exercises

group: Tea, Mikala, Alicia

2024-08-06

Market Segmentation

In efforts to help NutrientH20 better understand their social-media audience, I have investigated the data to help identify areas of interest. The first step I took was looking at the structure of the data to understand how to explore it best. Below you can see some of the work I did for that.

```
library(readr)
social_marketing <- read_csv("social_marketing.csv")
```

```
## New names:
## Rows: 7882 Columns: 37
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): ...1 dbl (36): chatter, current_events, travel, photo_sharing,
## uncategorized, tv_...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

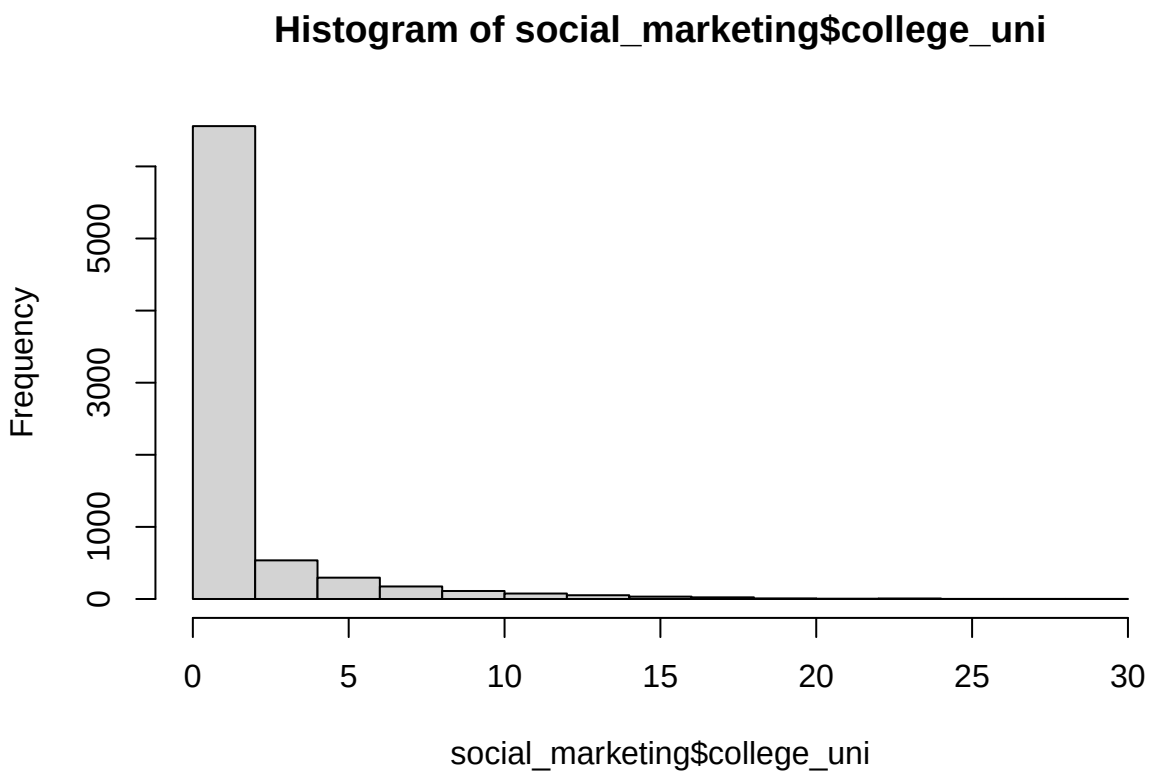
```
names(social_marketing)
```

```
##  [1] "...1"             "chatter"          "current_events"   "travel"
##  [5] "photo_sharing"    "uncategorized"    "tv_film"          "sports_fandom"
##  [9] "politics"         "food"             "family"           "home_and_garden"
## [13] "music"            "news"             "online_gaming"    "shopping"
## [17] "health_nutrition" "college_uni"      "sports_playing"   "cooking"
## [21] "eco"              "computers"        "business"         "outdoors"
## [25] "crafts"           "automotive"       "art"              "religion"
## [29] "beauty"           "parenting"        "dating"           "school"
## [33] "personal_fitness" "fashion"          "small_business"   "spam"
## [37] "adult"
```

```
# changing the user column name to user because the "...1" was weird

colnames(social_marketing) <- c("user","chatter", "current_events", "travel", "photo_sharing", "uncateg
"dating", "school", "personal_fitness", "fashion", "small_business", "spam",  "adult")


dim(social_marketing)
```

```
## [1] 7882   37
```

```
hist(social_marketing$college_uni)
```




```
summary(social_marketing)
```

```
##      user              chatter       current_events      travel
##  Length:7882        Min.   : 0.000   Min.   :0.000   Min.   : 0.000
##  Class :character   1st Qu.: 2.000   1st Qu.:1.000   1st Qu.: 0.000
##  Mode  :character   Median : 3.000   Median :1.000   Median : 1.000
##                     Mean   : 4.399   Mean   :1.526   Mean   : 1.585
##                     3rd Qu.: 6.000   3rd Qu.:2.000   3rd Qu.: 2.000
##                     Max.   :26.000   Max.   :8.000   Max.   :26.000
##  photo_sharing    uncategorized      tv_film      sports_fandom
##  Min.   : 0.000   Min.   :0.000   Min.   : 0.00   Min.   : 0.000
##  1st Qu.: 1.000   1st Qu.:0.000   1st Qu.: 0.00   1st Qu.: 0.000
##  Median : 2.000   Median :1.000   Median : 1.00   Median : 1.000
##  Mean   : 2.697   Mean   :0.813   Mean   : 1.07   Mean   : 1.594
##  3rd Qu.: 4.000   3rd Qu.:1.000   3rd Qu.: 1.00   3rd Qu.: 2.000
##  Max.   :21.000   Max.   :9.000   Max.   :17.00   Max.   :20.000
##     politics          food            family        home_and_garden
##  Min.   : 0.000   Min.   : 0.000   Min.   : 0.0000   Min.   :0.0000
##  1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 0.0000   1st Qu.:0.0000
##  Median : 1.000   Median : 1.000   Median : 1.0000   Median :0.0000
##  Mean   : 1.789   Mean   : 1.397   Mean   : 0.8639   Mean   :0.5207
```

