HW 4

Ashesh Shrestha

5/8/2021

Clustering and PCA

First of all I started by cleaning the data. I centered and scaled the data.

Then, I applied k-means technique for clustering. I used k =2 as there were 2 types of wine by color, red and white, with 25 starts. In order to see if the k-means has clustered data points by wine color into red and white wine, I compare the averages of chemical properties for white wine and red wine in our original data with that of the clustered data.

```
## # A tibble: 2 x 13
##   color fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## * <chr>         <dbl>            <dbl>       <dbl>          <dbl>     <dbl>
## 1 red            8.32            0.528       0.271           2.54    0.0875
## 2 white          6.85            0.278       0.334           6.39    0.0458
##   free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates alcohol
## *               <dbl>                <dbl>   <dbl> <dbl>     <dbl>   <dbl>
## 1                15.9                 46.5   0.997  3.31     0.658    10.4
## 2                35.3                 138.   0.994  3.19     0.490    10.5
##   quality
## *   <dbl>
## 1    5.64
## 2    5.88
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##            8.2895922            0.5319416            0.2695435
##       residual.sugar            chlorides  free.sulfur.dioxide
##            2.6342666            0.0883238           15.7647596
## total.sulfur.dioxide              density                   pH
##           48.6396835            0.9967404            3.3097200
##            sulphates              alcohol
##            0.6567194           10.4015216
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##           6.85167903           0.27458385           0.33524928
##       residual.sugar            chlorides  free.sulfur.dioxide
##           6.39402555           0.04510424          35.52152864
## total.sulfur.dioxide              density                   pH
##         138.45848785           0.99400486           3.18762464
##            sulphates              alcohol
##           0.48880511          10.52235888
```

If we compare averages of the chemical properties of red and white wine in our original data with the averages of chemical properties of red and white wine in clustered data, we can see that averages of chemical properties for red wine in both original and k-means clustered data are almost the same. Similarly, that averages of chemical properties for white wine are also pretty much the same in both original and clustered data. This hints towards the fact that k-means is easily capable of distinguishing the red wines from the white ones.

To further verify this, I have also made a confusion matrix. In the table, we can see that k-means has pretty accurately clustered data by wine color. With an accuracy of 98.5%, we can conclude that k-means clustering has done a very good job in terms of dimension reduction

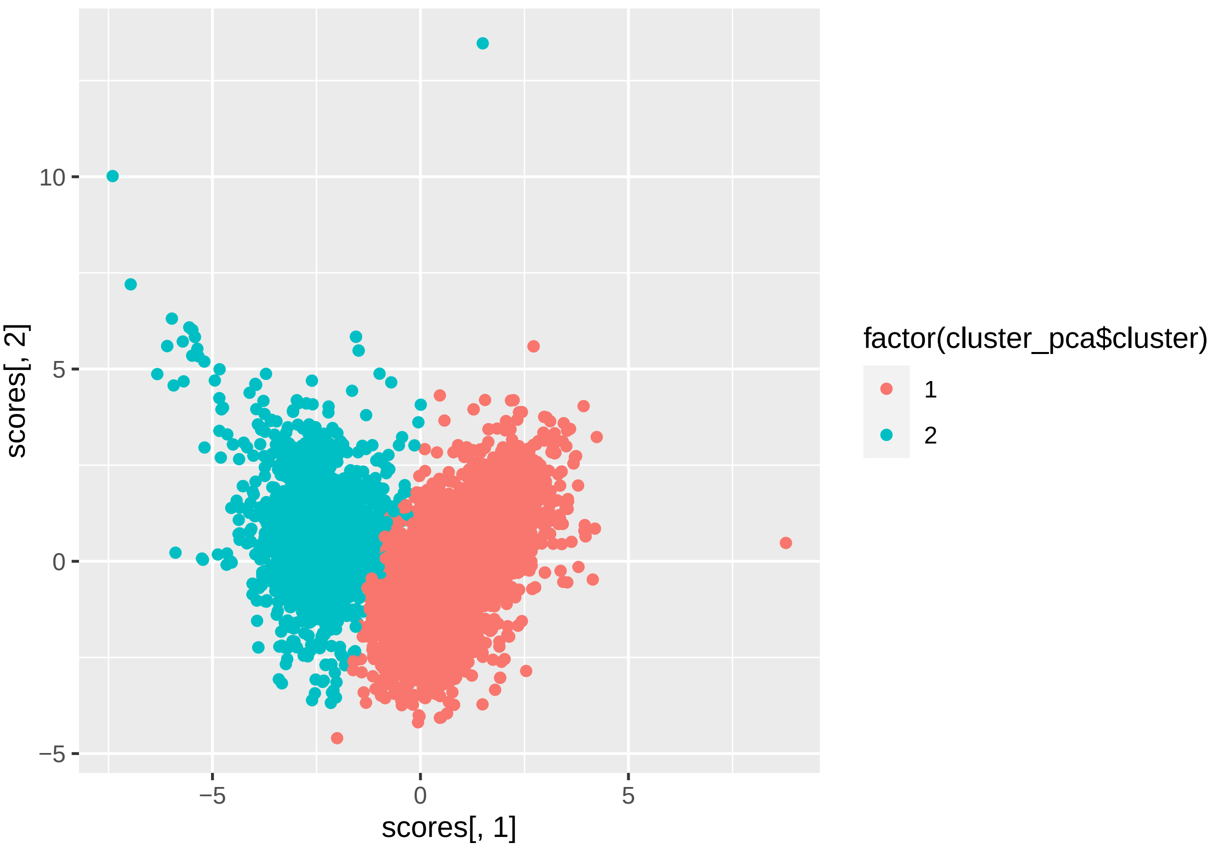
```
##        cluster
## color   red_hat white_hat
##   red      1575        24
##   white      68      4830
```

```
## [1] 0.9858396
```

