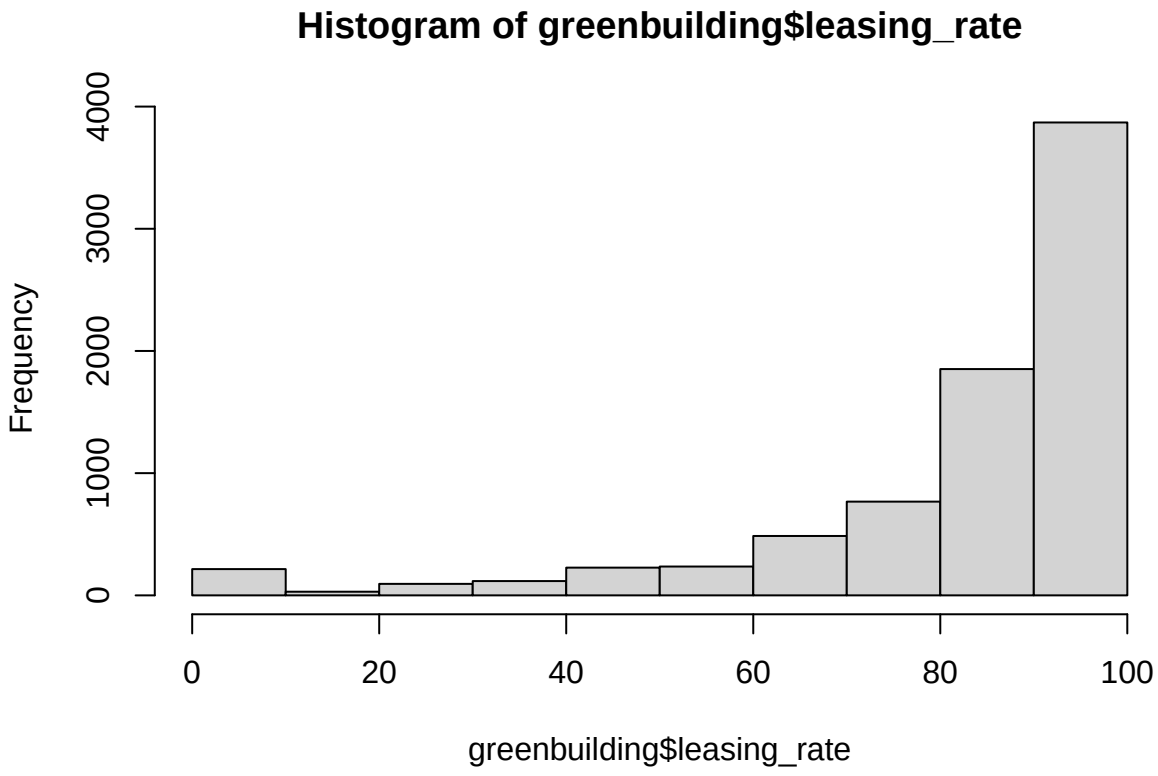
Predictive Modeling Part 2 Project

David Cruz

8/17/2020

Visual story telling part 1: green buildings

```
library(mosaic)
library(tidyverse)
library(ggplot2)
library(readr)
greenbuilding <- read.csv('/Users/davidcruz/Documents/Predictive Modeling2 Project/Green Buildings/green

# remove low occupancy buildings
hist(greenbuilding$leasing_rate)
```




```
favstats(~leasing_rate, data=greenbuilding)
```

```
##  min    Q1 median    Q3 max     mean       sd    n missing
##    0 77.85  89.53 96.44 100 82.60637 21.38031 7894       0
```

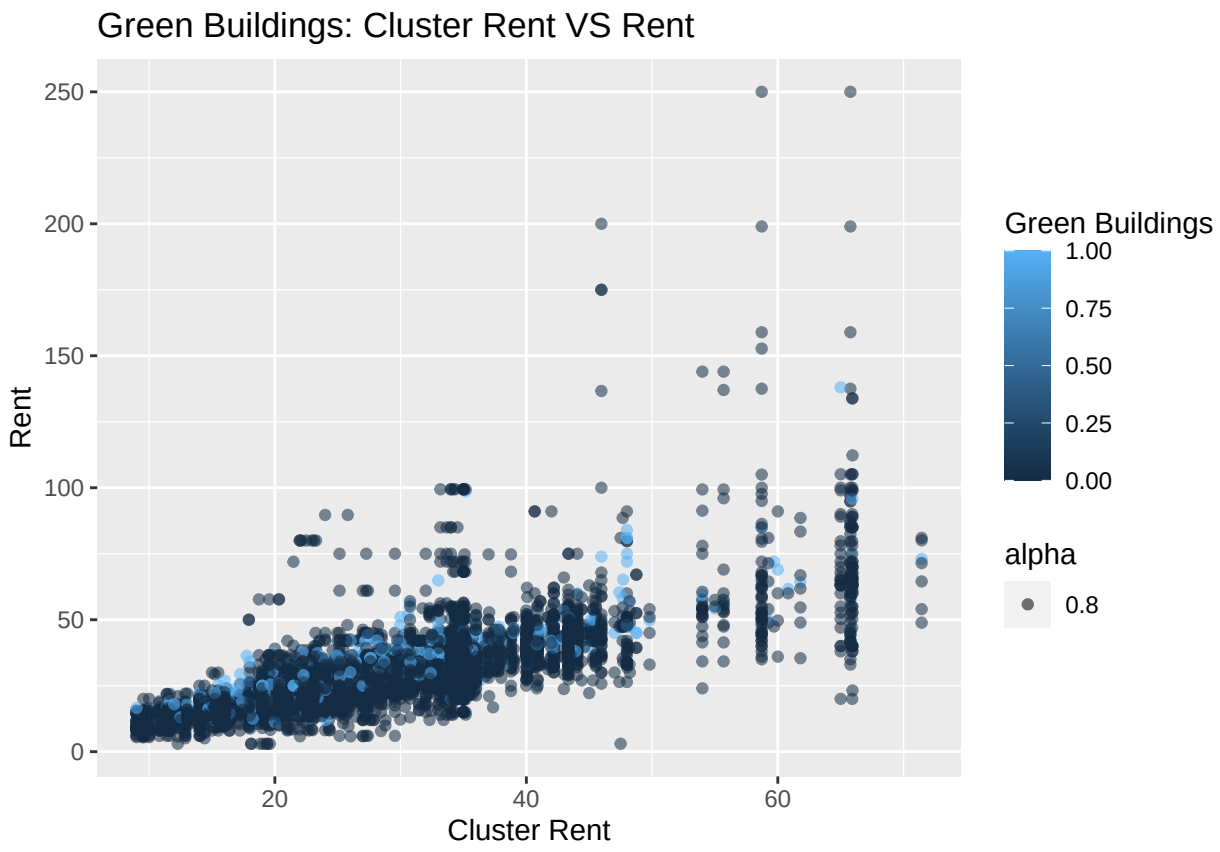
```
greenbuilding <- greenbuilding %>%
  filter(leasing_rate > 10)
```

I wanted to check out the staff's analysis first before discussing my own analysis and what I think he missed. As stated in his recommendation, there is a notable group of building that had very low occupancy rate(less than 10%). I think he was right in removing these buildings from this analysis with the given reason, so I will also remove them.*

```
# separate green and non-green buildings
tally(~green_rating, data=greenbuilding)
```

```
## green_rating
##    0    1
## 6995  684
```

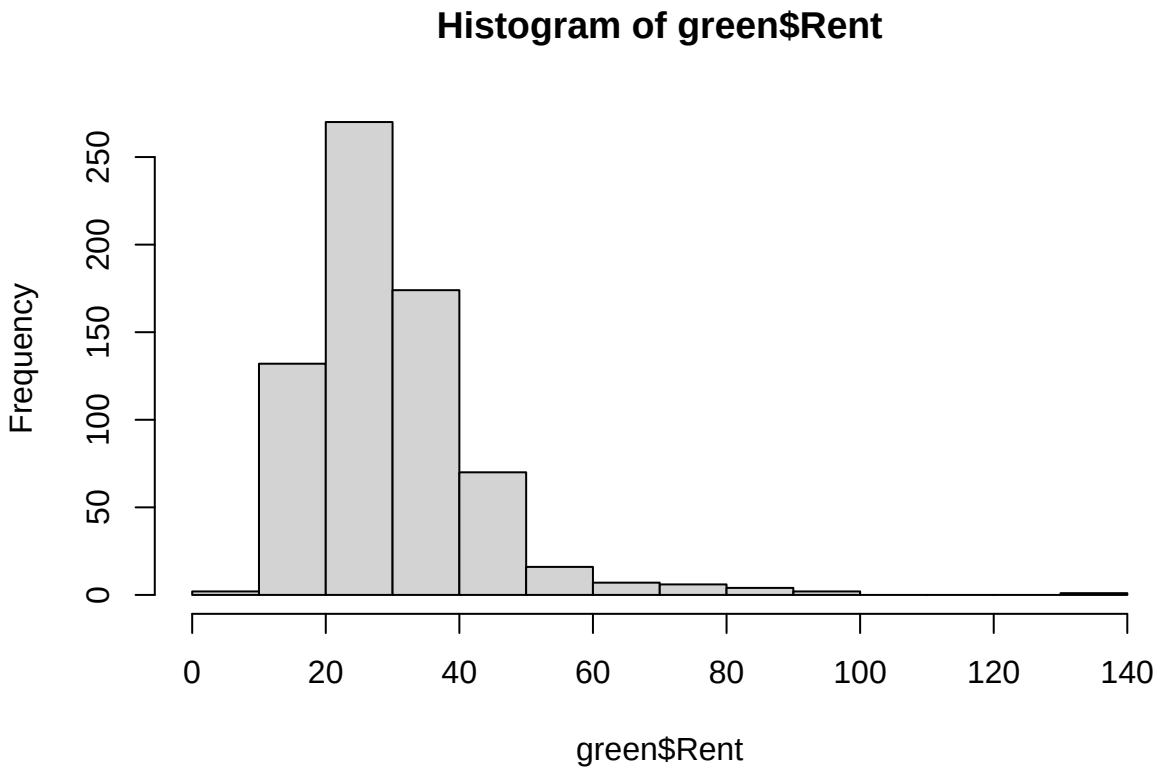
```
ggplot(data=greenbuilding) +
  geom_point(mapping=aes(x=cluster_rent, y=Rent, colour=green_rating, alpha=0.8)) +
  labs(x="Cluster Rent", y='Rent', title = 'Green Buildings: Cluster Rent VS Rent',
       color='Green Buildings')
```




```
green <- greenbuilding %>%
  filter(green_rating == '1')
non_green <- greenbuilding %>%
  filter(green_rating == '0')

favstats(~Rent, data=green)
```

```
##   min      Q1 median    Q3    max     mean       sd   n missing
##  8.87 21.4975   27.6 35.54 138.07 30.02848 12.95545 684       0
```

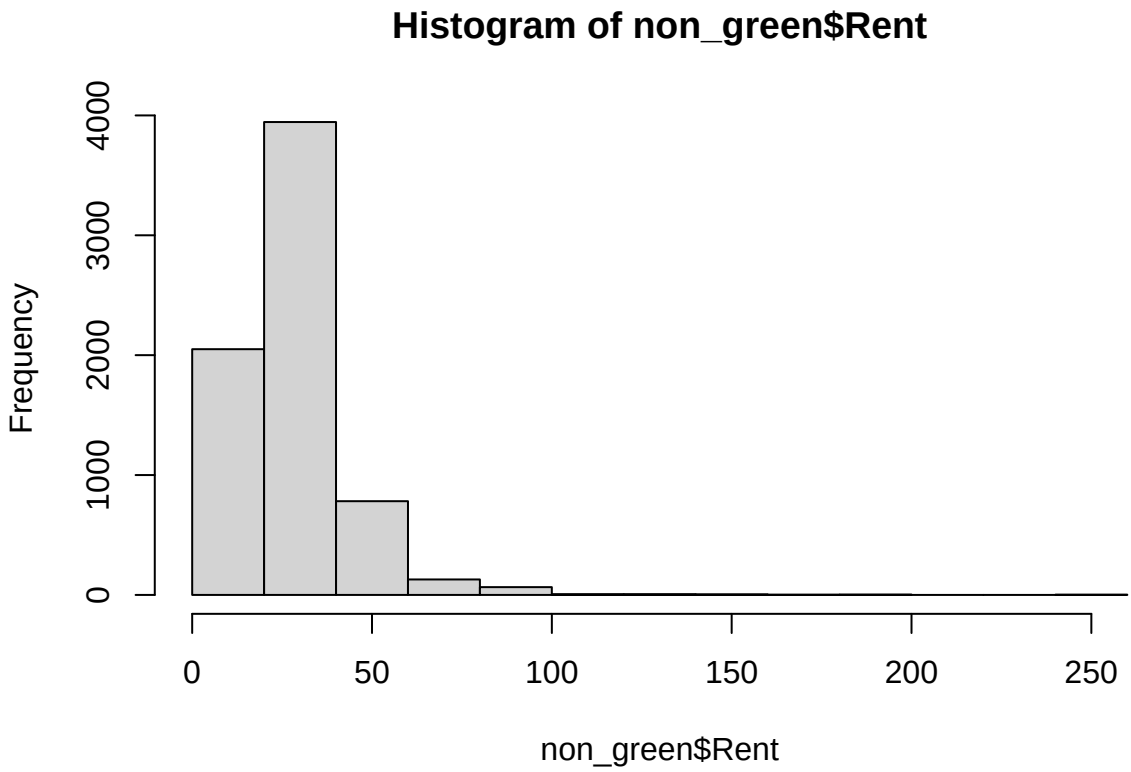
```
hist(green$Rent)
```



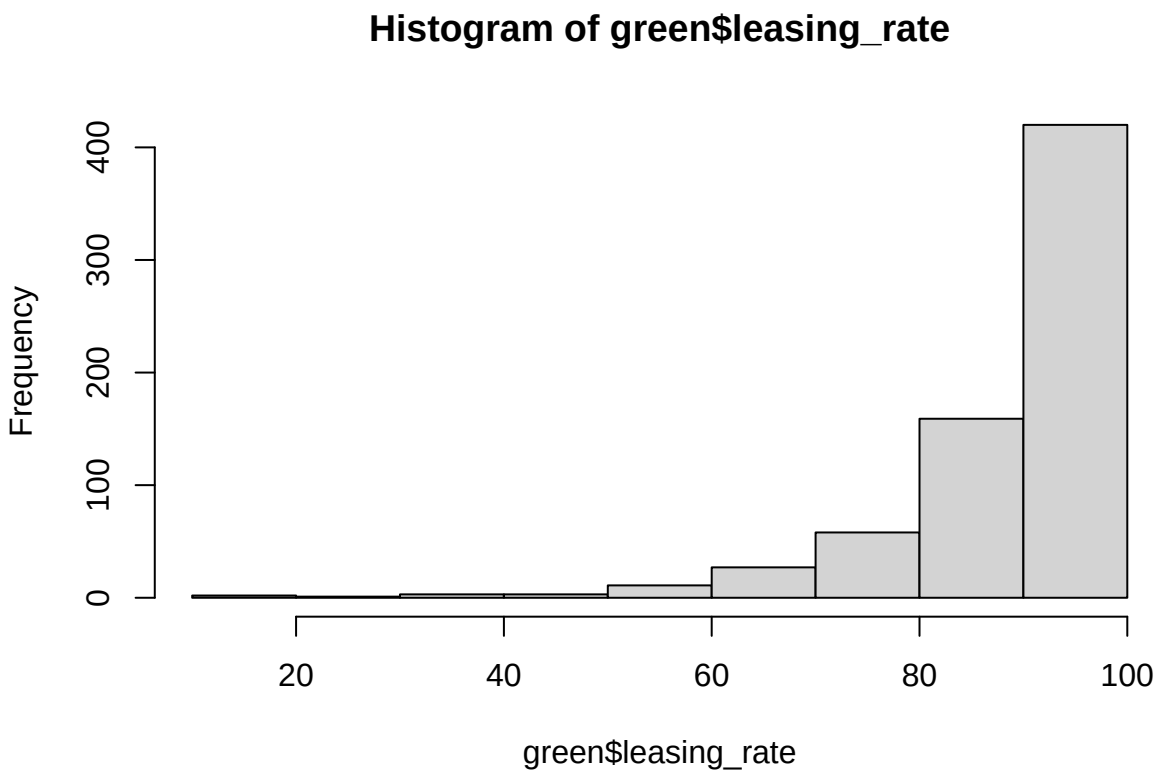

```
favstats(~Rent, data=non_green)
```

```
##   min    Q1 median    Q3 max     mean       sd    n missing
##  2.98 19.43  25.03 34.18 250 28.44478 15.32829 6995       0
```

```
hist(non_green$Rent)
```




```
hist(green$leasing_rate)
```




```
favstats(~leasing_rate, data=green)
```

```
##     min      Q1 median      Q3 max     mean       sd    n missing
```

```
##  12.39 85.4525 92.925 97.7025 100 89.41243 11.82425 684       0
```

The distributions of rent are skewed for both green and non_green buildings, so he was right in considering the median rent value. The median rent for green buildings is $27.6, and the median rent for non_green buildings is $25.03. Green buildings are, on average, $2.57 more per square foot. Generally speaking, since our building would be 250,000 square feet, we could earn about $642,500 more with a green building than with a non-green building. Considering the median leasing rate for green buildings, 92.92%, we could recuperate the baseline construction and green certification costs ($5m) in about 8.37 years through this revenue. However, this conclusion does not take into account other variables that could impact our profit (other factors that impacts rent).*

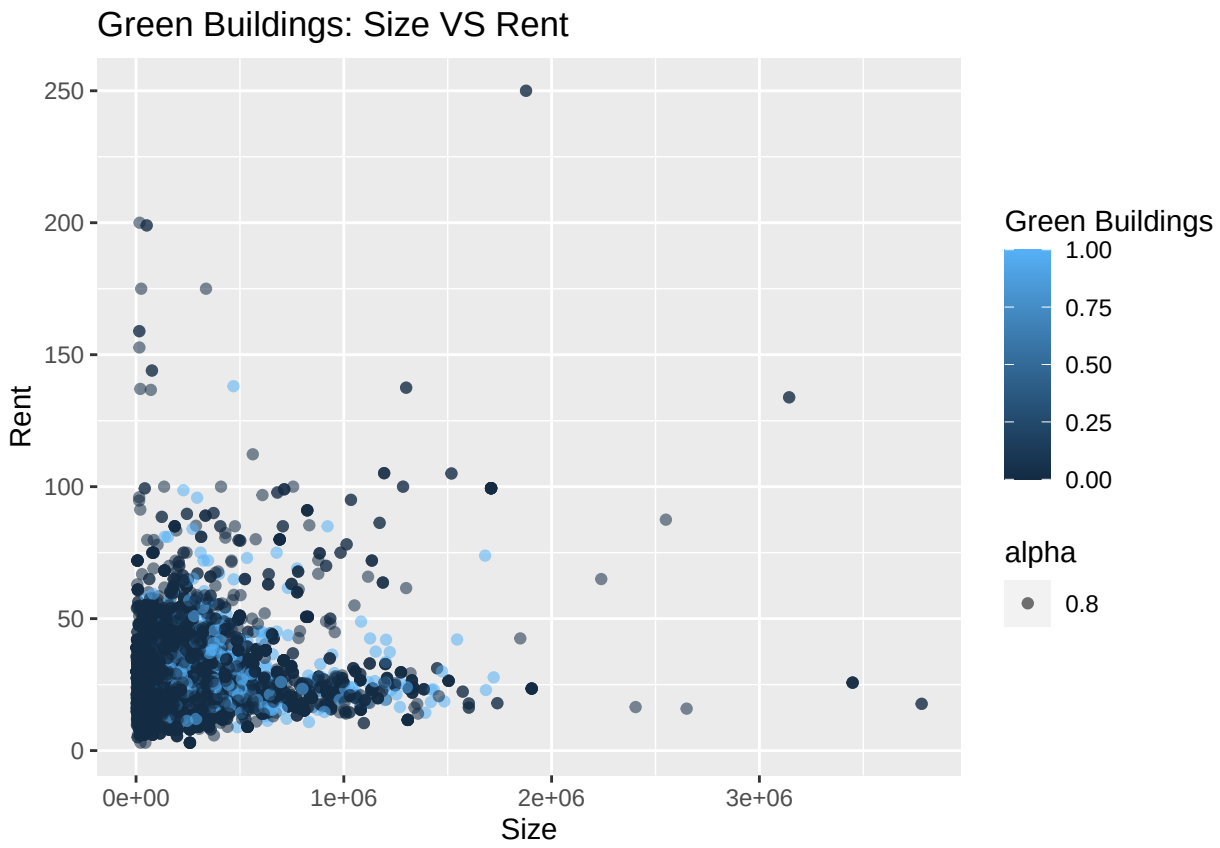
```
favstats(~size, data=green)
```

```
##    min     Q1 median       Q3     max     mean       sd   n missing
##  10560 120000 241199 417449.2 1721242 325965.2 289945.2 684       0
```

```
favstats(~size, data=non_green)
```

```
##   min    Q1 median     Q3     max     mean       sd    n missing
##  2378 48873 123250 285000 3781045 231007.2 299636.6 6995       0
```

```
ggplot(data=greenbuilding) +
  geom_point(mapping=aes(x=size, y=Rent, colour=green_rating, alpha=0.8)) +
  labs(x="Size", y='Rent', title = 'Green Buildings: Size VS Rent',
       color='Green Buildings')
```




```
favstats(~age, data=green)
```