```
##  3rd Qu.: 2.000   3rd Qu.: 2.000   3rd Qu.: 1.0000   3rd Qu.:1.0000
##  Max.   :37.000   Max.   :16.000   Max.   :10.0000   Max.   :5.0000
##      music             news        online_gaming       shopping
##  Min.   : 0.0000   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 0.000
##  Median : 0.0000   Median : 0.000   Median : 0.000   Median : 1.000
##  Mean   : 0.6793   Mean   : 1.206   Mean   : 1.209   Mean   : 1.389
##  3rd Qu.: 1.0000   3rd Qu.: 1.000   3rd Qu.: 1.000   3rd Qu.: 2.000
##  Max.   :13.0000   Max.   :20.000   Max.   :27.000   Max.   :12.000
##  health_nutrition  college_uni     sports_playing      cooking
##  Min.   : 0.000   Min.   : 0.000   Min.   :0.0000   Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.:0.0000   1st Qu.: 0.000
##  Median : 1.000   Median : 1.000   Median :0.0000   Median : 1.000
##  Mean   : 2.567   Mean   : 1.549   Mean   :0.6392   Mean   : 1.998
##  3rd Qu.: 3.000   3rd Qu.: 2.000   3rd Qu.:1.0000   3rd Qu.: 2.000
##  Max.   :41.000   Max.   :30.000   Max.   :8.0000   Max.   :33.000
##       eco            computers          business         outdoors
##  Min.   :0.0000   Min.   : 0.0000   Min.   :0.0000   Min.   : 0.0000
##  1st Qu.:0.0000   1st Qu.: 0.0000   1st Qu.:0.0000   1st Qu.: 0.0000
##  Median :0.0000   Median : 0.0000   Median :0.0000   Median : 0.0000
##  Mean   :0.5123   Mean   : 0.6491   Mean   :0.4232   Mean   : 0.7827
##  3rd Qu.:1.0000   3rd Qu.: 1.0000   3rd Qu.:1.0000   3rd Qu.: 1.0000
##  Max.   :6.0000   Max.   :16.0000   Max.   :6.0000   Max.   :12.0000
##      crafts          automotive           art             religion
##  Min.   :0.0000   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.000
##  1st Qu.:0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.000
##  Median :0.0000   Median : 0.0000   Median : 0.0000   Median : 0.000
##  Mean   :0.5159   Mean   : 0.8299   Mean   : 0.7248   Mean   : 1.095
##  3rd Qu.:1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.000
##  Max.   :7.0000   Max.   :13.0000   Max.   :18.0000   Max.   :20.000
##      beauty          parenting           dating            school
##  Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.0000
##  1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.0000
##  Median : 0.0000   Median : 0.0000   Median : 0.0000   Median : 0.0000
##  Mean   : 0.7052   Mean   : 0.9213   Mean   : 0.7109   Mean   : 0.7677
##  3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.0000
##  Max.   :14.0000   Max.   :14.0000   Max.   :24.0000   Max.   :11.0000
##  personal_fitness    fashion        small_business        spam
##  Min.   : 0.000   Min.   : 0.0000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.: 0.000   1st Qu.: 0.0000   1st Qu.:0.0000   1st Qu.:0.00000
##  Median : 0.000   Median : 0.0000   Median :0.0000   Median :0.00000
##  Mean   : 1.462   Mean   : 0.9966   Mean   :0.3363   Mean   :0.00647
##  3rd Qu.: 2.000   3rd Qu.: 1.0000   3rd Qu.:1.0000   3rd Qu.:0.00000
##  Max.   :19.000   Max.   :18.0000   Max.   :6.0000   Max.   :2.00000
##      adult
##  Min.   : 0.0000
##  1st Qu.: 0.0000
##  Median : 0.0000
##  Mean   : 0.4033
##  3rd Qu.: 0.0000
##  Max.   :26.0000
```

```
str(social_marketing)
```

```
## spc_tbl_ [7,882 x 37] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ user            : chr [1:7882] "hmjoe4g3k" "clk1m5w8s" "jcsovtak3" "3oeb4hiln" ...
##  $ chatter         : num [1:7882] 2 3 6 1 5 6 1 5 6 5 ...
##  $ current_events  : num [1:7882] 0 3 3 5 2 4 2 3 2 2 ...
##  $ travel          : num [1:7882] 2 2 4 2 0 2 7 3 0 4 ...
##  $ photo_sharing   : num [1:7882] 2 1 3 2 6 7 1 6 1 4 ...
##  $ uncategorized   : num [1:7882] 2 1 1 0 1 0 0 1 0 0 ...
##  $ tv_film         : num [1:7882] 1 1 5 1 0 1 1 1 0 5 ...
##  $ sports_fandom   : num [1:7882] 1 4 0 0 0 1 1 1 0 9 ...
##  $ politics        : num [1:7882] 0 1 2 1 2 0 11 0 0 1 ...
##  $ food            : num [1:7882] 4 2 1 0 0 2 1 0 2 5 ...
##  $ family          : num [1:7882] 1 2 1 1 1 1 0 0 2 4 ...
##  $ home_and_garden : num [1:7882] 2 1 1 0 0 1 0 0 1 0 ...
##  $ music           : num [1:7882] 0 0 1 0 0 1 0 2 1 1 ...
##  $ news            : num [1:7882] 0 0 1 0 0 0 1 0 0 0 ...
##  $ online_gaming   : num [1:7882] 0 0 0 0 3 0 0 1 2 1 ...
##  $ shopping        : num [1:7882] 1 0 2 0 2 5 1 3 0 0 ...
##  $ health_nutrition: num [1:7882] 17 0 0 0 0 0 1 1 22 7 ...
##  $ college_uni     : num [1:7882] 0 0 0 1 4 0 1 0 1 4 ...
##  $ sports_playing  : num [1:7882] 2 1 0 0 0 0 1 0 0 1 ...
##  $ cooking         : num [1:7882] 5 0 2 0 1 0 1 10 5 4 ...
##  $ eco             : num [1:7882] 1 0 1 0 0 0 0 0 2 1 ...
##  $ computers       : num [1:7882] 1 0 0 0 1 1 1 1 1 2 ...
##  $ business        : num [1:7882] 0 1 0 1 0 1 3 0 1 0 ...
##  $ outdoors        : num [1:7882] 2 0 0 0 1 0 1 0 3 0 ...
##  $ crafts          : num [1:7882] 1 2 2 3 0 0 0 1 0 0 ...
##  $ automotive      : num [1:7882] 0 0 0 0 0 1 0 1 0 4 ...
##  $ art             : num [1:7882] 0 0 8 2 0 0 1 0 1 0 ...
##  $ religion        : num [1:7882] 1 0 0 0 0 0 1 0 0 13 ...
##  $ beauty          : num [1:7882] 0 0 1 1 0 0 0 5 5 1 ...
##  $ parenting       : num [1:7882] 1 0 0 0 0 0 0 1 0 3 ...
##  $ dating          : num [1:7882] 1 1 1 0 0 0 0 0 0 0 ...
##  $ school          : num [1:7882] 0 4 0 0 0 0 0 0 1 3 ...
##  $ personal_fitness: num [1:7882] 11 0 0 0 0 0 0 0 12 2 ...
##  $ fashion         : num [1:7882] 0 0 1 0 0 0 0 4 3 1 ...
##  $ small_business  : num [1:7882] 0 0 0 0 1 0 0 0 1 0 ...
##  $ spam            : num [1:7882] 0 0 0 0 0 0 0 0 0 0 ...
##  $ adult           : num [1:7882] 0 0 0 0 0 0 0 0 0 0 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ...1 = col_character(),
##   ..   chatter = col_double(),
##   ..   current_events = col_double(),
##   ..   travel = col_double(),
##   ..   photo_sharing = col_double(),
##   ..   uncategorized = col_double(),
##   ..   tv_film = col_double(),
##   ..   sports_fandom = col_double(),
##   ..   politics = col_double(),
##   ..   food = col_double(),
##   ..   family = col_double(),
##   ..   home_and_garden = col_double(),
##   ..   music = col_double(),
##   ..   news = col_double(),
```

