SDS 323: Exercises 3

Aaron Grubbs Kaushik Koirala Khue Tran Matthew Tran

Problem 1: Predictive Model Building

Problem Overview:

Given the data on commercial rental properties, this report aims to generate the best predictive model for price. Using the generated predictive model, this report will additionally aim to determine the change in rental income per square foot given a building's green certification while holding the other features of the building constant.

Data Analysis and Process

In order to create the best model, this report aimed at starting with a simple intuitive linear model and then doing a stepwise selection process to select the proper features and interaction relations between the features.

Using the green_rating variable for the green certification requirement makes creating a predictive model simpler instead of dealing with the 2 categories. The other variables were then separated by their significance in predicting rent when paired with green_rating. Afterwards the variables were ranked in order of r2 to establish the best variables to use in the stepwise process of model improvement. The cluster rent category was removed since that data is based on the rent data. Amenities and Electricity costs both had r2 values in the tenths place and subsequently improved the model based on the decrease in rsme value. Systematically other variables were added and removed, but no other variables were able to reduce the rsme value further.

In code that looks like this:

```
lm_medium = lm(Rent ~ green_rating + amenities + Electricity_Costs, data=greenbuildings)
```

For the stepwise function other variables were included and experimented with such as size and leasing rate. In code that looks like this:

```
lm_step = step(lm_medium, scope=~(. + cd_total_07 + size + class_a + leasing_rate +
class_b)^3)
```

Results

Here is the output of the stepwise selection:

```
## lm(formula = Rent ~ green_rating + amenities + Electricity_Costs +
##     cd_total_07 + class_a + leasing_rate + size + class_b + Electricity_Costs:cd_total_07 +
##     Electricity_Costs:size + cd_total_07:size + leasing_rate:size +
##     amenities:class_b + Electricity_Costs:class_a + class_a:size +
##     amenities:cd_total_07 + Electricity_Costs:leasing_rate +
##     amenities:leasing_rate + green_rating:size + size:class_b +
##     leasing_rate:class_b + Electricity_Costs:class_a:size + Electricity_Costs:leasing_rate:size +
##     amenities:leasing_rate:class_b, data = greenbuildings)
```

Ultimately, it selected 25 variables for use in the optimized model.

Here are the RMSE values for the original linear model (on the left) and the output of the stepwise selection (on the right), calculated across 100 different train test splits.

```
##       V1       V2
## 16.44361 17.68858
```

Conclusion

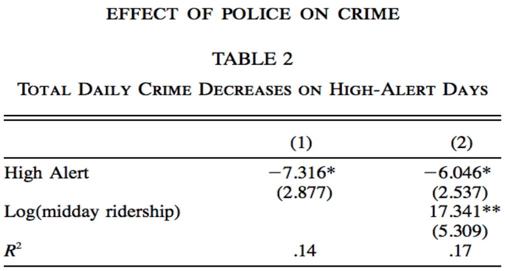
The initial simpler model had the lower RMSE than the result of the stepwise feature selection. It seemed that adding more features and interactions worsened the model and its accuracy as the first three variables were sufficient enough. The coefficients of the first simple model are:

```
##       (Intercept)      green_rating         amenities Electricity_Costs
##         3.8199654         0.4054184         3.5906206       735.3183503
```

Interpreting the green_rating coefficient, "green" certified properties seemed to improve rent by around 0.4054184 dollars per square foot.

Problem 2: What causes what?

1. The question that the researchers were looking at was whether increasing the number of police would reduce the rate of crime in a given city. Intuitively, it might make sense to approach the problem as an independent probability by varying the number of cops and observing fluctuations in crime rates. However, an experimental construct that randomly changes the number of cops on random days is not practical. Another problem arises from the the fact that crime rates also affect the number of police. So places with higher crime rates would naturally have more cops on the street at a given time, making it difficult to see the isolated effect of just increasing cops. A regression model that takes in crime rate and number of cops would not be able to account for such interactions.



EFFECT OF POLICE ON CRIME

TABLE 2

Total Daily Crime Decreases on High-Alert Days

	(1)	(2)
High Alert	−7.316* (2.877)	−6.046* (2.537)
Log(midday ridership)		17.341** (5.309)
R^2	.14	.17

Figure 1: The dependent variable is the daily total number of crimes in D.C. This table present the estimated coefficients and their standard errors in parenthesis. The first column refers to a model where the only variable used in the High Alert dummy whereas the model in column (2) controls form the METRO ridership. * refers to a significant coeficient at the 5% level, ** at the 1% level.

2. So the best setting for looking at the correlation between police and crime would control for the positive feedback loop. The researchers accomplished this by collecting data when the terrorist alert system levels are high in D.C., when more cops are put on the street for reasons unrelated to street crime. As summarized in table 2, two models were fitted: the one in the first column only has the dummy variable

High Alert while the model in the second column included a term for ridership. For both models, the High Alert variable have negative coefficients with similar standard errors and is significant at the 5% level. The coefficient from the first model implies that on a high alert day, where there are more cops on the streets, the daily number of crimes in D.C. decreases by around 7. Similarly, the second model coefficients imply that when Metro ridership increases by 10 (on a log scale), about 17 more crimes are committed, and 6 less crimes are committed on a high alert day. So from the results of table 2, increased number of cops on high alert days does decrease the number of crimes. The R-squared value for both models were relative low (0.14 and 0.17 respectively), this could indicate that a linear fit might not be the best way to model the correlation bewteen police and crime, but there was still significant decrease in crime with increased number of cops.

3. It was unknown if the lower crime rates when the threat level was orange was due to the increase in cops on the street or because there was an increase terrorist threat. The reduce crime could be caused by victims and perpetrators being more cautious about being outside during increased terrorist threats. Measuring the Metro ridership measures the general street traffic in DC during those times, which was shown to be relatively unaffected.This was another way to ensure that the correlations observed were just from the number of police.

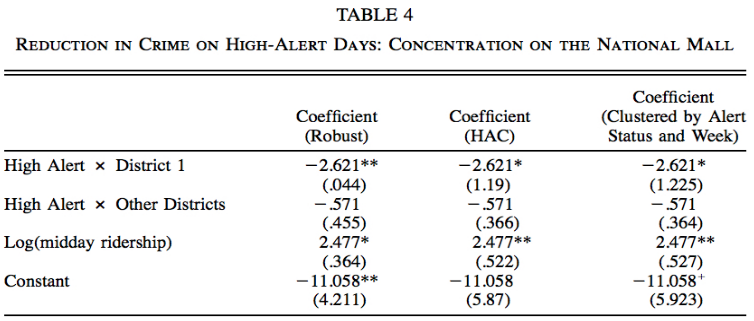


TABLE 4

REDUCTION IN CRIME ON HIGH-ALERT DAYS: CONCENTRATION ON THE NATIONAL MALL

	Coefficient (Robust)	Coefficient (HAC)	Coefficient (Clustered by Alert Status and Week)
High Alert × District 1	−2.621**	−2.621*	−2.621*
	(.044)	(1.19)	(1.225)
High Alert × Other Districts	−.571	−.571	−.571
	(.455)	(.366)	(.364)
Log(midday ridership)	2.477*	2.477**	2.477**
	(.364)	(.522)	(.527)
Constant	−11.058**	−11.058	−11.058^{+}
	(4.211)	(5.87)	(5.923)

Figure 2: The dependent variable is the daily total number of crimes in D.C. District 1 refers to a dummy variable associated with crime incidents in the first police district area. This table present the estimated coefficients and their standard errors in parenthesis.* refers to a significant coeficient at the 5% level, ** at the 1% level.

4. The analysis is further explored in table 4, looking at the effects of dummy variable High Alert and log(ridership) in different districts of D.C. by introducing interaction terms. The model from the first column has negative coefficients for interactions between High Alert and District 1, and High Alert and Other Districts, but only the first was significant. So the effect of increasing the number of police (as seen on high alert days) significantly decreases number of total crimes in District 1. The interaction coefficients tell us that on a high terrorist alert day, about 10 less crimes are commited in District 1 (-7.316 - 2.621) and for other districts, about 7 less crimes are committed. As for the ridership coefficient, the opposite correlation is seen where 2 more crimes are recorded when there are 10 times more rider, consistent with the results of table 2.

Problem 3: Clustering and PCA

Problem Overview:

The data for this problem contains information on 11 different chemical properties of 6500 different bottles of wine, as well as two additional classification variables of color (red or white) and quality (judged on a 1-10 scale).

Data and Analysis Process:

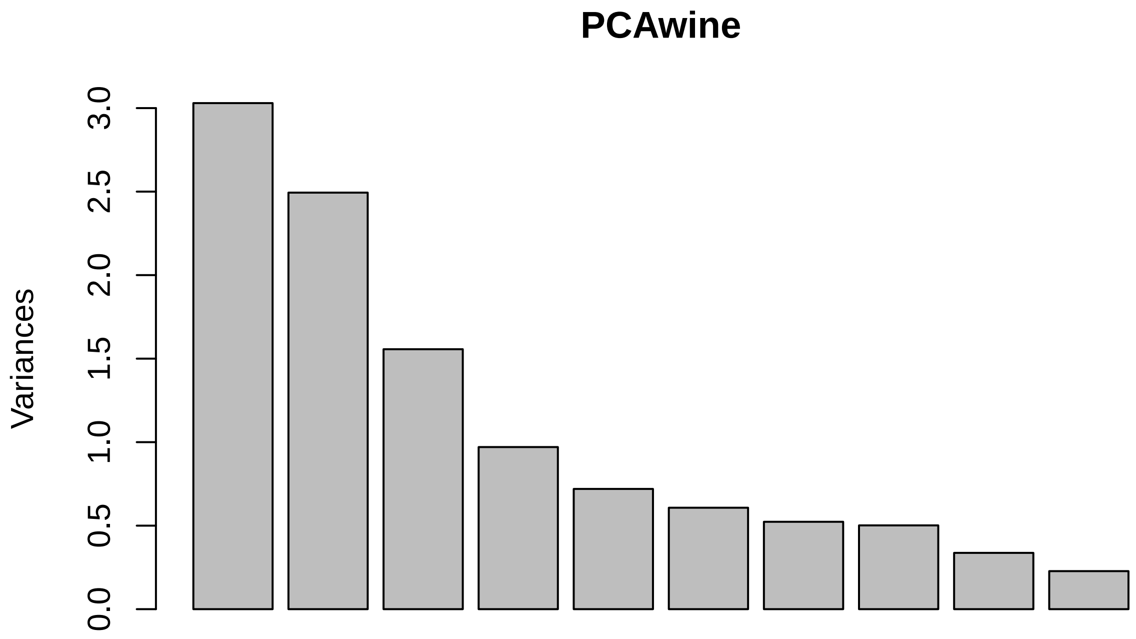
Our goal in this problem is to perform dimensionality reduction on the data: principal components analysis (PCA) and clustering. From there, we must summarize and see if our results can distinguish the red and white wines as well as the quality of the wines from our own intuition.