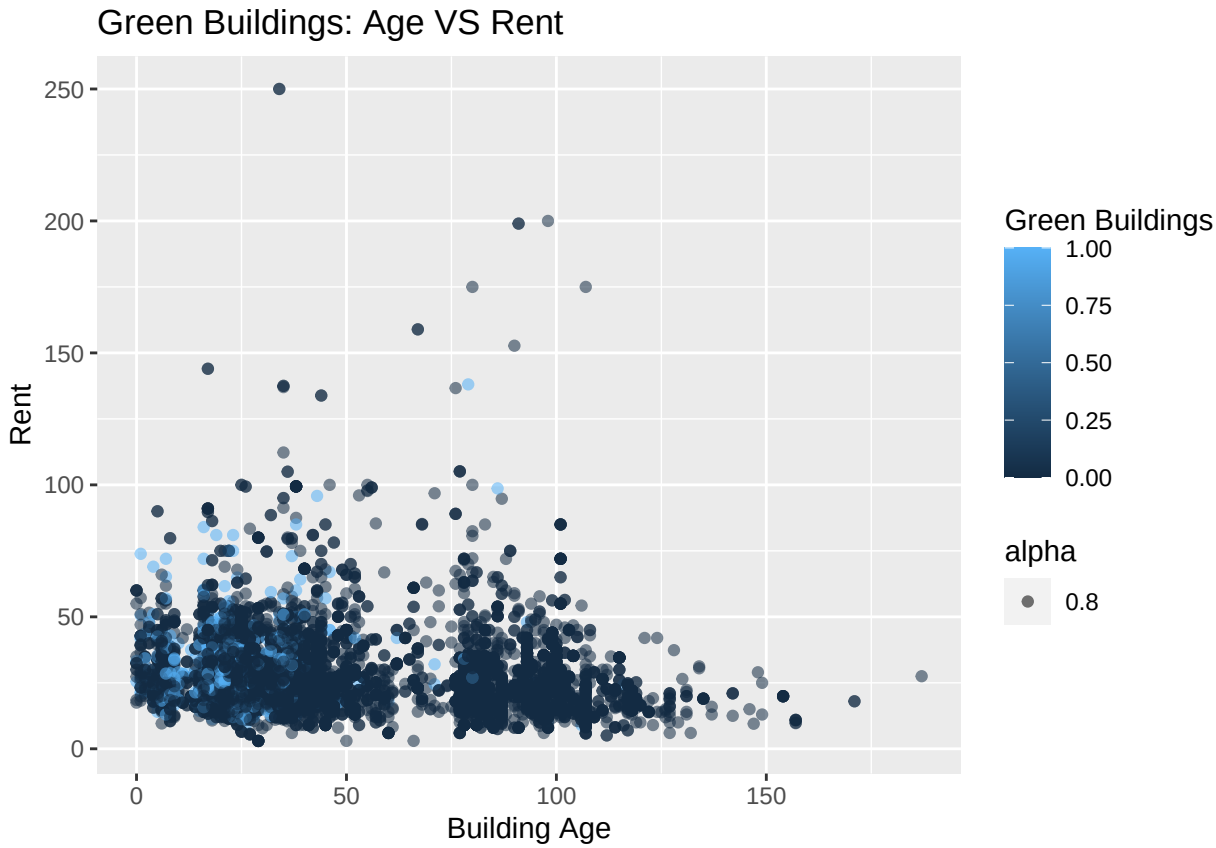
```
##  min Q1 median Q3 max     mean       sd   n missing
```

```
##    0 18     22 26 116 23.88012 15.55513 684       0
```

```
favstats(~age, data=non_green)
```

```
##  min Q1 median Q3 max     mean       sd    n missing
##    0 24     36 80 187 49.30808 32.46818 6995       0
```

```
ggplot(data=greenbuilding) +
  geom_point(mapping=aes(x=age, y=Rent, colour=green_rating, alpha=0.8)) +
  labs(x="Building Age", y='Rent', title = 'Green Buildings: Age VS Rent',
       color='Green Buildings')
```



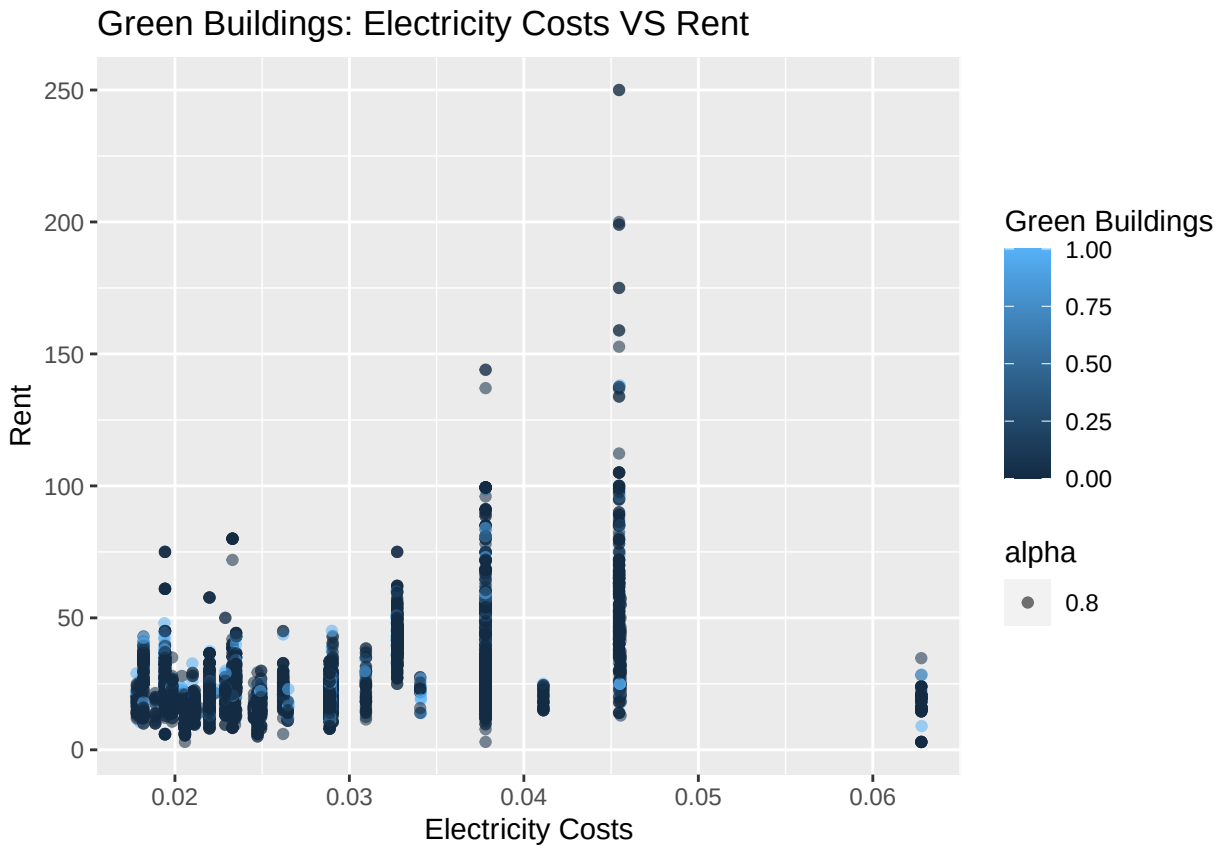

```
favstats(~Electricity_Costs, data=green)
```

```
##     min     Q1 median     Q3    max       mean          sd   n missing
##  0.0178 0.0235 0.0341 0.0378 0.0628 0.03158567 0.007819872 684       0
```

```
favstats(~Electricity_Costs, data=non_green)
```

```
##         min         Q1    median         Q3        max       mean          sd
##  0.01781946 0.02330012 0.0327374 0.03780774 0.06277843 0.03089267 0.008592756
##     n missing
##  6995       0
```

```
ggplot(data=greenbuilding) +
  geom_point(mapping=aes(x=Electricity_Costs, y=Rent, colour=green_rating, alpha=0.8)) +
  labs(x="Electricity Costs", y='Rent', title = 'Green Buildings: Electricity Costs VS Rent',
       color='Green Buildings')
```



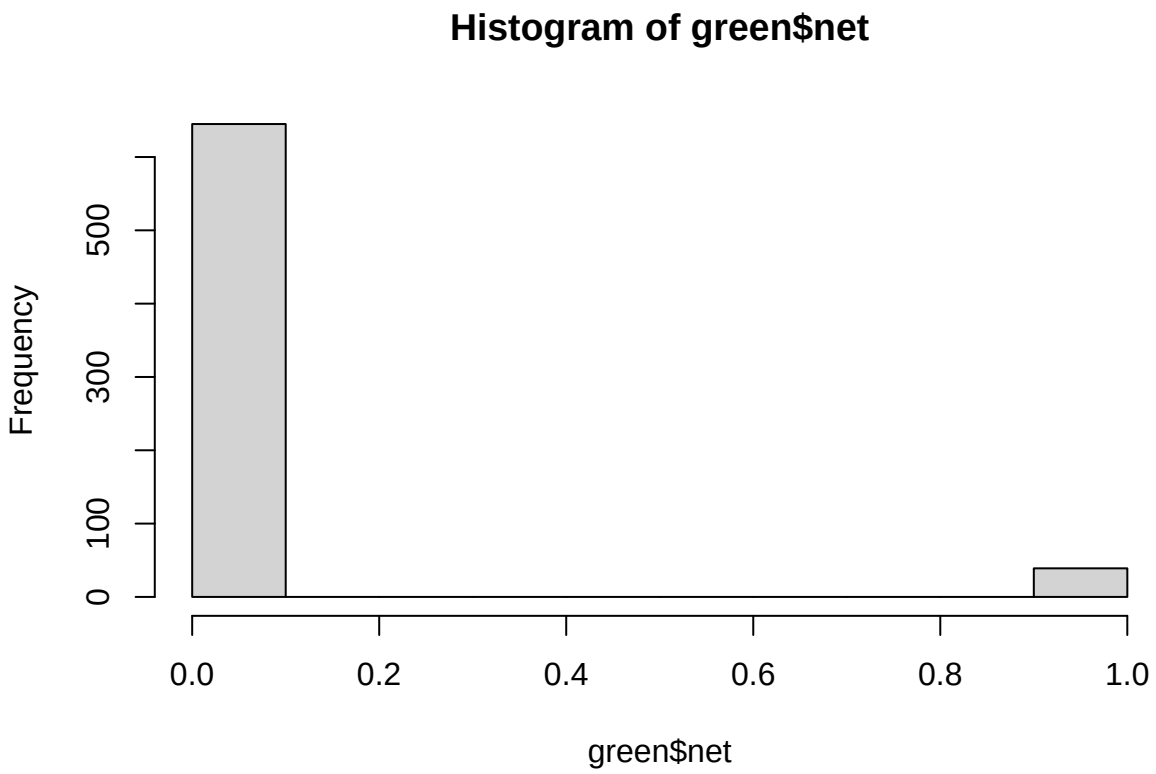

```
tally(~net, data=green)
```

```
## net
##   0   1
## 645  39
```

```
tally(~net, data=non_green)
```

```
## net
##    0    1
## 6761  234
```

```
hist(green$net)
```



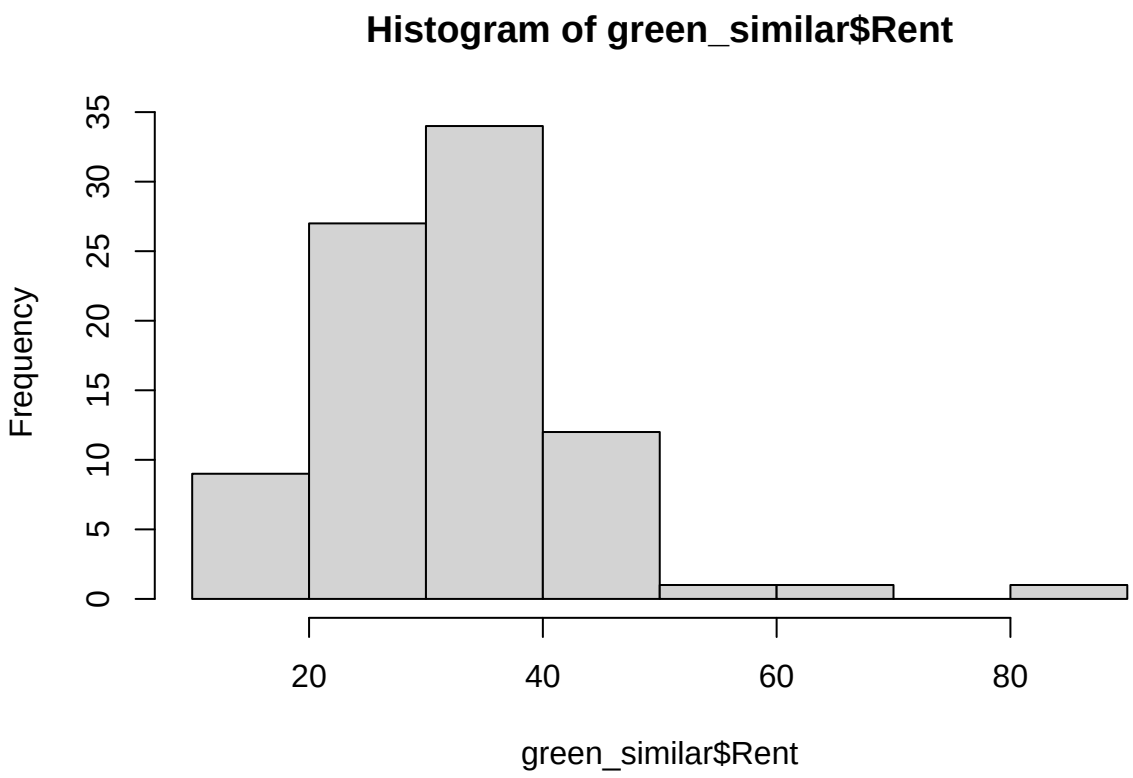

```
green_similar <- green %>%
  filter(size <= 300000 & size >= 200000) %>%
  filter(age<=24) %>%
  filter(net == 0)
nongreen_similar <- non_green %>%
  filter(size <= 300000 & size >= 200000) %>%
  filter(age<=24) %>%
  filter(net == 0)

favstats(~Rent, data=green_similar)
```

```
##    min   Q1 median   Q3 max     mean       sd  n missing
##  13.81 25.2  33.36 38.4  84 32.87612 11.01452 85       0
```

```
favstats(~Rent, data=nongreen_similar)
```

```
##  min Q1 median    Q3  max    mean       sd   n missing
##  9.1 24     35 42.89 89.7 34.6883 13.15785 300       0
```

```
hist(green_similar$Rent)
```




```
hist(nongreen_similar$Rent)
```


```
green_similar_nocover <- green %>%
  filter(size <= 300000 & size >= 200000) %>%
  filter(age<=24) %>%
```

```
  filter(net == 1)
nongreen_similar_nocover <- non_green %>%
  filter(size <= 300000 & size >= 200000) %>%
  filter(age<=24) %>%
  filter(net == 1)
favstats(~Rent, data=green_similar_nocover)
```

```
##   min      Q1 median      Q3   max     mean       sd n missing
##  17.5 19.5675  20.74 22.0925 27.63 21.38833 3.484867 6       0
```

```
favstats(~Rent, data=nongreen_similar_nocover )
```

```
##    min    Q1 median   Q3   max     mean       sd n missing
##  15.42 19.43  23.21 33.5 33.54 24.63556 7.043623 9       0
```

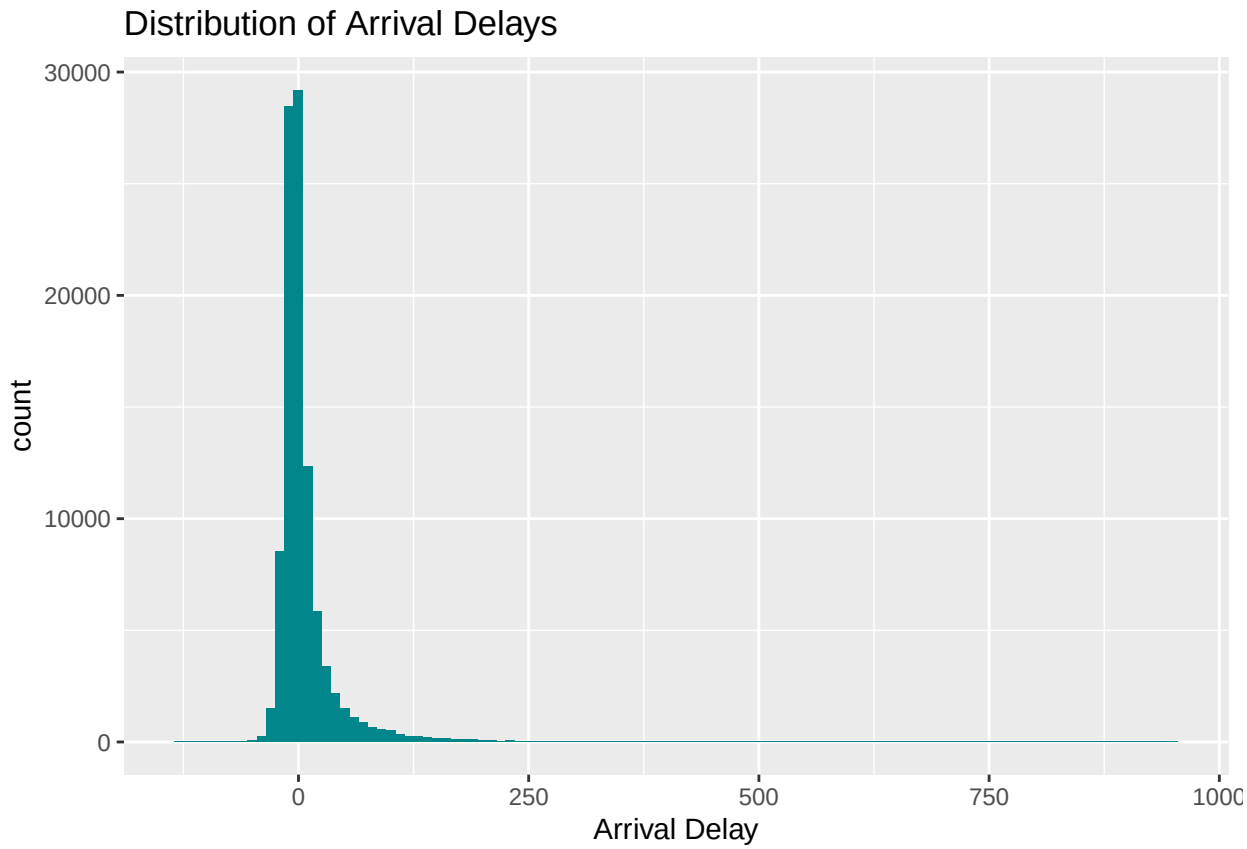
It seems that the proportion of green to non-green buildings changes with size, there is more smaller building that are non-green and there are more green buildings as size increases. The changes with building age as well. Green building are generally newer than non-green buildings. I had expected green buildings to have less electricity costs than non-green building, because one of the attractions of them is to cut recurring costs. On average green buildings' elevtrivity costs is higher than non-green buildings. This is a troublesome discovery, since a lot of buildings (both green and non-green) have utilities included in rent price. So let's take a look at green and non-green buildings with similar conditions as us. I'm only considering the buildings between 200,000 and 300,000 square feet, under the age 24 (Q1), and have utilities covered by rent. Now that we are only considering buildings with similar coniditions as ours, we see that green building's rent is cheaper than non-green buildings on average. Even if we do not cover utilities, the rent for green buildings are still less than non-gree buildings. This suggests we might earn less than non-green buildings with this project. Thus, investing in a green building woudl not be worth it.*

