SocialM

```
sm_clean = select(social, -matches("uncategorized"))
sm_cleaned = subset(sm_clean, adult < 5)
```

The Median number of adult tags is 5, so we decide to take out data points containing more than 5 adult tweets per week. also we found out for spam, it ranges from 0-2 with mean around 0.3, so we conclude that the filter has done a pretty good job already. We not gonna touch the spam column or drop anything. Becasue uncategorized tags don’t help in segmentation, we drop that whole column.

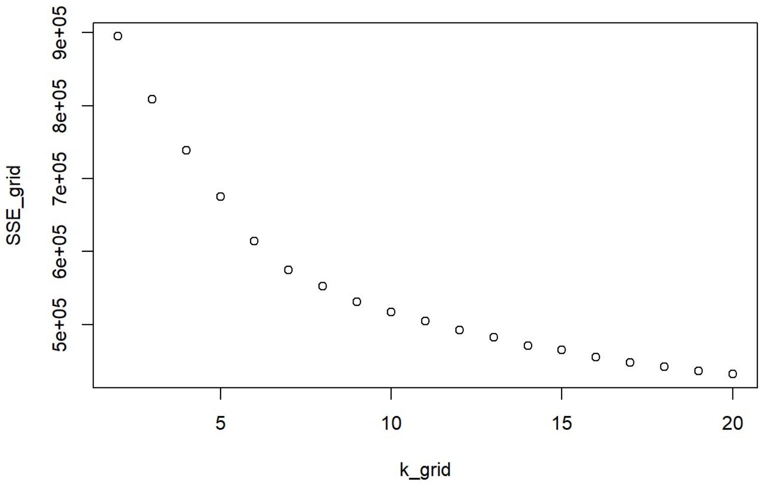
```
set.seed(1)
k_grid = seq(2, 20, by=1)
SSE_grid = foreach(k = k_grid, .combine='c') %do% {
  cluster_k = kmeans(sm_cleaned, k, nstart=50)
  cluster_k$tot.withinss
}
```

```
## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations

## Warning: did not converge in 10 iterations
```

```
plot(k_grid, SSE_grid)
```



After seeing the results from k-means, we decide we will pick k = 7

```
k = 7
cluster_k = kmeans(sm_cleaned, k, nstart=50)
```

K-means clustering with 7 clusters of sizes 917, 583, 475, 3533, 544, 376, 1164

```
for(i in 1:k) {
  index_temp = which(cluster_k$cluster == i)
  df_temp = sm_cleaned[c(index_temp), ]
  assign(paste("ddat",i,sep="_"),df_temp)
  assign(paste("index",i,sep="_"),index_temp)
}
```

Now we have 7 different df for each cluster

###A quick try on Hie-clustering

```
sm_distance_matrix = dist(social, method='euclidean')
hmin = hclust(sm_distance_matrix, method='complete')
cluster3 = cutree(hmin, k=10)
summary(factor(cluster3))
```

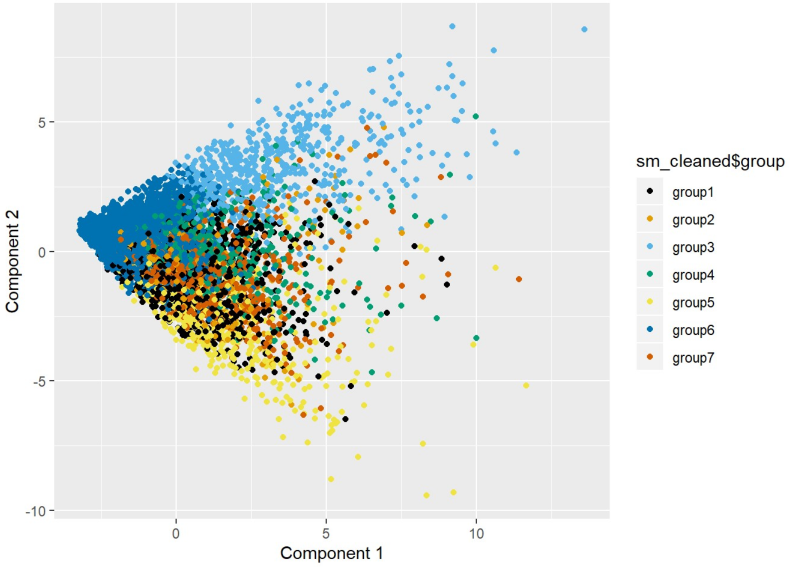
```
##    1    2    3    4    5    6    7    8    9   10
##  347 6155  489   70  110  420  218   31   22   20
```

We have tried method “complete” and “single”, also k = 5-10. The results have been disappointing because there’s always one big group that takes over majority of data points. We are showing the best one we’ve got. So we decide to stick with results from K-means for now.

