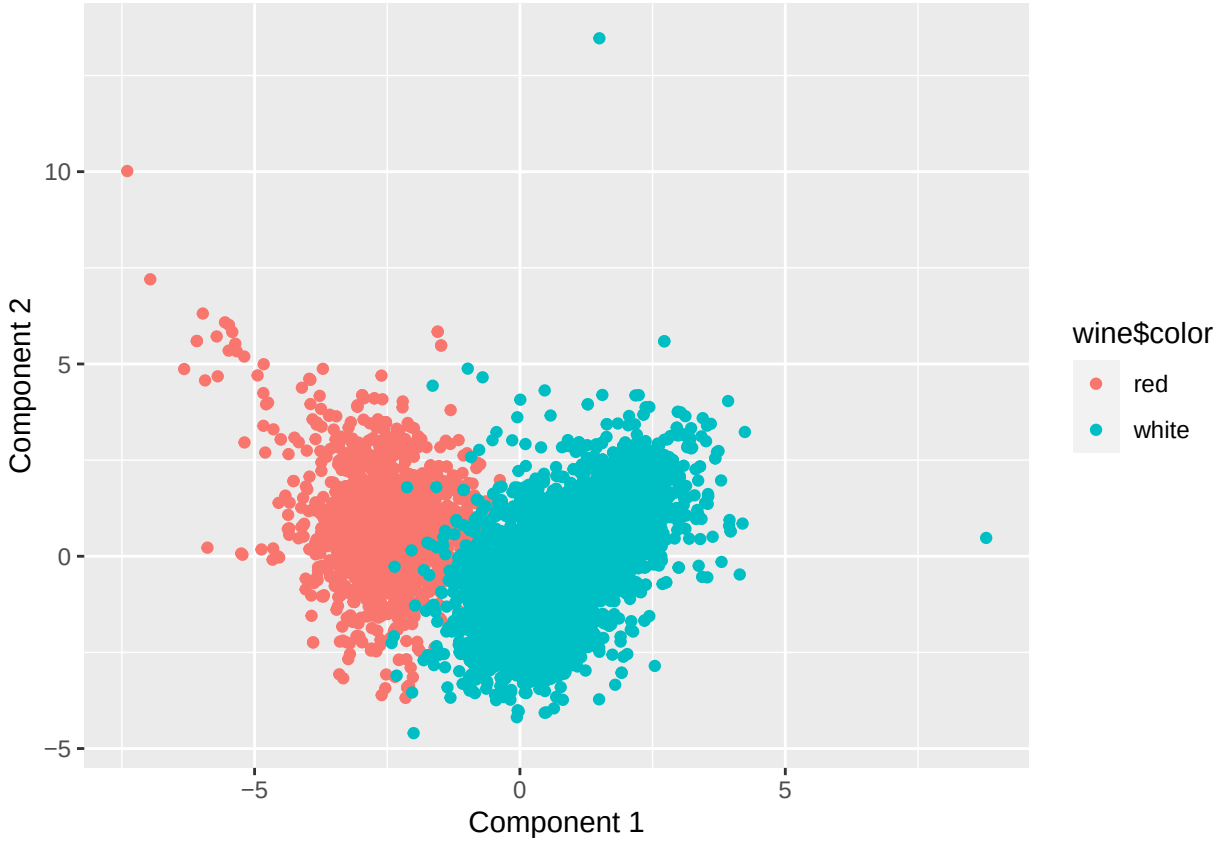
Data Mining HW4

Joey Herrera

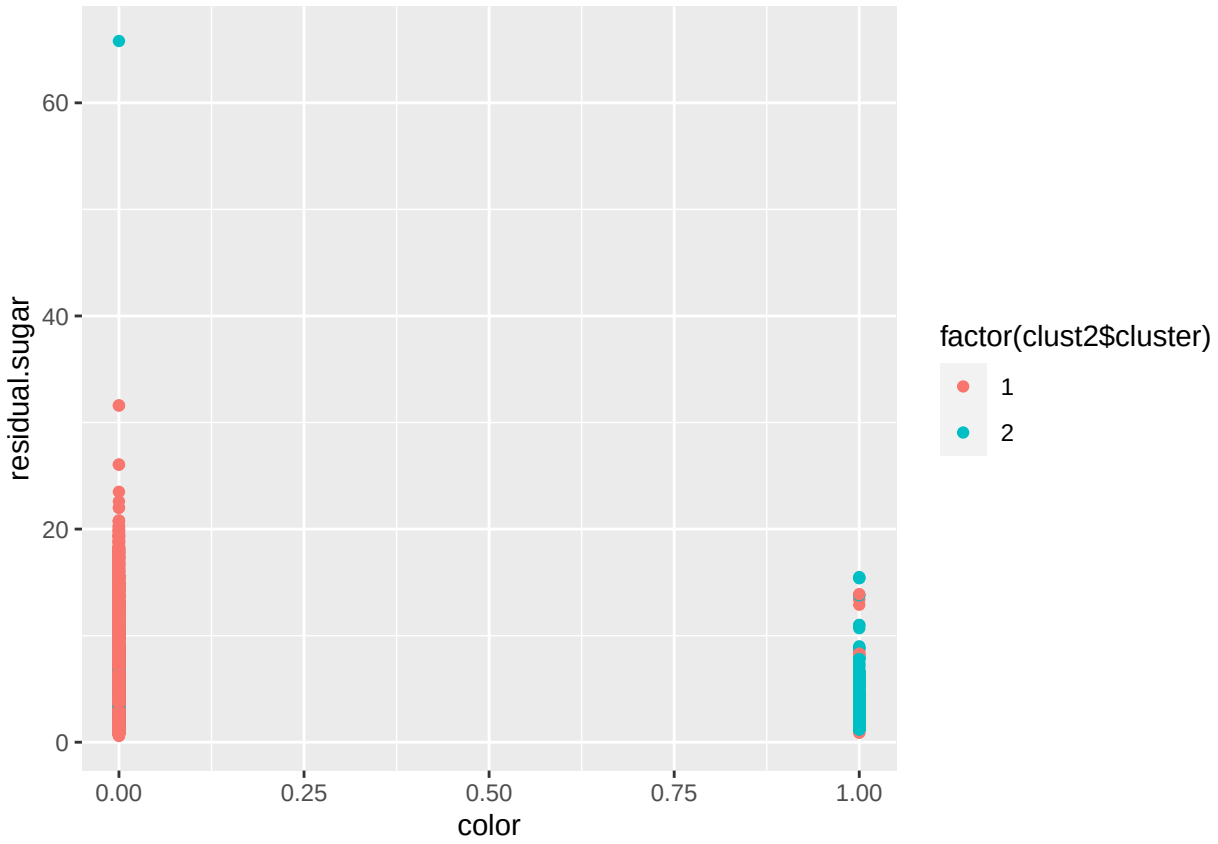
4/25/2021

Data Mining Assignment #4

Question 1: Run both PCA and a clustering algorithm of your choice on the 11 chemical properties (or suitable transformations thereof) and summarize your results. Which dimensionality reduction technique makes more sense to you for this data? Convince yourself (and me) that your chosen method is easily capable of distinguishing the reds from the whites, using only the "unsupervised" information contained in the data on chemical properties. Does your unsupervised technique also seem capable of distinguishing the higher from the lower quality wines?

