Intro to ML Final Exam 2

Abhinav Sharma

1/8/2021

Raw .Rmd code available at —– https://github.com/abhinav-sharma-6167/Intro-to-ML-2

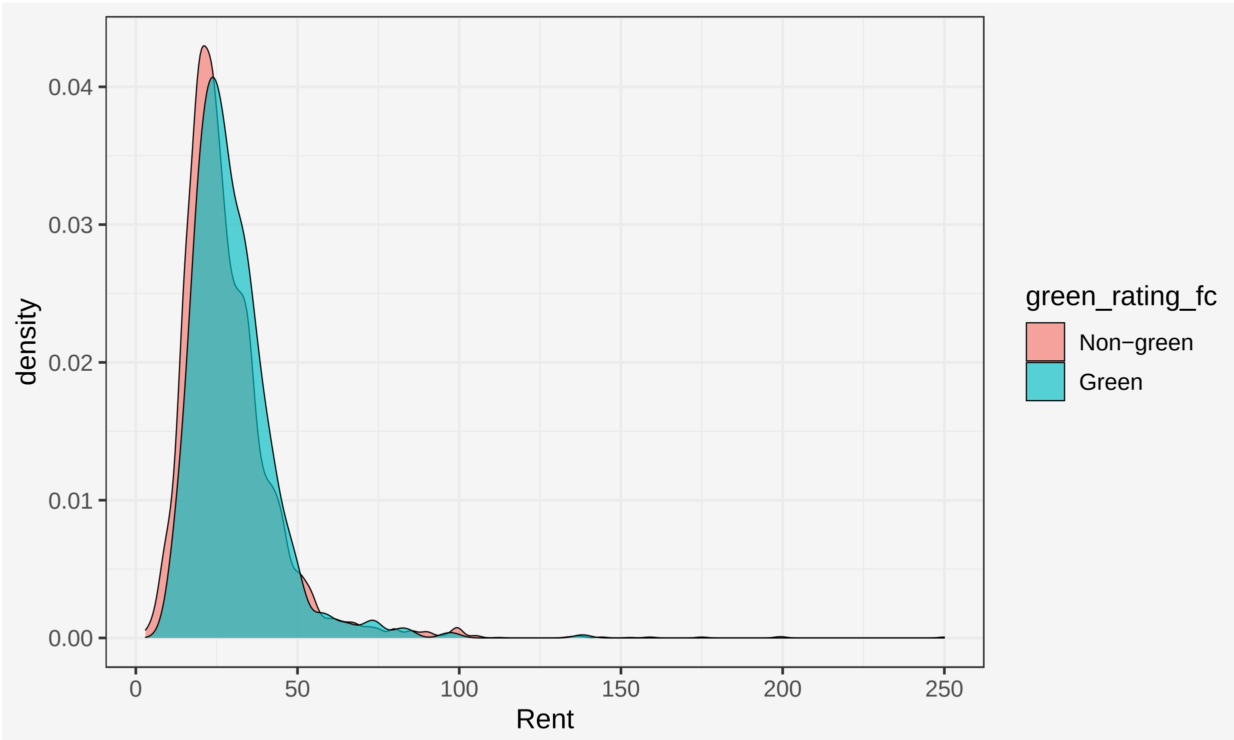
Setup - Loading libraries and setting working directory. Key libraries include ggplot2, plotly for exploration; data.table and dplyr data wrangling. Setting up custom ggplot theme for all plotting purposes.

1. Visual story telling part 1: green buildings

The details given in the case helps us get a sense of the problem. To quickly summarize, the data encompasses of 7,894 commercial rental properties of which 685 are green buildings.To avail a control set for the 685 properties, the creators of this data leveraged all non-rated buildings within a quarter-mile radius of the green-certified building. On average we have 12 non-rated nearby properties for each green property. The idea is green houses would be more attractive living options given lower recurring costs, better indoor environments, longer economically valuable lives and in general, the good PR they enjoy. The goal is to validate whether investing in a green building be worth it, from an economic perspective. Specifically, in a new 15-story mixed-use building on East Cesar Chavez, just across I-35 from downtown with baseline construction costs being $100 million and a 5% expected premium for green certification.

Building upon this premise, we now start exploring the data. We start by checking the number of green houses and distributions of the variables used in stat-guru's analysis to stress-test the assumptions made.

```
Percentage of green buildings :
 8.68 %
```

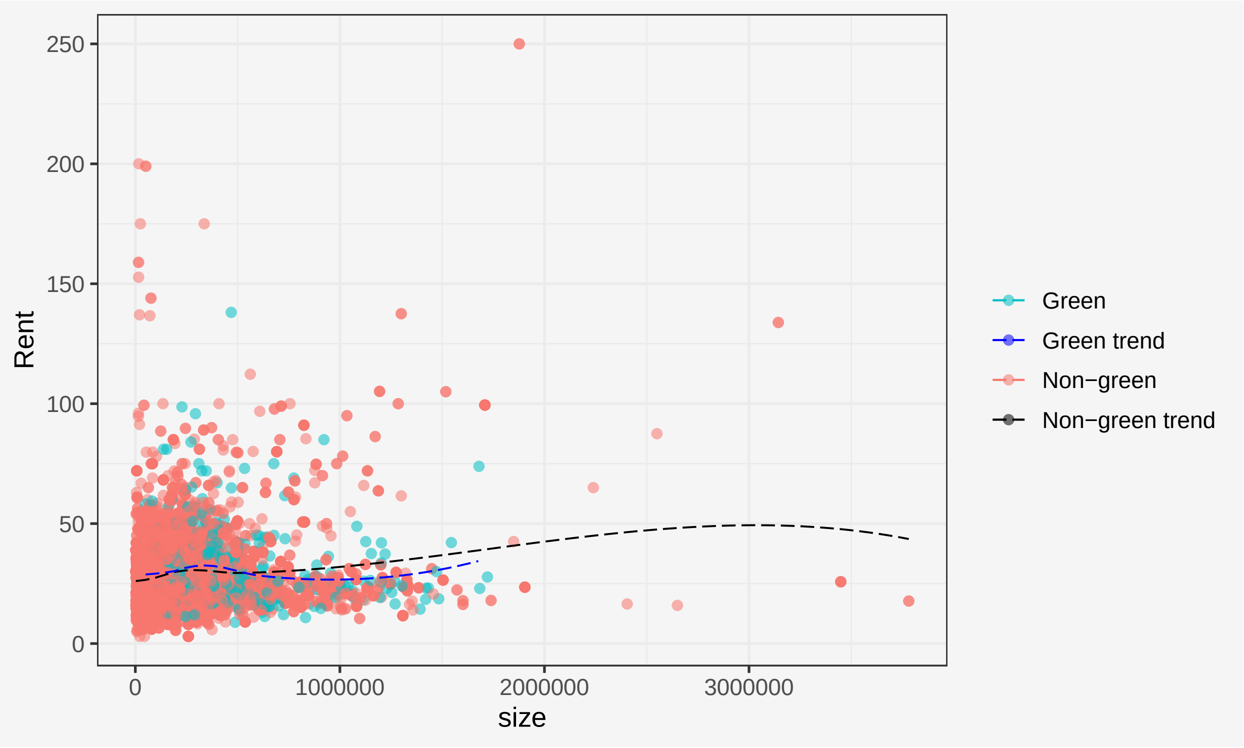



```
Summary Stats :

   green_rating Mean_Rent Med_Rent SD_Rent IQR_Rent
1:            1     30.03    27.60   12.96    14.04
2:            0     28.44    25.03   15.33    14.75
```

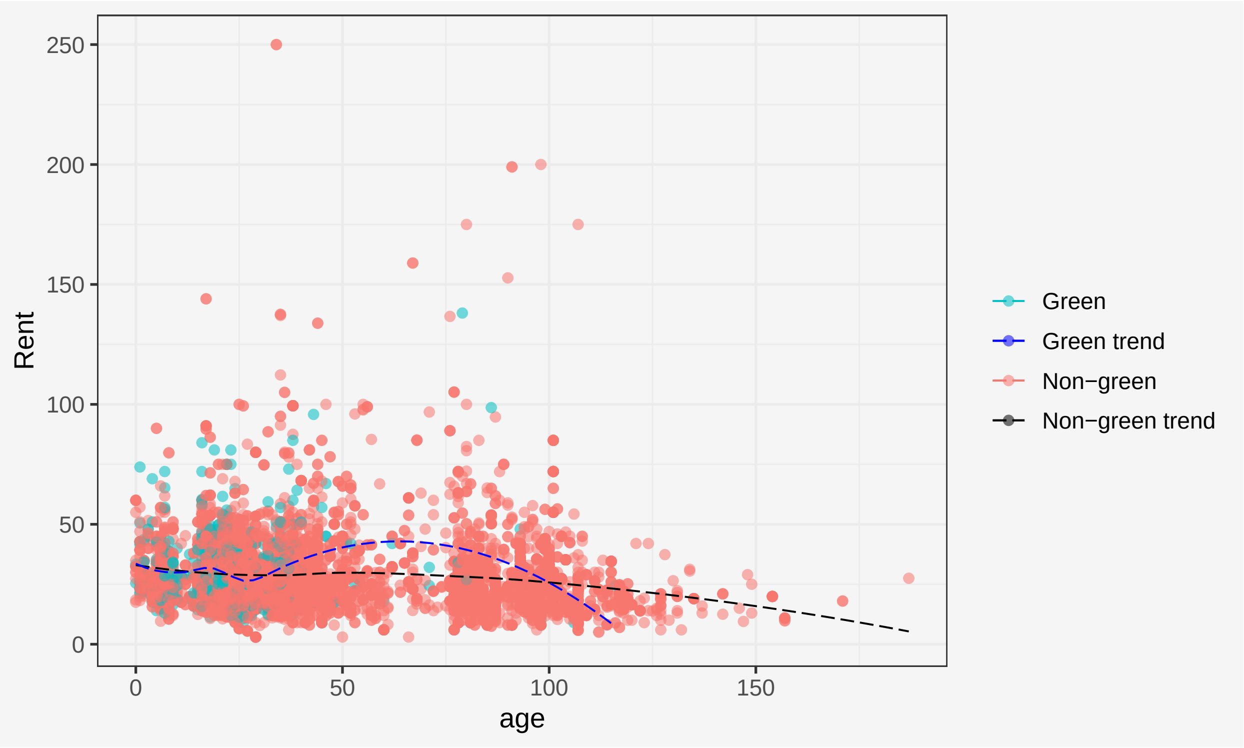
We see Rent for green houses is slightly larger than other houses. The rent variable is extremely right skewed so it makes sense to use median as the measure for centrality instead of mean. The statistical summary suggests Rent distribution is slightly more spread out for non-green buildings given it has higher SD and IQR. However, the graph looks pretty much the same for both categories implying, non-green building Rent has more outliers compared to green buildings.

However, this difference in medians could arise due to multiple confounding variables. Few hypotheses for this price change could be : 1. Properties with higher size may have higher rent 2. Older houses might have lower rent 3. Renovated houses may have higher rent 4. Houses in better class society and with more amenities may have higher rent 5. Houses with more Gas and Electricity costs may have higher rent, given they have more high energy-consuming facilities 6. Only specific type of green buildings (say LEED) may offer difference in rent

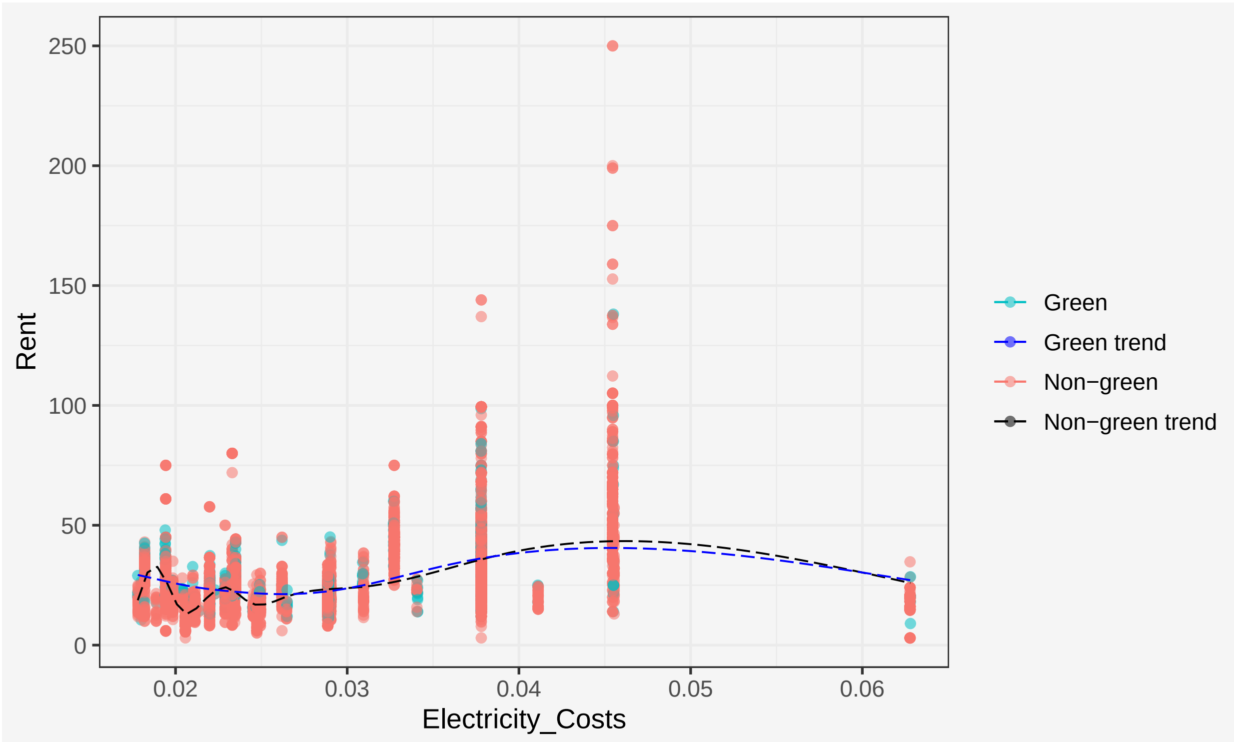
Let's explore them one by one. We use the non-parametric Generalized Additive Model (GAM) based smoothening to capture the non-linearity in the trends in our scatter plots.

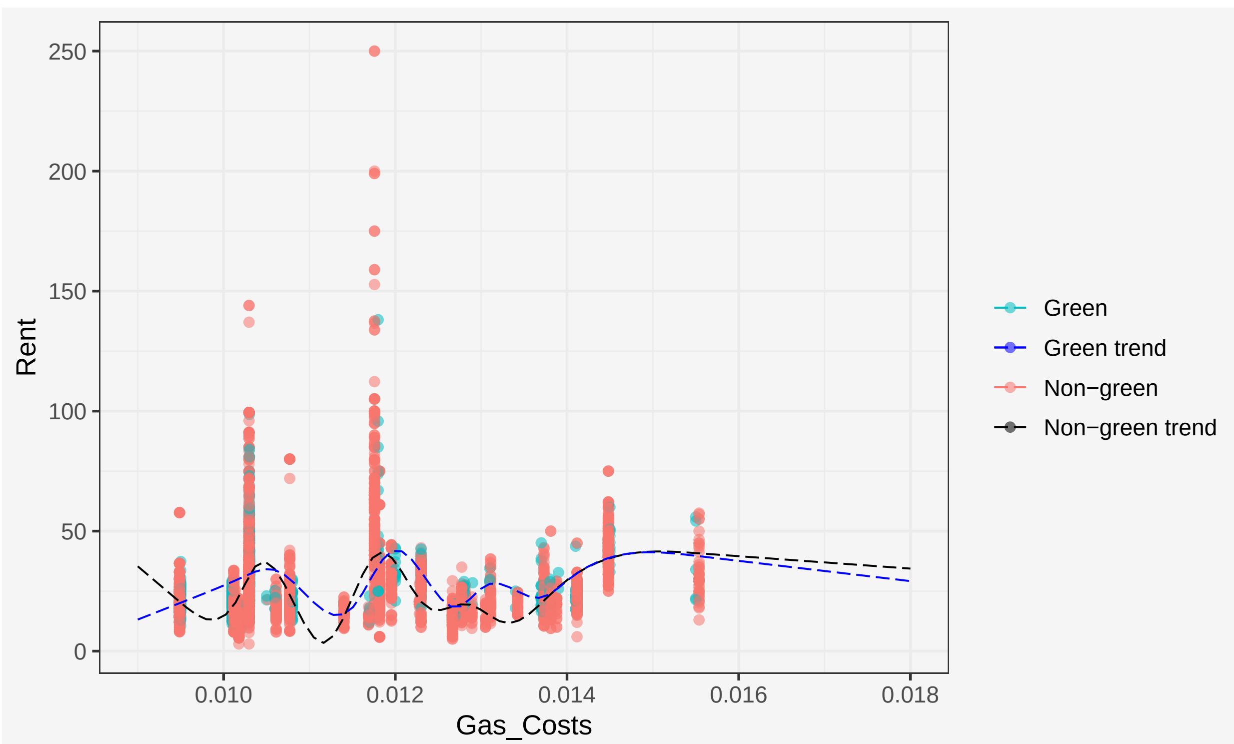


We observe almost no difference in Rent for green buildings vs other uptil size 900000, post which rent of non-green buildings slightly increases. However, this difference could be due to more outliers for non-green category and the confidence intervals of the fitted curves would intersect, negating any significance associated with this difference.