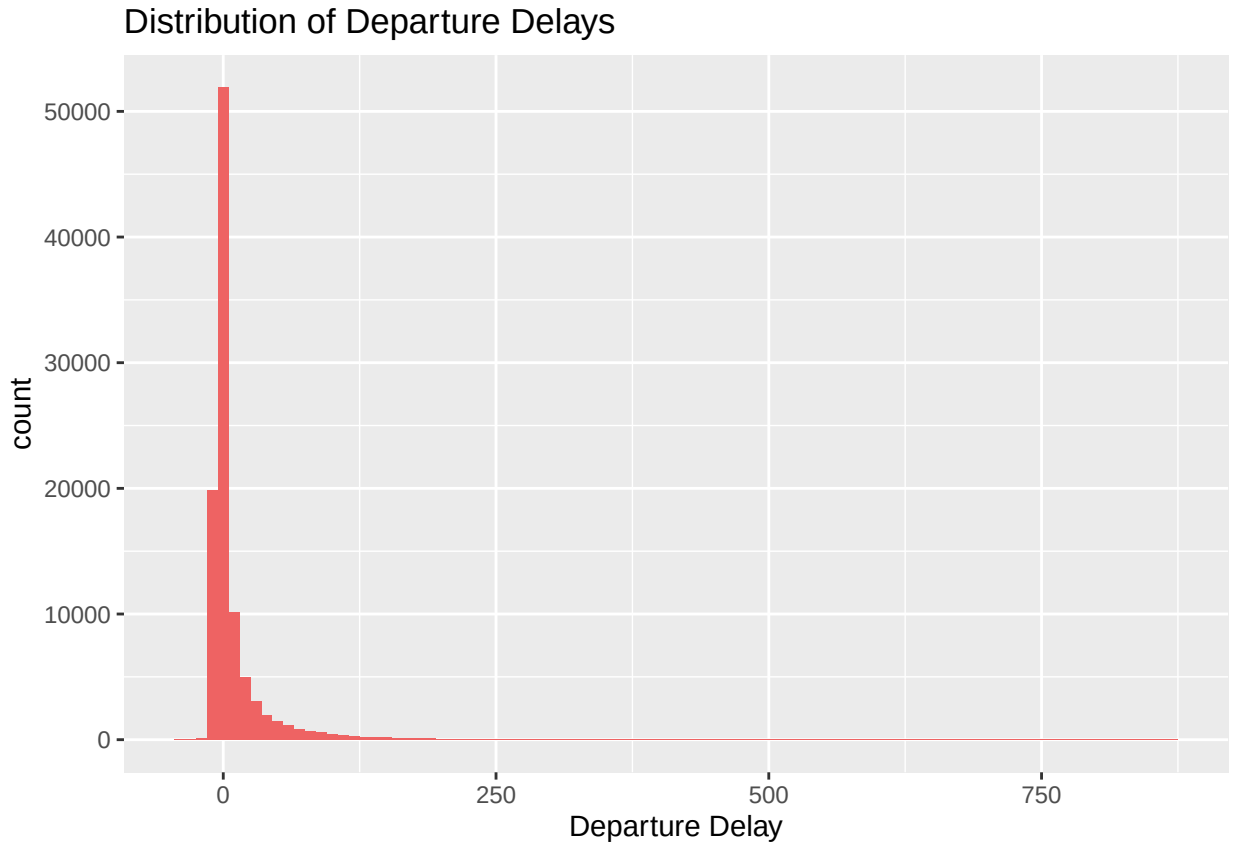
Visual Story Telling: Part 2-Flights at ABIA

Visualizations

Exploratory Data Analysis

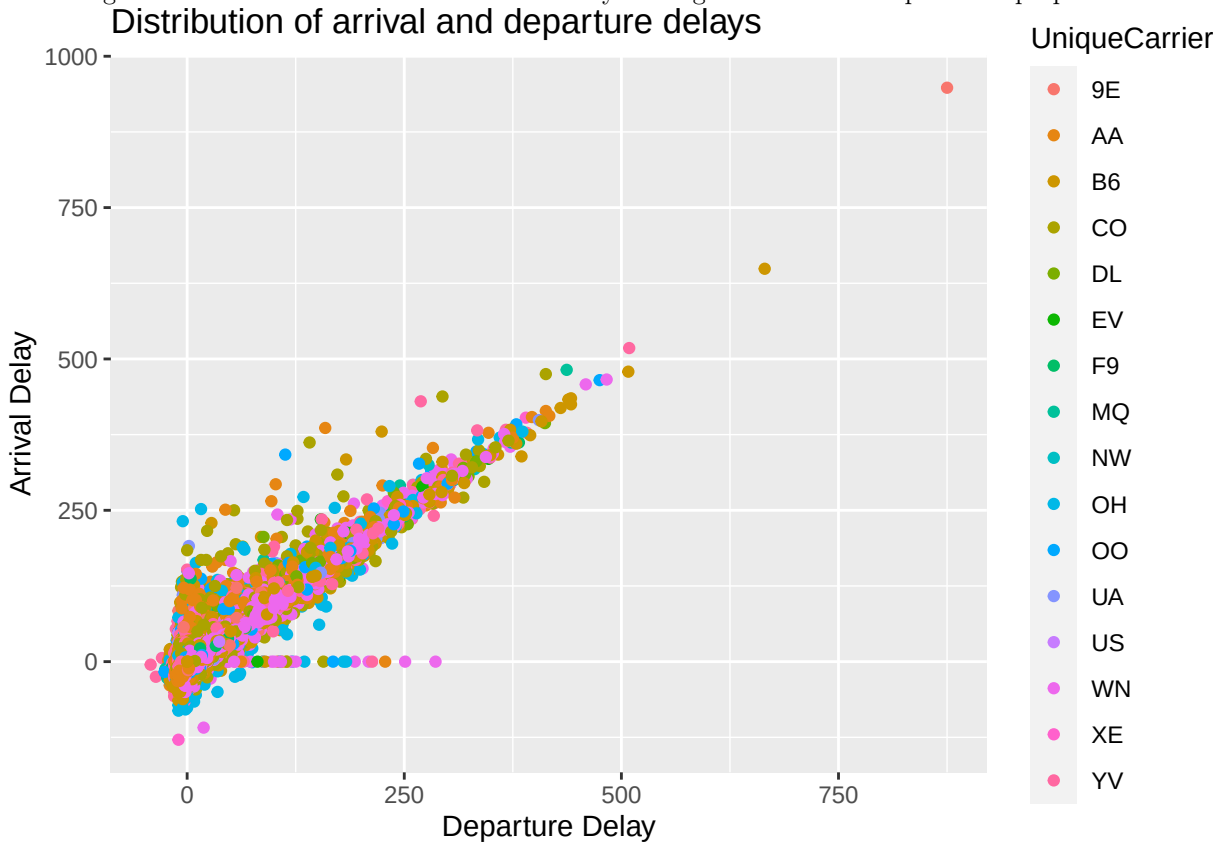
Exploratory data analysis of 'flight' data during the year 2008 at Austin airport. Data set contains ~99k records of flight data


Distribution of Arrival Delays
30000
20000
10000
0
count
0
250
500
750
1000
Arrival Delay


Distribution of Departure Delays
50000
40000
30000
20000
10000
0
count
0
250
500
750
Departure Delay

Based on the histograms of arrival and departure delays, the mean delays of both are centered around zero with large skews in both towards delays. Based on the histograms there are a few instances where departures and delays are early, I thought this was interesting considering that I was unaware flights were able to depart early besides everyone arriving early.

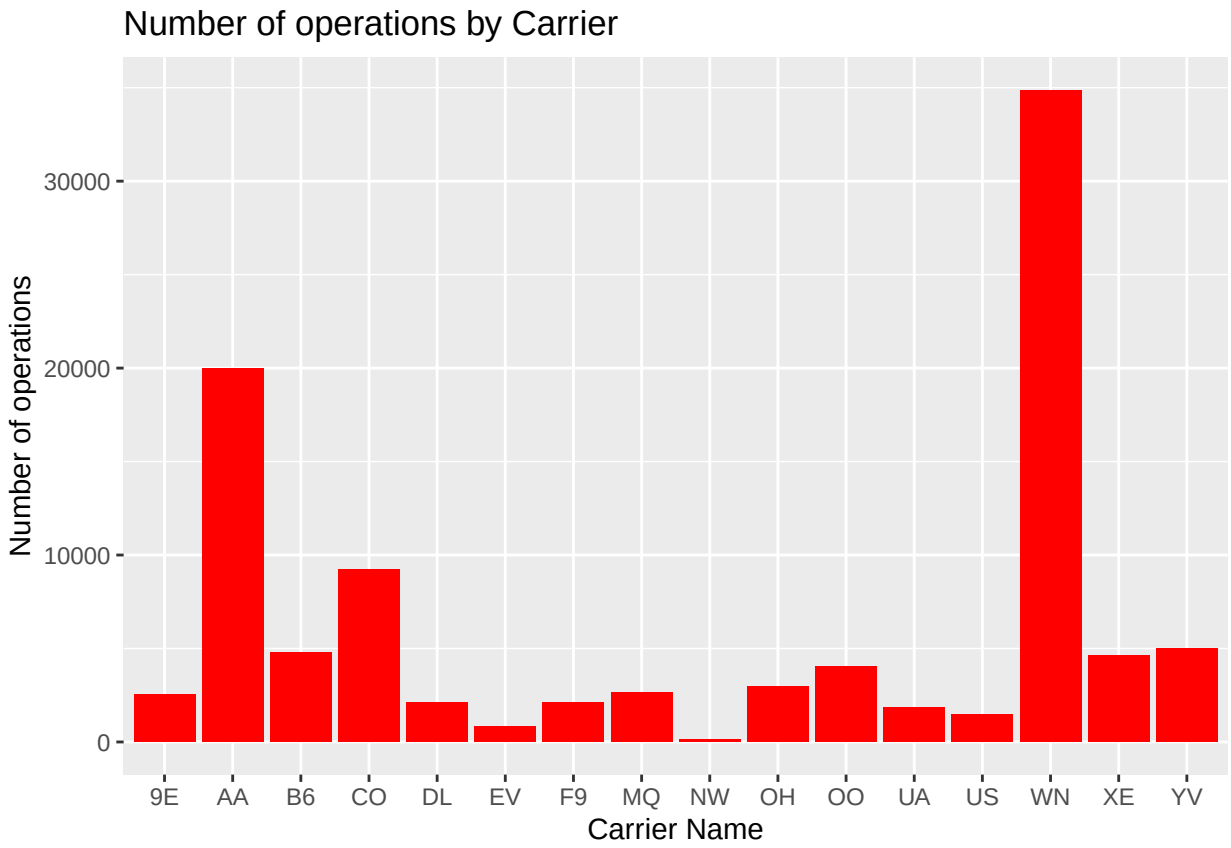
Next step taken was to look at the correlation between arrival and departure delays to observe if any particular carrier is deviating from normal behavior. This will be done by looking at a distribution plot with proper color coding.



There are outliers for some of the carriers when looking at the distribution plot. Almost perfect correlation between the arrival and departure delays suggesting a linear relationship. Some carriers did compensate for the departure delays(going really really fast) were outliers.

Air carrier operation at Austin Airport

Next is to observe the carrier operation at Austin Airport.



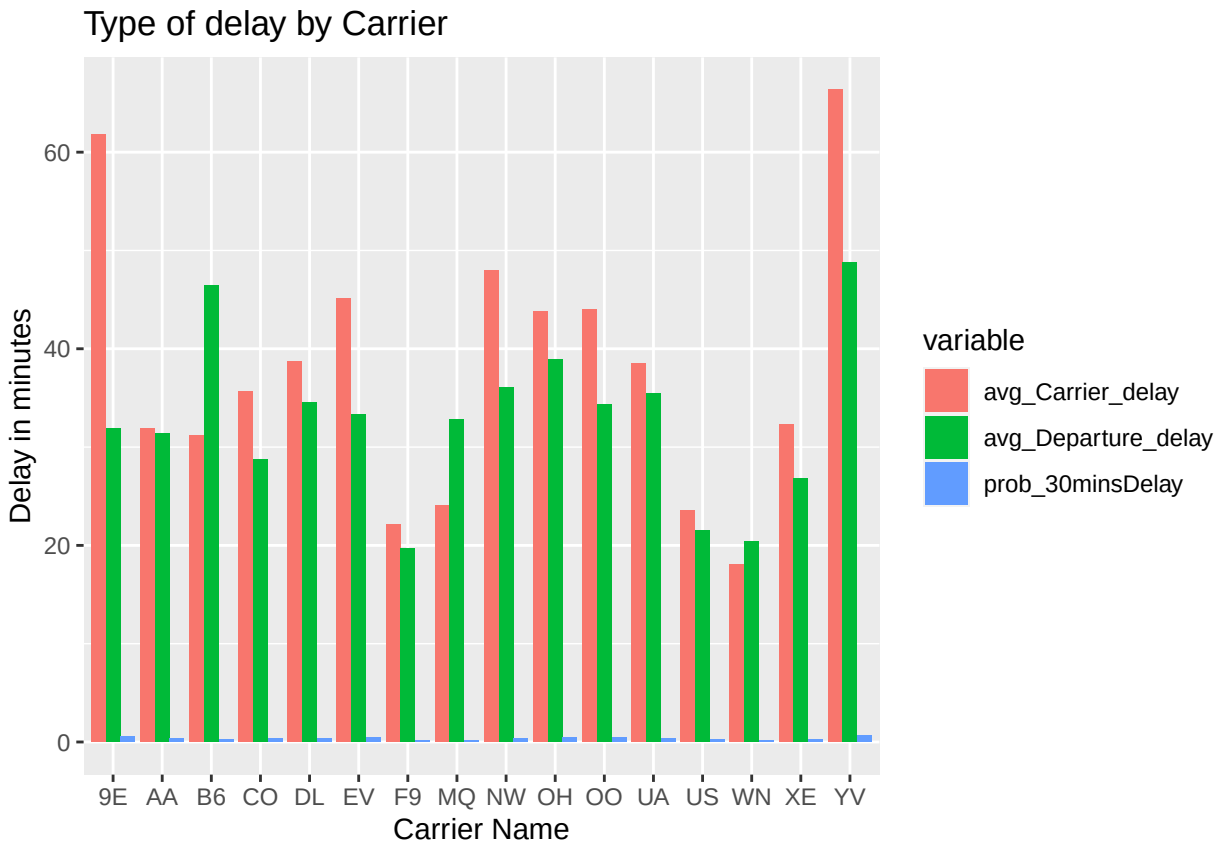
Southwest(WN) tops the list with almost 40k operations, followed by Alaskan Airlines(AA).

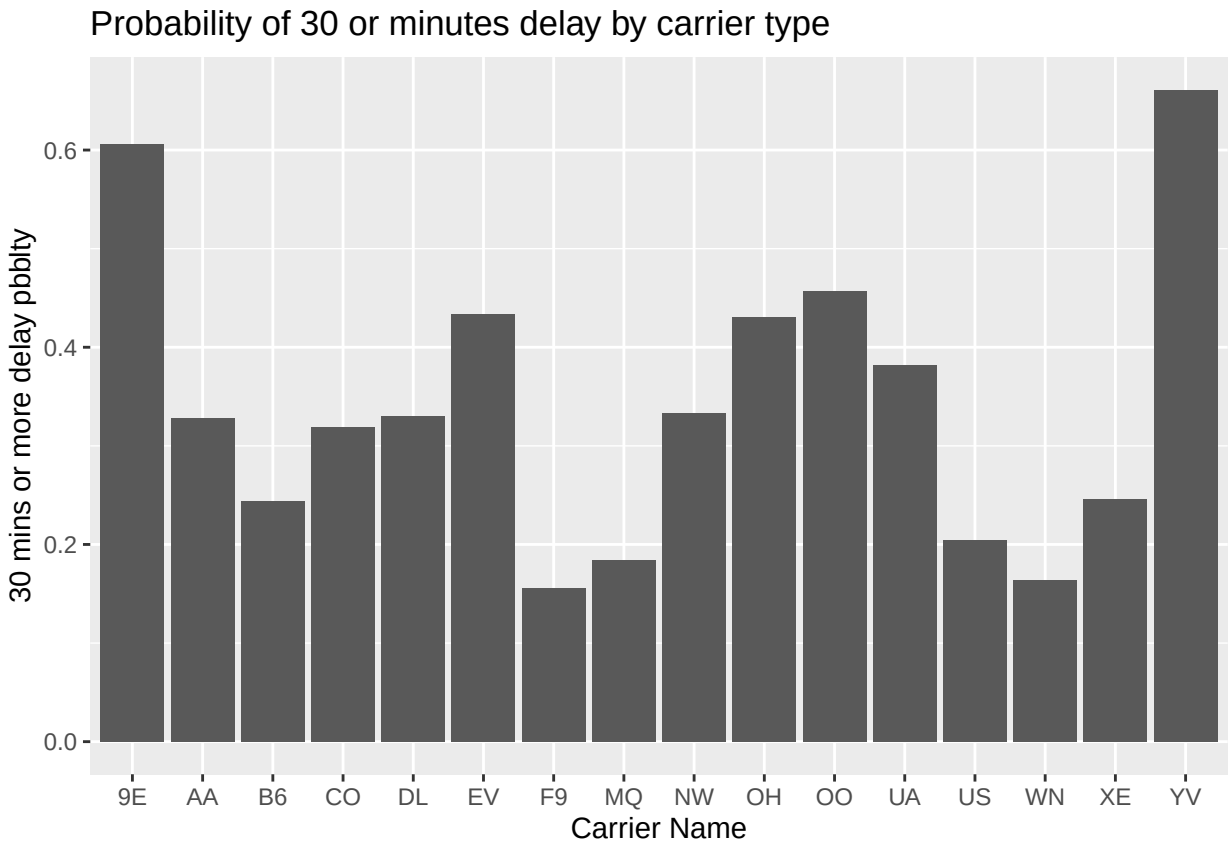
What is the most reliable carrier?

Probability(carrier delay > 30 mins): 1. Lowest: Southwest(WN) and Frontier Airlines(F9) 2. YV and 9E > 60%.

Summary of Reliable Carriers: Southwest(WN) is the most reliable with 40k operations suggesting the results are reliable. The avg carrier delay is 18 minutes. Avg departure delay is less than avg arrival delay. The airlines – F9 MQ, US, WN and XE > 30% probability of delay >30 min, but they have a low number of operations unlike Southwest. Relatively, Alaskan Airlines(AA) which has ~20k operations outperforms many carriers that have less operations.

Unreliable Carriers: 1. 9E and YV have an avg carrier delay > 1hr. 2. With just 121 operations, NW has a high avg carrier delay of 48 minutes.