In order to solve this problem, we first need to perform PCA, look at the loadings and principal components of each summary, and form a conclusion. We also need to perform k-means clustering, and check if the clusters can distinctly distinguish our data in terms of quality and color.

After analysis has been done, we compare our models to see which one makes the most sense to us, and see if either analysis did a better job in distinguishing the reds and whites as well as the quality.

```
wine = read.csv("https://raw.githubusercontent.com/jgscott/SDS323/master/data/wine.csv", header=TRUE)

wine_variables = wine %>% select(-quality, -color)
```

```
PCAwine = prcomp(wine_variables, scale=TRUE)
plot(PCAwine)
```

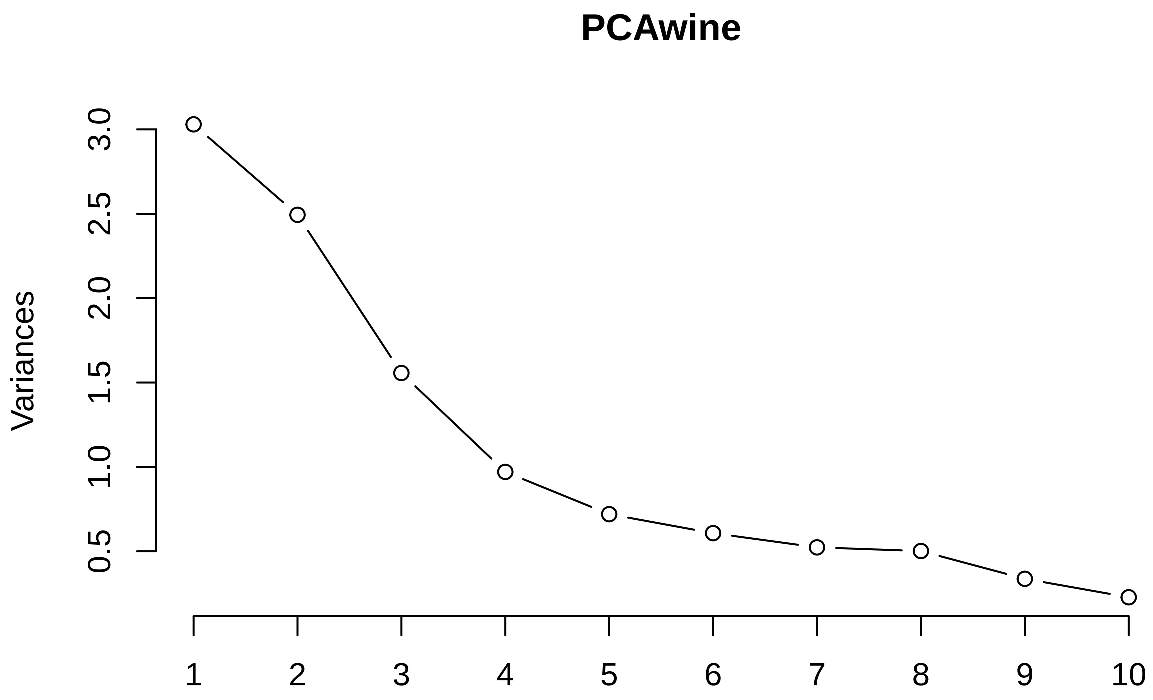


```
round(PCAwine$rotation[,1:3],2)
```

```
##                        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
```

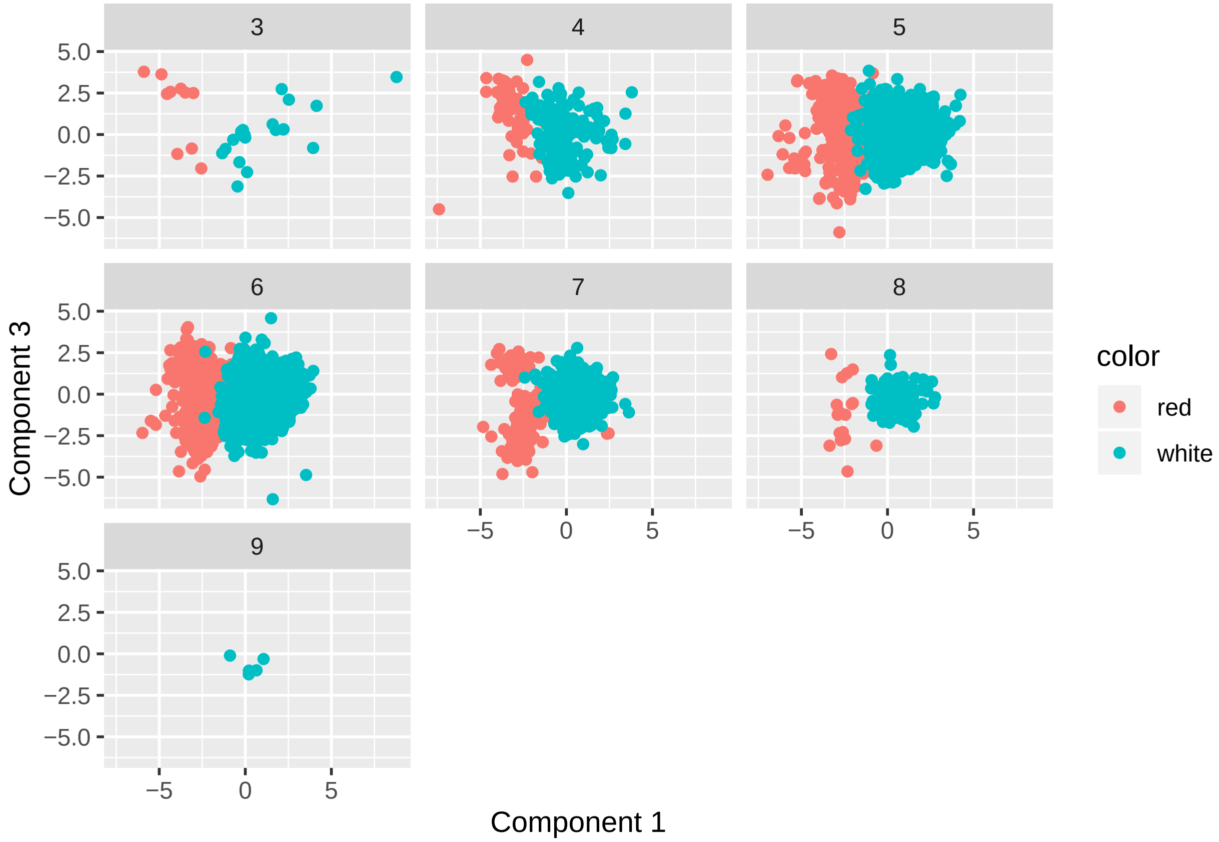
After running PCA, we can see that from PC1 and PC2, the cumulative proportion of variance accounts for already roughly 50.21% of the data.

```
screeplot(PCAwine, npcs = min(10, length(PCAwine$sdev)),type="lines")
```