We see for houses aged 40-90 have some differenciation in rent with respect to houses being green vs non-green

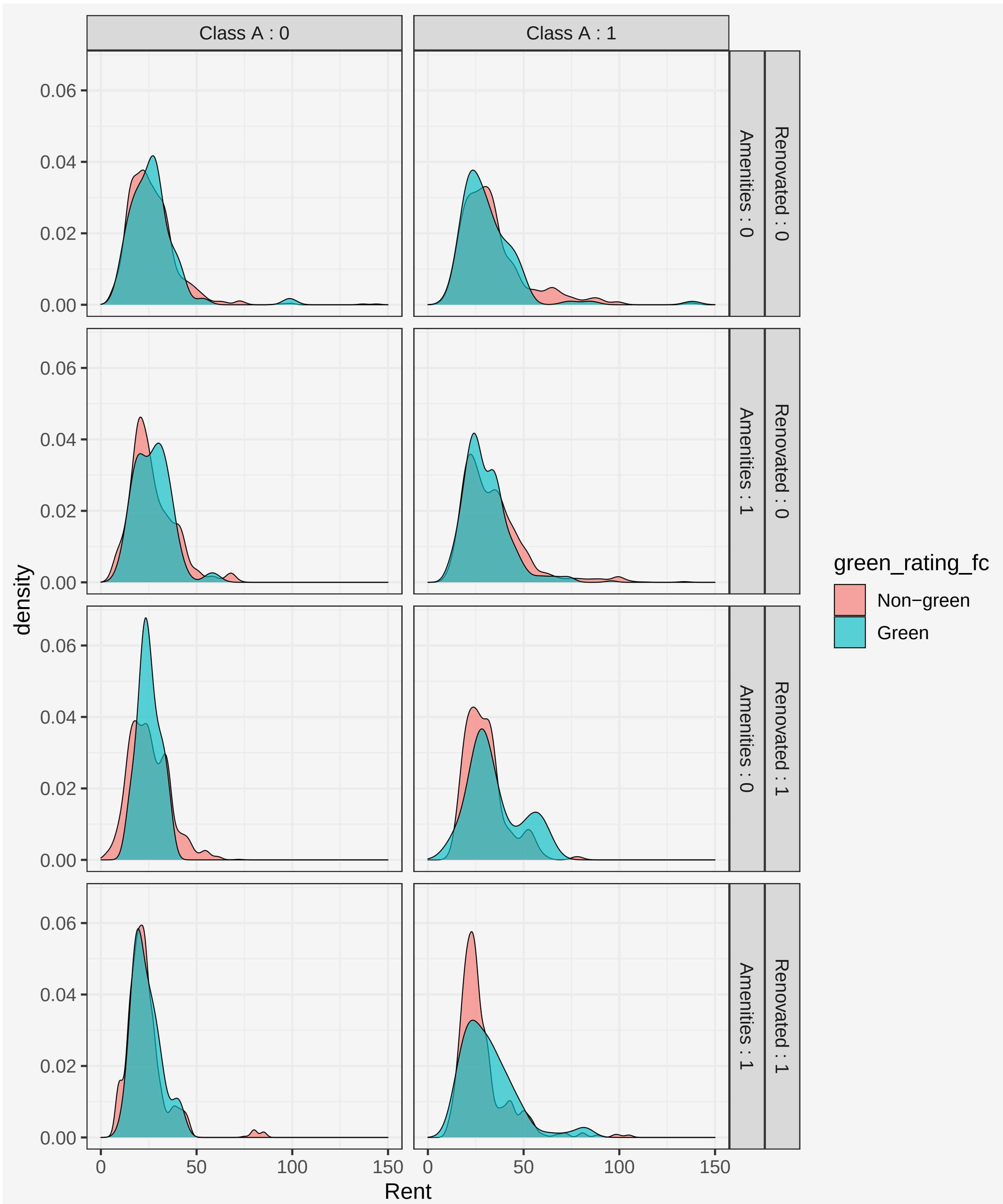




There's hardly any difference in Gas and electricity utilization, apart from a small segment of Gas costs between 0.011-0.012 where green buildings have lesser gas costs.

Even with continuous variables, we get the sense that the difference between prices of green building vs not wouldn't be as straightforwards as difference of median rents of the two categories. The localized regression trendlines convey that for certain properties of a house, some of the price difference could be explained by confounding variables such as age, gas costs.

We also explore categorical variables and check whether different categories lead to change in Rent.



Rent of house being green vs non-green differs widely in distribution, specifically in cases where house is renovated, is in upper class society and has additional amenities. Given, East Cesar Chavez looks very developed and say, the house has undergone renovations and includes amenities, the stat-guru's assumption of all green houses having incremental rent of $2.6 than their non-green houses is erroneous.

```
The median price difference in such a case would be : $ 0.75
```

This implies that the duration of cost recuperation would be higher than 7.7 years. Assuming, the real difference in medians upon accounting for all confounding variables was $ 0.75, the premium of investment in a green building would be recovered in about 26.67 years. Hence, availing green status to ensure more probitability might not be a good strategy.

Because the proportion of green buildings in very less, thereby green vs non-green buildings have unequal sample sizes. We can use Anova to have a final comparison of means of unequal samples and remove Rent higher than 100 as outlier. We can include covariates we've explored in the EDA above such as recurring costs, size and age of building, class and amenities to quantify their effect sizes.

```
                              Feature      Coef    F_stat  P_val
 1: green_rating_fc                      6.5335   10.1345 0.0015
 2: size                                -0.0172  183.8325 0.0000
 3: age                                  0.0000   56.7926 0.0000
 4: renovated                            0.0353   99.2252 0.0000
 5: class_a                             -2.9395  234.2568 0.0000
 6: class_b                              6.9193   39.3476 0.0000
 7: amenities                            2.7857   14.0708 0.0002
 8: Gas_Costs                            0.6698    5.0544 0.0246
 9: Electricity_Costs                 -692.6059 1612.2222 0.0000
10: empl_gr                            756.3225    6.2707 0.0123
11: green_rating_fc:size                 0.0354    4.5498 0.0330
12: green_rating_fc:age                  0.0000    2.1967 0.1384
13: green_rating_fc:renovated            0.0493    1.8038 0.1793
14: green_rating_fc:class_a              1.9534    2.0126 0.1560
15: green_rating_fc:class_b             -0.2348    0.2521 0.6156
16: green_rating_fc:amenities           -2.7601    1.0341 0.3092
17: green_rating_fc:Gas_Costs           -1.1879    0.0308 0.8607
18: green_rating_fc:Electricity_Costs   40.3885    0.4524 0.5012
19: green_rating_fc:empl_gr            -30.2841    2.5275 0.1119
20: Residuals                            0.0924        NA     NA
```

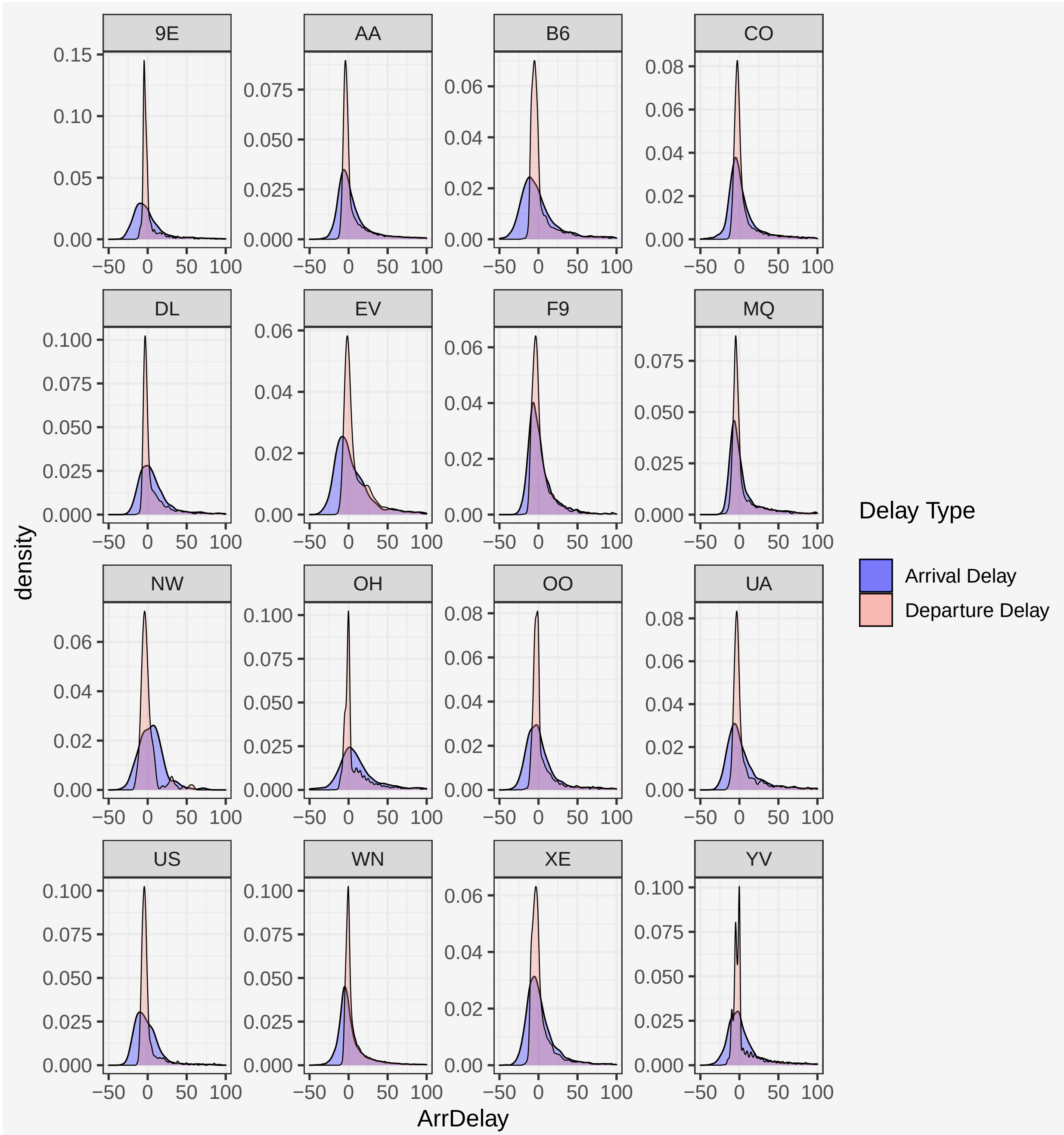
Supporting our EDA, we see all covariates have an impact on the rent of the building and solely attributing the rent to green status by differencing medians would be incorrect.

2. Visual story telling part 2: flights at ABIA

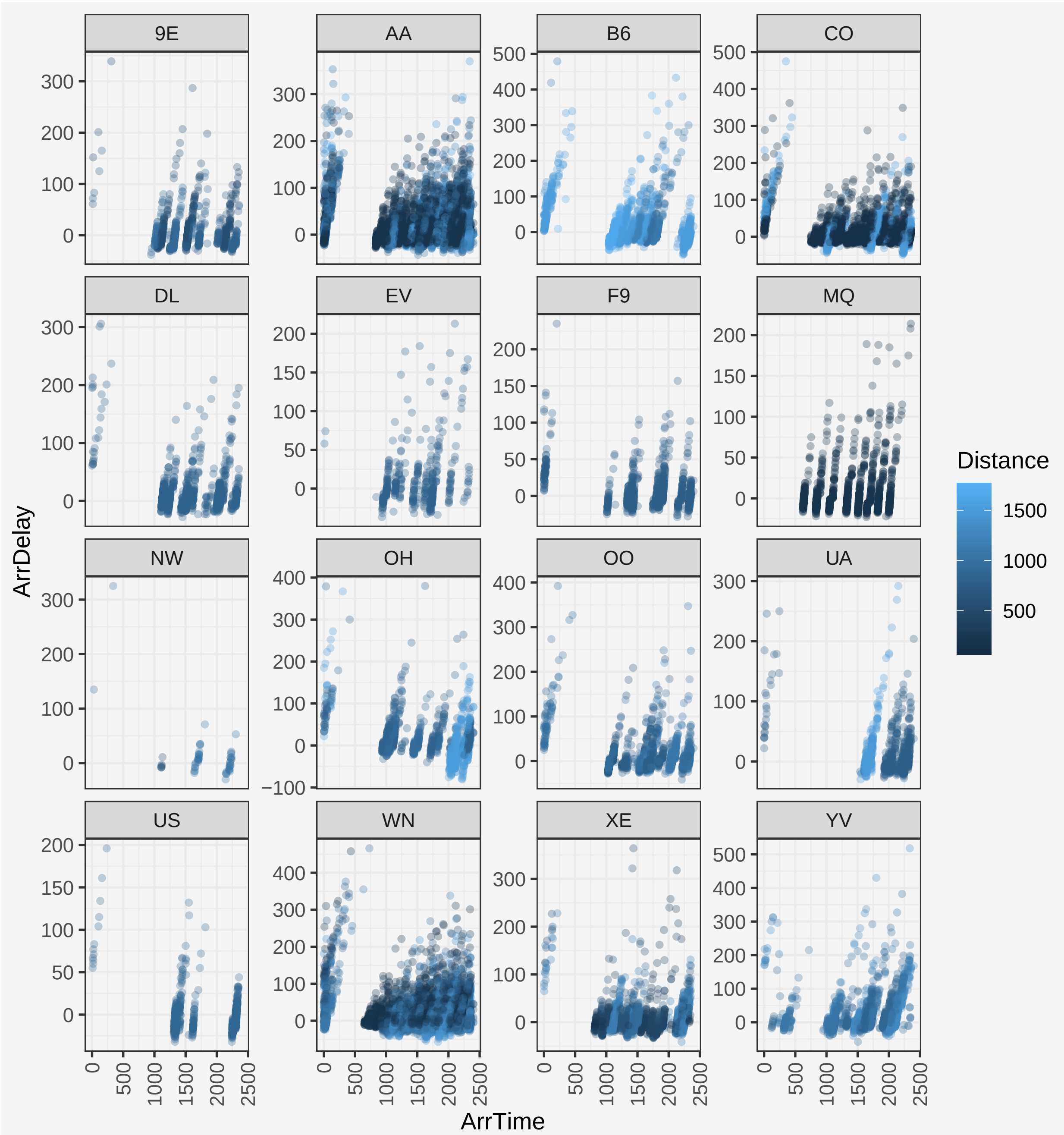
We start by exploring the data using summary functions such as str, summary, Hmisc::describe() and DescTools::Desc() to get a sense of the overall data.

