluster the building belongs to, or location can be a compounding variable that affect the rent.Since the developer wants to build the building on East Cesar Chavez, just across I-35 from downtown, he can looks at the cluster rent of this location.

Age



Most of the green buildings are younger than non-green buildings. However, there is no clear correlation between age and rent, which means that younger age building may not resulting in a higher rent.

```
## Mean age of green building is  = 23.84526
```

```
## Mean age of not green building is  = 49.46733
```

Since the concept of the green building is quite new, the mean age of green building is younger than not green building, and is about 24 years old. So, the assumption that on-staff stat guru made about the green buildings that theywill be earning rents for 30 years or more is reasonable.

Size

```
## # A tibble: 2 x 3
##   green_rating med_size mean_size
##          <int>    <int>     <dbl>
## 1            0   118696   225977.
## 2            1   241150   325781.
```

We can see that green building have larger size than non green building.

There is subtle correlation between size and rent that larger the size, higher the rent.

With all these compounding variables, it becomes very hard to tell whether the increased Revenue Per Square Foot in green buildings really is due to their green rating. The best way to answer this question, is that we need to adjust all other compounding variables to the same level, and than, compare the rent of green and non green buildings.For example, we can find two identical buildings, one is green, one is not green, in the same area and was built at same year. We can make these matches building in to a new data sets and then compare their mean or median. If the rate of green building is still higher, we can then be confident about the economic value of the green buildings.

Although it is very hard to find enough amount of the exact two same buildings, we can still use this methodology of controlling compounding variables into our analysis.Let's first consider the size of the building. In our case, our building would be 250,000 square feet. let look at the rent difference in this size of building.

```
## Median value for green buildings in the 20k-30k sqft size range = 28.82
```

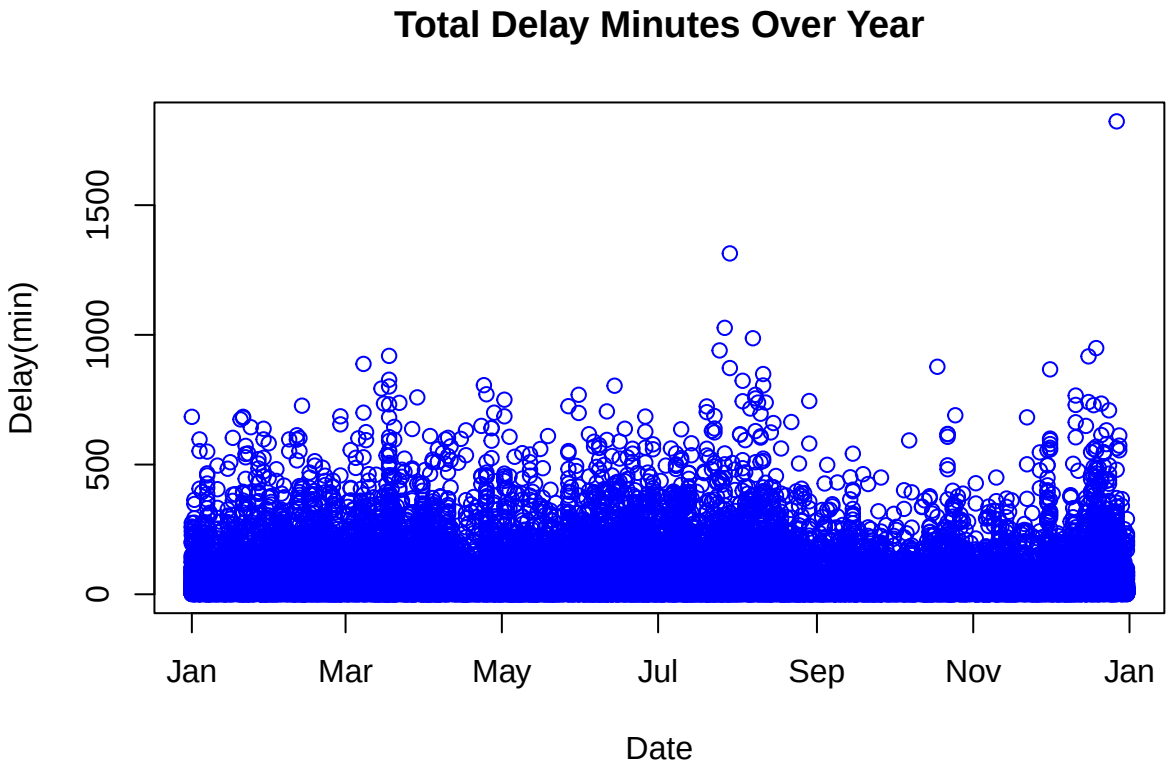
```
## Median value for non-green buildings in the 20k-30k sqft size range = 27.95
```

In this size range, the green building have a higher range than non green building with a premium about 1 dollar.

In conlusion,the anlysis of the on-staff stat guru is not correct because it fails in considering important compounding variables that affect the rent. Our suggestion is that our developer should focus first on the location of the building and whether it is a class a building. After considering about all these variables, and use our methodology of controlling compounding variables, the developer can then decide whether is worth to pay a 5% expected premium for green certification.

Question 2: Flights at ABIA

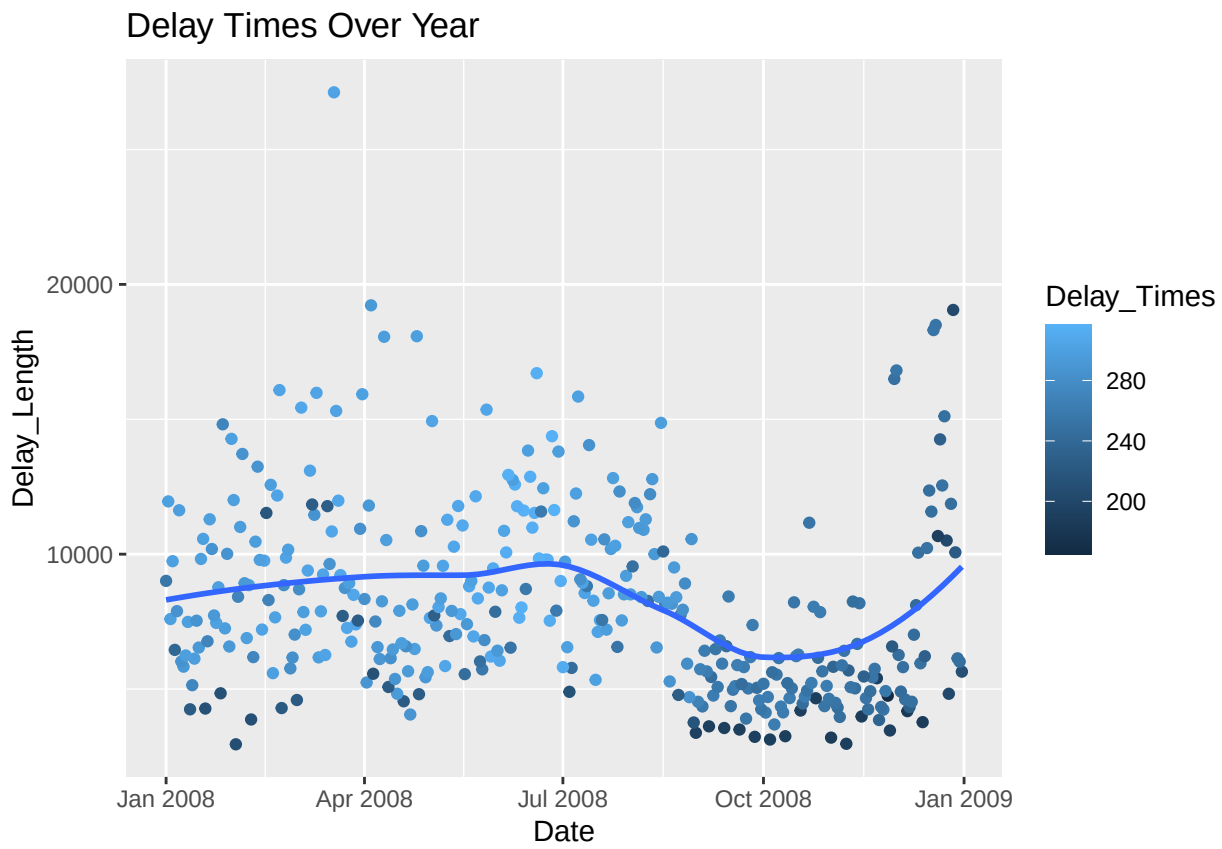
(a)What time of a year has the minimum delays?



Note that we define the term "delay" as being deviant from the scheduled time no matter it's later or early, since they are both creating inconvenience for certain passengers. Thus, the "total delay" column is using the sum of absolute value of departure and arrival delay.

We can't clearly see what time of year has the minimum delays by minutes, since many points are overlapped.

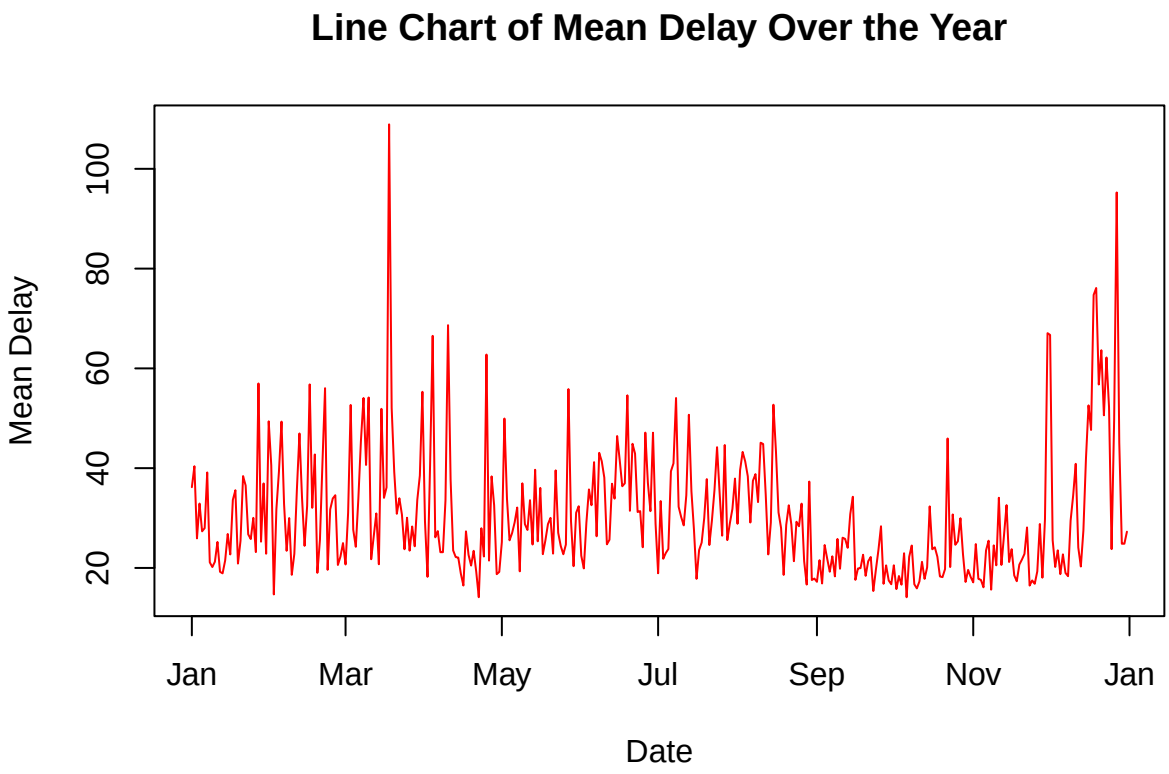
Therefore we're going to try to count the delay times to see when is the fewest delays.

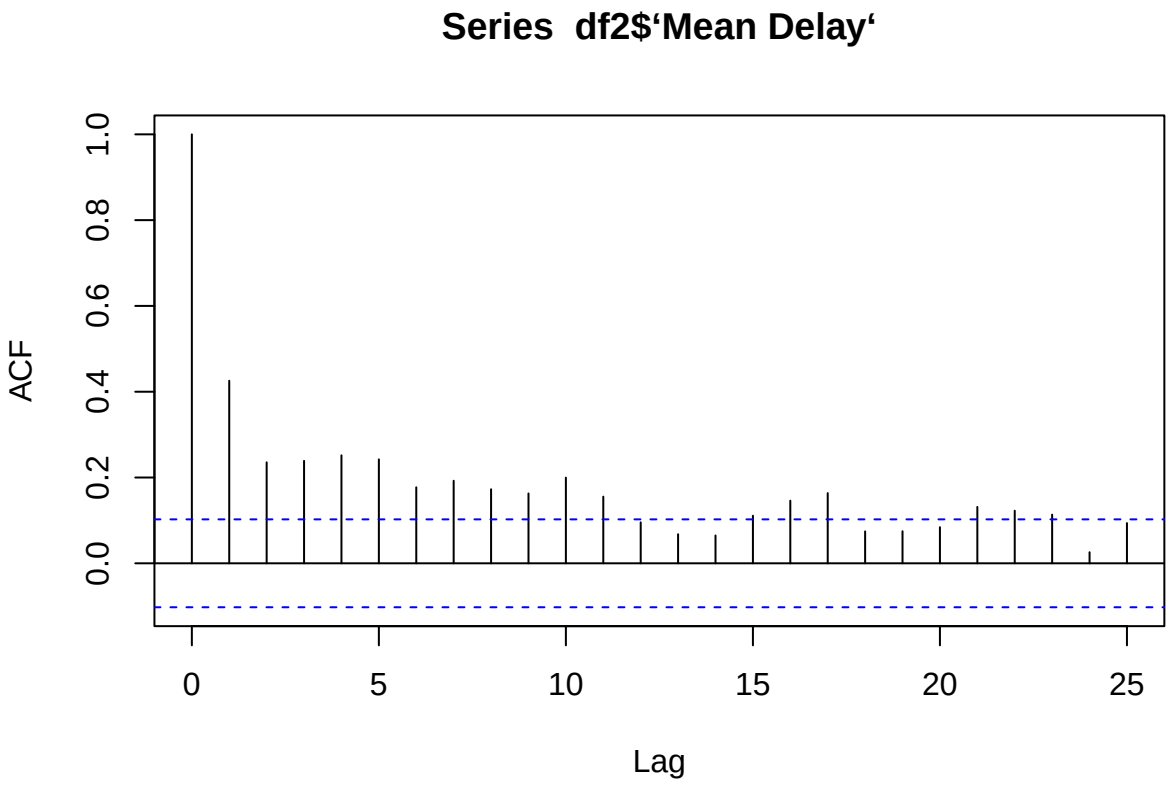


This plot shows the relationships between Delay_Length and Dates, while the ggplot automatically put months on x-axis rather than using specific dates. And each point was colored depending on the delay times. So we can clearly see how long and how many times of delays over a entire year.

According to the plots shown above, there would be minimum delays during September to November. We got this conclusion based on three reasons. First, there are fewer points spreaded from September to November, which means there are less delays. Also, most points during that period have shorter Delay_Length than other points. Moreover, most pinots in the period of September to November have the darkest color, which indciates there are less number of delays.

One more thing we want to specify is about why we choose to analyze the delays by period in a year instead of the specific day. Since there are lots of factors would affect flights in a day, such as weather, mechanical problems, and air regulations, our predictions about the delay by days would not help people to make correct decisions. We can't know if today's story would also happen on the same day in the next year. Moreover, people usually would have a rough time range to fly, and they won't cancel their trips just because the day has "high possible delay" based on previous years.

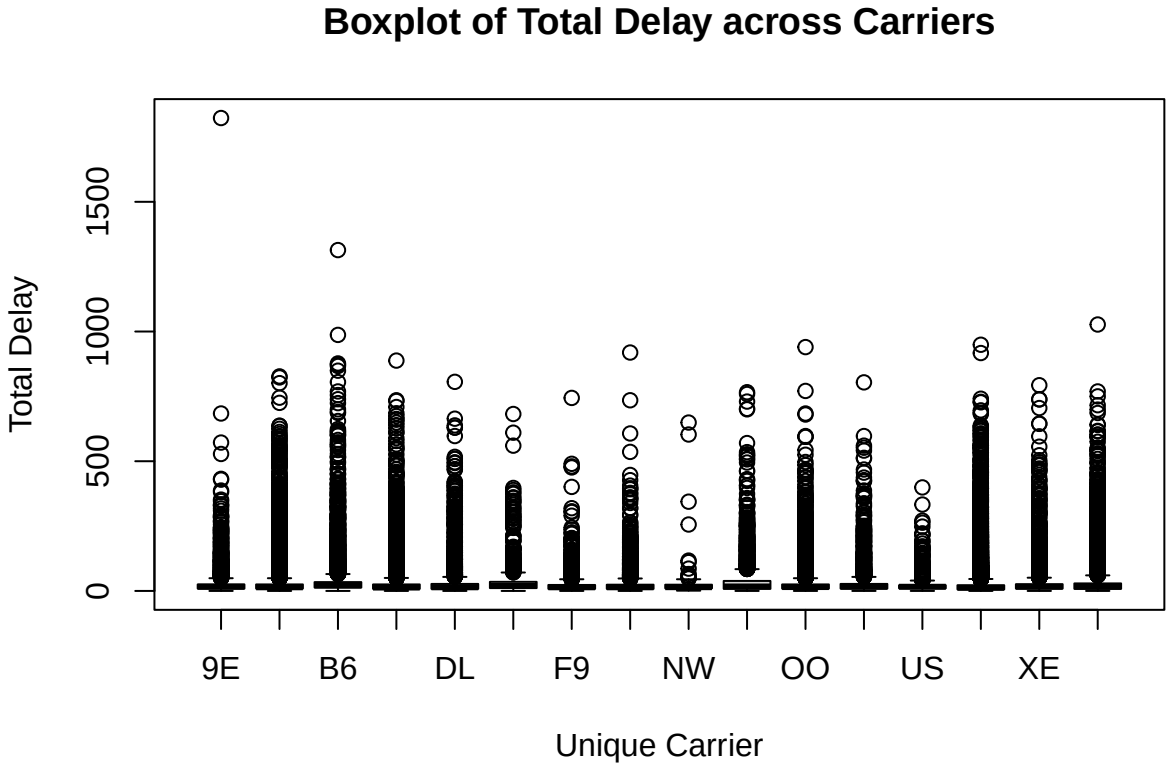

Line Chart of Mean Delay Over the Year
Mean Delay
20
40
60
80
100
Jan
Mar
May
Jul
Sep
Nov
Jan
Date