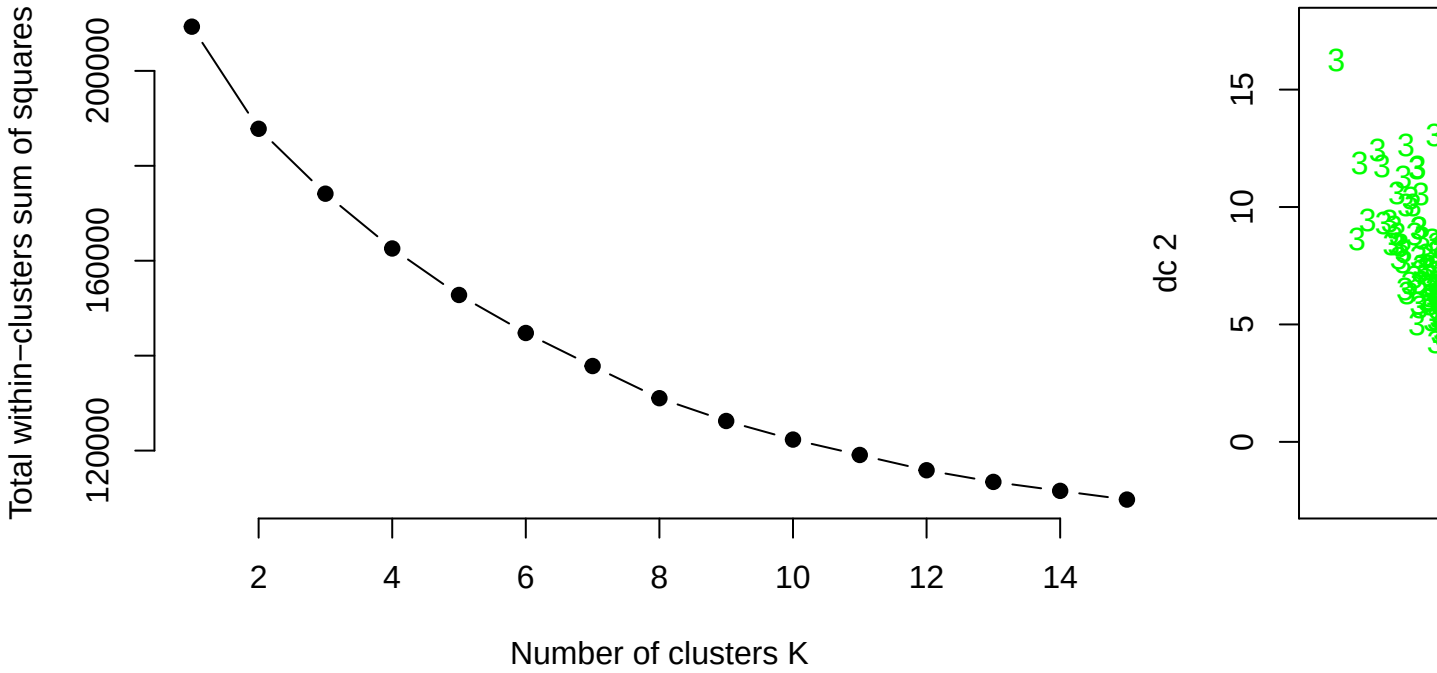


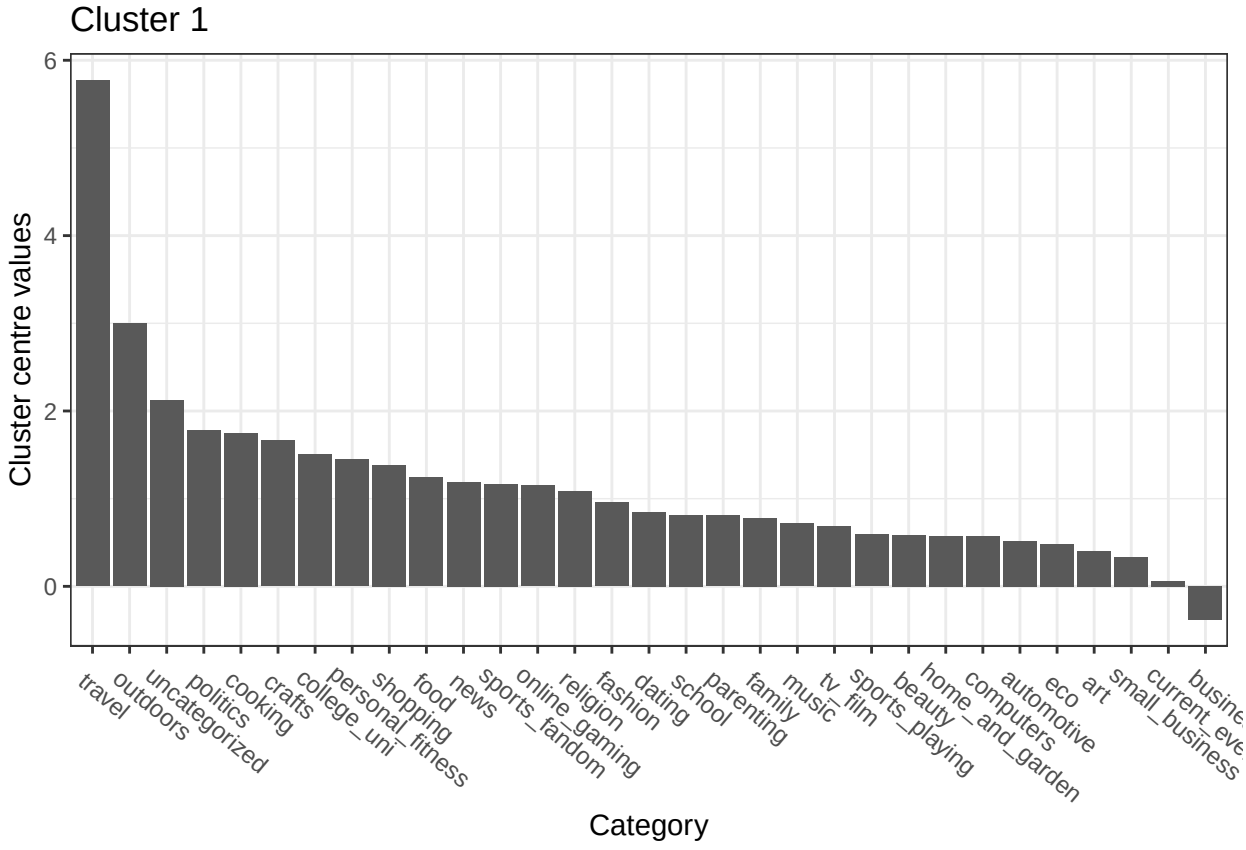

```
## [1] 0.1189974
```

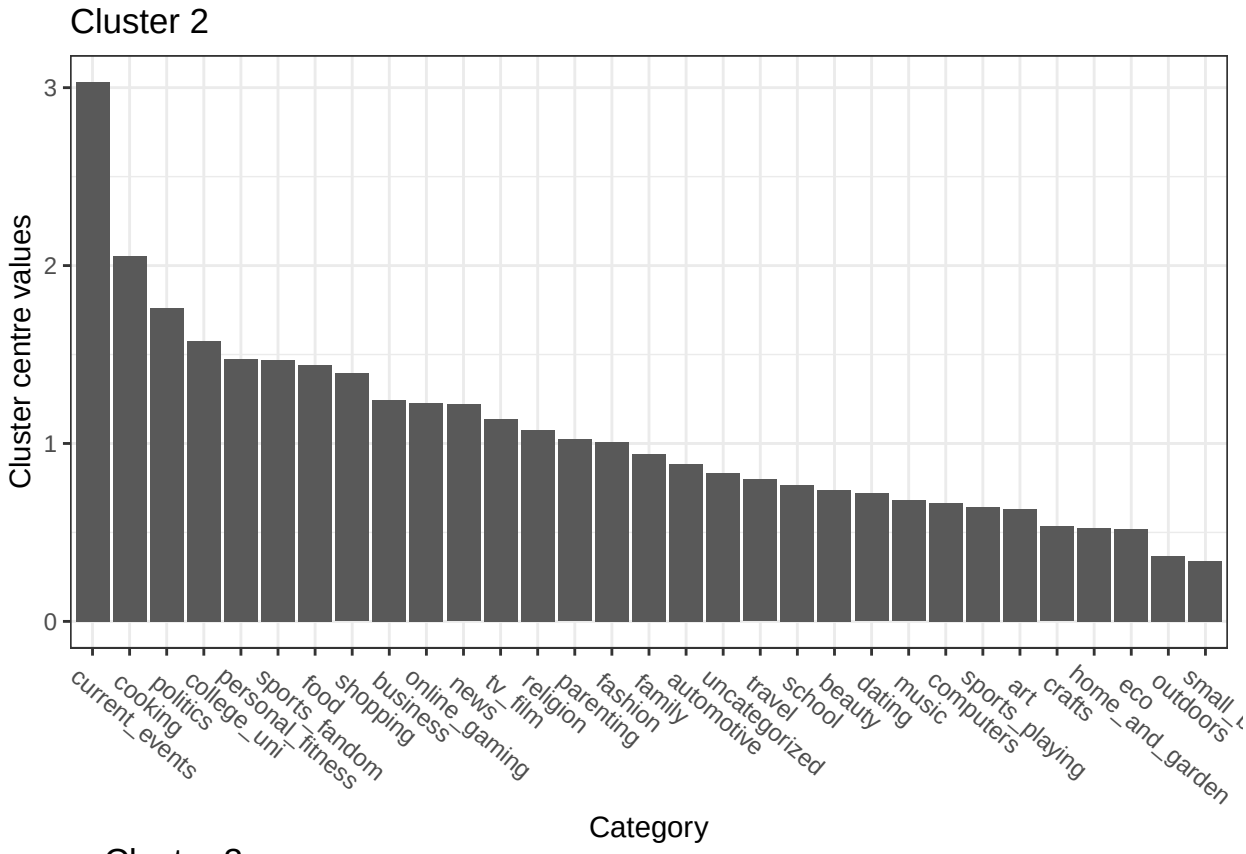


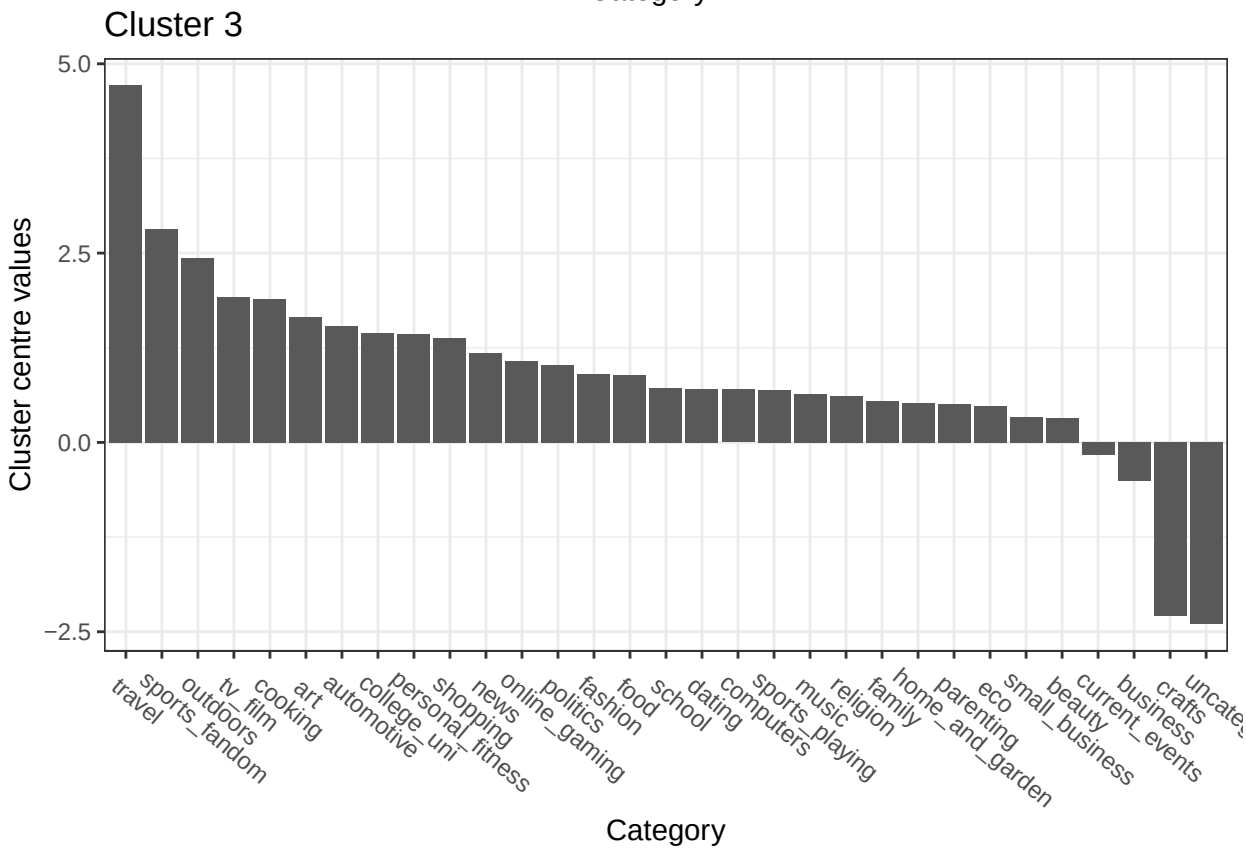
Answer: After performing a PCA and K-Means clustering technique on the wine data, I believe that K-Means clustering is more effective at distinguishing red wine from white wine given the clustering technique's higher RMSE. Since the K-Means clustering technique only produces two different clusters, it will be ineffective at predicting the quality of a wine because there are more than two different clusters when predicting the quality of the wine.