After, k-means, I proceeded to perform Principal Component Analysis (PCA). As seen in the table below, the first three principal components form 64.3% of the total variance in the data set, which is significantly high. Hence, I used first three components to perform clustering.

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6     PC7
## Standard deviation     1.7407 1.5792 1.2475 0.98517 0.84845 0.77930 0.72330
## Proportion of Variance 0.2754 0.2267 0.1415 0.08823 0.06544 0.05521 0.04756
## Cumulative Proportion  0.2754 0.5021 0.6436 0.73187 0.79732 0.85253 0.90009
##                            PC8     PC9   PC10    PC11
## Standard deviation     0.70817 0.58054 0.4772 0.18119
## Proportion of Variance 0.04559 0.03064 0.0207 0.00298
## Cumulative Proportion  0.94568 0.97632 0.9970 1.00000
```

```
##                        PC1   PC2   PC3
## fixed.acidity        -0.24  0.34 -0.43
## volatile.acidity     -0.38  0.12  0.31
## citric.acid           0.15  0.18 -0.59
## residual.sugar        0.35  0.33  0.16
## chlorides            -0.29  0.32  0.02
## free.sulfur.dioxide   0.43  0.07  0.13
## total.sulfur.dioxide  0.49  0.09  0.11
## density              -0.04  0.58  0.18
## pH                   -0.22 -0.16  0.46
## sulphates            -0.29  0.19 -0.07
## alcohol              -0.11 -0.47 -0.26
```



```
##        cluster
## color   red_hat white_hat
##   red        24      1575
##   white    4816        82
```

```
## [1] 0.01631522
```

The clustering done by using the scores from three principal components also did a good job. The accuracy level stood at 98.3 %. But, PCA is not as straight forward as k-means. I used the scores from the principal components to form the clusters. As the accuracy of k-means is relatively higher and it is straight forward, I conclude that it makes more sense to use k-means technique for the given data.

The quality of wine is being rated in a scale of 1-10, however, in our data set there is no rating of 1,2 or 10. Thus, the wine in our data set was rated between 2 and 9 inclusive. I performed k-means with k= 7 and 25 starts.