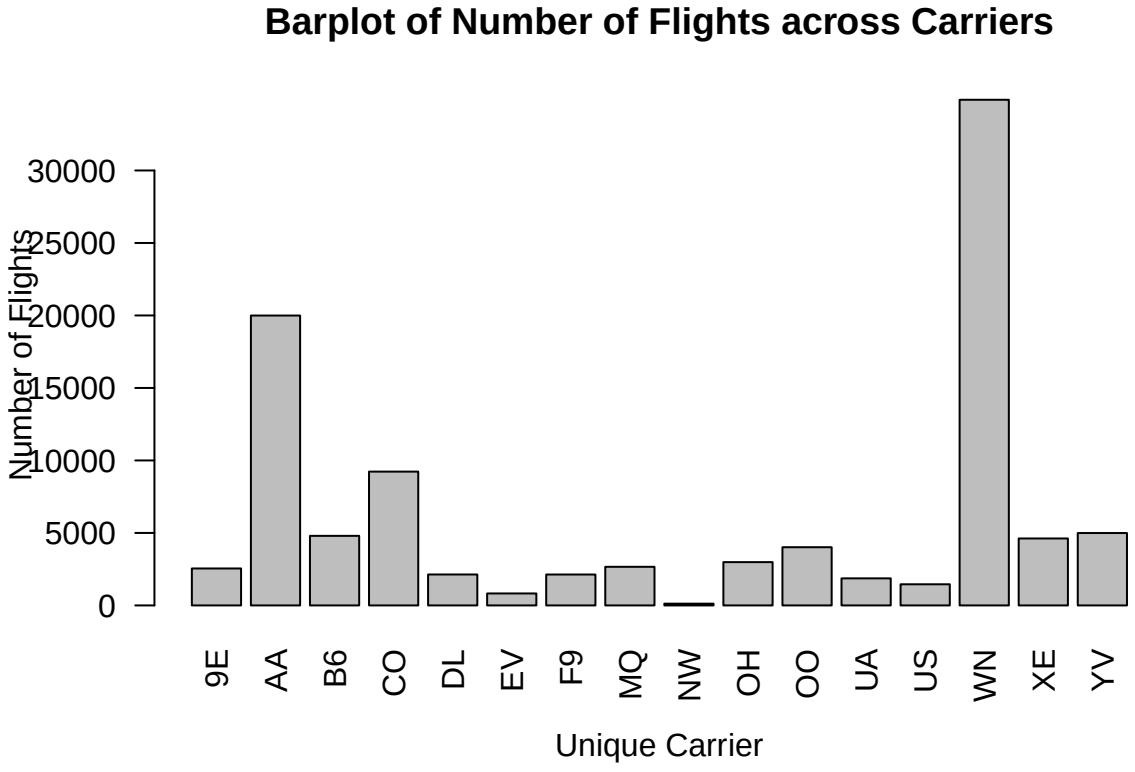


In order to see the pattern better, we represent the same data with a line chart as well. From the line chart, we observe that the mean delay reaches its lowest during the period from September to November, which is the best time of the year if you and your family plan to do a annual trip.

Additionally, we checked for autocorrelaion in total delay. The acf plot indicates that the most obvious pattern is first degree autocorrelation, which is reasonable because delays have Domino effect (airport terminals and gates are occupied)

(b)Which UniqueCarrier we recommend to choose and which not?





```
##       9E       AA       B6       CO       DL       EV       F9       MQ
## 28.33722 31.69388 44.14960 32.14094 35.85111 41.71287 23.56036 29.79125
##       NW       OH       OO       UA       US       WN       XE       YV
## 34.90678 38.95775 32.67368 35.00054 21.23230 26.19585 29.48625 38.52921
```

```
##       9E       AA       B6       CO       DL       EV       F9       MQ
## 59.20306 56.16241 82.42154 61.19511 72.95057 72.07204 39.11816 54.69964
##       NW       OH       OO       UA       US       WN       XE       YV
## 88.11540 66.56212 63.21297 64.88341 28.70741 49.68890 56.59090 76.51914
```

By creating a boxplot of total delays across different unique carriers, we found that all carriers have a relatively large number of high outliers (that is why the boxes are barely shown). This also result in that SDs are generally larger than means.

The mean value of total delay table circles EV, OH, and YV as the three carriers with longest average delays (38-41); US, WN, F9 are among the carriers with the shortest delays (21-23)

The standard deviation of each carriers' delays pinpoints NW, YV, B6 as the three most volatile carrier in terms of delay lengths (76-88); US appears to be the most stable one and stands out from others (28.7).

We combine our findings with how many flights does each carrier has. The barplot indicates that while US has the lowest mean delay and it's the most stable one, it's also has few flights for us to choose; WN has the most flights but it's surprisingly not the carrier with the worst performance overall.

At the same time, we can analyze this question by looking at times and lengths of delays together, as shown below.

```
##    UniqueCarrier Delay_Times Delay_Length
## 2             AA       19995       614893
## 14            WN       34876       907241
```

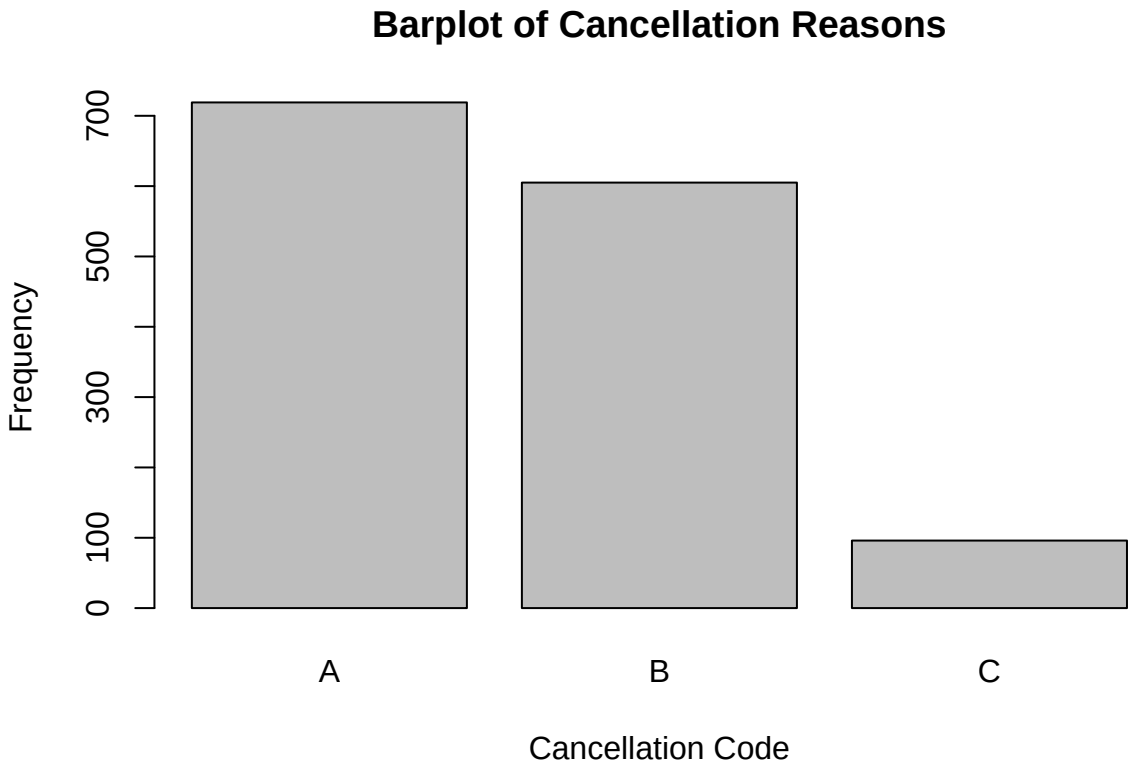
```
##   UniqueCarrier Delay_Times.x Delay_Length.x Delay_Times.y Delay_Length.y
## 1            EV           825          33704           825          33704
## 2            F9          2132          50160          2132          50160
## 3            NW           121           4119           121           4119
## 4            UA          1866          64681          1866          64681
## 5            US          1458          30893          1458          30893
```

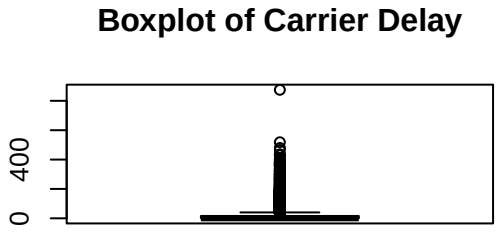
To sum up, We applied different methods to evaluate different carriers based on the delay times and length of delay. According to the analysis above, we would recommend people to choose EV, NW, UA, US, and F9. These unique carriers have fewer delay times and shorter length of delay. We don't recommend people to take AA and WN, due to their extremely large number of delay times. Keep in mind that some of these high quality carriers have few flights so it might be hard to buy their tickets.

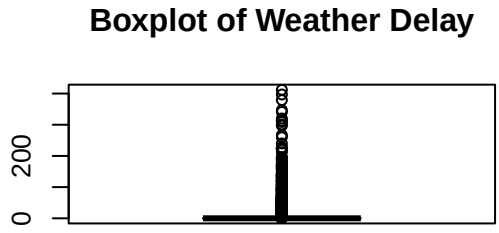
What's the most common reason for cancellation and delay?

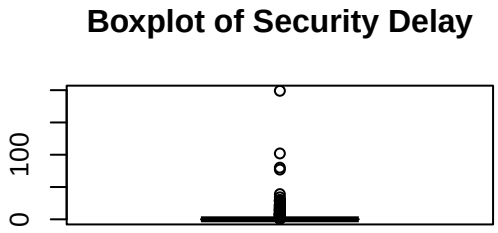
```
##
##     0     1
## 97840  1420
```

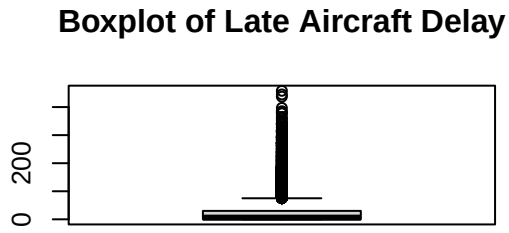
```
## [1] 0.01430586
```


Barplot of Cancellation Reasons
Frequency
0
100
300
500
700
A
B
C
Cancellation Code










```
## [1] 19747

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00    0.00   15.39   16.00  875.00   79513

## [1] 19747

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00    0.00    2.24    0.00  412.00   79513

## [1] 19747

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00    2.00   12.47   16.00  367.00   79513

## [1] 19747

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00    0.00    0.07    0.00  199.00   79513

## [1] 19747

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00    6.00   22.97   30.00  458.00   79513
```

First, we made a table to see that the proportion of flights that was cancelled was only about 1.43%. So passengers needn't to worry too much about their flights being cancelled.

Next, a barplot is genereated to check which is the most frequent reason for flight cancellation (A-carrier, B-weather, C-NAS, D-security). Carrier(719) ends up being the most common one, followed by weather(605) and NAS(96). There's no case for security reason in our dataset. However, we should be cautious with saying that there's no flight being delayed for security reasons since the data could be missing.

Lastly, we want to see whether the most common delay reason also causes the longest delay. We calculate the sample size for all delay reasons(19747) and found their summary values. Carrier delay has the highest mean value(15.39) followed by NAS(12.47); NAS also has the highest median value(2.00). We can conclude that carrier's reasons such as maintenance or crew problems, aircraft cleaning, baggage loading, fueling, etc cause the longest delay. Other reasons' lengths are trivial compared to this.

Question 3: Portfolio modeling

We constructed three different portfolios of exchange-traded funds, or ETFs, and used bootstrap resampling to analyze the short-term tail risk of our portfolios. Each of these three portfolios is based on a distinct method of watching market behavior. We'll determine the 5% value at risk for each portfolio using 20 trading day bootstrap resampling on a $100,000 capital investment utilizing the last five years of daily data. At the end of the day, each of these portfolios is redistributed to maintain the given portfolio weights.

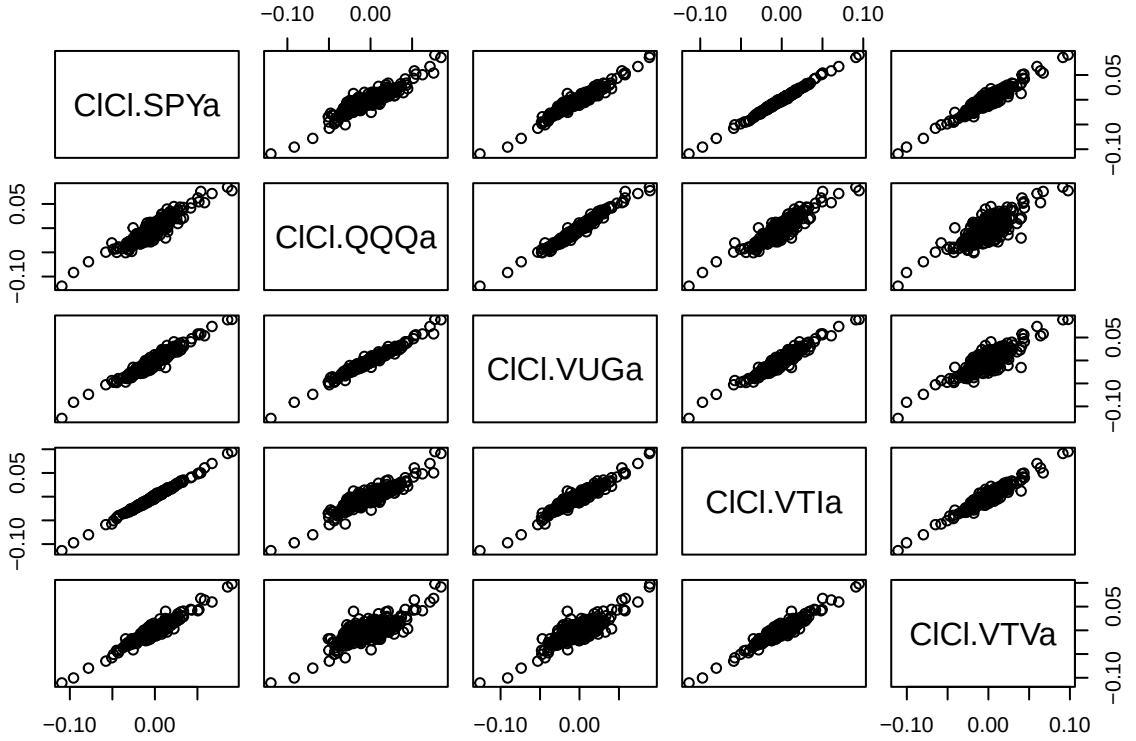
Portfolio 1: "Safe Portfolio - Capture the market return"

Portfolio 1 is built on the idea of tracking the market passive movement. The main idea to follow SP&500, Nasdaq market return, and include other ETFs that also has a huge market cap which means they are normally more safe.

- **20% SPY** SPY has long been one of the best ETFs to invest in; it is the single largest ETF in terms of assets under management (AUM), with over $360 billion. It's incredibly liquid with over 74 million shares traded per day. It is one of the simplest ways to invest on major business in the United States. SPY is an excellent location to lodge money for both new and experienced investors. It will give us returns that almost closely match the market without requiring any investigation.
- **20% QQQ** The Invesco QQQ ETF, which tracks the Nasdaq-100, an index of the Nasdaq Stock Market's 100 largest nonfinancial members, is another failsafe way to track a significant index.
- **20% VUG** VUG ETF concentrates on large-cap U.S.-based growth equities, and its largest holdings include Apple (AAPL), Microsoft (MSFT), Amazon.com (AMZN), Facebook (FB), and Alphabet (GOOG, GOOGL), despite VUG having 280 different stocks.
- **20% VTI** keep track of the CRSP US Total Market Index's performance. Growth and value styles are represented in large, mid, and small-cap equity.Uses an index-sampling approach that is controlled passively.
- **20% VTV** supplement of VTI index.The CRSP US Large Cap Value Index, which measures the investment return of large-capitalization value equities, is being tracked. Provides a simple way to track the performance of many of the country's most popular value stocks.