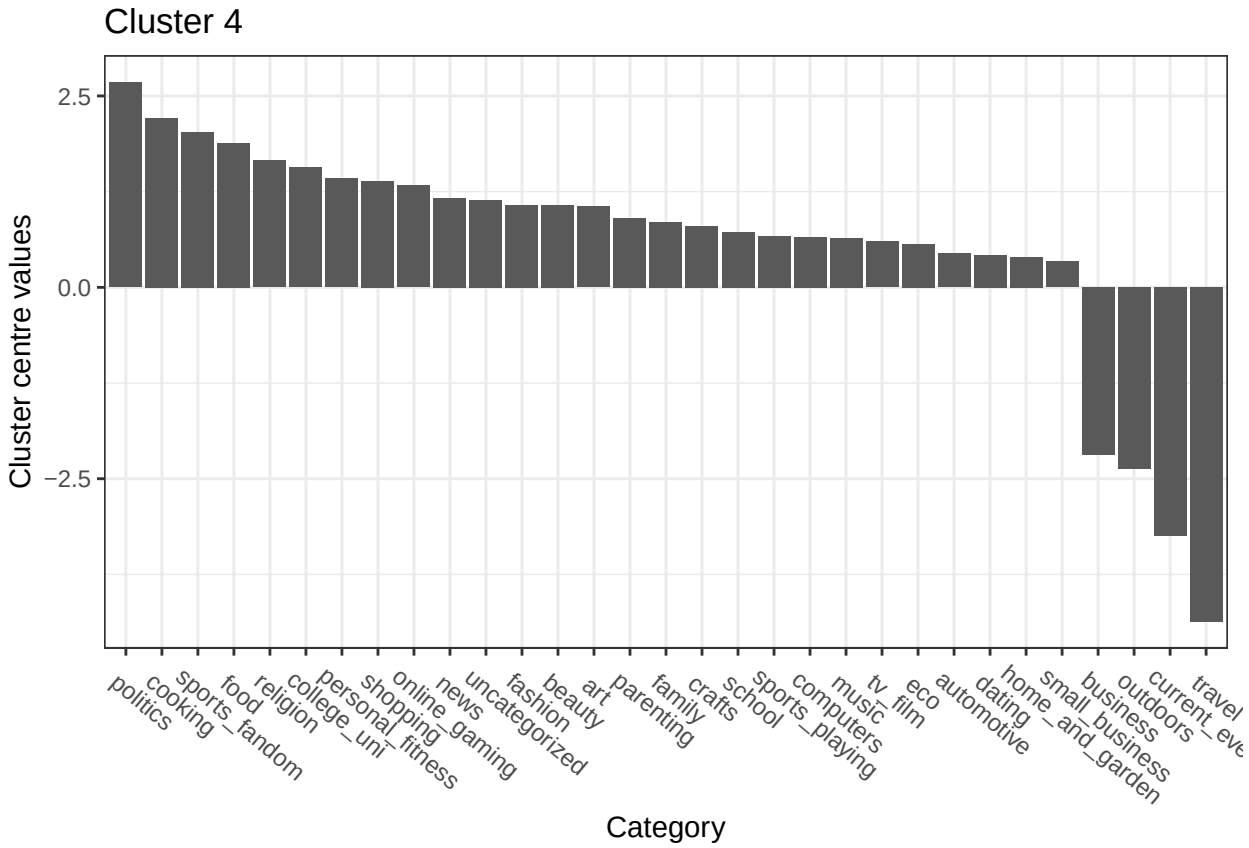
Question 2: Market Segmentation Your task to is analyze this data as you see fit, and to prepare a (short!) report for NutrientH20 that identifies any interesting market segments that appear to stand out in their social-media audience. You have complete freedom in deciding how to pre-process the data and how to define "market segment." (Is it a group of correlated interests? A cluster? A principal component? Etc. You decide the answer to this question—don't ask me!) Just use the data to come up with some interesting, well-supported insights about the audience and give your client some insight as to how they might position their brand to maximally appeal to each market segment.