```
##             cluster2$cluster
## wine$quality   1   2   3   4   5   6   7
##            3   7   4   2   5   4   6   2
##            4  24  14   2  65  21  64  26
##            5 651 183  30 449  77 479 269
##            6 640 259  19 549 549 346 474
##            7 122 138   2 137 446  43 191
##            8  22  12   0  27  97   4  31
##            9   0   0   0   1   4   0   0
```

We can see in the confusion matrix that k-means clustering has not been able to clearly distinguish between different qualities of wine. For example, all of the clusters have significant number of wines rated 5, 6 and 7. There is no clear distinction.

Market Segmentation

First of all I started by cleaning the data set. The data set originally contained 7,882 data points and 36 variables.

As there are lots of spam and pornography bots, I filtered out all the users whose tweet fell into 'spam' and 'adult' category. Then I deleted 'spam' and 'adult' variables form the data set. I also excluded 'uncategorized' and. 'chatter' variables from the data set as they did not seem to provide any insights in my analysis. So, I ended up with 32 variables and 7,309 data points.

In order to identify market segments, I performed cluster analysis. Since the data did not show any kinds of hierarchy, I resorted to K-means cluster analysis. I have used K-means++ .

For using K-means clustering we need to find the optimal number of clusters or the value of K. In order to find the optimal value of K , I have used Elbow plot and CH index.

Figure 2.1 Elbow Plot

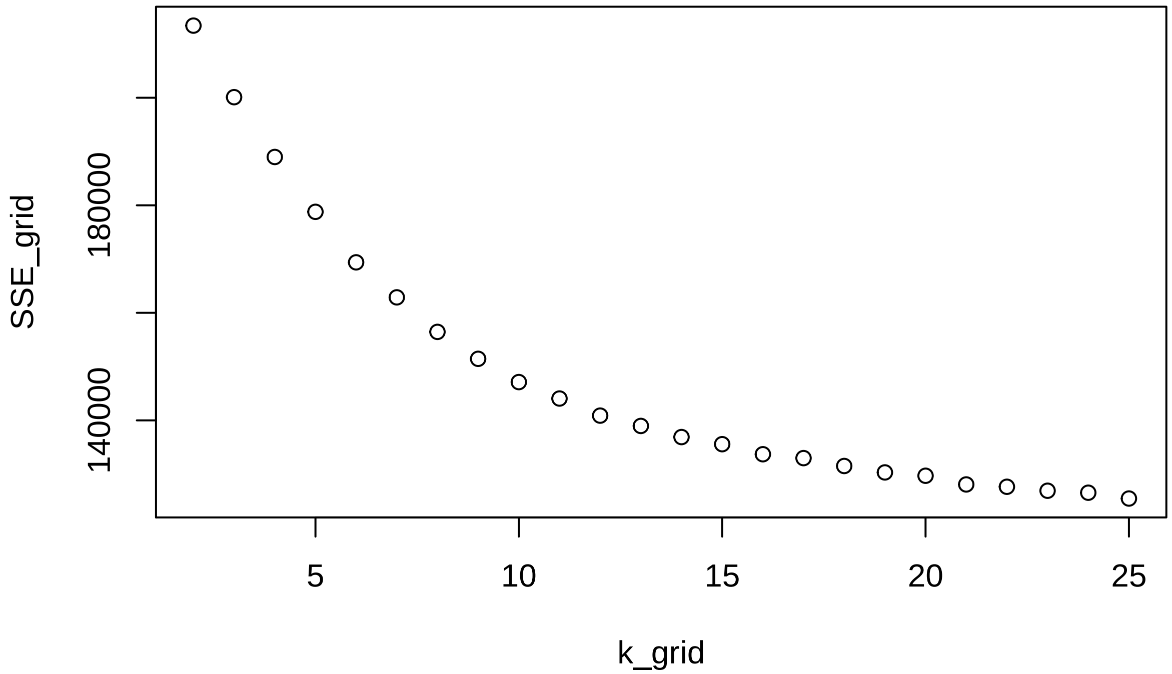
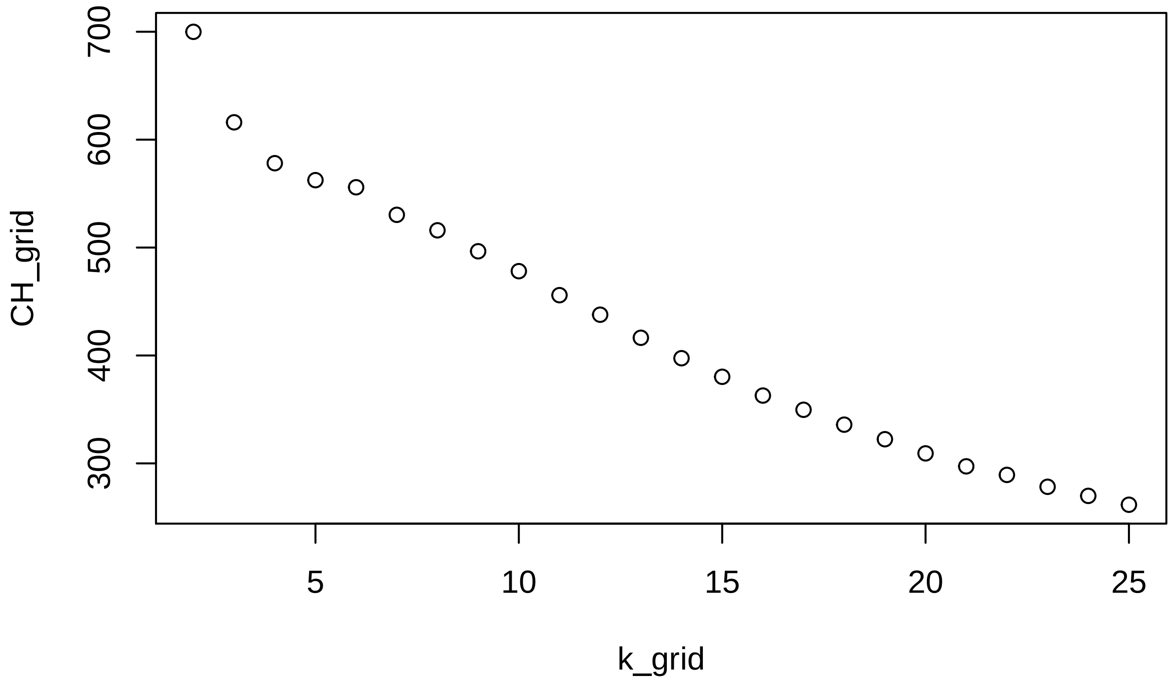
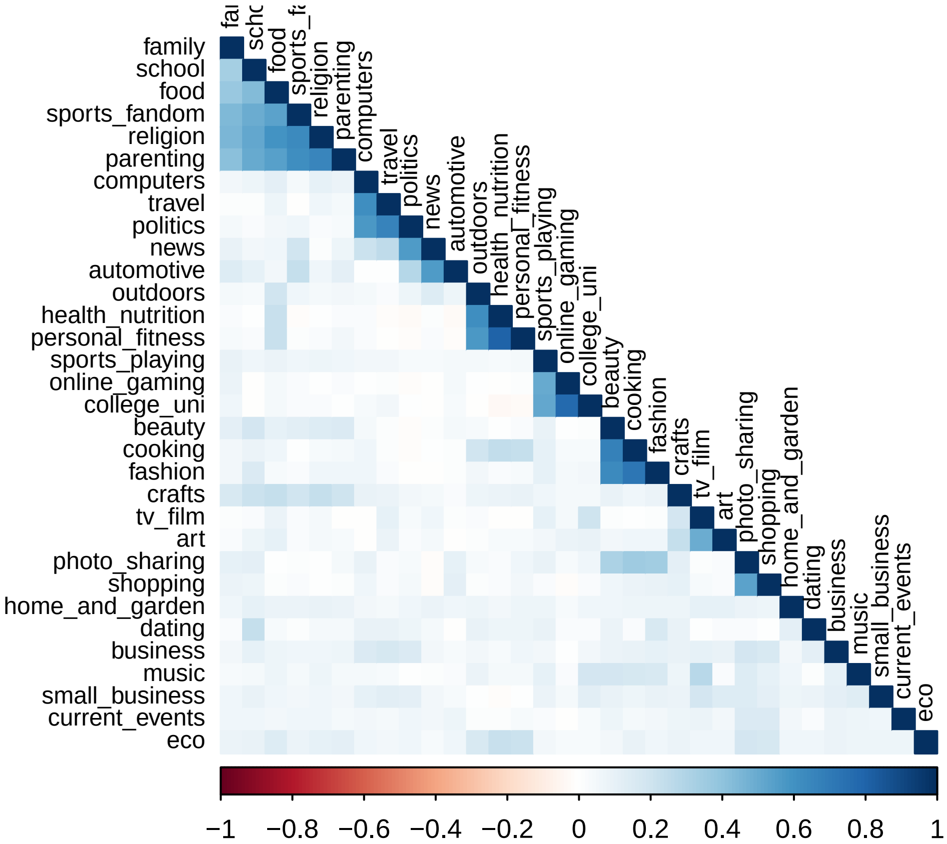


Figure 2.2 CH index plot



The optimal value of K is not clear from the graph. Nevertheless, these plots have hinted towards number 5. For further confirmation, I have plotted a correlogram and have tried to identify possible singularities among the variables.

Figure 2.3 Correlogram



From the correlogram, we can see that some subgroups of the variables are highly correlated with each other. The variables 'family', 'school', 'food', 'sports_fandom', 'religion' seems to be highly correlated. Likewise, 'Computers', 'travel', 'politics', 'news' and 'automotive' are correlated. 'Outdoors', 'health_nutrition' and 'personal_fitness' are correlated with each other. 'Sports_playing', 'online_gaming' and 'college_uni' are also correlated. Lastly, 'beauty', 'cooking' and 'fashion' also seem to have a significant correlation. Therefore, the correlogram corroborates that the optimal value of K is 5.

Summary of cluster 1