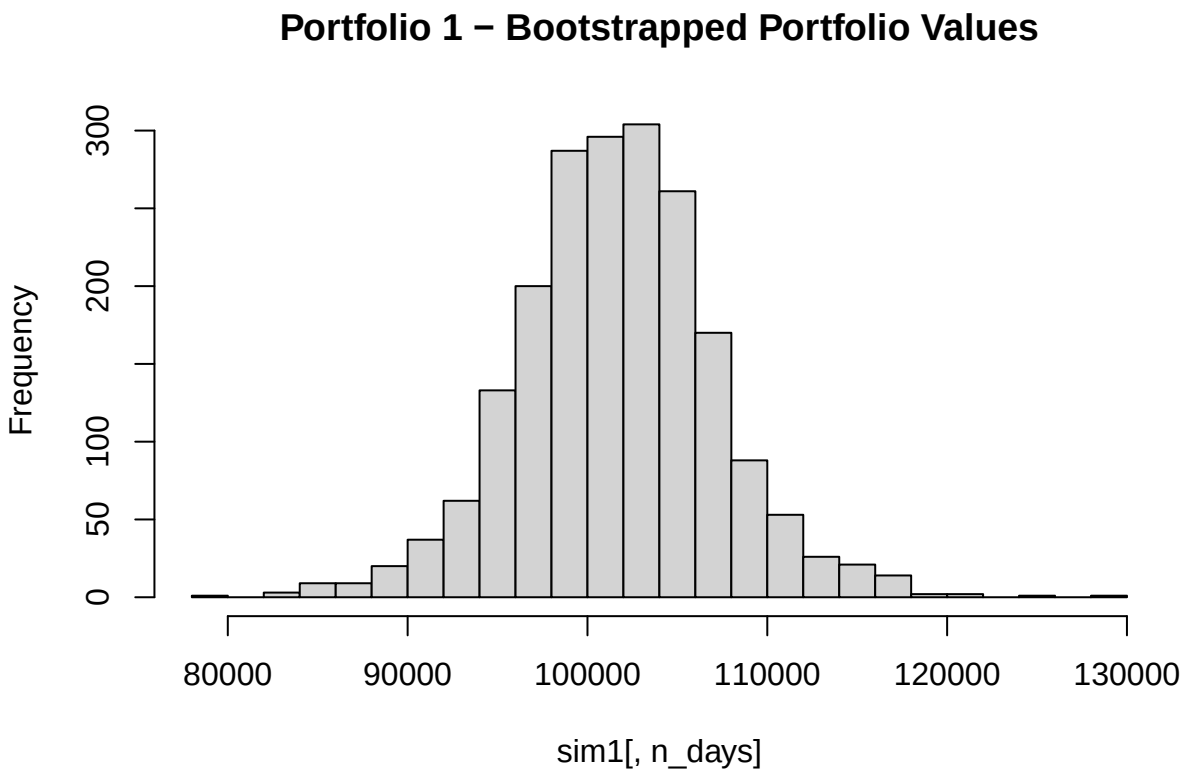
```
##            SPY.Open SPY.High  SPY.Low SPY.Close SPY.Volume SPY.Adjusted
## 2016-08-15 199.4942 200.0501 199.4851  199.6765   49813500     199.6765
## 2016-08-16 199.2299 199.3028 198.6466  198.6466   53213600     198.6466
## 2016-08-17 198.6831 199.1661 197.7899  199.0203   75134300     199.0203
## 2016-08-18 198.9929 199.5033 198.8745  199.4669   52989300     199.4668
```

```
## 2016-08-19 198.9656 199.3666 198.4461  199.1752   75443000     199.1752
## 2016-08-22 198.9200 199.4122 198.5281  199.1661   61368800     199.1661
```

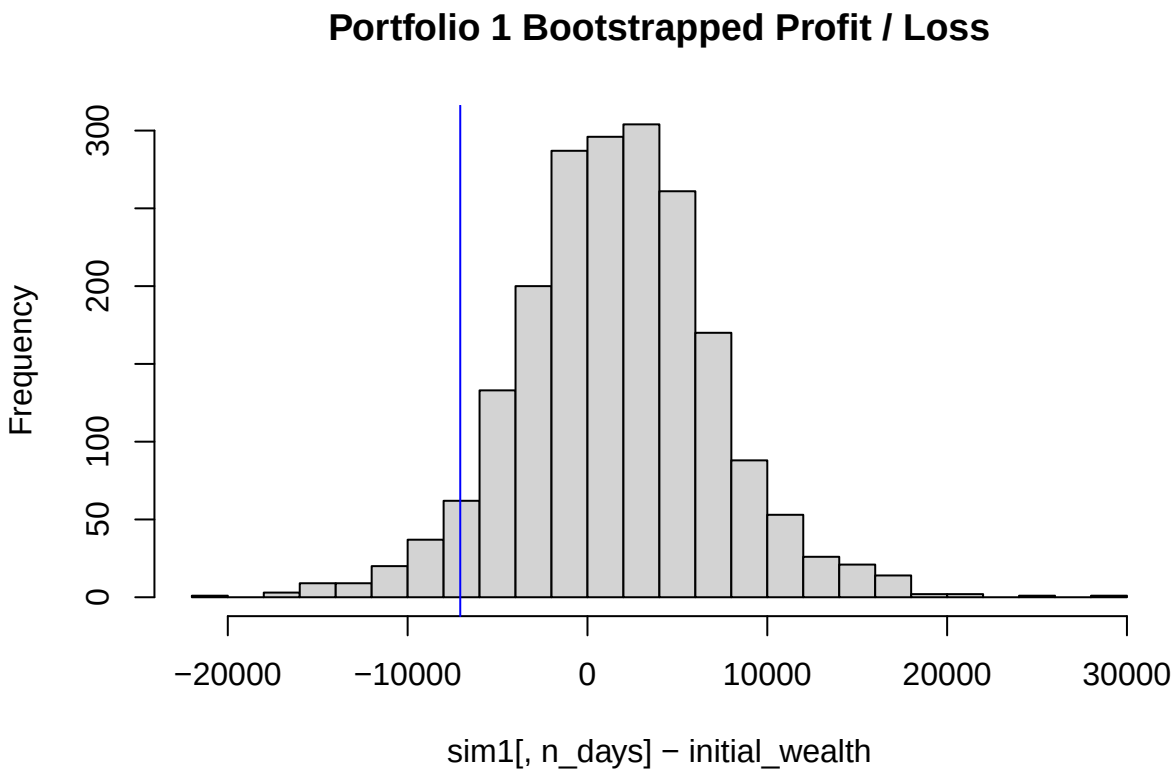


The Close-to-Close earnings of each of these ETFs are highly connected, as shown in the pairs correlation matrix above. As a result, as one goes up, the others follow suit. They all fall down when one goes down. This shows that all of these ETFs are following market trends and going in the same direction.

Then we simulate the portfolio's 20-day trading term.


Portfolio 1 − Bootstrapped Portfolio Values
Frequency
0
50
100
200
300
80000
90000
100000
110000
120000
130000
sim1[, n_days]

```
## [1] 1607.895
```




```
##        5%
## -7074.158
```

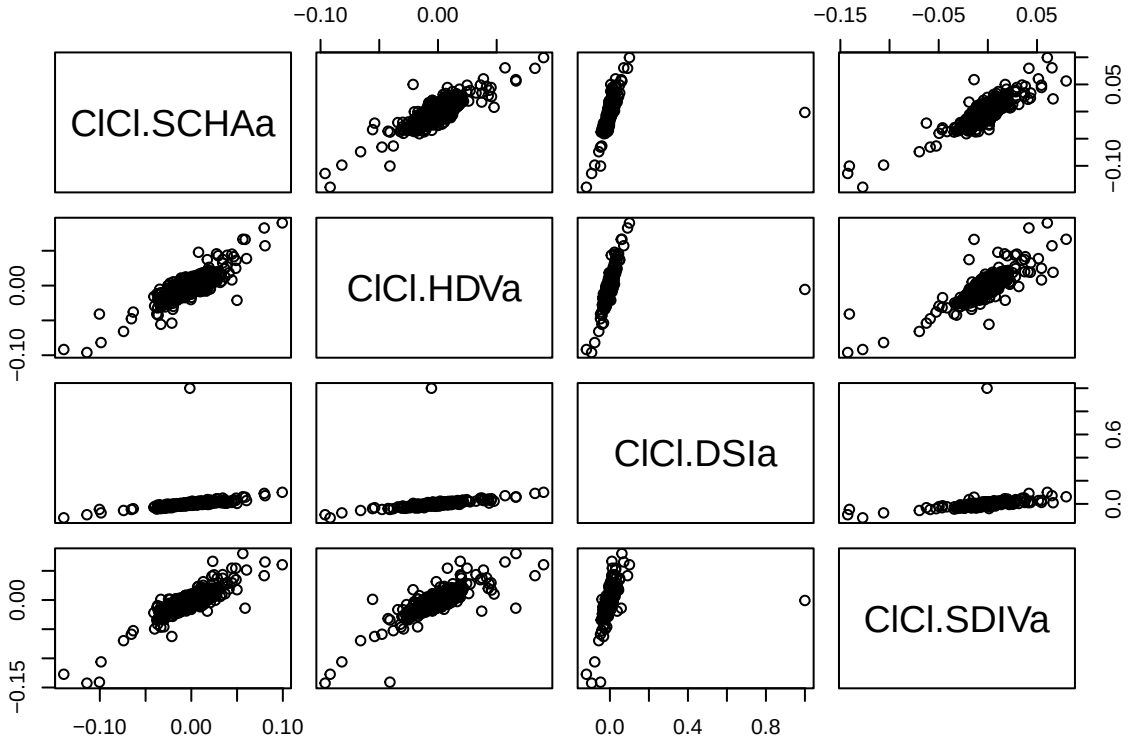
Based on the histograms of portfolio values and earnings, we can see that, while there is still a chance of a loss after a 20-day bootstrapped period, the mean earnings still yield a profit of **about $1607.895**. For this portfolio, VAR at 5% over a 20-day bootstrapped period is **approximately $7074.158**.

Portfolio 2: "Aggressive portfolio - Risk Award"

Portfolio 2 is designed to profit from market volatility. These exchange-traded funds (ETFs) are designed for income investors survive in volatile market situations. The portfolio is broken down as follows:

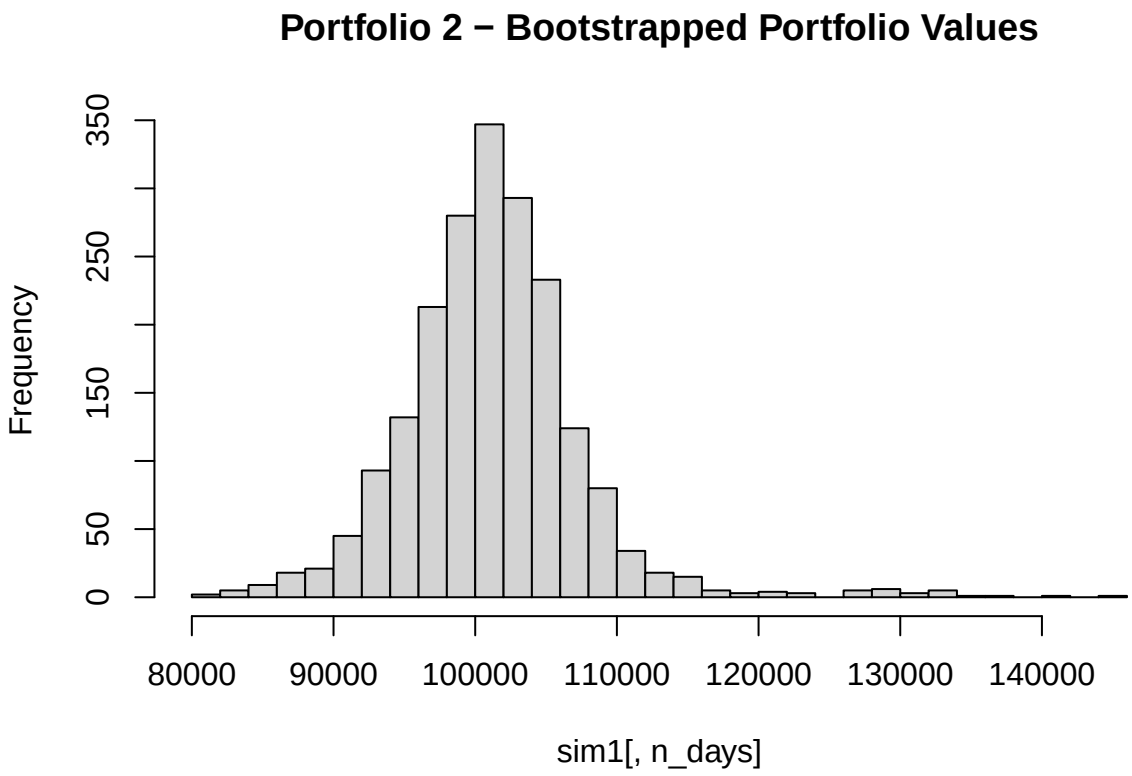
- **25% SCHA** SCHA can give investor access to the stock market in the United States, but it only invests in smaller companies, which have a higher growth potential than much larger corporations. It has around 1,850 holdings and a market value of $4.6 billion, making it a very well-diversified fund. 10x Genomics (TXG), a gene sequencing firm, and Darling Ingredients (DAR), a rendering company that converts animal byproducts into valuable end products, are two of its top ten holdings.
- **25% HDV** iShares Core High Dividend ETF is designed for income investor. With 74 holdings in its portfolio, it is more concentrated than some of the other funds on this list, but its cost ratio is just 0.08 percent, and its dividend yield is 3.7 percent - a nice figure in the current age of low interest rates that the US has been in for almost a decade. This is one of the best ETFs to buy because of its strong yield and solid stock portfolio. Exxon Mobil (XOM), Johnson & Johnson, and Verizon Communications are three of HDV's major holdings (VZ).

- **25% DSI** iShares MSCI KLD 400 Social ETF (DSI): over the last ten years, the ETF has returned an annualized 13.8 percent. The fund focuses on enterprises situated in the United States that have an environmental, social, and governance (ESG) bent, eliminating businesses such as alcohol, firearms, gambling, tobacco, and genetically modified organisms. It has a 0.25 percent expense ratio, 404 positions, and Microsoft, Alphabet, Tesla (TSLA), and Nvidia Corp. (NVDA) are among its top holdings.
- **25% SDIV** Global X SuperDividend ETF is a mutual fund that invests in superdividend. It invests in around 100 of the world's highest-dividend-yielding equities assets. SDIV not only pays a high yield, but it also pays once a month, ensuring a steady stream of paydays. The yield is definitely high, but these equities clearly have a higher risk profile than established U.S. corporations.

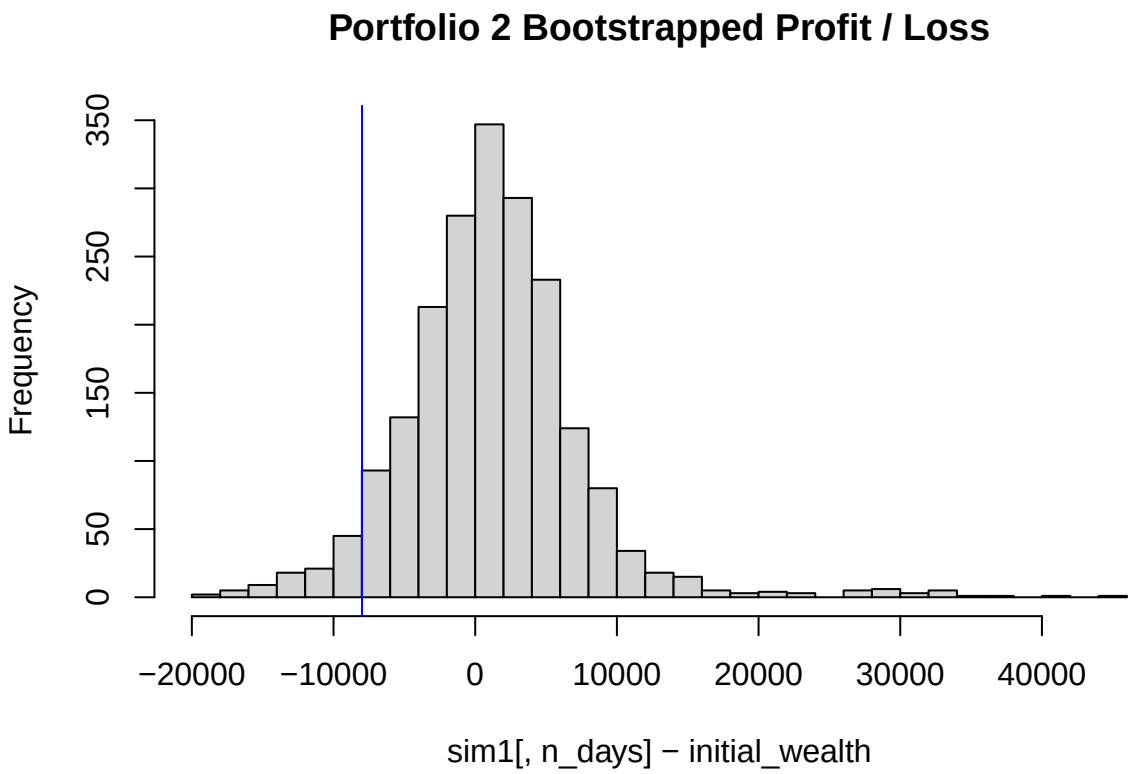


From the pairings plot, we observe that these four ETFs are substantially less correlated than the ETFs in Portfolio 1. As a result, we anticipate greater volatility and a higher Value at Risk in this portfolio.

Then, we simulate the 20-day trading period of this portfolio.


Portfolio 2 − Bootstrapped Portfolio Values
Frequency
0
50
150
250
350
80000
90000
100000
110000
120000
130000
140000
sim1[, n_days]

```
## [1] 1231.849
```




```
##        5%
## -7987.424
```

Based on the histograms of portfolio values and earnings, we can see that, while there is still a chance of a loss after a 20-day bootstrapped period, the mean earnings still yield a profit of **about $1231.849**. For this portfolio, VAR at 5% over a 20-day bootstrapped period is **approximately $7987.424**. Although portfolio 2 generates a lower mean profit with a larger VAR than portfolio 1. We can witness a small probability to generate a profit more than **$40000** which is unable to be achieved by portfolio 1.

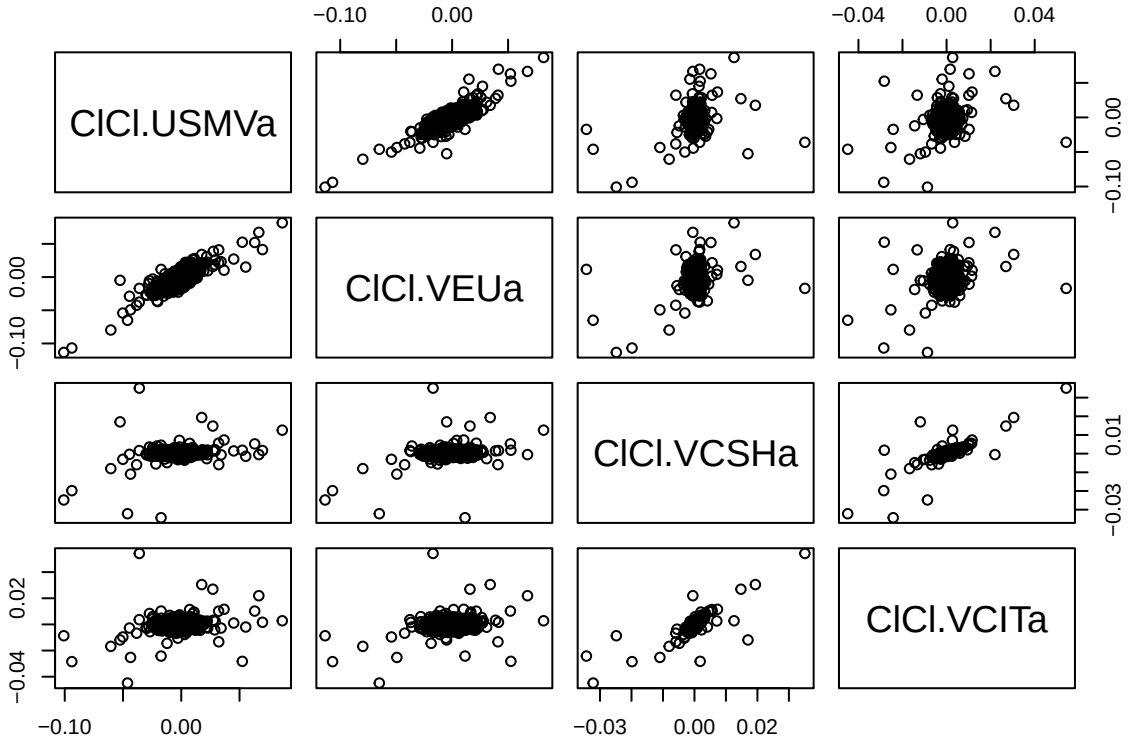
Portfolio 3: "Diversified Portfolio - Risk Averse"

Portfolio 3 seeks to avoid market volatility to a large instance, whereas Portfolio 2 was established on the premise of profiting from market volatility. These "safe" ETFs are meant to survive market volatility, but their prospective returns are lower. The portfolio is broken down as follows:

- **25% USMV** suitable for investors who participate in the stock market but don't want to deal with the volatility, provide a large portion of the gains that an investment in the SP500 would provide, but with a far lower portion of the risk. It achieves so by investing in equities with a reduced volatility history, such as Eli Lilly (LLY), Kroger (KR), Waste Management (WM), and Johnson & Johnson (JNJ), which are among its top holdings. Since its inception in 2011, USMV has achieved its primary goal, collecting 76 percent of stock market upside and only 62 percent of stock market fall.
- **25% VEU** Vanguard FTSE All-World ex-US ETF is an exchange-traded fund that tracks the performance of the FTSE All-World index (VEU). Switching investments geographically is one of the most effective ways to genuinely diversify, as it exposes you to different economies and currencies.