Preliminary univariate EDA shows : 1. DepTime and ArrTime have 1.4% and 1.6% missing values respectively. Variables CarrierDelay, WeatherDelay, NASDelay, SecurityDelay and LateAircraftDelay all have 80.1% missing values. 2. There is some seasonality with respect to month with lesser flights towards year end in months Sept-Dec'08. Similarly there are fewer flights on weekends compared to weekdays. 3. Total 16 unique airline carriers with top three carriers operating most flights being Southwest (WN), American Airlines (AA) and Continental (CO) 4. ActualElapsedTime has bimodal distribution. ArrDelay, DepDelay distribution has very high kurtosis implying high number of outliers. Distance follows a spread-out distributions with 3 modes, meaning there could be some categorization such as local, national and international flights. Variables CarrierDelay, WeatherDelay, NASDelay, SecurityDelay and LateAircraftDelay 5. Most incoming flights to and from Austin are connected with the cities DAL, DFW, IAH, PHX etc.

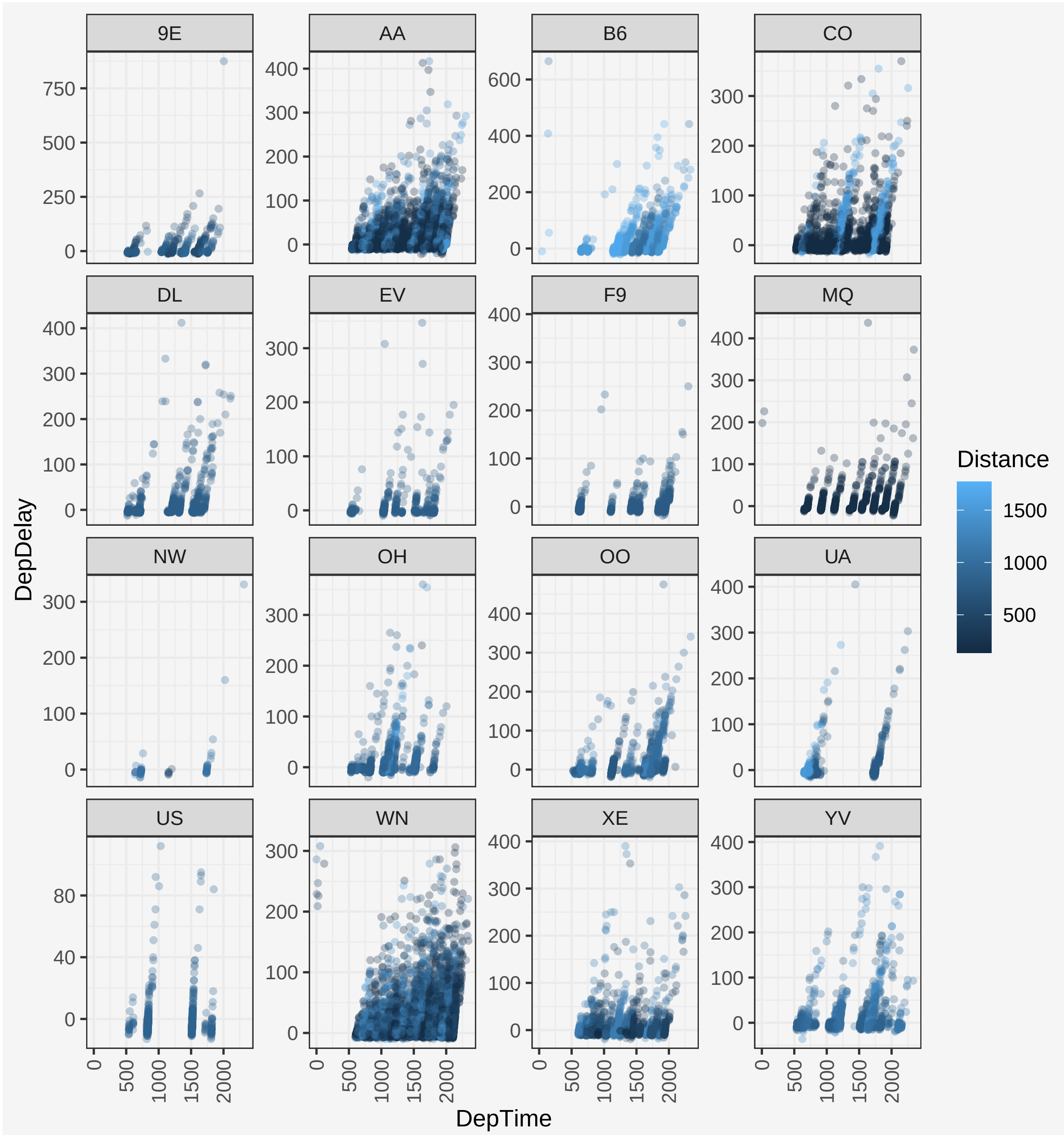
We start by studying if there is any pattern in Arrival and Departure delay by airline carriers.



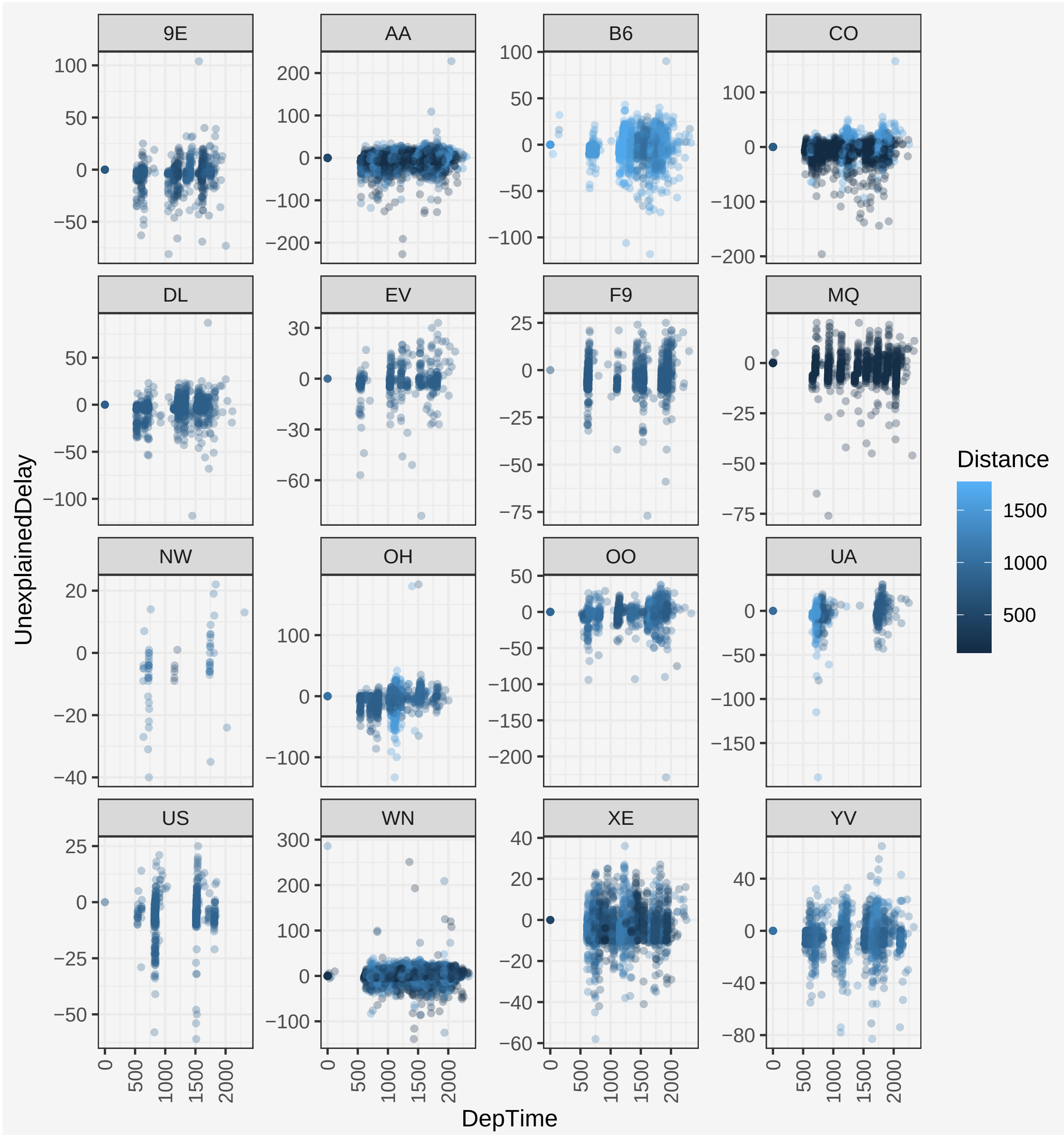
We observe that Departure delays for most carriers are often less in magnitude (be it positive or negative) than arrival delays, with an exception being F9 carrier where arrival vs departure delays are more of less the same.



Above graph shows Arrival Delays for flights arriving in Austin are particularly high when there are multiple flights of the same carrier arriving with very little time between each other.This could probably be explained by the air-space congestion issue. We also see marginally more delays when flights are for longer distances. Carriers YV and B6 have higher delays associated with them.



We avail the same insight for Departure delays being more in number and magnitude when time between multiple flights lined one after another is very less. Additionally, we check for delays that are unexplained by the parameters to validate if there are any patterns therein as well.



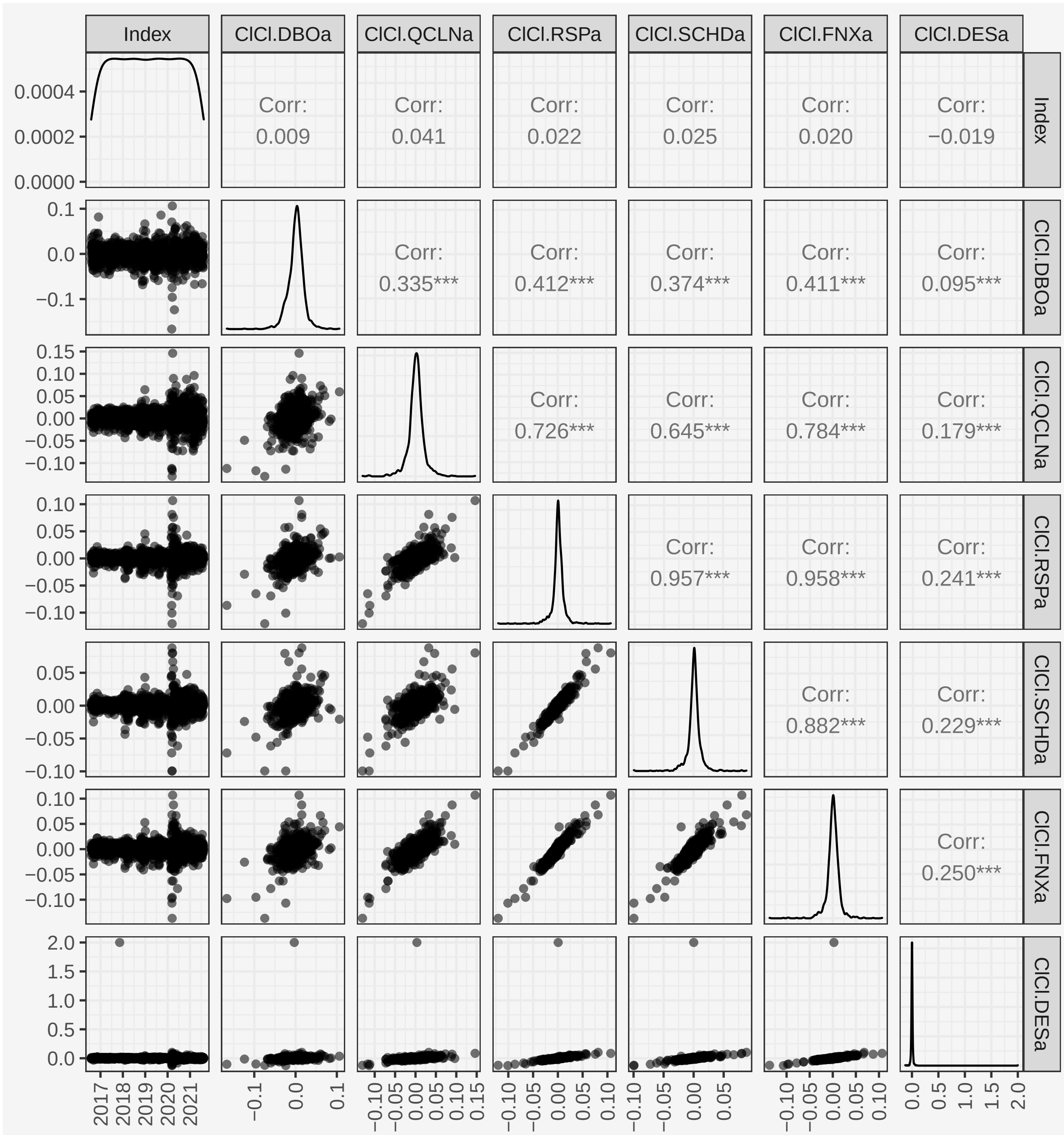
We see fewer patterns in Unexplained Delay but still flights right after one another seem to be prone to delays, specially in carriers such as MQ, WN etc.

3. Portfolio Modeling

Portfolio includes 6 ETF's which has a mix of two large cap blend equities, a mid cap blend equity, a small cap blend equity, oil/gas commodity ETF and alternative energy equities. Blend equity ETF's are a mixture of growth as well as value equities, thereby ensuring an overall diverse portfolio to reduce risk associated with the investment.

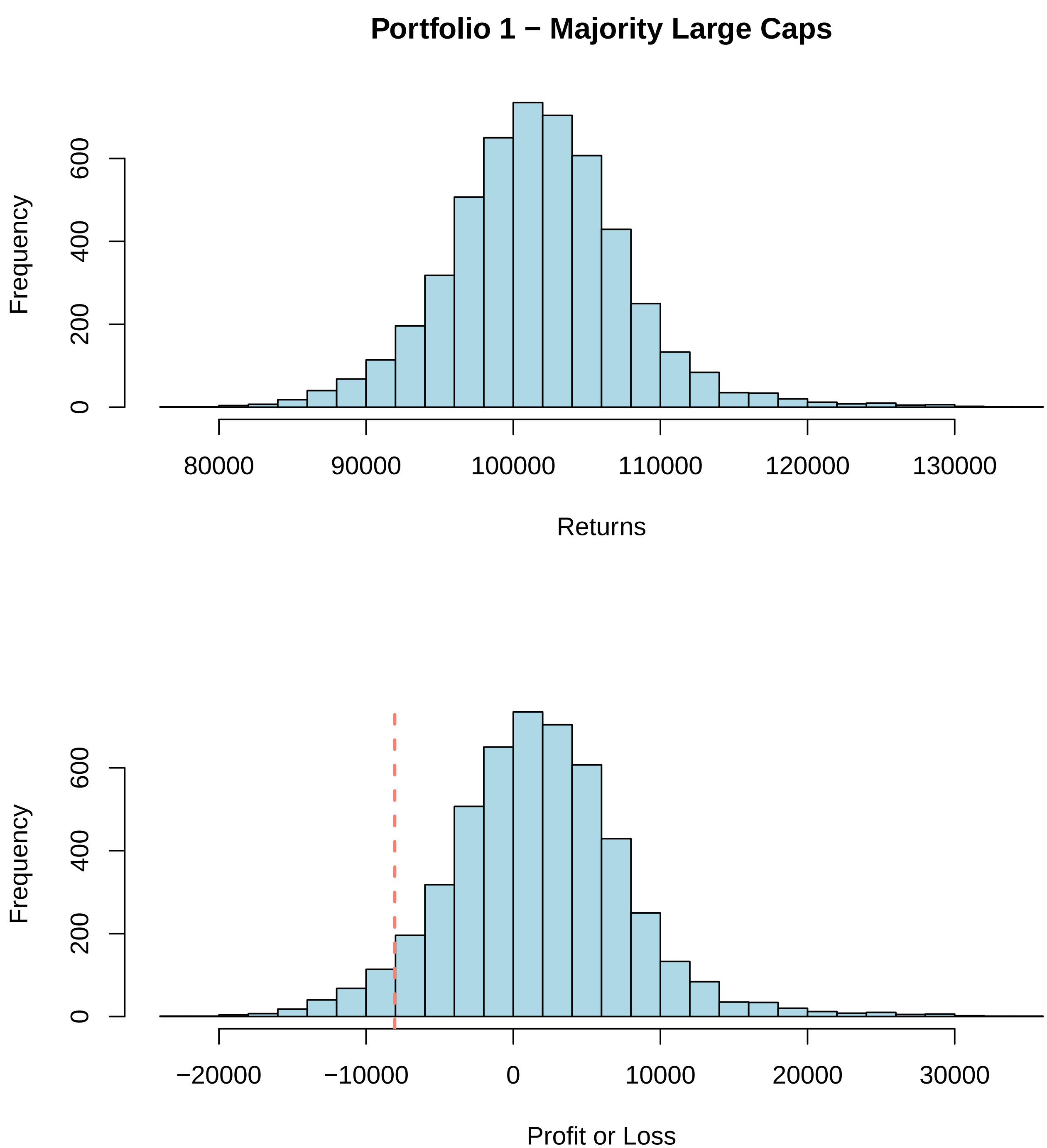
The stocks imported are energy, oil or gas commodities, large cap ETFs distributed between two securities, small and mid-cap respectively.