```
##   ..   online_gaming = col_double(),
##   ..   shopping = col_double(),
##   ..   health_nutrition = col_double(),
##   ..   college_uni = col_double(),
##   ..   sports_playing = col_double(),
##   ..   cooking = col_double(),
##   ..   eco = col_double(),
##   ..   computers = col_double(),
##   ..   business = col_double(),
##   ..   outdoors = col_double(),
##   ..   crafts = col_double(),
##   ..   automotive = col_double(),
##   ..   art = col_double(),
##   ..   religion = col_double(),
##   ..   beauty = col_double(),
##   ..   parenting = col_double(),
##   ..   dating = col_double(),
##   ..   school = col_double(),
##   ..   personal_fitness = col_double(),
##   ..   fashion = col_double(),
##   ..   small_business = col_double(),
##   ..   spam = col_double(),
##   ..   adult = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

The next step I took was finding the most relevant categories. I captured the median of each column because I wanted to identify categories that were popular among the followers. A median greater than 0 indicates that at least half of the users have posted about something in that category at least once. This is important because it means the category is relevant to a significant portion of the user base.Below is the chunk where I explored that.

```
categories <- social_marketing[,!names(social_marketing) %in% 'user']
names(categories)
```

```
##  [1] "chatter"          "current_events"   "travel"           "photo_sharing"
##  [5] "uncategorized"    "tv_film"          "sports_fandom"    "politics"
##  [9] "food"             "family"           "home_and_garden"  "music"
## [13] "news"             "online_gaming"    "shopping"         "health_nutrition"
## [17] "college_uni"      "sports_playing"   "cooking"          "eco"
## [21] "computers"        "business"         "outdoors"         "crafts"
## [25] "automotive"       "art"              "religion"         "beauty"
## [29] "parenting"        "dating"           "school"           "personal_fitness"
## [33] "fashion"          "small_business"   "spam"             "adult"
```

```
str(categories)
```

```
## tibble [7,882 x 36] (S3: tbl_df/tbl/data.frame)
##  $ chatter         : num [1:7882] 2 3 6 1 5 6 1 5 6 5 ...
##  $ current_events  : num [1:7882] 0 3 3 5 2 4 2 3 2 2 ...
##  $ travel          : num [1:7882] 2 2 4 2 0 2 7 3 0 4 ...
##  $ photo_sharing   : num [1:7882] 2 1 3 2 6 7 1 6 1 4 ...
```

```
##  $ uncategorized   : num [1:7882] 2 1 1 0 1 0 0 1 0 0 ...
##  $ tv_film         : num [1:7882] 1 1 5 1 0 1 1 1 0 5 ...
##  $ sports_fandom   : num [1:7882] 1 4 0 0 0 1 1 1 0 9 ...
##  $ politics        : num [1:7882] 0 1 2 1 2 0 11 0 0 1 ...
##  $ food            : num [1:7882] 4 2 1 0 0 2 1 0 2 5 ...
##  $ family          : num [1:7882] 1 2 1 1 1 1 0 0 2 4 ...
##  $ home_and_garden : num [1:7882] 2 1 1 0 0 1 0 0 1 0 ...
##  $ music           : num [1:7882] 0 0 1 0 0 1 0 2 1 1 ...
##  $ news            : num [1:7882] 0 0 1 0 0 0 1 0 0 0 ...
##  $ online_gaming   : num [1:7882] 0 0 0 0 3 0 0 1 2 1 ...
##  $ shopping        : num [1:7882] 1 0 2 0 2 5 1 3 0 0 ...
##  $ health_nutrition: num [1:7882] 17 0 0 0 0 0 1 1 22 7 ...
##  $ college_uni     : num [1:7882] 0 0 0 1 4 0 1 0 1 4 ...
##  $ sports_playing  : num [1:7882] 2 1 0 0 0 0 1 0 0 1 ...
##  $ cooking         : num [1:7882] 5 0 2 0 1 0 1 10 5 4 ...
##  $ eco             : num [1:7882] 1 0 1 0 0 0 0 0 2 1 ...
##  $ computers       : num [1:7882] 1 0 0 0 1 1 1 1 1 2 ...
##  $ business        : num [1:7882] 0 1 0 1 0 1 3 0 1 0 ...
##  $ outdoors        : num [1:7882] 2 0 0 0 1 0 1 0 3 0 ...
##  $ crafts          : num [1:7882] 1 2 2 3 0 0 0 1 0 0 ...
##  $ automotive      : num [1:7882] 0 0 0 0 0 1 0 1 0 4 ...
##  $ art             : num [1:7882] 0 0 8 2 0 0 1 0 1 0 ...
##  $ religion        : num [1:7882] 1 0 0 0 0 0 1 0 0 13 ...