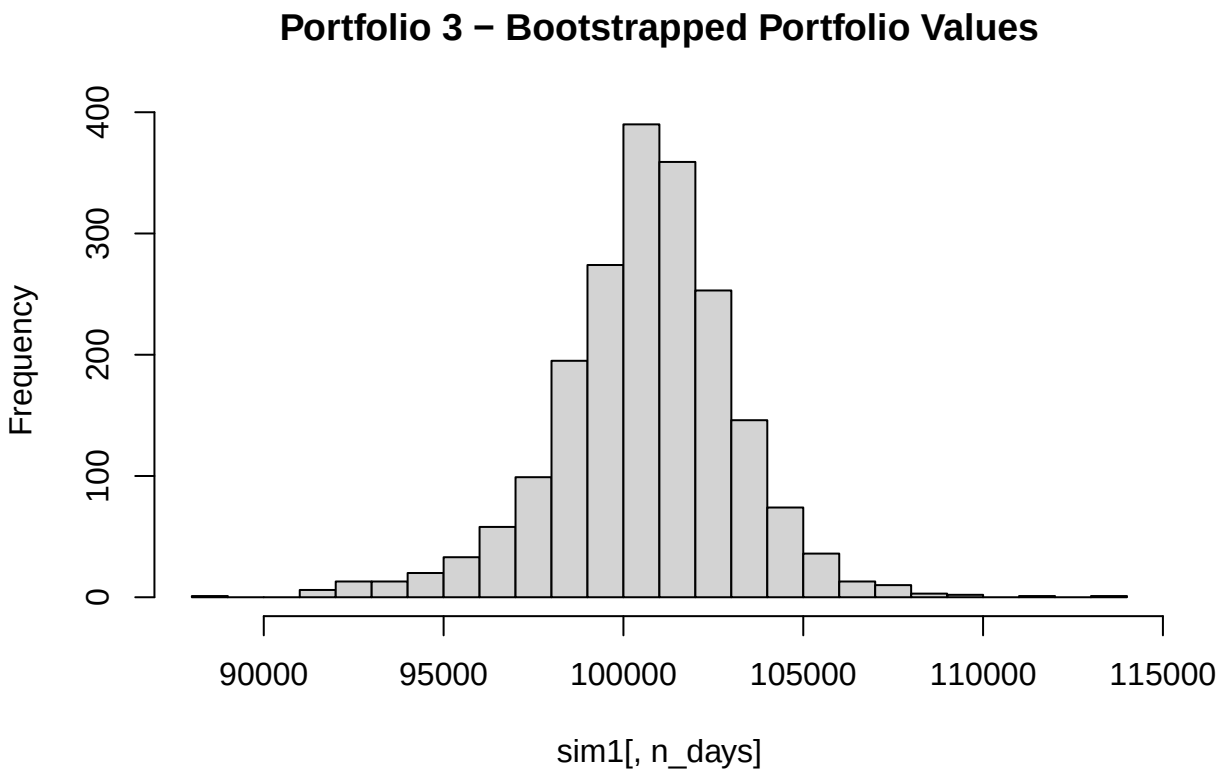
With the exception of the United States, VEU has almost 3,500 stocks across all continents. Europe (40.2 percent), the Pacific (27.2 percent), and emerging markets (27.6 percent) are the three largest geographical allocations (25.6 percent).

- **25% VCSH** Vanguard Short-Term Corporate Bond ETF is a mutual fund that invests in short-term corporate bonds (VCSH) With an emphasis on high-quality corporate debt and the average bond in the portfolio due in just three years, investors may be confident that their loans will be fully repaid. Although the yield is lower than that of a typical SP 500 dividend stock, the stability you'll find in this short-term bond fund is likely to outweigh the volatility you'll find in the stock market. With $41 billion in assets, this safe-haven fund is a top choice for many risk-averse investors.
- **25% VCIT** Vanguard Intermediate-Term Corporate Bond ETF is a mutual fund that invests in intermediate-term corporate bonds (VCIT). It provides a means to focus just on corporate bonds for individuals searching for a bit more return than an aggregate bond fund that is primarily weighted toward low-risk government bonds. T

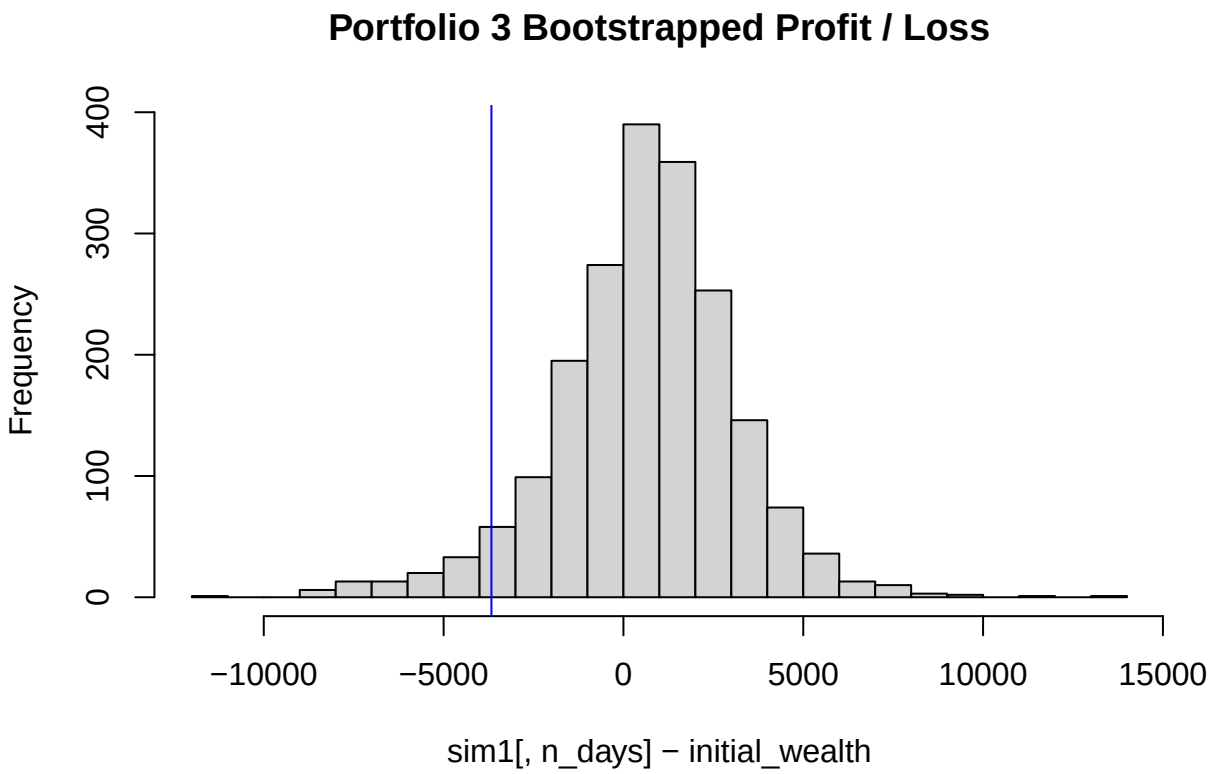


These ETFs are less correlated than the ETFs in Portfolio 1 and Portfolio 2, according to the pairs correlation matrix.

Then, we simulate the 20-day trading period of this portfolio.


Portfolio 3 − Bootstrapped Portfolio Values
Frequency
0
100
200
300
400
90000
95000
100000
105000
110000
115000
sim1[, n_days]

```
## [1] 630.5331
```




```
##        5%
## -3670.662
```

Based on the histograms of portfolio values and earnings, we can see that, while there is still a chance of a loss after a 20-day bootstrapped period, the mean earnings still yield a profit of **about $630.5331**. For this portfolio, VAR at 5% over a 20-day bootstrapped period is **approximately $3670.662**. We witnessed a narrower range for the profit/less, generated a less VAR and less return, witch is consistent with the concept of the portfolio 3 risk averse portfolio construction.

In summary

- Portfolio 1 is excellent for long-term investing. The 5% VaR and the return are the most balanced. And according to many theories, passive investment in market index is the best investment strategy in a long term. Mean earnings **about $1607.895**, VAR at 5% over a 20-day bootstrapped period is **approximately $7074.158**.
- Portfolio 2 is a superior risk fund for the income investors who are willing to generate a higher return at a cost of potential loss. Mean earnings **about $1231.849**, VAR at 5% over a 20-day bootstrapped period is **approximately $7987.424**, with the potential to generate a highest profit more than **$40000**.
- Portfolio 3 is the safest option for risk-averse investors over a 20-day trading period. Mean earnings **about $630.5331**, VAR at 5% over a 20-day bootstrapped period is **approximately $3670.662**.

In general, these portfolios are suitable for investors with different preference. Investors may be further diversify their portfolios by investing in these 3 portfolios simultaneously. We suggests them to invest 65% in portfolio 3, 25% in portfolio 1, and 10% in portfolio 2 to build up a more rounded total portfolio.

Question 4: Market segmentation

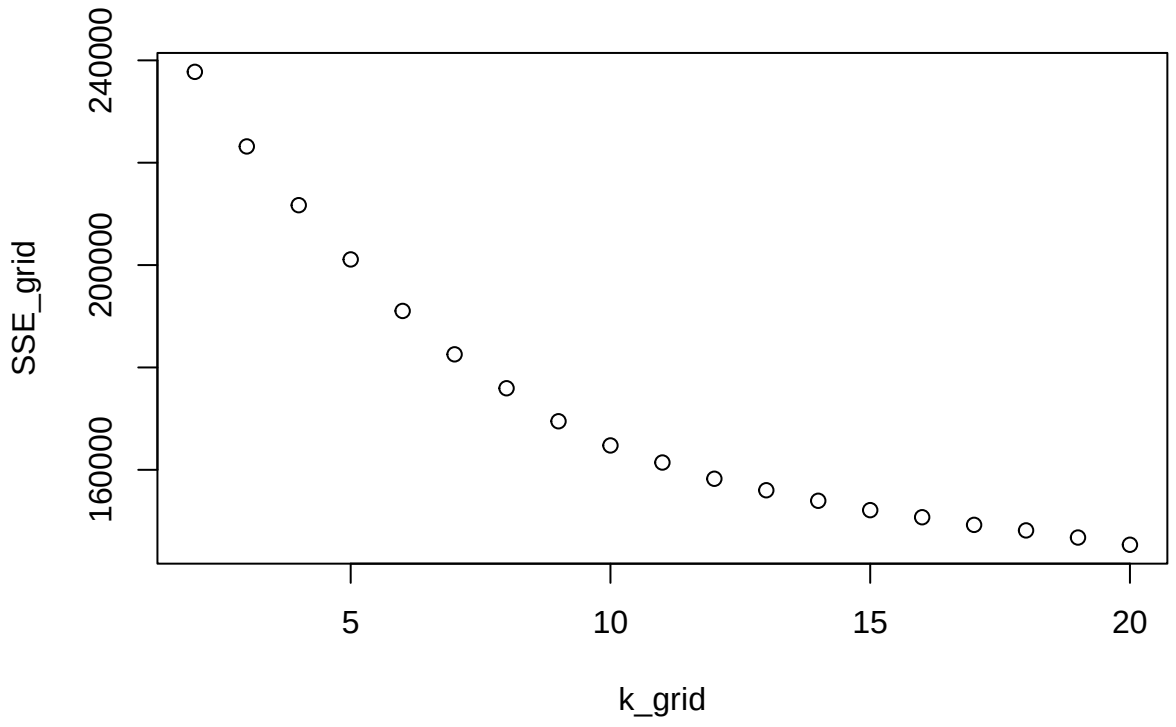
Our approach to segment the market is to use both kmeans clustering and hierachical clustering to split the users into differnt based on their characteristics. Then, we will report the market segments we have and analyze how the company could potentially take advantage of this information.

K-means Clustering

To start with, we must decide the optimal number of K, which impacts the interpretability and efficiency of our model. Although we can't cross validate, we will use two quantitative model selection criteria - elbow plot and CH index. In case that the results from these two techniques don't agree, we will use the "satisfice, don't optimize" rule and hand pick a suitable K value. We first per-processed our dataset by getting rid of the three useless columns.

```
## [1] 7882   34
```

```
## Warning: did not converge in 10 iterations
```




```
## Warning: did not converge in 10 iterations
```

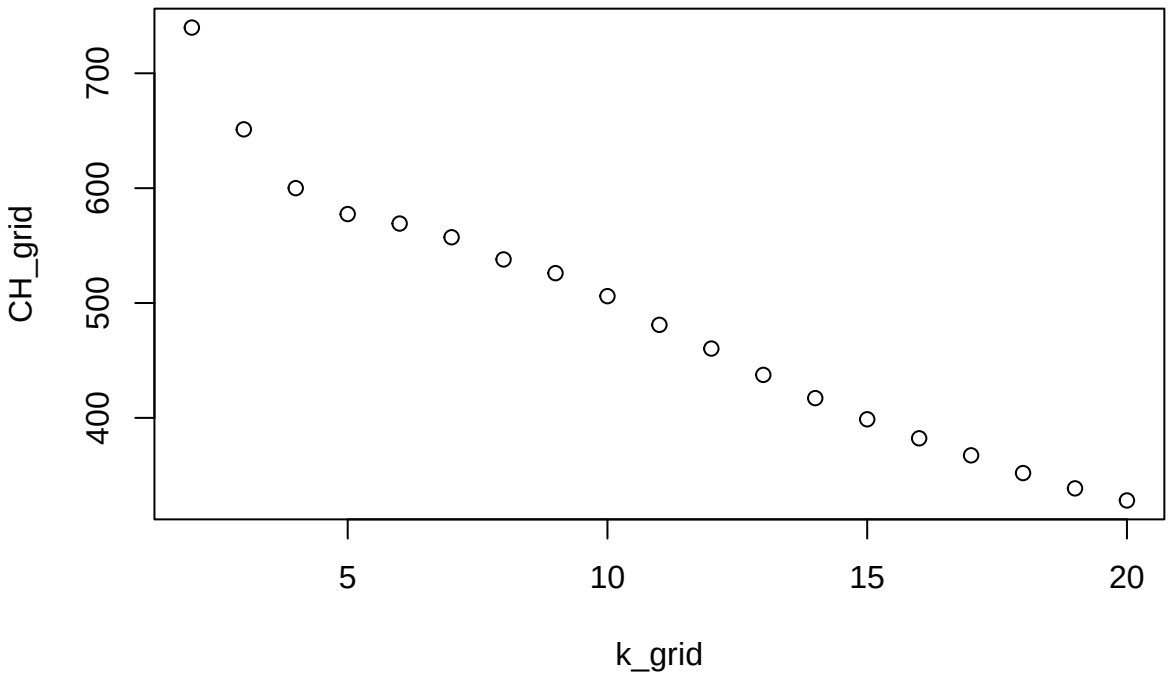
```
## Warning: did not converge in 10 iterations
```

```
## Warning: did not converge in 10 iterations
```

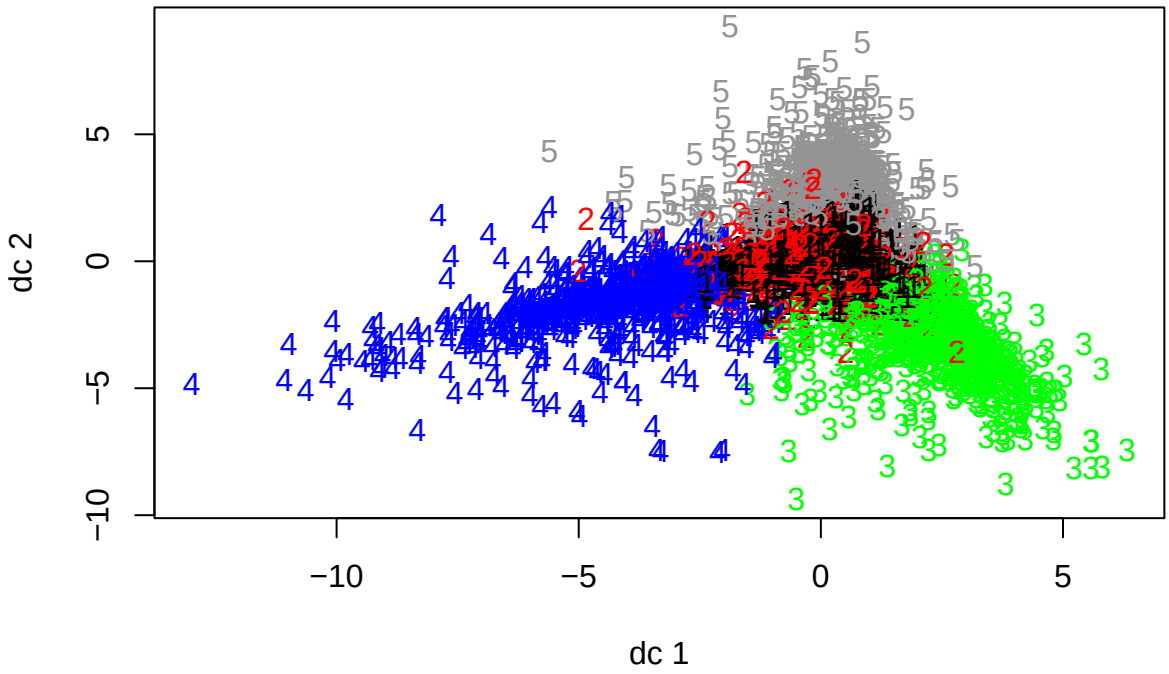
```
## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations
```



Considering the results from the two plots (we want larger CH index and smaller SSE) and the quality of our model, we decide that the number of clusters to be 5.