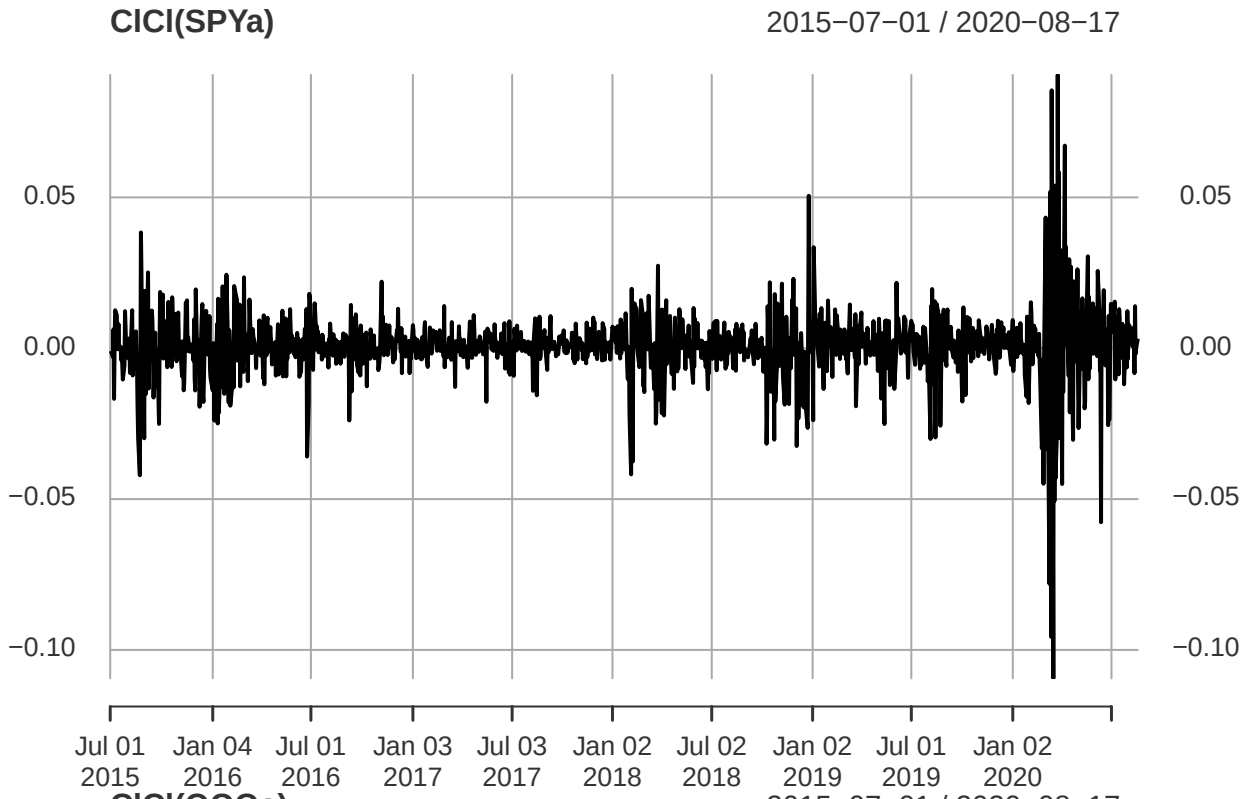

Type of delay by Carrier
Delay in minutes
60
40
20
0
9E AA B6 CO DL EV F9 MQ NW OH OO UA US WN XE YV
Carrier Name
variable
avg_Carrier_delay
avg_Departure_delay
prob_30minsDelay

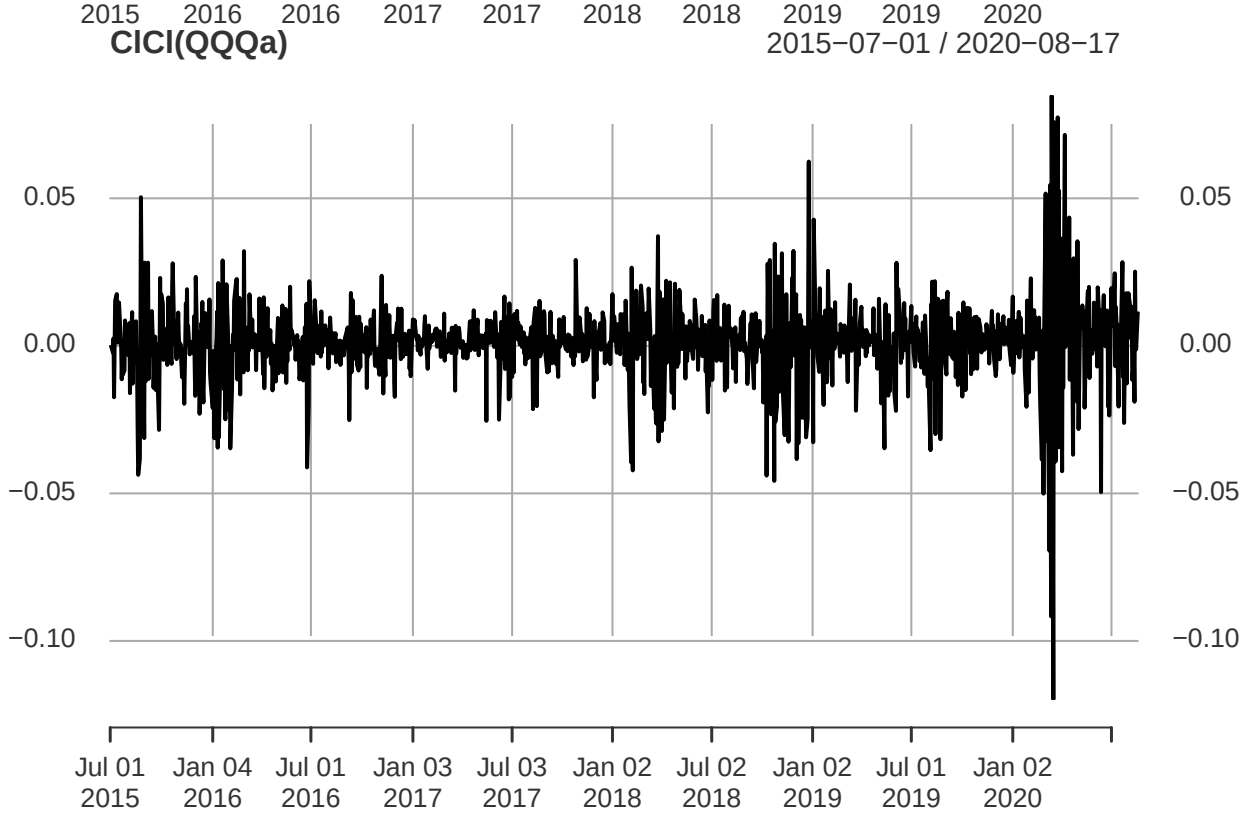

Probability of 30 or minutes delay by carrier type
30 mins or more delay pbblty
0.6
0.4
0.2
0.0
9E AA B6 CO DL EV F9 MQ NW OH OO UA US WN XE YV
Carrier Name

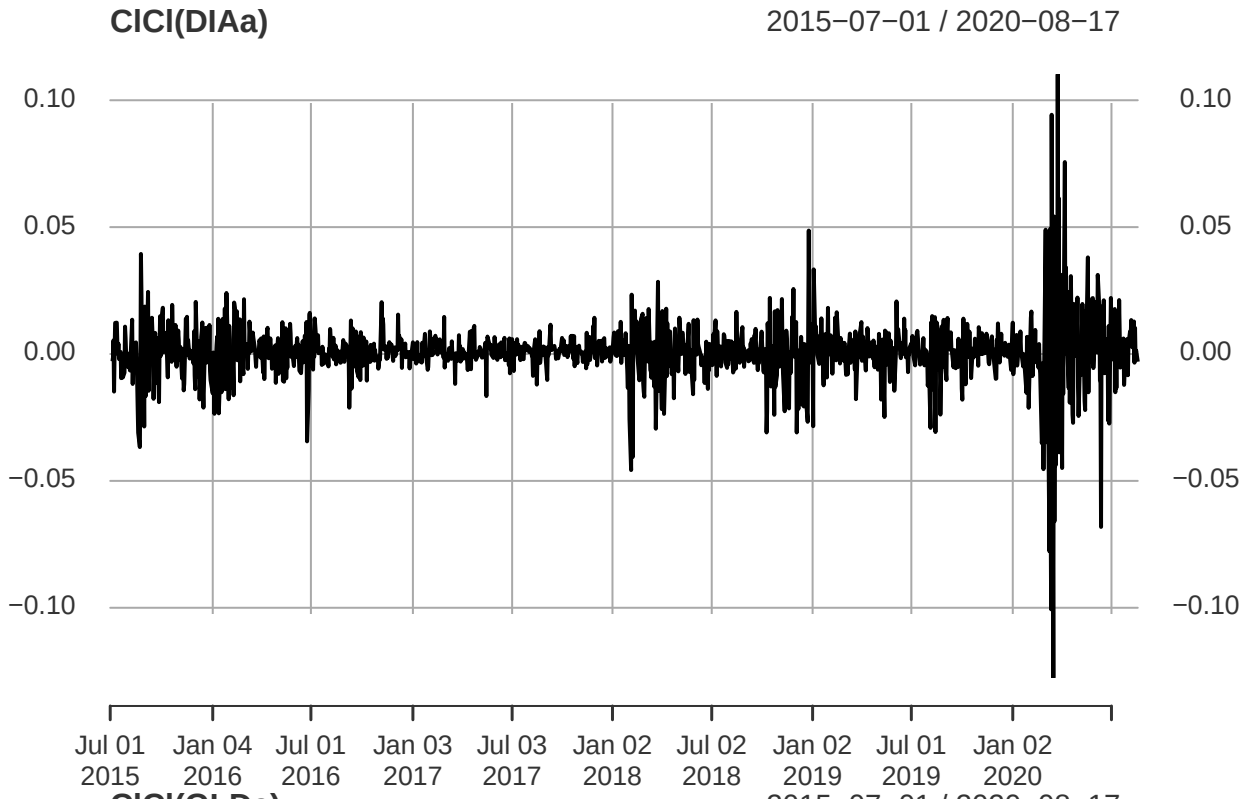
Portfolio Modeling

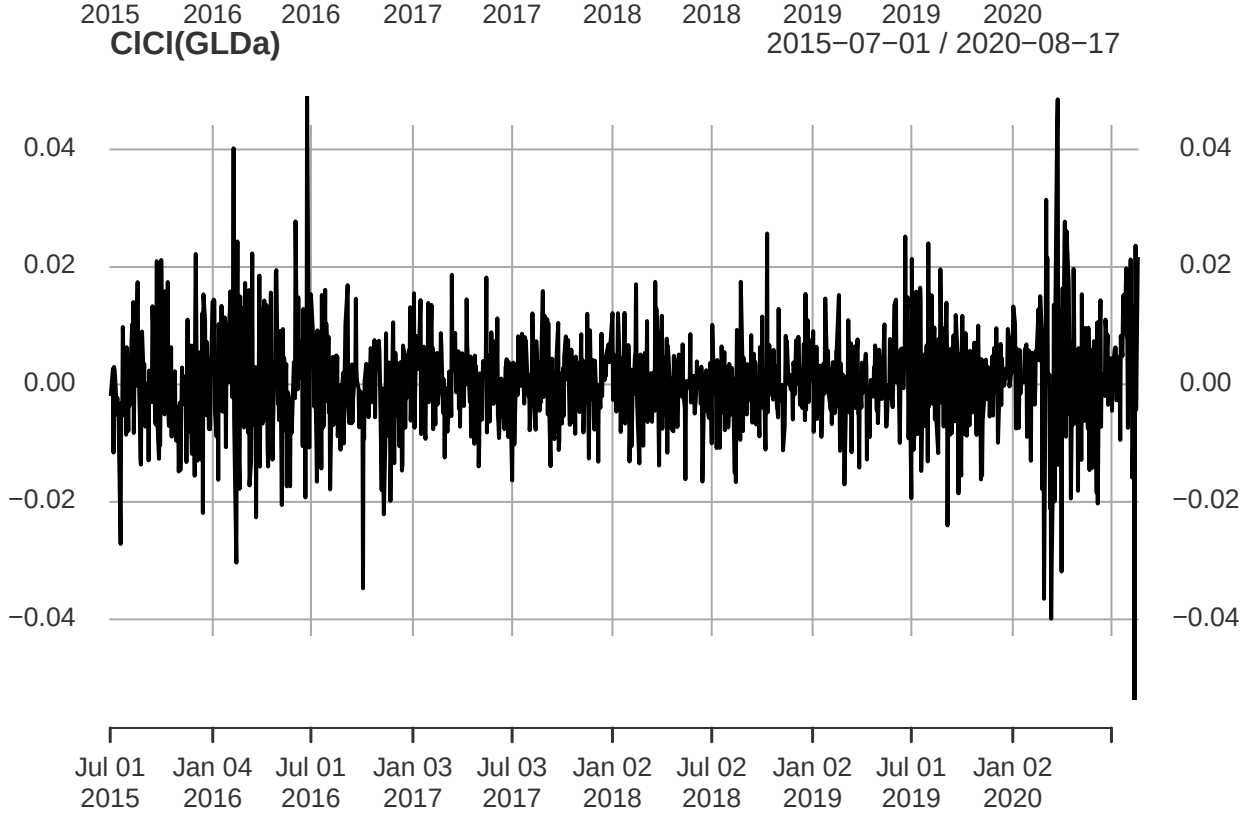
The ETFs selected were SPY, QQQ, DIA, and GLD. SPY follows the S&P 500 which are large cap stocks, QQQ follows the NASDAQ which is full of tech heavy stocks, DIA follows the DJIA which includes Apple, and GLD follows gold which is typically used as a hedge during times of market instability.

Data importing from the interweb and processing



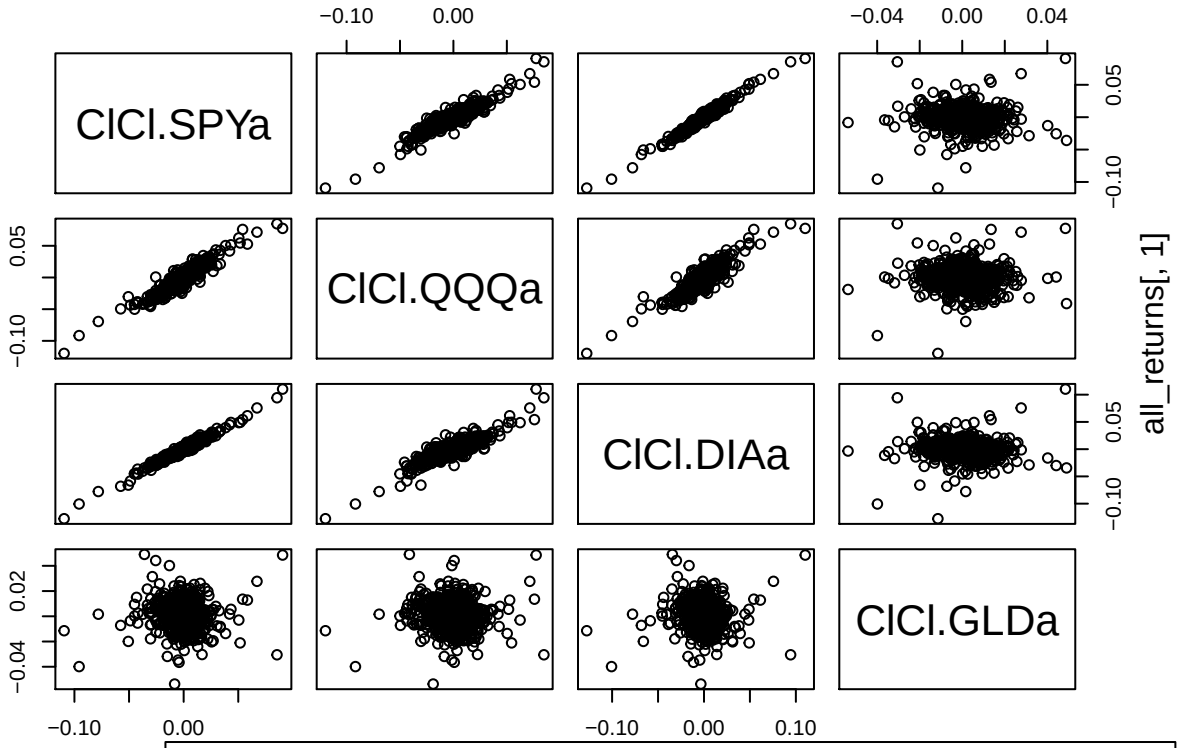


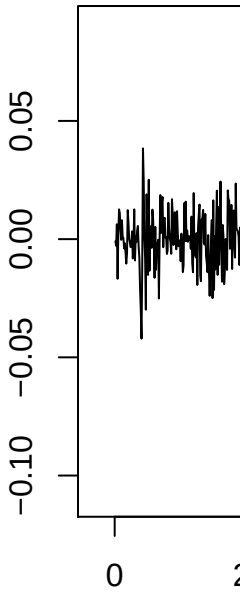


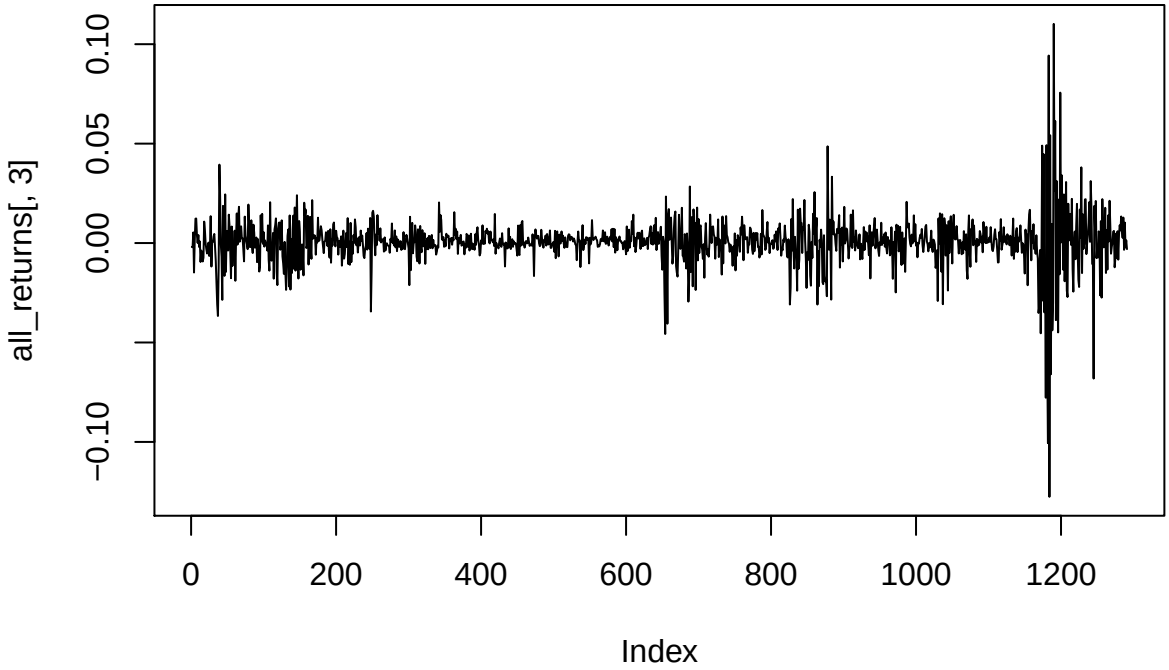


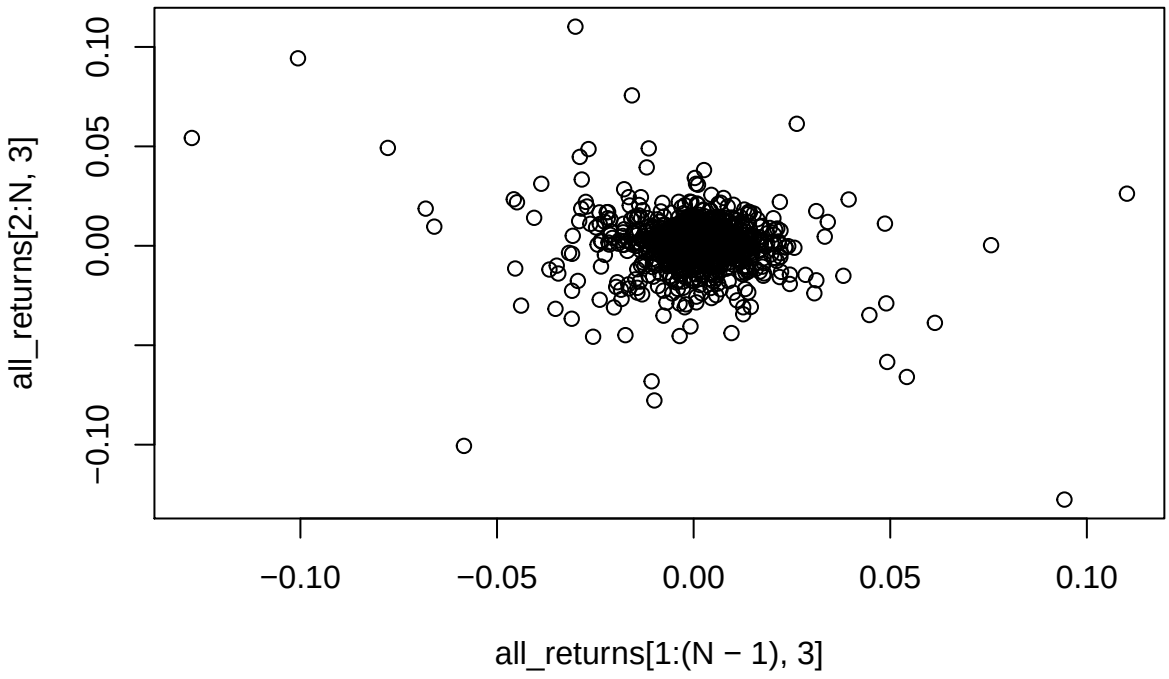

```
##                ClCl.SPYa     ClCl.QQQa    ClCl.DIAa    ClCl.GLDa
## 2015-07-01            NA            NA           NA           NA
## 2015-07-02 -0.0009156723  0.0002779744 -0.001972892 -0.001964645
## 2015-07-06 -0.0028459650 -0.0023158870 -0.002089862  0.002684288
```

```
## 2015-07-07  0.0062887142  0.0025070010  0.005377021 -0.011600893
## 2015-07-08 -0.0167772567 -0.0174122903 -0.014806085  0.002979361
## 2015-07-09  0.0018090011 -0.0005655293  0.001485686  0.002430507
```