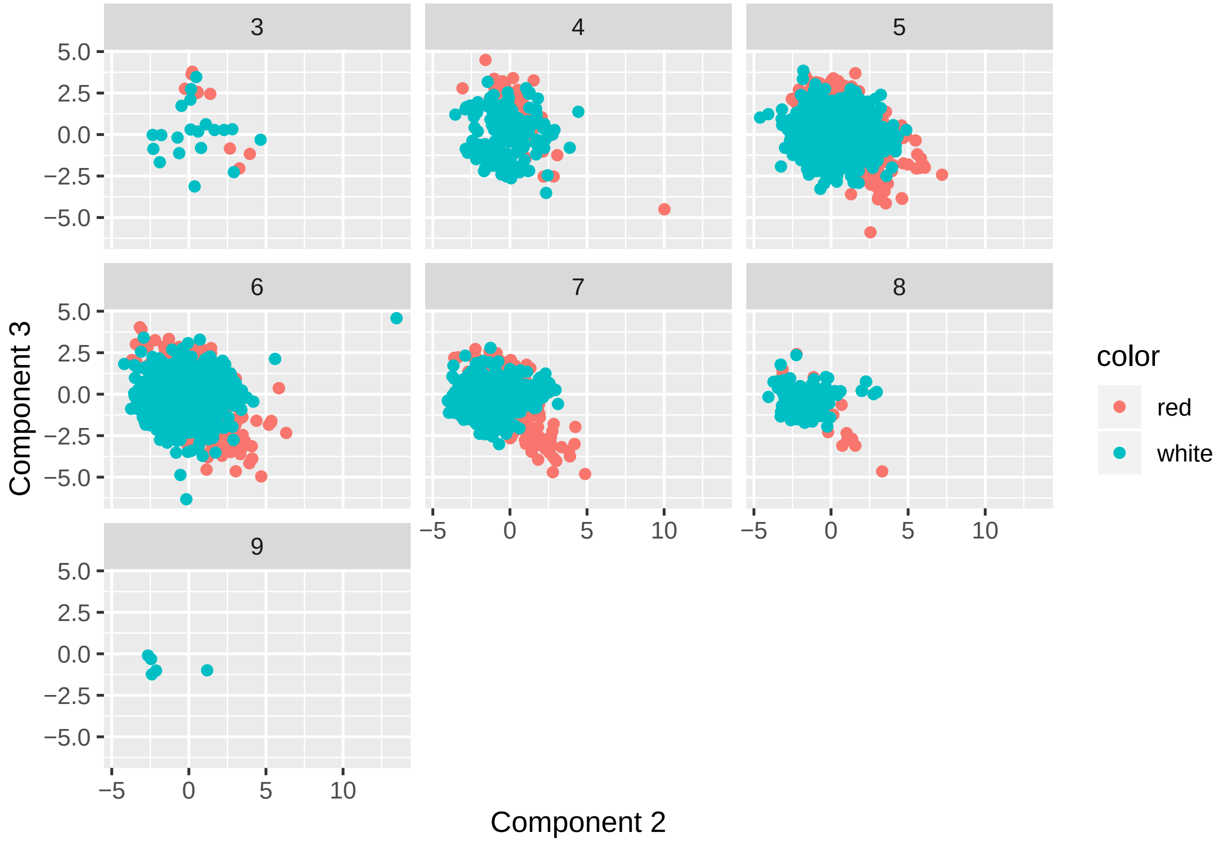


The scree plot entails that our proportion variance does decrease over time as the number of principal components increases.

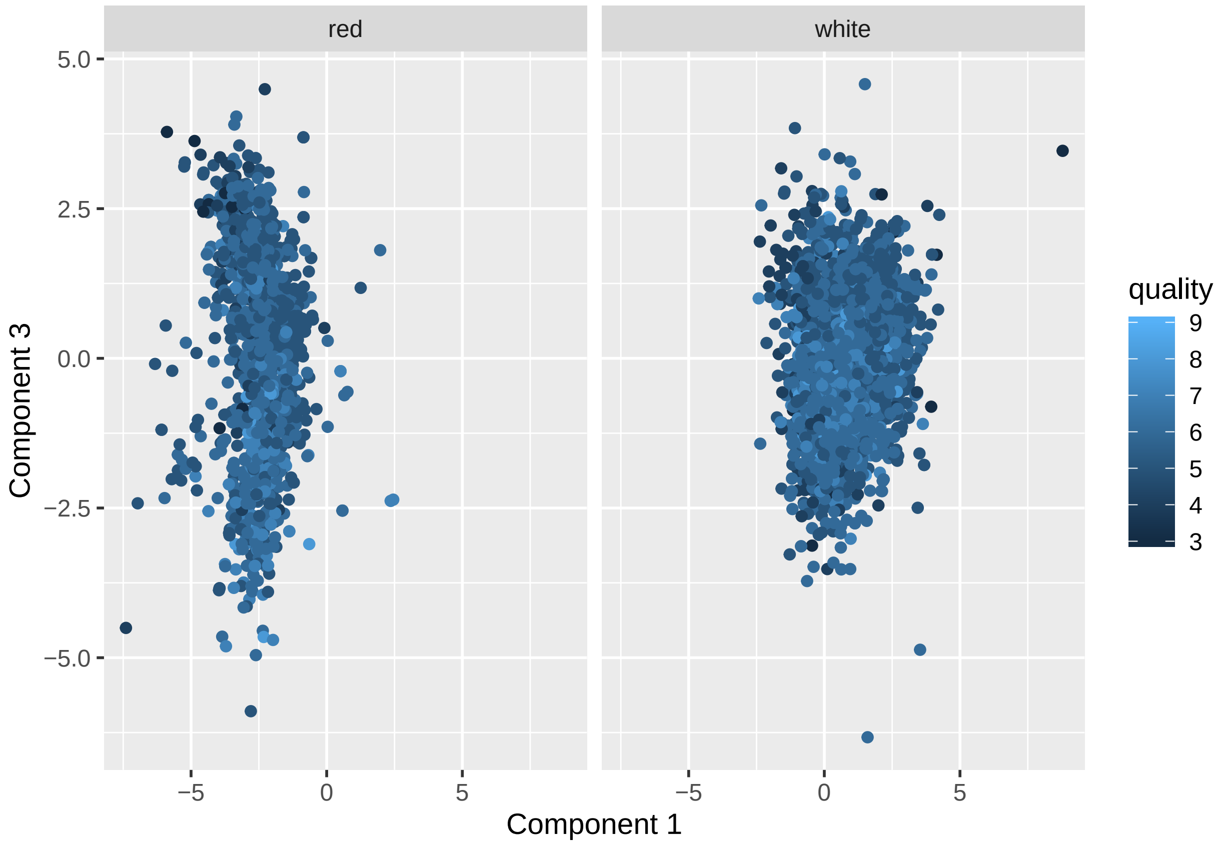
```
wine_scores = PCAwine$x

qplot(wine_scores[,1], wine_scores[,3], color=color, facets=~quality,xlab='Component 1', ylab='Componen
```



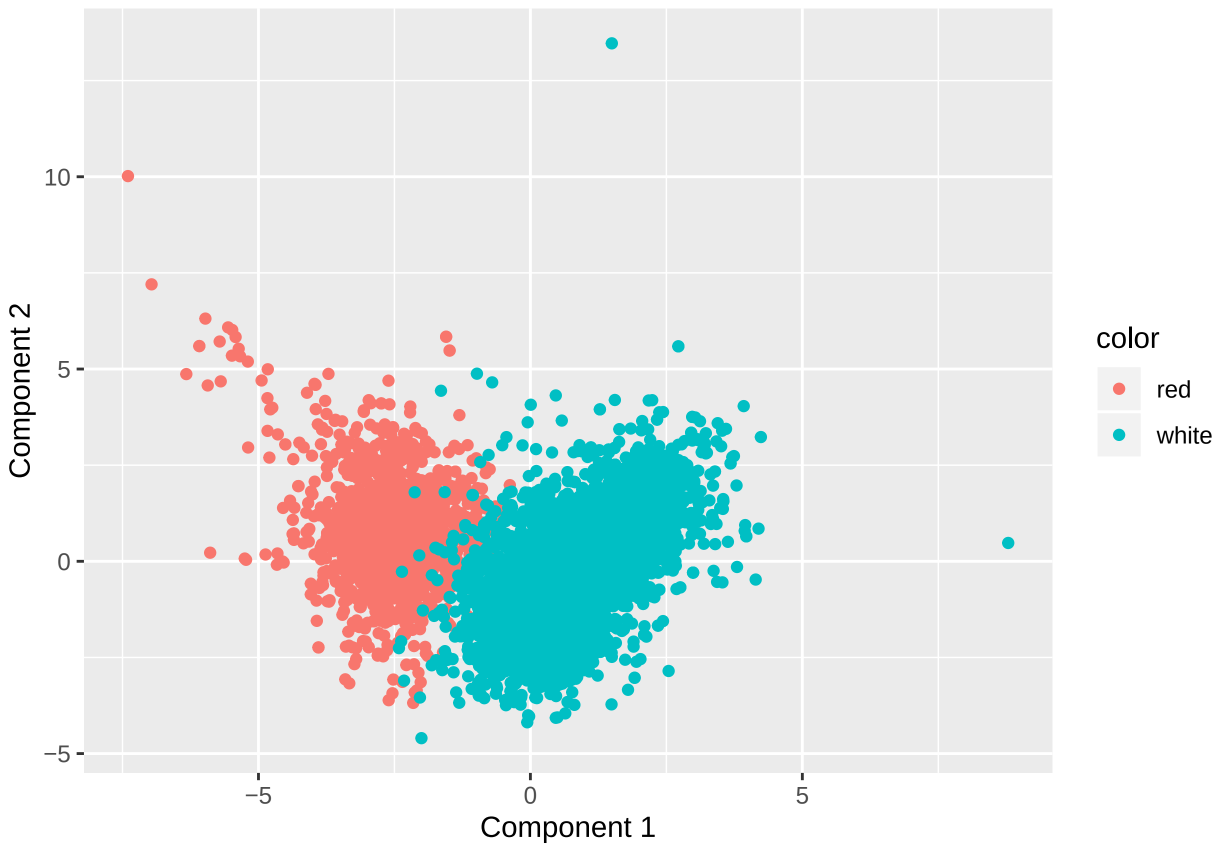
```
qplot(wine_scores[,2], wine_scores[,3], color=color, facets=~quality,xlab='Component 2', ylab='Componen
```



```
qplot(wine_scores[,1], wine_scores[,3], color=quality, facets=~color,xlab='Component 1', ylab='Componen
```



```
qplot(wine_scores[,1], wine_scores[,2], color=color, xlab='Component 1', ylab='Component 2', data=wine)
```