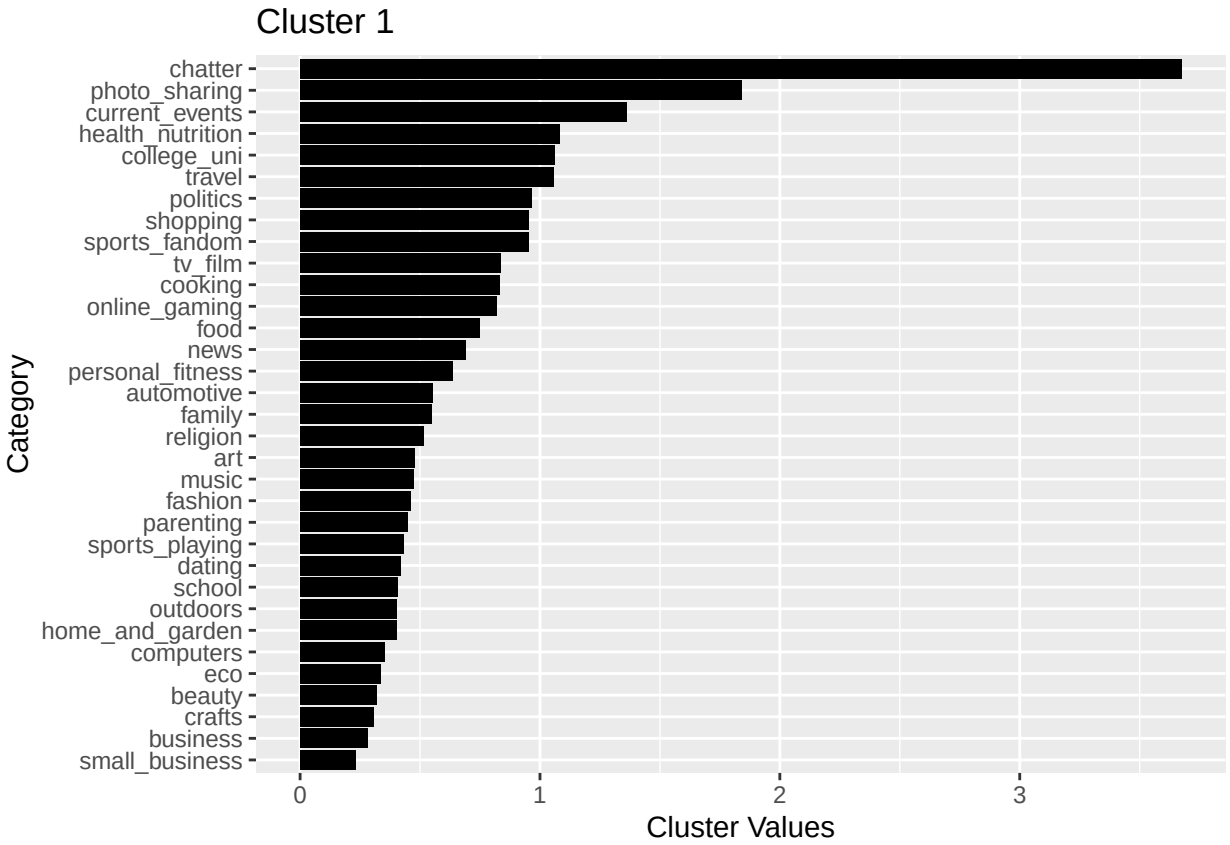

```
## [1] 201103.6
```

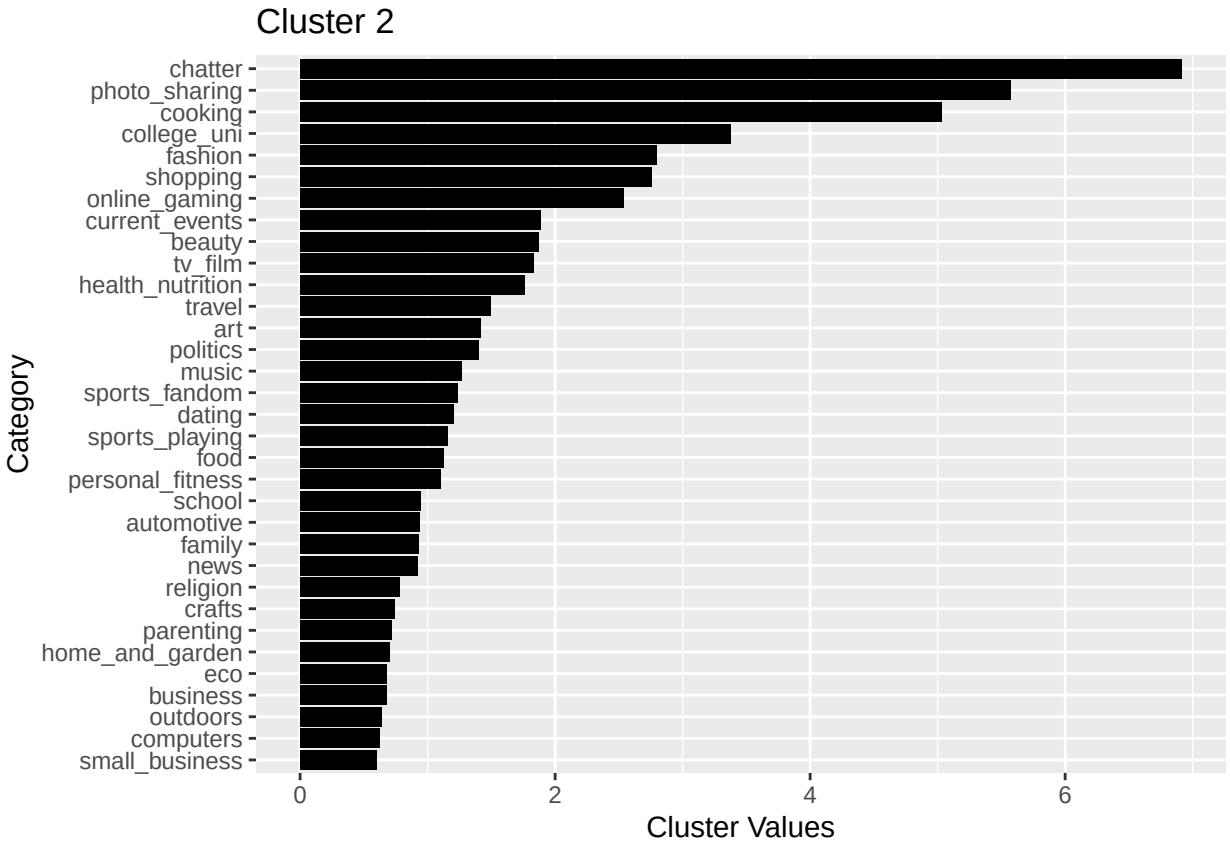
```
## [1] 58969.44
```

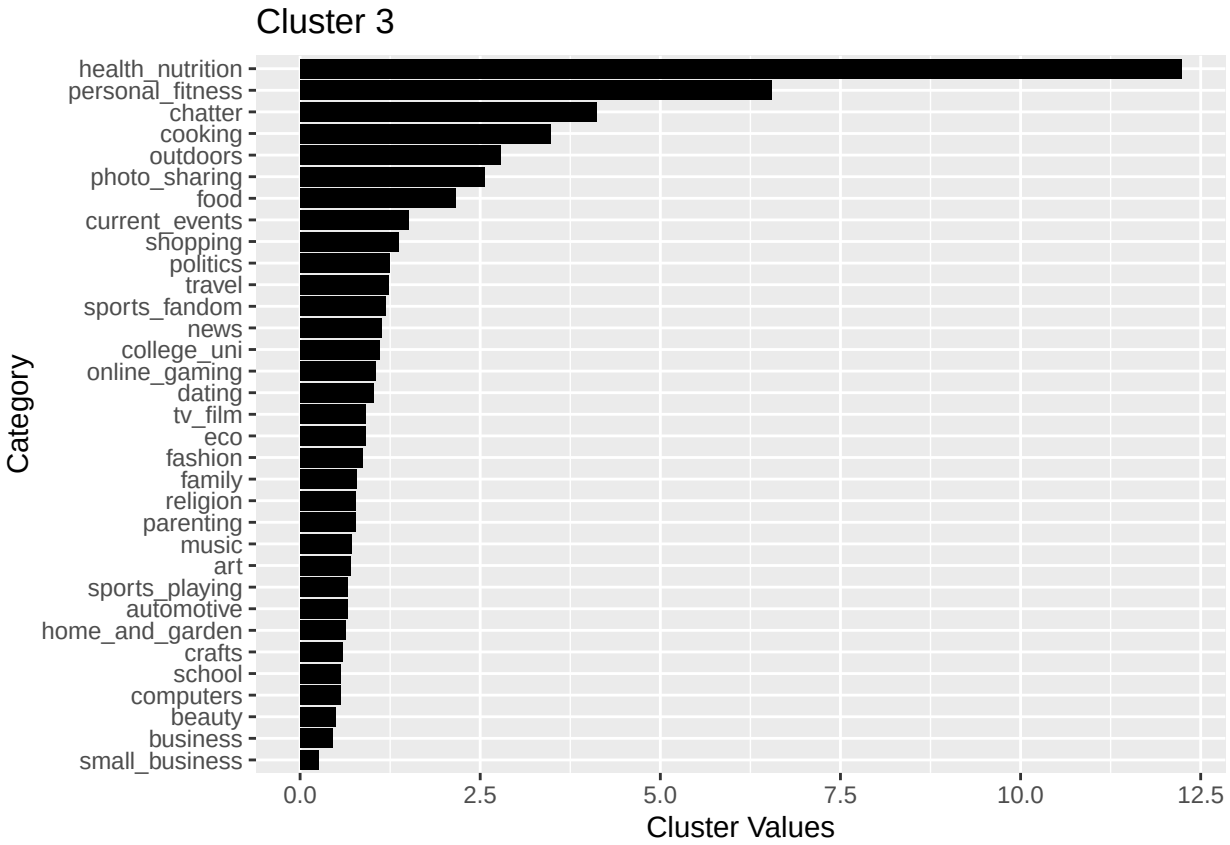
```
##  [1]  2  4  5  6 11 14 15 16 19 20 22 23 24 25 26 27 29 31 35 36
```

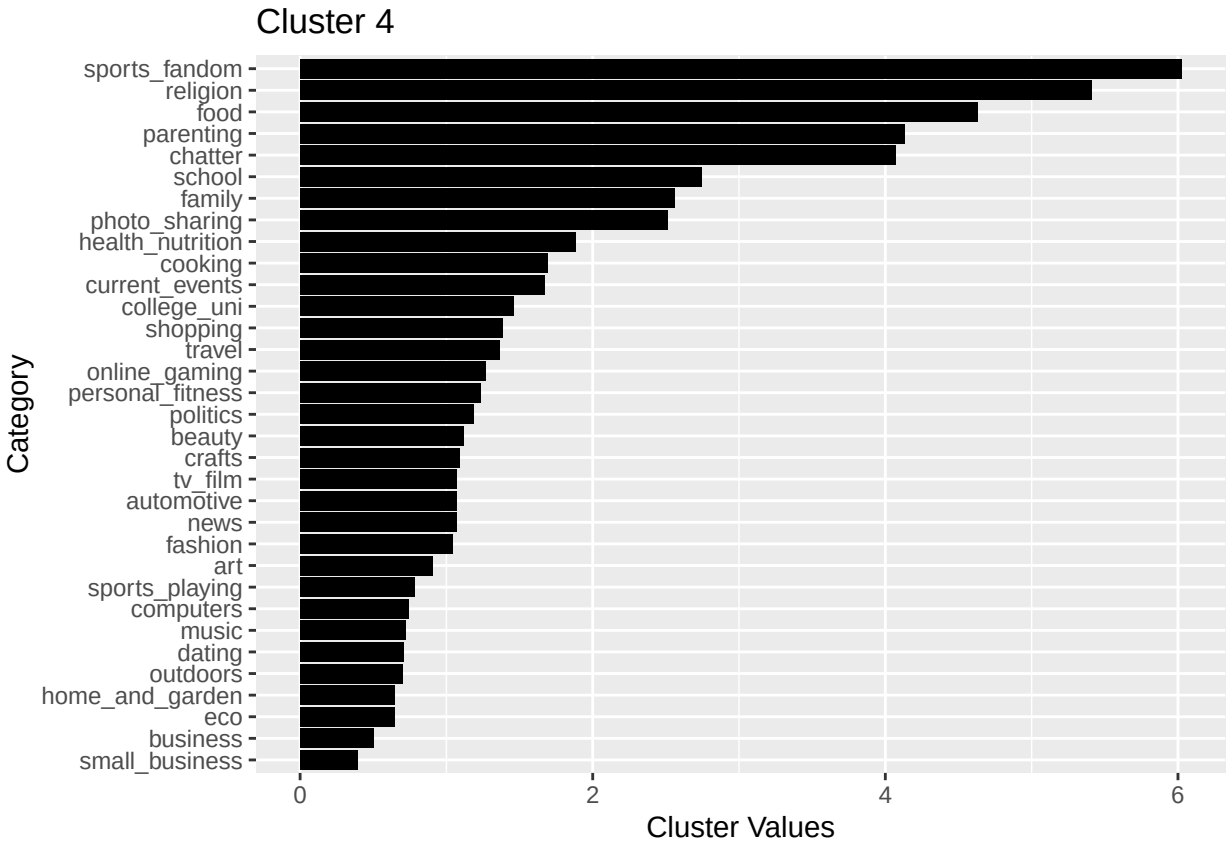
After running kmeans clustering, we plot the data points in different colors and obtained a total withiness of 201103.6 and betweeness of 58969.44.

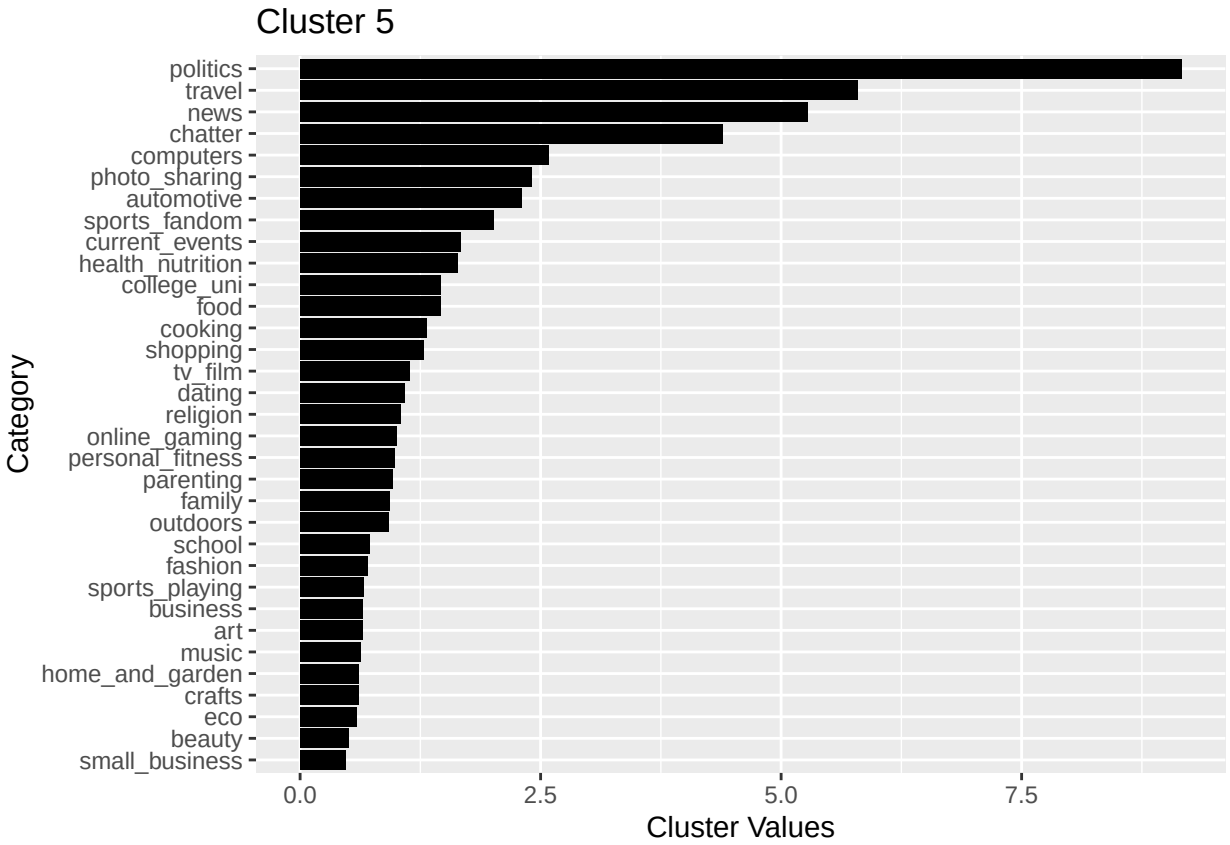
Now we begin analyzing the key features representing each cluster.