```
##    Mode   FALSE    TRUE
## logical    6448     861
```

Summary of Cluster 2

```
##    Mode   FALSE    TRUE
## logical    6739     570
```

Summary of Cluster 3

```
##    Mode   FALSE    TRUE
## logical    6626     683
```

Summary of Cluster 4

```
##    Mode   FALSE    TRUE
## logical    2855    4454
```

Summary of Cluster 5

```
##    Mode   FALSE    TRUE
## logical    6568     741
```

What are the clusters?

```
##   current_events           travel    photo_sharing          tv_film
##        1.5226481        1.2276423        2.7119628        1.0162602
##    sports_fandom         politics             food           family
##        1.1382114        1.2346109        2.0929152        0.7642276
##  home_and_garden            music             news    online_gaming
##        0.6248548        0.7468060        1.1068525        1.1521487
##         shopping health_nutrition      college_uni   sports_playing
##        1.4901278       11.9349593        1.2810685        0.6701510
##          cooking              eco        computers         business
##        3.2775842        0.9024390        0.5528455        0.4680604
##         outdoors           crafts       automotive              art
##        2.6817654        0.6085947        0.6387921        0.7700348
##         religion           beauty        parenting           dating
##        0.7340302        0.4192799        0.7317073        1.0174216
##           school personal_fitness          fashion   small_business
##        0.5563298        6.3925668        0.7793264        0.2868757
```