As a quick check, we validate whether our ETFs have any correlation with each other. Given all ETFs somewhat follow market dynamics, some correlation is expected. However, the FNX ticker seems to have returns correlated with both the large cap securities.




```
Mean :  101683.2
```

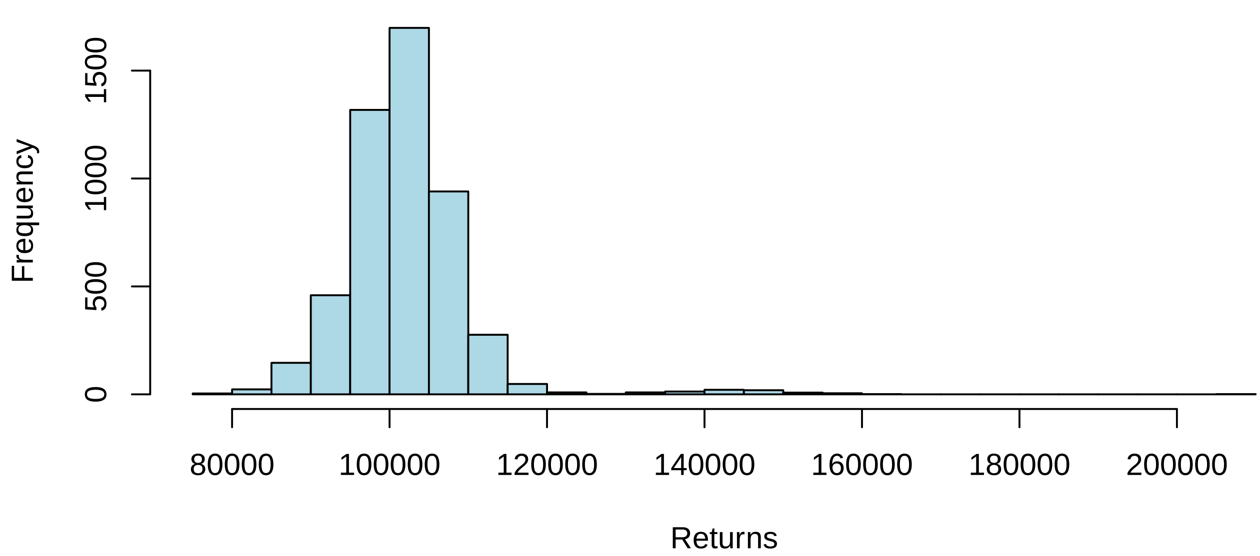
```
Average Profit :  1683.177
```

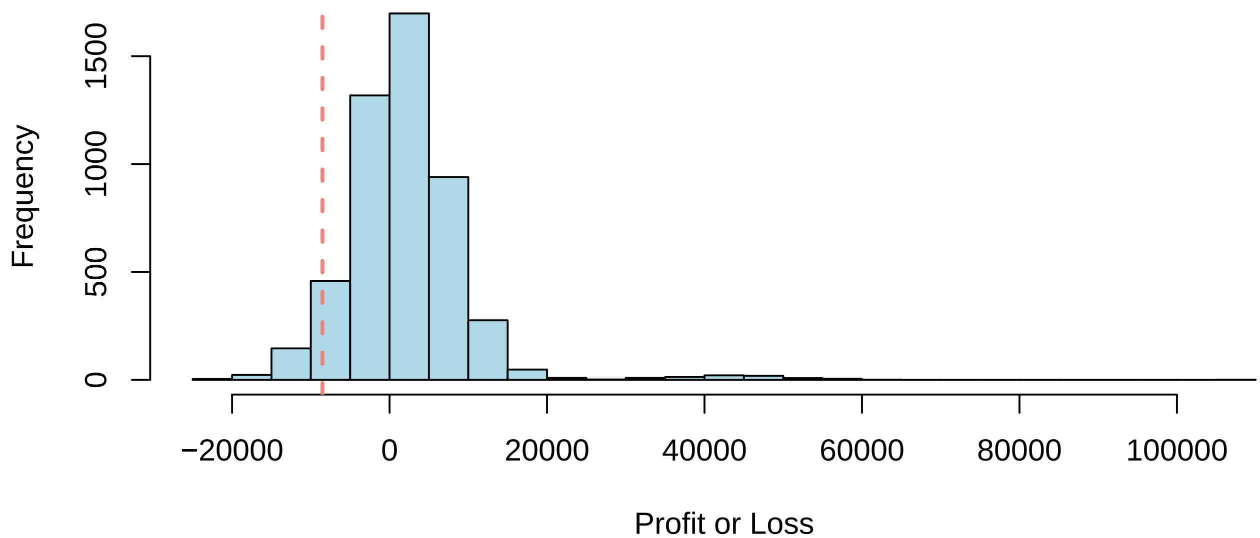



```
5% Value at Risk :  -8052.518
```

For the first possibility, 60% capital is allocated for large cap ETF, 10% capital to small and mid cap ETF's each, 10% each for alternative energy and oil/gas commodities. For this allocation, if we invest \$100k, we are 95% confident that our worst 20 trading day loss will not exceed \$8052.51 with an average profit of \$1683.2

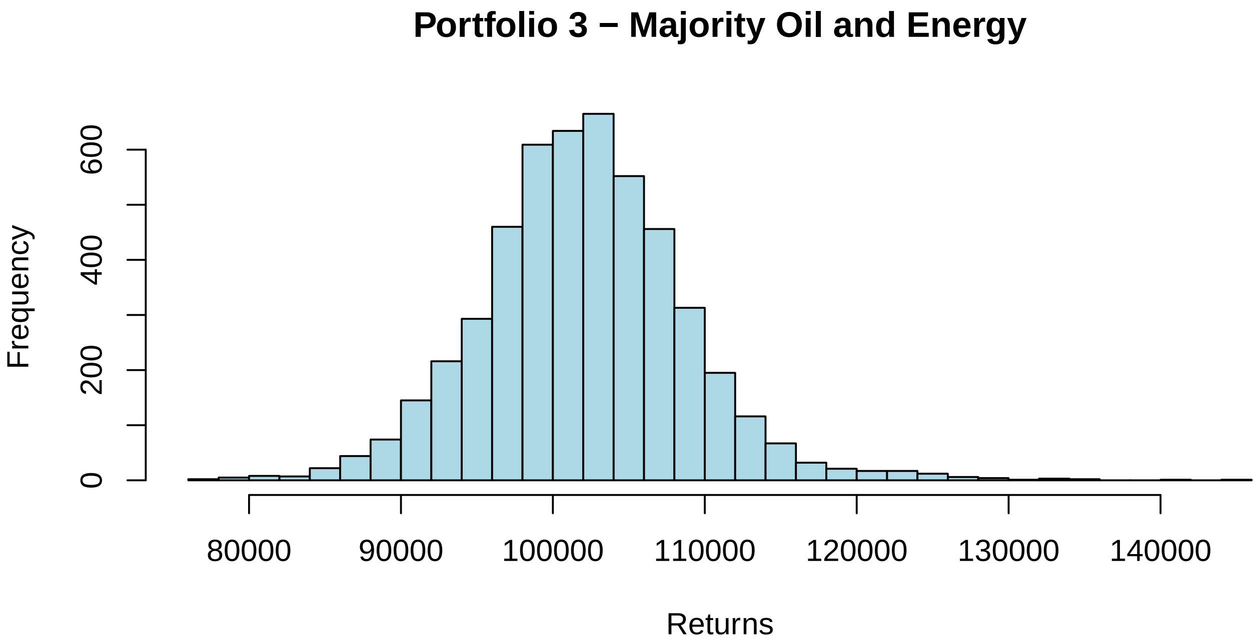
Portfolio 2 − Mix of mid and large caps

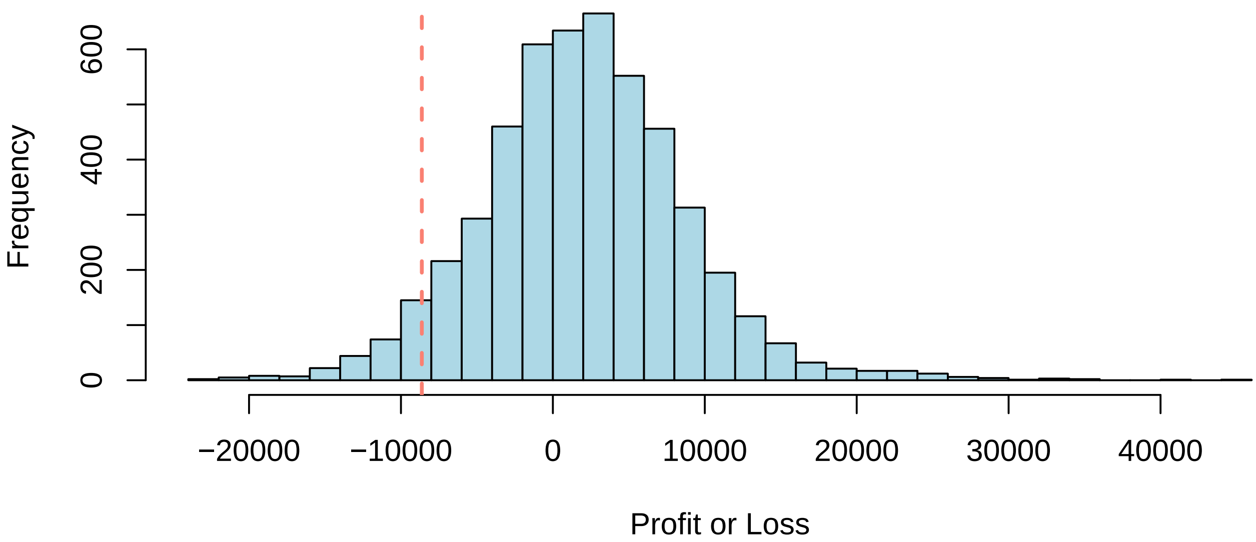





```
Mean :  102014.5

Average Profit :  2014.508

 5% Value at Risk :  -8526.643
```

For the second possibility, 40% capital is allocated for large cap ETF, 20% capital to small and mid cap ETF's each, 10% each for alternative energy and oil/gas commodities. For this allocation, if we invest \$100k, we are 95% confident that our worst 20 trading day loss will not exceed \$8526.64 with an average profit of \$2014.5





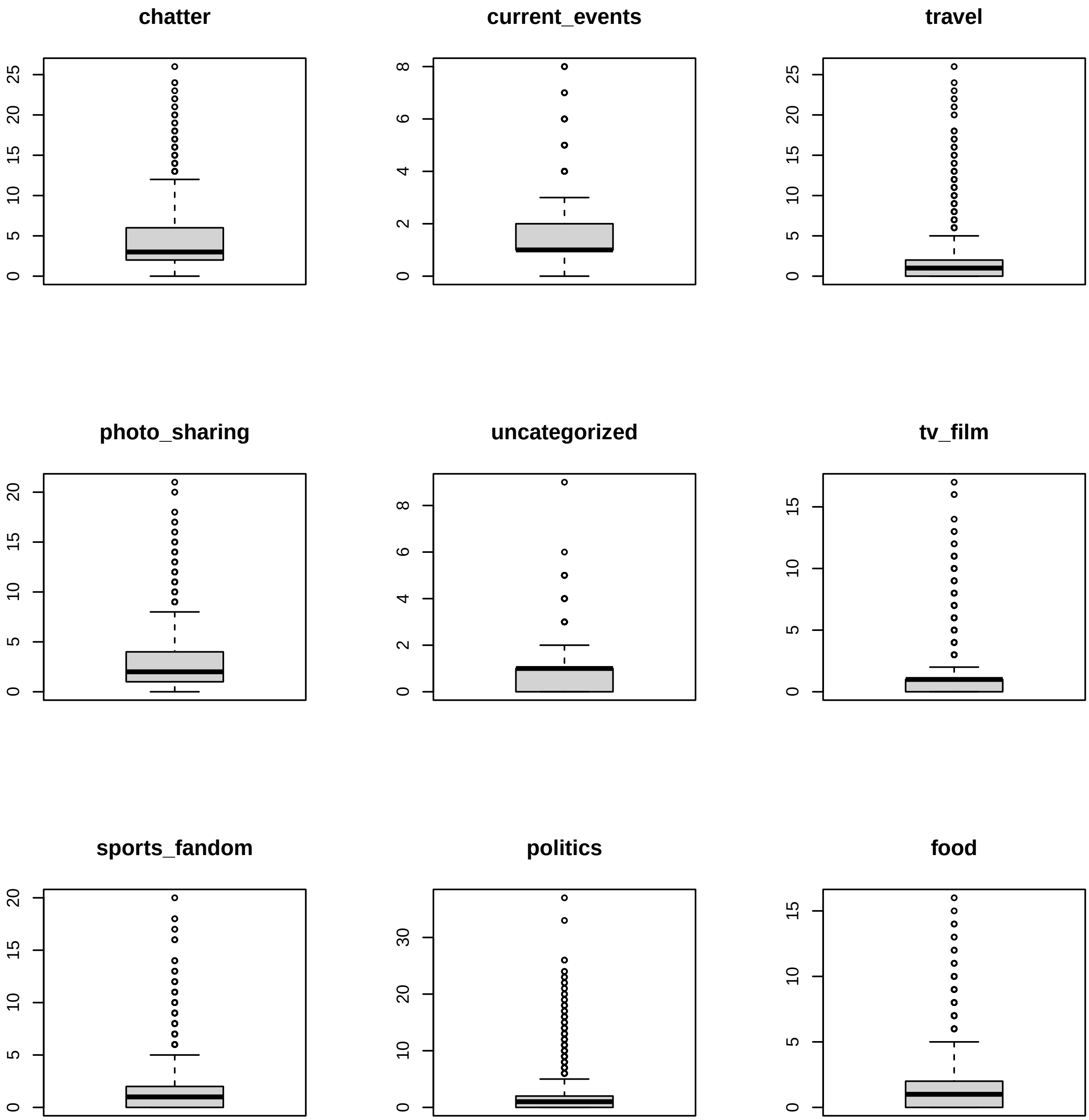

```
Mean :  102058.6

Average Profit :  2058.611

 5% Value at Risk :  -8625.87
```