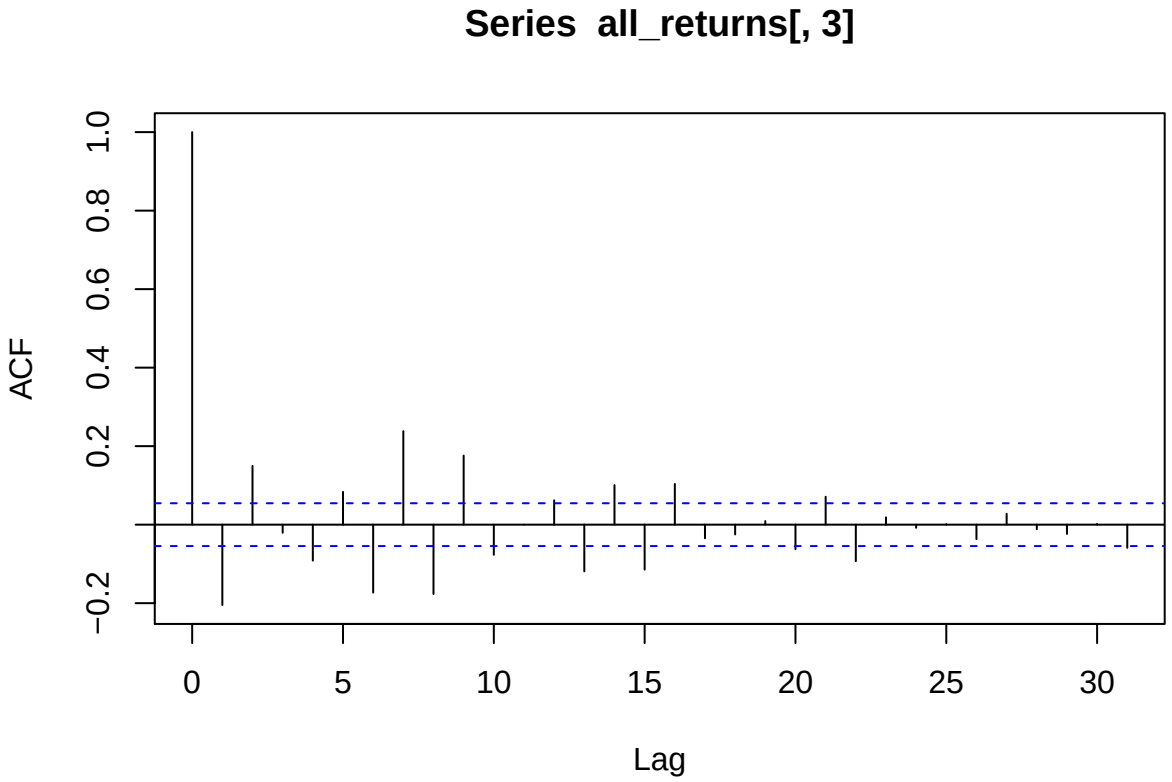






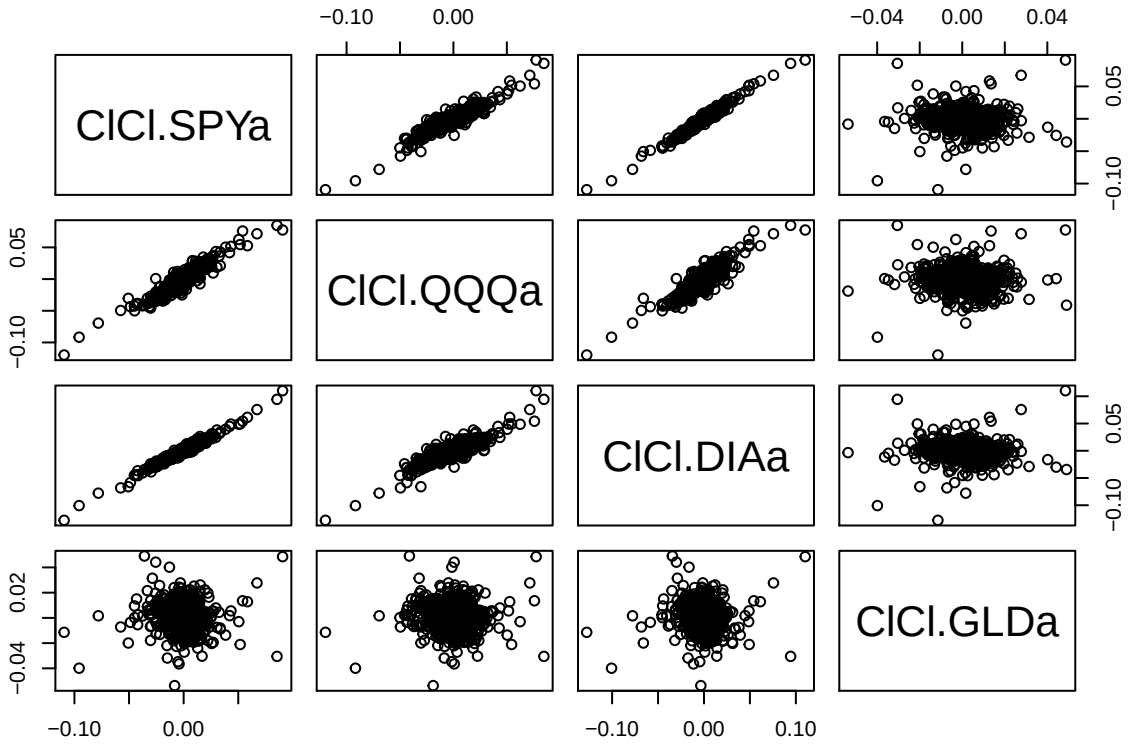


The ditrbutions plotted in the first graph make sense as the market has been doing well the past five years overall so there is a linear trend. The plots comparing with GLD also make sense as even right now GLD is being used as a hedge against inflation and has reached all time highs. There is no correlation between day to day returns.



The graphs show that returns are uncorrelated from one day to the next which is important as they would be easy to exploit by looking at past information suggesting a weak form market efficiency.

Setup



Verifying data makes sense again for a single day.

Sample Simulation

$100000 to invest with equal weighting in each Update the value of holdings Assumes an equal allocation to each asset

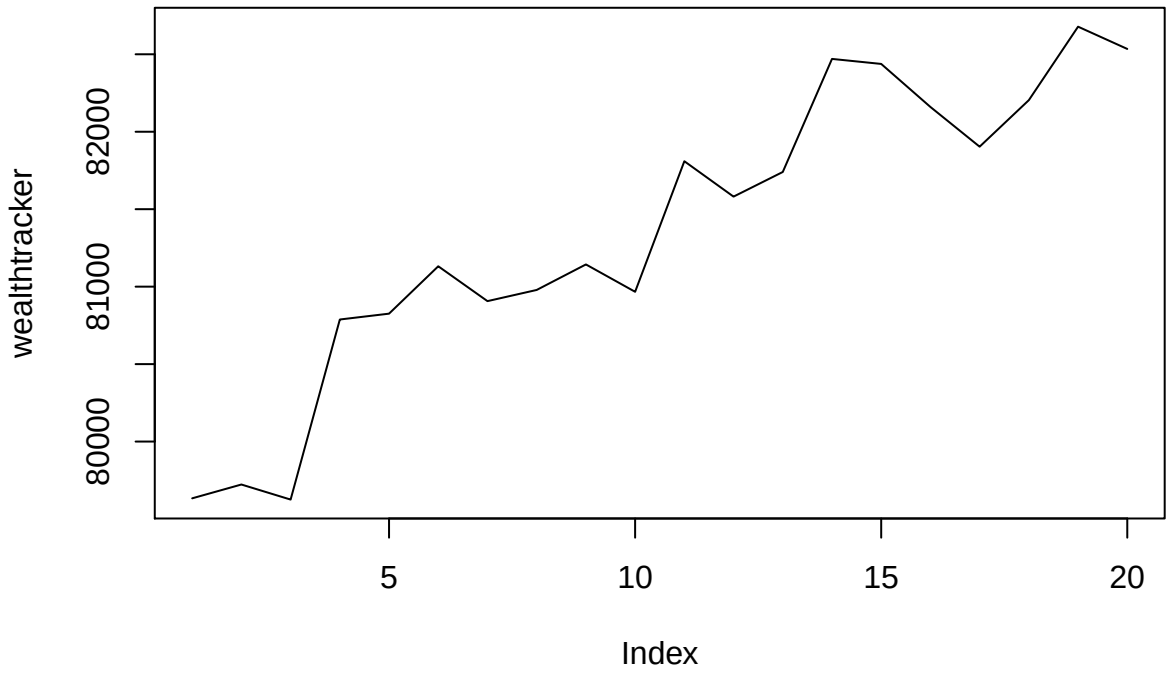
```
##            ClCl.SPYa ClCl.QQQa ClCl.DIAa ClCl.GLDa
## 2018-04-16  20164.43  20152.45   20167.5  20028.25
```

```
## [1] 80512.63
```

Lost about $18500.

Now loop over four trading weeks let's run the following block of code 5 or 6 times to eyeball the variability in performance trajectories

```
## [1] 82534.46
```



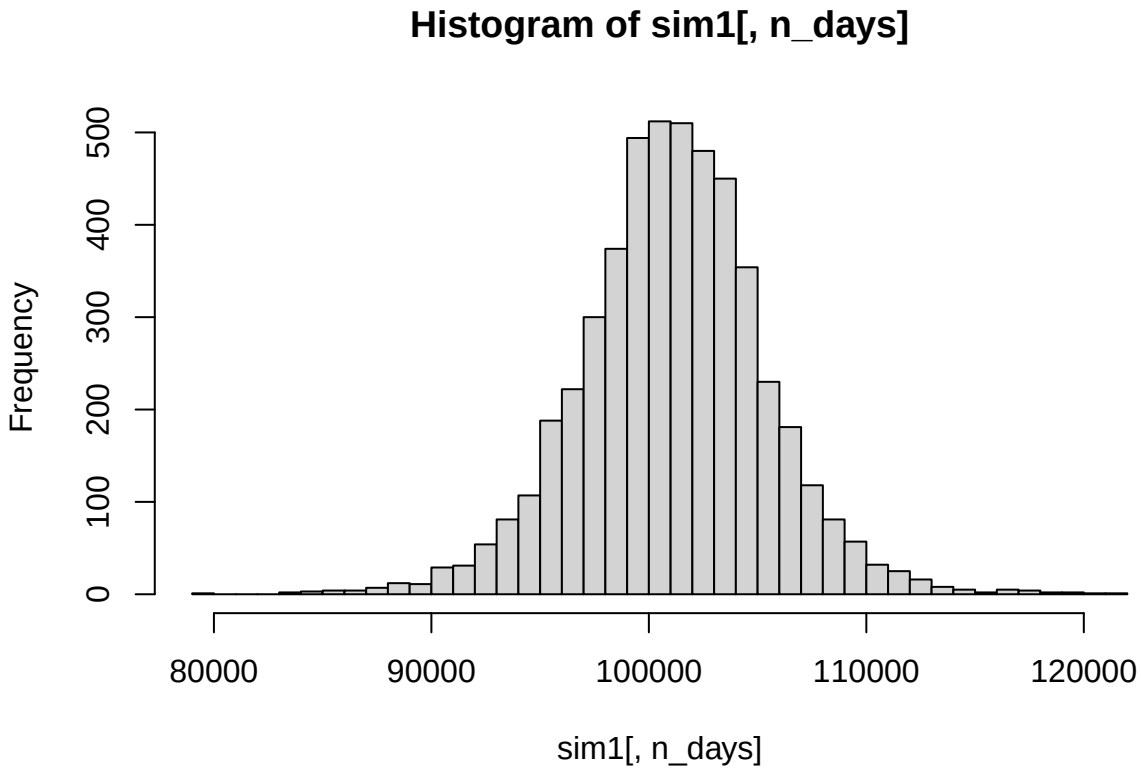
Lost $15000 this time.

**Simulations with Boorstrapping

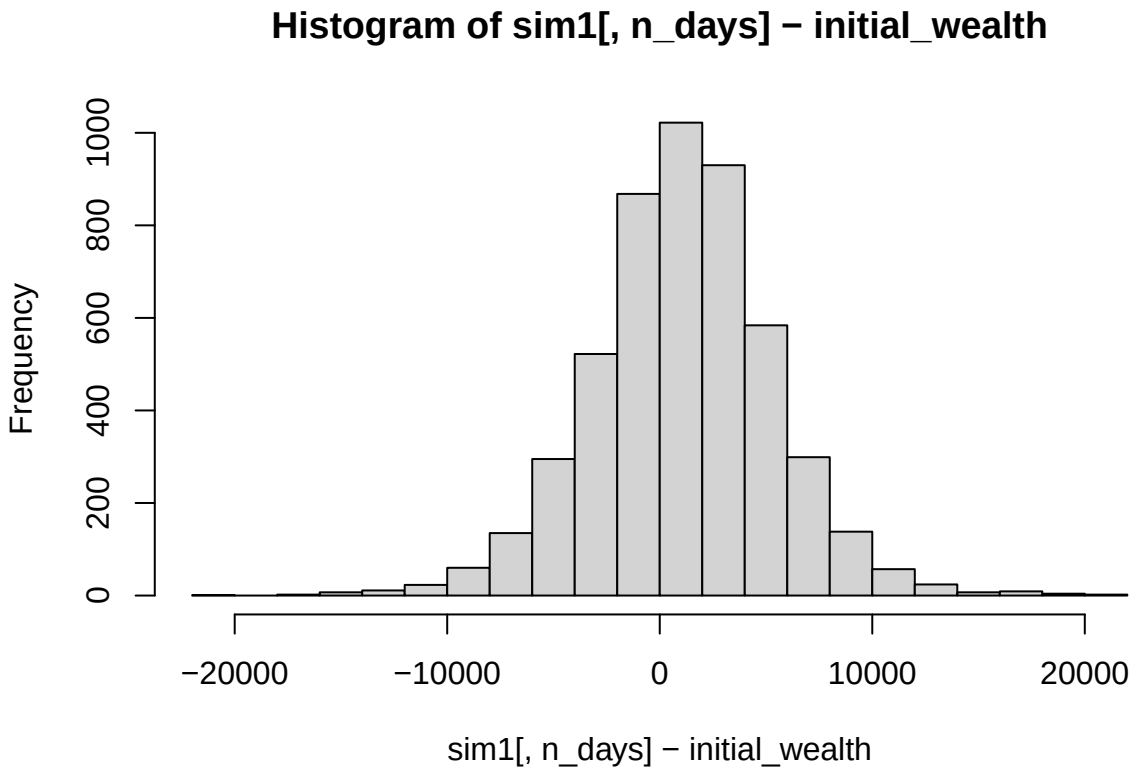
Simulation 1

$100000 to invest with equal weighting in each Update the value of holdings Assumes an equal allocation to each asset

```
##               [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## result.1  99938.41  98928.64  99009.57  99126.25  99216.81  99079.83 100398.25
## result.2 101071.22 102383.07 101707.57 100991.09 100943.11 101433.68 102040.87
## result.3  99953.94 100443.87  98917.69  97282.07  96917.26  96815.07  94325.09
## result.4 100135.55 100000.34 101479.18 102363.62 102936.78 104240.86 105446.01
## result.5 100202.49 100446.98 100653.25 101891.21 102013.65 101283.16  98421.39
## result.6 100318.47 100413.24 100539.35 101309.82 101324.57 101592.02 101485.05
##               [,8]      [,9]     [,10]     [,11]     [,12]     [,13]     [,14]
## result.1 100632.23 100965.35 100582.86 100762.54 100934.55 101872.80 102525.74
## result.2 101586.18 101877.58 102564.11 103417.41 103458.14 105820.24 105898.90
## result.3  95208.90  96372.75  94603.27  95075.77  95099.69  95481.15  95794.01
## result.4 106441.80 106506.00 106946.95 105739.78 105203.98 104943.48 104717.69
## result.5  99201.03  99510.70  98860.04  99292.87  97872.58  97585.16  97652.22
## result.6 104933.66 104892.95 105287.69 105081.38 104974.56 105348.78 105440.23
##              [,15]     [,16]     [,17]     [,18]     [,19]     [,20]
## result.1 103311.29 105060.72 104672.80 104252.37 104851.90 104416.56
## result.2 106581.37 106892.74 106906.47 107483.81 107624.50 108364.67
## result.3  96332.73  96513.91  97387.29  97685.77  97798.67  97959.45
## result.4 104799.01 102684.10 103649.55 104216.44 101986.39 102340.59
## result.5  97774.12  98044.85  97746.30  98205.73  98448.87  96604.79
## result.6 105475.57 105711.86 106445.41 106216.79 106264.72 106622.98
```




```
## [1] 101094
## [1] 1093.987
## [1] 0.01093987
```



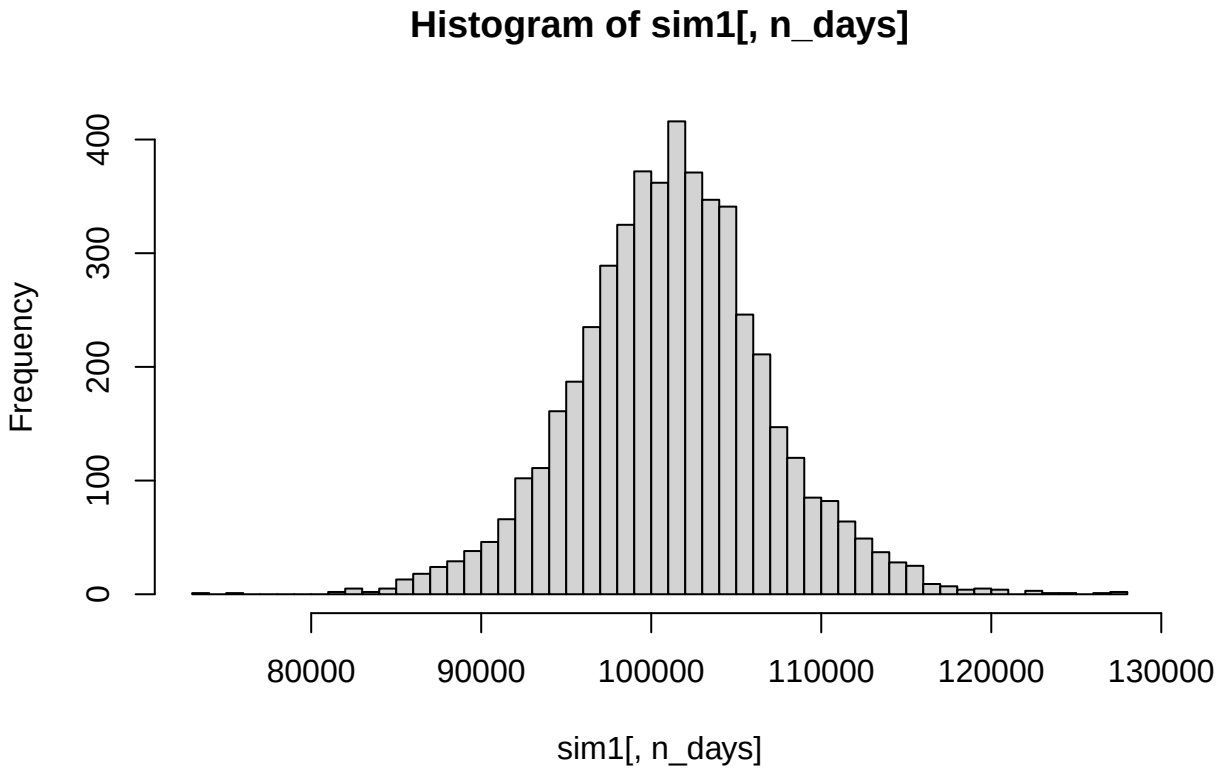
```
##        5%
## -5824.537
```

The expected profit is $1270.42 and a return of 1.27% which shows some return. The 5% VaR = 5744.82 so thats how much they can expect to lose at the 5% quantile performance.