```
##   current_events           travel    photo_sharing          tv_film
##        1.7596491        1.5017544        6.0526316        1.2122807
##    sports_fandom         politics             food           family
##        1.1298246        1.3807018        1.0526316        0.8964912
##  home_and_garden            music             news    online_gaming
##        0.6245614        1.3140351        1.0175439        1.5280702
##         shopping health_nutrition      college_uni   sports_playing
##        2.1035088        2.1929825        2.1052632        0.9368421
##          cooking              eco        computers         business
##       10.5122807        0.5614035        0.7368421        0.6228070
##         outdoors           crafts       automotive              art
##        0.7929825        0.6368421        0.8859649        0.9807018
##         religion           beauty        parenting           dating
##        0.8315789        3.8175439        0.7385965        0.9105263
##           school personal_fitness          fashion   small_business
##        0.9736842        1.3298246        5.4105263        0.5210526
```

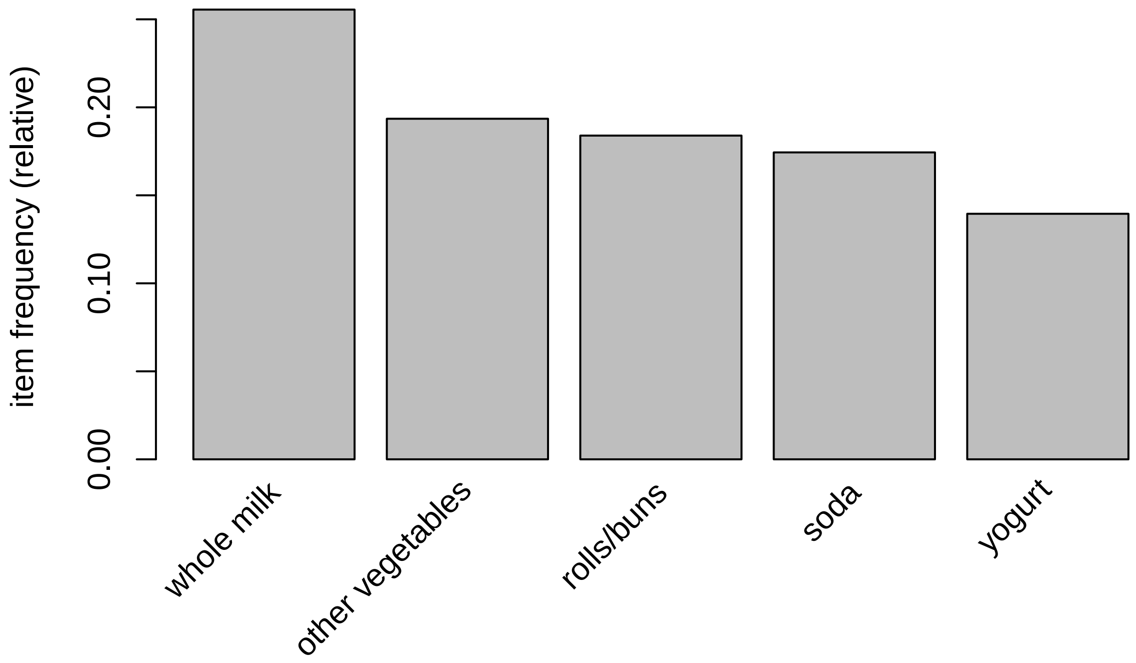
```
##   current_events           travel    photo_sharing          tv_film
##        1.6368960        5.5519766        2.5065886        1.2254758
##    sports_fandom         politics             food           family
##        2.0190337        8.8550512        1.4597365        0.9341142
##  home_and_garden            music             news    online_gaming
##        0.6193265        0.6354319        5.2503660        1.1566618
##         shopping health_nutrition      college_uni   sports_playing 
##        1.3674963        1.6588580        1.6676428        0.7188873 
##          cooking              eco        computers         business 
##        1.2635432        0.5885798        2.4568082        0.6793558 
##         outdoors           crafts       automotive              art 
##        0.9121523        0.6354319        2.3396779        0.7510981 
##         religion           beauty        parenting           dating 
##        1.0263543        0.4597365        0.9311859        1.0614934 
##           school personal_fitness          fashion   small_business 
##        0.7262079        1.0102489        0.6749634        0.4743777 

##   current_events           travel    photo_sharing          tv_film 
##        1.4499326        1.0880108        2.3298159        1.0498428 
##    sports_fandom         politics             food           family 
##        0.9696902        1.0159407        0.7608891        0.5875617 
##  home_and_garden            music             news    online_gaming 
##        0.4384823        0.5913785        0.6755725        1.1632241 
##         shopping health_nutrition      college_uni   sports_playing 
##        1.2887292        1.0781320        1.5422093        0.5646610 
##          cooking              eco        computers         business 
##        0.8547373        0.3776381        0.3614728        0.3414908 
##         outdoors           crafts       automotive              art 
##        0.3834755        0.3643916        0.5756623        0.6349349 
##         religion           beauty        parenting           dating 
##        0.5246969        0.3354288        0.4492591        0.5534351 
##           school personal_fitness          fashion   small_business 
##        0.4573417        0.6409969        0.5229008        0.2748092 