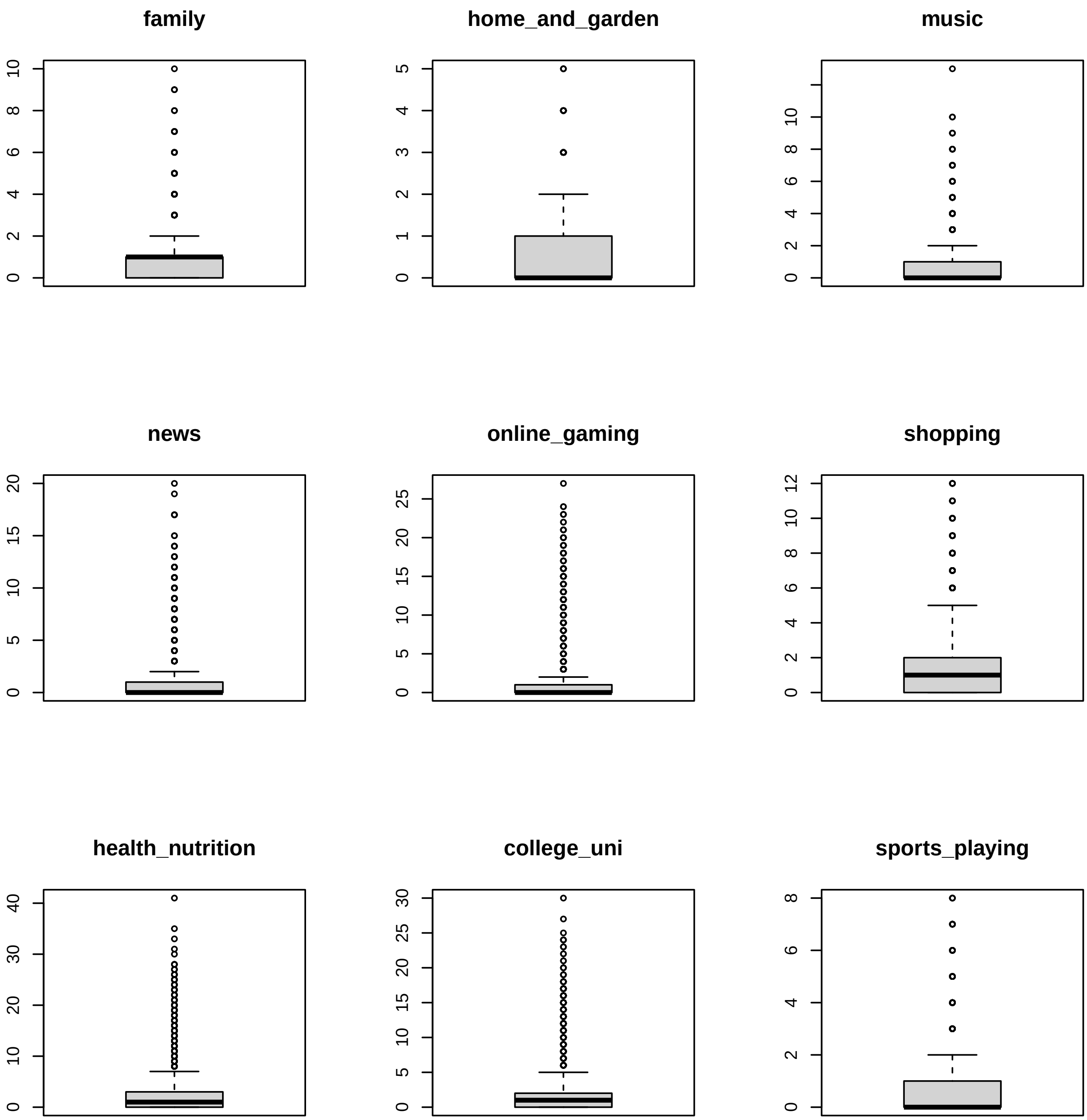
For the third possibility, 20% capital is allocated for large cap ETF, 10% capital to small and mid cap ETF's each, 30% each for alternative energy and oil/gas commodities. For this allocation, if we invest $100k, we are 95% confident that our worst 20 trading day loss will not exceed $8625.87 with an average profit of $2058.61

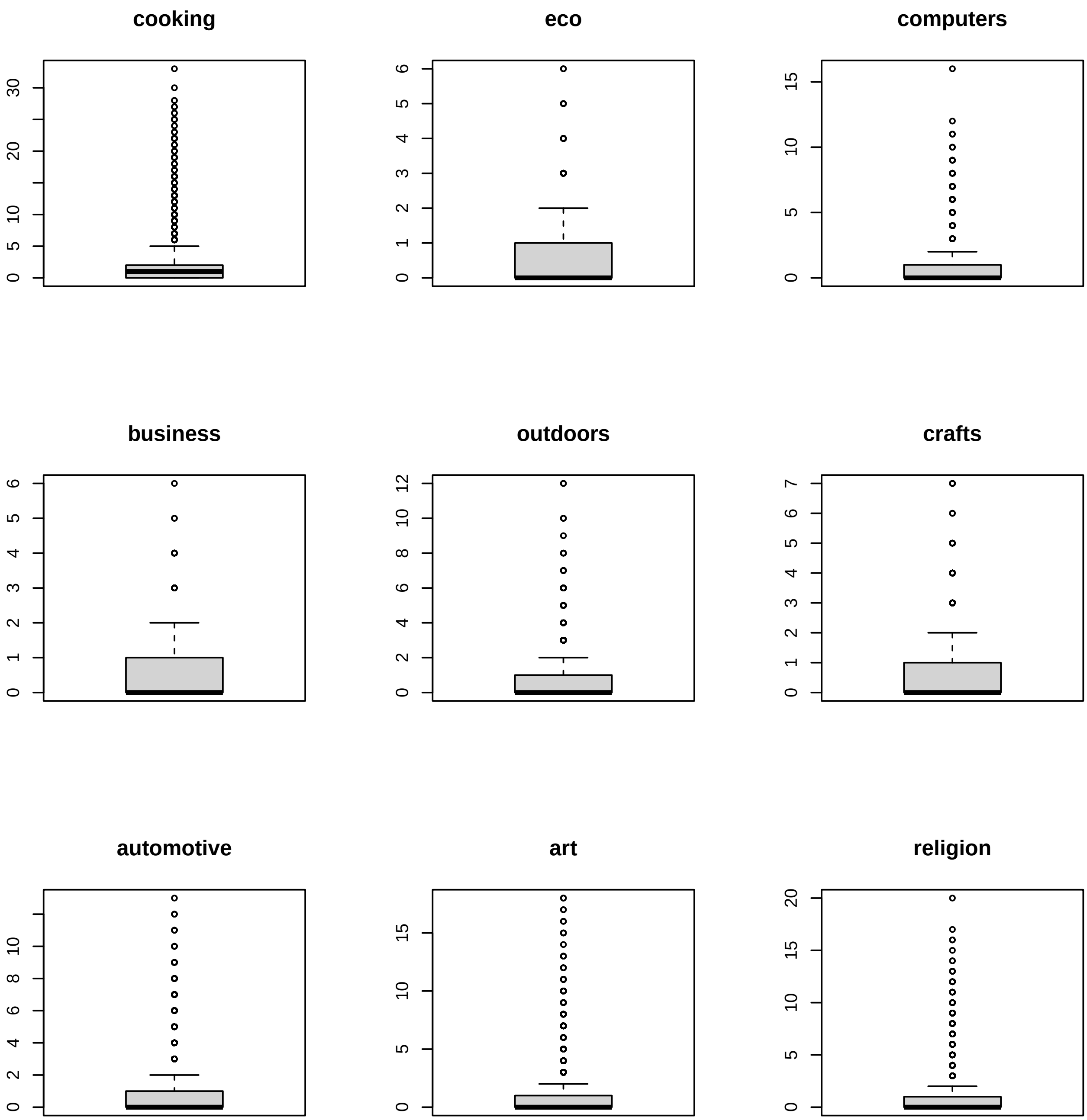
We tried a safe bet in a stable market, with major investments in large cap in Portfolio 1. Portfolio 2 had a well-balanced portfolio with equal proportion of initial wealth distributed amidst the large caps, mid and small caps to leverage diversification benefits. Portfolio 3 had more focus on high-risk high-reward bet on oil, gas and alternative energy ETFs given their volatility. It turns out, we see best returns with our Oil and Energy heavy portfolio, however value which worst 20 trading day loss will not exceed with 95% confidence also increases the most for this portfolio, given aforementioned volatility.

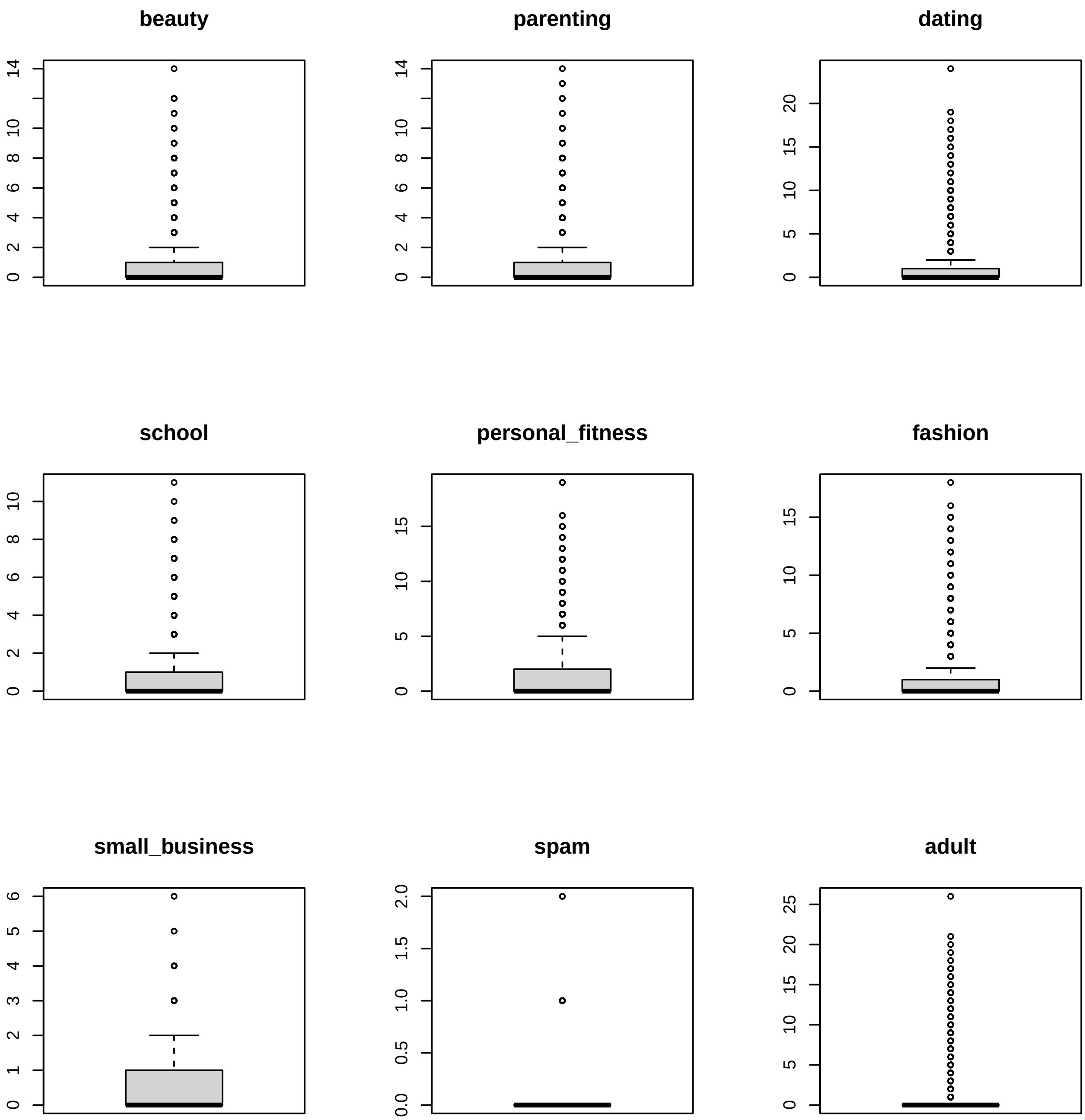
4. Market Segmentation

We start by exploring the datasets using boxplots to get a sense of univariate distributions of each of the variables. Boxplots help us understand which variables have more outliers or too spread out distributions that might not be helpful in explaining variance in dependent variable.

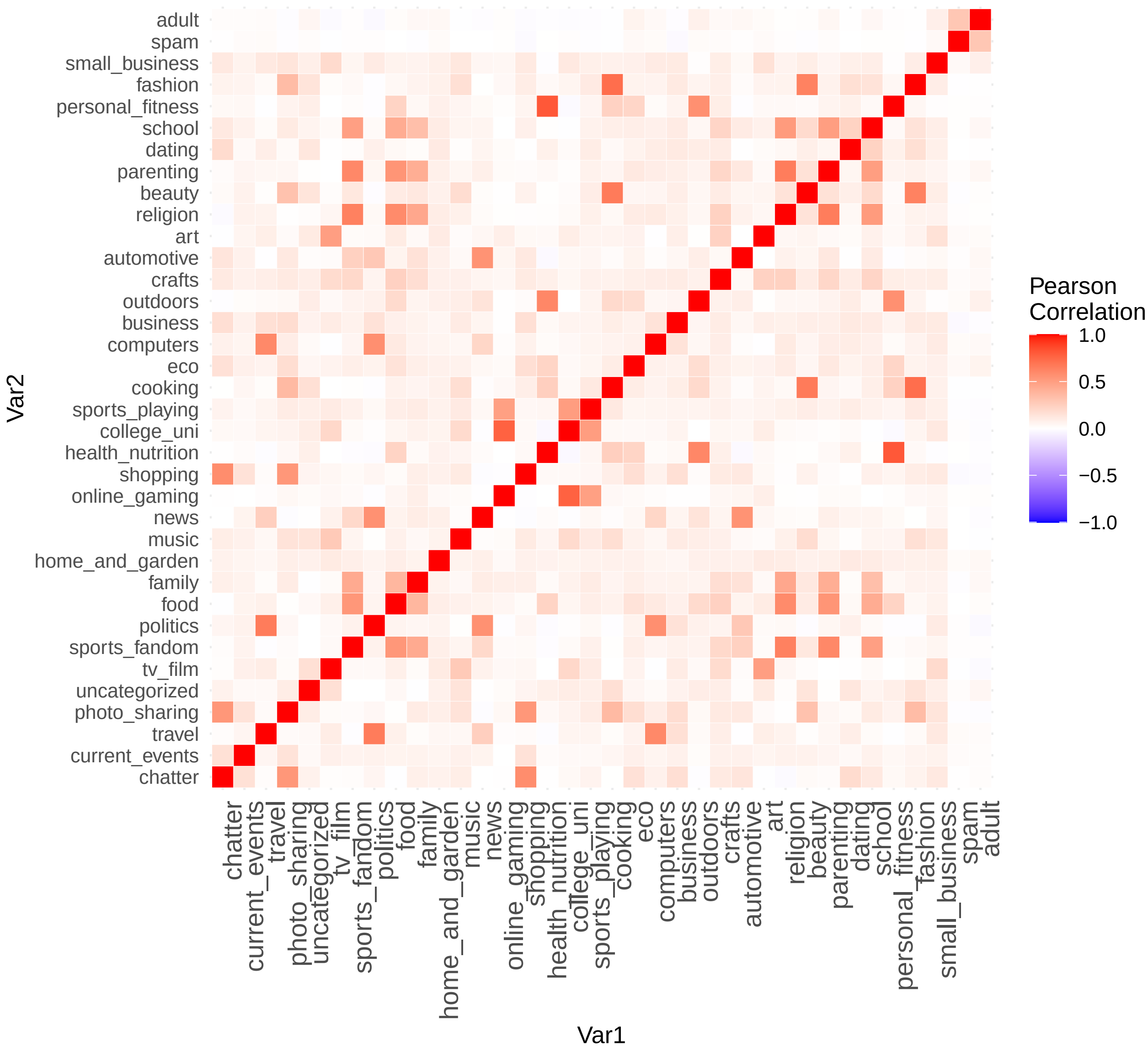



family
0 2 4 6 8 10
home_and_garden
0 1 2 3 4 5
music
0 2 4 6 8 10
news
0 5 10 15 20
online_gaming
0 5 10 15 20 25
shopping
0 2 4 6 8 10 12
health_nutrition
0 10 20 30 40
college_uni
0 5 10 15 20 25 30
sports_playing
0 2 4 6 8