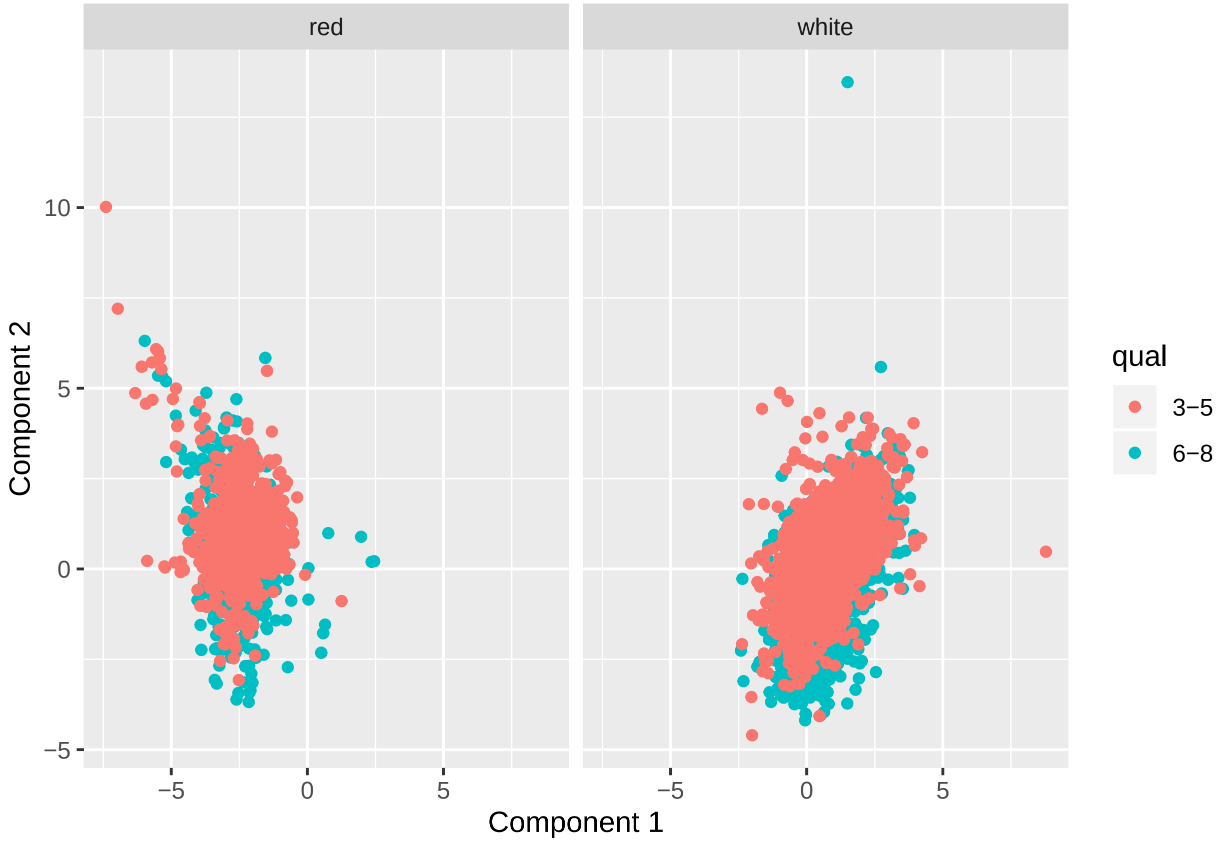
When comparing PC1, PC2, PC3 in the context of our red and white colors, we see that the noise in the data is not distinguishable as the number of principal components increase; they tend to "weave together".

Since the PCA with all 3-8 quality points didn't work well (we could not see clear separation) we try to split quality into 2 groups, 5 and below and above 5 and run another PCA.

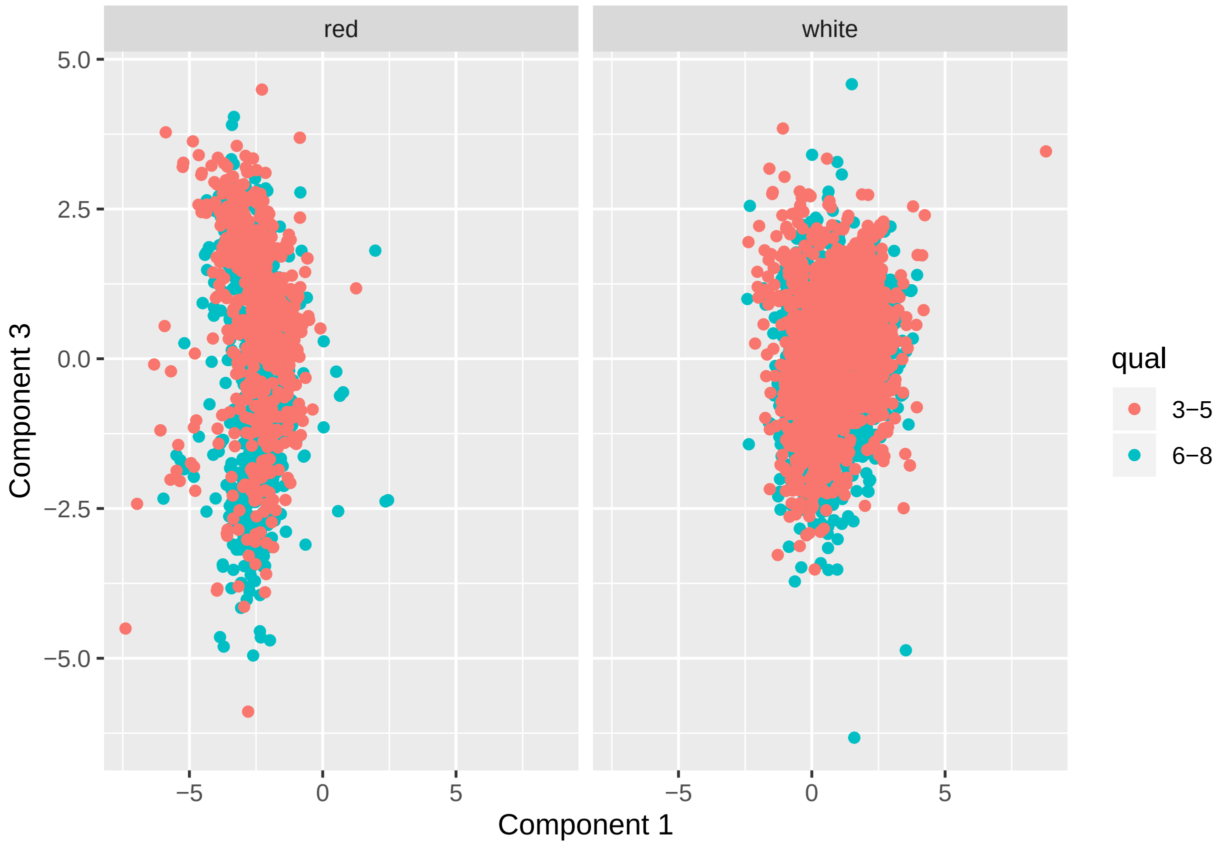
```
high_qual <- wine %>% filter(grepl("6|7|8", quality)) %>% mutate(qual="6-8") %>% select(-quality)
low_qual <- wine %>% filter(grepl("3|4|5", quality)) %>% mutate(qual="3-5") %>% select(-quality)
wine_by_qual <- full_join(high_qual, low_qual)
```

```
## Joining, by = c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar", "chlorides", "
```

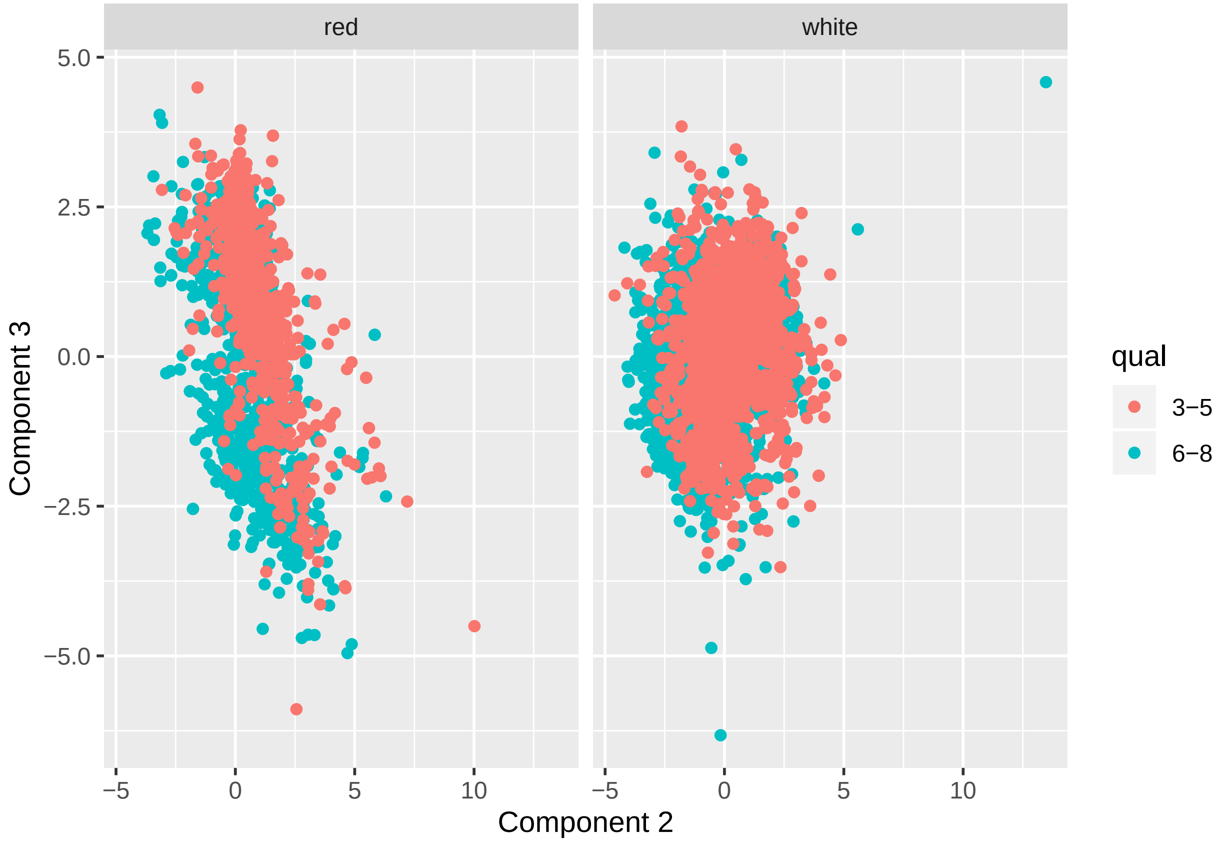
```
wine_var_only = wine_by_qual %>% select(-color, -qual)

PCAwine_qual = prcomp(wine_var_only, scale=TRUE)

qual_scores = PCAwine_qual$x

qplot(qual_scores[,1], qual_scores[,2], color=qual, facets=~color,xlab='Component 1', ylab='Component 2
```