###PCA We need to label each data with a group number (7 different segmentations!) for later The group assignments are based on results from k means
```r sm_cleaned[‘group’] = NA
sm_cleaned\(group[index_1] = 'group1' sm_cleaned\)group[index_2] = ‘group2’ sm_cleaned\(group[index_3] = 'group3' sm_cleaned\)group[index_4] = ‘group4’ sm_c
sm_cleaned\)group[index_6] = ‘group6’ sm_cleaned$group[index_7] = ‘group7’ ```

```
r z = sm_cleaned[,1:35] pc1 = prcomp(z, scale.=TRUE)
```

```
r loadings = pc1$rotation scores = pc1$x qplot(scores[,1], scores[,2], color = sm_cleaned$group, xlab = 'Component 1', ylab = 'Component 2') +scal
```



5
0
-5
-10
Component 2
0
5
10
Component 1
sm_cleaned$group
group1
group2
group3
group4
group5
group6
group7

It’s obvious that there are some distinct groups such as 3, 5 and 6. Let’s do some head & tail to analyze what component 1 & 2 represents.

```
r o1 = order(loadings[,1], decreasing = TRUE) colnames(z)[head(o1,10)]
```

```
##  [1] "religion"      "food"           "parenting"      "sports_fandom" ##  [5] "school"         "family"         "beauty"         "crafts" ##  [9] "
```

```
r colnames(z)[tail(o1,10)]
```

```
##  [1] "health_nutrition" "home_and_garden"  "dating" ##  [4] "current_events"   "art"              "tv_film" ##  [7] "college_uni"      "online_
```

```
r colnames(z)[o1[20:25]]
```

```
## [1] "shopping"       "sports_playing" "chatter"        "music" ## [5] "small_business" "travel"
```

```
r o2 = order(loadings[,2], decreasing = TRUE) colnames(z)[head(o2,10)]
```

```
##  [1] "religion"      "sports_fandom" "parenting"      "food" ##  [5] "school"        "family"        "news"           "automotive" ##  [9] "adult
```

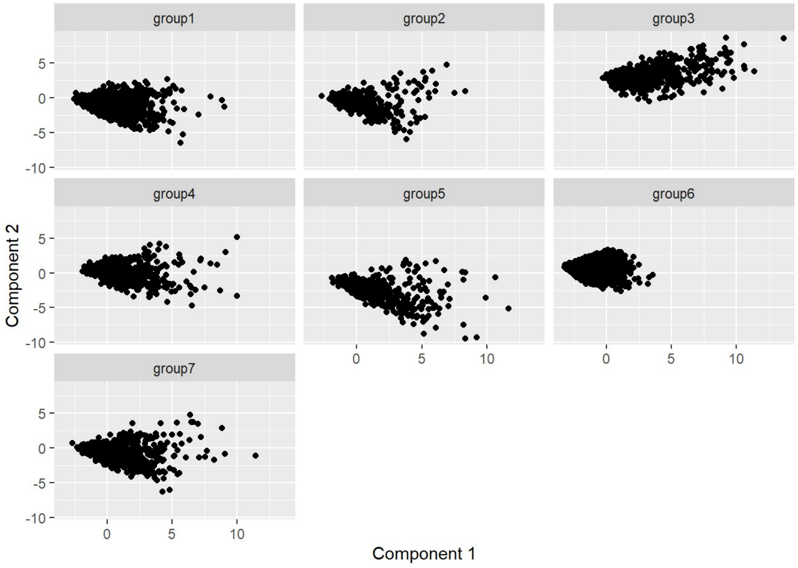
```
r colnames(z)[tail(o2,10)]
```

```
##  [1] "outdoors"         "health_nutrition" "personal_fitness" ##  [4] "music"            "chatter"          "beauty" ##  [7] "shopping"
```

```
r colnames(z)[o2[20:25]]
```

```
## [1] "online_gaming"  "eco"            "small_business" "sports_playing" ## [5] "business"       "college_uni"
~~~

Seperate them into individual plots to see lcearly
```r sm.group = sm_cleaned$group
qplot(scores[,1], scores[,2], facets=~sm.group, xlab = ‘Component 1’, ylab = ‘Component 2’) ```



Group3 is closer to head of PC1 and PC2 than any oth

“school” “family” “beauty” “crafts” “cooking” “fashion”
“religion” “sports_fandom” “parenting” “food” “school” “family” “news” “automotive” “adult” “crafts”
We conclude that this group should be more family oriented, older adults who’s already married.
~~~

Group6 is close to PC1 tail and PC2 middle: “dating” “current_events” “art” “tv_film” “college_uni” “online_gaming” “adult” “spam” “computers” “travel” “tv_film” “dating” “current_events” “online_gaming”

# We conclude this group is close to college students, young single adults.

Group 5 is close to tail of PC 2: “health_nutrition” “personal_fitness” “music” “chatter” “beauty” “shopping” “fashion” “cooking” “photo_sharing”

# We conclude this group as working woman, or our “mum” group

Other clusters that were kinda in the middles, pretty similar:

They do contain some unique features that stood out, such as “business”,“small business”, and “news” that were not as common as other groups. So we identify these groups as various working class, hard workers, or entrepreneur.