##  $ beauty          : num [1:7882] 0 0 1 1 0 0 0 5 5 1 ...
##  $ parenting       : num [1:7882] 1 0 0 0 0 0 0 1 0 3 ...
##  $ dating          : num [1:7882] 1 1 1 0 0 0 0 0 0 0 ...
##  $ school          : num [1:7882] 0 4 0 0 0 0 0 0 1 3 ...
##  $ personal_fitness: num [1:7882] 11 0 0 0 0 0 0 0 12 2 ...
##  $ fashion         : num [1:7882] 0 0 1 0 0 0 0 4 3 1 ...
##  $ small_business  : num [1:7882] 0 0 0 0 1 0 0 0 1 0 ...
##  $ spam            : num [1:7882] 0 0 0 0 0 0 0 0 0 0 ...
##  $ adult           : num [1:7882] 0 0 0 0 0 0 0 0 0 0 ...
```

```r
sum(is.na(categories))
```

```
## [1] 0
```

```r
# want to return the medians of each column, I had chatGPT write this chunk for me because I didn't wan
medians <- list(
  chatter = median(categories$chatter, na.rm = TRUE),
  current_events = median(categories$current_events, na.rm = TRUE),
  travel = median(categories$travel, na.rm = TRUE),
  photo_sharing = median(categories$photo_sharing, na.rm = TRUE),
  uncategorized = median(categories$uncategorized, na.rm = TRUE),
  tv_film = median(categories$tv_film, na.rm = TRUE),
  sports_fandom = median(categories$sports_fandom, na.rm = TRUE),
  politics = median(categories$politics, na.rm = TRUE),
  food = median(categories$food, na.rm = TRUE),
  family = median(categories$family, na.rm = TRUE),
  home_and_garden = median(categories$home_and_garden, na.rm = TRUE),
  music = median(categories$music, na.rm = TRUE),
  news = median(categories$news, na.rm = TRUE),
  online_gaming = median(categories$online_gaming, na.rm = TRUE),
```

```
  shopping = median(categories$shopping, na.rm = TRUE),
  health_nutrition = median(categories$health_nutrition, na.rm = TRUE),
  college_uni = median(categories$college_uni, na.rm = TRUE),
  sports_playing = median(categories$sports_playing, na.rm = TRUE),
  cooking = median(categories$cooking, na.rm = TRUE),
  eco = median(categories$eco, na.rm = TRUE),
  computers = median(categories$computers, na.rm = TRUE),
  business = median(categories$business, na.rm = TRUE),
  outdoors = median(categories$outdoors, na.rm = TRUE),
  crafts = median(categories$crafts, na.rm = TRUE),
  automotive = median(categories$automotive, na.rm = TRUE),
  art = median(categories$art, na.rm = TRUE),
  religion = median(categories$religion, na.rm = TRUE),
  beauty = median(categories$beauty, na.rm = TRUE),
  parenting = median(categories$parenting, na.rm = TRUE),
  dating = median(categories$dating, na.rm = TRUE),
  school = median(categories$school, na.rm = TRUE),
  personal_fitness = median(categories$personal_fitness, na.rm = TRUE),
  fashion = median(categories$fashion, na.rm = TRUE),
  small_business = median(categories$small_business, na.rm = TRUE),
  spam = median(categories$spam, na.rm = TRUE),
  adult = median(categories$adult, na.rm = TRUE)
)
medians
```

```
## $chatter
## [1] 3
##
## $current_events
## [1] 1
##
## $travel
## [1] 1
##
## $photo_sharing
## [1] 2
##
## $uncategorized
## [1] 1
##
## $tv_film
## [1] 1
##
## $sports_fandom
## [1] 1
##
## $politics
## [1] 1
##
## $food
## [1] 1
##
## $family
```

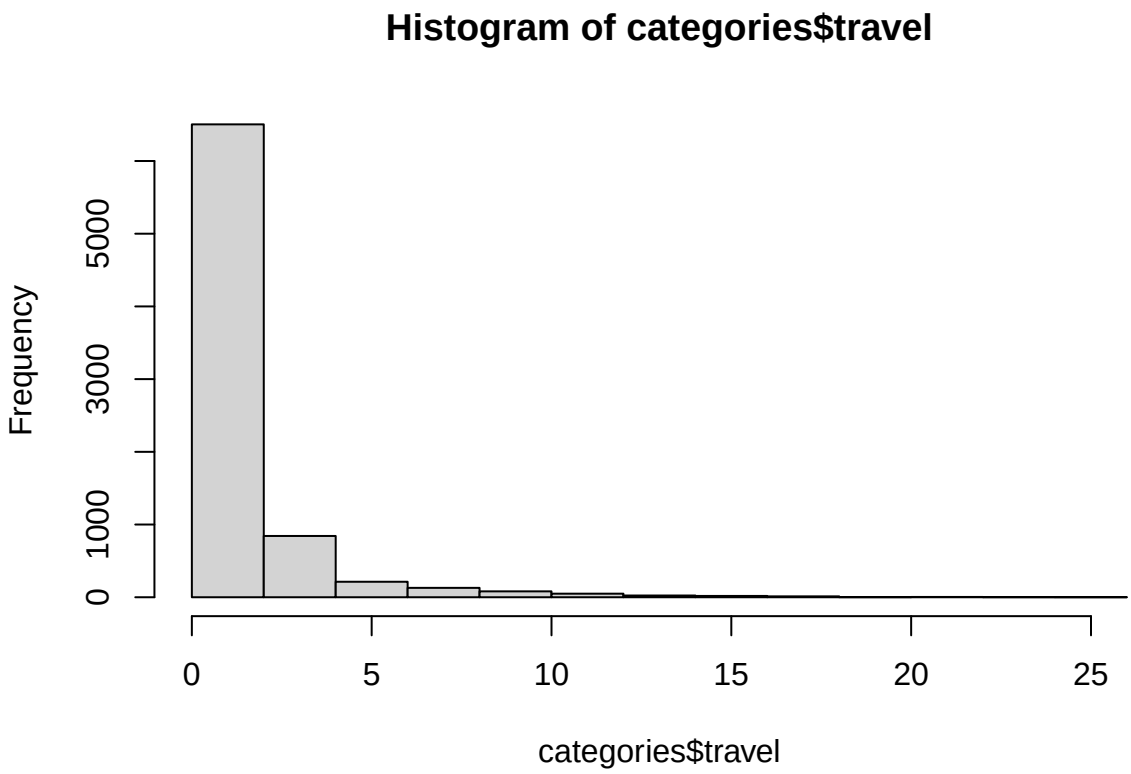
```
## [1] 1
##
## $home_and_garden
## [1] 0
##
## $music
## [1] 0
##
## $news
## [1] 0
##
## $online_gaming
## [1] 0
##
## $shopping
## [1] 1
##
## $health_nutrition
## [1] 1
##
## $college_uni
## [1] 1
##
## $sports_playing
## [1] 0
##
## $cooking
## [1] 1
##
## $eco
## [1] 0
##
## $computers
## [1] 0
##
## $business
## [1] 0
##
## $outdoors
## [1] 0
##
## $crafts
## [1] 0
##
## $automotive
## [1] 0
##
## $art
## [1] 0
##
## $religion
## [1] 0
##
## $beauty
```

```
## [1] 0
##
## $parenting
## [1] 0
##
## $dating
## [1] 0
##
## $school
## [1] 0
##
## $personal_fitness
## [1] 0
##
## $fashion
## [1] 0
##
## $small_business
## [1] 0
##
## $spam
## [1] 0
##
## $adult
## [1] 0
```