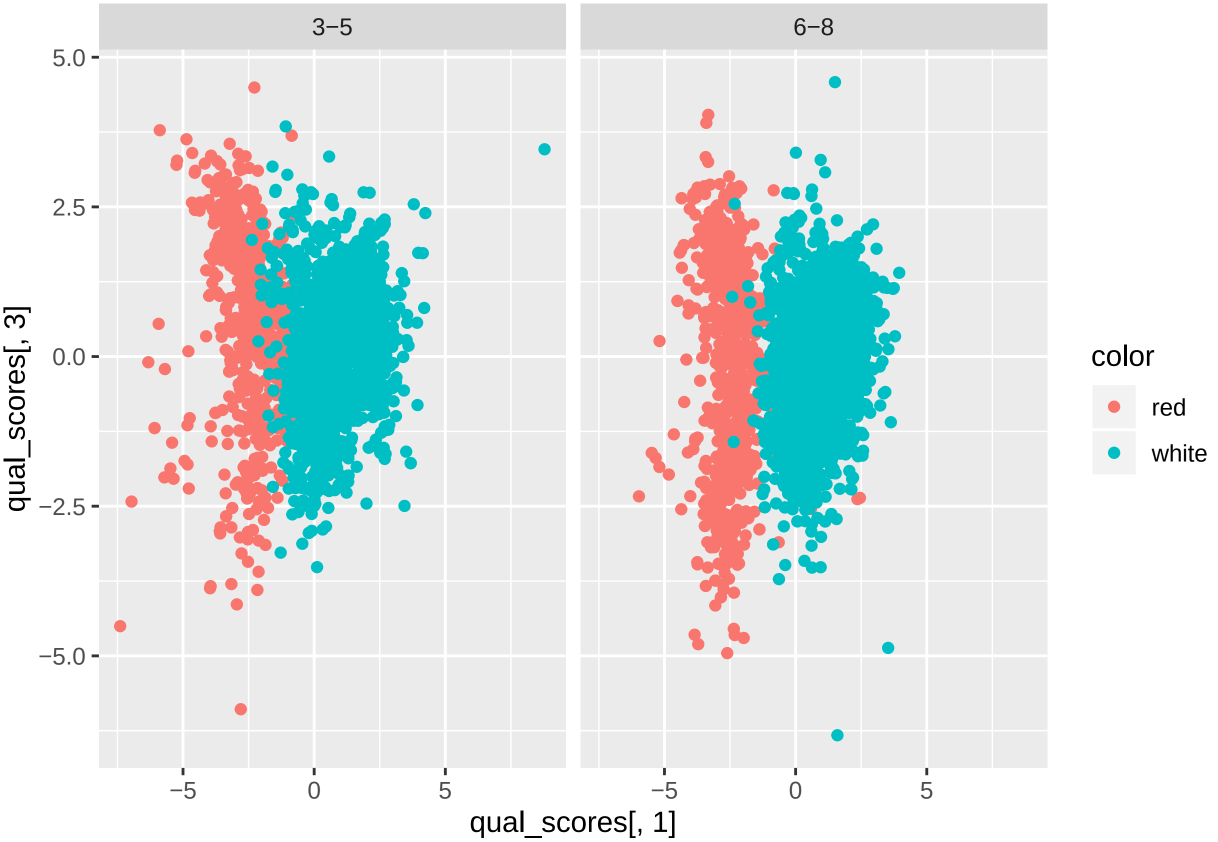
```
qplot(qual_scores[,1], qual_scores[,3], color=qual, facets=~color,xlab='Component 1', ylab='Component 3
```



```
qplot(qual_scores[,2], qual_scores[,3], color=qual, facets=~color,xlab='Component 2', ylab='Component 3
```



```
qplot(qual_scores[,1], qual_scores[,3], color=color, facets=~qual, data=wine_by_qual)
```



We can now see a separation between higher and lower quality, but it is still not clear using the PCA method.

```
loadings = PCAwine$rotation

# 3 most negatively associated variables
loadings[,1] %>% sort %>% head(3)
```

```
## volatile.acidity        sulphates        chlorides
##       -0.3807575       -0.2941352       -0.2901126
```

```
lm1 = lm(quality ~ residual.sugar+free.sulfur.dioxide+total.sulfur.dioxide, data=wine)

# 3 most positively associated variables
loadings[,1] %>% sort %>% tail(3)
```

```
##       residual.sugar  free.sulfur.dioxide total.sulfur.dioxide
##            0.3459199            0.4309140            0.4874181
```

```
lm2 = lm(quality ~ volatile.acidity+sulphates+chlorides, data=wine)
```

Now, we perform clustering to see if it provides us better information about the quality and color variables of the data. This is done using k-means clustering.

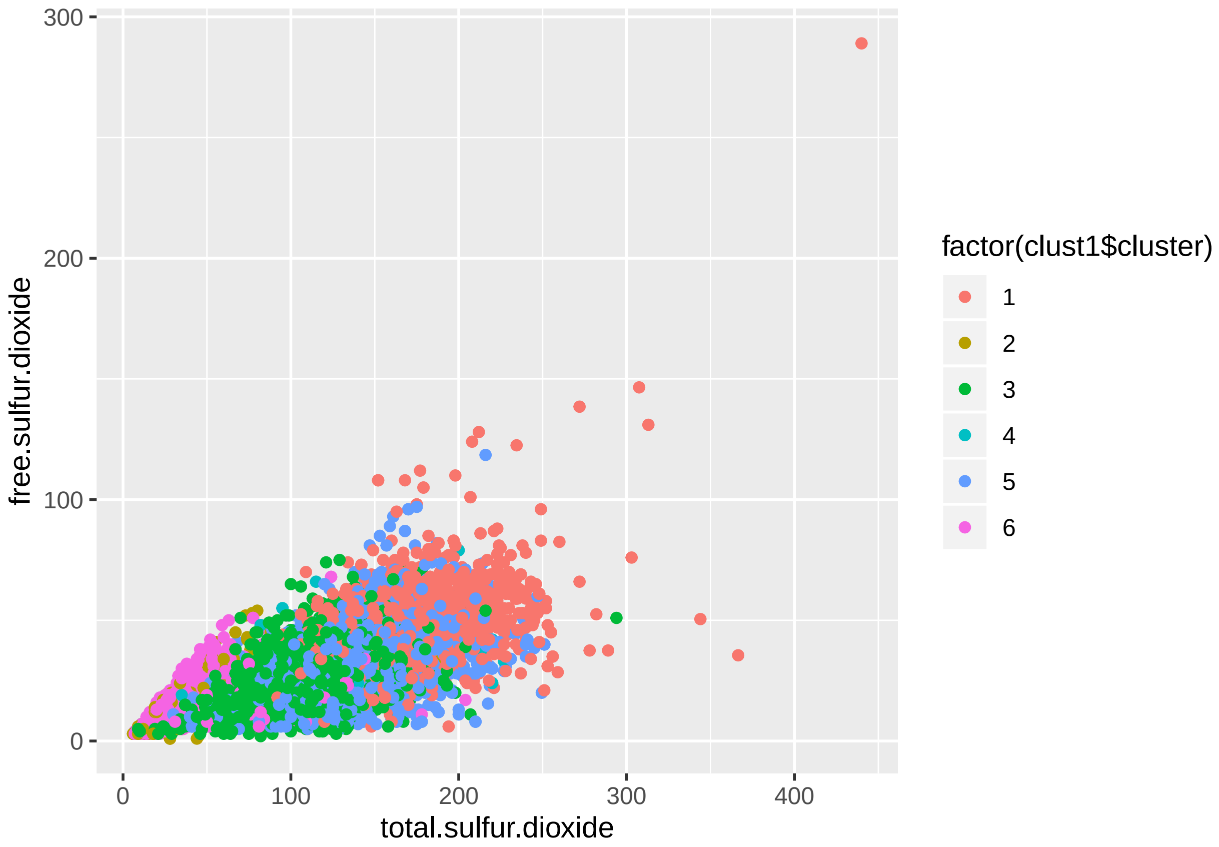
```
library(ggplot2)
#library(LICORS)  # for kmeans++
#library(foreach)
#library(mosaic)

wine = read.csv("https://raw.githubusercontent.com/jgscott/SDS323/master/data/wine.csv", header=TRUE)
```

```r
# Convert variable "color" to a numeric column vector.
# "1" for red and "2" for white
wine$color <- as.numeric(wine$color)

# Center and scale the data
X = wine[,-(12:13)]
X = scale(X, center=TRUE, scale=TRUE)

# Extract the centers and scales from the rescaled data (which are named attributes)
mu = attr(X,"scaled:center")
sigma = attr(X,"scaled:scale")

# Run k-means with 6 clusters and 50 starts
clust1 = kmeans(X, 6, nstart=50)
cluster_number = 1
```