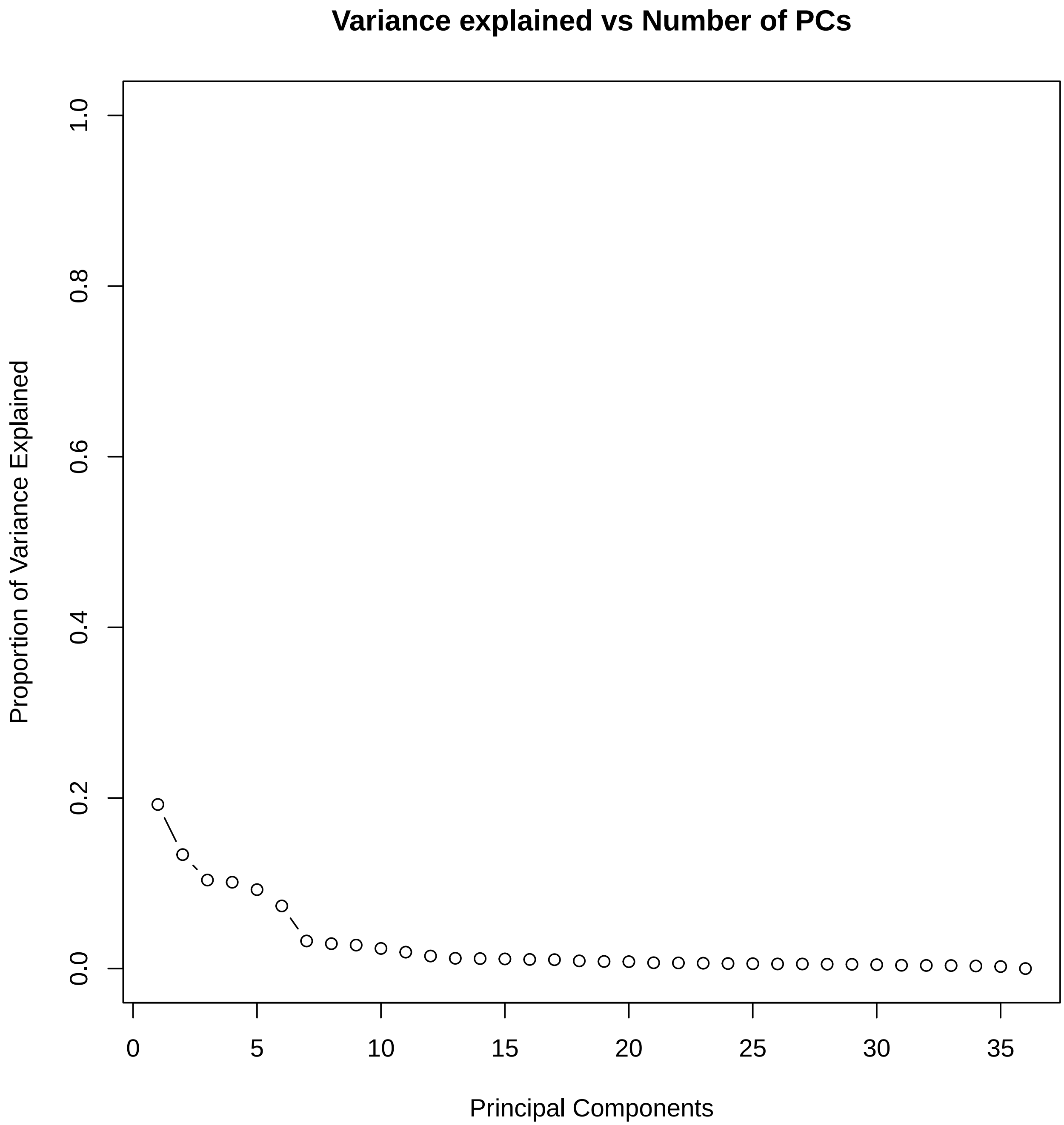

cooking
0
5
10
20
30
eco
0
1
2
3
4
5
6
computers
0
5
10
15
business
0
1
2
3
4
5
6
outdoors
0
2
4
6
8
10
12
crafts
0
1
2
3
4
5
6
7
automotive
0
2
4
6
8
10
art
0
5
10
15
religion
0
5
10
15
20



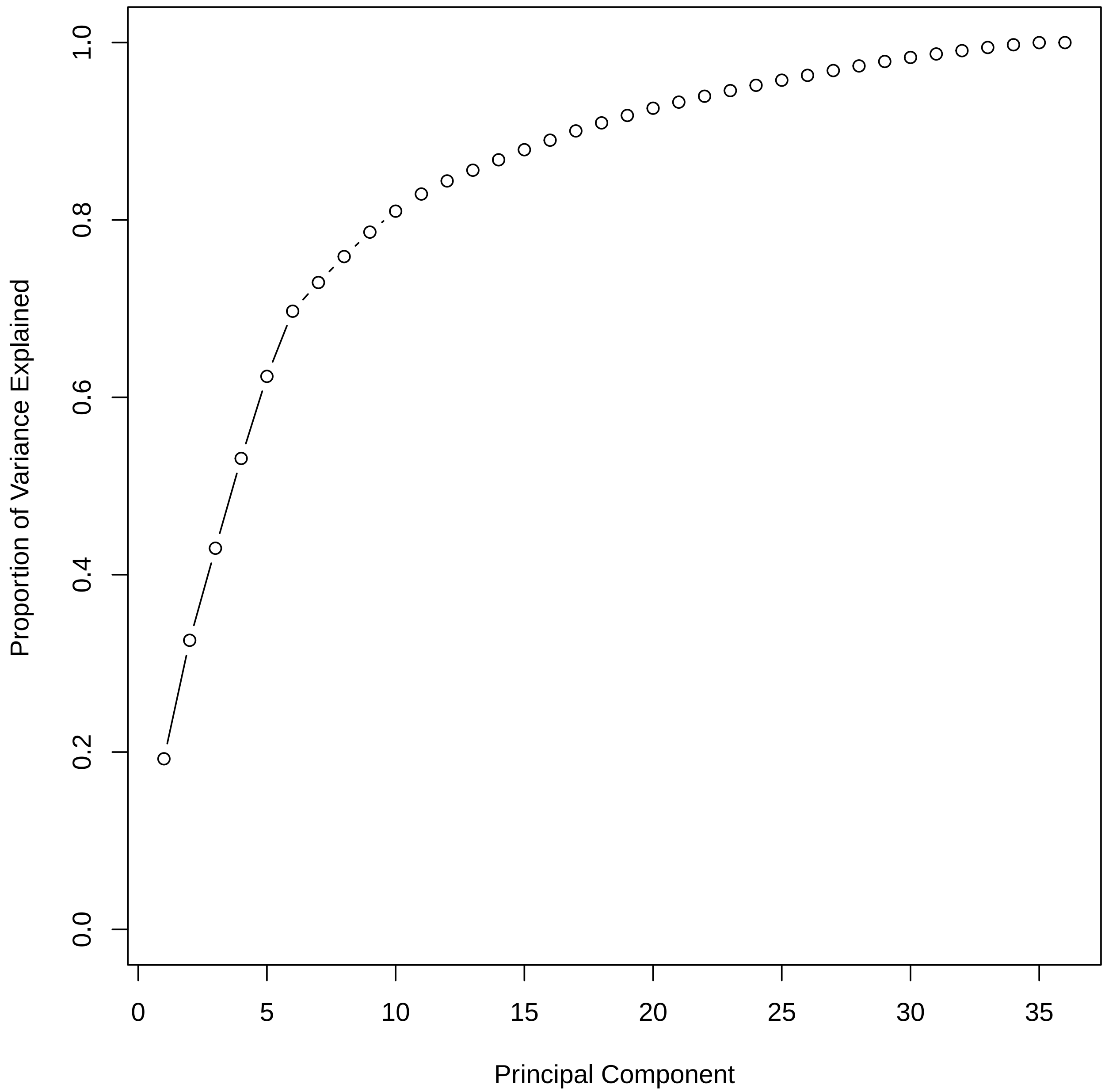
Further on, we explore bivariate associations by a corrplot. While we see few correlations in the 0.3-0.7 range, very few values are higher than that.



High correlations within tags is seem within : 1. Travel, Politics and Computers 2. Sports fandom, Religion and Parenting 3. College university and Online gaming 4. Health nutrition, Outdoors and Personal fitness 5. Cooking, Beauty and Fashion



After, 7 Principal components there seems to be a drop of variation explained per component and thus this can be used for further analysis.The cumulative variance explained and variable intution from PCA:

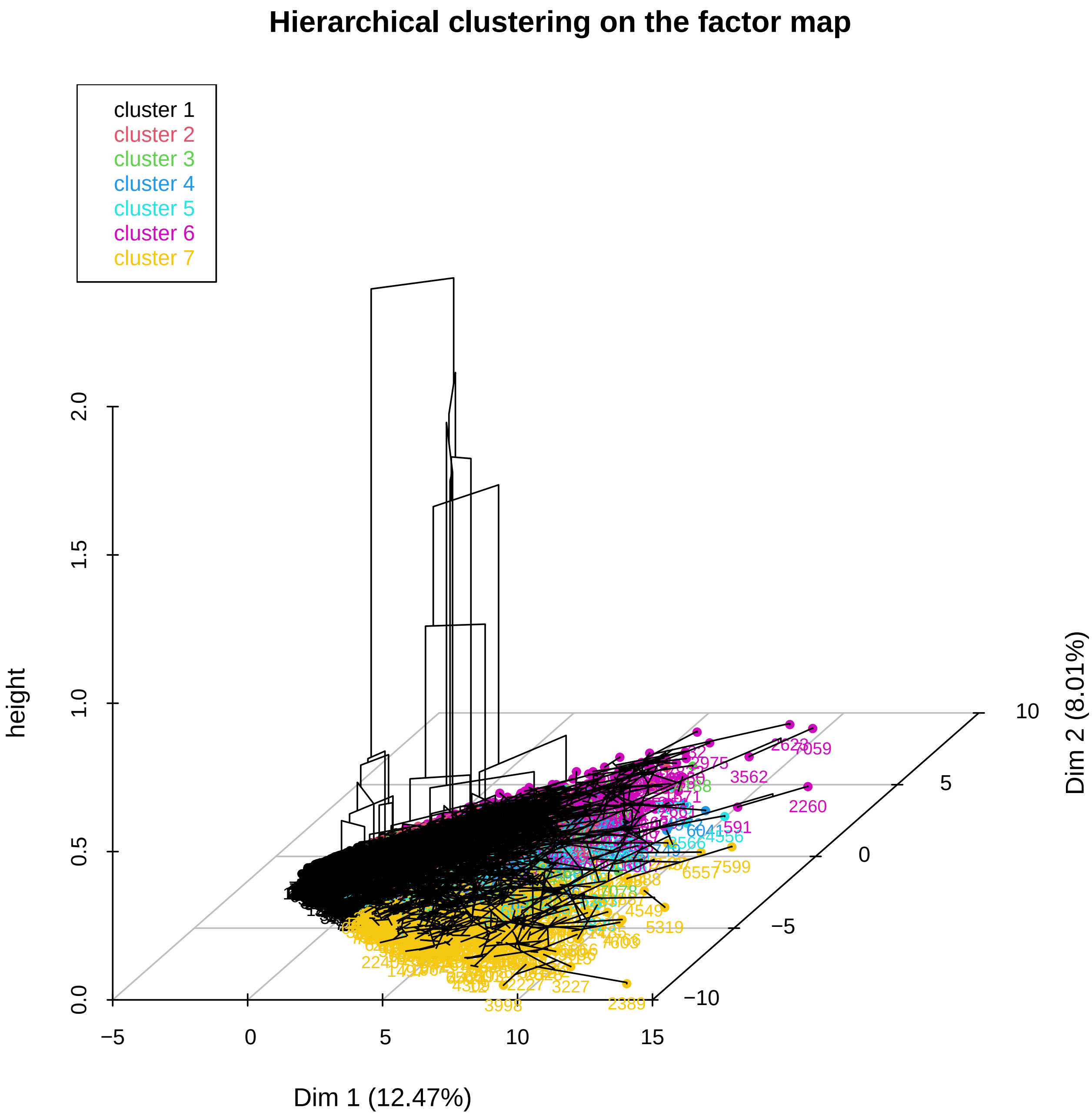



```
$loadings

Loadings:
                 PC1    PC2    PC3    PC4    PC5    PC6    PC7
chatter                  0.821               -0.148
current_events
travel                          0.497                      -0.475
photo_sharing            0.460                0.253
tv_film                                                    -0.270
sports_fandom                                       0.580  0.154
politics                        0.765
food                                                0.399 -0.132
family                                              0.189
```

```
home_and_garden
music
news                            0.328                       0.586
online_gaming                         -0.666
shopping                 0.306
health_nutrition  0.883
college_uni                           -0.723               -0.108
sports_playing                        -0.142
cooking                                       0.832
eco
computers                       0.199                      -0.141
business
outdoors          0.156
crafts
automotive                                                  0.383
art                                                        -0.262
religion                                             0.503 -0.142
beauty                                        0.255
parenting                                            0.365
dating                                                     -0.118
school                                               0.230
personal_fitness  0.416
fashion                                       0.383
small_business
spam
adult

                 PC1   PC2   PC3   PC4   PC5   PC6   PC7
SS loadings    1.000 1.000 1.000 1.000 1.000 1.000 1.000
Proportion Var 0.028 0.028 0.028 0.028 0.028 0.028 0.028
Cumulative Var 0.028 0.056 0.083 0.111 0.139 0.167 0.194

$rotmat
            [,1]        [,2]        [,3]        [,4]         [,5]         [,6]
[1,]  0.93099687  0.15515877 -0.01223470 -0.02462586  0.318667817  0.058614615
[2,] -0.27256380  0.82857806  0.21182179 -0.23088275  0.363471074  0.093216639
[3,]  0.08120565 -0.24578522  0.85159958 -0.34112087 -0.157724821  0.257888072
[4,]  0.08591401  0.30133687  0.35359131  0.82217701 -0.300919086 -0.006021501
[5,] -0.20243417 -0.37113982  0.17848672  0.36676882  0.801313251  0.094609361
[6,] -0.03167198  0.01468759 -0.26546436  0.08461340 -0.093689468  0.955180828
[7,]  0.05473136  0.01446981 -0.04903405 -0.10986101  0.005435302 -0.003335372
             [,7]
[1,] -0.058562899
[2,] -0.013837921
[3,]  0.005156792
[4,]  0.101116098
[5,]  0.062007337
[6,]  0.001508381
[7,]  0.991100729
```