##   current_events           travel    photo_sharing          tv_film 
##        1.6518219        1.2955466        2.6545209        1.1106613 
##    sports_fandom         politics             food           family 
##        5.8475034        1.1160594        4.5087719        2.4601889 
##  home_and_garden            music             news    online_gaming 
##        0.6599190        0.7462888        1.0202429        1.2793522 
##         shopping health_nutrition      college_uni   sports_playing 
##        1.4844804        1.8771930        1.5236167        0.8002699 
##          cooking              eco        computers         business 
##        1.6261808        0.6423752        0.7152497        0.4885290 
##         outdoors           crafts       automotive              art 
##        0.6990553        1.0634278        1.0256410        0.8839406 
##         religion           beauty        parenting           dating 
##        5.1916329        1.0877193        4.0013495        0.8299595 
##           school personal_fitness          fashion   small_business 
##        2.6626181        1.1848853        1.0377868        0.3967611 
```

After running the k-means++ with K=5, I checked the number of data points in each cluster. The cluster with highest number of data points had 4454 data points. That is about 60 percent of the total number of data points I considered in the analysis. This cluster includes the people who have tweeted less than 2 times on an average in all the categories. This could mean that most of the followers of "NutrientH20" are not active users of "twitter" or social media in general. Despite not being active users of social media, these people are following "Nutrient20" which means that the current social media marketing strategy is working quite well.

The cluster with the lowest number of people, on an average tweeted more about 'photo sharing', 'cooking', and 'fashion'. Therefore, in order to attract and appeal to more of the people who are more interested in photo sharing, cooking and fashion the company should position their brand in a way that it seems like it is related to photo sharing or cooking or fashion.

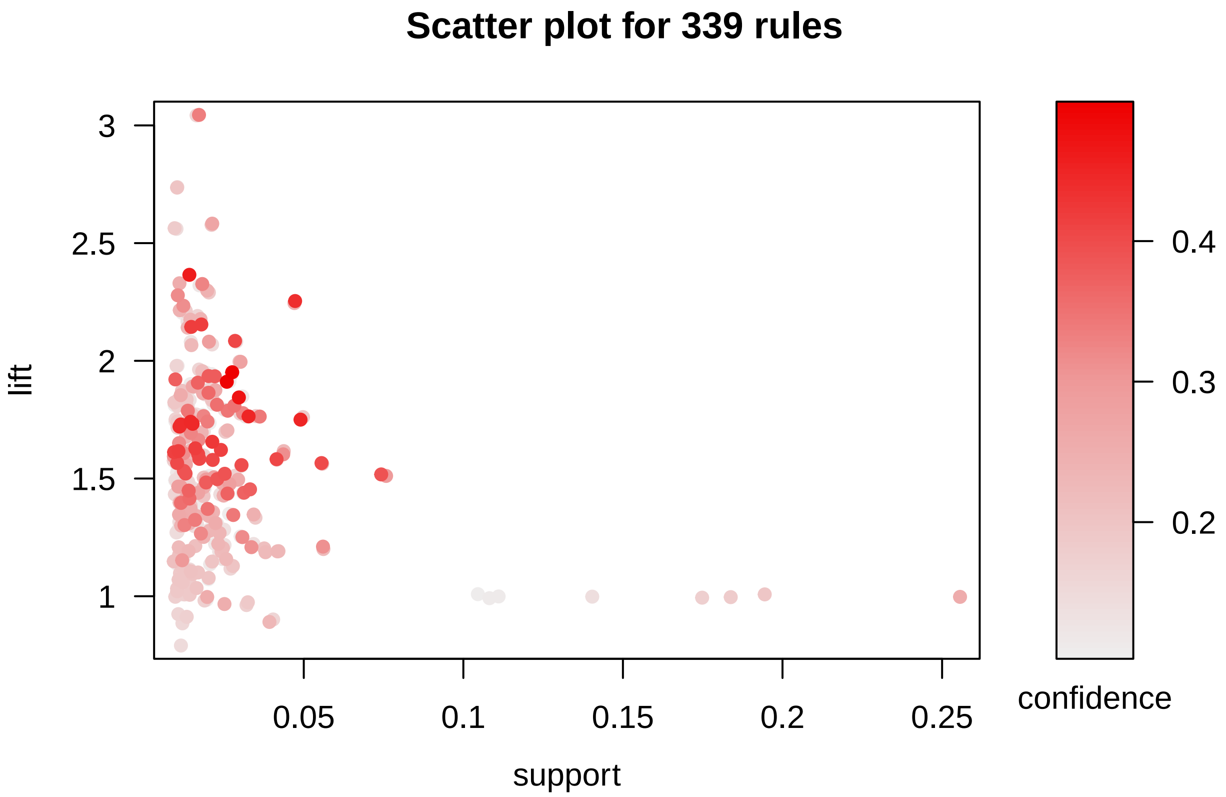
Association rules for grocery purchases

Figure 3.1 Top 5 items with highest support



In order to find various association rules I used 'apriori' function with support = 0.01, confidence= 0.1 and maximum length of 2. By doing so, I got a set of 339 rules. Then, I checked the items with highest support. As seen in the figure, top 5 items with highest support are whole milk, other vegetables, rolls/buns, soda and yogurt.

Figure 3.2 Plot of Association rules



In order to find strong associations, we need the association rules with high lift and high confidence. For picking up thresholds for lift and confidence, I made a plot of the association rules. As we can see in the figure 3.2, we do not have lot of points above the confidence of 0.3 and above the lift of 2. Therefore, I used threshold of 0.3 for the confidence and 2 for the lift. This give 9 association rules.