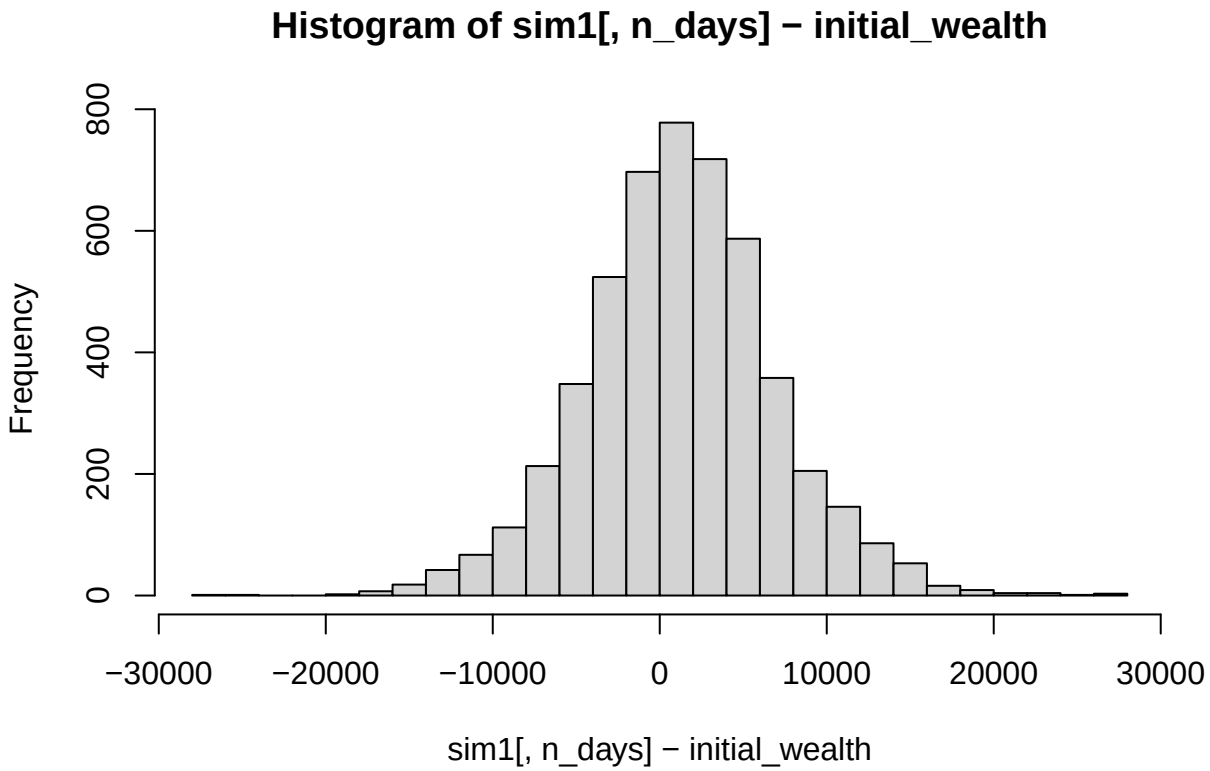
Simulation 2

$100000 to invest with aggressive weighting in SPY, QQQ, DIA and nothing in GLD. Update the value of holdings.

```
##               [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## result.1  99940.31  98858.20 100288.48  97801.29  94589.54  94853.48  95966.90
## result.2  96295.25  96178.14  98195.24  96769.03  97391.18  97666.41  97734.88
## result.3 100336.52 101226.11 100250.99 100399.35 100398.93 100586.79 100412.96
## result.4  97996.98  98078.07  98133.84  97836.59  98867.41  99473.24  99577.13
## result.5  99983.18 100029.28  99784.39  99116.89 100883.67 101514.53 101380.65
## result.6 100420.96 101016.28 100884.56 101515.64 102546.95  98224.68  99032.67
##               [,8]      [,9]     [,10]     [,11]     [,12]     [,13]     [,14]
## result.1  96669.27  93267.96  93032.42  92811.88  93381.36  93181.37  93185.69
## result.2  97899.62  98248.77  99386.19  99405.75  97686.32  97830.99  97909.70
## result.3 101291.60 101531.38 102009.68 101708.17 100744.96 101642.97 102221.12
## result.4  99591.22  98654.33  98603.17  98933.11  98954.71  99770.29 100257.94
## result.5 102008.44 102318.42 100808.97 103382.52 102626.15 103236.78 103764.21
## result.6  98912.74  99307.75 100024.18 100582.78 100625.06  98690.58  97834.13
##              [,15]     [,16]     [,17]     [,18]     [,19]     [,20]
## result.1  97131.42  97265.73  97640.00  97660.15  98041.33  98106.39
## result.2  99598.47 100578.39 102103.43 103684.61 104586.03 103890.93
## result.3 101934.44 102095.60 101287.36 101151.38 100635.84 100567.56
## result.4 101502.05 101038.77 101223.30 101550.87 101263.20 104307.08
## result.5 104135.54 103708.25 104672.57 105012.12 107285.43 107648.50
## result.6  98631.91  99120.94  97879.97  97905.06  97868.58  98422.62
```




```
## [1] 101237.2
## [1] 1237.238
## [1] 0.01237238
```



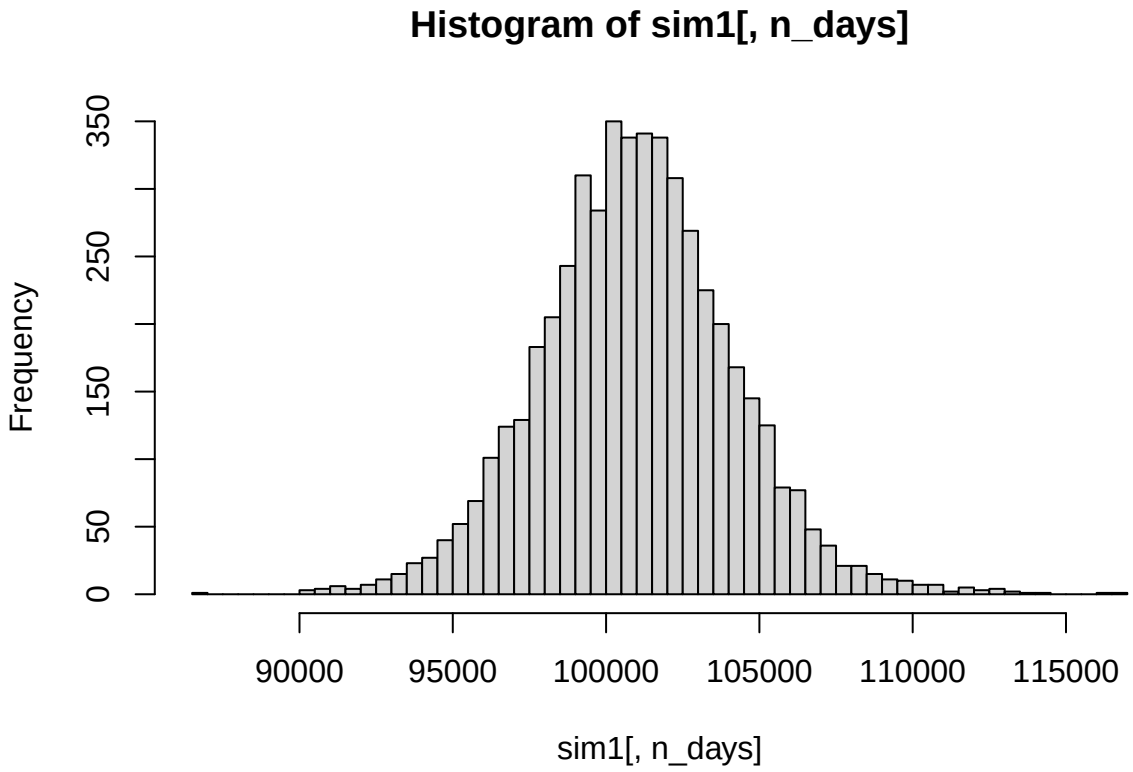
```
##        5%
## -7976.729
```

The expected profit is 1283.11 and a return of 1.28% over 4 weeks and this produces returns which appear to be normal based on the plots which could be due to normal variability in the market since that is where all the weights are. The 5% VaR = 7726.34 so thats how much they can expect to lose at the 5% quantile performance. This VaR is higher than in simulation 1 meaning it is rskier. This isn't bad but there is no gold hedge so when times are tough the portfolio will likely take a huge fall since there is full exposure to the market but when times are good it will likely perform better.

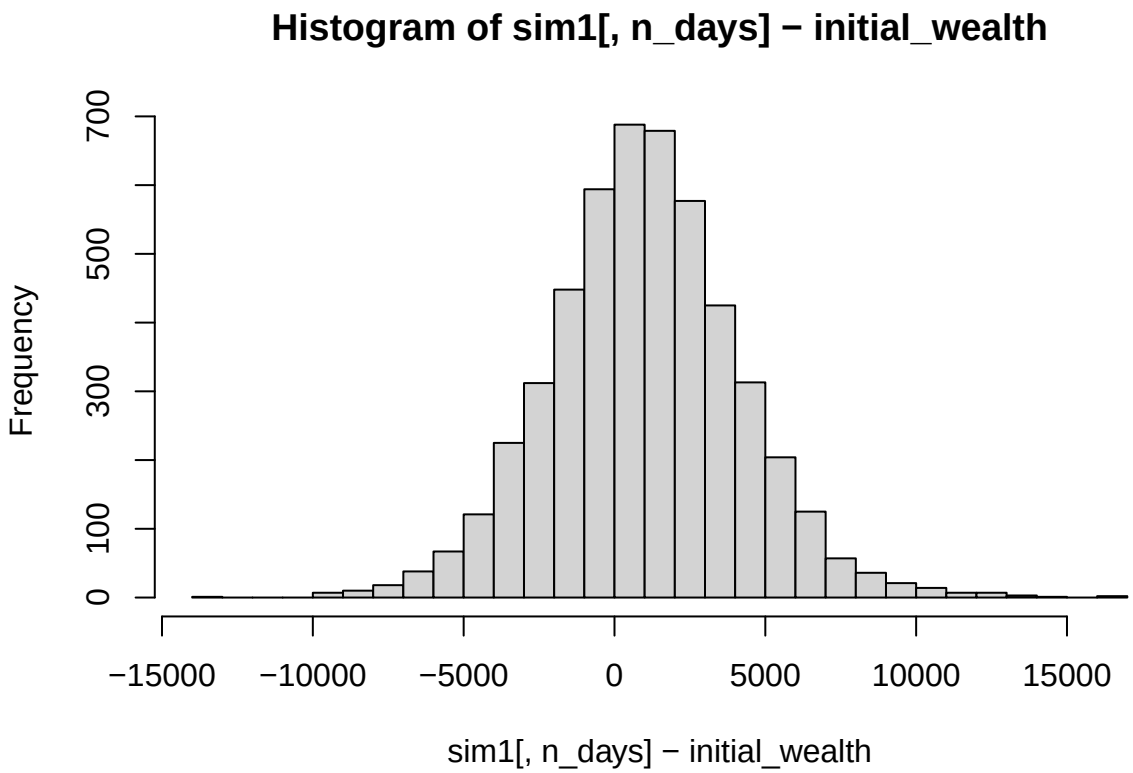
Simulation 3

$100000 to invest in mostly weighting in GLD and less in SPY, QQQ, DIA. Update the value of holdings

```
##               [,1]     [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## result.1 100256.72 100032.4 100147.81  99957.05 100275.49  99950.63 100000.87
## result.2  99578.33 100037.5 100945.60 101444.40 100883.56 100624.32  99544.24
## result.3  99746.14 100287.4 100256.69 100341.01 100211.84 100513.63 101181.60
## result.4  99868.33 100643.9 101210.47 103294.32 102403.08 102452.44 101184.23
## result.5  99813.01  99567.4  99490.71  99223.20  99006.76  97992.76  97619.96
## result.6 100642.05 100005.6  99880.83 100683.68 100864.04 101507.28 101230.05
##               [,8]      [,9]     [,10]     [,11]     [,12]     [,13]     [,14]
## result.1 100019.18 100126.83  99825.25  98927.80  98667.39  99189.12  99205.85
## result.2  98834.07  98403.59  97653.65  97333.18  97009.08  97132.59  97060.93
## result.3 100650.96 101015.64 101327.49 100536.06 100603.01 101813.66 102010.43
## result.4 100979.23 100061.18  99890.57  99465.28  99603.52  99451.59  99720.76
## result.5  97753.91  96192.92  95931.88  94768.28  94266.35  94342.17  94608.36
## result.6 101456.25 101256.64 102129.51 102239.20 104213.32 105392.73 104429.09
##              [,15]     [,16]     [,17]     [,18]     [,19]     [,20]
## result.1  99323.54  98753.42  98593.17  99893.15 100439.64 100558.77
## result.2  97546.47  97176.94  96801.22  97013.47  96240.27  96136.41
## result.3 102416.63 103563.40 102667.80 101815.98 102864.95 102974.54
## result.4 100676.79 101048.54 100969.90 100988.97 101543.91 101855.24
## result.5  94588.44  95804.12  96042.17  96189.19  96000.25  96413.66
## result.6 103456.20 103033.34 102399.39 103279.08 103758.30 103028.68
```




```
## [1] 100992.5
## [1] 992.4836
## [1] 0.009924836
```



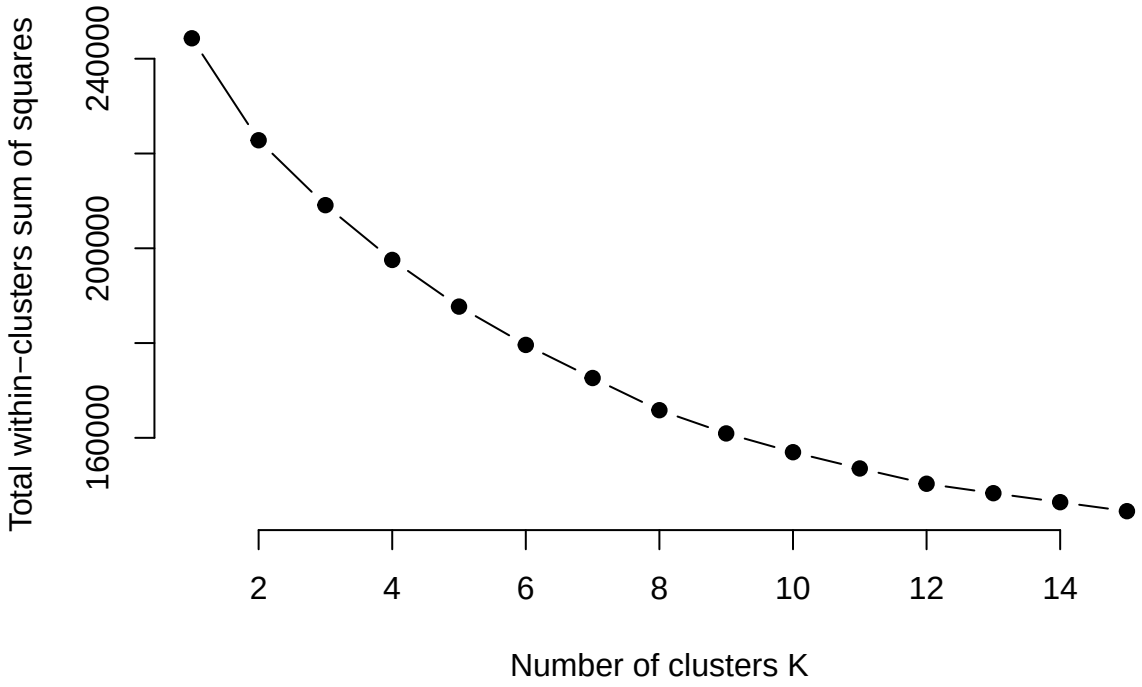
```
##        5%
## -4038.783
```

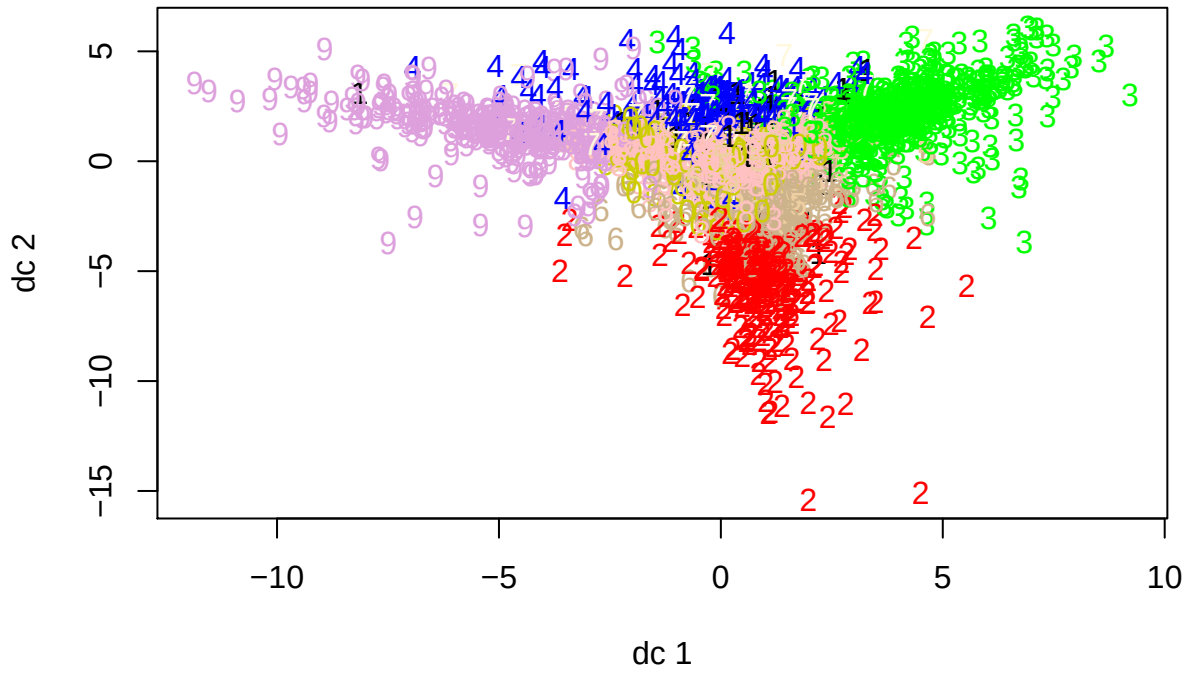
The expected profit is $932.78 and a return of 0.93% over 4 weeks and this produces returns which appear to be normal based on the plots which could be due to normal variability in the market since that is where all the weights are. The 5% VaR = 4244.35 so thats how much they can expect to lose at the 5% quantile performance. This VaR is lower than in simulation 1 meaning it performs better. This still has investments in the market but is much more stable and less risky becasue of the investments in gld as a hedge but slightly lower investments.