Total within−clusters sum of squares
120000
160000
200000
2
4
6
8
10
12
14
Number of clusters K
dc 2
0
5
10
15
3


Cluster 1
Cluster centre values
0
2
4
6
travel
outdoors
uncategorized
politics
cooking
crafts
college_uni
personal_fitness
shopping
food
news
sports_fandom
online_gaming
religion
fashion
dating
school
parenting
family
music
tv_film
sports_playing
beauty
home_and_garden
computers
automotive
eco
art
small_business
current_eve
busine
Category


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


Cluster 3
Cluster centre values
5.0
2.5
0.0
−2.5
travel
sports_fandom
outdoors
tv_film
cooking
art
automotive
college_uni
personal_fitness
shopping
news
online_gaming
politics
fashion
food
school
dating
computers
sports_playing
music
religion
family
home_and_garden
parenting
eco
small_business
beauty
current_events
business
crafts
Category

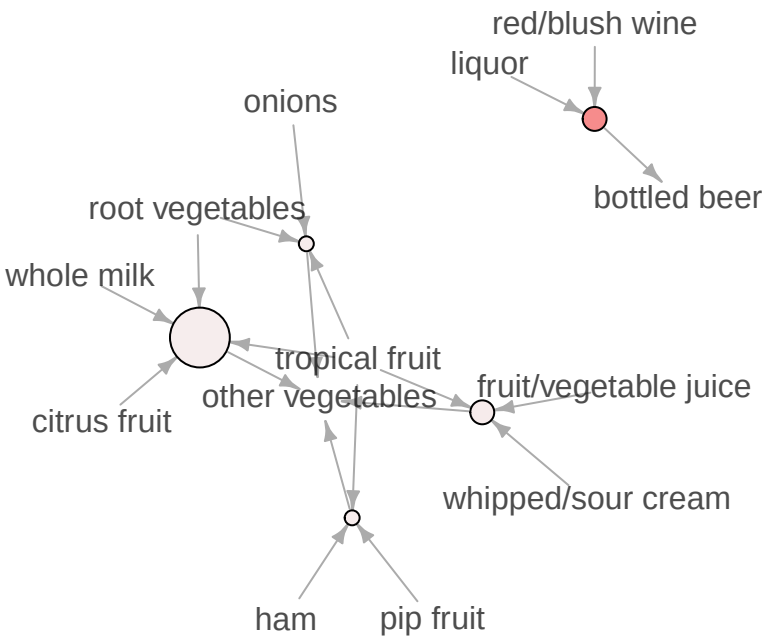


To understand NutrientH20's social-media audiance better and improve their messaging, I fit a Kmeans++ clustering model to the relevant topics that tweets were sorted into. I did not include chatter, spam, photography, health nutrition, or adult because they were highly correlated to other features and did not offer much additional insight when looking at the four different clusters. The for clusters displayed in the plots above indicate four different social media marketing segments which are

1. travel, outdoors
2. current events, cooking, politics, college uni
3. travel, sports fandom, outdoors
4. politics, cooking, sport fandom, food

Based on the K-Means clustering, there are four different clusters which seem to have an overlap in areas like sports fandom, traveling, and outdoors. These clusters seem to have various other interests including politics, cooking, current events, and college university. Given this information, I recommend for NutrientH20 to specify their social media marketing campaigns towards these specific topics.

Question 3: Revisit the notes on association rule mining and the R example on music playlists: playlists.R and playlists.csv. Then use the data on grocery purchases in groceries.txt and find some interesting association rules for these shopping baskets. The data file is a list of shopping baskets: one person's basket for each row, with multiple items per row separated by commas – you'll have to cobble together a few utilities for processing this into the format expected by the "arules" package. Pick your own thresholds for lift and confidence; just be clear what these thresholds are and how you picked them. Do your discovered item sets make sense? Present your discoveries in an interesting and concise way.