```
##     lhs                    rhs                  support    confidence coverage
## [1] {onions}            => {other vegetables} 0.01423488 0.4590164  0.03101169
## [2] {berries}           => {yogurt}           0.01057448 0.3180428  0.03324860
## [3] {hamburger meat}    => {other vegetables} 0.01382816 0.4159021  0.03324860
## [4] {cream cheese}      => {yogurt}           0.01240468 0.3128205  0.03965430
## [5] {chicken}           => {other vegetables} 0.01789527 0.4170616  0.04290798
## [6] {beef}              => {root vegetables}  0.01738688 0.3313953  0.05246568
## [7] {curd}              => {yogurt}           0.01728521 0.3244275  0.05327911
## [8] {whipped/sour cream} => {other vegetables} 0.02887646 0.4028369  0.07168277
## [9] {root vegetables}   => {other vegetables} 0.04738180 0.4347015  0.10899847
##     lift     count
## [1] 2.372268 140
## [2] 2.279848 104
## [3] 2.149447 136
## [4] 2.242412 122
## [5] 2.155439 176
## [6] 3.040367 171
## [7] 2.325615 170
## [8] 2.081924 284
## [9] 2.246605 466
```

We can see that almost all of associations that make sense. The first association in table shows that onions implies other vegetables. These two items definitely go together. Beef and root vegetables are also consumed together. Likewise, the association between hamburger meat and other vegetable also makes right sense.

Figure 3.3 Groceries association rules

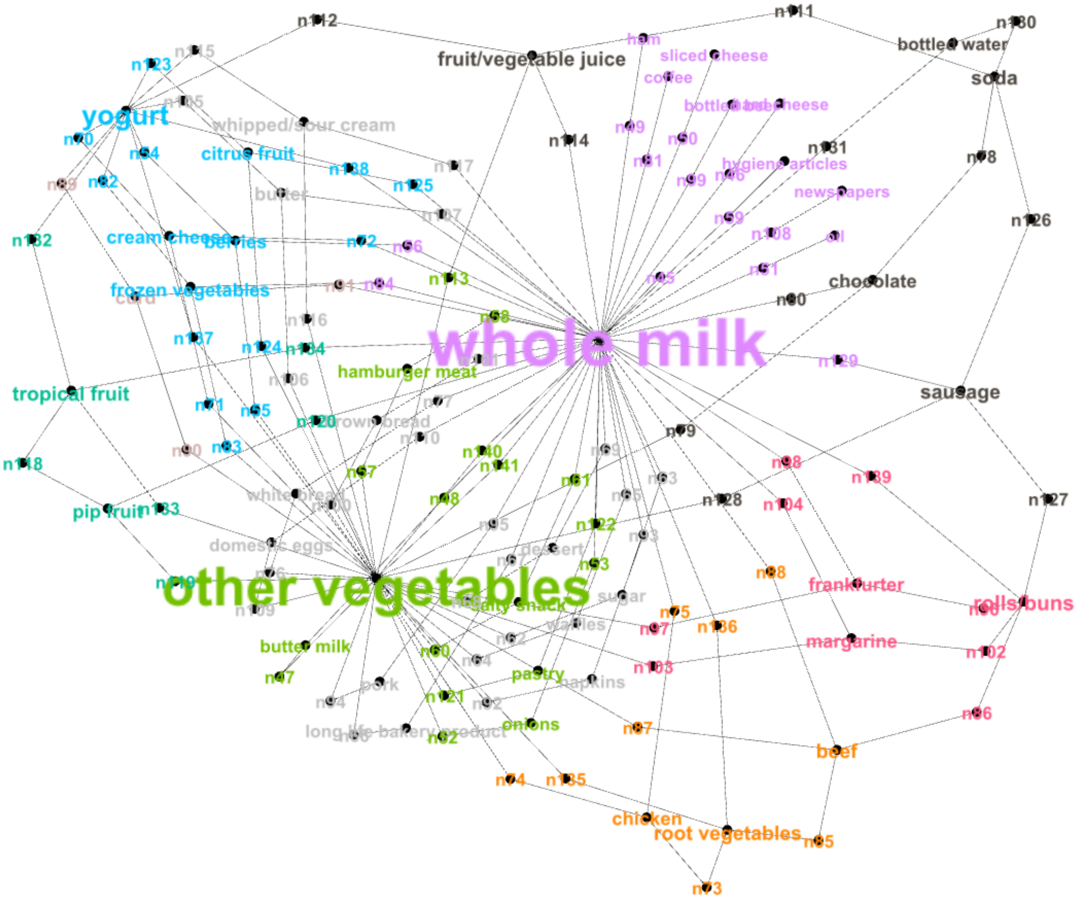


Figure 1: Groceries rules

The figure 3.3 can help us visualize the item sets. We can see other vegetables, butter milk, hamburger meat, onions form a set, which does make sense. Likewise, items like margarine, roll buns and frankfurter which are consumed together make a set. Similarly, we can see that items like citrus fruit, yogurt and cream cheese are grouped together. From an association point of view, it also makes perfect sense.

Author Attribution