Cluster 1
chatter
photo_sharing
current_events
health_nutrition
college_uni
travel
politics
shopping
sports_fandom
tv_film
cooking
online_gaming
food
news
personal_fitness
automotive
family
religion
art
music
fashion
parenting
sports_playing
dating
school
outdoors
home_and_garden
computers
eco
beauty
crafts
business
small_business
Category
0
1
2
3
Cluster Values


Cluster 2
chatter
photo_sharing
cooking
college_uni
fashion
shopping
online_gaming
current_events
beauty
tv_film
health_nutrition
travel
art
politics
music
sports_fandom
dating
sports_playing
food
personal_fitness
school
automotive
family
news
religion
crafts
parenting
home_and_garden
eco
business
outdoors
computers
small_business
Category
0
2
4
6
Cluster Values


Cluster 3
health_nutrition
personal_fitness
chatter
cooking
outdoors
photo_sharing
food
current_events
shopping
politics
travel
sports_fandom
news
college_uni
online_gaming
dating
tv_film
eco
fashion
family
religion
parenting
music
art
sports_playing
automotive
home_and_garden
crafts
school
computers
beauty
business
small_business
Category
0.0
2.5
5.0
7.5
10.0
12.5
Cluster Values

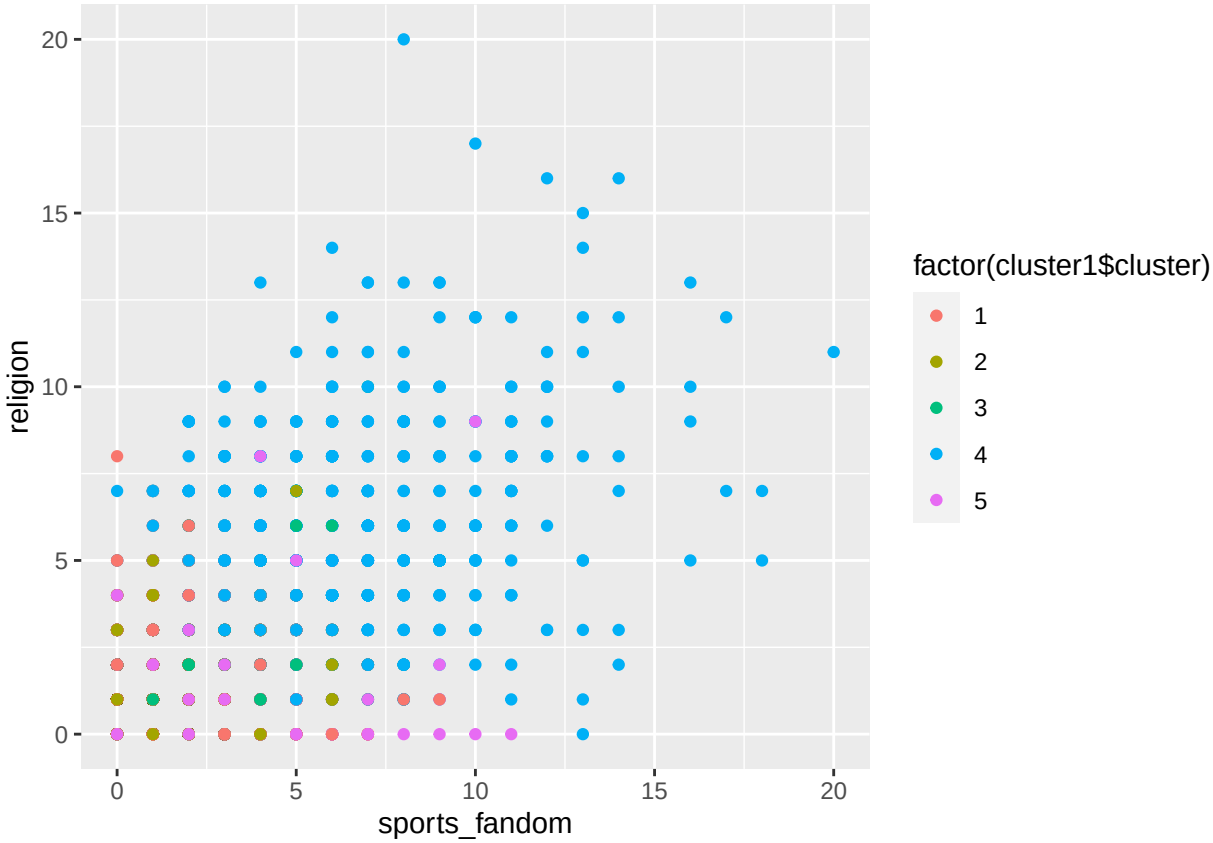

Cluster 4
sports_fandom
religion
food
parenting
chatter
school
family
photo_sharing
health_nutrition
cooking
current_events
college_uni
shopping
travel
online_gaming
personal_fitness
politics
beauty
crafts
tv_film
automotive
news
fashion
art
sports_playing
computers
music
dating
outdoors
home_and_garden
eco
business
small_business
Category
0
2
4
6
Cluster Values

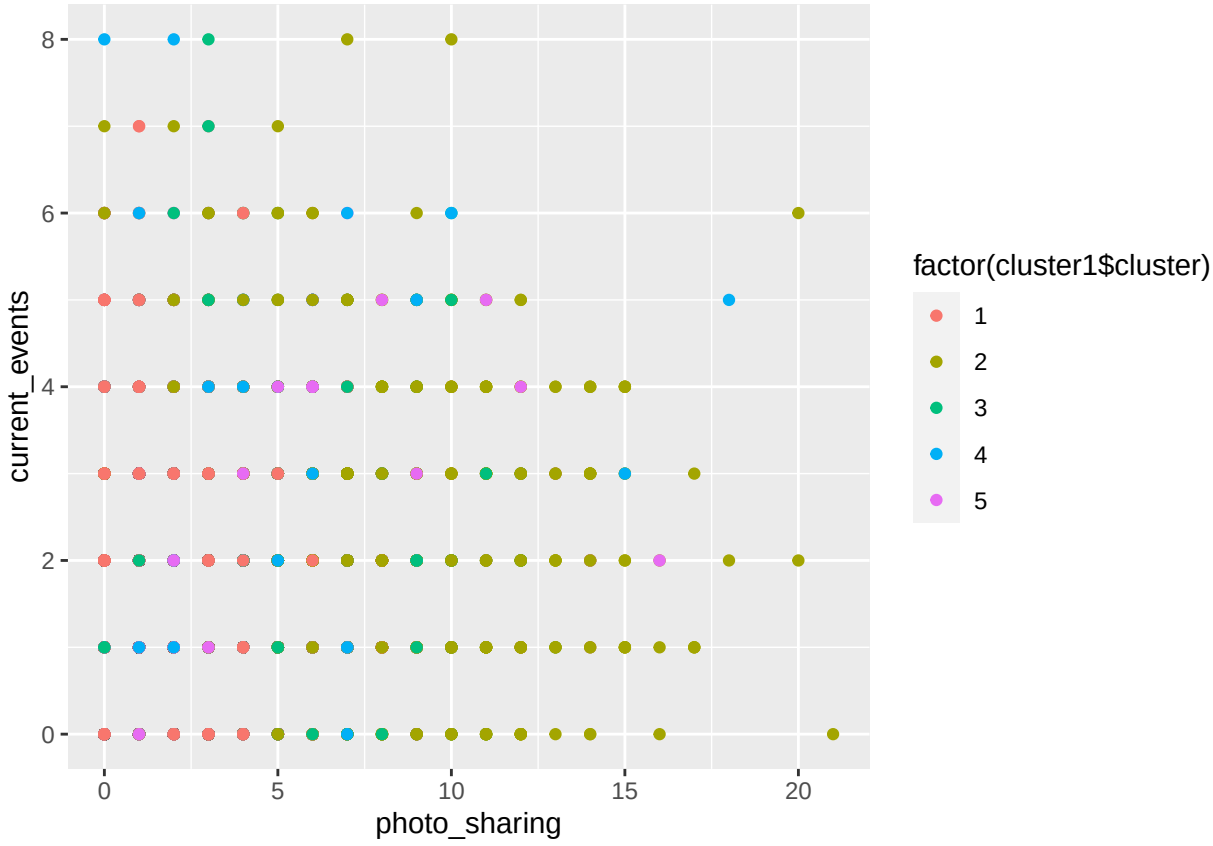


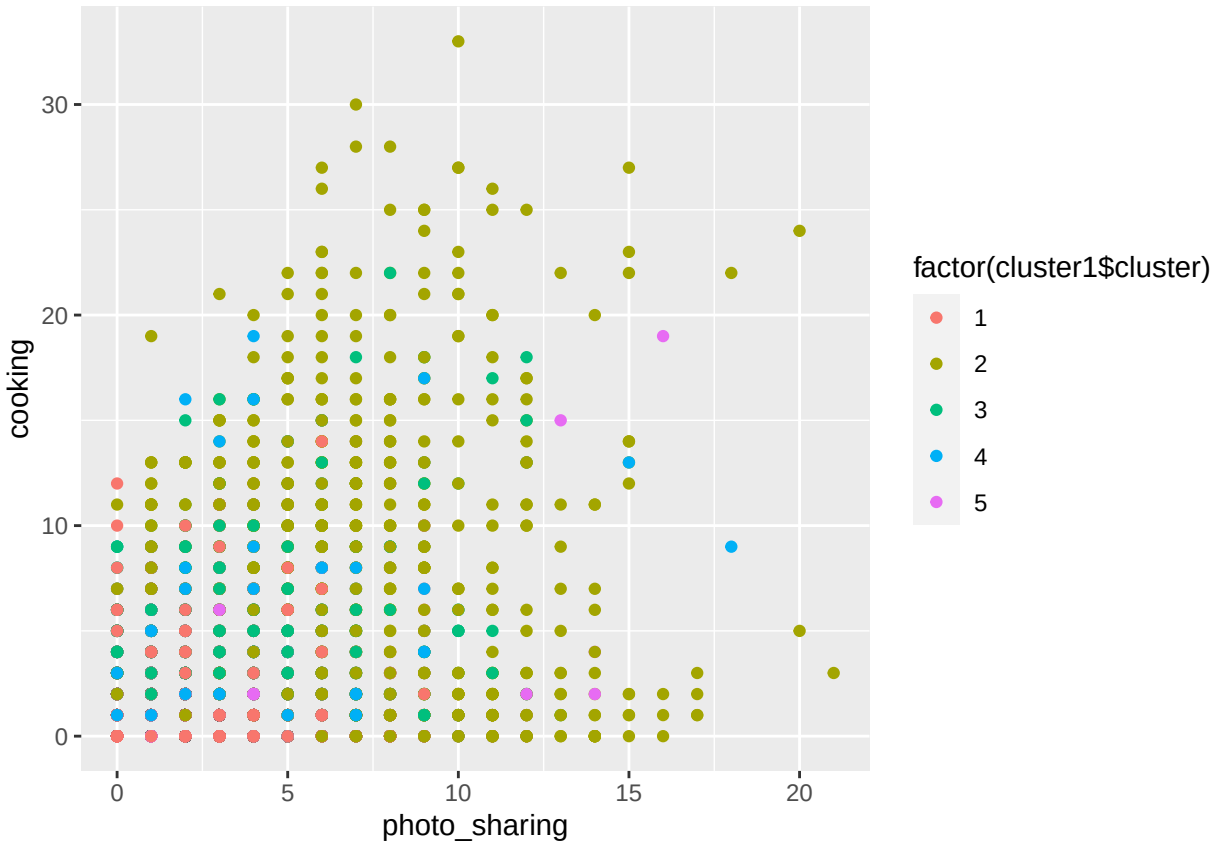
The five market segments we found are characterized by the following features: Cluster 1: sports fandom, religion, food Cluster 2: chatter, photo sharing, current events Cluster 3: chatter, photo sharing, cooking Cluster 4: health nutrition, personal fitness, chatter Cluster 5: politics, travel, news

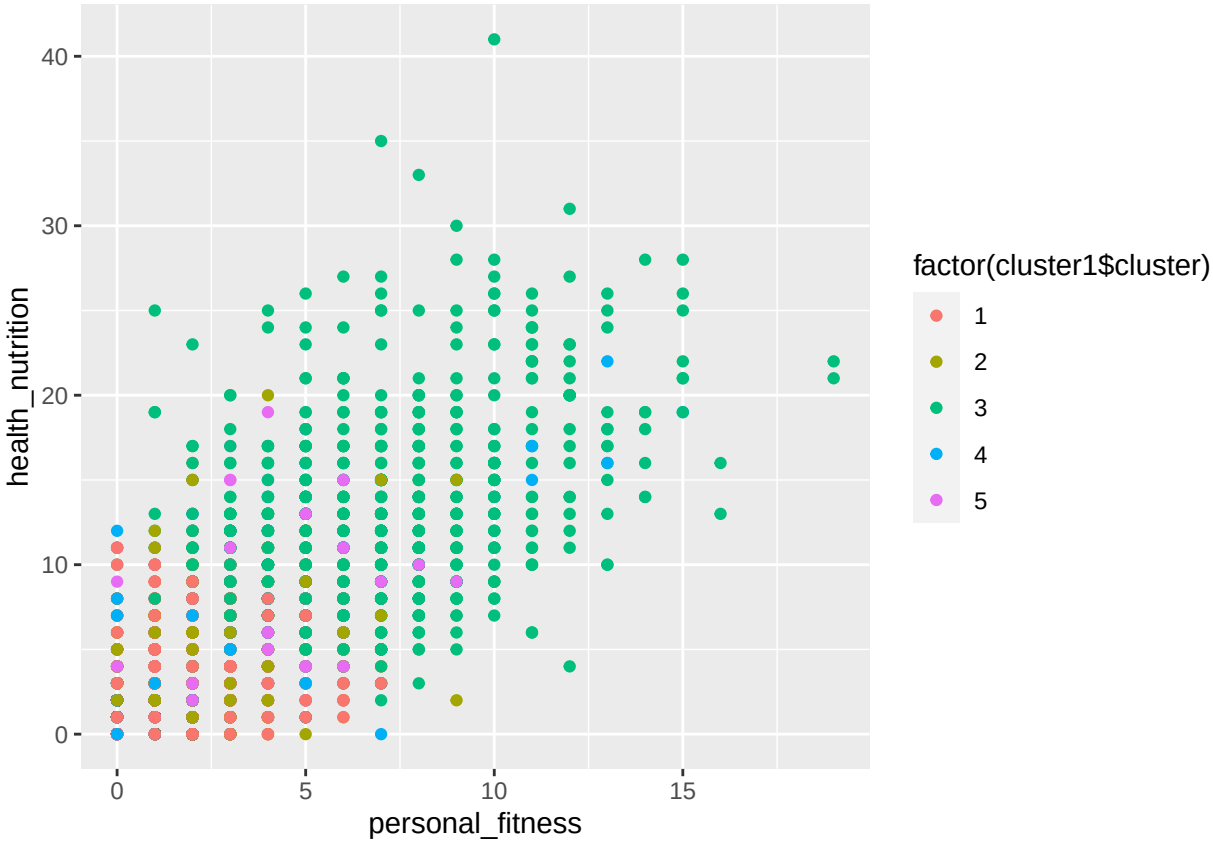
- Cluster 1 is characterized by religious, sports fan users who love exploring food; We can consider recommending religions-related cuisines or sports to them.
- Cluster 2 is represented by active users who share photos frequently and care about current events; They are likely to be interested in the latest news from social platform, so we can utilize social media to reach them.
- Cluster 3 is dominated by active users who love posting photos and cooking; We can push advertisements about food recipes or TV shows about food to them.
- Cluster 4 is a cluster of active users who like working out, body building, and healthy lifestyle; We should consider presenting products like protein bars, organic/fresh fruits to them.
- Cluster 5 is characterized by users who are passionate about politics, watching news, and travelling aroung the world; Political talk shows and news report about foreign countries might be their favorties.

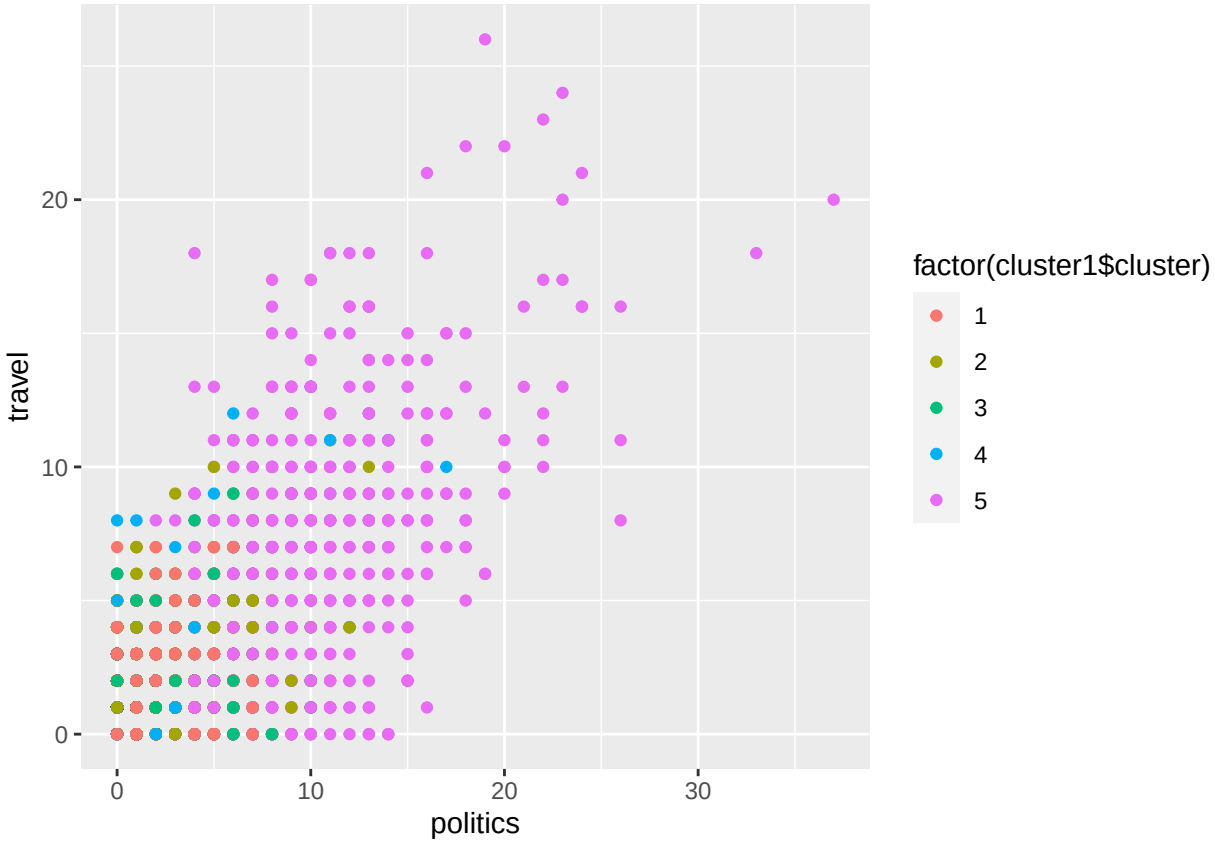
To reinforce our conclusions above, we also make a few plots with cluster membership.