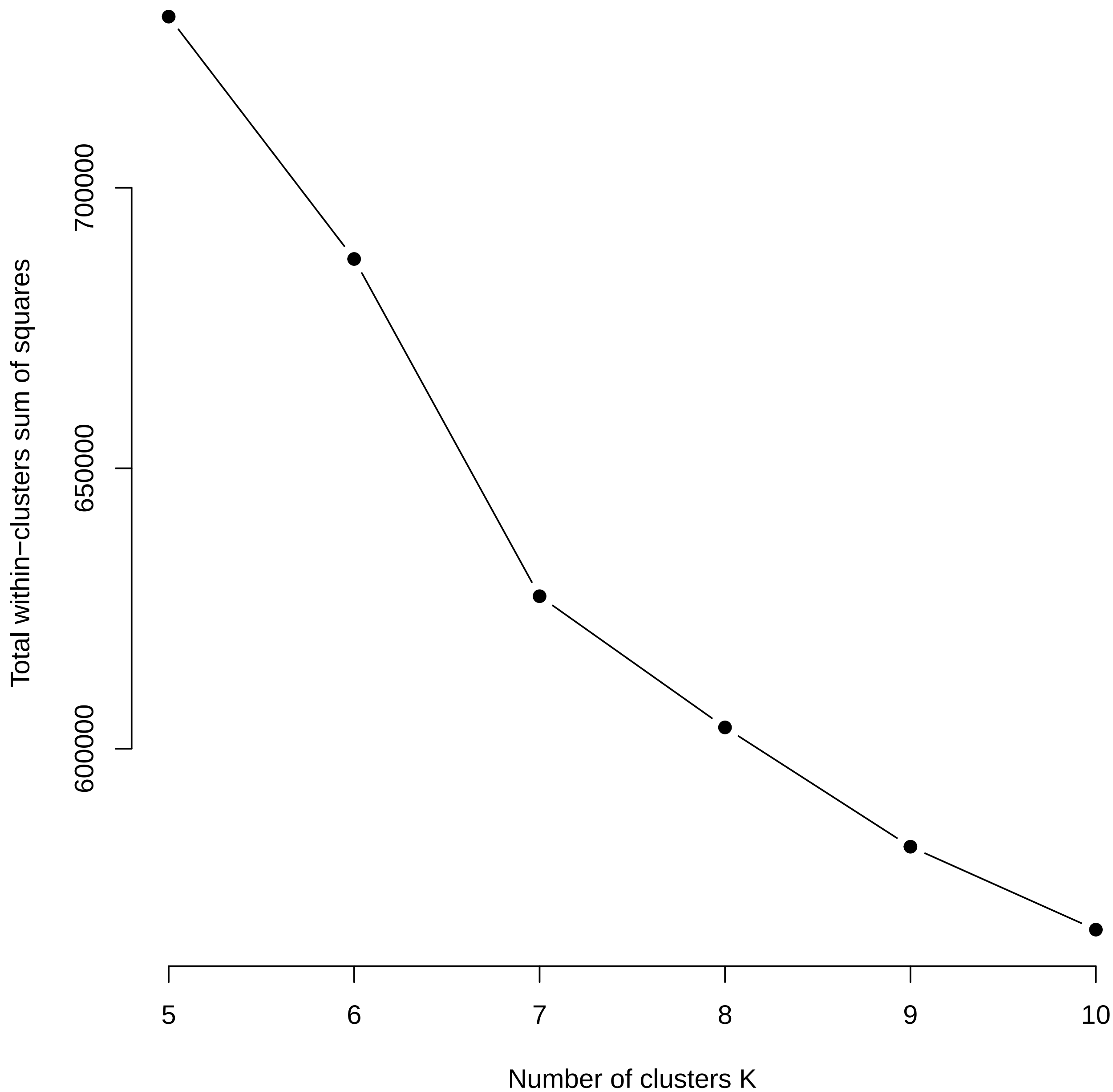
Cumulative graph shows saturation proportion of variance starting beyond 7 principal components.

Similar to insights in correlation plots, we identified factors to be composed of sections related to health & fitness: health_nutrition, outdoors and personal_fitness; shopping, chatter and photo sharing related to shopping; travel, politics, news and computers, college segment: college_uni, online gaming and sports playing; chatter & photo sharing related to cooking, beauty, fashion and so on. We tried to analyze the same using hclust and k-means clustering

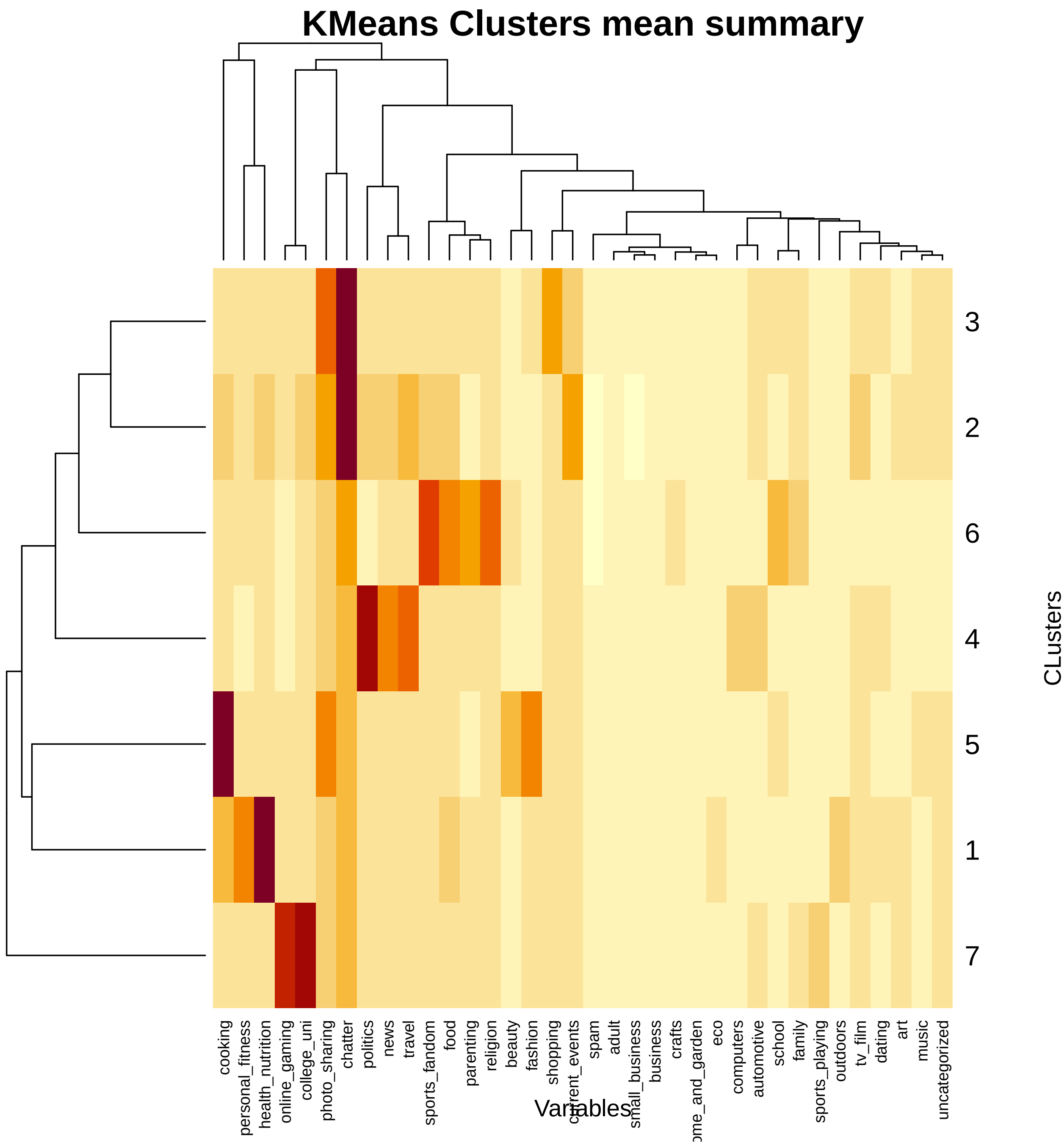


Recommended number of clusters = 7 with hierarchial clustering. Here we get an intuition of distance between different clusters via the dendogram. We see post the cut that creates 10 clusters, the distance does not reduce considerably upon increasing number of clusters.

Given number of datapoints is higher, and all variables are in the same unit (not requiring scaling) we can explore KMeans clustering with k ranging from 5 to 10 based on above graphs.



As expected, we're getting an elbow at K = 7. Leveraging the k value for k-means clustering and analyzing the market segments formed so.



Based on the final model, below were the customer segments. Based on their mean value of features and it's deviation from overall mean, we conjecture the general characterstic of a cohort that sets them apart from the rest.

Fitness Enthusiasts : Segment with cooking, personal_fitness, health_and_nutrition and relevant chatter variables higher than average

Travel Lovers : Chatter and photo sharing relevant to current events, sports fandom, travel

Shopaholics : Chatter and photo sharing relevant to shopping

News junkies : Politics, news and travel

Stylists : Chatter and photo sharing relevant to cooking, beauty and fashion

Older generation : Sports fandom, politics, religion and parenting

Teenagers or College students : Online gaming, sports playing and info related college universities

5. Author Attribution

Let us set up the readerPlain function

Let's getting the list of train folder names and print the first 5 folder names from 'C50Train'

```
AaronPressman
AlanCrosby
AlexanderSmith
BenjaminKangLim
BernardHickey
```

Getting the list of all the files from all the 50 train folders and printing the names of the first 5 txt files

```
ReutersC50/C50train/AaronPressman/106247newsML.txt
ReutersC50/C50train/AaronPressman/120600newsML.txt
ReutersC50/C50train/AaronPressman/120683newsML.txt
ReutersC50/C50train/AaronPressman/136958newsML.txt
ReutersC50/C50train/AaronPressman/137498newsML.txt
```

Reading all 50 x 50 files and printing the metadata of the 1st train file

```
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 1998
```

Let's clean up the file names by removing the directory location details. This uses the piping operator from magrittr. And let's also rename the articles. Priinting the first train file

```
$AaronPressman106247newsML.txt
<<PlainTextDocument>>
Metadata:  7
Content:  chars: 1998
```

Now that we have the documents in a vector, let's create a text mining corpus for us to start analyzing text

```
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 2500
```

Let's use some pre-processing/tokenization steps. tm_map just maps some function to every document in the corpus

Let's remove some stopwords by using the functions available in tm

Let's create a Document term Matrix and remove those terms that have count 0 in >95% of docs given sparse variables wouldn't anyways be helpful

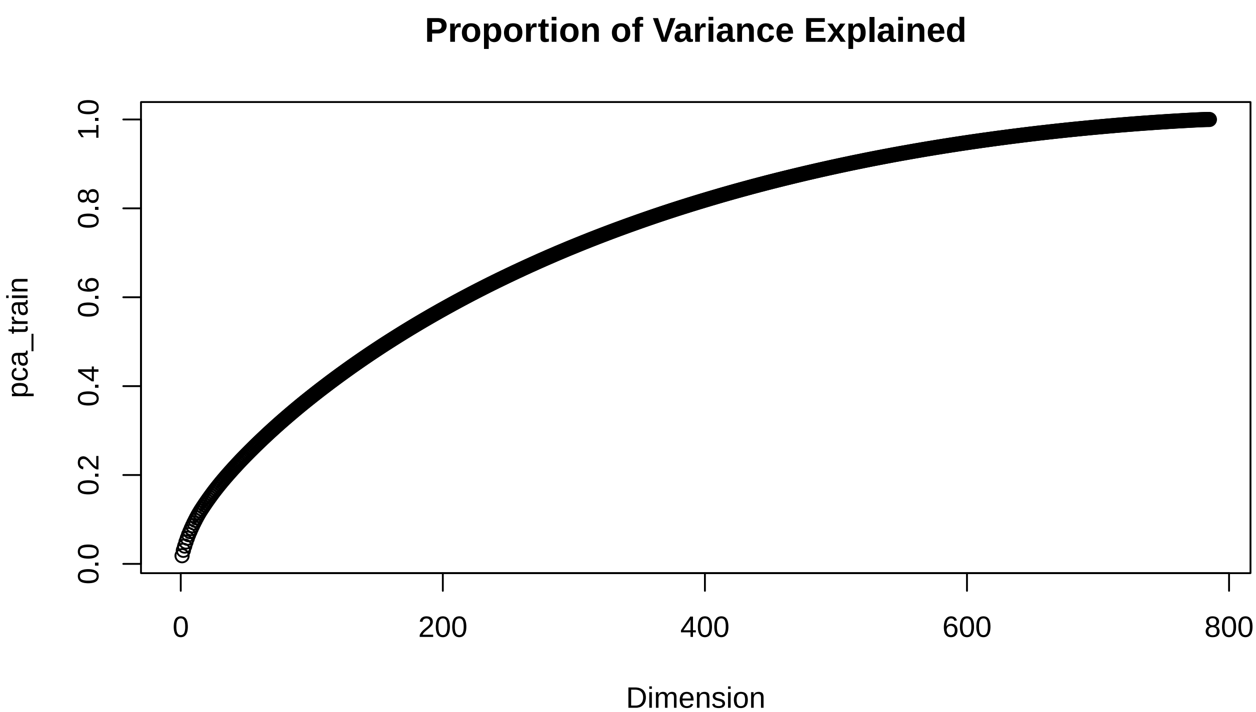
```
<<DocumentTermMatrix (documents: 2500, terms: 801)>>
Non-/sparse entries: 280686/1721814
Sparsity           : 86%
Maximal term length: 18
Weighting          : term frequency (tf)
```

Constructing TF IDF weights

Let's create the y values, which is, the names of folders are the names of the authors

```
AaronPressman
AaronPressman
AaronPressman
AaronPressman
AaronPressman
```

Since,We have way too many features, let's try using PCA for dimensionality reduction



We see that around 90 percent of variation in data is explained by 500 components

Let's repeat all the above data processing steps for test data

```
Warning in tm_map.SimpleCorpus(my_documents_test, content_transformer(tolower)):
transformation drops documents
```

```
Warning in tm_map.SimpleCorpus(my_documents_test,
content_transformer(removeNumbers)): transformation drops documents
```

```
Warning in tm_map.SimpleCorpus(my_documents_test,
content_transformer(removePunctuation)): transformation drops documents
```

```
Warning in tm_map.SimpleCorpus(my_documents_test,
content_transformer(stripWhitespace)): transformation drops documents

Warning in tm_map.SimpleCorpus(my_documents_test,
content_transformer(removeWords), : transformation drops documents
```

Our train data, target variable and test sets are ready, Let's try Random Forest and Naive Baye algorithms to predict author of any given article.

```
Classification Accuracy with Naive Bayes :  0.4456
```

With Naive Bayes, we achieve an accuracy of 44.56 percent. Using accuracy, we get a general sense of our hit-rate, for better error metrics we may explore precision, recall values in multi-class settings as well.

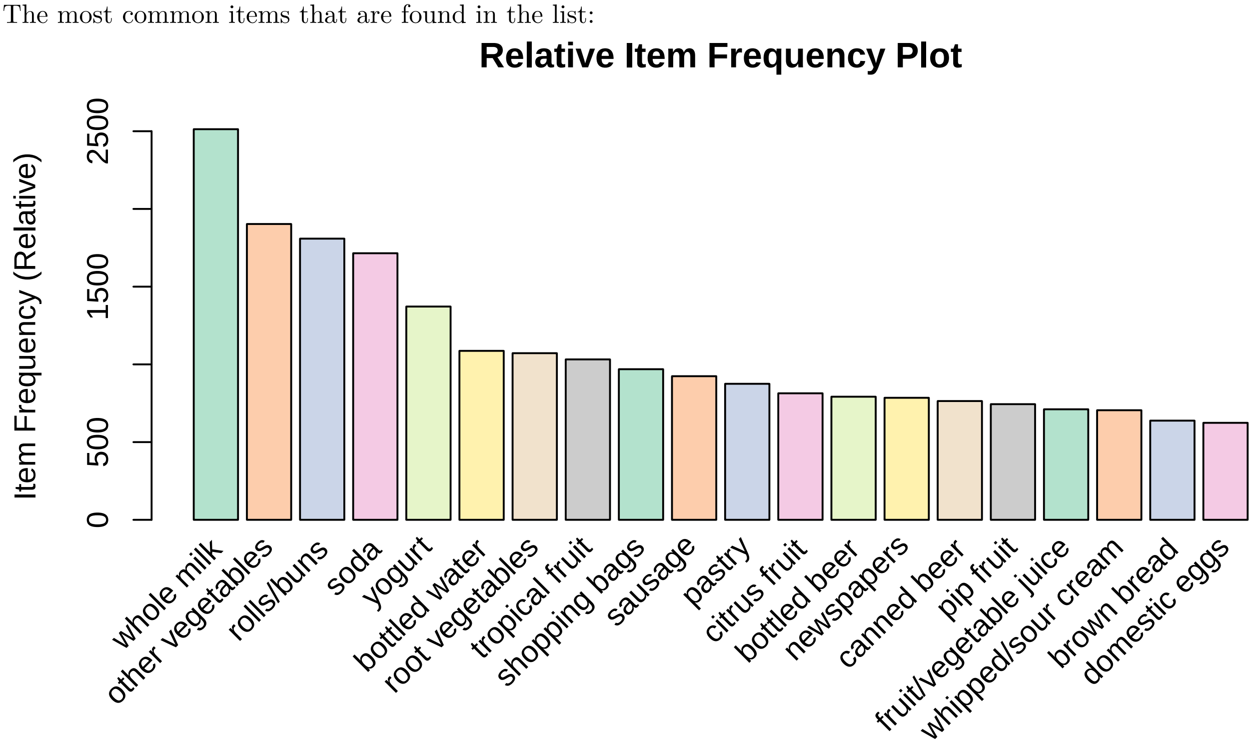
Let's use Random Forests with number of predictors equal to 40 and ntrees = 1600

```
Classification Accuracy with RandomForest :  0.5096
```

With Random forests, we achieve an accuracy of 50.96 percent with an improvement over the NB model, although computational time increases considerably in RF. Currently we removed the words in test that the model hasn't seen through train data. Ideally, we'd want to create a list of words present in the test data and absent in train and add those with 0 frequency in train corpus. Essentially, similar to adding extra levels of a factor dependent variable in multi-class classification so that the train data despite having never seen a particular level is aware of it's existence.