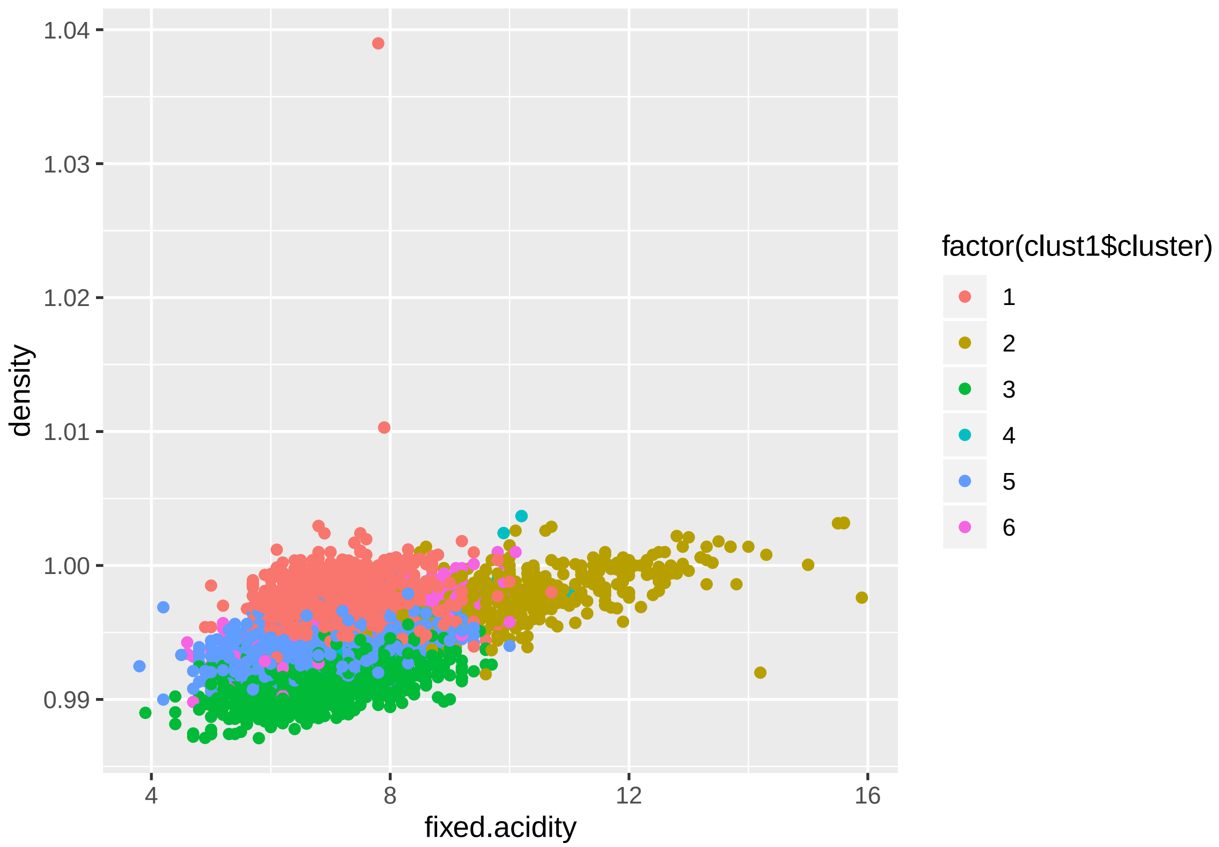
```
##        fixed.acidity     volatile.acidity          citric.acid
##           7.00626598           0.28093670           0.36345908
##       residual.sugar            chlorides  free.sulfur.dioxide
##          12.20198210           0.05005563          46.84047315
## total.sulfur.dioxide              density                   pH
##         171.42519182           0.99733616           3.14004476
##            sulphates              alcohol
##           0.49037084           9.49923274
```

```
## [1] "total.sulfur.dioxide"
```

```
## [1] "free.sulfur.dioxide"
```



```
qplot(fixed.acidity, density, data=wine, color=factor(clust1$cluster))
```



After performing k-means clustering, we see that the data can be distinguished better in terms of clusters.

Conclusion:

From both PCA and k-means clustering, we can see that k-means clustering offers a better visualization of distinguishing the quality and reds/whites in the data since k-means allows us to look at it in clusters.

PCA, on the other hand, does not: it is unable to distinguish the red and white wines and the higher and lower quality wines.

However in term of what “makes sense”, PCA has better interpretability compared to k-means clustering, because it summarize the information in components while k-means summarizes the data in chunks.

Problem 4: Market Segmentation

Problem Overview:

NutrientH2O has collected twitter data on 7882 of their followers. For each of their followers, the data collectors have counted the frequencies the follower will tweet about 36 given topics, such as sports or politics, over a week. Proper analysis of this data could lead to better targeting and marketing for products. This report will aim to identify different groups or types of customers, or segments of the market, that are part of the customer base for Nutrient H2O.

Data and Analysis Process

Following the principle of satisfice and some trial and error, 5 clusters will be identified using the KMEANS clustering algorithm. Much more than 5 will probably be unfeasible for NutrientH20 to carry out targeted ad campaigns for all clustered groups.

Each cluster will represent a segment of the market and a “kind” of customer to target for NutrientH20. What each cluster represents will be determined by the Euclidean coordinates of the centroid (the average point in the cluster). If the coordinates for any one topic are particularly high, it means that the frequency that the average person in that cluster tweets about that topic is also high. Among the topics, the topic of “chatter” has been discarded from the dataset as it is very general and not that useful for marketing purposes.

Results

The centroid coordinates for the first cluster are:

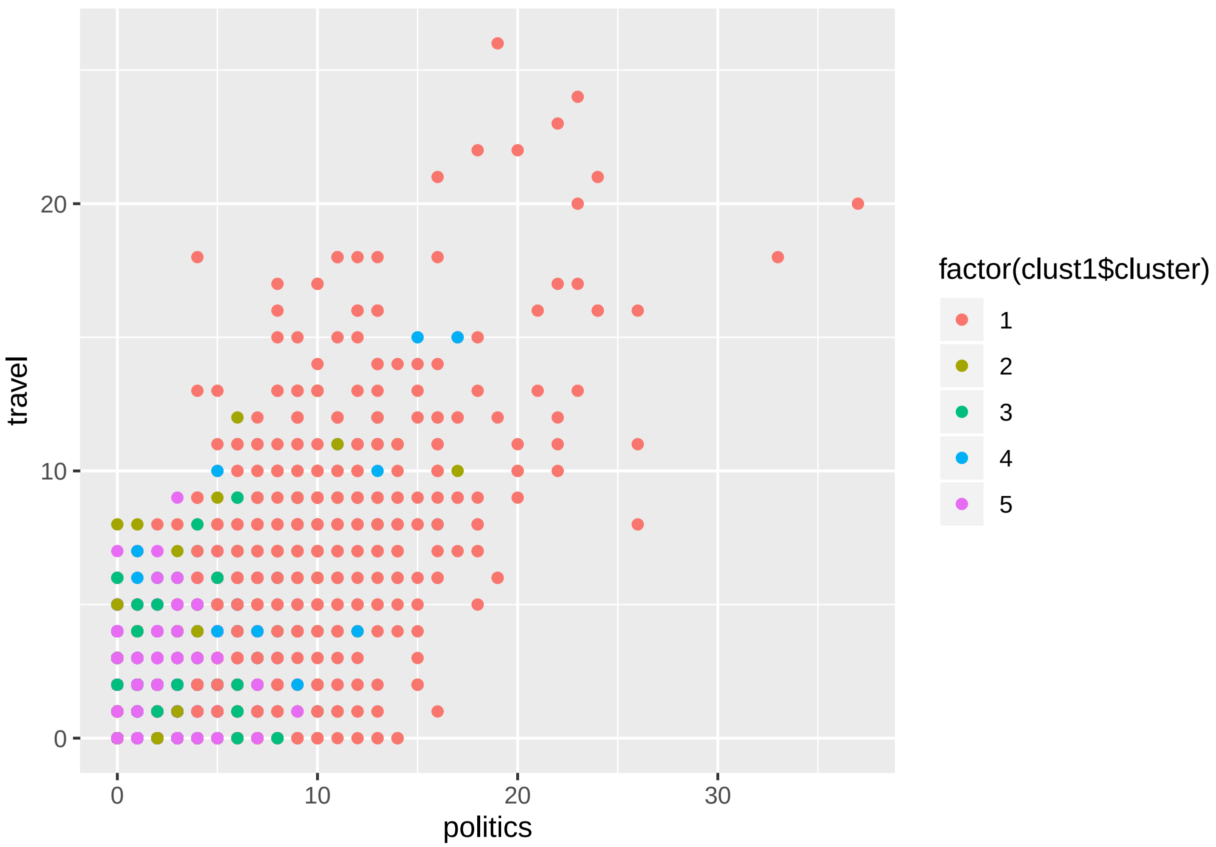
```
##   current_events           travel    photo_sharing    uncategorized 
##      1.647966339      5.569424965      2.517531557      0.789621318 
##          tv_film    sports_fandom         politics             food 
##      1.223001403      1.991584853      8.837307153      1.446002805 
##           family  home_and_garden            music             news 
##      0.918653576      0.614305750      0.636746143      5.204768583 
##    online_gaming         shopping health_nutrition      college_uni 
##      1.189340813      1.378681627      1.663394109      1.685834502 
##   sports_playing          cooking              eco        computers 
##      0.704067321      1.270687237      0.597475456      2.469845722 
##         business         outdoors           crafts       automotive 
##      0.673211781      0.922861150      0.650771388      2.318373072 
##              art         religion           beauty        parenting 
##      0.753155680      1.009817672      0.467040673      0.925666199 
##           dating           school personal_fitness          fashion 
##      1.063113604      0.709677419      1.001402525      0.684431978 
##   small_business             spam            adult 
##      0.478260870      0.008415147      0.270687237
```

The topics that stand out most in this cluster are:

```
## [1] "politics"
```

```
## [1] "travel"
```

The following is a plot of the users based on the frequency with which they tweet about politics and travel. The users are color coded by cluster.