religion
sports_fandom
0
5
10
15
20
factor(cluster1$cluster)
1
2
3
4
5


current_events
photo_sharing
0
2
4
6
8
0
5
10
15
20
factor(cluster1$cluster)
1
2
3
4
5


cooking
photo_sharing
0
10
20
30
0
5
10
15
20
factor(cluster1$cluster)
1
2
3
4
5

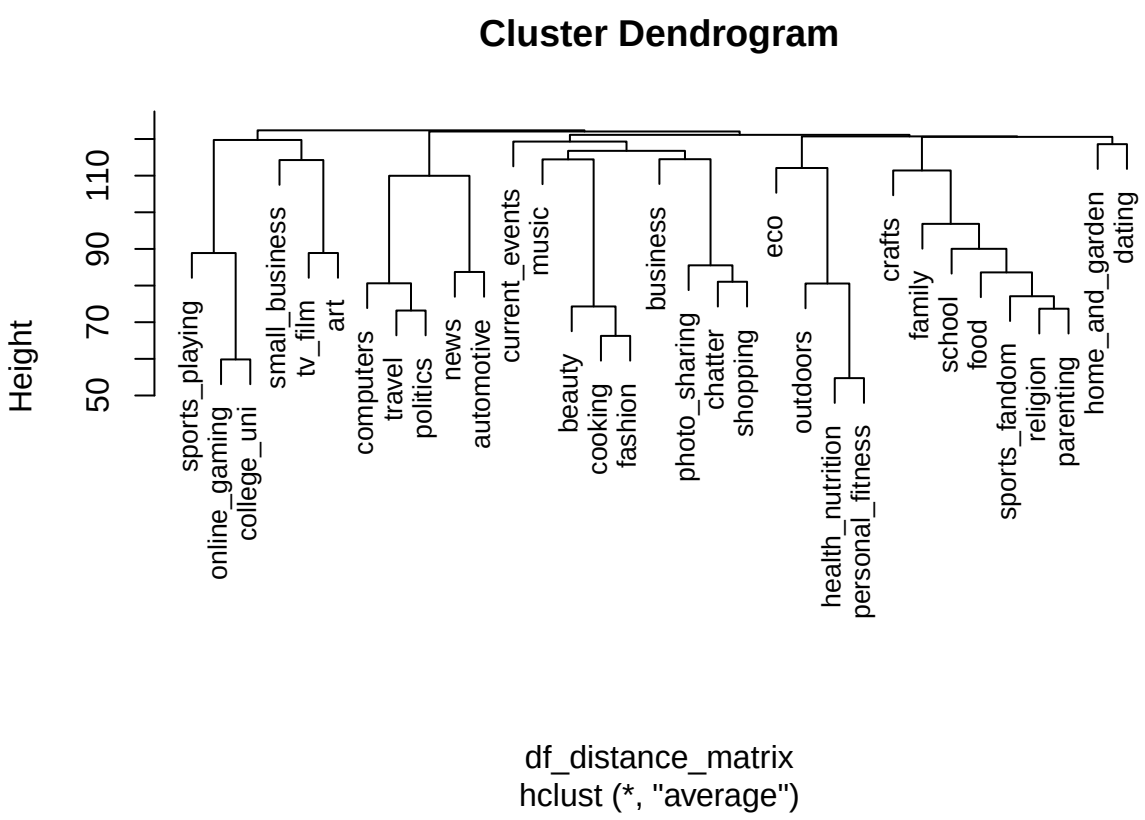

health_nutrition
personal_fitness
0
10
20
30
40
0
5
10
15
factor(cluster1$cluster)
1
2
3
4
5



Based on the scatterplots above, we can clearly see that: * Blue dots(factor 4) represent the users who stand out as high in the combination of religion and sports fandom. * Purple dots(factor 5) represents the users who rank high in the photo sharing and current events combination. * Purple dots(factor 5) **also** indicates outstanding users in the combination of photo sharing and cooking. * Orange dots(factor 1) represents the users who have high degrees in both fitness and nutrition. * Yellow dots(factor 2) marks the users with high degrees in politics and travel.

Hierachical Clustering

We also want to use hierachical clustering on our dataset in order to compare our findings from k means clustering.




```
## 1 2 3 4 5
## 9 5 6 9 4
```

By examing the tree diagram above, we can identify the following market segments: * College students (those who like online games, sports, arts, college, etc) * Business professionals or politicians (users who travels, have automotives, watch news, and use computers) * Athletes, body builders, or models (do outdoor exercises, care about fitness and nutrition) * Middle class young females (who are passionate about fashion, listen to music, do online shopping, share photos about their life and active on the Internet). * Rebellious male teenagers (huge fan of sports starts, dating others, aganist parenting, mostly surrounded by family and school)

Overall, Kmeans and hierachical clustering obtained similar market segments. Monitoring twitter trends helps the company to gain a better understanding of approaching a specific customer and react to customers' preference changes. How the company could benefit from our analysis depends on what its target customers are and the functions nature of their products.

Question 5: Author attribution

In this question, we need to build the best performing model to predict the author of an article on the basis of that article's textual content. Our solution is divided in three parts. First part is the data preprocessing, including tokenization, building doc-term-matrixes for train and test data, and balanced the new words in test data. The second part is the PCA analysis for dimension reduction. The last part is modelling, where we used knn, random forests and naive bays models on our data, and then pick the best performing model based on the accuracy on test data.

Part 1 Data Preprocessing

First, we read in our train data, by rolling 50 directories together into a single corpus. We also cleaned up the file names and renamed the articles. We also built a corpus for all the documents.

Second, we took some pre-processing/tokenization steps, including: 1) make everything lowercase 2) remove numbers 3) remove punctuation 4) remove excess white-space 5) Remove stopwords("SMART","en")

Third, we created DTM(doc-term-matrix) for the train data set

```
## <<DocumentTermMatrix (documents: 2500, terms: 32241)>>
## Non-/sparse entries: 473695/80128805
## Sparsity           : 99%
## Maximal term length: 72
## Weighting          : term frequency (tf)
```

As we can see that after pre-processing, there are 2500 documents with 32241 terms(such a huge number), and the Sparsity is 99%

Fourth, we dropped those terms that only occur in one or two documents.This is a common step: the noise of the "long tail" (rare terms) can be huge, and there is nothing to learn if a term occurred once. Here, wer removed those terms that have count 0 in >95% of docs.

```
## <<DocumentTermMatrix (documents: 2500, terms: 660)>>
## Non-/sparse entries: 224397/1425603
## Sparsity           : 86%
## Maximal term length: 18
## Weighting          : term frequency (tf)
```

After doing this step, our terms significantly dropped to 660 and Sparsity becomes 86%

Last, we constructed the TF IDF weights for the DTM and created dense matrix for train data called Train.

After We got our ideal train data set, we repeated all these steps again for the C50 test documents and finally got our test data set as well.

```
## <<DocumentTermMatrix (documents: 2500, terms: 33048)>>
## Non-/sparse entries: 480577/82139423
## Sparsity           : 99%
## Maximal term length: 71
## Weighting          : term frequency (tf)
```

For test data, after pre-processing, there are 2500 documents with 33048 terms, and the Sparsity is 99%

```
## <<DocumentTermMatrix (documents: 2500, terms: 676)>>
## Non-/sparse entries: 228410/1461590
## Sparsity           : 86%
## Maximal term length: 18
## Weighting          : term frequency (tf)
```

After doing this step, our terms significantly dropped to 676 and Sparsity becomes 86%. We noticed that there are 16 new word occurred, compared to the train data.

To deal with the "new word issue" the way we choose is to ignore these 16 words, since it only about 2% of our data. We redo the doc term matrix again for our train data.

```
## <<DocumentTermMatrix (documents: 2500, terms: 660)>>
## Non-/sparse entries: 225031/1424969
## Sparsity           : 86%
## Maximal term length: 18
## Weighting          : term frequency (tf)
```

Now, both our train and test data sets have 660 terms.

Part 1 Dimensionality reduction using PCA analysis

First, let's check if there are columns that are zero on our test and train data set.

Train:

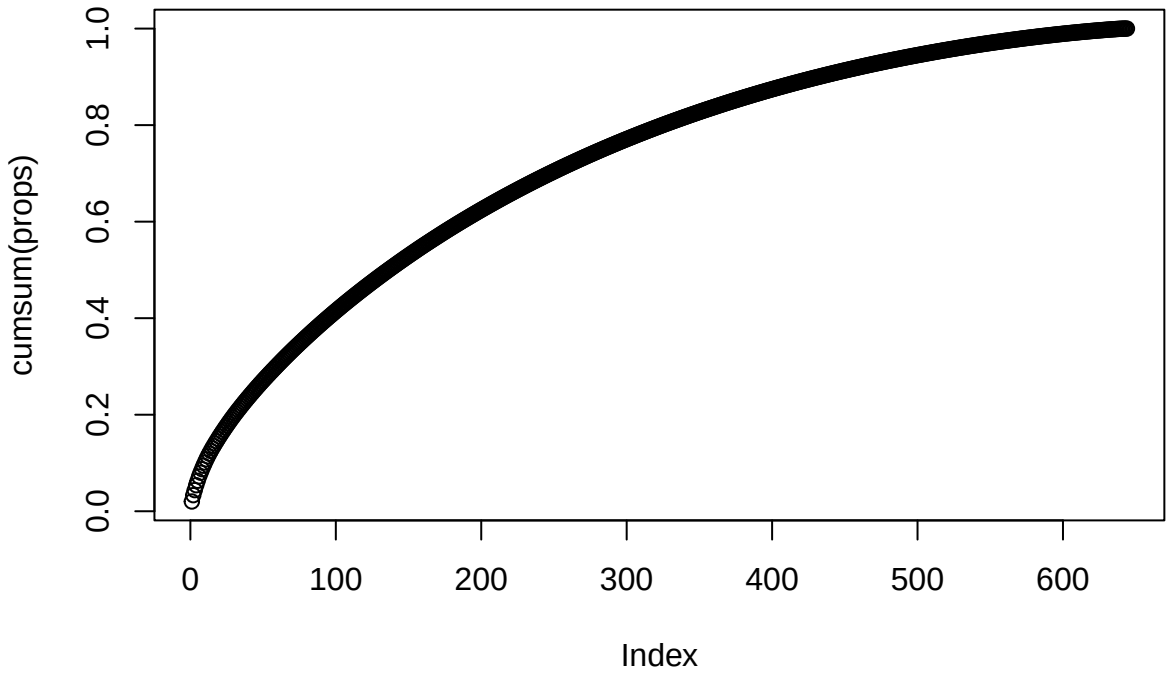
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.802   7.822   8.754  10.403  37.594
```

Test:

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.830   7.837   8.738  10.476  43.609
```

We need to remove those columns.

Then, we run the PCA analysis on the data to choose check our principle components.



For this case, we use 200 principle components that explain about 60% of the variance. We also need to reformat our data

Our our data is ready to run the model

Part 3 Modelling

Knn

First, we tried the Knn model with k = 10

```
## [1] 0.0324
```

The accuracy is very low, only 3.24%. It is a very bad model.

Random Forest

Then We Used Randaom Forest model for our data and the mtry = 14, the square root of 200.

```
## [1] 0.7336
```

The model shows a very high accuracy of 73.36%, much better than knn.

Naive Bays

Finally, let's come to the method that our professor used on class: Naive bays,

```
## [1] 0.04
```

The accuracy is also incredibly low, only 4%. It is a very also bad model.

Conclusion

Base on these three attempts of modelling, the random forest model performed best on our data with 73.36% accuracy.

Question 6: Association rule mining

Use the data on grocery purchases in groceries.txt and find some interesting association rules for these shopping baskets. Pick your own thresholds for lift and confidence; just be clear what these thresholds are and how you picked them. Do your discovered item sets make sense? Present your discoveries in an interesting and concise way.

Notes: Like with the first problem: this is an exercise in visual and numerical story-telling. Do be clear in your description of what you've done, but keep the focus on the data, the figures, and the insights your analysis has drawn from the data. The data file is a list of baskets: one row per basket, with multiple items per row separated by commas. You'll have to cobble together a few utilities for processing this into the format expected by the "arules" package. This is not intrinsically all that hard, but it is the kind of data-cleaning and pre-processing wrinkle you'll encounter frequently on real problems, where your software package expects data in one format and the data comes in a different format. Figuring out how to bridge that gap is part of the assignment, and so we won't be giving tips on this front.

In this question, we're discovering relationships among commonly purchased together grocery items.

```
##       V1                 V2                 V3                 V4
##  Length:15296       Length:15296       Length:15296       Length:15296
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

In this txt file, each row represents a basket, and there are four spaces in each baskets. However, the number of items in each baskets are different, which results in lots of missing values in this file. So we are going to do the data-cleaning first.

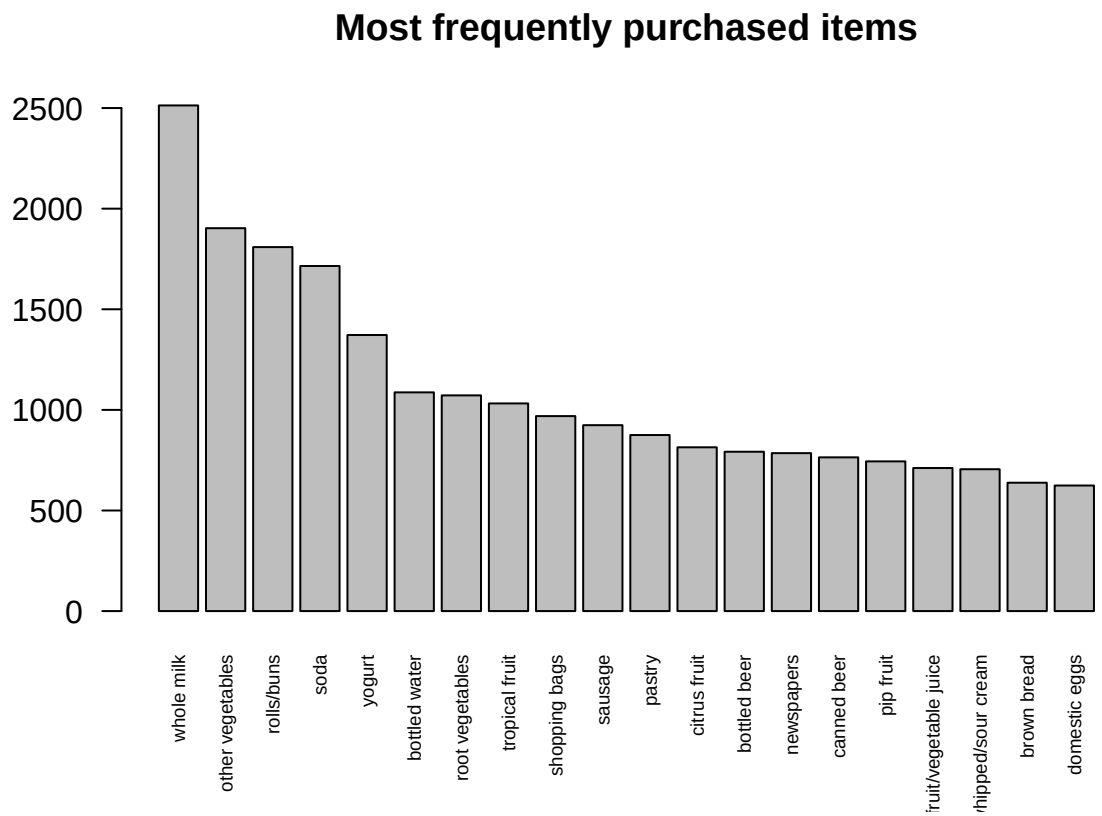
```
## 'data.frame':    43367 obs. of  2 variables:
##  $ customer: int  1 1 1 1 2 2 2 3 4 4 ...
##  $ value   : chr  "citrus fruit" "semi-finished bread" "margarine" "ready soups" ...
```

```
##     customer        value
##  Min.   :    1   Length:43367
##  1st Qu.: 3814   Class :character
##  Median : 7620   Mode  :character
##  Mean   : 7650
##  3rd Qu.:11482
##  Max.   :15296
```

In order to find out shopping patterns among customers, we need to conclude the most frequent items in the dataset.

```
##    Length     Class      Mode
##     43367 character character
```

```
##  [1] "citrus fruit"             "semi-finished bread"
##  [3] "margarine"                "ready soups"
##  [5] "tropical fruit"           "yogurt"
##  [7] "coffee"                   "whole milk"
##  [9] "pip fruit"                "yogurt"
## [11] "cream cheese "            "meat spreads"
## [13] "other vegetables"         "whole milk"
## [15] "condensed milk"           "long life bakery product"
## [17] "whole milk"               "butter"
## [19] "yogurt"                   "rice"
```



Based on the plot above, the most popular item in the dataset is whole milk, and then other vegetables, rolls/buns, soda, and yogurt are also top popular items.

In order to figure out the associations among these items, we are going to apply the apriori method.

```
## transactions as itemMatrix in sparse format with
##  15296 rows (elements/itemsets/transactions) and
##  169 columns (items) and a density of 0.01677625
##
## most frequent items:
##       whole milk other vegetables       rolls/buns             soda
##             2513             1903             1809             1715
##           yogurt          (Other)
##             1372            34055
##
## element (itemset/transaction) length distribution:
## sizes
##    1    2    3    4
## 3485 2630 2102 7079
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   2.835   4.000   4.000
##
## includes extended item information - examples:
##             labels
## 1 abrasive cleaner
## 2 artif. sweetener
```

```
## 3   baby cosmetics
##
## includes extended transaction information - examples:
##   transactionID
## 1             1
## 2             2
## 3             3

## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##       4  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 152
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 15296 transaction(s)] done [0.01s].
## sorting and recoding items ... [71 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [45 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].