Market Segmentation

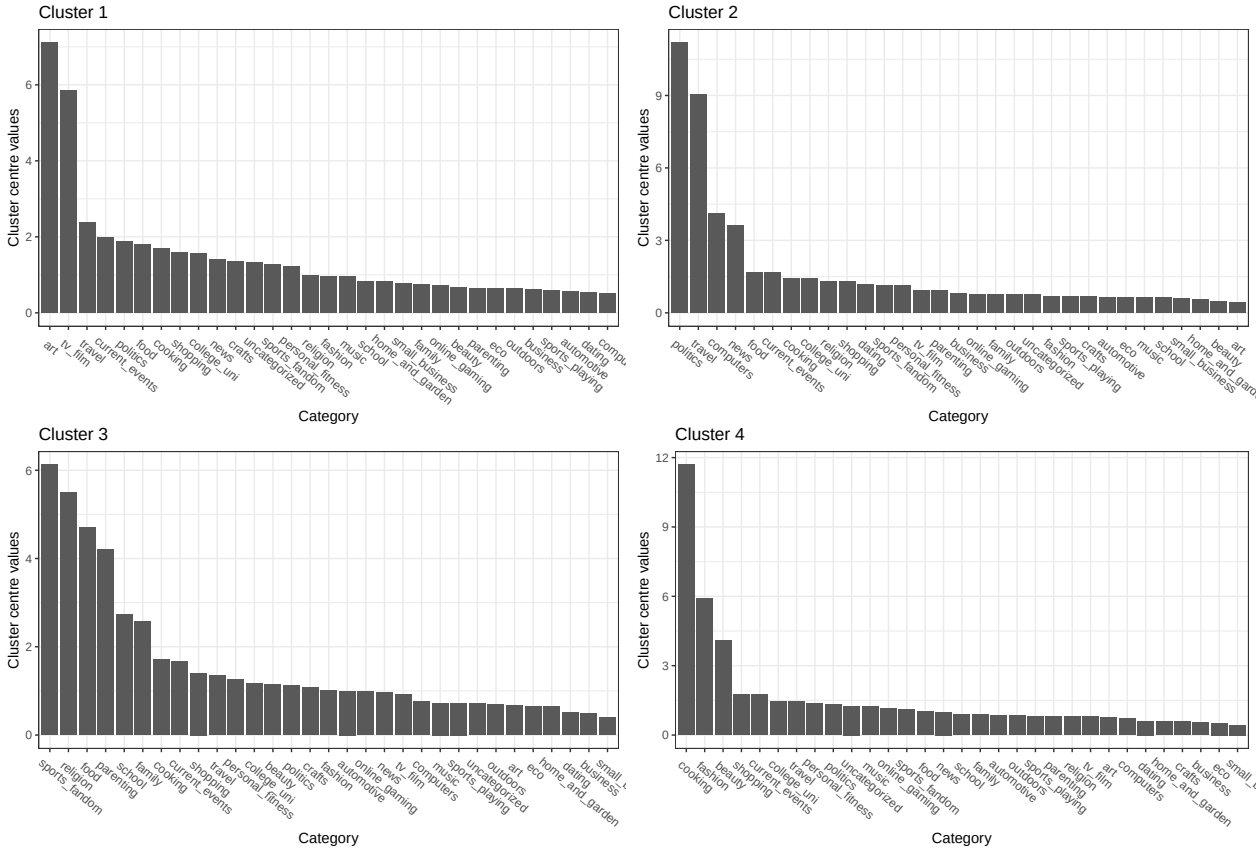
Data Processing and the such

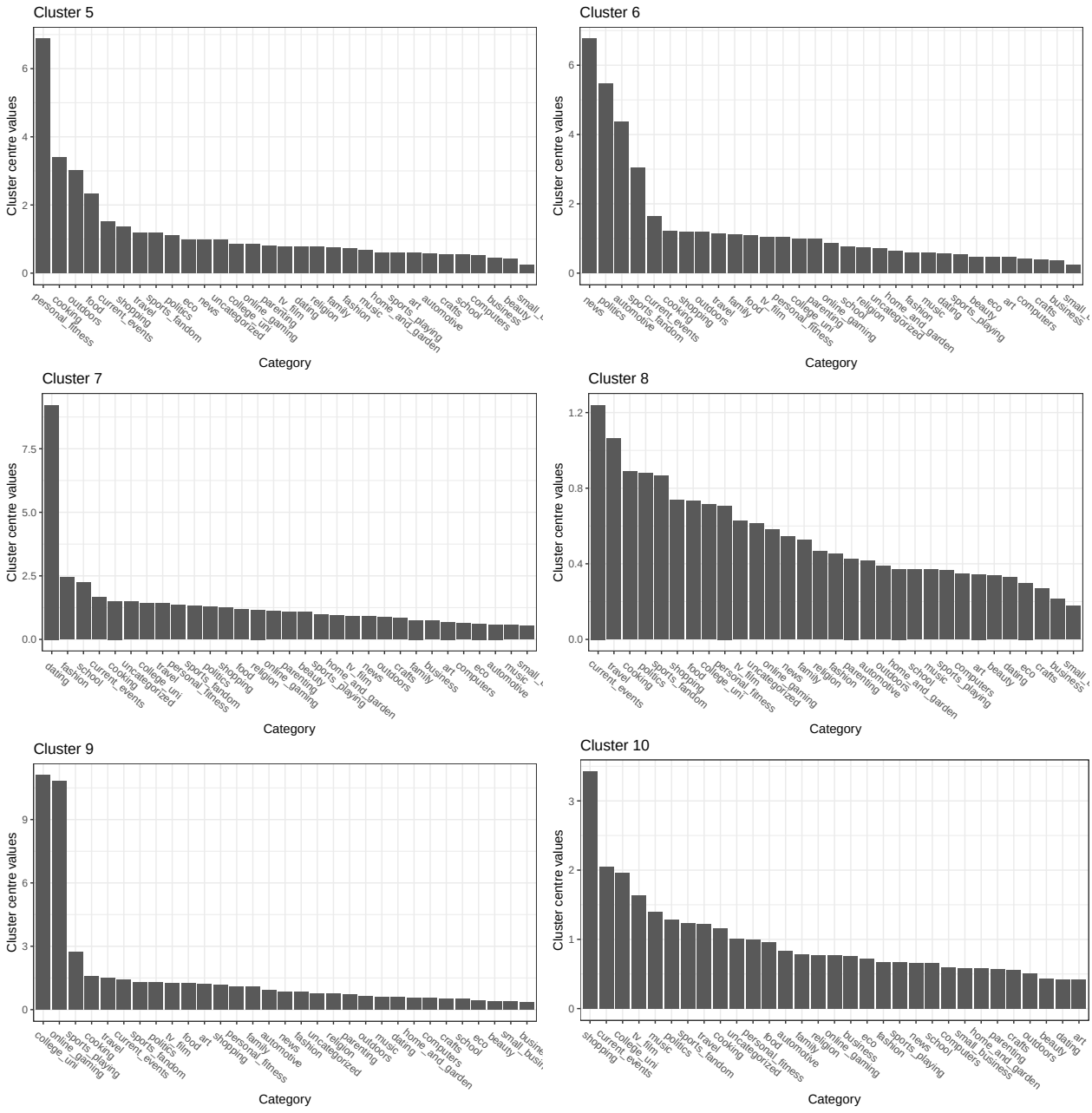
```
##  [1] 244311.0 222796.5 209096.5 197541.5 187691.6 179608.8 172624.6 165830.9
##  [9] 160933.2 156969.4 153538.4 150309.6 148321.7 146424.0 144523.7
```





Ugly clustering over the data.

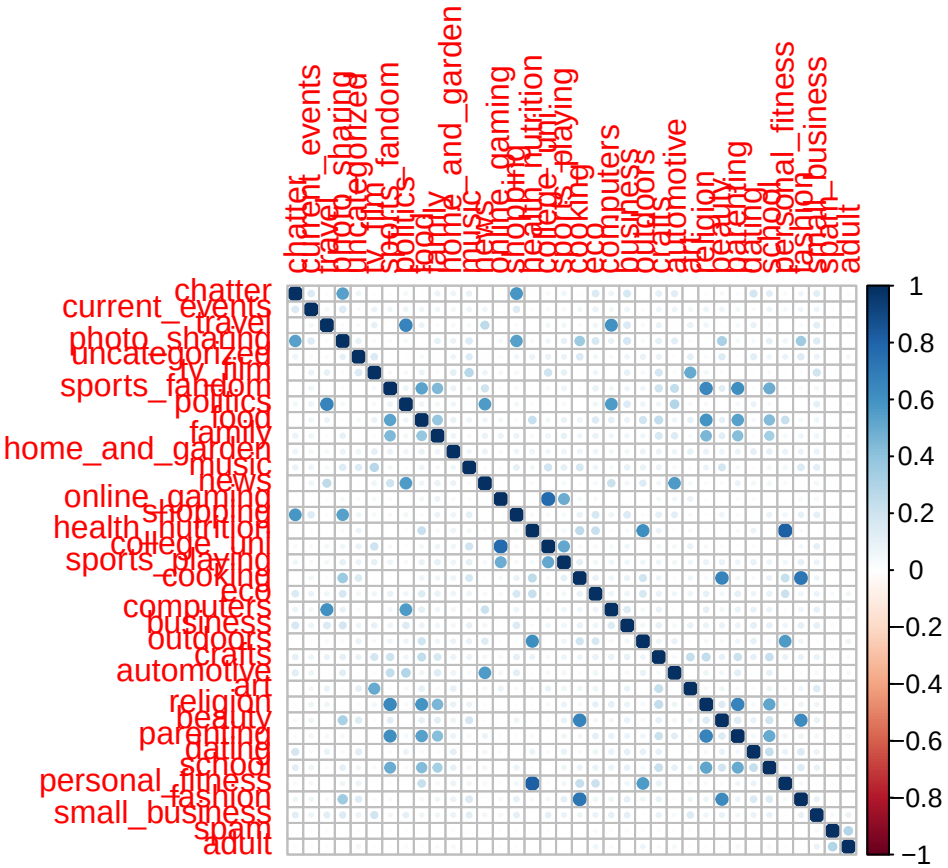




Market segmentation

K-means using PCA data

Five clusters was the most ideal after testing and five variables were removed including spam, chatter and adult.



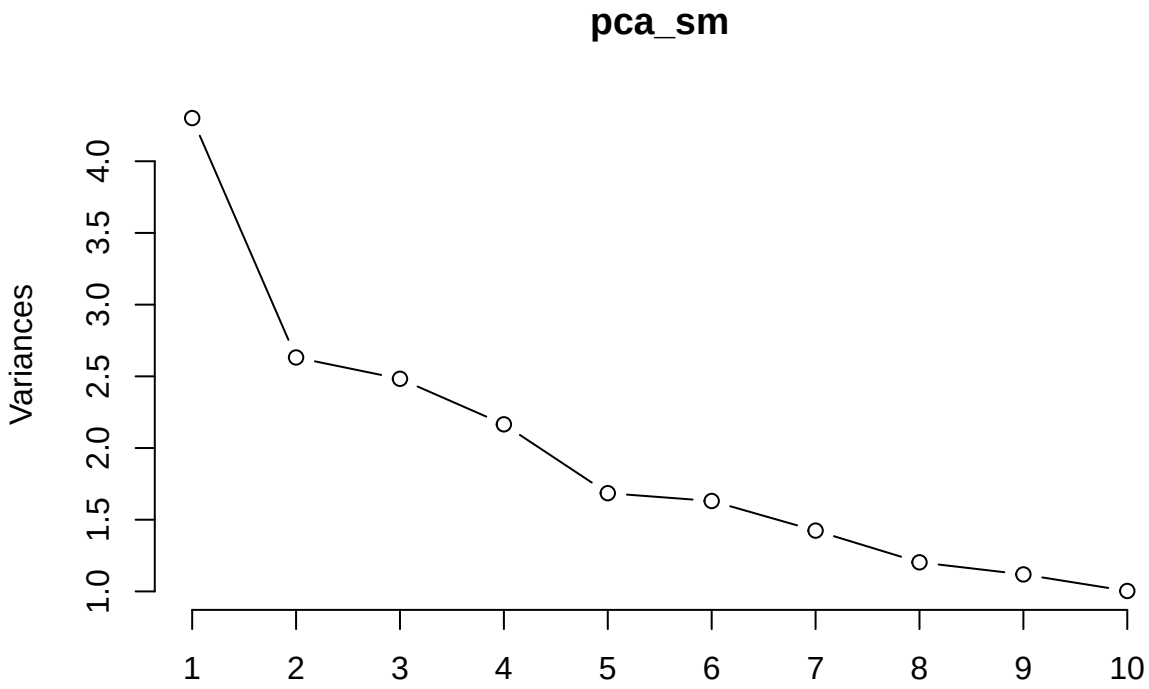
Correlation plot

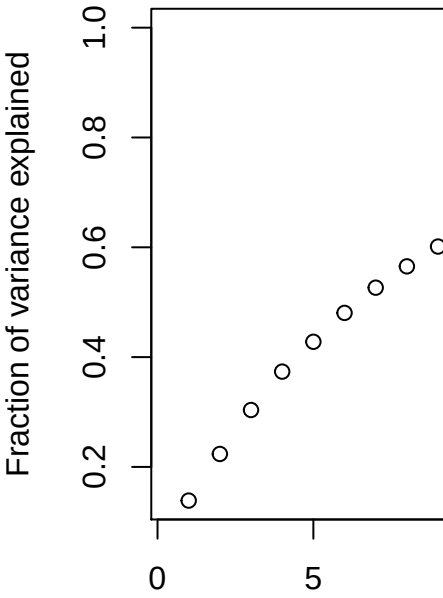
Many variables are correlated. some examples include: personal fitness and health nutrition,and online gaming and college university variables have a high correlation.

Principal Component Analysis PCA will be used to reduce the dimensions to create fewer uncorrelated variables.

```
## Importance of components:
##                           PC1     PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     2.0739 1.62224 1.57567 1.47152 1.29814 1.2770 1.19334
## Proportion of Variance 0.1387 0.08489 0.08009 0.06985 0.05436 0.0526 0.04594
## Cumulative Proportion  0.1387 0.22363 0.30372 0.37357 0.42793 0.4805 0.52647
##                           PC8     PC9    PC10    PC11    PC12    PC13    PC14
## Standard deviation     1.0968 1.05760 1.00154 0.96661 0.95607 0.93754 0.93123
## Proportion of Variance 0.0388 0.03608 0.03236 0.03014 0.02949 0.02835 0.02797
## Cumulative Proportion  0.5653 0.60136 0.63372 0.66386 0.69334 0.72170 0.74967
##                           PC15    PC16    PC17    PC18    PC19   PC20   PC21
## Standard deviation     0.90714 0.89512 0.83291 0.80770 0.75366 0.6953 0.6704
## Proportion of Variance 0.02655 0.02585 0.02238 0.02104 0.01832 0.0156 0.0145
## Cumulative Proportion  0.77621 0.80206 0.82444 0.84548 0.86381 0.8794 0.8939
##                           PC22    PC23   PC24    PC25    PC26   PC27    PC28
## Standard deviation     0.65360 0.64035 0.6323 0.61717 0.59883 0.5945 0.55315
## Proportion of Variance 0.01378 0.01323 0.0129 0.01229 0.01157 0.0114 0.00987
## Cumulative Proportion  0.90768 0.92091 0.9338 0.94609 0.95766 0.9691 0.97893
##                           PC29    PC30    PC31
## Standard deviation     0.48605 0.47625 0.43602
## Proportion of Variance 0.00762 0.00732 0.00613
```

```
## Cumulative Proportion  0.98655 0.99387 1.00000
```





Based on the Kaiser criterion, drop principal components with eigen values less than 1.0.

```
cumsum(pca_var1)[10]
```

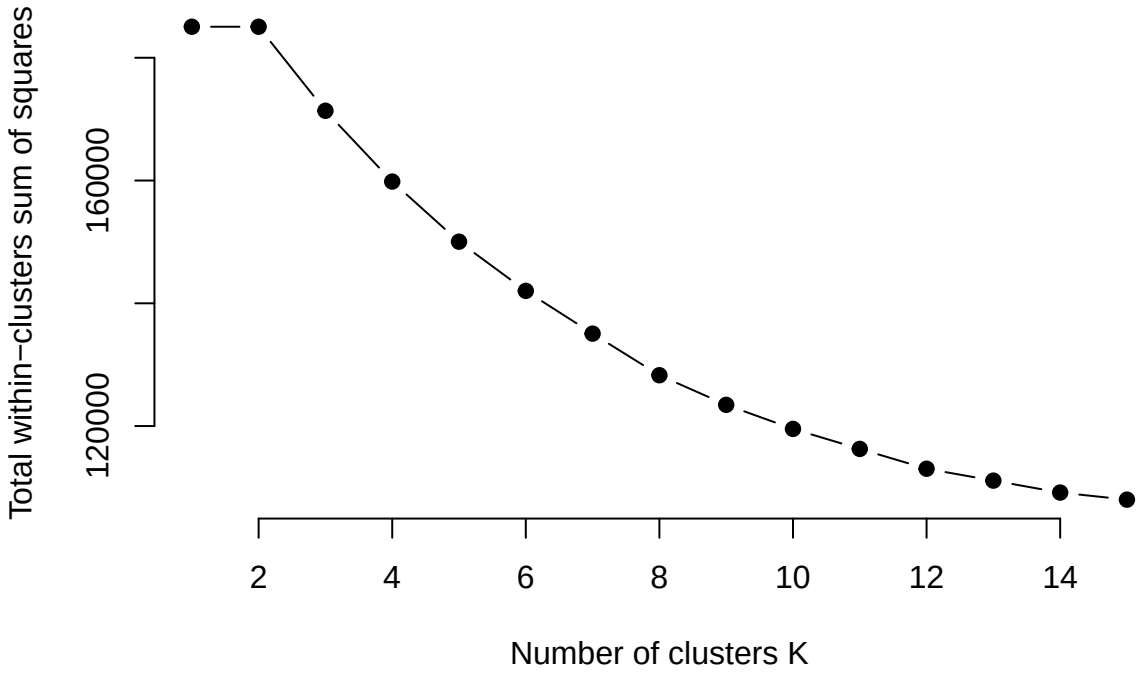
```
## [1] 0.6337156
```

63.37% of the variation is explained using 10 princial components.

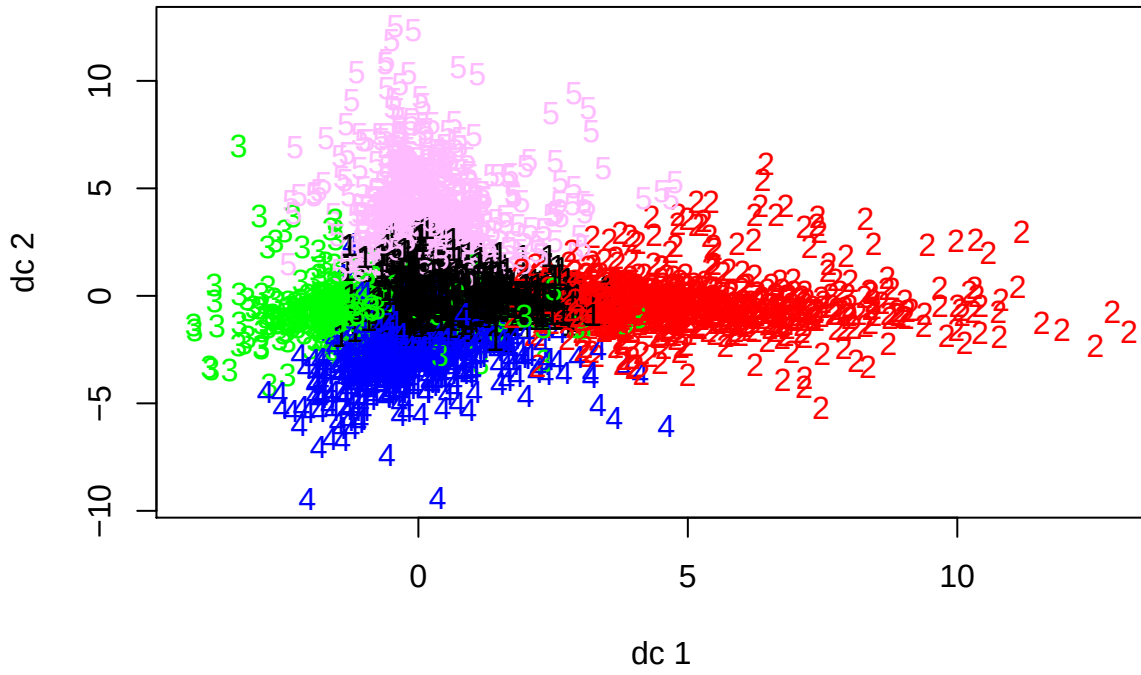
K-Means

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

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



Hard to determine the number of clusters so we are using k=5 and see how it works.