Answer: To find some interesting association rules for different grocery baskets, I chose a 1.5% support rate, 80% confidence, and lift greater than 1 to focus on the goods that cause a have a high probablit of occuring if another good is bought. The graph for rules above whole milk and other vegetables are very likely to be bought in carts that are purchasing multiple goods. Also, bottled beer is likely to be purchased with other forms of alcohol, such as wine and liquor.

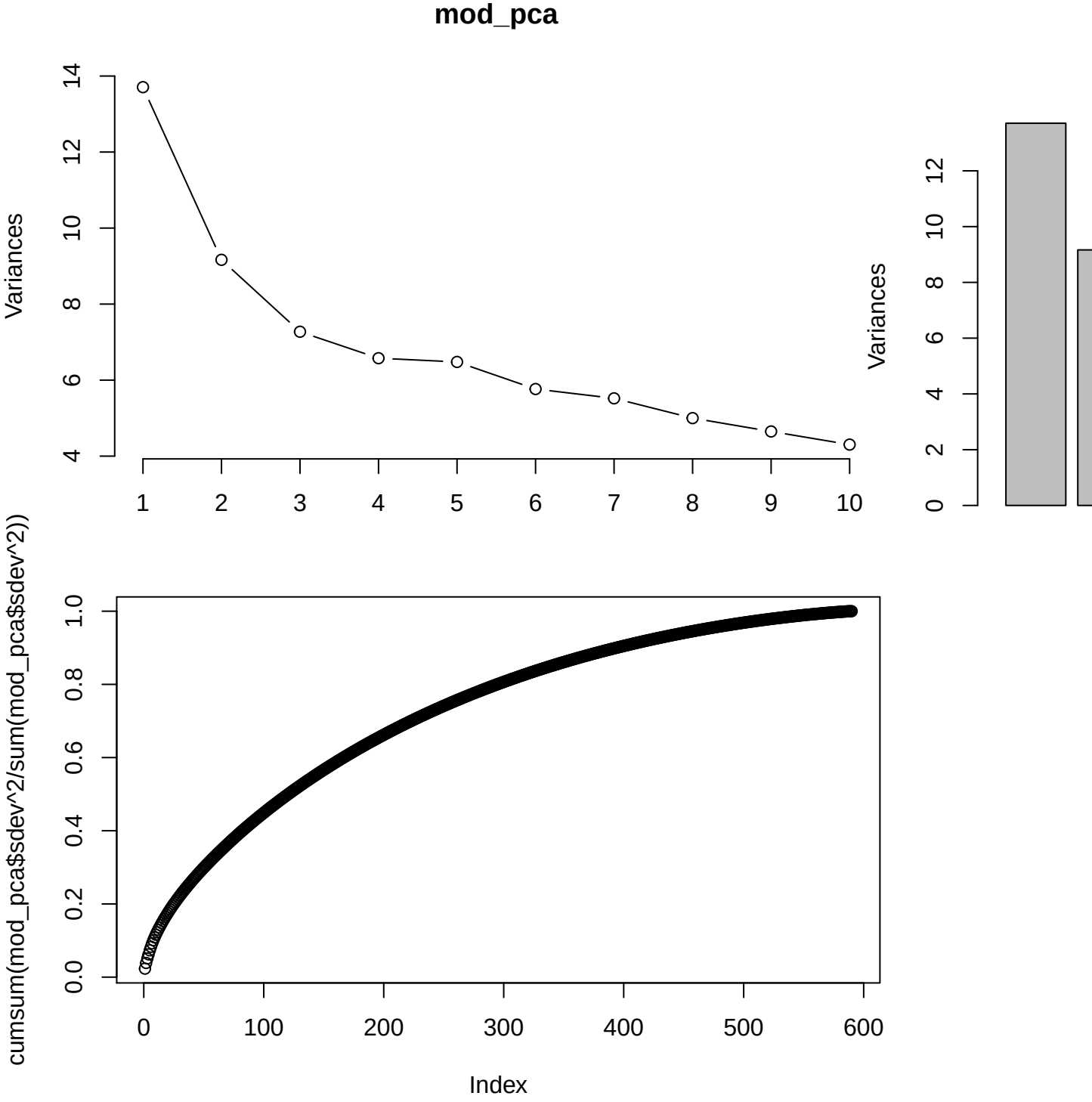
Question 4: Revisit the Reuters C50 corpus that we explored in class. Your task is to build the best model you can, using any combination of tools you see fit, for predicting the author of an article on the basis of that article's textual content. Describe clearly what model you are using, how you constructed features, and so forth.

Answer: The first step to creating a predictive model to match the author to their respective articles based on each article's textual content is to format the articles into a dataset that supervised techniques will accept. First, I loaded the data and used a wrapper function to indicate I wanted each article to be identified by the file name and be in English. As I read in the articles, I noticed that the Reuters C50 directory for training data has 50 other subdirectories, one per author, with 50 unique articles in each sub-directory. To extract all of the relevant information, I had to "glob" the file paths of all fifty sub-directories into a single object. Next, I took a substring of the authors' names and globbed the text files associated with each author's name. Then, I appended all of the file paths and authors' names. After creating this object, I turned it into a corpus, which contained 2500 elements, one for each of the 50 articles the 50 different authors wrote.