As can be seen from the plot above, the most frequent tweeters on topics politics and travel (top right of the graph) are mostly all in cluster 1.

The centroid coordinates for the second cluster are:

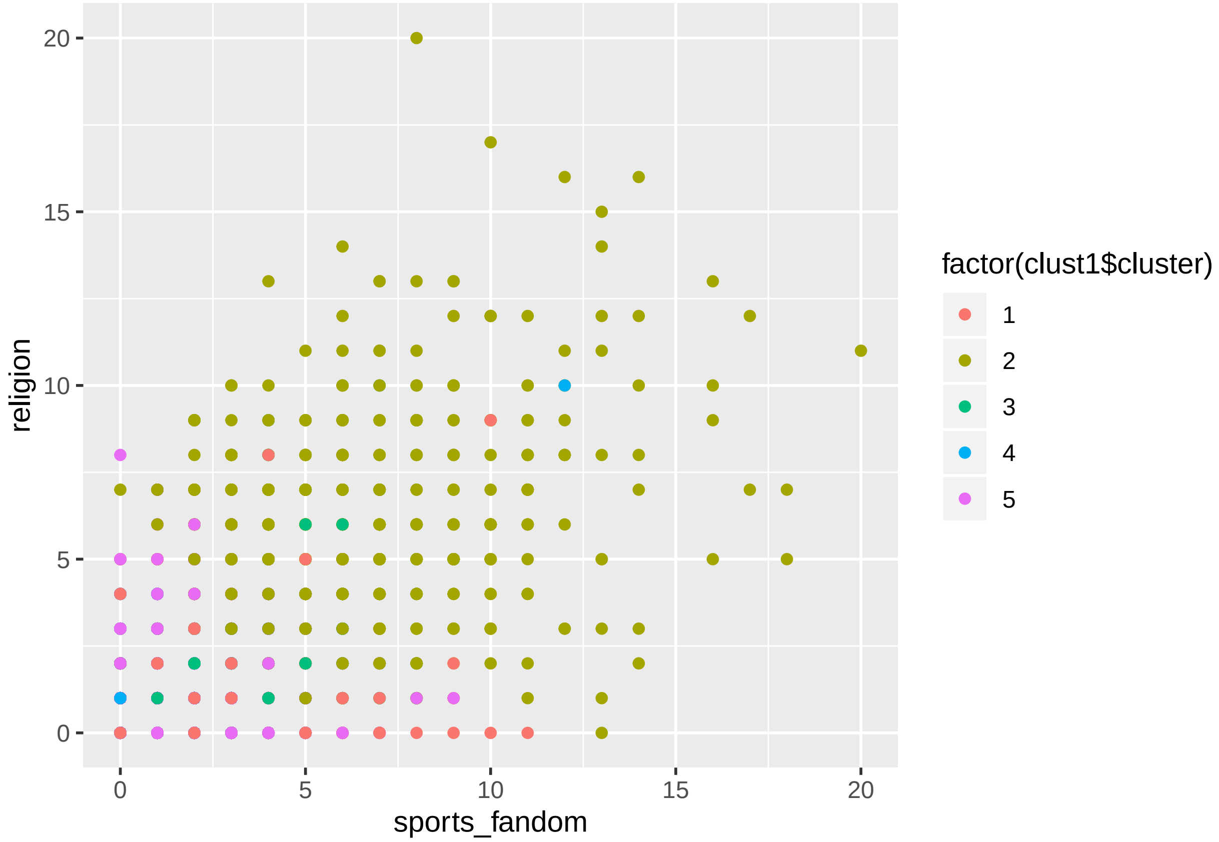
```
##   current_events           travel    photo_sharing    uncategorized
##       1.67598475       1.36467598       2.62134689       0.76493011
##          tv_film    sports_fandom         politics             food
##       1.10546379       5.86912325       1.17662008       4.53748412
##           family  home_and_garden            music             news
##       2.48030496       0.66200762       0.75984752       1.04828463
##    online_gaming         shopping health_nutrition      college_uni
##       1.28589581       1.46759848       1.86277001       1.54129606
##   sports_playing          cooking              eco        computers
##       0.80432020       1.59085133       0.65311309       0.74459975
##         business         outdoors           crafts       automotive
##       0.49936468       0.70775095       1.07369759       1.04955527
##              art         religion           beauty        parenting
##       0.88818297       5.24396442       1.08005083       4.01905972
##           dating           school personal_fitness          fashion
##       0.77255400       2.68869123       1.20203304       1.00635324
##   small_business             spam            adult
##       0.41041931       0.00635324       0.40406607
```

The topics that stand out most in this cluster are:

```
## [1] "sports_fandom"
```

```
## [1] "religion"
```

The following is a plot of the users based on the frequency with which they tweet about sports_fandom and religion. The users are color coded by cluster.



As can be seen from the plot above, the most frequent tweeters on topics sports_fandom and religion (top right of the graph) are mostly all in cluster 2.

The centroid coordinates for the third cluster are:

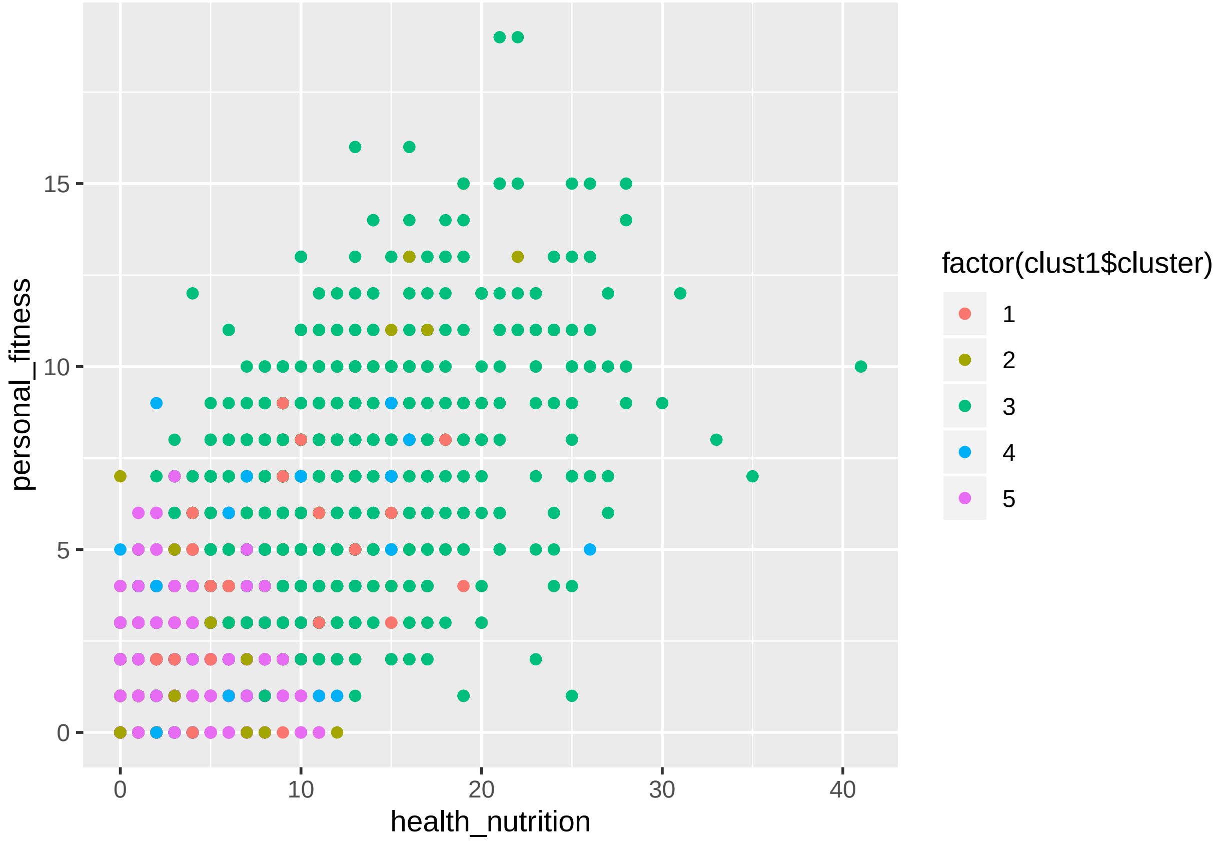
```
##   current_events           travel    photo_sharing    uncategorized
##      1.552828175      1.234791889      2.704375667      0.970117396
##          tv_film    sports_fandom         politics             food
##      1.034151547      1.160085379      1.246531483      2.106723586
##           family  home_and_garden            music             news
##      0.792956243      0.635005336      0.767342583      1.087513340
##    online_gaming         shopping health_nutrition      college_uni
##      1.197438634      1.487726788     11.843116329      1.335112060
##   sports_playing          cooking              eco        computers
##      0.691568837      3.252934899      0.911419424      0.550693703
##         business         outdoors           crafts       automotive
##      0.477054429      2.672358591      0.597652081      0.675560299
##              art         religion           beauty        parenting
##      0.749199573      0.754535752      0.416221985      0.754535752
##           dating           school personal_fitness          fashion
##      1.024546425      0.589114194      6.354322305      0.776947705
##   small_business             spam            adult
##      0.293489861      0.006403415      0.421558164
```

The topics that stand out most in this cluster are:

```
## [1] "health_nutrition"
```

```
## [1] "personal_fitness"
```

The following is a plot of the users based on the frequency with which they tweet about health_nutrition and personal_fitness. The users are color coded by cluster.



As can be seen from the plot above, the most frequent tweeters on topics health_nutrition and personal_fitness (top right of the graph) are mostly all in cluster 3.