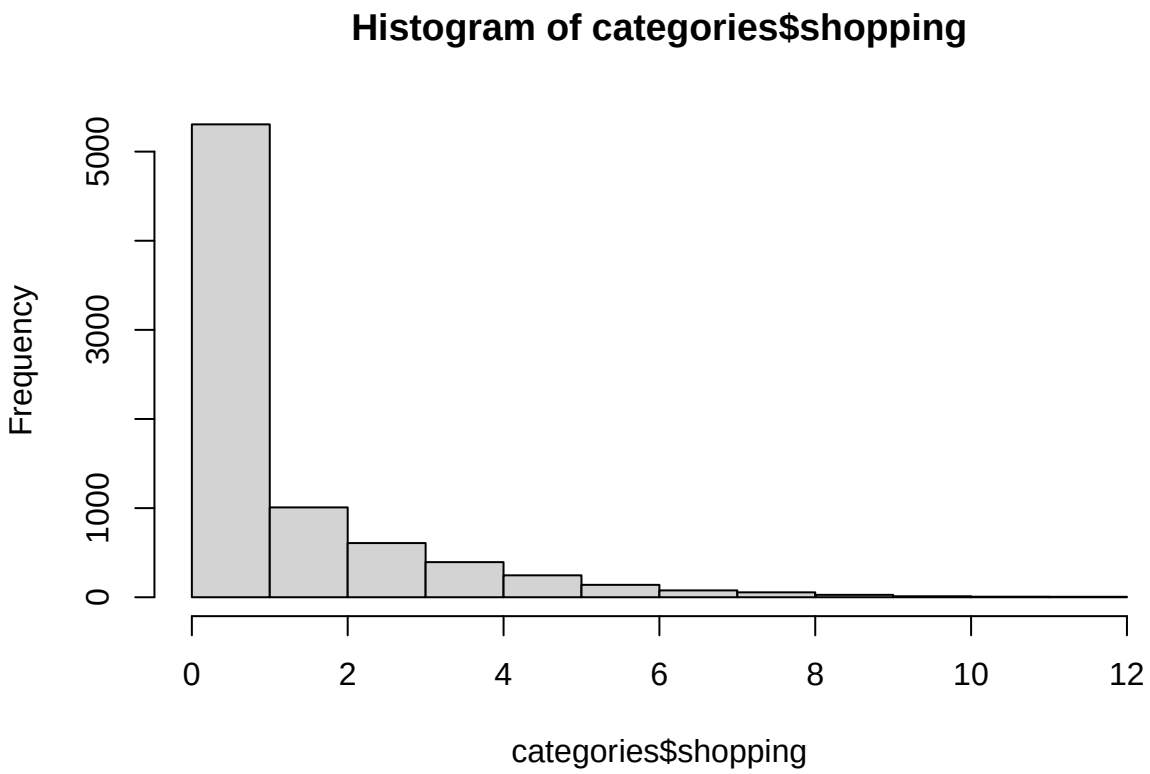
From this output we see that chatter, current_events, travel, photo_sharing, tv_film, sports_fandom, politics, food, family, shopping, health_nutrition college_uni, and cooking (also the uncategorized category, which is irrelevant in this case) are all columns that have medians that are greater than zero. Within the week span that this sample was taken, NutrientH20's customers are talking about topics that fall into these categories. By looking at a few hisograms of each category, the data is very left skew, meaning that most users are not posting anything about these categories (0 posts). The median gives us a good idea which topics are actually being talked about by the followers.

Here are some histograms that demonstrate the skew:

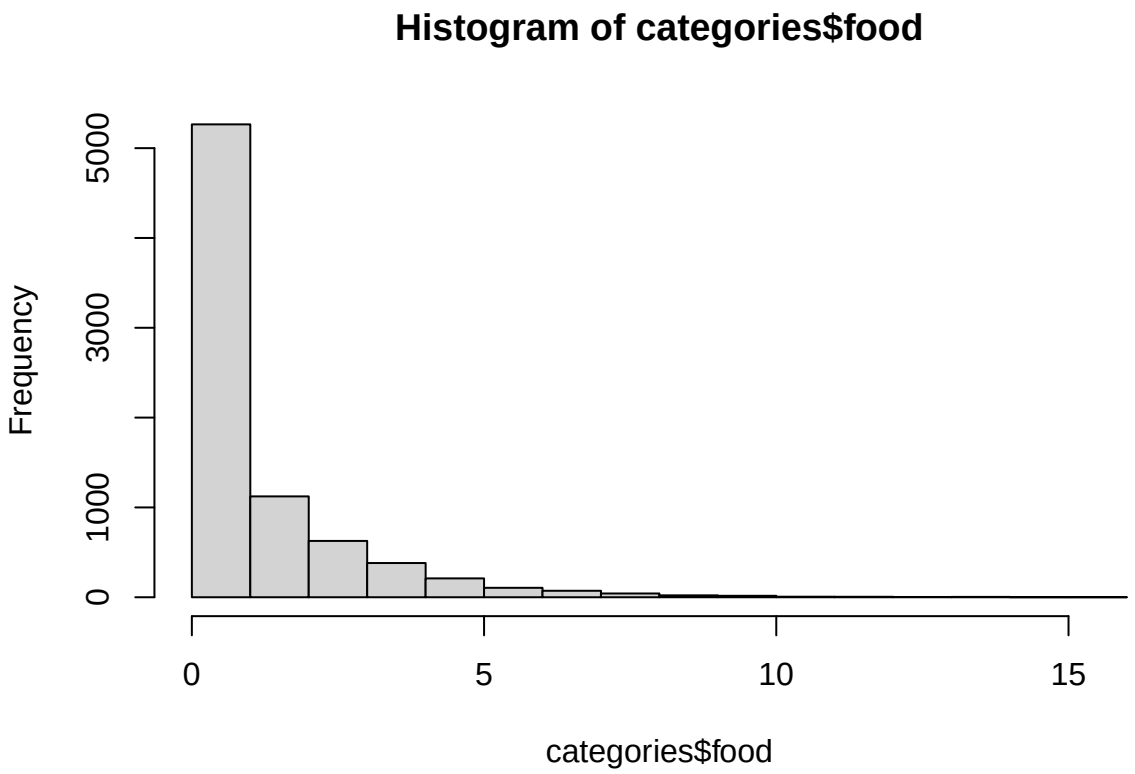
```
hist(categories$travel)
```




```
hist(categories$shopping)
```




```r
hist(categories$food)
```



Next, we will perform Principal Component Analysis (PCA) on relevant categories to reduce dimesnionality and use those components to do hierarchical clustering. Doing this ensures that clustering is based on the most significant features of the data, making it more robust and interpretable. A brief explantaion about how PCA and hierarchical clustering work:

PCA - Reduces the complexity of the data, which can help in visualizing patterns and relationships. Simplifying the data can make clustering algorithms more effective and easier to interpret.

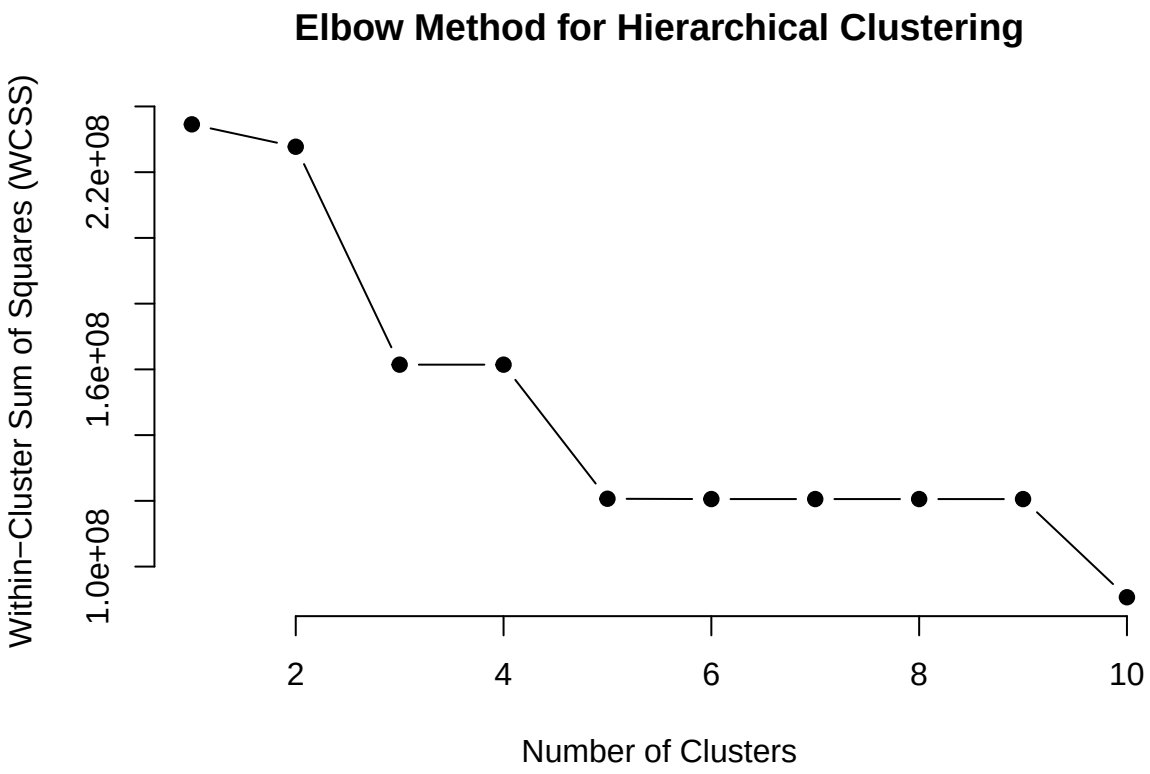
Hierachical Clustering- Directly segments your data into clusters, making it clear which individuals belong to which segment.Also, it does not require assumptions about the number of clusters ahead of time. We decided the optimal number of clusters based on the elbow plot.

```
library(dplyr)
```