6. Association Mining Rules

We are trying to identify patterns in shopping behavior of customers from a list of their grocery purchase. The most common items that are found in the list:

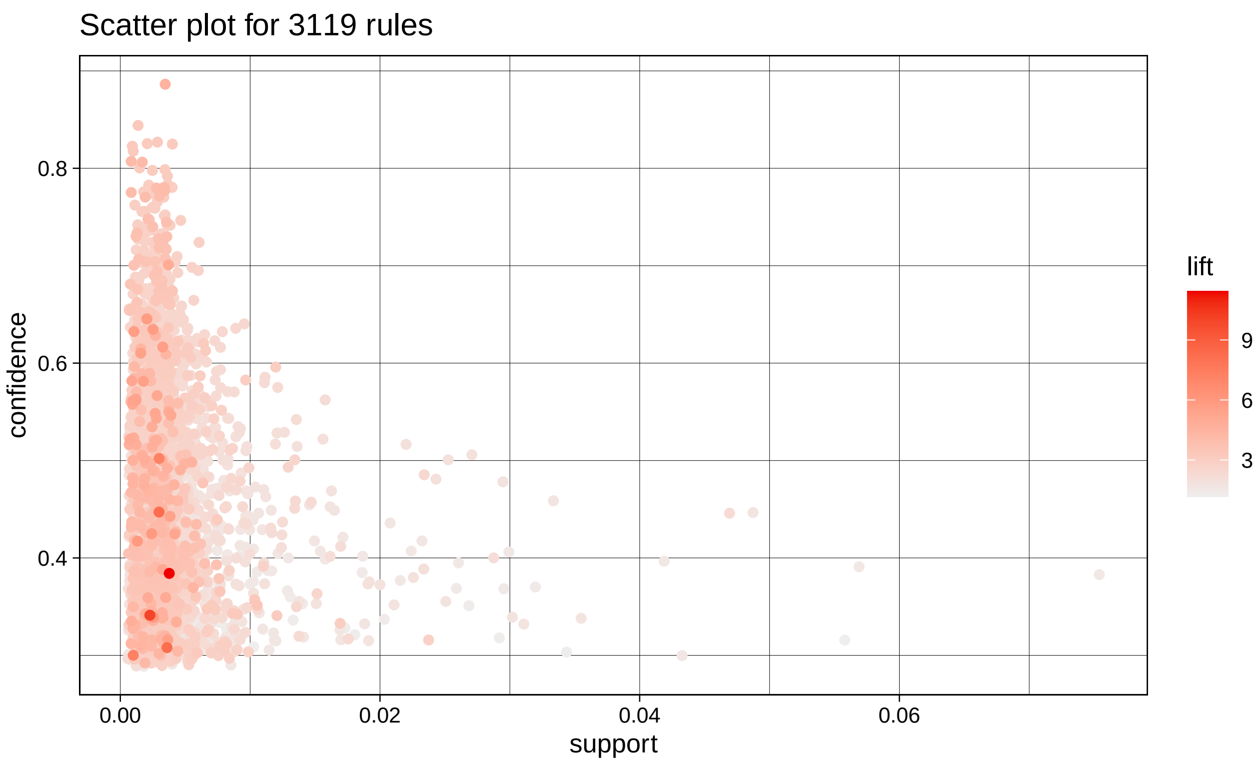


Next we try to model for association mining rules based apriori algorithm:

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
       0.01    0.1    1 none FALSE            TRUE       5   0.002      1
 maxlen target  ext
     40  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 19

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [147 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [11060 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].

Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
        0.1    0.1    1 none FALSE            TRUE       5   0.002      1
 maxlen target  ext
     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 19

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [147 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [8340 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].

Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
        0.3    0.1    1 none FALSE            TRUE       5   0.002      1
 maxlen target  ext
     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
```

```
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 19

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [147 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [3119 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

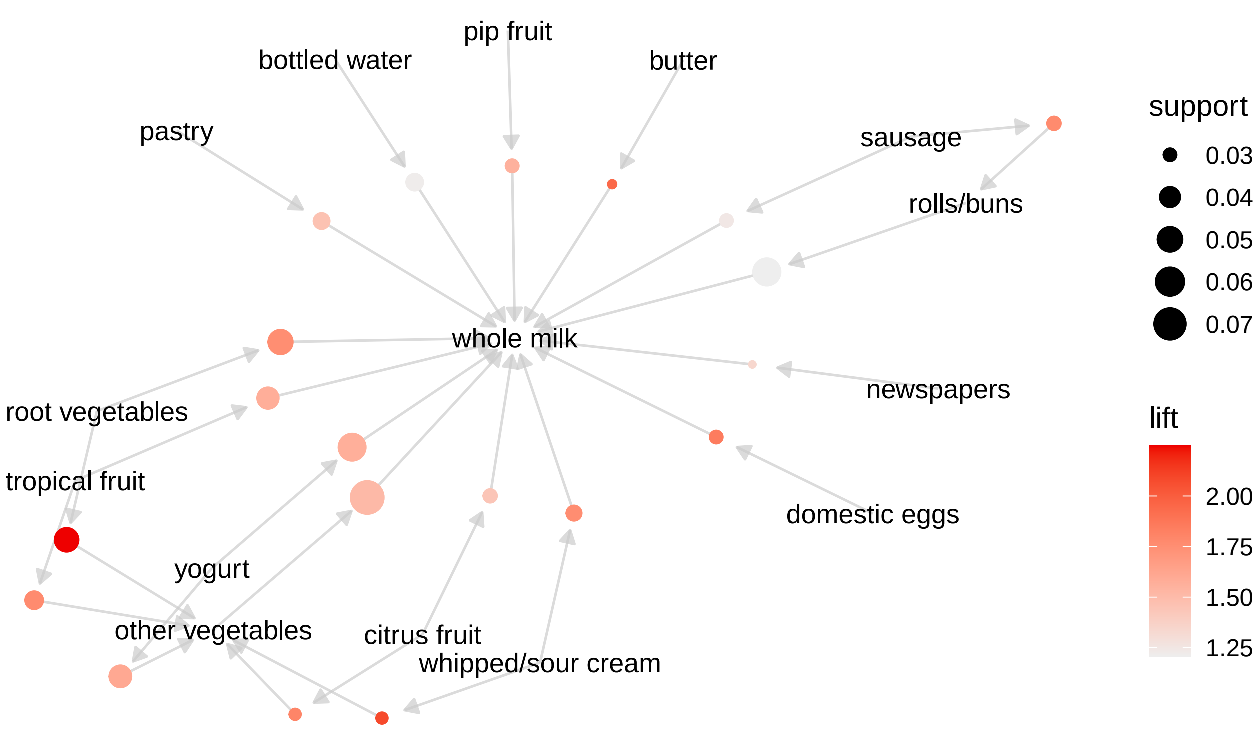
We checked the results for rules varying support, confidence; we assumed that an item must occur atleast 20 times to be considered in a rule and tried to restrict confidence in successive turns, from 0.01 to 0.3, i.e. conditional probability of an item to be atleast 0.3 to be considered in the rule finally. We got around 3119 rules finally based on aforementioned conditions. Next we restrict with conditions on lift, an item should be atleast 5 times more likely to occur with the associated items in the list.




```
    lhs                          rhs                  support confidence    coverage      lift count
[1] {liquor}                  => {bottled beer}     0.004677173  0.4220183 0.011082867  5.240594    46
[2] {popcorn}                 => {salty snack}      0.002236909  0.3098592 0.007219115  8.192110    22
[3] {Instant food products} => {hamburger meat}     0.003050330  0.3797468 0.008032537 11.421438    30
[4] {other vegetables,
     rice}                    => {root vegetables}  0.002236909  0.5641026 0.003965430  5.175325    22
[5] {herbs,
     yogurt}                  => {root vegetables}  0.002033554  0.5714286 0.003558719  5.242537    20
[6] {processed cheese,
     whole milk}              => {white bread}      0.002135231  0.3043478 0.007015760  7.230099    21
[7] {flour,
     whole milk}              => {sugar}            0.002846975  0.3373494 0.008439248  9.963457    28
```

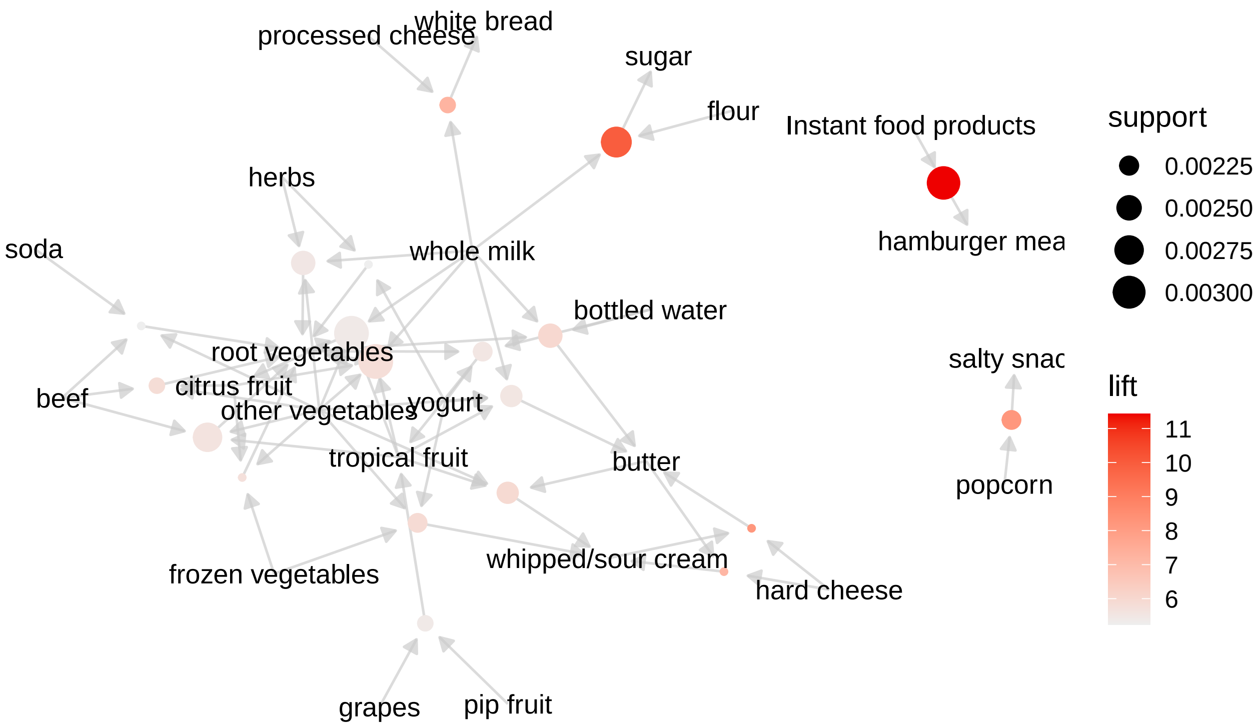
```
[8]  {grapes,
      pip fruit}             => {tropical fruit}     0.002135231  0.5675676 0.003762074  5.408941    21
[9]  {frozen meals,
      tropical fruit}        => {pip fruit}          0.002135231  0.3888889 0.005490595  5.140756    21
[10] {butter,
      hard cheese}           => {whipped/sour cream} 0.002033554  0.5128205 0.003965430  7.154028    20
[11] {hard cheese,
      whipped/sour cream}    => {butter}             0.002033554  0.4545455 0.004473818  8.202669    20
[12] {berries,
      whole milk}            => {whipped/sour cream} 0.004270463  0.3620690 0.011794611  5.050990    42
[13] {herbs,
      other vegetables,
      whole milk}            => {root vegetables}    0.002440264  0.6000000 0.004067107  5.504664    24
[14] {grapes,
      other vegetables,
      whole milk}            => {tropical fruit}     0.002033554  0.5263158 0.003863752  5.015810    20
[15] {frozen vegetables,
      other vegetables,
      yogurt}                => {whipped/sour cream} 0.002236909  0.4230769 0.005287239  5.902073    22
[16] {citrus fruit,
      frozen vegetables,
      other vegetables}      => {root vegetables}    0.002033554  0.6250000 0.003253686  5.734025    20
[17] {beef,
      butter,
      whole milk}            => {root vegetables}    0.002033554  0.5555556 0.003660397  5.096911    20
[18] {beef,
      citrus fruit,
      other vegetables}      => {root vegetables}    0.002135231  0.6363636 0.003355363  5.838280    21
[19] {beef,
      citrus fruit,
      whole milk}            => {root vegetables}    0.002236909  0.5641026 0.003965430  5.175325    22
[20] {beef,
      other vegetables,
      tropical fruit}        => {root vegetables}    0.002745297  0.6136364 0.004473818  5.629770    27
[21] {beef,
      tropical fruit,
      whole milk}            => {root vegetables}    0.002541942  0.5555556 0.004575496  5.096911    25
[22] {beef,
      other vegetables,
      soda}                  => {root vegetables}    0.002033554  0.5714286 0.003558719  5.242537    20
[23] {butter,
      other vegetables,
      tropical fruit}        => {whipped/sour cream} 0.002338587  0.4259259 0.005490595  5.941818    23
[24] {bottled water,
      root vegetables,
      whole milk}            => {butter}             0.002440264  0.3333333 0.007320793  6.015291    24
[25] {bottled water,
      root vegetables,
      yogurt}                => {tropical fruit}     0.002236909  0.5789474 0.003863752  5.517391    22
[26] {butter,
      other vegetables,
      whole milk,
      yogurt}                => {tropical fruit}     0.002338587  0.5348837 0.004372140  5.097463    23
[27] {other vegetables,
```

```
      tropical fruit,
      whole milk,
      yogurt}                => {butter}           0.002338587  0.3066667 0.007625826  5.534067    23
[28] {citrus fruit,
      other vegetables,
      tropical fruit,
      whole milk}            => {root vegetables}  0.003152008  0.6326531 0.004982206  5.804238    31
[29] {citrus fruit,
      other vegetables,
      root vegetables,
      whole milk}            => {tropical fruit}   0.003152008  0.5438596 0.005795628  5.183004    31
[30] {other vegetables,
      root vegetables,
      tropical fruit,
      whole milk}            => {citrus fruit}     0.003152008  0.4492754 0.007015760  5.428284    31

Available control parameters (with default values):
layout   =  list(fun = function (graph, dim = 2, ...)  {     if ("layout" %in% graph_attr_names(graph))
edges    =  <environment>
nodes    =  <environment>
nodetext     =  <environment>
colors   =  c("#EE0000FF", "#EEEEEEFF")
engine   =  ggplot2
max  =  100
verbose  =  FALSE
```




```
Available control parameters (with default values):
layout   =  list(fun = function (graph, dim = 2, ...)  {     if ("layout" %in% graph_attr_names(graph))
edges    =  <environment>
nodes    =  <environment>
```

```
nodetext     =  <environment>
colors   =  c("#EE0000FF", "#EEEEEEFF")
engine   =  ggplot2
max  =  100
verbose  =  FALSE
```



!arules_gephi.png