```
## <<DocumentTermMatrix (documents: 2500, terms: 32241)>>
## Non-/sparse entries: 473695/80128805
## Sparsity           : 99%
## Maximal term length: 49
## Weighting          : term frequency (tf)
```

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

I started by making a corpus containing all the 2500 documents by 50 authors from the training directory, after which I did some pre-processing and tokenization. I converted everything to lower case, removed numbers, removed punctuations, striped excess white-space, and removed stop words. After tokenization I created a document term matrix which contain 2500 rows, each row representing a document, and 32241 columns, each of which represent a term. I then deleted all the sparse terms. I removed all those terms with zero countsin in more that 95% of the documents, after which I was left with 660 terms. I then created matrix of TF-IDF weighs using the document term matrix.

```
## Importance of first k=15 (out of 644) components:
##                            PC1     PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     3.61251 2.90050 2.59588 2.56025 2.43294 2.40581 2.23539
## Proportion of Variance 0.02026 0.01306 0.01046 0.01018 0.00919 0.00899 0.00776
## Cumulative Proportion  0.02026 0.03333 0.04379 0.05397 0.06316 0.07215 0.07991
##                            PC8     PC9    PC10    PC11    PC12    PC13    PC14
## Standard deviation     2.17112 2.14298 2.07460 2.04517 2.01322 1.89647 1.89484
## Proportion of Variance 0.00732 0.00713 0.00668 0.00649 0.00629 0.00558 0.00558
## Cumulative Proportion  0.08723 0.09436 0.10104 0.10754 0.11383 0.11941 0.12499
##                           PC15
## Standard deviation     1.85654
## Proportion of Variance 0.00535
## Cumulative Proportion  0.13034
```

In the next step, I scrubbed off all the columns with entries equals to zero, after which I was left with 644 columns. Then, I conducted the Principal Component Analysis (PCA) with rank 15. 15 principal components were able to capture 13% variation of the data. After conducting the PCA, I created a new column named 'author' and entered author names corresponding to their respective documents.

After dimensionality reduction using PCA, I conducted multinomial logistic regression, k-nearest neighbors regression and random forest model with 15 principal components for prediction of author of a particular document.

After building the models using training set, I created a corpus from the testing set. The testing data also contained 50 authors and 50 documents per author totaling to 2500 documents. I repeated the tokenization steps and conducted other pre-processing step which I did with the training data, and created a document term matrix. Like before, I removed the terms that had count of zero in more than 95% of the documents and constructed a matrix of TF-IDF weights from document term matrix. In the next step, I scrubbed off all the columns with entries equals to zero after which I was left with 660 columns. Like for the training set, I conducted the Principal Component Analysis (PCA) with rank 15. 15 principal components were able to capture 13% variation of the data for the testing set as well. After conducting the PCA, I created a new column named 'author' and entered author names corresponding to their respective documents.

```
## <<DocumentTermMatrix (documents: 2500, terms: 33048)>>
## Non-/sparse entries: 480577/82139423
## Sparsity           : 99%
## Maximal term length: 48
## Weighting          : term frequency (tf)
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   0.000   5.783   7.671   8.602  10.244  42.143
```

```
## Importance of first k=15 (out of 660) components:
##                            PC1     PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     3.70364 2.92879 2.63165 2.57208 2.44603 2.37248 2.24411
## Proportion of Variance 0.02078 0.01300 0.01049 0.01002 0.00907 0.00853 0.00763
## Cumulative Proportion  0.02078 0.03378 0.04427 0.05430 0.06336 0.07189 0.07952
##                            PC8     PC9    PC10   PC11    PC12    PC13    PC14
## Standard deviation     2.20491 2.17896 2.09630 2.0391 2.01742 1.96086 1.95260
## Proportion of Variance 0.00737 0.00719 0.00666 0.0063 0.00617 0.00583 0.00578
## Cumulative Proportion  0.08689 0.09408 0.10074 0.1070 0.11321 0.11903 0.12481
##                          PC15
## Standard deviation     1.8709
## Proportion of Variance 0.0053
## Cumulative Proportion  0.1301
```

Prediction accuracy of Multinomial logistic regression

```
## [1] 0.0272
```

Prediction accuracy of k-nearest neighbours

```
## [1] 0.0052
```

Prediction accuracy of randomforest

```
## [1] 0.0344
```

In the next step, I made predictions on the test set using the models which I built using training data. I created confusion matrices for each of the 3 models and calculated accuracy rates. The highest level of accuracy that I could obtain was just 3.5 percetn using random forest