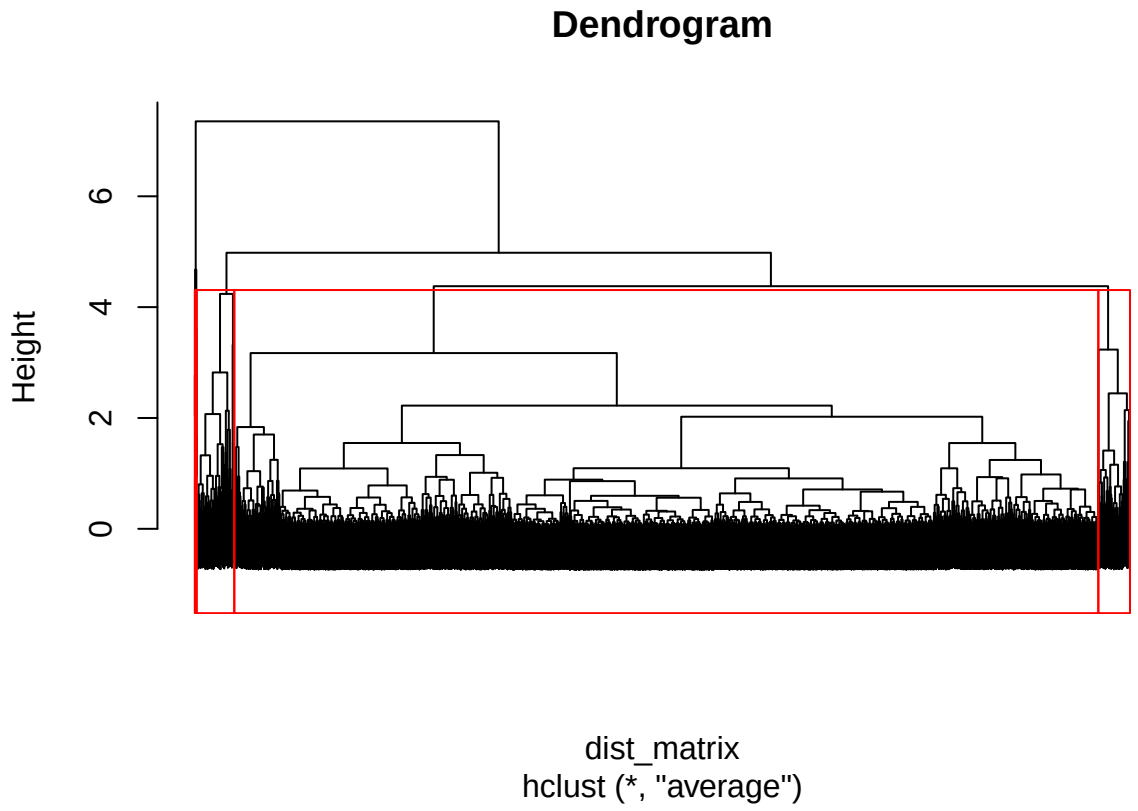
```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(stats)
library(ggplot2)
library(ggdendro)
library(reshape2)
```

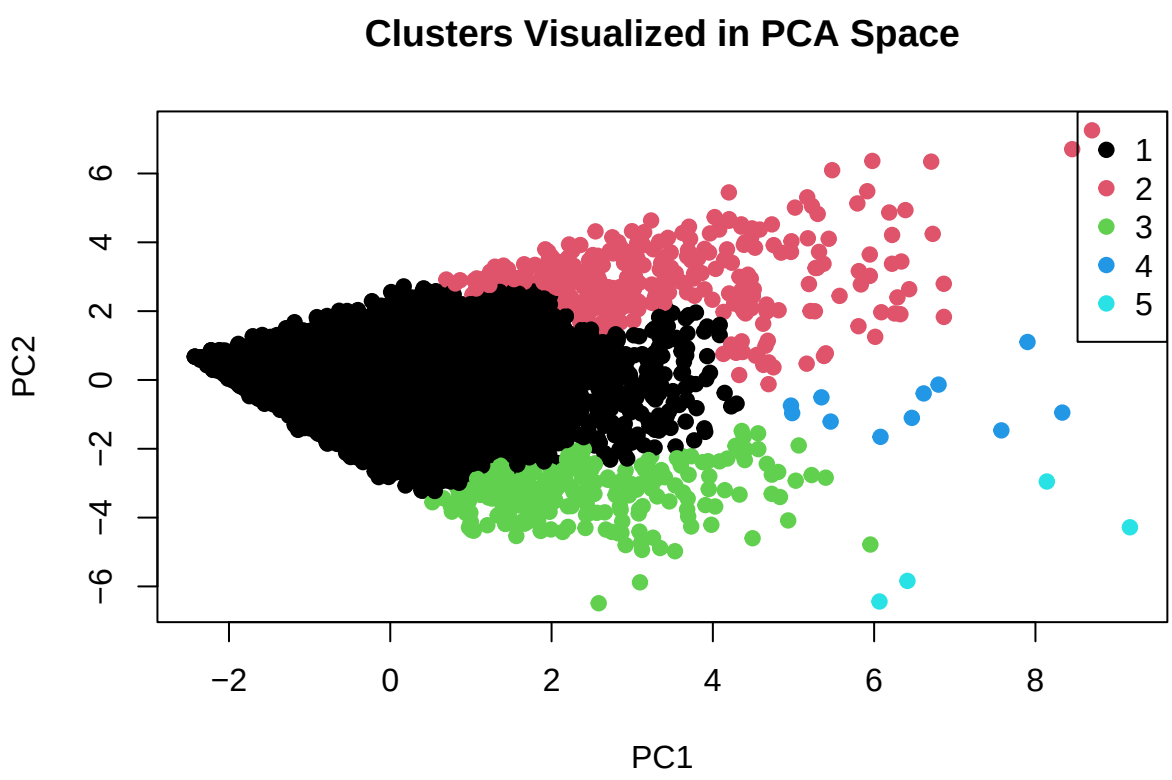
```r
relevant <- categories[, c('current_events', 'travel', 'photo_sharing', 'tv_film',
                           'sports_fandom', 'politics', 'food', 'family',
                           'shopping', 'health_nutrition', 'college_uni', 'cooking')]


pca_result <- prcomp(relevant, scale. = TRUE)
pca_data <- data.frame(pca_result$x[, 1:2])  # Extract the first two principal components

# compute Euclidean distance on PCA-reduced data
dist_matrix <- dist(pca_data, method = "euclidean")


hc <- hclust(dist_matrix, method = "average")

# determine optimal number of clusters using elbow plot
wcss <- numeric()
for (k in 1:10) {
  clusters <- cutree(hc, k = k)
  wcss[k] <- sum(sapply(unique(clusters), function(cluster) {
    cluster_data <- pca_data[clusters == cluster, ]
    sum(dist(cluster_data)^2)
  }))
}

plot(1:10, wcss, type = "b", pch = 19, frame = FALSE,
     xlab = "Number of Clusters",
     ylab = "Within-Cluster Sum of Squares (WCSS)",
     main = "Elbow Method for Hierarchical Clustering")
```




```r
plot(hc, labels = FALSE, main = "Dendrogram")
rect.hclust(hc, k = 5, border = "red")
```