The next step of preprocessing the data involved removing numbers, capitalization, punctuation, white space, and stop words using the content transformer function in the tm package. Next, I was able to turn my corpus into a document term matrix (DTM) and turned it into a data frame consisting of 2500 observations for N remaining number of terms. I also extracted a vector of the author's names by taking the file names for each article and cleaning the names until all that was left is the authors' names. Both the vector of authors' names and the matrix of terms have a length of 2500. Apply the same steps to the testing set data to get a matrix of length 2500 by the same number of features in the training matrix.

Finally, I used the PCA dimension reduction technique to turn the higher number of features into smaller principal components. I first took all of the columns filled with 0's out of the training and testing matrices

and reduced the number of columns in these matrices to only use intersecting columns. Then, I ran PCA on the training data and used the model to predict authors' names in the testing data. Afterward, I plotted the principal components in a line graph and decided to use 265 principal components because they preserve approximately 75% of the variance in the original data.

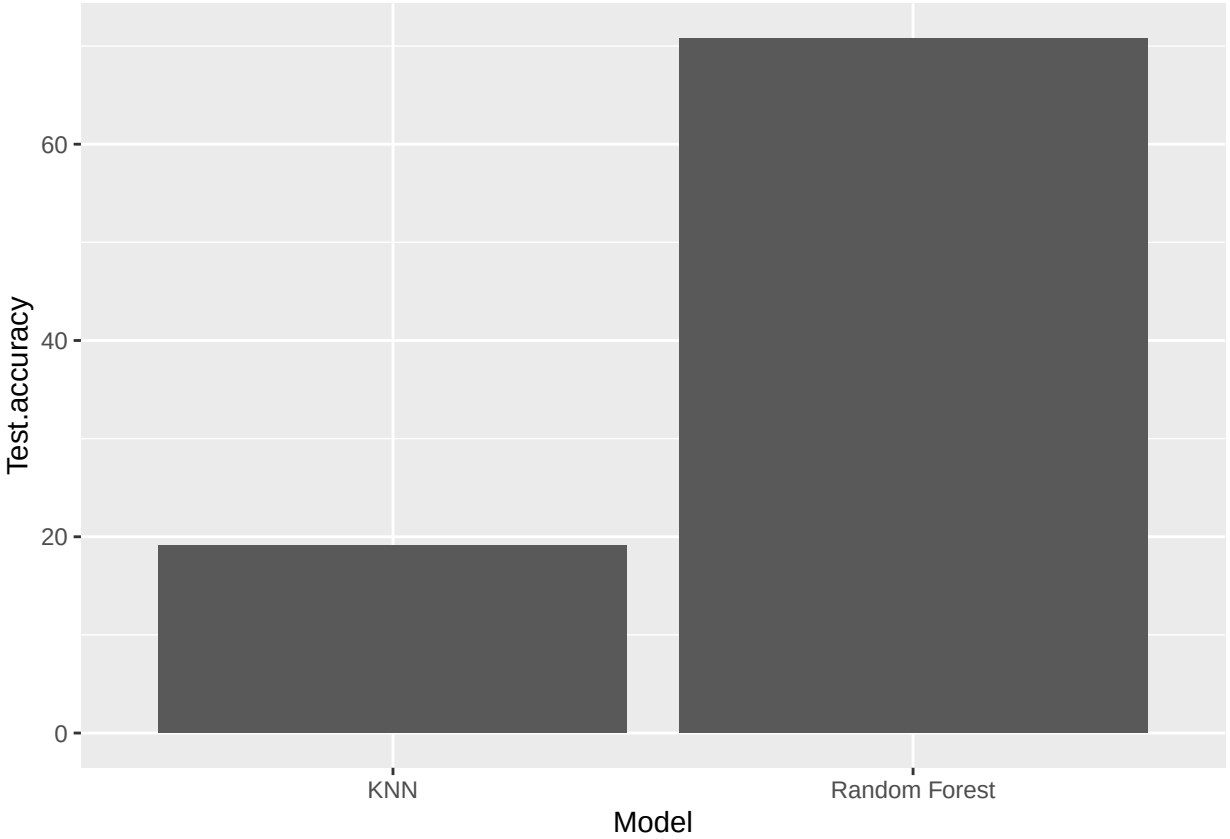


Using the 256 principal components, I took the original data points in the PCA predictive model ("mod_pca") in a data frame and did the same thing for the test data. These were named "train_class" and "test_class."

Now, I can use supervised learning techniques. In particular, I am focusing on predictive models using a

random forest and KNN. After using these techniques, I found the following results.

```
## [1] 29.08
```



The more effective predictive model is the random forest, which outperformed the KNN model by almost 40%.