The centroid coordinates for the fourth cluster are:

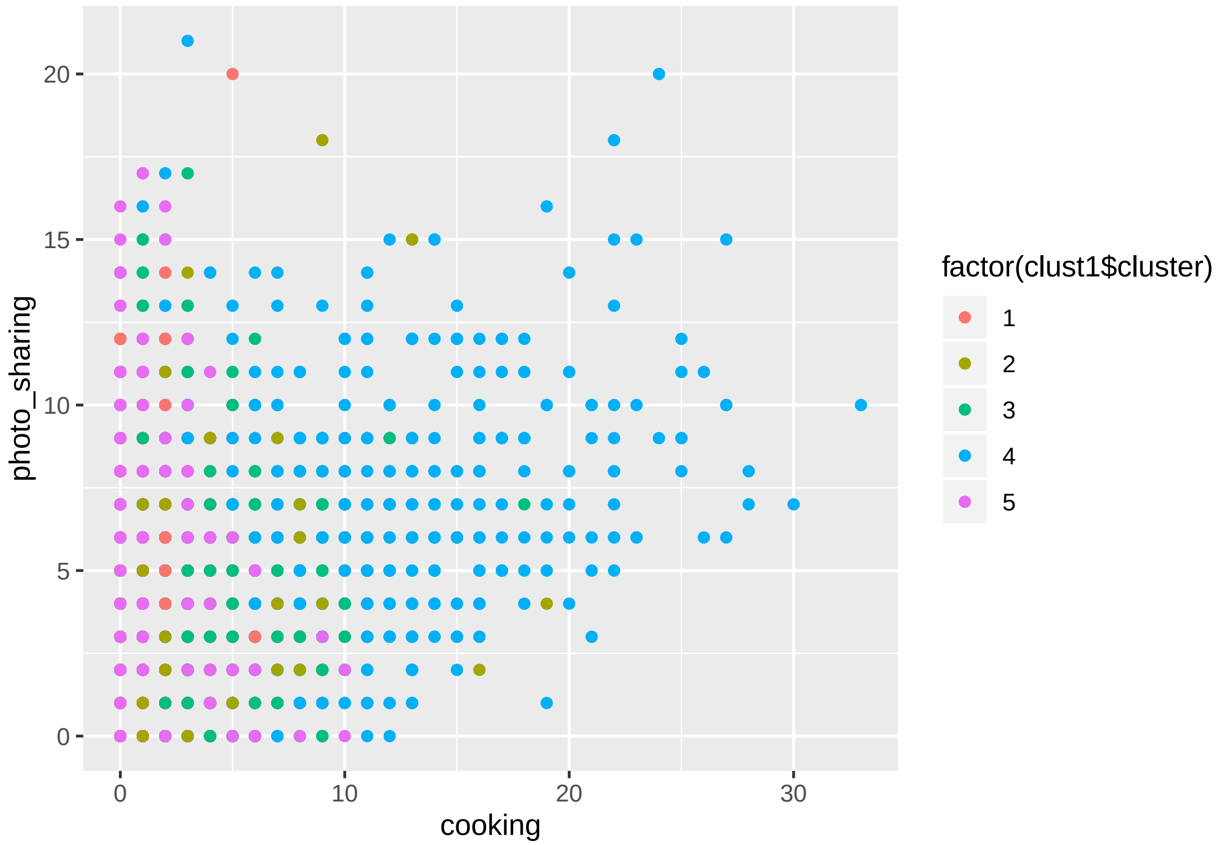
```
##   current_events           travel    photo_sharing    uncategorized 
##      1.766556291      1.514900662      6.082781457      1.293046358 
##          tv_film    sports_fandom         politics             food 
##      1.180463576      1.165562914      1.403973510      1.102649007 
##           family  home_and_garden            music             news 
##      0.912251656      0.642384106      1.284768212      1.044701987 
##    online_gaming         shopping health_nutrition      college_uni 
##      1.478476821      2.104304636      2.271523179      2.038079470 
##   sports_playing          cooking              eco        computers 
##      0.932119205     10.607615894      0.579470199      0.746688742 
##         business         outdoors           crafts       automotive 
##      0.617549669      0.841059603      0.647350993      0.902317881 
##              art         religion           beauty        parenting 
##      0.995033113      0.870860927      3.804635762      0.807947020 
##           dating           school personal_fitness          fashion 
##      0.996688742      0.985099338      1.352649007      5.480132450 
##   small_business             spam            adult 
##      0.529801325      0.003311258      0.415562914
```

The topics that stand out most in this cluster are:

```
## [1] "cooking"
```

```
## [1] "photo_sharing"
```

The following is a plot of the users based on the frequency with which they tweet about cooking and photo_sharing. The users are color coded by cluster.



As can be seen from the plot above, the most frequent tweeters on topics cooking and photo_sharing (top right of the graph) are mostly all in cluster 4.

The centroid coordinates for the fifth cluster are:

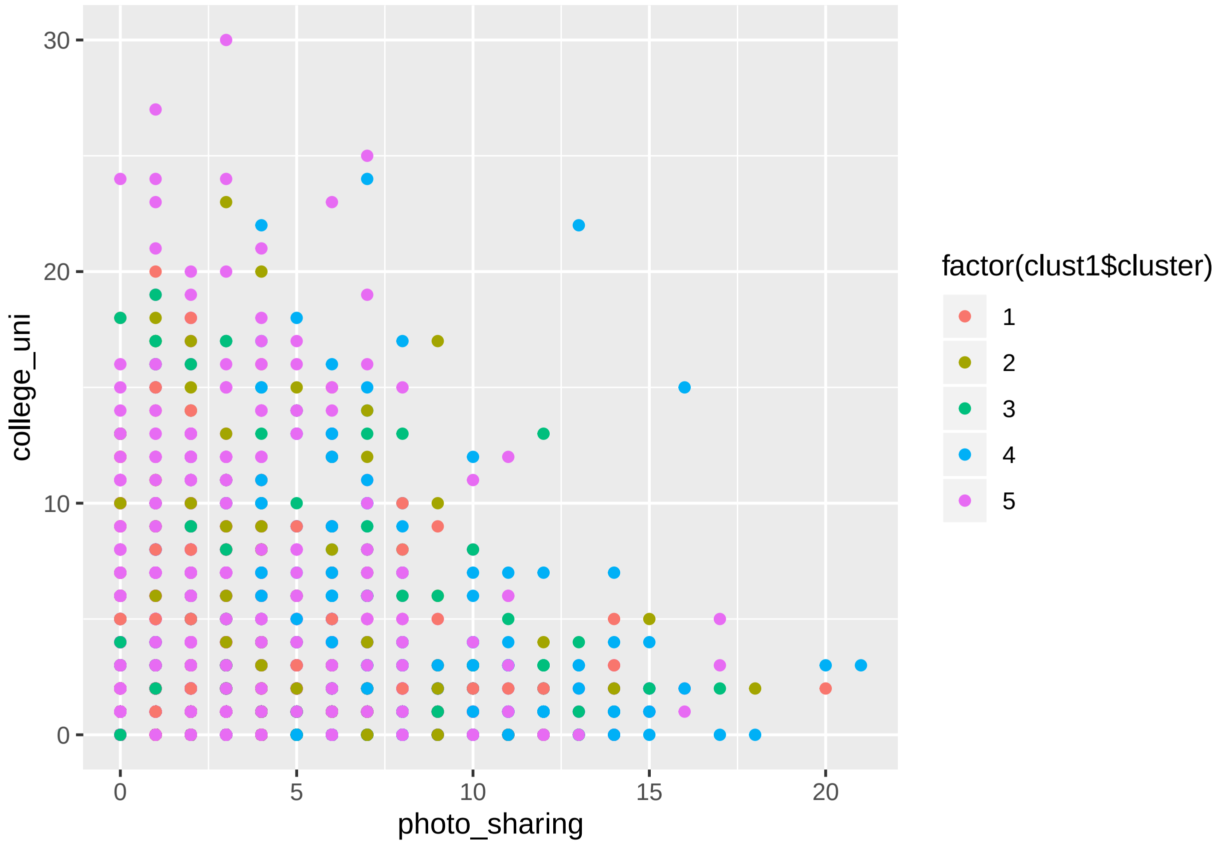
```
##   current_events           travel    photo_sharing    uncategorized
##      1.448874200      1.110514357      2.311505887      0.733939269
##          tv_film    sports_fandom         politics             food
##      1.035323280      0.977897129      1.002891964      0.779384425
##           family  home_and_garden            music             news
##      0.600702334      0.446601942      0.579838876      0.684982442
##    online_gaming         shopping health_nutrition      college_uni
##      1.167733939      1.269985540      1.056393307      1.511258005
##   sports_playing          cooking              eco        computers
##      0.556083454      0.854575501      0.391241479      0.372237141
##         business         outdoors           crafts       automotive
##      0.339392687      0.401156786      0.373063417      0.595744681
##              art         religion           beauty        parenting
##      0.655649659      0.527576947      0.348481719      0.463540591
##          dating          school personal_fitness          fashion
##     0.552571783     0.471390209      0.638917579      0.524065276
##   small_business            spam            adult
##     0.287543896     0.006610205      0.417682297
```

The topics that stand out most in this cluster are:

```
## [1] "photo_sharing"
```

```
## [1] "college_uni"
```

The following is a plot of the users based on the frequency with which they tweet about photo_sharing and college_uni. The users are color coded by cluster.



As can be seen from the plot above, the most frequent tweeters on topics photo_sharing and college_uni (top right of the graph) are mostly all in cluster 5.

Conclusions

The four market segments that best seem to be indicated from the Twitter data are:

- Users that cook and share photos often (cooking, photo_sharing)
- Consumers that are into health and nutrition and personal fitness (health_nutrition, personal_fitness)
- Politically informed/opinionated travellers (politics, travel)
- And religious sports fanatics (religion, sports_fandom)

A fifth cluster for photo sharing college/university students was tested but the data didn't cluster well.

NutrientH20 can make good use of these clusterings by marketing differently to customers in those different groups.