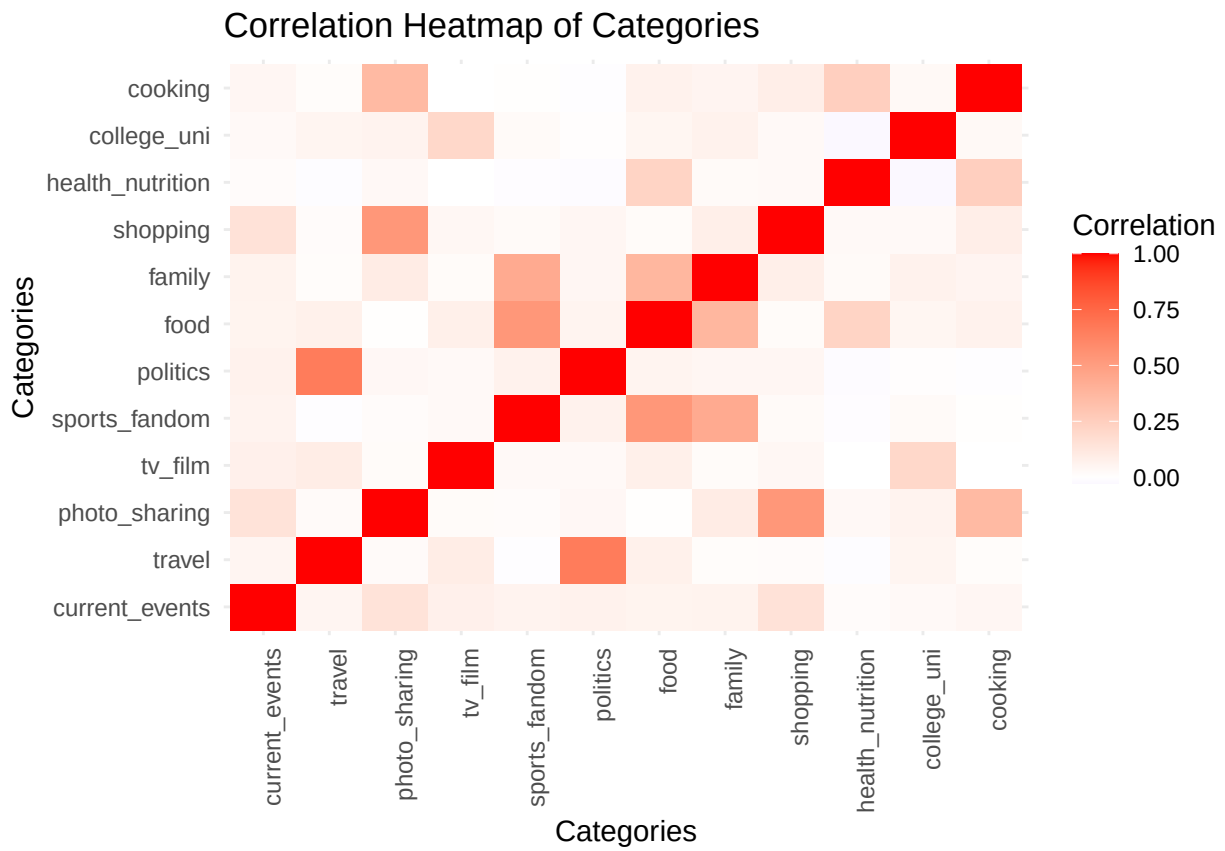

```
clusters <- cutree(hc, k = 5)


plot(pca_data[, 1], pca_data[, 2], col = clusters, pch = 19,
     xlab = "PC1", ylab = "PC2",
     main = "Clusters Visualized in PCA Space")
legend("topright", legend = unique(clusters), col = unique(clusters), pch = 19)
```



By examining the elbow plot, we identified the optimal number of clusters as 5. This is where adding more clusters does not significantly reduce the within-cluster sum of squares (WCSS). When we visualize the dendrogram with rectangles, we see the clusters identified by the hierarchical clustering. Each rectangle represents a distinct cluster. The clusters Visualization in PCA space shows how different market segments are distributed. Each color in the plot represents a different market segment, which helps to identify distinct audience profiles.

Another step I took was examining the correlation matrix between variables. By analyzing the correlation coefficients, we can identify which categories are closely related or strongly correlated. The heatmap visually represents this correlation matrix, with color gradients indicating the strength of the correlations. I wanted to look at this because understanding which variables are correlated can provide better insights into the structure of the clusters identified by hierarchical clustering.

```
relevant_names <- c('current_events', 'travel', 'photo_sharing', 'tv_film',
                    'sports_fandom', 'politics', 'food', 'family',
                    'shopping', 'health_nutrition', 'college_uni', 'cooking')


correlation_matrix <- cor(categories[, relevant_names], use = "complete.obs")
correlation_melted <- melt(correlation_matrix)

ggplot(correlation_melted, aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, name = "Correlation") +
  theme_minimal() +
  labs(title = "Correlation Heatmap of Categories", x = "Categories", y = "Categories") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



From the correlation matrix, the top 3 most highly correlated variables pairs are: 1. Politics and Travel 2. Shopping and Photo Sharing 3. Sports Fandom and Food

One important thing that I wanted to point out in the correlation matrix is that Sports Fandom and Food, Family and Food, and Family and Sports fandom all have very similar correlations. This could be a possible market segment for Nutrient H20 to tap into! Familes who are into sports, they could be throwing nieghborhood watch parties to all get together and try new receipies that they see online and watch the big game of the weekend! NutrientH20 could consider creating a few posts about "gameday bites" and give suggestions for creative snacks to bring to the watch parties.

Another possible segment to tap into is The followers who are into politics and traveling. I think this segment is the working professionals who want to voice their opinion on politics and who have extra cash to travel! A way to target them could be by posting about various cities that host political or social movement rallies/events.

A third possible segment are the shoppers and photo sharers. A good way to connect with this segment could be by running campaigns that encourage photo sharing of purchases, such as contests or hashtag challenges, to increase brand visibility and engagement!