##      lhs                    rhs                support    confidence
## [1]  {}                  => {soda}             0.11212082 0.1121208
## [2]  {}                  => {rolls/buns}       0.11826621 0.1182662
## [3]  {}                  => {other vegetables} 0.12441161 0.1244116
## [4]  {}                  => {whole milk}       0.16429132 0.1642913
## [5]  {curd}              => {whole milk}       0.01261768 0.3683206
## [6]  {butter}            => {whole milk}       0.01438285 0.4036697
## [7]  {whipped/sour cream} => {whole milk}      0.01144090 0.2482270
## [8]  {pip fruit}         => {tropical fruit}   0.01268305 0.2607527
## [9]  {tropical fruit}    => {pip fruit}        0.01268305 0.1879845
## [10] {pip fruit}         => {other vegetables} 0.01091789 0.2244624
## [11] {pip fruit}         => {whole milk}       0.01255230 0.2580645
## [12] {pastry}            => {rolls/buns}       0.01019874 0.1782857
## [13] {citrus fruit}      => {tropical fruit}   0.01248692 0.2346437
## [14] {tropical fruit}    => {citrus fruit}     0.01248692 0.1850775
## [15] {citrus fruit}      => {other vegetables} 0.01281381 0.2407862
## [16] {other vegetables}  => {citrus fruit}     0.01281381 0.1029953
## [17] {citrus fruit}      => {whole milk}       0.01281381 0.2407862
## [18] {sausage}           => {rolls/buns}       0.01078713 0.1785714
## [19] {sausage}           => {other vegetables} 0.01261768 0.2088745
## [20] {other vegetables}  => {sausage}          0.01261768 0.1014188
## [21] {sausage}           => {whole milk}       0.01255230 0.2077922
## [22] {bottled water}     => {soda}             0.01464435 0.2060718
## [23] {soda}              => {bottled water}    0.01464435 0.1306122
```

```
## [24] {tropical fruit}     => {root vegetables}  0.01098326 0.1627907
## [25] {root vegetables}    => {tropical fruit}   0.01098326 0.1567164
## [26] {tropical fruit}     => {other vegetables} 0.01549425 0.2296512
## [27] {other vegetables}   => {tropical fruit}   0.01549425 0.1245402
## [28] {tropical fruit}     => {whole milk}       0.01830544 0.2713178
## [29] {whole milk}         => {tropical fruit}   0.01830544 0.1114206
## [30] {root vegetables}    => {other vegetables} 0.02536611 0.3619403
## [31] {other vegetables}   => {root vegetables}  0.02536611 0.2038886
## [32] {root vegetables}    => {whole milk}       0.02262029 0.3227612
## [33] {whole milk}         => {root vegetables}  0.02262029 0.1376840
## [34] {yogurt}             => {rolls/buns}       0.01189854 0.1326531
## [35] {rolls/buns}         => {yogurt}           0.01189854 0.1006081
## [36] {yogurt}             => {other vegetables} 0.01588651 0.1771137
## [37] {other vegetables}   => {yogurt}           0.01588651 0.1276931
## [38] {yogurt}             => {whole milk}       0.02425471 0.2704082
## [39] {whole milk}         => {yogurt}           0.02425471 0.1476323
## [40] {soda}               => {rolls/buns}       0.01425209 0.1271137
## [41] {rolls/buns}         => {soda}             0.01425209 0.1205086
## [42] {rolls/buns}         => {whole milk}       0.01830544 0.1547816
## [43] {whole milk}         => {rolls/buns}       0.01830544 0.1114206
## [44] {other vegetables}   => {whole milk}       0.04086036 0.3284288
## [45] {whole milk}         => {other vegetables} 0.04086036 0.2487067
##      coverage   lift     count
## [1]  1.00000000 1.000000 1715
## [2]  1.00000000 1.000000 1809
## [3]  1.00000000 1.000000 1903
## [4]  1.00000000 1.000000 2513
## [5]  0.03425732 2.241875  193
## [6]  0.03563023 2.457036  220
## [7]  0.04609048 1.510895  175
## [8]  0.04864017 3.864800  194
## [9]  0.06746862 3.864800  194
## [10] 0.04864017 1.804191  167
## [11] 0.04864017 1.570774  192
## [12] 0.05720450 1.507495  156
## [13] 0.05321653 3.477820  191
## [14] 0.06746862 3.477820  191
## [15] 0.05321653 1.935400  196
## [16] 0.12441161 1.935400  196
## [17] 0.05321653 1.465605  196
## [18] 0.06040795 1.509911  165
## [19] 0.06040795 1.678898  193
## [20] 0.12441161 1.678898  193
## [21] 0.06040795 1.264779  192
## [22] 0.07106433 1.837944  224
## [23] 0.11212082 1.837944  224
## [24] 0.06746862 2.322805  168
## [25] 0.07008368 2.322805  168
## [26] 0.06746862 1.845898  237
## [27] 0.12441161 1.845898  237
## [28] 0.06746862 1.651444  280
## [29] 0.16429132 1.651444  280
## [30] 0.07008368 2.909216  388
## [31] 0.12441161 2.909216  388
```

```
## [32] 0.07008368 1.964566  346
## [33] 0.16429132 1.964566  346
## [34] 0.08969665 1.121648  182
## [35] 0.11826621 1.121648  182
## [36] 0.08969665 1.423611  243
## [37] 0.12441161 1.423611  243
## [38] 0.08969665 1.645907  371
## [39] 0.16429132 1.645907  371
## [40] 0.11212082 1.074810  218
## [41] 0.11826621 1.074810  218
## [42] 0.11826621 0.942117  280
## [43] 0.16429132 0.942117  280
## [44] 0.12441161 1.999064  625
## [45] 0.16429132 1.999064  625
```

According to the results above, there are 45 combinations which match the criteria. In this part. we used support as 0.01, confidence as 0.1 and maxlen as 4 to identify the rules of shopping. And we may conclude more specific rules by adjusting the value of support and confidence. (We don't change the maxlen since there should be maximum 4 items in each basket.)

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5   0.015      1
##  maxlen target  ext
##       4  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 229
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 15296 transaction(s)] done [0.01s].
## sorting and recoding items ... [56 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [20 rule(s)] done [0.00s].
## creating S4 object  ... done [0.01s].
```

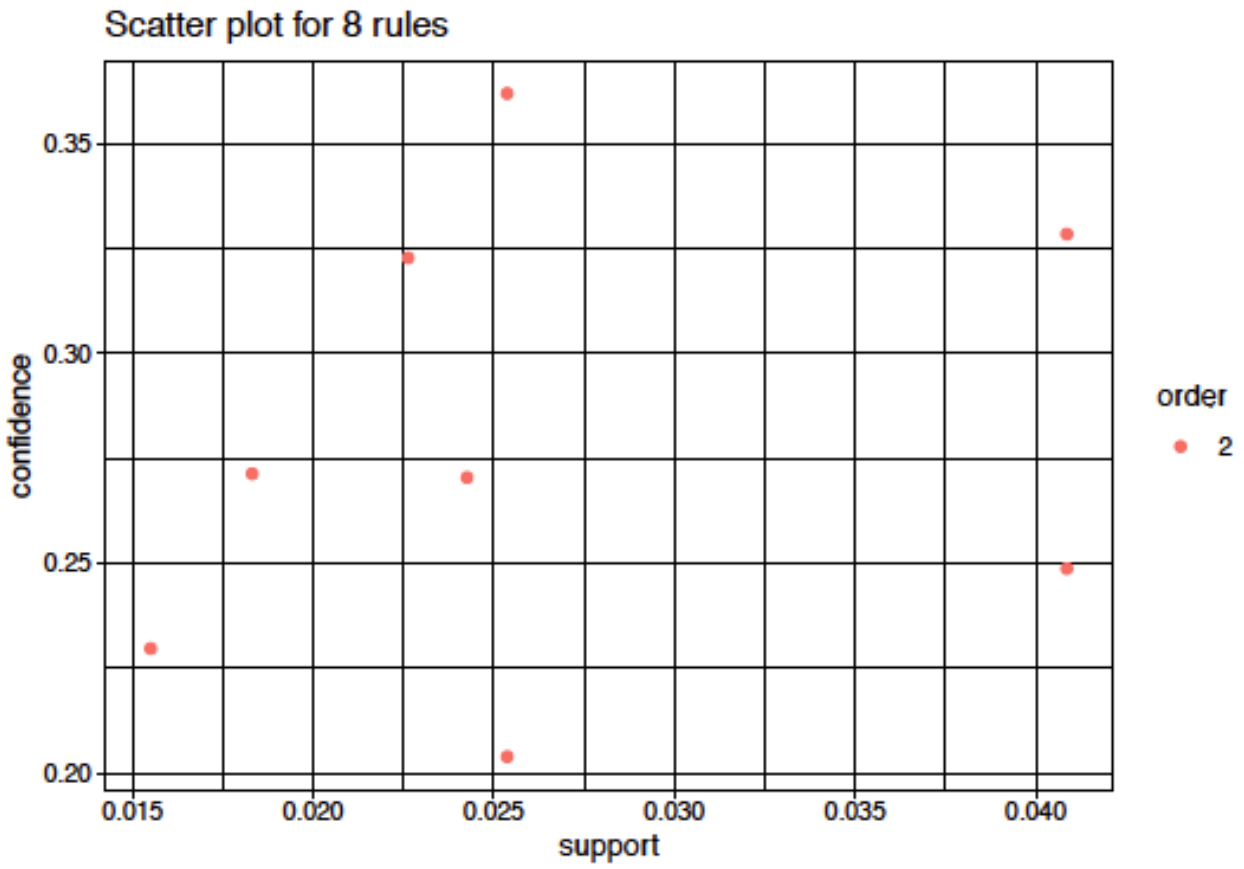
```
##      lhs                   rhs                support    confidence coverage
## [1]  {}                 => {soda}             0.11212082 0.1121208  1.00000000
## [2]  {}                 => {rolls/buns}       0.11826621 0.1182662  1.00000000
## [3]  {}                 => {other vegetables} 0.12441161 0.1244116  1.00000000
## [4]  {}                 => {whole milk}       0.16429132 0.1642913  1.00000000
## [5]  {tropical fruit}   => {other vegetables} 0.01549425 0.2296512  0.06746862
## [6]  {other vegetables} => {tropical fruit}   0.01549425 0.1245402  0.12441161
## [7]  {tropical fruit}   => {whole milk}       0.01830544 0.2713178  0.06746862
## [8]  {whole milk}       => {tropical fruit}   0.01830544 0.1114206  0.16429132
## [9]  {root vegetables}  => {other vegetables} 0.02536611 0.3619403  0.07008368
## [10] {other vegetables} => {root vegetables}  0.02536611 0.2038886  0.12441161
```

```
## [11] {root vegetables}  => {whole milk}       0.02262029 0.3227612  0.07008368
## [12] {whole milk}       => {root vegetables}  0.02262029 0.1376840  0.16429132
## [13] {yogurt}           => {other vegetables} 0.01588651 0.1771137  0.08969665
## [14] {other vegetables} => {yogurt}           0.01588651 0.1276931  0.12441161
## [15] {yogurt}           => {whole milk}       0.02425471 0.2704082  0.08969665
## [16] {whole milk}       => {yogurt}           0.02425471 0.1476323  0.16429132
## [17] {rolls/buns}       => {whole milk}       0.01830544 0.1547816  0.11826621
## [18] {whole milk}       => {rolls/buns}       0.01830544 0.1114206  0.16429132
## [19] {other vegetables} => {whole milk}       0.04086036 0.3284288  0.12441161
## [20] {whole milk}       => {other vegetables} 0.04086036 0.2487067  0.16429132
##      lift     count
## [1]  1.000000 1715
## [2]  1.000000 1809
## [3]  1.000000 1903
## [4]  1.000000 2513
## [5]  1.845898  237
## [6]  1.845898  237
## [7]  1.651444  280
## [8]  1.651444  280
## [9]  2.909216  388
## [10] 2.909216  388
## [11] 1.964566  346
## [12] 1.964566  346
## [13] 1.423611  243
## [14] 1.423611  243
## [15] 1.645907  371
## [16] 1.645907  371
## [17] 0.942117  280
## [18] 0.942117  280
## [19] 1.999064  625
## [20] 1.999064  625

## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5   0.015      1
##  maxlen target  ext
##       4  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 229
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[169 item(s), 15296 transaction(s)] done [0.01s].
## sorting and recoding items ... [56 item(s)] done [0.00s].
## creating transaction tree ... done [0.01s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [8 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
##     lhs                   rhs                support    confidence coverage
## [1] {tropical fruit}   => {other vegetables} 0.01549425 0.2296512  0.06746862
## [2] {tropical fruit}   => {whole milk}       0.01830544 0.2713178  0.06746862
## [3] {root vegetables}  => {other vegetables} 0.02536611 0.3619403  0.07008368
## [4] {other vegetables} => {root vegetables}  0.02536611 0.2038886  0.12441161
## [5] {root vegetables}  => {whole milk}       0.02262029 0.3227612  0.07008368
## [6] {yogurt}           => {whole milk}       0.02425471 0.2704082  0.08969665
## [7] {other vegetables} => {whole milk}       0.04086036 0.3284288  0.12441161
## [8] {whole milk}       => {other vegetables} 0.04086036 0.2487067  0.16429132
##     lift     count
## [1] 1.845898 237
## [2] 1.651444 280
## [3] 2.909216 388
## [4] 2.909216 388
## [5] 1.964566 346
## [6] 1.645907 371
## [7] 1.999064 625
## [8] 1.999064 625
```

When the support increased to 0.015, there are 20 types of combination. And then we get 8 different types of combination when we increased the confidence from 0.1 to 0.2. Based on the results above, we noticed that some combinations occurred frequently, such as vegetables + whole milk, other vegetables + tropical fruit, and tropical fruit and whole milk. It indicates that when a customer buy tropical, it highly possible that he/she would purchase the whole milk or other vegetables as well. So these three items are made of the



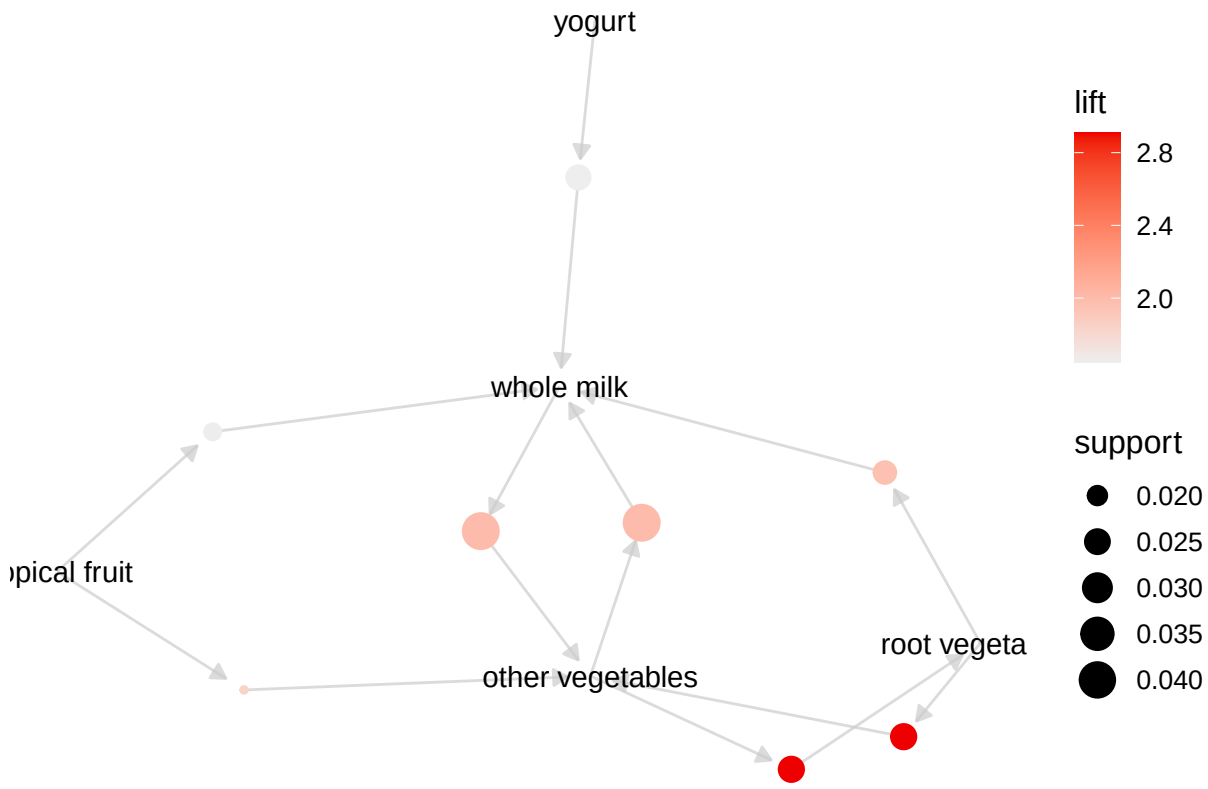
Let's see more details about the rules we concluded above

```
##     lhs                   rhs                support    confidence coverage
## [1] {tropical fruit}   => {other vegetables} 0.01549425 0.2296512  0.06746862
## [2] {tropical fruit}   => {whole milk}       0.01830544 0.2713178  0.06746862
## [3] {root vegetables}  => {other vegetables} 0.02536611 0.3619403  0.07008368
## [4] {other vegetables} => {root vegetables}  0.02536611 0.2038886  0.12441161
## [5] {root vegetables}  => {whole milk}       0.02262029 0.3227612  0.07008368
## [6] {yogurt}           => {whole milk}       0.02425471 0.2704082  0.08969665
## [7] {other vegetables} => {whole milk}       0.04086036 0.3284288  0.12441161
## [8] {whole milk}       => {other vegetables} 0.04086036 0.2487067  0.16429132
##     lift     count
## [1] 1.845898 237
## [2] 1.651444 280
## [3] 2.909216 388
## [4] 2.909216 388
## [5] 1.964566 346
## [6] 1.645907 371
## [7] 1.999064 625
## [8] 1.999064 625

##     lhs                   rhs                support    confidence coverage
## [1] {root vegetables}  => {other vegetables} 0.02536611 0.3619403  0.07008368
## [2] {root vegetables}  => {whole milk}       0.02262029 0.3227612  0.07008368
## [3] {other vegetables} => {whole milk}       0.04086036 0.3284288  0.12441161
##     lift     count
## [1] 2.909216 388
## [2] 1.964566 346
## [3] 1.999064 625

## set of 8 rules
##
## rule length distribution (lhs + rhs):sizes
## 2
## 8
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       2       2       2       2       2       2
##
## summary of quality measures:
##     support          confidence        coverage            lift
##  Min.   :0.01549   Min.   :0.2039   Min.   :0.06747   Min.   :1.646
##  1st Qu.:0.02154   1st Qu.:0.2439   1st Qu.:0.06943   1st Qu.:1.797
##  Median :0.02481   Median :0.2709   Median :0.07989   Median :1.982
##  Mean   :0.02664   Mean   :0.2796   Mean   :0.09724   Mean   :2.116
##  3rd Qu.:0.02924   3rd Qu.:0.3242   3rd Qu.:0.12441   3rd Qu.:2.227
##  Max.   :0.04086   Max.   :0.3619   Max.   :0.16429   Max.   :2.909
##      count
##  Min.   :237.0
##  1st Qu.:329.5
##  Median :379.5
##  Mean   :407.5
##  3rd Qu.:447.2
##  Max.   :625.0
##
## mining info:
```

```
##         data ntransactions support confidence
##  grocs_trans         15296   0.015        0.2
```



In conclusion, whole milk, vegetables, and tropical fruits are highly associated items, while they are most popular items in the dataset. Upon close placement of pairs that see high confidence and support, we can expect to see positive results in sales as they give ease of access to shoppers while filling their cart. For items that see high confidence in X,Y pairs, we must try to place Y close to X if we are not doing so for a similar reason. Complementary items of high association pairs can be placed at checkout counter as well to drive impulse purchase behavior. Therefore, we could learn more about purchase patterns by checking association rules for shopping basket, and then improve our sales and services based on customers' shopping behaviors.