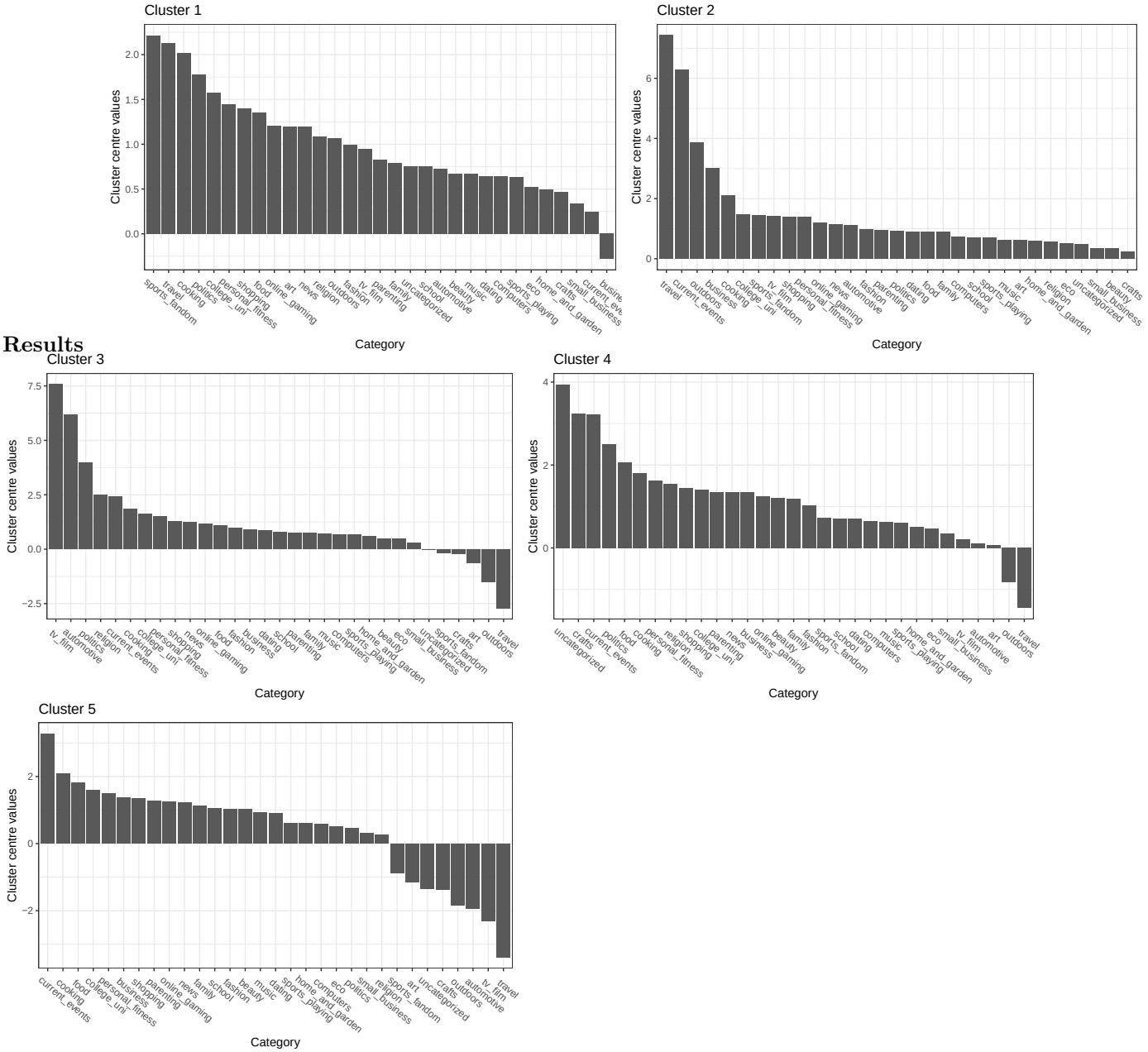


Cluster visualization

The clusters have clear boundaries.

Cluster character ID

Results



Market segments identified Using K-Means clustering, we can identify unique market segments that NutrientH20 can make use of to advertise better.

1. Sports Fandom, Travel, Cooking, politics, College Uni Cluster 1 is younger people with extroverted interests.
2. Travel, current events, outdoors, Business, cooking, college Uni Cluster 2 has people who are in college, athletic, and finance oriented.
3. TV Film, Automotive, Politics, religion Cluster 3 seems like people who are older, more conservative.
4. Crafts, current events, politics, food, cooking Cluster 4 - seems like people who are into artistic stuff.
5. Current Events, cooking, food, college uni Cluster 5 - college age people who stay indoors but want to maintain healthy intakes.

Author Attribution

Analysis

Our team ran into difficulties getting our code to work to create our model. Instead, outlined is our method for our workflow that we intended to execute to create our model that predicts the author of an article on the basis of that article's textual content:

1. Read in relevant data and libraries
2. Clean up file names for easier processing
3. Create a text mining corpus
4. Use the tm_map library for pre-processing and tokenization. This includes converting to lowercase, removing numbers, removing punctuation, removing excess white-space, and removing stopwords.
5. Create a doc-term-matrix from the corpus, then remove sparse terms. We could also construct tf_idf weights if we wanted to use these as features in a predictive model.
6. For dimensionality reduction, use PCA. This serves to extract relevant features from the large corpus and eliminate multicollinearity while not missing out on relevant information from the variables.
7. For classification, we would have tried a random forest, logistic regression, and KNN. Because this is a classification problem, we would use accuracy to measure the effectiveness of our models.

Summary

Our team struggled with step 4 outlined above. We could not get the data to come out correctly, and this meant that future steps could not be done. From this, we learned that data cleaning skills are an important and valuable part of a data scientist's toolbox. Without sufficient technical ability and/or ability to troubleshoot, abilities such as analysis and modeling become obsolete.

Association rule mining

```
## [1] "citrus fruit,semi-finished bread,margarine,ready soups"
## [2] "tropical fruit,yogurt,coffee"
## [3] "whole milk"
## [4] "pip fruit,yogurt,cream cheese ,meat spreads"
## [5] "other vegetables,whole milk,condensed milk,long life bakery product"
## [6] "whole milk,butter,yogurt,rice,abrasive cleaner"
```

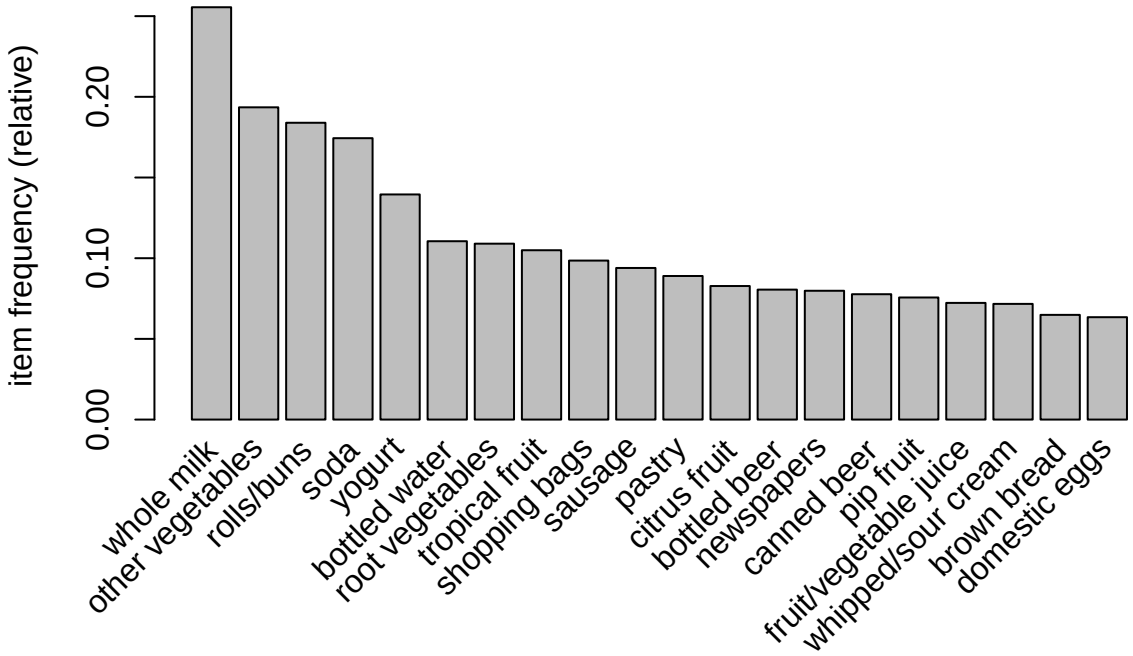
Processing, exploratory analysis

Transform data into 'transactions' class

```
## transactions as itemMatrix in sparse format with
##  9835 rows (elements/itemsets/transactions) and
##  169 columns (items) and a density of 0.02609146
##
## 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    5    6    7    8    9   10   11   12   13   14   15   16
## 2159 1643 1299 1005  855  645  545  438  350  246  182  117   78   77   55   46
##   17   18   19   20   21   22   23   24   26   27   28   29   32
##   29   14   14    9   11    4    6    1    1    1    1    3    1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   4.409   6.000  32.000
##
## includes extended item information - examples:
##             labels
## 1 abrasive cleaner
## 2 artif. sweetener
## 3   baby cosmetics
```

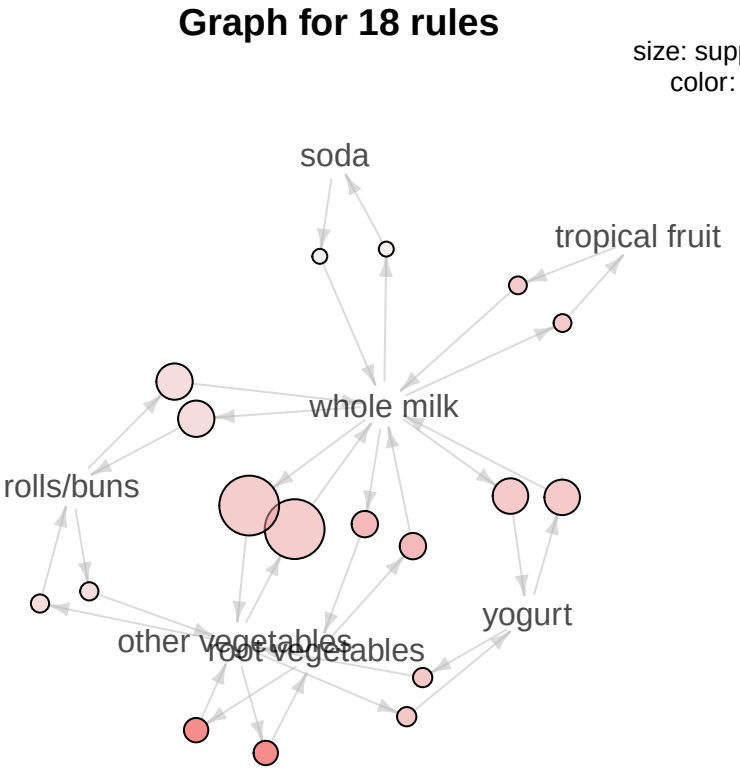
Translation of summary results: There are 9835 transactions in our dataset. Whole milk, vegetables, buns, soda, and yogurt are the most frequently bought items. Based on the median, half of the transactions contain 3 or less items and those items are more than likely going to be milk, vegetables, buns, or soda.



Networks support > 0.04, confidence > 0.1 and length $<= 2$

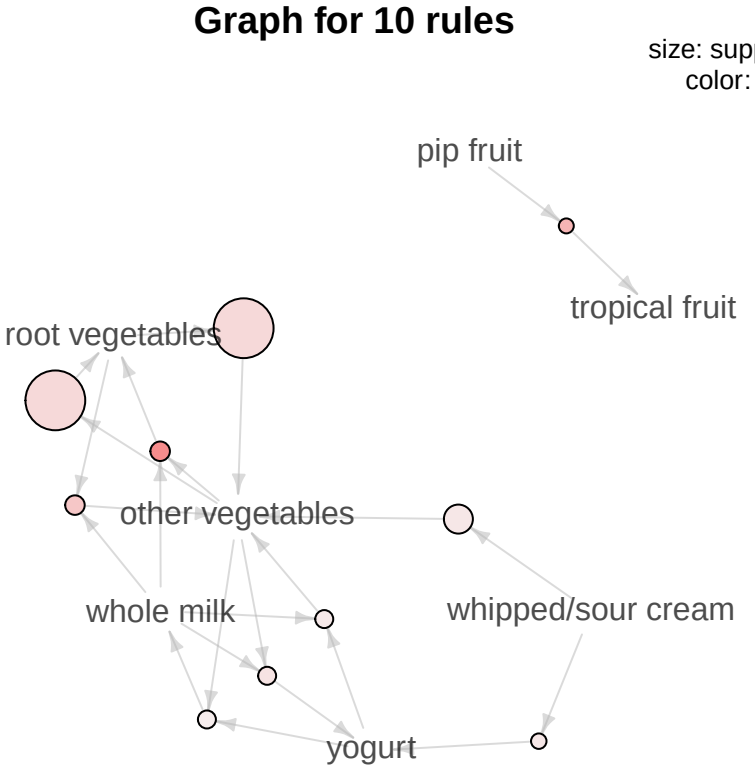
```
##      lhs                   rhs                support    confidence coverage
## [1]  {tropical fruit}   => {whole milk}       0.04229792 0.4031008  0.1049314
## [2]  {whole milk}       => {tropical fruit}   0.04229792 0.1655392  0.2555160
## [3]  {root vegetables}  => {other vegetables} 0.04738180 0.4347015  0.1089985
## [4]  {other vegetables} => {root vegetables}  0.04738180 0.2448765  0.1934926
## [5]  {root vegetables}  => {whole milk}       0.04890696 0.4486940  0.1089985
## [6]  {whole milk}       => {root vegetables}  0.04890696 0.1914047  0.2555160
## [7]  {soda}             => {whole milk}       0.04006101 0.2297376  0.1743772
## [8]  {whole milk}       => {soda}             0.04006101 0.1567847  0.2555160
## [9]  {yogurt}           => {other vegetables} 0.04341637 0.3112245  0.1395018
## [10] {other vegetables} => {yogurt}           0.04341637 0.2243826  0.1934926
## [11] {yogurt}           => {whole milk}       0.05602440 0.4016035  0.1395018
## [12] {whole milk}       => {yogurt}           0.05602440 0.2192598  0.2555160
```

```
## [13] {rolls/buns}       => {other vegetables} 0.04260295 0.2316197  0.1839349
## [14] {other vegetables} => {rolls/buns}       0.04260295 0.2201787  0.1934926
## [15] {rolls/buns}       => {whole milk}       0.05663447 0.3079049  0.1839349
## [16] {whole milk}       => {rolls/buns}       0.05663447 0.2216474  0.2555160
## [17] {other vegetables} => {whole milk}       0.07483477 0.3867578  0.1934926
## [18] {whole milk}       => {other vegetables} 0.07483477 0.2928770  0.2555160
##      lift      count
## [1]  1.5775950 416
## [2]  1.5775950 416
## [3]  2.2466049 466
## [4]  2.2466049 466
## [5]  1.7560310 481
## [6]  1.7560310 481
## [7]  0.8991124 394
## [8]  0.8991124 394
## [9]  1.6084566 427
## [10] 1.6084566 427
## [11] 1.5717351 551
## [12] 1.5717351 551
## [13] 1.1970465 419
## [14] 1.1970465 419
## [15] 1.2050318 557
## [16] 1.2050318 557
## [17] 1.5136341 736
## [18] 1.5136341 736
```



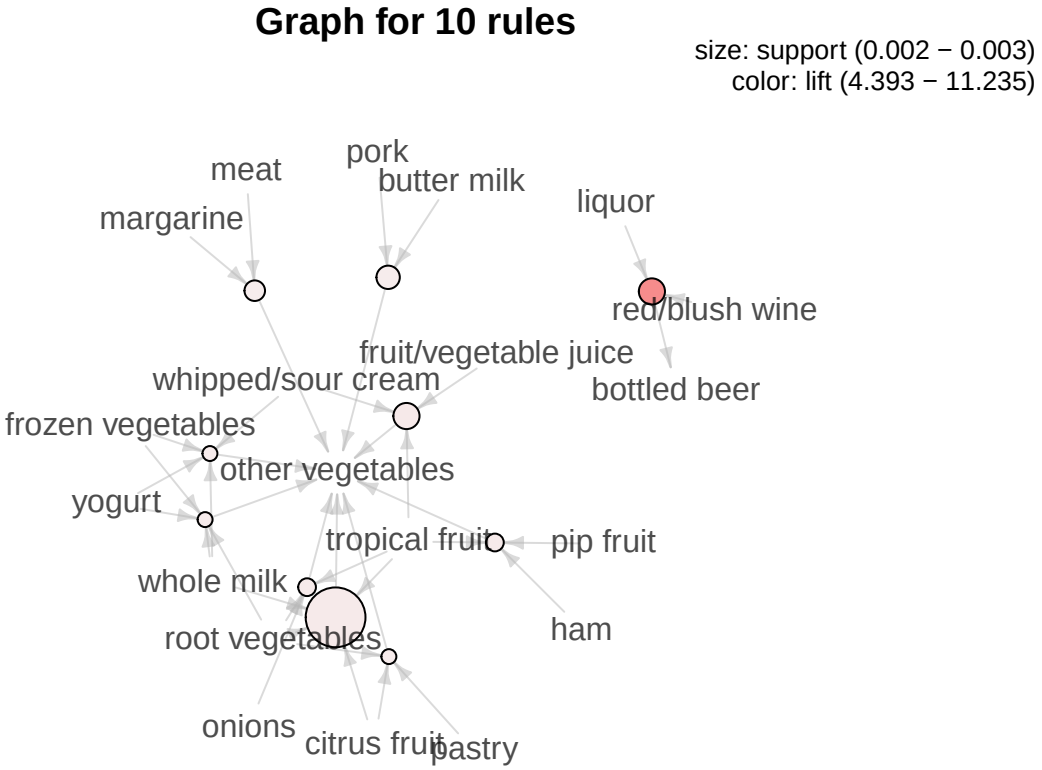
here are 18 rules generated. Those numbers were used by trial and error to get results we could look at visually and interpret. min length 2 is ideal or we would have a monstrous plot with hundreds of rules. It is obvious most relationships in this item set include whole milk, yogurt, buns, and soda which matches what was predicted earlier when looking at the summary results and the item frequency plot.

Decrease support & increase confidence network with support > 0.02, confidence > 0.2 and length <= 2



72 rules were generated and includes a lot more items. A graph with 10 rules was displayed as it was the easiest to look at as there are lot more paths than before due loosening the conditions. Whole milk and vegetables are still the most common.

Decrease the support & increase confidence level again network with support > 0.0015, confidence > 0.8 and length <= 2



60 rules were generated with 153 items with way more paths than before, one of the reasons the graph with 10 rules displayed was chosen. Vegetables and whole milk are still top contenders.

Summary Given that the last network had the roadest parameters, it gave us the most information. Looking at the association rules we can infer: People are more likely to buy bottled beer if they purchased red wine or liquor with confidence of 0.9. If people buy flour, root vegetables, and whipped/sour cream they will buy whole milk with confidence of 1! They should probably put some type of stand with those items next to the dairy section at HEB. lol Whole milk and vegetables are the most common items purchased by customers in each network explored.